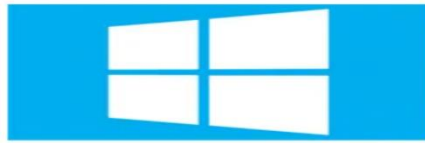


Systèmes d'Exploitation et Unix



Chapitre 1:

Introduction aux systèmes d'exploitation

Motivations

- Les systèmes d'exploitation (SE) au quotidien:
 - Machines à laver, consoles de jeux, Smartphones, ordinateurs, SmartTV...
 - «Le système démarre», «Le système a planté» ,« version 3.0.1 du système », « bug système » ,
- Académique/Professionnel:
 - Génie Logiciel: le logiciel s'appuie sur un SE
 - Administration réseau: Les machines du réseaux (PCs,routeurs, serveurs, ...) possèdent des SE!
- Mais encore: Les SE ont été particulièrement importants dans le développement de l'informatique, à côté de l'évolution technologique des ressources matérielles.

Motivations



- L'étude des SE permet d'accéder à des formations avancées (Systèmes distribués, virtualisation, cloud computing, Big Data ...).
- *un ordinateur se compose d'éléments matériels « Hardware » et d'éléments logiciels « Software ».*

Cependant, comment coexistent-ils?

Quel est le rôle joué par le SE?

Comment réalise-t-il ce rôle?

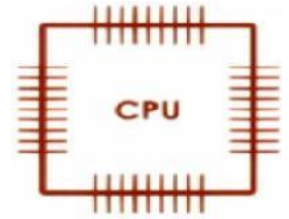
- Un SE est fortement lié aux ressources matérielles sur lesquelles il s'exécute.
- Il **exploite l'ensemble des instructions** exécutables par l'ordinateur (càd le matériel) et **gère ses ressources**.

Chapitre 1: introduction aux SE

1. Rappel: point sur le matériel
2. Rôles d'un SE
3. Définition d'un SE
4. Structure interne des SE
5. Le noyau
6. Quelques familles de SE
7. La notion de multitâches multi-utilisateurs

1. Rappel: point sur le matériel
2. Rôles d'un SE
3. Définition d'un SE
4. Structure interne des SE
5. Le noyau
6. Quelques familles de SE
7. La notion de multitâches multi-utilisateurs

1. Rappel: point sur le matériel!



1) Processeur:

L'Unité Centrale de Traitement (UCT) ou processeur central (**CPU**), est « le cerveau » de l'ordinateur qui interprète et exécute les instructions du programme situées en mémoire centrale.

- L'UCT est composée de l'**Unité arithmétique et logique** (UAL) et de l'**Unité de commande** (ou de contrôle).
- L'UAL effectue les opérations arithmétiques et logiques.
- L'Unité de commande dirige le fonctionnement de toutes les autres unités: UAL, mémoire, entrées / sorties, etc., en leur fournissant les signaux de cadence (l'horloge) et de commande.

1. Rappel: point sur le matériel!

2)Mémoire:

Typical access time

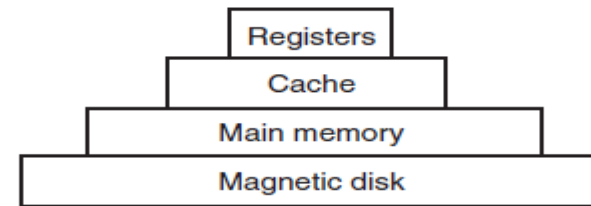
Typical capacity

1 nsec

2 nsec

10 nsec

10 msec



<1 KB

4 MB

1-8 GB

1-4 TB

•Les registres:

- des petites unités de mémoire situées directement à l'intérieur du processeur (**CPU**).
- Ils jouent un rôle crucial dans le fonctionnement de l'ordinateur en permettant un accès ultra-rapide aux données que le processeur doit traiter immédiatement
- Ils sont les types de mémoire les plus rapides dans l'architecture de l'ordinateur, bien plus rapides que la RAM et la mémoire cache, car ils sont intégrés directement dans le CPU.

•La mémoire cache:

- Placée entre le processeur (CPU) et la RAM pour stocker les données ou instructions les plus souvent utilisées par le processeur.
- mémoire très rapide, de petite taille.

•La mémoire vive:

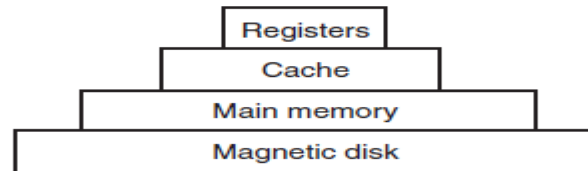
- RAM (Random Access Memory), on **peut accéder instantanément à n'importe quelle espace mémoire**.
- Volatile (Les données sont perdues lorsque l'ordinateur est éteint.)
- Contient les données et les instructions des applications en cours.
- Mémoire rapide, de taille plus importante et à prix moyen.

1. Rappel: point sur le matériel!

2)Mémoire:

Typical access time

1 nsec
2 nsec
10 nsec
10 msec



Typical capacity

<1 KB
4 MB
1-8 GB
1-4 TB

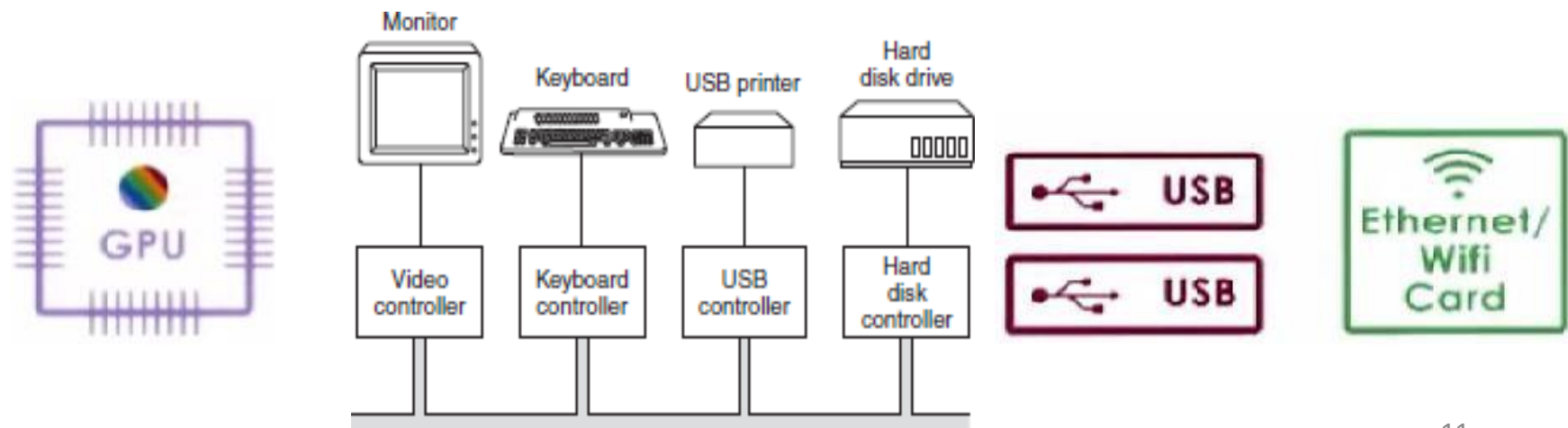


- La mémoire morte:
 - ROM (Read Only Memory) **mémoire en lecture seule**.
 - Permanente, contenant des microprogrammes enregistrés à l'usine sur des puces électroniques de la carte mère, contenant les routines de démarrage de l'ordinateur BIOS (Basic Input Output System).
 - C'est une mémoire non volatile, rapide et à bas coût.
- Le disque:
 - sert principalement à stocker les données d'une manière non-volatile.
 - L'accès est mécanique, introduisant un délai important en lecture/écriture relativement à la RAM.
 - Sert éventuellement à « étendre » la RAM.
 - Un espace plus important et à coût bas (selon les technologies).

1. Rappel: point sur le matériel!

3) Les périphériques d'entrée/sortie (E/S):

- Permettent le dialogue (échange d'informations) avec ce qui se trouve à l'extérieur de la machine.
- Ils se composent généralement de deux parties:
 - **Le matériel** (physique).
 - **Le contrôleur**: une puce ou un ensemble de puces qui contrôle physiquement le périphérique. Présente au système d'exploitation une interface « simple », nommée Driver, pour recevoir les commandes et retourner leurs résultats.



1. Rappel: point sur le matériel!

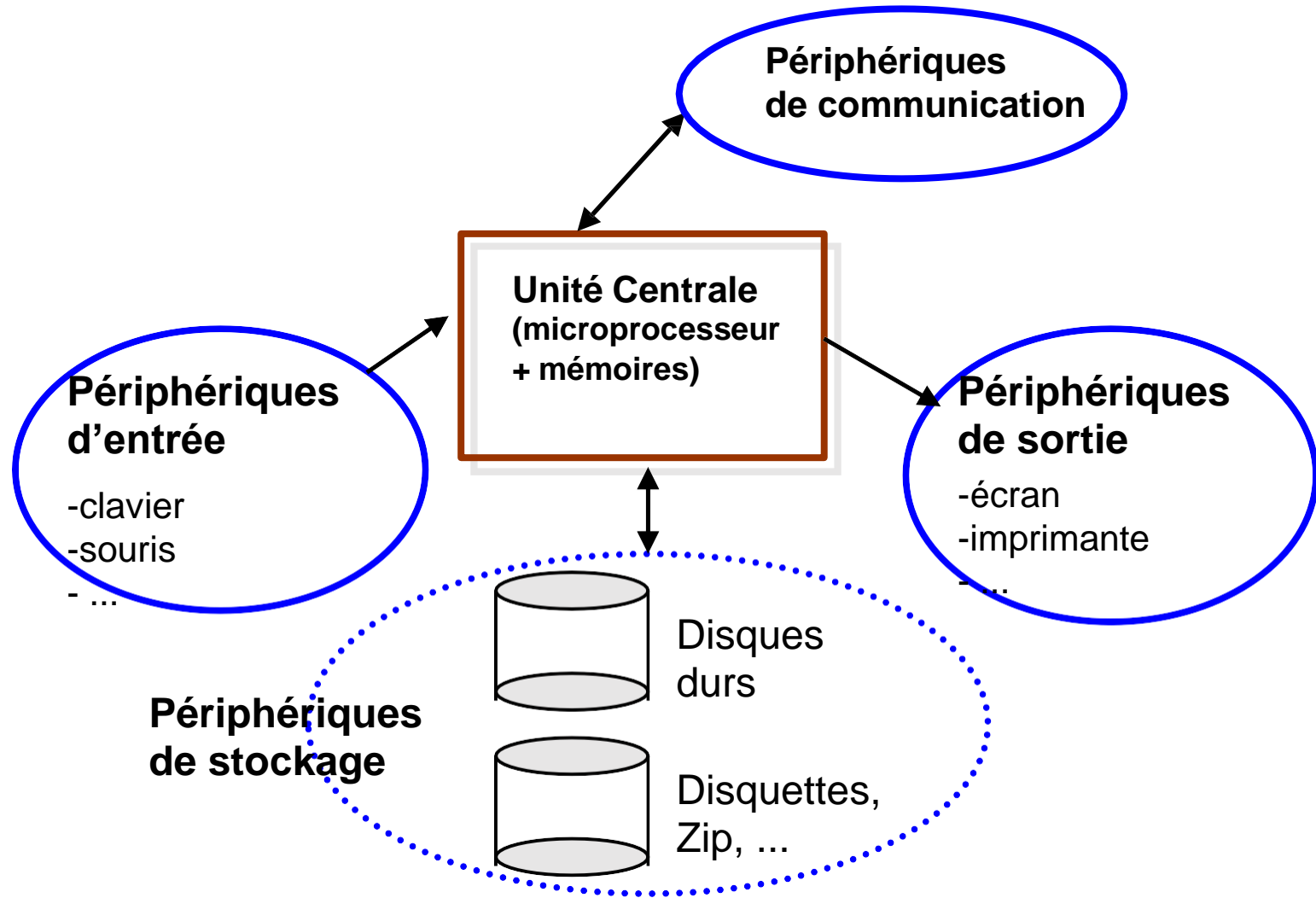
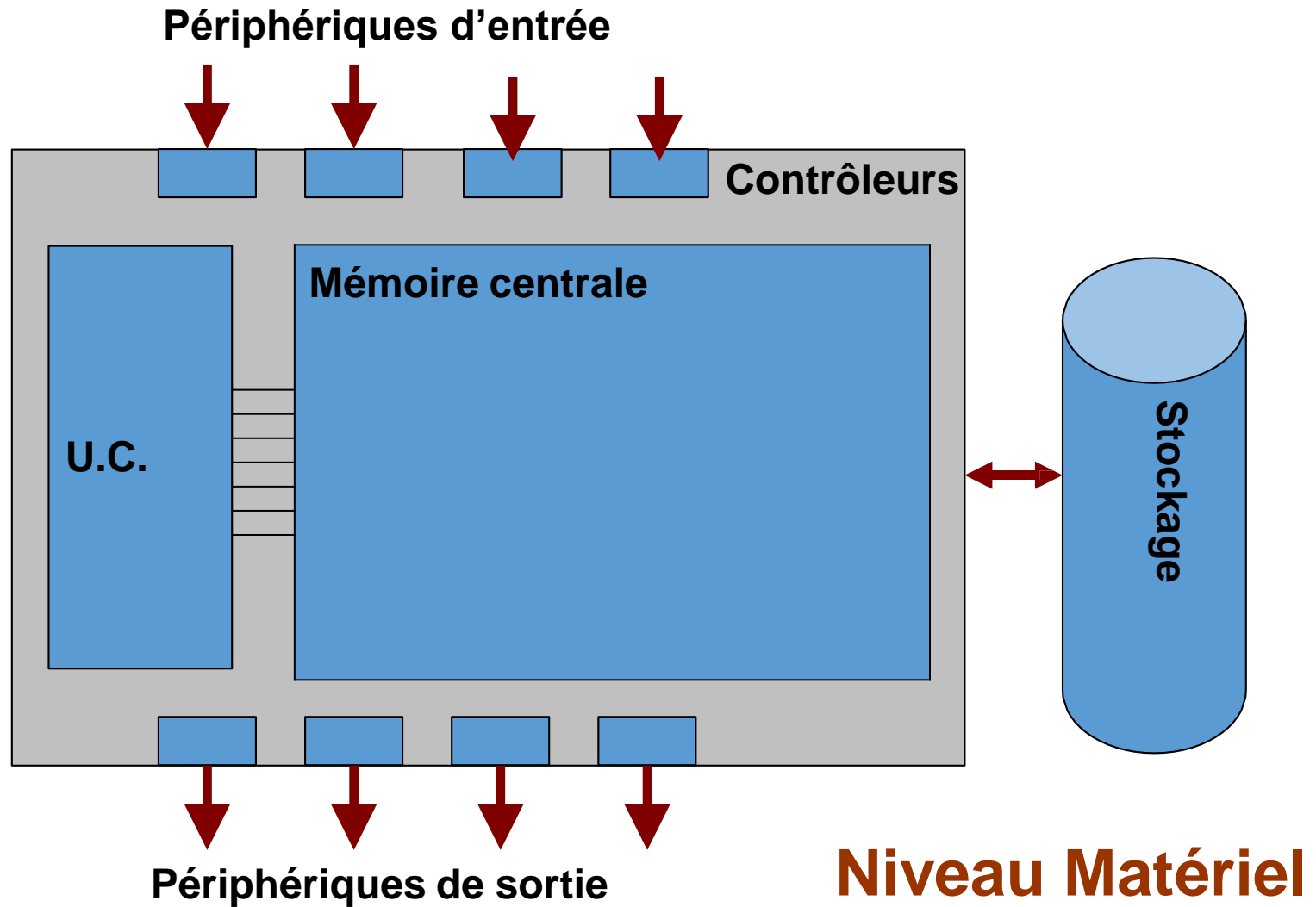
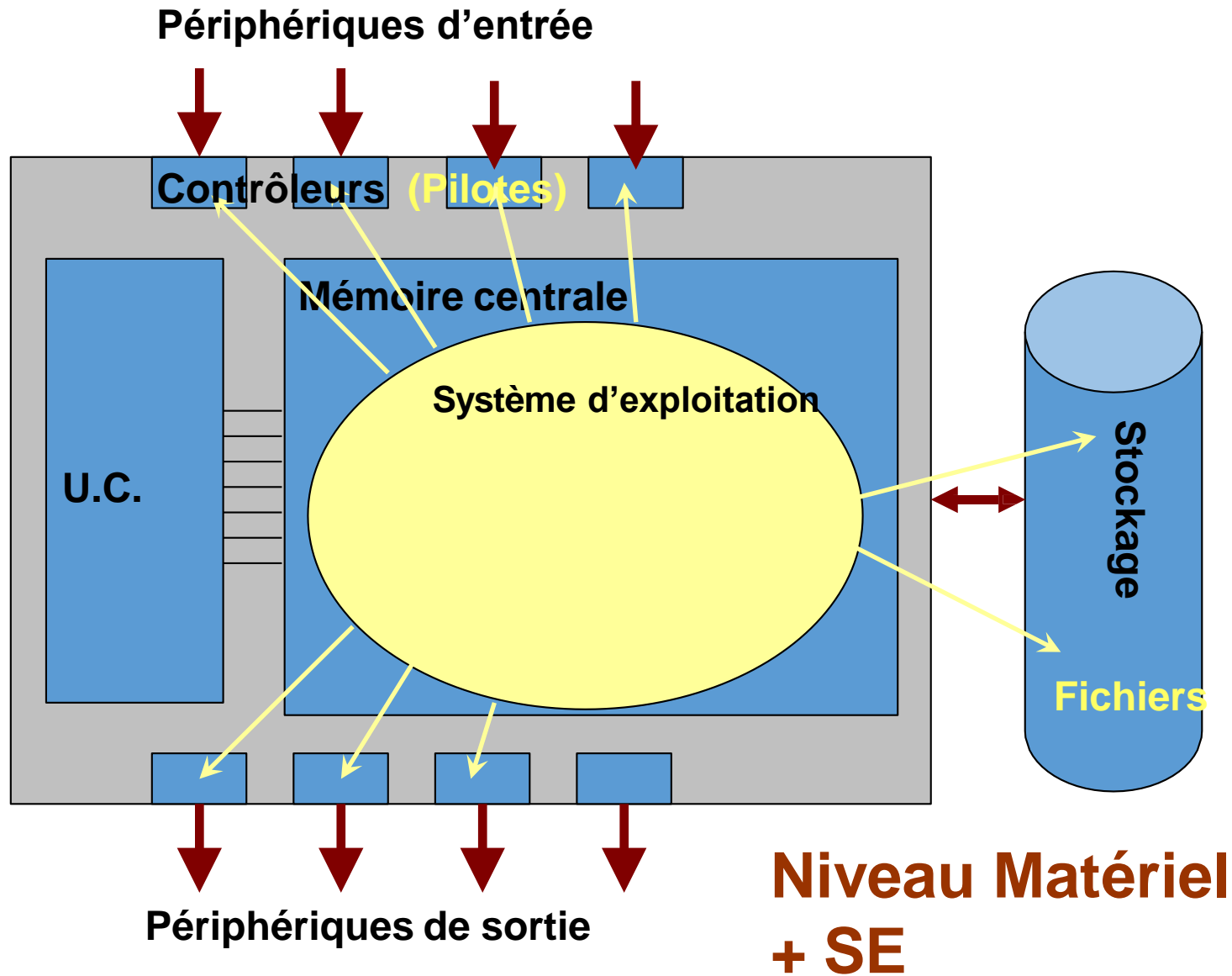


Schéma matériel général

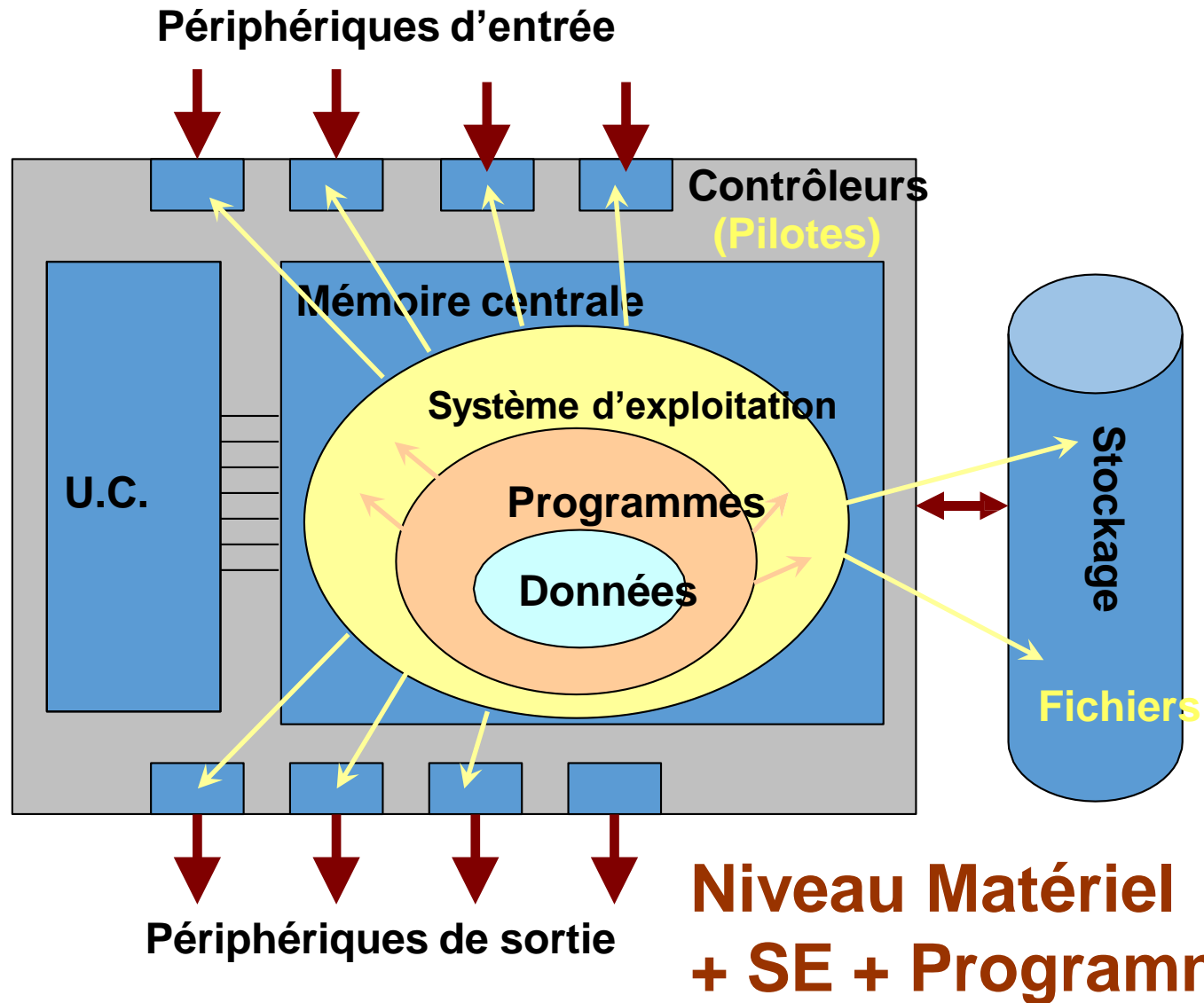
1. Rappel: point sur le matériel!



1. Rappel: point sur le matériel!



1. Rappel: point sur le matériel!



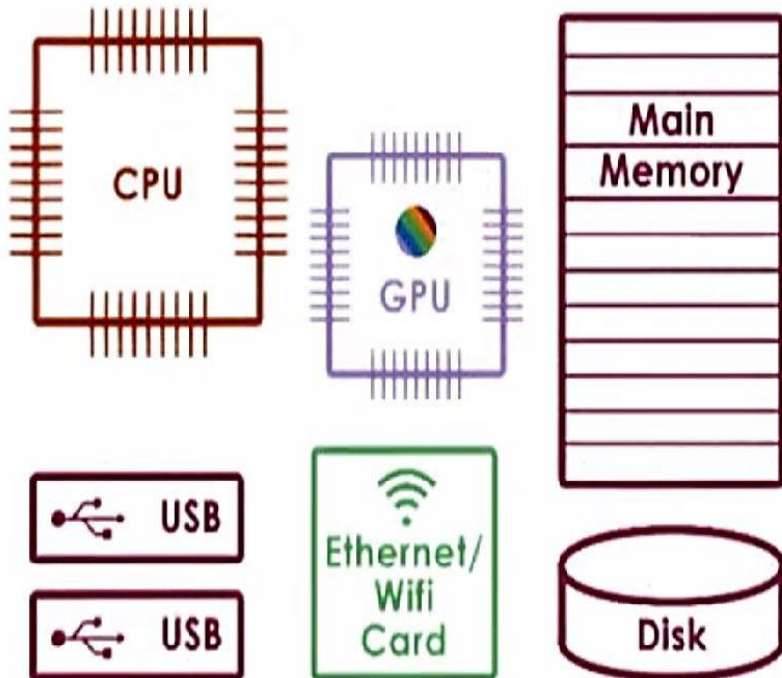
1. Rappel: point sur le matériel
2. **Rôle d'un SE**
3. Définition d'un SE
4. Structure interne des SE
5. Le noyau
6. Quelques familles de SE
7. La notion de multitâches multi-utilisateurs

2. Rôle d'un SE

Définition : ensemble de programmes de gestion du système qui permettent de gérer les éléments fondamentaux de l'ordinateur: Le matériel , les logiciels , la mémoire - les données , les réseaux.

Rôles majeurs:

- **Cache la complexité du hardware**, que ce soit pour les applications ou pour les développeurs: diverses technologies de stockage, de transmission,
- **Gère les ressources:** allocation de mémoire, ordre d'exécution.
- **Isolation et protection:** des programmes, des utilisateurs, ...



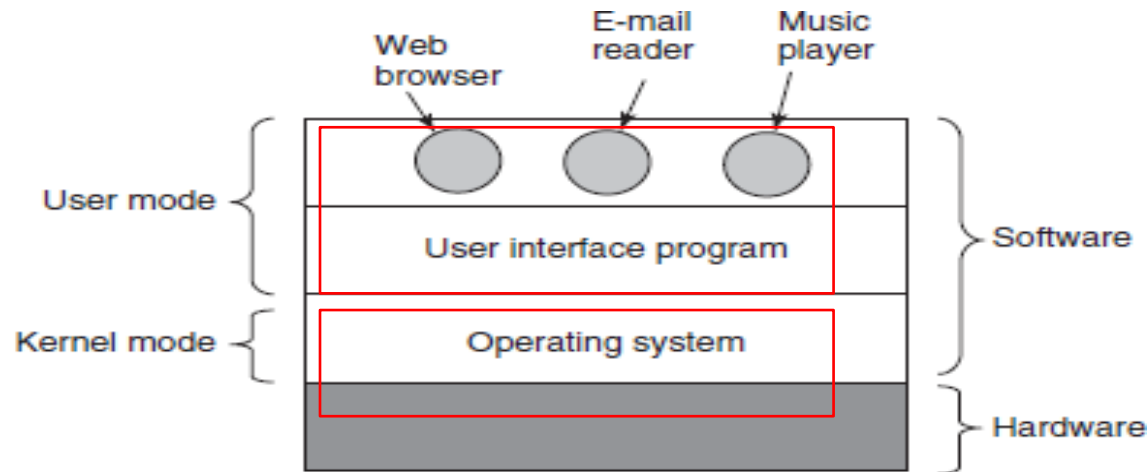
2. Rôle d'un SE

- **D'une manière simple:** simplifier l'interaction avec le matériel puis contrôler et gérer son utilisation.
- **D'une manière plus soutenue (académique):**
 - **La gestion des processus:** un processus est un programme en cours d' exécution. Le SE doit gérer l'allocation de ressources aux processus en proposant à chacun un environnement dans lequel il peut s'exécuter en toute sécurité.
 - **La gestion de la mémoire :** Le SE doit gérer l'allocation de mémoire aux processus et contrôler physiquement les emplacements auxquels peut accéder un processus.
 - **La gestion des périphériques:** Le SE doit gérer les périphériques afin de leur permettre d'être partagés de manière efficace entre les processus.
 - **La gestion du système de fichiers :** Le SE doit gérer des structures de données permettant de créer, stocker, supprimer, lire (etc. ...) les informations et de les organiser dans des fichiers sur des mémoires secondaires (disque dur, CD-ROM, clé USB, disques SSD, etc.).
 - **La protection:** l'accès aux données doit être réglementé puisqu'il existe plusieurs utilisateurs et processus. Le SE doit garantir que les fichiers, les segments de mémoire, etc ... ne peuvent être utilisés que par les processus ayant obtenu l'autorisation appropriée.
 - **La sécurité:** Le SE doit posséder des mécanismes pour se défendre contre les attaques externes et internes. Elles comprennent les virus, des attaques de déni de service (Denial of Service: DoS).

1. Rappel: point sur le matériel
2. Rôles d'un SE
- 3. Définition d'un SE**
4. Structure interne des SE
5. Le noyau
6. Quelques familles de SE
7. La notion de multitâches multi-utilisateurs

3. définition d'un SE

- **Définition:** *Un système d'exploitation est une couche logicielle indispensable pour exploiter, d'une manière simple, les ressources matériels d'un ordinateur.*



- Deux modes de fonctionnement (en général):
 - le mode noyau (mode superviseur, privilégié): dispose d'un accès complet à tout le matériel et peut exécuter toutes les instructions que la machine est capable d'exécuter.
 - Le mode utilisateur: le reste du système s'exécute en mode utilisateur, dans lequel seul un sous-ensemble des instructions de la machine est exécutable.
 - Le programme d'interface utilisateur: le shell ou l'interface graphique, est le niveau le plus bas du logiciel en mode utilisateur, et permet à l'utilisateur de démarrer d'autres programmes, comme un navigateur Web, un lecteur de courrier électronique ou un lecteur de musique...

I. Généralités sur les SE

1. Rappel: point sur le matériel
2. Définition d'un SE
3. Rôles d'un SE
- 4. Structure interne des SE**
5. Le noyau
6. Quelques familles de SE
7. La notion de multitâches multi-utilisateurs

4. Structures internes de SE

- **Monolithique (d'un seul bloc) :**

- L'ensemble du SE s'exécute en un seul programme en mode noyau.
- Le SE est écrit comme une collection de procédures, reliées entre elles dans un seul programme binaire exécutable unique.
- **monotâche et en ligne de commande**, c'est-à-dire qu'il permettait d'exécuter une seule tâche à la fois et n'offrait pas d'interface graphique.

✓ **Implémentation simple.**

↓ **Difficile à maintenir!**

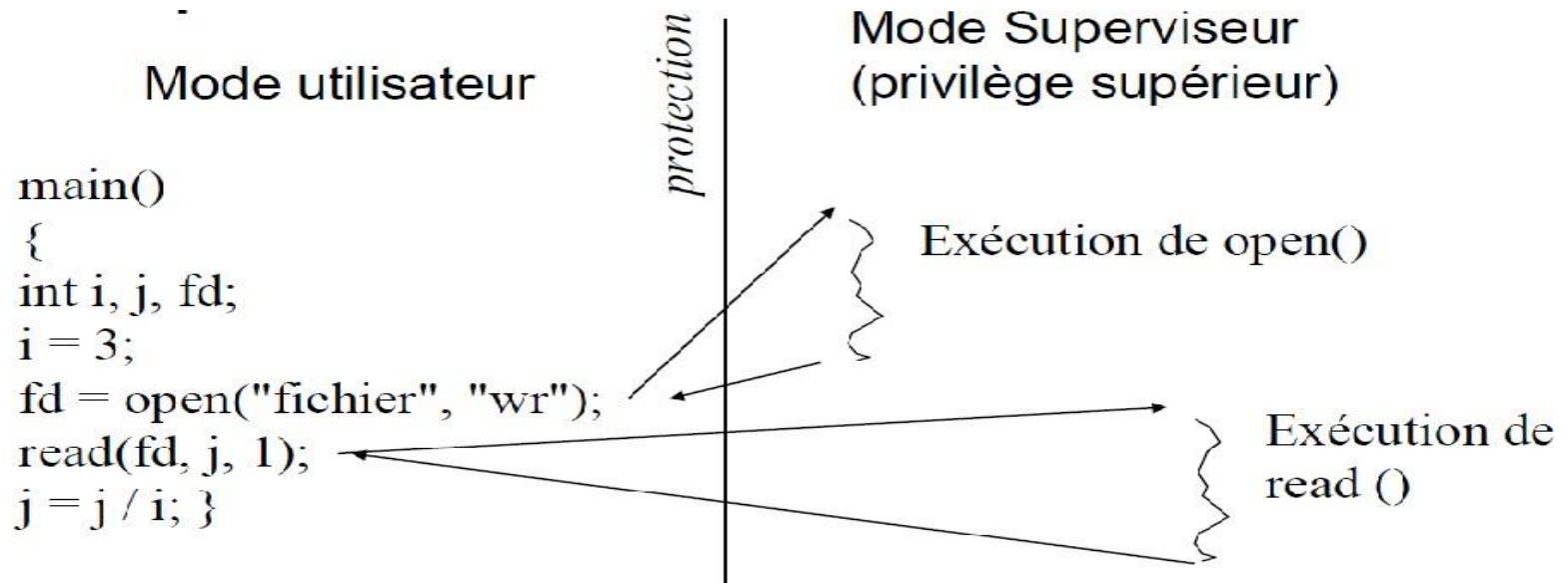
- MS-DOS est un exemple d'un tel système.

- **Systèmes à modes noyau et utilisateur:**

- Le SE démarre en mode noyau, ce qui permet d'initialiser les périphériques et de mettre en place des routines de service, et commute ensuite en mode utilisateur.
- En mode utilisateur, on utilise les **appels système** pour avoir accès à ce qui a été prévu par le système.
- **Appel système** : Lorsqu'une application en mode utilisateur souhaite lire un fichier, elle ne peut pas accéder directement au disque. Elle doit passer par un appel système qui transfère le contrôle au noyau, qui exécute ensuite la tâche requise (lecture du fichier) en mode noyau et renvoie les résultats à l'application en mode utilisateur.

4. Structures internes de SE

• Systèmes à modes noyau et utilisateur: Appel système



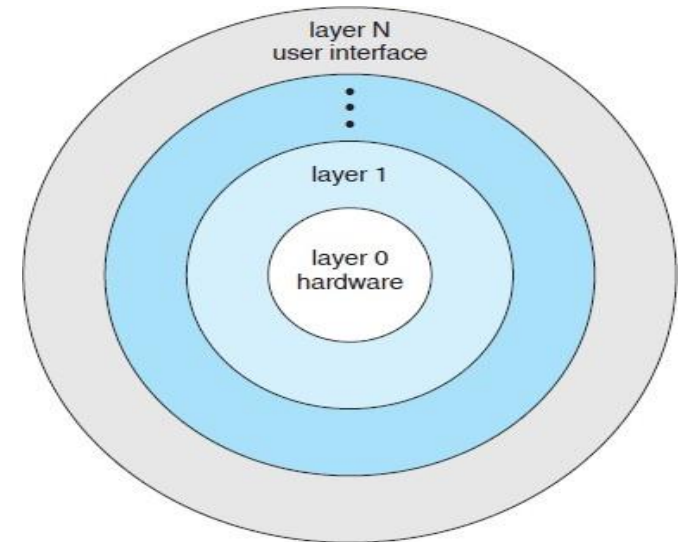
- 1) Le noyau reçoit l'appel système,
- 2) Vérifie qu'il s'agit d'une demande valable (en particulier du point de vue des droits d'accès),
- 3) Exécute,
- 4) Renvoie au mode utilisateur.

*Le noyau est généralement exécuté dans un espace mémoire séparé de l'espace des applications: **espace noyau**. Par opposition à **l'espace utilisateur**. Le passage entre ces deux espaces se fait via des appels systèmes. L'intérêt de cette séparation est que le système ne crash/se plante pas si une application plante.*

4. Structures internes de SE

- **Systèmes en couches (généralisation):**

- Les **systèmes en couches** sont une approche d'architecture logicielle qui organise un système complexe en plusieurs niveaux ou couches. Chaque couche est responsable d'un aspect spécifique du système, et les couches interagissent les unes avec les autres selon des règles définies.
- Chaque couche réalise sa fonction en s'appuyant exclusivement sur l'ensemble des fonctions de la couche qui lui est immédiatement inférieure.



Couche matérielle : Comprend le matériel physique, comme le processeur, la mémoire et les périphériques.

Couche du noyau : Gère les ressources matérielles, y compris la gestion des processus, la gestion de la mémoire et les opérations d'entrée/sortie.

Couche des services système : Fournit des services aux applications, comme les appels système pour la gestion des fichiers et des processus.

Couche des applications : Comprend les programmes et applications qui utilisent les services fournis par les couches inférieures.

4. Structures internes de SE

- **Systèmes à micro Kernel:**

- Un **système à micro-noyau** (ou microkernel en anglais) est une architecture de système d'exploitation qui vise à minimiser la taille et la complexité du noyau (kernel) en déplaçant autant de fonctionnalités que possible en dehors du noyau, vers des processus en espace utilisateur. Cette approche permet une plus grande modularité, une meilleure stabilité et une sécurité accrue.

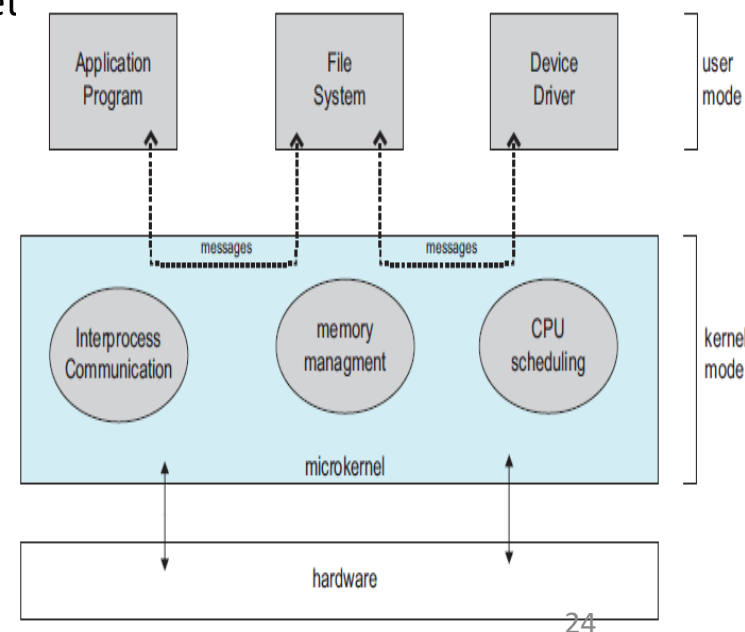
- Un noyau minimal (appelé micronoyau) de taille de code réduite.

- Se contente en général de gestionnaires de tâches et de mémoire simples, et un mécanisme de communication entre processus.

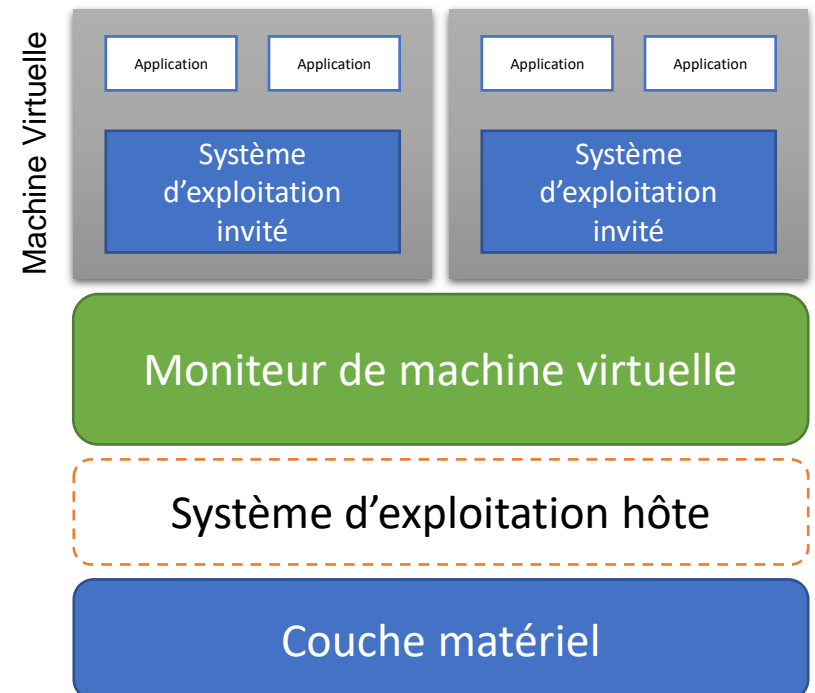
- Les gestionnaires de périphériques, les gestionnaires d'appels système, etc... sont implémentés en tant que programmes système et

✓ Un SE plus simple à étendre.

↓ Problèmes de performance.



- OS Machine virtuelle
- Possibilité de mettre plusieurs OS sur une machine physique.
- Le **moniteur de machine virtuelle** (hyperviseur) intercepte les instructions privilégiées envoyées par l'OS invité, les vérifie (politique de sécurité) et les exécute.
- XEN, VMWare, VirtualBox, etc.



I. Généralités sur les SE

1. Rappel: point sur le matériel
2. Définition d'un SE
3. Rôles d'un SE
4. Structure interne des SE
- 5. Le noyau**
6. Quelques familles de SE
7. La notion de multitâches multi-utilisateurs

5. Le noyau

- Le noyau (kernel en anglais) comporte un certain nombre des plus importantes routines (sous-programmes) du SE. Il est chargé en mémoire vive à l'initialisation du système. Les autres routines, moins critiques, sont appelées des **utilitaires**.
- Le noyau d'un système d'exploitation se compose de quatre parties principales : **le gestionnaire de tâches** (ou des processus), **le gestionnaire de mémoire**, **le gestionnaire de fichiers** et **le gestionnaire de périphériques d'entrée-sortie**.
- Il possède également deux parties auxiliaires : **le chargeur du système d'exploitation** et **l'interpréteur de commandes**.
 - **Le chargeur du système d'exploitation:**
 - Appelé, pour PC et MAC, BIOS (pour Basic Input Output System) et est chargé à une adresse bien déterminée en mémoire RAM.
 - Ce logiciel initialise les périphériques, charge un secteur du disque, et exécute ce qui y est placé.
 - **L'interpréteur de commandes (shell en anglais):**
 - Est souvent considéré comme une partie du SE
 - Exécute une boucle infinie qui affiche une invite (montrant par là que l'on attend quelque chose), lit le nom du programme et les paramètres saisis par l'utilisateur à ce moment-là et l'exécute.

5. Le noyau

- **Gestionnaire de tâches (ordonnanceur) :**
 - Divise le temps en laps de temps (en anglais **slices**, tranches),
 - Décide périodiquement d'interrompre le processus en cours et de démarrer (ou reprendre) l'exécution d'un autre.
- **Gestionnaire de mémoire:**
 - Connaître les parties libres et les parties occupées de la mémoire,
 - Allouer de la mémoire aux processus qui en ont besoin,
 - Récupérer la mémoire utilisée par un processus lorsque celui-ci se termine,
 - Traiter le va-et-vient entre le disque et la mémoire principale lorsque cette dernière doit être étendue.
- **Gestionnaire de fichiers:**
 - Faire abstraction des spécificités des disques et des autres périphériques d'entrée-sortie,
 - Offrir au programmeur un modèle agréable et facile d'emploi.
- **Gestionnaire de périphériques:**
 - Envoyer les commandes aux périphériques,
 - Intercepter les interruptions,
 - Traiter les erreurs.

1. Rappel: point sur le matériel
2. Définition d'un SE
3. Rôles d'un SE
4. Structure interne des SE
5. Le noyau
- 6. Quelques familles de SE**
7. La notion de multitâches multi-utilisateurs

6. Quelques familles de SE

- L'existence des SE depuis fort longtemps ainsi que leur utilisation dans différents domaines technologiques ont permis l'émergence de plusieurs familles de SE.
- **Mainframe:**
 - Ordinateurs de taille d'une pièce qu'on trouve dans les centres de données d'entreprise.
 - Grandes capacités: 1000 disques et des millions de gigaoctets de données.
 - Les SE sont amenés à traiter plusieurs tâches à la fois, dont la plupart nécessitent des quantités importantes d'E/S.



6. Quelques familles de SE

- Serveurs:

- PC à ressources importantes ou des workstations ou des Mainframes.
- Les SE doivent être capables de gérer plusieurs utilisateurs à la fois sur un réseau et de partager des ressources matérielles et logicielles.



- Multiprocesseurs:

- Un moyen pour obtenir une puissance de calcul majeure est de connecter plusieurs CPU dans un seul système.
- Ces systèmes ont besoin de SE assez spéciaux, des variations de SE pour serveurs.

6. Quelques familles de SE

- **Personal Computer:** Les SE pour PC modernes supportent la multiprogrammation, pour fournir un bon support à un seul utilisateur
Windows, Linux, Macintosh



- Handheld:

- Connu auparavant sous le nom de PDA (Personal Digital Assistant), ancêtre des smartphones et des tablettes.
- La plupart de ces appareils possèdent des CPU multicoeurs, GPS, caméras, capteurs...
- Ils ont des exigences particulières ce qui nécessite des SE sophistiqués et bien adaptés.



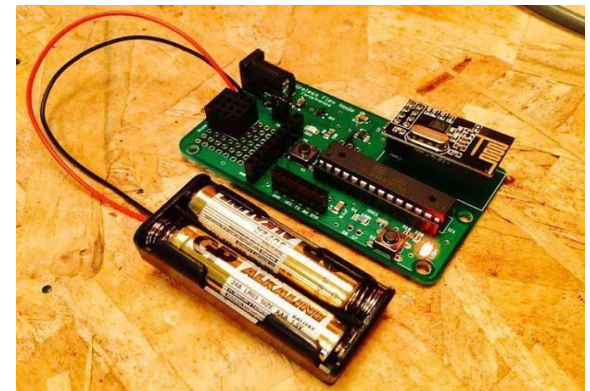
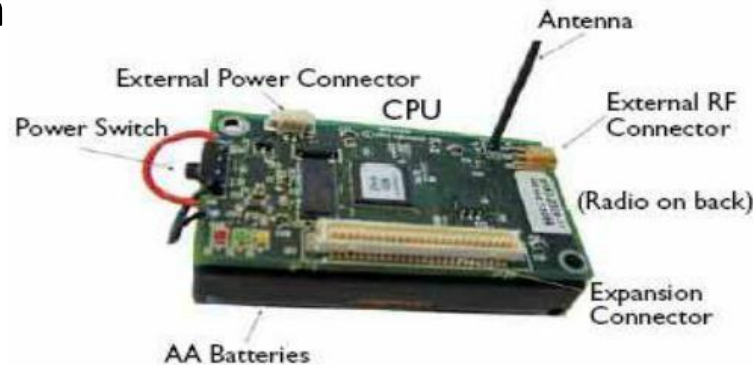
6. Quelques familles de SE

- Les systèmes embarqués:

- S'exécutent sur des circuits pour des périphériques n'acceptant pas de futures installations de logiciels par l'utilisateur (Ex: fours à micro-ondes, les téléviseurs, les voitures, etc): tout est en ROM.
- Ces SE n'ont pas besoin de considérer l'aspect de protection contre les logiciels malveillants, ce qui entraîne une simplification de la conception.

- Nœud capteur:

- Unités qui composent un réseau de capteurs sans fil.
- Chaque nœud est composé de CPU, RAM, ROM, capteurs (température, humidité, lumière, mouvement ...): un vrai ordinateur.
- Le SE doit être petit et simple car les nœuds ont peu de RAM et sont contraints en ' ' .



6. Quelques familles de SE

- **Systèmes temps réel:**

- Se caractérisent par le temps (ou le délai) comme contrainte.
- Si le système doit fournir des garanties absolues qu'une certaine action se produira à un certain moment (ou dans un certain intervalle temps), on parle d'un système en temps réel strict (**Hard real-time**).
- Si le système doit fournir des garanties avec une certaine probabilité et ainsi un délai plus grand mais acceptable, on parle de temps réel souple (**Soft real-time**).

1. Rappel: point sur le matériel
2. Définition d'un SE
3. Rôles d'un SE
4. Structure interne des SE
5. Le noyau
6. Quelques familles de SE
7. La notion de multitâches multi-utilisateurs

7. La notion de multitâches multi-utilisateurs

- **Systèmes multitâches:**

- Appelés aussi multi-programmés
- Permettent l'exécution de plusieurs tâches à la fois : exécuter un programme utilisateur, lire les données d'un disque, afficher des résultats sur un terminal.
- Ce contexte fait appel aux notions suivantes:
 - **Processus (et non programme):** un processus est une instance de programme en exécution. Le processus est représenté par un programme (le code), ses données et son état d'avancement communément appelés variables d'environnement .
 - **Temps partagé:** le micro-processeur à un instant donné, n'exécute réellement qu'un seul processus. Faire passer le processeur d'un processus à un autre, en exécutant chaque programme pendant quelques dizaines de millisecondes, donne l'impression que tout s'exécute en même temps : c'est **Le pseudo-parallélisme**.
 - **Espace mémoire d'un processus:** chaque processus possède son propre espace mémoire, non accessible aux autres processus. On parle de **l'espace d'adressage du processus**.

7. La notion de multitâches multi-utilisateurs

- **Systèmes multi-utilisateurs:**
 - Capables d'exécuter de façon **concurrente** et **indépendante** des applications appartenant à plusieurs utilisateurs.
 - **Concurrente?:**
 - Les applications sont actives au même moment et se disputent l'accès aux différentes ressources.
 - **Indépendante?:**
 - chaque application peut réaliser son travail sans se préoccuper de ce que font les applications des autres utilisateurs.
 - Ce contexte fait appel aux notions suivantes:
 - **Utilisateurs:** matérialisé par un espace privé de travail sur la machine. Chaque utilisateur est identifié par un numéro unique, appelé l'identifiant de l'utilisateur, ou UID (pour l'anglais User IDentifier).
 - **Groupe d'utilisateurs:** permet de partager de façon sélective le matériel avec d'autres utilisateurs. Un groupe est également identifié par un numéro unique dénommé identifiant de groupe, ou GID (pour l'anglais Group IDentifier).
 - **Super-utilisateur:** ou encore superviseur (root en anglais), un utilisateur particulier qui peut pratiquement tout faire dans la mesure où le SE ne lui applique jamais les mécanismes de protection. Il peut, en particulier, accéder à tous les fichiers du système et interférer sur l'activité de n'importe quel processus en cours d'exécution.