

R1.04 : Introduction aux systèmes d'exploitation et à leur fonctionnement

BUT Informatique – Semestre 1

Jean-François ANNE, Athénais VAGINAY

jean-francois.anne@unicaen.fr, athenais.vaginay@unicaen.fr



Avant-propos

Athénaïs Vaginay, mcf IUT / GREYC, athenais.vaginay@unicaen.fr

Jean-Francois Anne, jean-francois.anne@unicaen.fr

- Organisation R1.04 :
 - 8 **cours magistraux** (CM) – 1h amphi– promo
 - 8 séances de **travaux dirigés** (TD) – 2h sur machine – en groupe
 - 4 séances de **travaux pratiques** (TP) – 2h sur machine – en demi-groupe
- Documents sur <https://nanls.github.io/classes/index.html>, mais **prenez des notes**.
- Une note individuelle (semaine du 11 nov) + une note en binôme (QCM à chaque TP)
- Questions posables en cours et/ou par mail (objet + **du contexte**)

Sommaire (prévision)

	CM (8)	TD (8)	TP (4)
Caractéristiques et structure des systèmes d'exploitations (couches noyaux, shell, apps)	1		
Historique et types des systèmes d'exploitations	2		
Langage de commandes : commandes de base	2	1, 2	1, 2, 3, 4
- Gestion des utilisateurs (caractéristiques, création, suppression, etc.)	3	1	1
- Gestion des fichiers (arborescence, types, droits, etc.)	3, 4	1, 2, 3, 4	1
- Gestion des processus (création, destruction, suivi, communications etc.)	5	5, 6	1
- Installation et configuration d'un système	6		1
Langage de commandes : introduction à la programmation des scripts	7, 8	7, 8	2, 3, 4

Focus sur Linux mais TP2, 3, 4 sur Windows

Installation d'un système d'exploitation (Linux et Windows) : R1.03, avec JFA



Bibliographie et Webographie

- Richard W. Stevens. Advanced Programming in the Unix Environment. Addison-Wesley, 2013.
- Michael Kerrisk. The Linux Programming Interface : A Linux and UNIX System Programming Handbook., No Starch Press, 2010
- Graham Glass. Unix for Programmer and Users. Prentice Hall, 1993.
- Joëlle Delacroix. Linux, Programmation système et réseau. Dunod, 2003
- **Les slides de JFA 2023-2024**
- <http://e-classroom.over-blog.com/les-systemes-d-exploitation>
- <https://distrowatch.com/>
- <https://www.linuxfromscratch.org>
- <https://www.lpi.org/our-certifications/>
- http://n.grassa.free.fr/sysrezo/systeme/Cours_Systeme.pdf



Je s'appelle ROOT !



Présentation des systèmes d'exploitation

Tout au long de cette ressource, nous apprendrons **comment fonctionnent les systèmes d'exploitation**. Mais, avant d'approfondir nos connaissances de **Linux** et **Windows**, deux des plus fascinants systèmes d'exploitation, nous allons définir **ce qu'est un système d'exploitation** et en voir **les composantes**.



Deux catégories de programmes

- Les **programmes d'application** : résolvent les problèmes des utilisateurs.
- Les **programmes systèmes** : pour le fonctionnement des ordinateurs.

Le programme « **système d'exploitation** » est le programme fondamental des programmes systèmes. Il contrôle les ressources de l'ordinateur et fournit la base sur laquelle seront construits les autres programmes.



Systeme d'Exploitation (SE) = Operating System (OS)

Vous en connaissez déjà plusieurs... des idées ? :)))

- L****
- W*****

Systeme d'Exploitation (SE) = Operating System (OS)



Des guerres entre entreprises :

- « Passées », entre Unix (cours 2), entre Microsoft et Linux
- Présentes, [entre Microsoft et Google](#)

Des débats sans fin pour savoir « lequel est le meilleur ».

Android, iOS ... et plein d'autres !



Fonctions des systèmes d'exploitation

Exécuter un programme sur un ordinateur « nu »

Pour **exécuter un programme**, il faut :

- Aller le chercher sur le disque dur :
 - Trouver sa position
 - Lire les mots qui le décrivent
- Le mettre en mémoire
 - Lui allouer un espace
- L'exécuter...
 - Gestion du clavier par ce programme?
 - Gestion de l'écran ?

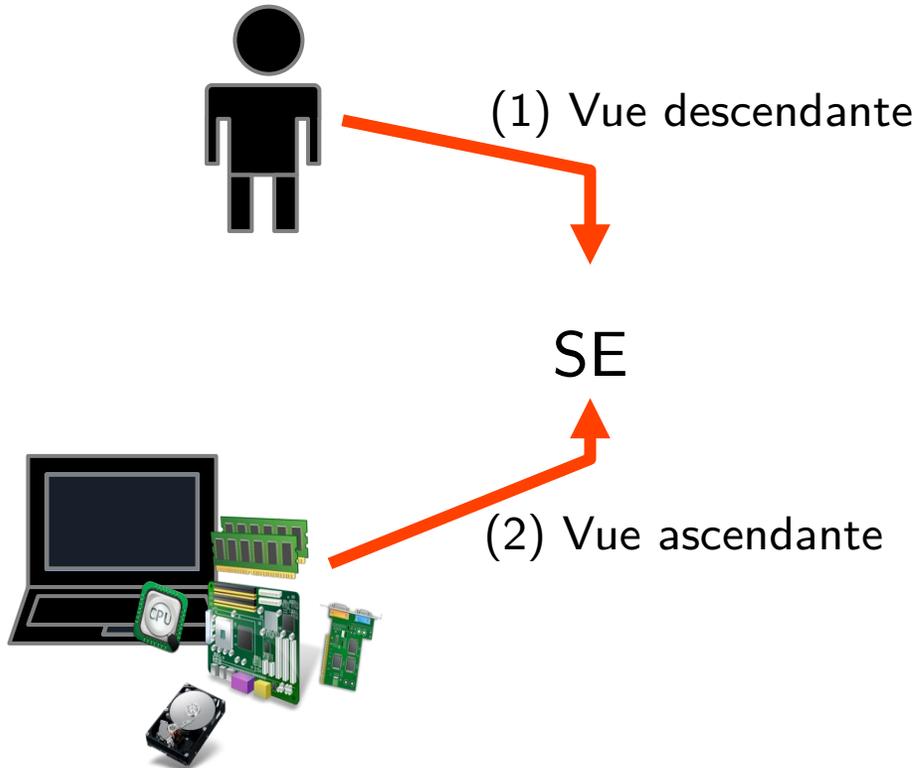
Mais un ordinateur nu, c'est :

- Programmation en **langage binaire** seulement
- Accès aux périphériques (clavier, écran, ...) très **difficiles**
- Exécution d'**un seul programme** à la fois

⇒ **Un ordinateur nu est « inutilisable »**

Les deux vues du SE

Intermédiaire entre l'utilisateur.ice et le système informatique



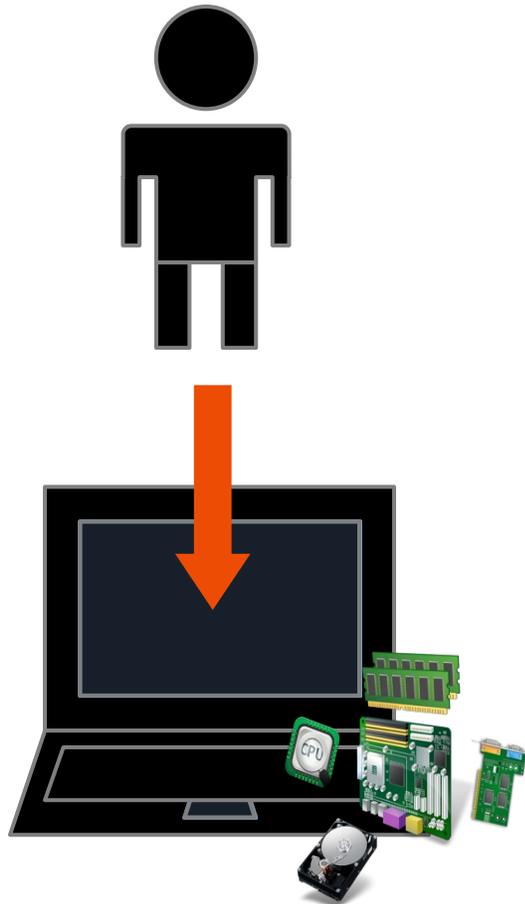
(1) Présenter **une machine virtuelle** : Son rôle est de masquer les éléments fastidieux liés au matériel et permettre à l'utilisateur.ice une exploitation simple et efficace de la machine ;

(2) Être **un gestionnaire de ressources** : Gérer l'**ordonnancement** et le contrôle de l'allocation des processeurs, des mémoires et des périphériques d'E/S entre les différents programmes qui y font appel.

E/S = Entrées/Sorties

Vue descendante :

Le SE est une machine virtuelle (= machine étendue)



Un ordinateur contient du **matériel complexe** : un ou plusieurs processeurs, une mémoire principale, des horloges, des terminaux, des disques, des interfaces de connexion à des réseaux, des périphériques d'entrées/sorties....

Via un mécanisme d'**abstraction** (= masquer les éléments complexes fastidieux à gérer), le système d'exploitation transforme cet assemblage de chips et de circuits en un appareil plus utilisable.

⇒ simplification de l'interface humain-machine (notée « IHM »)



Vue descendante :

Le SE est une machine virtuelle (= machine étendue)

Fonction	Ressource matérielle	Concept abstrait
Gestion des données persistantes (accès, modification, stockage, partage)	Disques avec tête de lecture, disque SSD, clés USB, ...	Fichier, répertoire (via le système de fichier), instructions « lire_fichier »
Gestion des communications avec l'utilisateur (entrée/sortie)	Clavier, écran, souris	GUI (fenêtres) / terminal (ligne de commandes)
Gestion des activités (création, suivi, destruction, erreur)	Processeur, programme sur disque, mémoire centrale	Processus (cours dédié plus tard)
Mémoire de travail	RAM	Mémoire virtuelle

GUI = Graphical User Interface

Vue descendante :

Le SE est une machine virtuelle (= machine étendue)

Exemple vis-à-vis de l'utilisation de la mémoire : la **virtualisation de l'allocation mémoire**

Allocation des adresses mémoires disponibles aux **processus** (= programme en cours d'exécution).

Un processus demande plus de mémoire → il lui est alloué les **prochaines** adresses disponibles (en RAM, ou sur disque = swap). On se retrouve donc **potentiellement** avec des **blocs pas consécutifs**, ce qui est **fastidieux** à gérer pour le programmeur.

→ **pour cacher la complexité, le SE travaille de concert avec le MMU** (Memory Management Unit, un circuit sur la carte mère) : il font une **abstraction** et laisse le processus croire que la mémoire qui lui est allouée est consécutive.

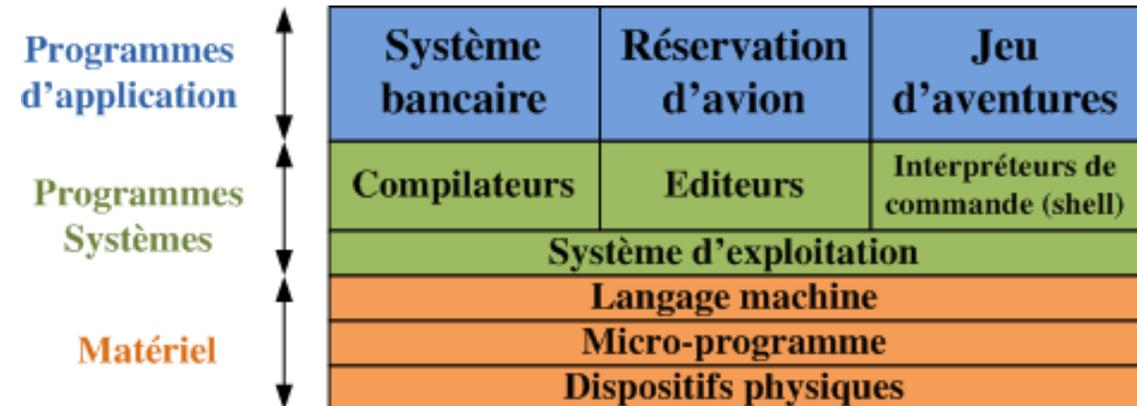
PHYSICAL MEMORY ADDRESSES	VIRTUAL MEMORY ADDRESSES	ALLOCATED TO:
0-999	0-999	PROGRAM A
1000-1999	0-999	PROGRAM B
2000-2999	1000-1999	PROGRAM A
3000-3999	(NOT ALLOCATED)	(NOT ALLOCATED)
...

Vue descendante :

Le SE est une machine virtuelle (= machine étendue)

Un ordinateur...

- contient des **dispositifs physiques** : ils se composent de circuits intégrés, de fils électriques, de périphériques physiques ...
- contient un **microprogramme** = un logiciel de contrôle des périphériques (interprète).
- utilise le **langage machine** = un ensemble d'**instructions élémentaires (bas niveau)** (entre 50 et 300) pour effectuer le déplacement des données, des calculs, ou la comparaison de valeurs.
- utilise un **système d'exploitation**, qui propose un ensemble d'**instructions plus haut niveau**, comme LIRE_FICHER que peuvent utiliser les **programmes d'application**).



Vue ascendante : le SE est un gestionnaire des ressources



=



- Mémoire
- Processeur
- Périphériques
- Fichiers
- Stockage
- ...

Un ordinateur se compose de **plusieurs ressources** qu'il faut **partager** entre les différents utilisateurs (multi-utilisateur) et programmes (multi-tâche).



Vue ascendante : le SE est un gestionnaire des ressources

- Partage **entre les programmes (processus)** :
le rôle de policier du SE permet d'éviter les conflits d'utilisation de la mémoire, des périphériques d'entrées/sorties, des interfaces réseau... etc.
- Partage **entre les usagers** :
le partage de la mémoire et surtout sa protection demeure une priorité absolue.

⇒ **En tout temps, un bon système d'exploitation connaît l'utilisateur d'une ressource, ses droits d'accès, son niveau de priorité.**

Exemples :

- 3 programmes essaient d'imprimer simultanément leurs résultats sur une même imprimante ⇒ recours à un fichier tampon sur disque.
- accès simultané à une donnée ; lecture et écriture concurrentes (par deux processus) sur un même compteur.



Rôles du système d'exploitation

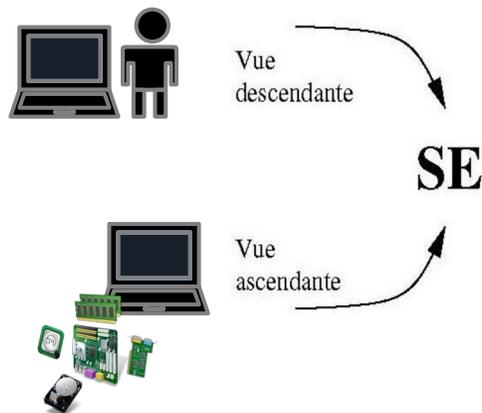
- **Transformer une machine matérielle en une machine utilisable**, c-à-d fournir des outils adaptés aux besoins indépendamment des caractéristiques physiques.
- **Gérer les ressources.**

Mais il faut également la garantie d'un bon niveau en matière de :

- **Sécurité** : intégrité, contrôle des accès confidentialité...
- **Fiabilité** : degré de satisfaction des utilisateurs même dans des conditions hostiles et imprévues,
- **Efficacité** : performances et optimisation du système pour éviter tout surcoût en termes de temps et de places consommées par le système au détriment de l'application.

Rôles du système d'exploitation

- **Transformer une machine matérielle en une machine utilisable**, c-à-d fournir des outils adaptés aux besoins indépendamment des caractéristiques physiques. [vue descendante]
- **Gérer les ressources.** [vue ascendante]



Mais il faut également la garantie d'un bon niveau en matière de :

- **Sécurité** : intégrité, contrôle des accès confidentialité...
- **Fiabilité** : degré de satisfaction des utilisateurs même dans des conditions hostiles et imprévues,
- **Efficacité** : performances et optimisation du système pour éviter tout surcoût en termes de temps et de places consommées par le système au détriment de l'application.



Divers rôles du système d'exploitation

machine virtuelle, gestionnaire. de ressource (+ sécurité, fiabilité, efficacité)

- **Gestion du processeur** : le SE est chargé de gérer l'allocation du processeur entre les différents programmes grâce à un algorithme d'ordonnancement. Le type d'ordonnanceur est totalement dépendant du système d'exploitation, en fonction de l'objectif visé.
- **Gestion de la mémoire vive** : le SE est chargé de gérer l'espace mémoire alloué à chaque application et, le cas échéant, à chaque usager. En cas d'insuffisance de mémoire physique, le système d'exploitation peut créer une zone mémoire sur le disque dur. Cela permet de faire fonctionner des applications nécessitant plus de mémoire qu'il n'y a de mémoire vive disponible sur le système. En contrepartie cette mémoire est beaucoup plus lente.
- **Gestion des entrées/sorties** : le SE permet d'unifier et de contrôler l'accès des programmes aux ressources matérielles par l'intermédiaire des pilotes (appelés également gestionnaires de périphériques ou gestionnaires d'entrée/sortie).



Divers rôles du système d'exploitation

machine virtuelle, gestionnaire. de ressource (+ sécurité, fiabilité, efficacité)

- **Gestion de l'exécution des applications** : le système d'exploitation est chargé de la bonne exécution des applications en leur affectant les ressources nécessaires à leur bon fonctionnement. Il permet à ce titre de « tuer » une application ne répondant plus correctement.
- **Gestion des droits** : le système d'exploitation est chargé de la sécurité liée à l'exécution des programmes en garantissant que les ressources ne sont utilisées que par les programmes et utilisateurs possédant les droits adéquats.
- **Gestion des fichiers** : le système d'exploitation gère la lecture et l'écriture dans le système de fichiers et les droits d'accès aux fichiers par les utilisateurs et les applications.
- **Gestion des informations** : le système d'exploitation fournit un certain nombre d'indicateurs permettant de diagnostiquer le bon fonctionnement de la machine.



Autres qualités requises du système d'exploitation

Un système d'exploitation doit se « faire oublier » : la fonction d'un ordinateur est d'exécuter les applications, pas le système d'exploitation.

- Utilisation efficace des ressources
- Fiabilité
- Tolérance aux fautes (du matériel, des utilisateurs, des programmes)
- La qualité de l'interface (en particulier pour les systèmes interactifs)
- Convivialité
- Simplicité d'utilisation
- Documentation
- Bonne intégration au réseau
- Sécurité et protection
- Répertoire étendu des fonctions



Deux modes de fonctionnement de l'OS

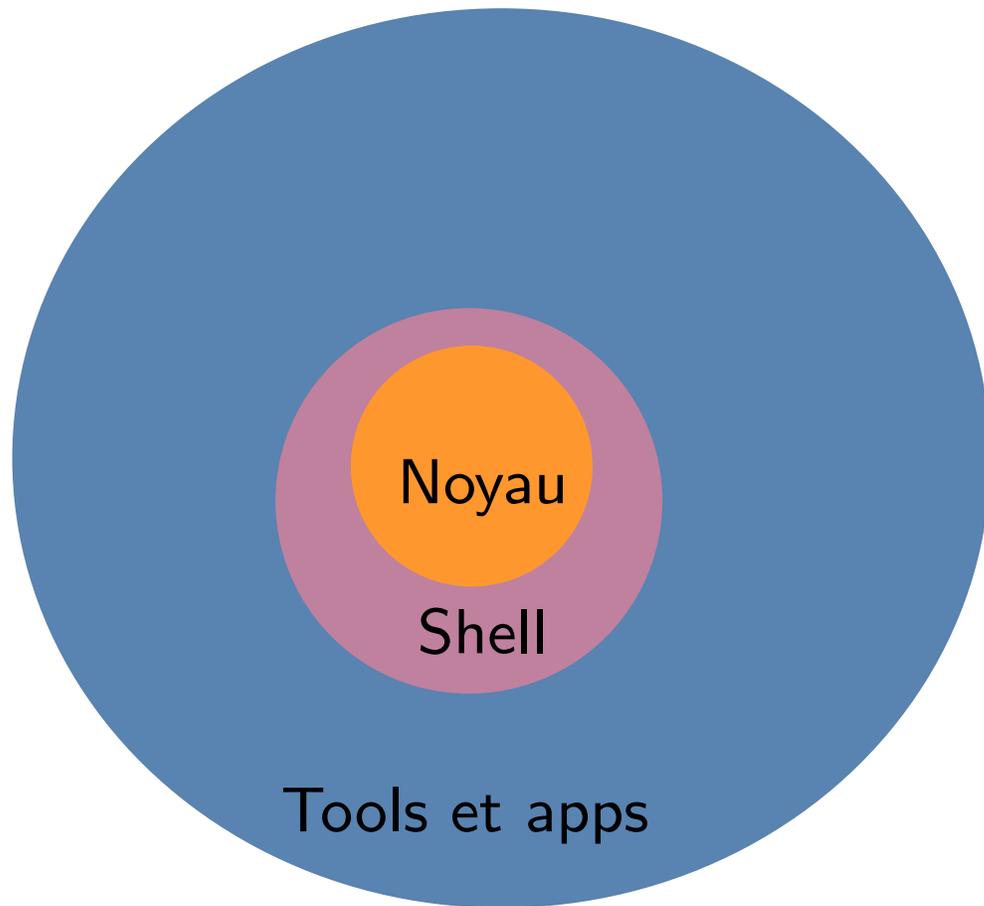
- Le **mode noyau** ou **superviseur** (nécessaire pour l'exécution de certaines instructions qui ont besoin de droits)
- Le **mode utilisateur** (le mode de base, pour des programmes tq un compilateur, un éditeur, programmes utilisateurs...).

⇒ selon le type de programme qu'il a à gérer (programme d'application ou programme système) et des droits, l'OS va le lancer dans un de ces deux modes.



Composants des systèmes d'exploitation

SE : segmentation en couches « d'oignon »

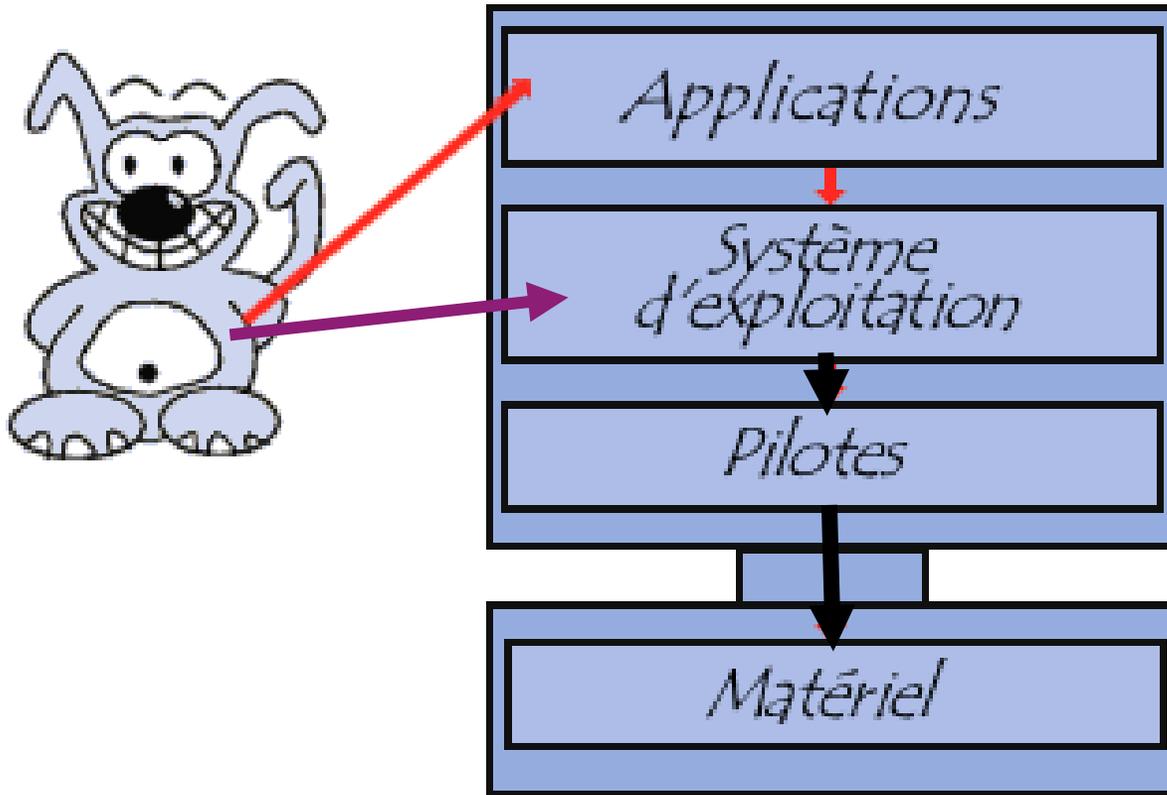


Les parties d'un système Unix.

- Le **noyau** (kernel) représentant les **fonctions fondamentales** du SE, tq la gestion de la mémoire, des processus, des fichiers (système de fichiers), des entrées-sorties principales, et des fonctionnalités de communication.
- L'**interpréteur de commande** (en anglais Shell, traduisez « coquille » par opposition au noyau) permettant la communication avec le système d'exploitation par l'intermédiaire d'un langage de commandes (scripts).
- Un ensemble (plus ou moins minimal) de **tools et d'apps**.



Recap : les différentes couches qui constituent un ordinateur



Votre interaction avec un SE :

- Via les **applications** (en GUI)
- Via le **shell** (langage de commandes)

Bientôt !



Recap : Les fonctions d'un système d'exploitation

Fonctions « visibles »

- Interface utilisateur
- Accès aux périphériques
- Gestion des disques
- Lancement des programmes

Fonctions « cachés »

- Partage du processeur
- Partage de la mémoire
- Gestion d'événements
- Mécanismes de synchronisation entre les programmes