

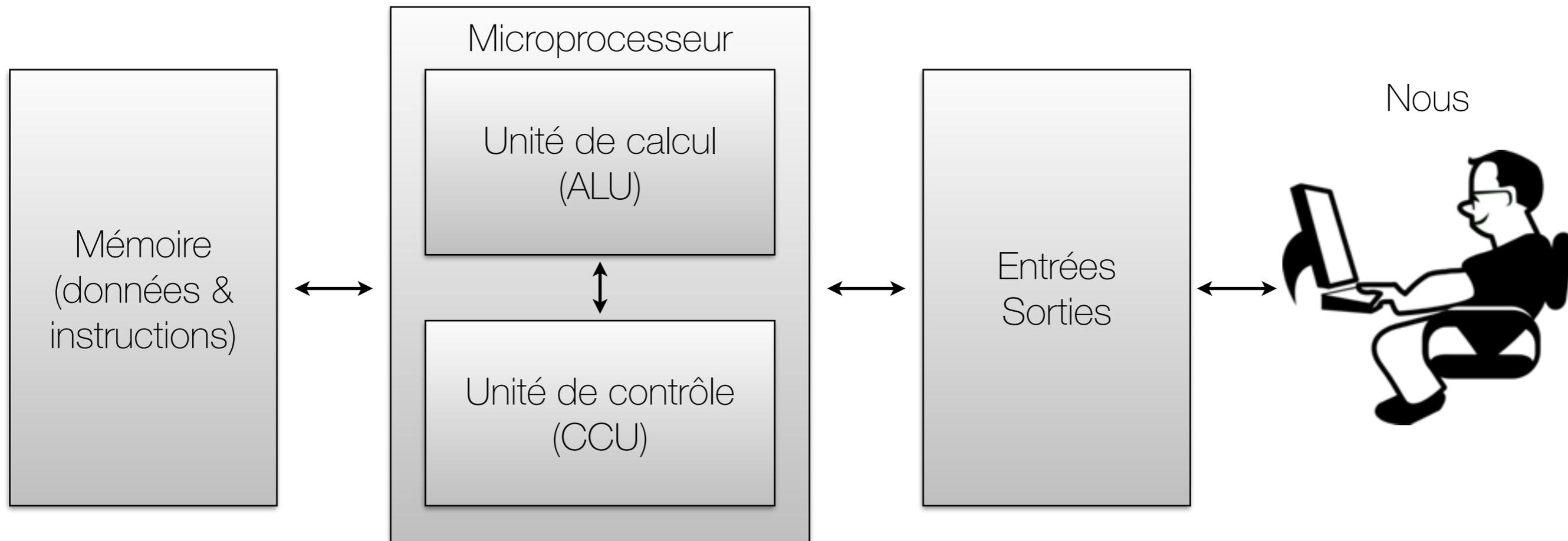
Introduction aux Systèmes d'Exploitation



UNIX PEOPLE ARE HAPPY

GIF-1001 Ordinateurs: Structure et Applications, Hiver 2015
Jean-François Lalonde

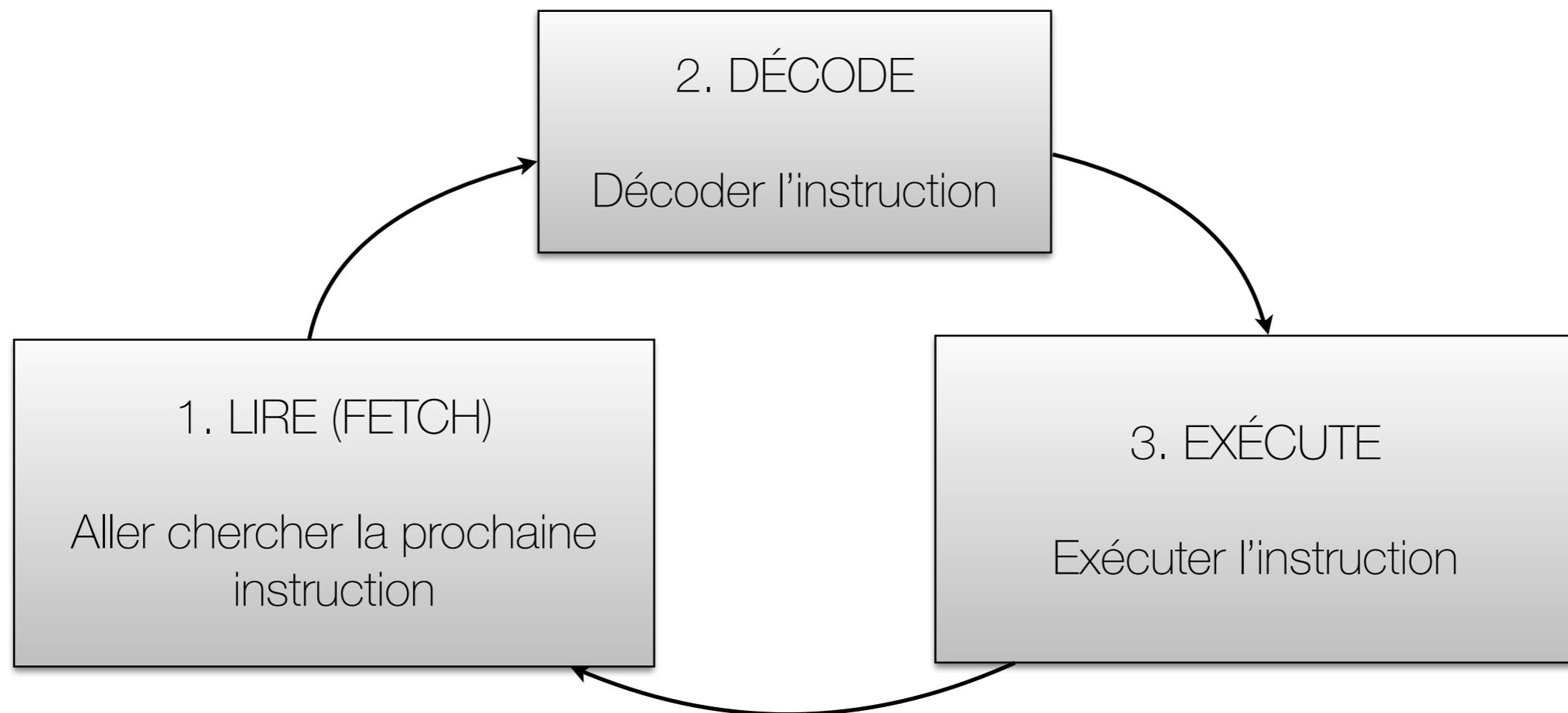
Rappel: Architecture von Neumann



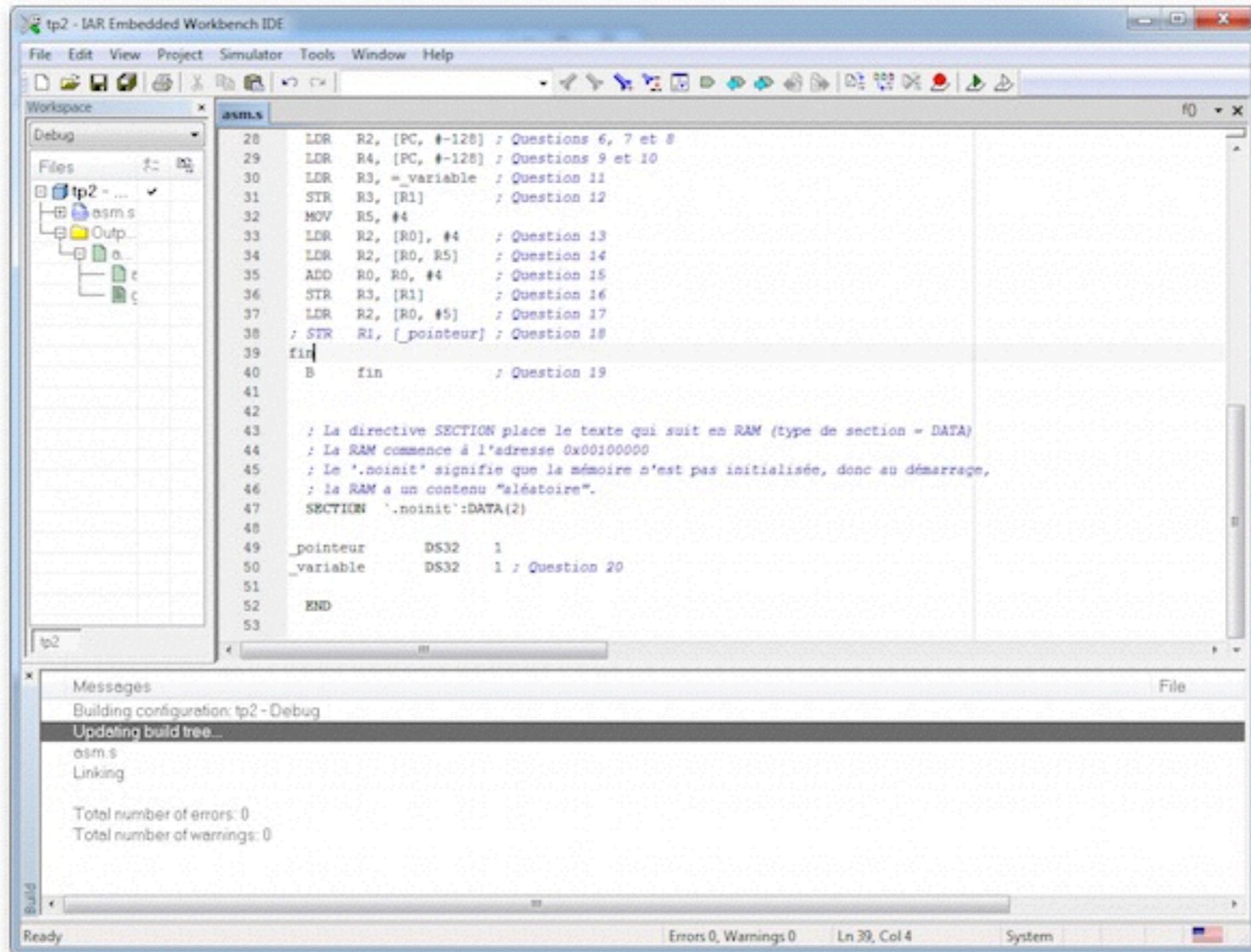
Rappel: cycle d'instructions

Que fait le microprocesseur?

1. Lire: aller chercher la prochaine instruction
2. Décode: décode l'instruction (détermine ce qu'il y a à faire)
3. Exécute: exécuter l'instruction



Rappel: écriture et exécution d'un programme



The screenshot displays the IAR Embedded Workbench IDE interface. The main window shows the assembly source file 'asm.s' with the following code:

```
28 LDR R2, [PC, #-128] ; Questions 6, 7 et 8
29 LDR R4, [PC, #-128] ; Questions 9 et 10
30 LDR R3, =_variable ; Question 11
31 STR R3, [R1] ; Question 12
32 MOV R5, #4
33 LDR R2, [R0], #4 ; Question 13
34 LDR R2, [R0, R5] ; Question 14
35 ADD R0, R0, #4 ; Question 15
36 STR R3, [R1] ; Question 16
37 LDR R2, [R0, #5] ; Question 17
38 ; STR R1, [_pointeur] ; Question 18
39 fin
40 B fin ; Question 19
41
42
43 ; La directive SECTION place le texte qui suit en RAM (type de section = DATA)
44 ; La RAM commence à l'adresse 0x00100000
45 ; Le '.noinit' signifie que la mémoire n'est pas initialisée, donc au démarrage,
46 ; la RAM a un contenu "aléatoire".
47 SECTION '.noinit':DATA(2)
48
49 _pointeur DS32 1
50 _variable DS32 1 ; Question 20
51
52 END
53
```

The Messages window at the bottom shows the build process:

```
Messages
Building configuration: tp2 - Debug
Updating build tree...
asm.s
Linking

Total number of errors: 0
Total number of warnings: 0
```

The status bar at the bottom indicates 'Ready', 'Errors 0, Warnings 0', 'Ln 39, Col 4', and 'System'.

Questions

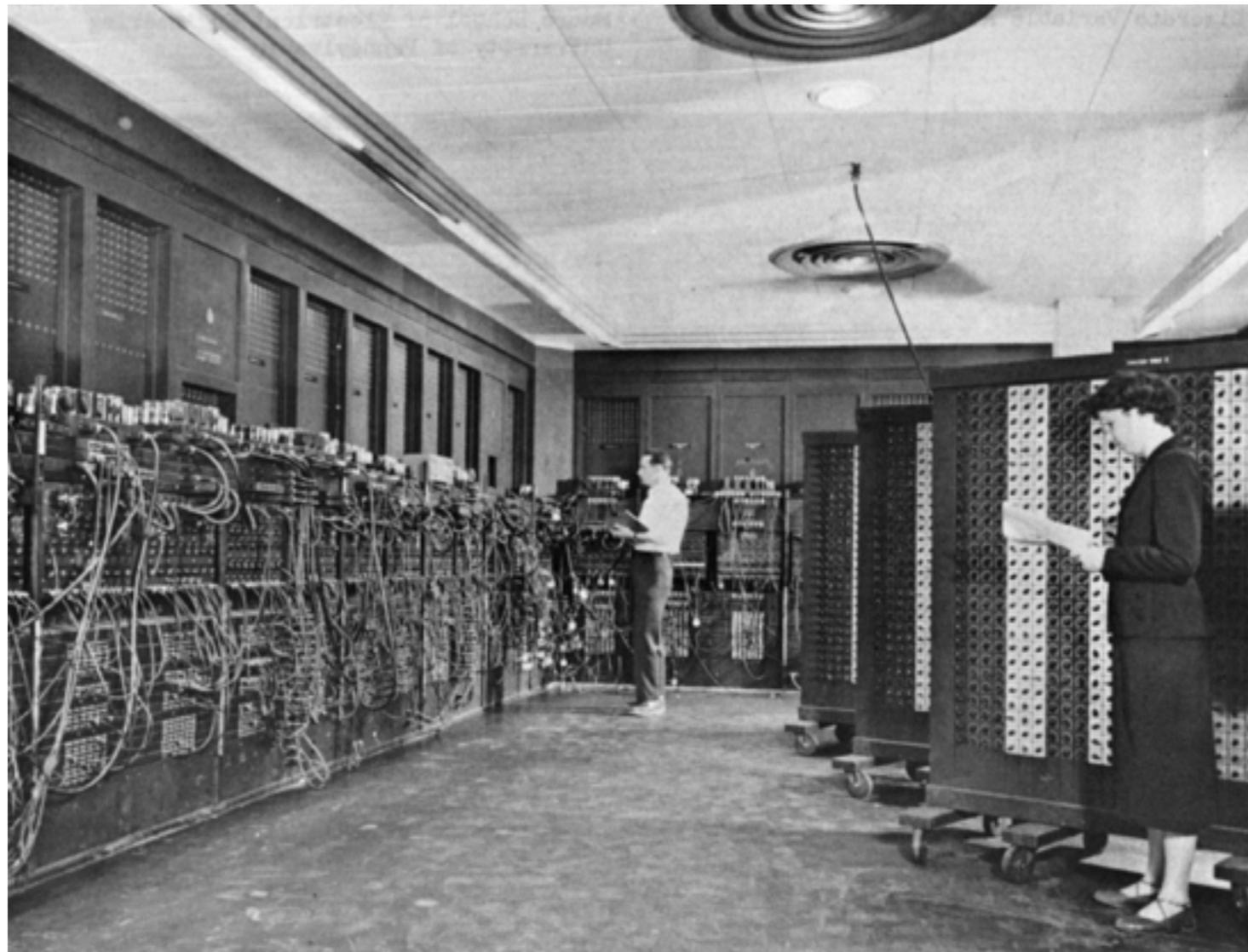
- Comment le programme a-t-il été chargé en mémoire?
- Comment faire pour dire à l'ordinateur de débiter l'exécution d'un programme?
- Comment peut-on interagir avec un programme?
- Que doit-on faire quand un programme est terminé?
- Comment peut-on exécuter plus d'un programme?
- Pourquoi ne produit-on pas d'exécutables dans les TPs?

ENIAC (Maulchy & Eckert, 1946)

calculs de balistique durant la 2e Guerre Mondiale

les panneaux à gauche étaient les programmes: il fallait programmer manuellement en branchant et débranchant ces connexions

18,000 tubes à vide, 15,000 pieds carrés, 30 tonnes, 140 kwatts puissance



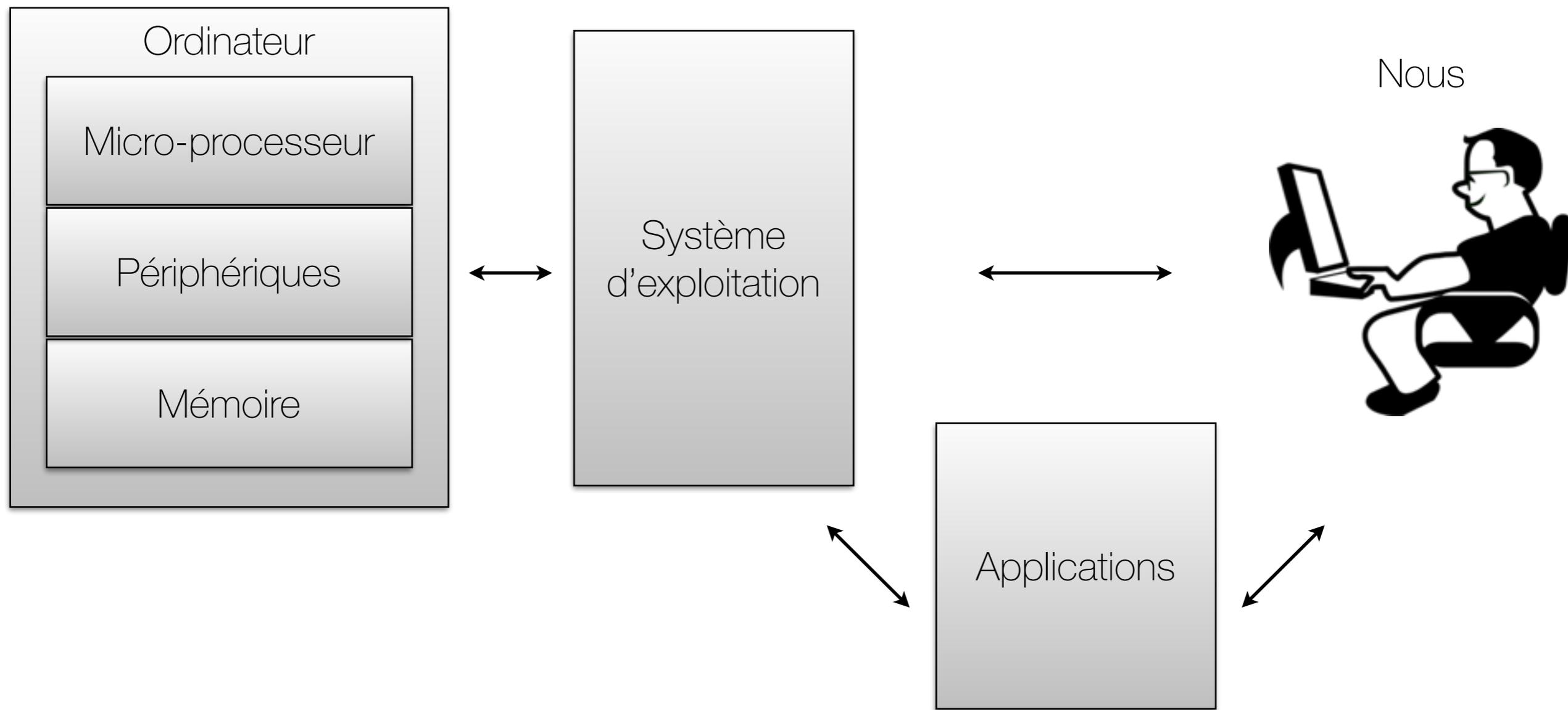
Les premiers ordinateurs

- Les premiers ordinateurs étaient constitués d'un microprocesseur, une mémoire, un périphérique d'entrée, quelques voyants/bascules et une imprimante...
- Le programmeur introduisait le programme à exécuter dans la mémoire du microprocesseur à partir du périphérique d'entrée (lire lecteur de carte perforée...). Le microprocesseur était déconnecté pendant ce temps!
- Des bascules permettaient de lancer l'exécution du programme par le microprocesseur.
- Des voyants donnaient les statuts d'erreurs et les résultats du programme. En cas d'erreur, le programmeur regardait directement la mémoire...
- Un seul programme était exécuté à la fois. Il fallait s'inscrire sur des listes afin de pouvoir utiliser l'ordinateur.

Systeme d'exploitation (SE) — définition générale

- Un système d'exploitation est
 - un ensemble de programmes
 - qui gèrent les ressources matérielles d'un ordinateur et qui rendent ces ressources disponibles à l'utilisateur,
 - de telle sorte que l'utilisateur peut utiliser efficacement son ordinateur.

Systeme d'exploitation



Analogies



- Un illusionniste :
 - Fait disparaître certaines limites du matériels
 - Donne l'illusion que la machine a une mémoire infinie et une infinité de processeurs



- Un gouvernement :
 - Protège les utilisateurs les uns des autres
 - Partage des ressources de façon efficace et équitable

Restrictions vs. possibilités

- S.E. est l'équivalent de vivre en société
 - Partage des ressources, incluant CPU
 - Règlement à suivre : perte de liberté
 - Augmente les possibilités



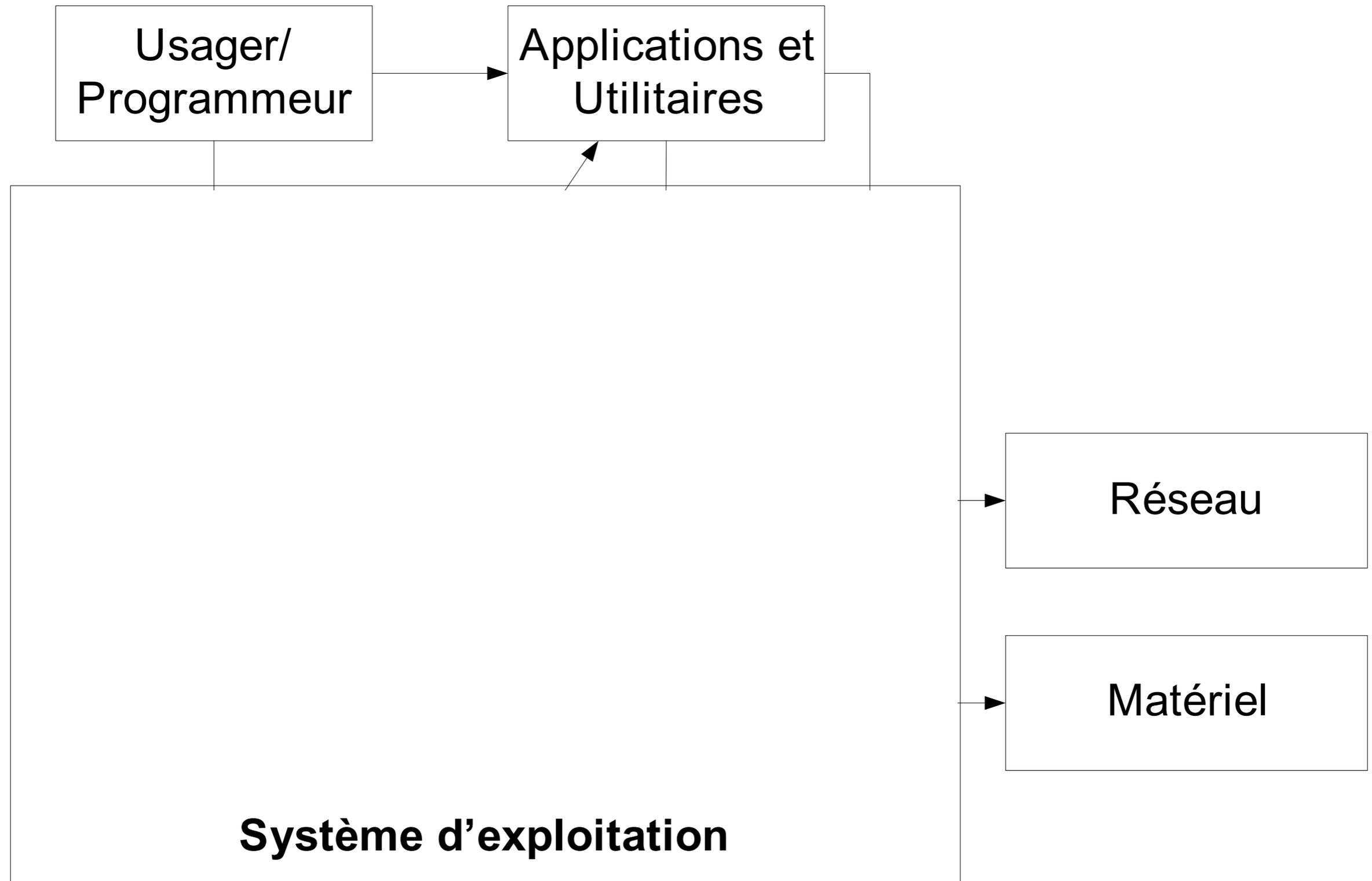
Rôles du système d'exploitation

- Les principaux rôles d'un système d'exploitation sont:
 - Fournir une interface usager conviviale.
 - Gérer, lire et exécuter des programmes.
 - Gérer les ressources matérielles de l'ordinateur.
- Le but d'un système d'exploitation est de faciliter l'utilisation de l'ordinateur et de rendre son utilisation efficace.

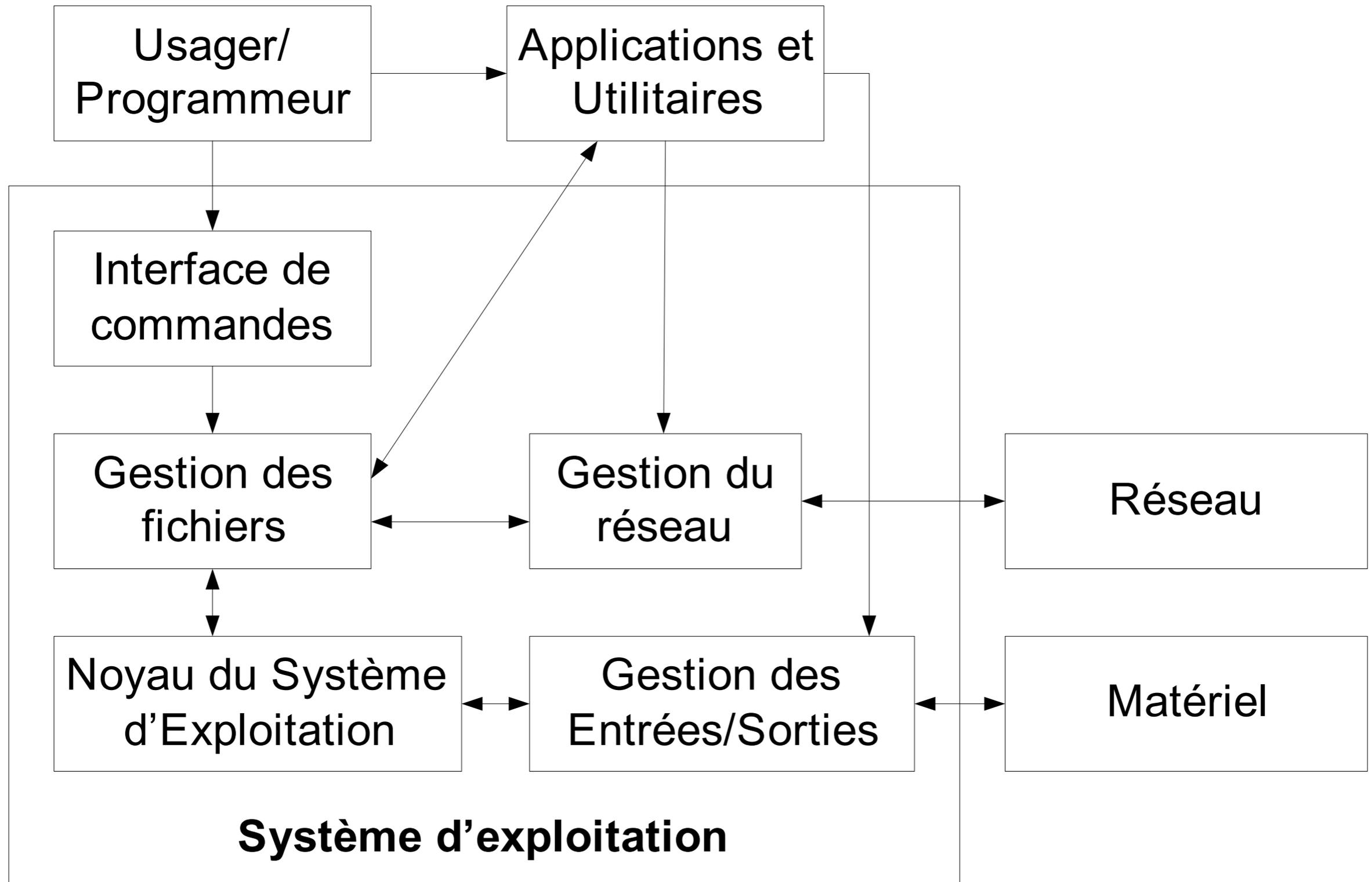
Le système d'exploitation: un programme

- Le système d'exploitation est un programme.
 - Il occupe une partie de la mémoire de l'ordinateur.
 - Il est chargé en mémoire par le BIOS.
 - Le rôle du système d'exploitation est de lancer d'autres programmes. Il doit « abandonner » le contrôle du microprocesseur aux autres programmes pour que ceux-ci puissent être exécutés.

Vision simplifiée des services d'un SE



Vision simplifiée des services d'un SE



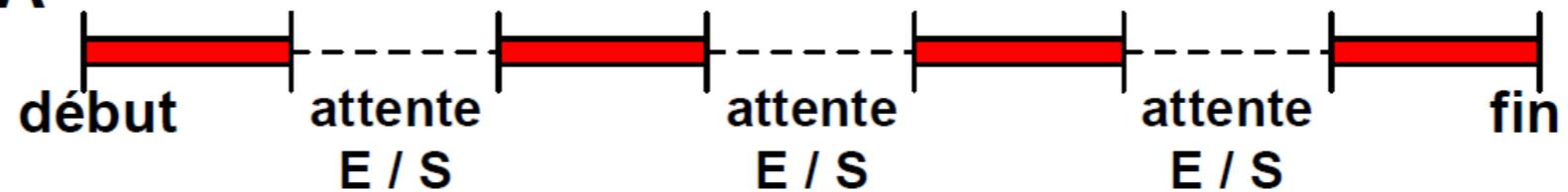
Vision simplifiée des services d'un SE

- Un système d'exploitation doit fournir les services suivants:
 - Une interface de commandes pour l'utilisateur
 - Un support pour la création de programmes
 - Une gestion des fichiers
 - Une gestion des programmes et applications
 - Une façon de poursuivre le démarrage de l'ordinateur amorcé par le BIOS
 - Un service des interruptions (Software et Hardware)
 - Une gestion des entrées/sortie et du matériel qui y est relié.
 - Des services pour les réseaux et pour le matériel qui y est associé.
 - Une mesure des statistiques et performances de l'ordinateur.

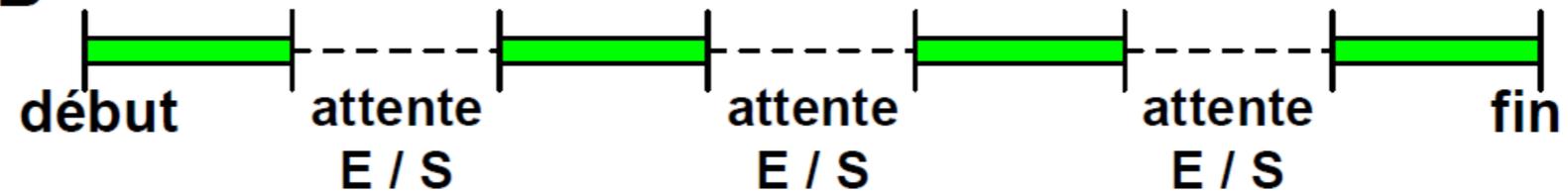
Processus

- E/S = entrée-sortie
 - ex: lecture ou écriture sur le disque

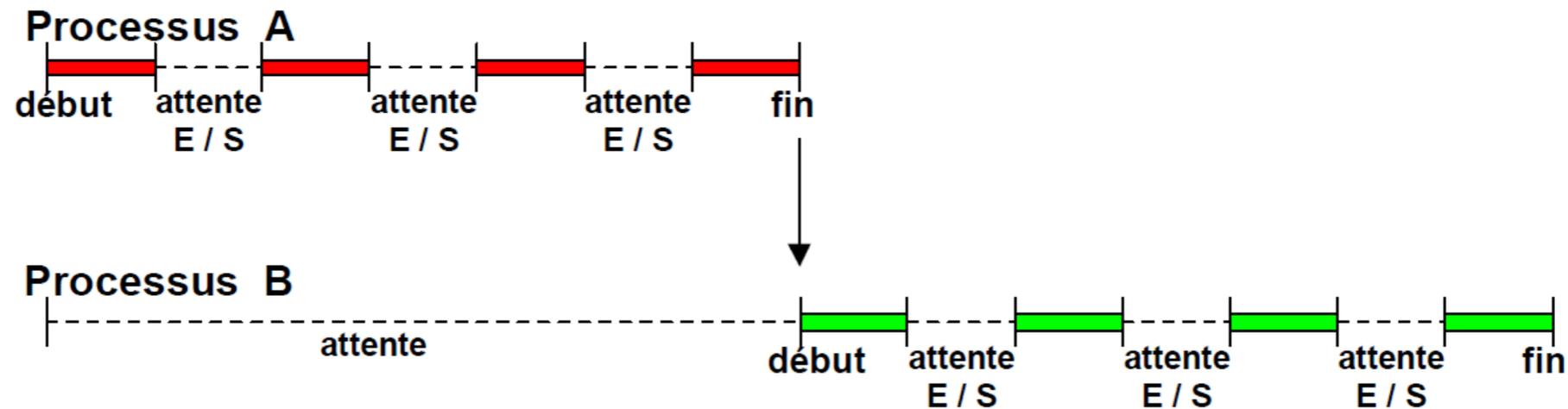
Processus A



Processus B

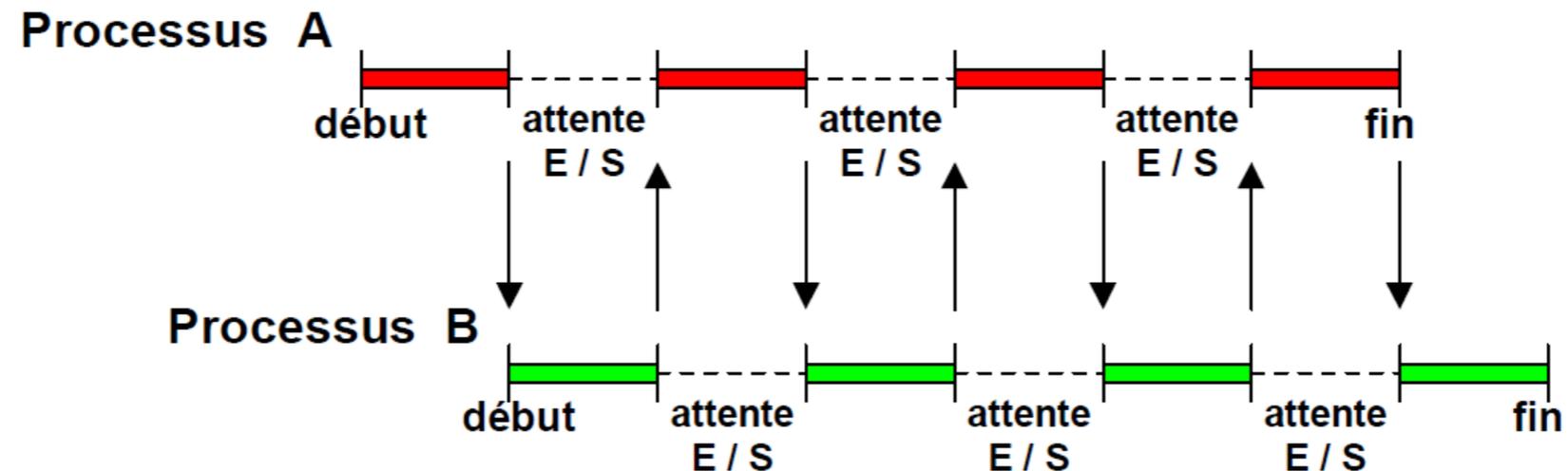


SE “simples” (e.g. DOS)



- Un seul programme peut être exécuté à la fois
- Tout de même pas si simple! Il peut:
 - gérer des périphériques.
 - traiter les interruptions du système.
 - fournir un interpréteur de commandes pour traiter les requêtes de l'utilisateur
 - charger un programme en mémoire, puis de l'exécuter
 - fournir un ensemble de routines disponible pour la création de programmes.
 - gérer les fichiers et les accès disques.

SE “multi-tâches” (e.g. UNIX)



- détermine la séquence de programmes à exécuter selon plusieurs facteurs (priorité, utilisation d’entrées/sorties, conflit de ressources, etc.)
- offre des protections et des niveaux de sécurité pour les accès aux ressources de l’ordinateur.
- gère une mémoire et des ressources pouvant être communes à plusieurs processus.
- gère la communication avec d’autres ordinateurs.

Type: SE dans les PCs

- Certainement les plus familiers
 - Monotâche — multitâches
 - Importance GUI conviviale
 - Support pour nombreux périphériques
 - Nombreux logiciels disponibles
- Exemples:
 - Windows XP/Vista/7/8
 - MacOS
 - Ubuntu (Linux)



Type: Serveurs

- Application typiques :
 - web
 - fichiers – email
- Grand nombre de petites tâches
- Exemples de S.E. :
 - Solaris
 - FreeBSD (Berkeley Software Distribution)
 - Linux
 - Windows Server 20xx



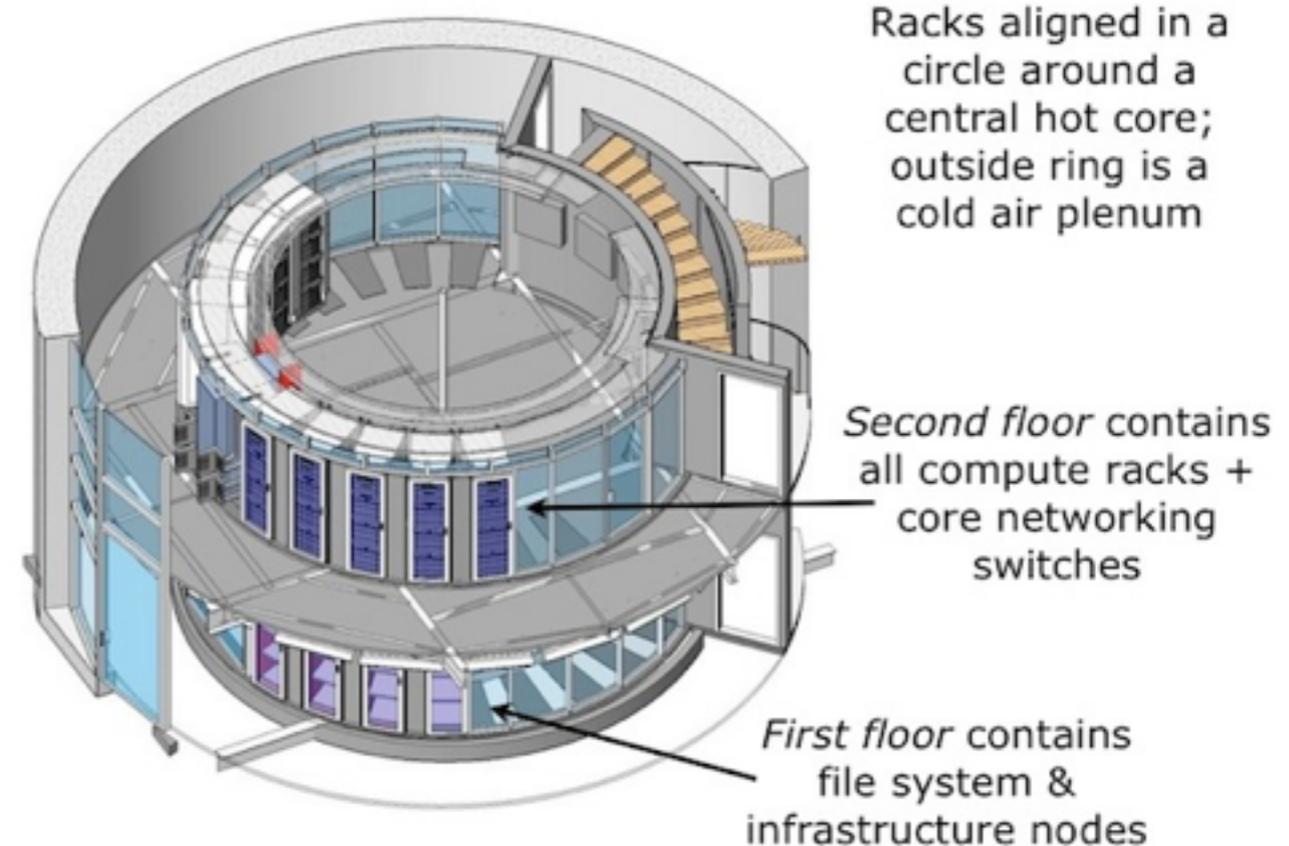
Type: Serveurs à grande échelle

- Google: >450,000 serveurs
- Facebook: >60,000 serveurs
- Utilisent souvent des versions propriétaires comme S.E. (développent leurs propres S.E.)
- 1.3% de l'électricité mondiale est utilisée pour les "data centers"



Type: Super-calculateurs

- Grosses tâches
 - calcul scientifiques: météo, séquençage de génôme, etc.



Type: Systèmes embarqués

- S.E. simplifiés
 - peu ou pas d'interface usager
 - taille mémoire + CPU réduite
- Peuvent être mono-tâches
- Pas de démarrage de nouveaux processus par l'utilisateur : statique
- Exemples :
 - QNX, VxWorks
 - Kernel Linux + Busybox



Type: Systèmes temps-réel

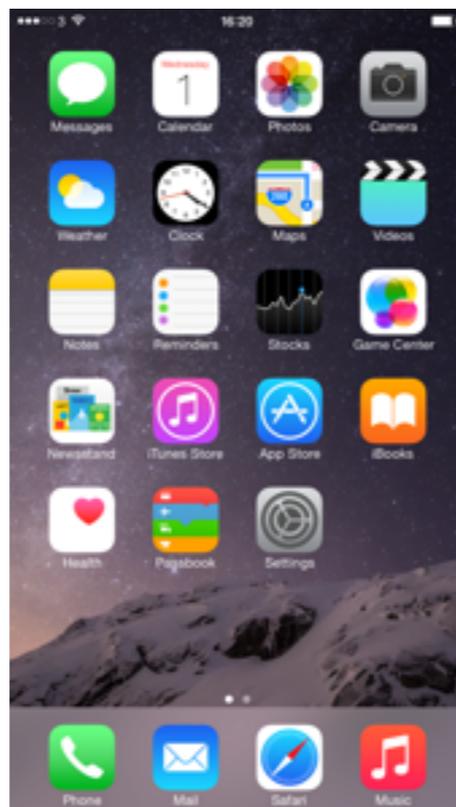
- Met l'accent sur la prédictibilité des temps d'exécution
- e.g. exécuter tâche exactement à chaque milliseconde
- Exemples:
 - QNX
 - VxWorks
 - FreeRTOS
 - Real Time Linux



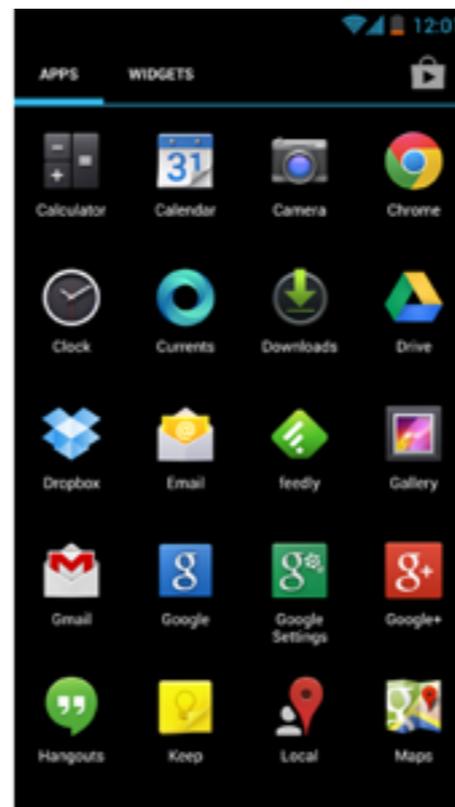
Type: Systèmes mobiles

- Systèmes fermés (pas d'ajout périphérique)
- Petit / CPU limitée / téléphonie / réseau / GPS
- Faible consommation électricité
- Emphase sur l'interface graphique, pas la performance calcul
- Démarrage rapide, utilisation courte

Apple iOS



Android OS



Windows Phone OS



Références et exercices

- Références
 - Englander: Section 15.1, 15.2 (début) et 15.8
- Exercices
 - <http://vision.gel.ulaval.ca/~jflalonde/cours/1001/h15/index.html#ressources>