

Introduction aux systèmes

V. Berry

Polytech - Université

Montpellier



Fondamentaux de l'Architecture et du Système des ordinateurs

C. Tibermacine

V. Berry

C. Fiorio

Q. Péntek

M. Morand

Objectif :
éviter ça ->



De



à

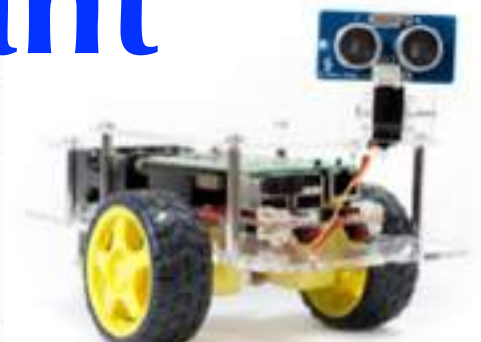


By Argonne National Laboratory's
Flickr page

Connaissances

- Architecture : composants & circulation de l'information
- Représentation des données en mémoire
- Entrées/Sorties
- Mécanisme d'interruption
- Bibliothèque, pilote

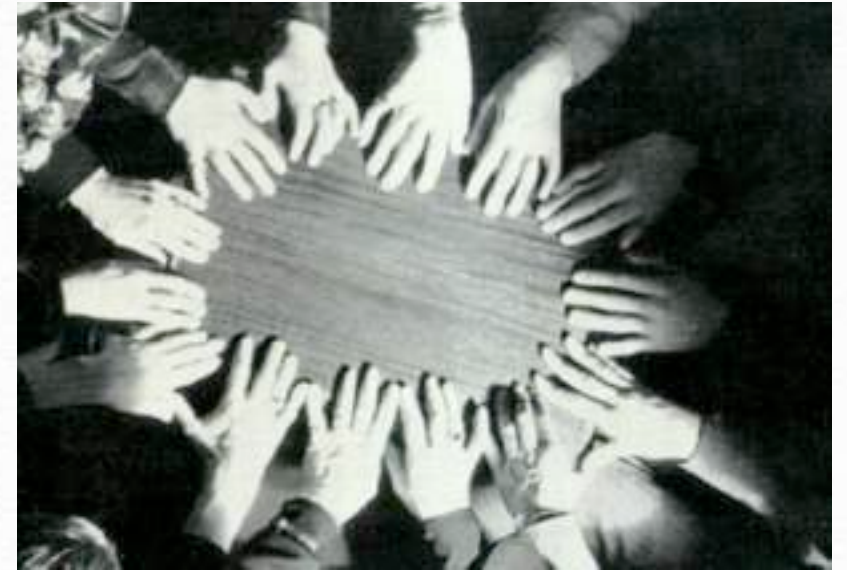
en passant
par



Compétences

- Développer des applications utilisant les appels systèmes
- Utiliser & créer des bibliothèques systèmes
- Administrer un système
- Interfacer des capteurs électroniques
- Mener à bien un projet

Les premières



- Cours — Introduction à l'architecture des ordinateurs
- **Cours — Introduction aux systèmes d'exploitation**
Focus sur la gestion des fichiers
- Cours — le S.E. Unix
- TP : Terminal - Commandes de base - Chemins
- TP : Droits - Archives - Filtres
- TP : programmation shell + commandes réseau
- Cours Programmation shell + commandes réseau
- CM/TP — connexion à distance + administration système + introduction au Raspberry



PLAGE



Introduction aux Systèmes d'exploitation

Introduction

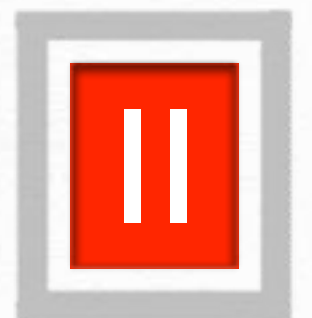
Objectif de **l'ordinateur** : manipuler de **l'information** (d'où «informatique»)

Dans un ordinateur,
l'information peut

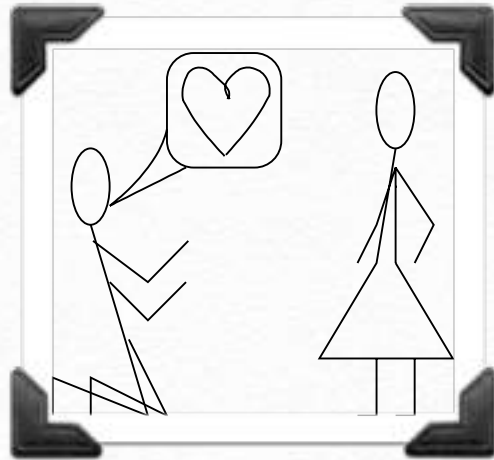
- avoir différents **types** :
programme, texte, son, vidéo, ...
- subir différents **traitements**
 -
 -
 -
 -



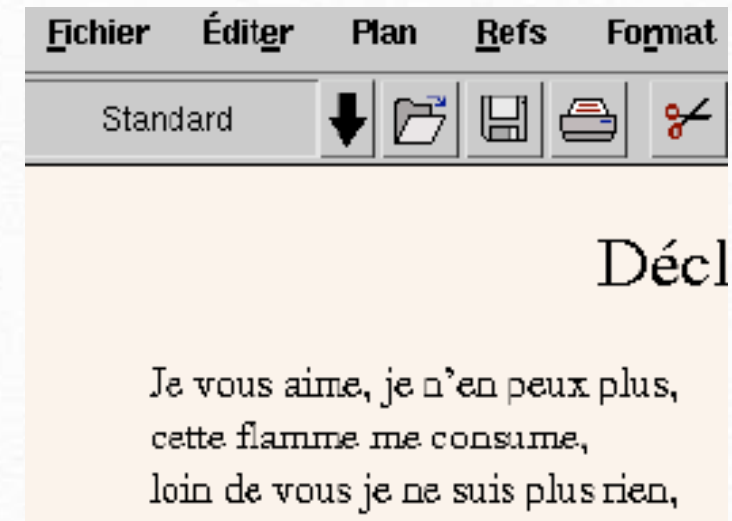
Bit.Code (Julius Popp)



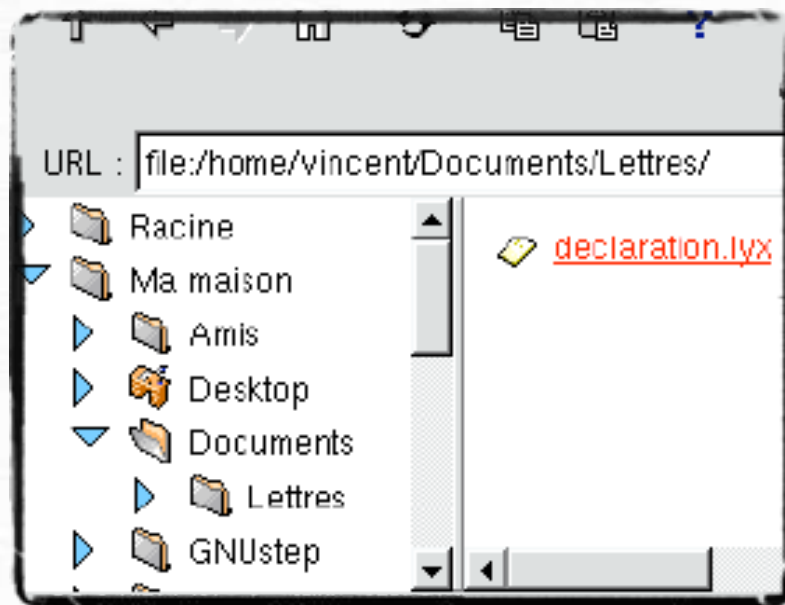
L'information a différents **aspects** suivant son niveau de représentation / l'**acteur** qui la considère



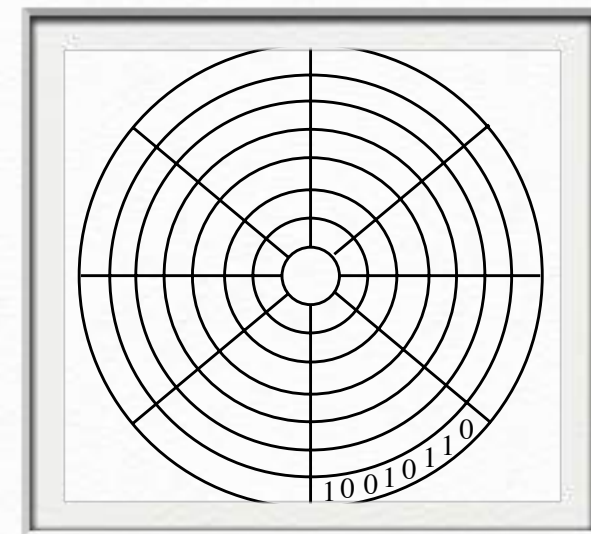
Sémantique : sens que l'information a pour les **humains**, forme (senti)mentale.



Structure : cette information peut être traduite dans un document structuré en **mots, phrases, paragraphes** écrit grâce à un **logiciel**.



Logique : ce document est **stocké** dans un **fichier**, repéré par un **nom /dossier** grâce au **Système d'Exploitation**



Physique : le fichier est stocké en **binaire** dans des **secteurs** (parfois dispersés) sur des pistes magnétiques du **disque**

Systeme d'exploitation

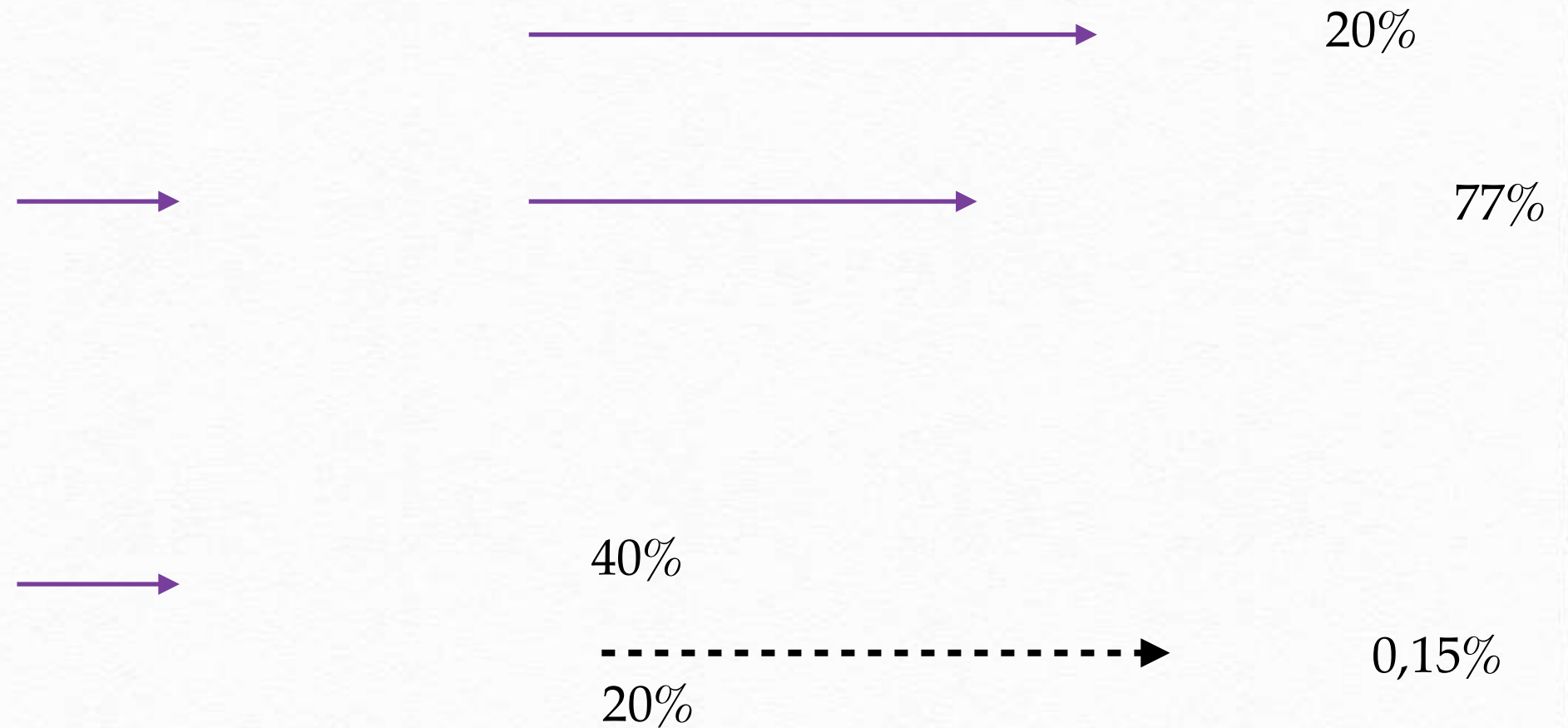
Objectif du **systeme d'exploitation** : faciliter les **traitements** de l'information dans un ordinateur

Un utilisateur n'interagit pas directement avec le matériel

Il lui faut un système qui

- ★ permet aux programmes d'utiliser les périphériques, le réseau, ...
- ★ tourne en permanence dès le démarrage de l'ordinateur
- ★ offre un classement **logique** des informations stockées
- ★ peut dialoguer avec les humains (interface)
- ★ permet l'exécution et le développement de programmes

Quelques systèmes d'exploitation



suivant les matériels / utilisations :

Serveurs

Ordinateurs de
bureau (PC)

Tablettes

Smartphone

SE des développeurs

(enquête Stack Overflow 2018)

Windows 52%

MacOS 26%

Unix : 22%

2016

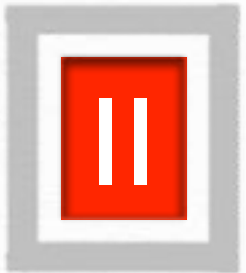
Windows 50%

MacOS 27%

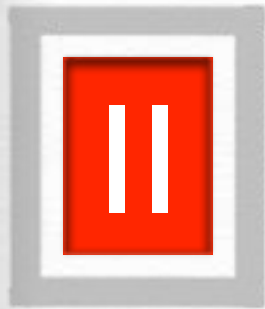
Unix : 23%

2018

Le S.E. par l'exemple



Après la mise sous tension de
ordinateur, à quel moment a lieu
votre **premier contact** avec le
Système d'Exploitation ?



Rôle du S.E. dans cette interaction :



✱ Affiche son bureau



réseau local

sécurité

@CRIP



Interaction utilisateur - S.E. - matériel

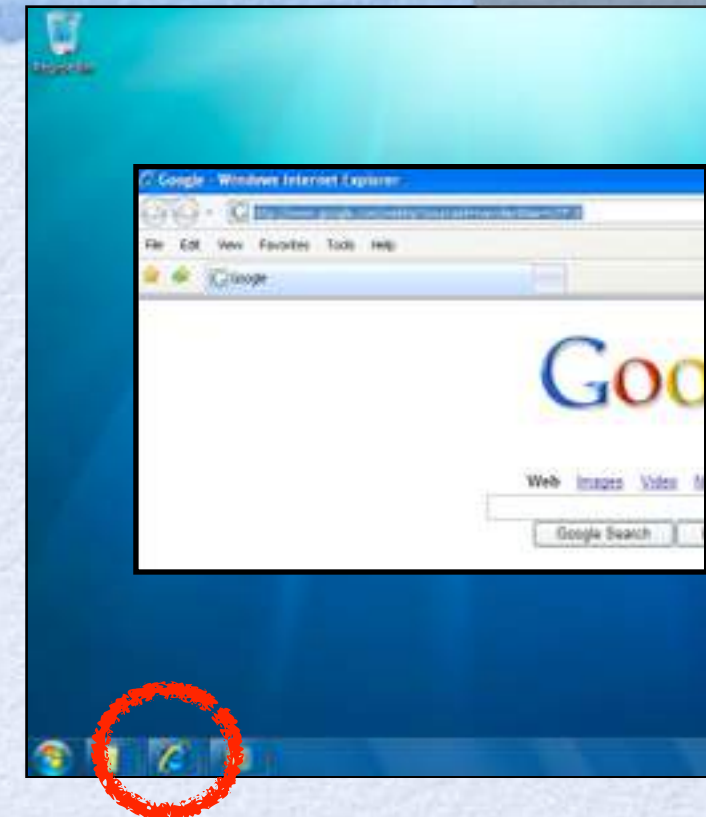


Autre exemple d'interaction

quand intervient un S.E. ?

Si maintenant double-clique sur l'icône du (qui est une)

1. le ... charge le en
2. le envoie **le code** du dans le et affiche une fenêtre à l'écran
3. le demande au de le connecter à une page web distante
4. en pilotant la **carte**, le négocie la connexion et l'échange d'informations avec le site distant, puis affiche le contenu du site



Définition d'un SE

Il est impossible de définir simplement un S.E., celui-ci joue **plusieurs rôles** que nous allons essayer de cerner.

- chef d'orchestre



- gestionnaire

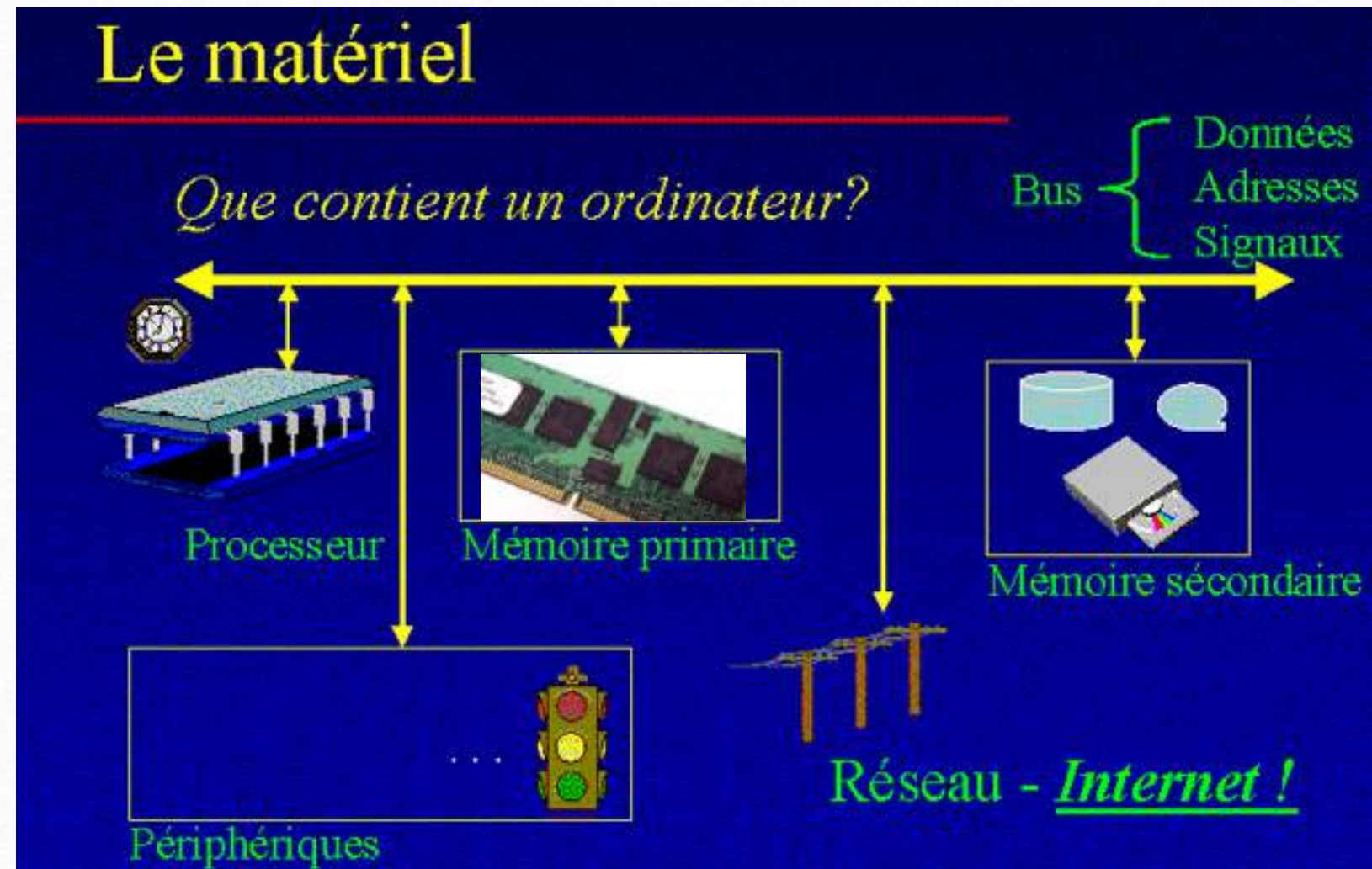


- interprète





Chef d'orchestre



- Sous la baguette du SE que l'information transite entre composants matériels
- Si panne ou erreur matérielle, le **SE** doit trouver des solutions
- C'est le SE qui est en charge de coordonner les composants de l'ordinateur

Gestionnaire



Un S.E. ne produit aucune information ou ressource.

Comme un *gouvernement*, il **gère les ressources** en arbitrant les accès concurrents, il définit les règles d'accès

Plus précisément, il aura en charge la gestion

★ système multitâches

- des **programmes** en cours d'exécution : chargement en mémoire, accès au processeur, synchronisation

- de l'organisation de l'espace **mémoire** et de l'organisation des informations sur **disque**

★ système multiutilisateurs

- des **utilisateurs** : identification, protection des informations, zones de partages d'information

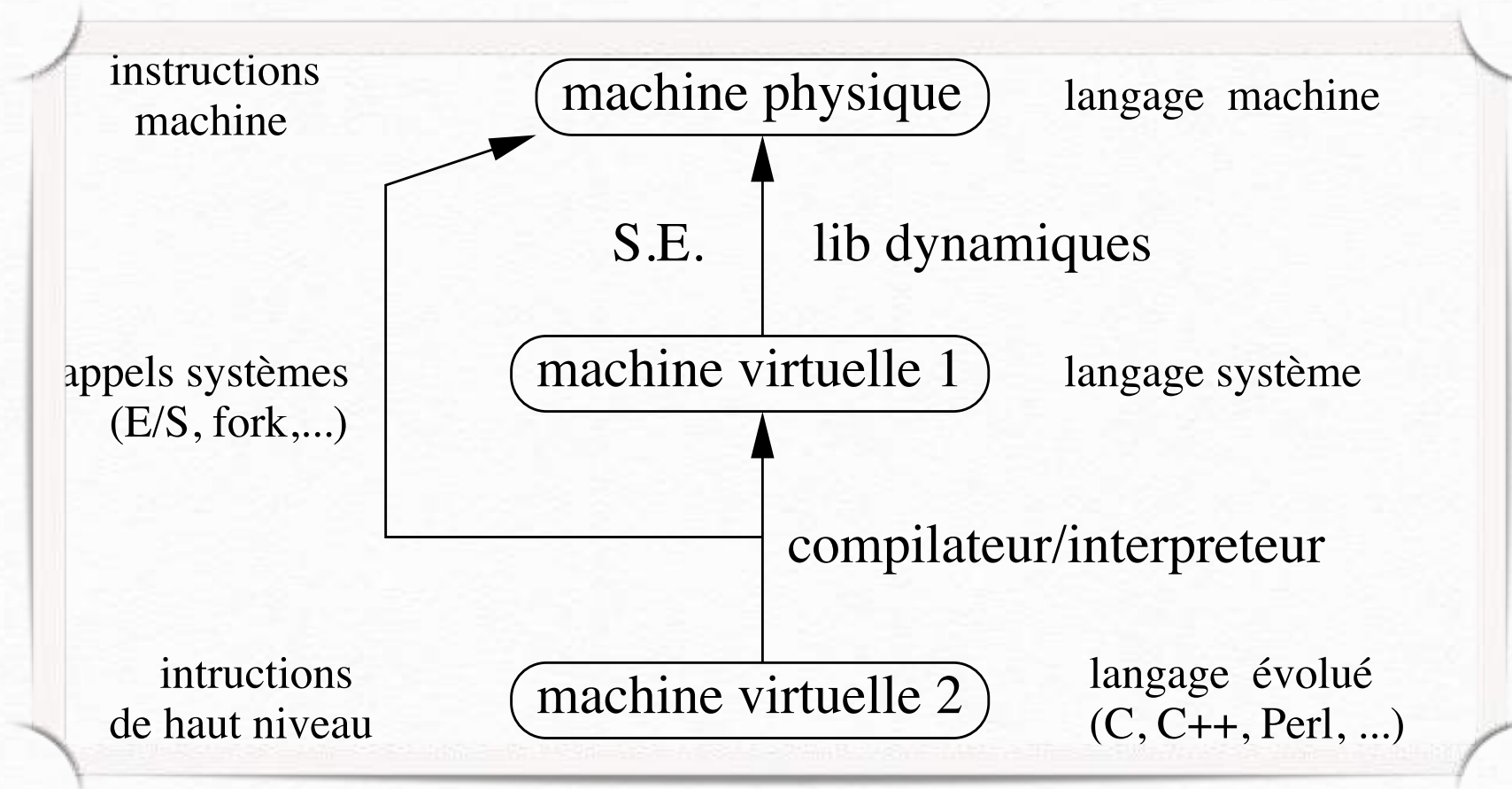


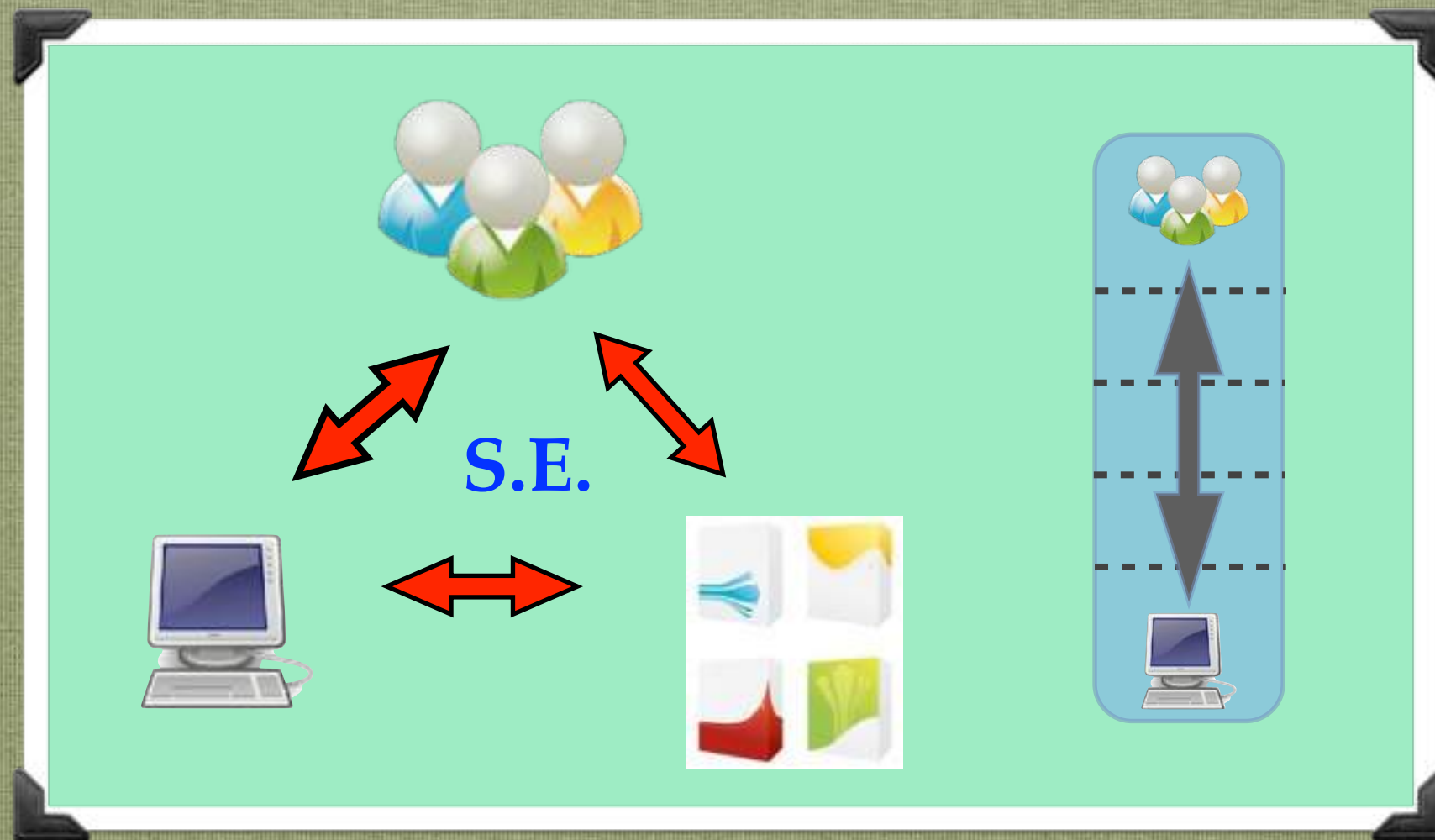
Interprète



Le S.E. a pour rôle de nous affranchir de la complexité du matériel, il sert de **traducteur** en proposant un/des langage/s de plus haut niveau.

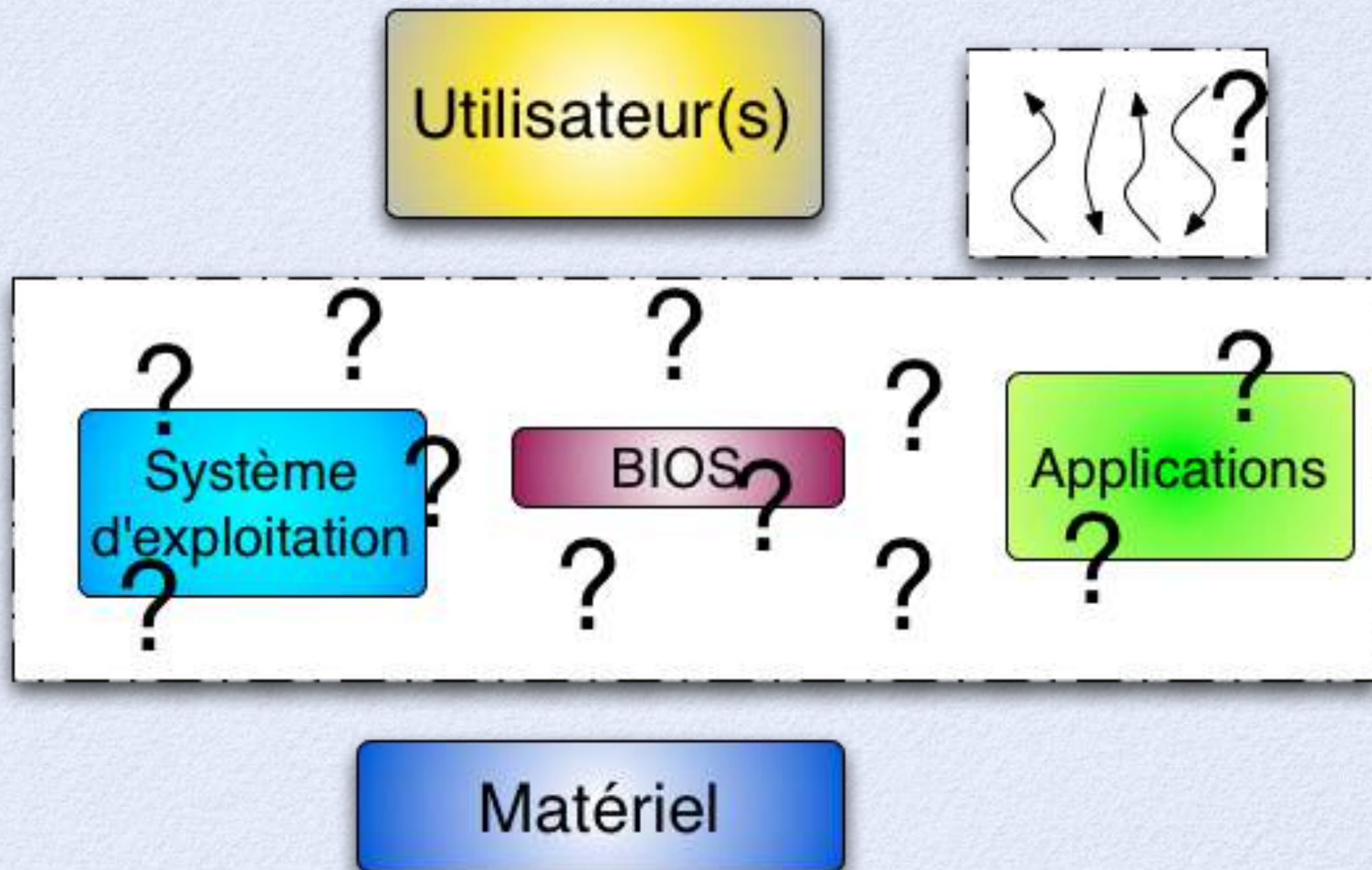
Les applications de niveau supérieur communiquent au S.E. leurs intentions par le biais d'**appels systèmes**





Interactions entre acteurs du
poste de travail

PLACE DU S.E. DANS LE FONCTIONNEMENT D'UN ORDINATEUR



COUCHES D'ABSTRACTION

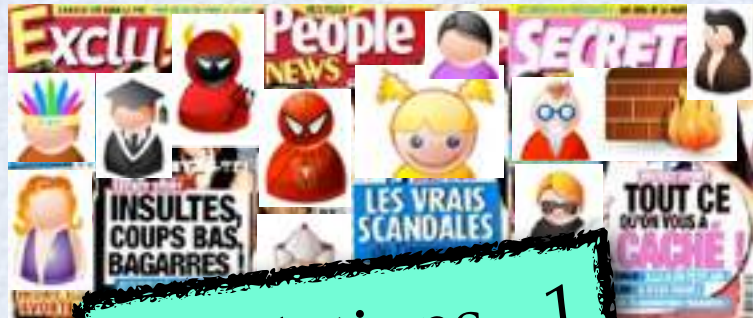
L'informatique
pour les nuls
(niveau bac+2)

niveau abstrait



niveau concret





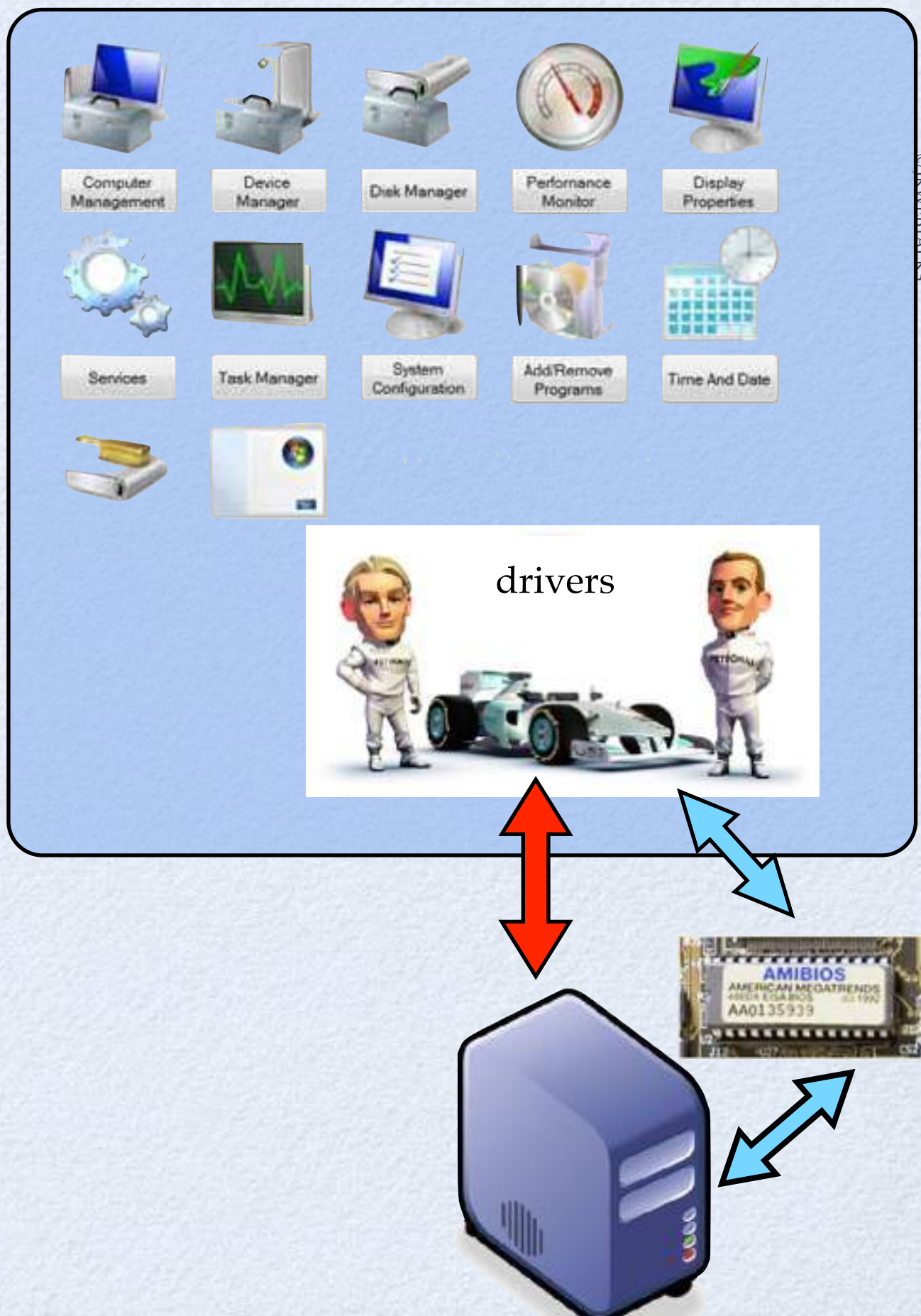
Révélation - 1

- « Le » système d'exploitation est en fait un ensemble de programmes !!!! (ta daaaa !)
- Exemple : les pilotes (drivers) sont en fait ... de petits programmes qui gèrent les périphériques, et parfois directement sans passer par le BIOS (oh !)

« Do you miss the nice days of Minix-1.1, when men were men and wrote their own device drivers?"

Linus Torvalds (ouch !)

Systeme d'exploitation





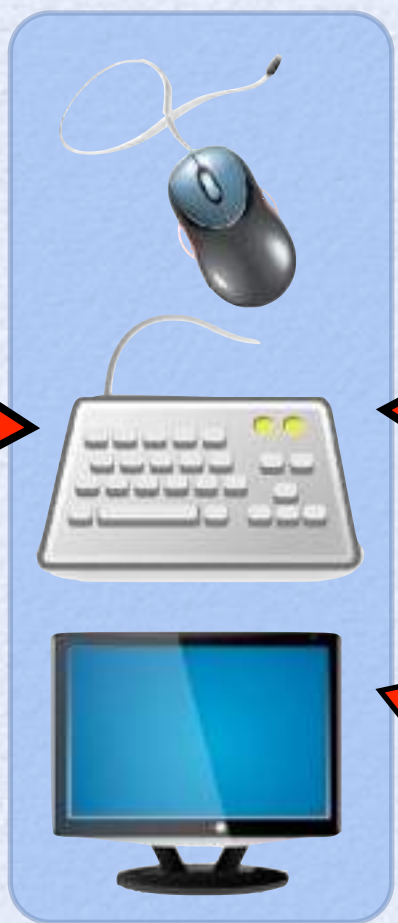
Révélation - 2

INTERACTIONS

elles sont bien plus nombreuses que dans la hiérarchie simple !
périphériques E/S



utilisateurs



B.I.O.S



22



S.E.

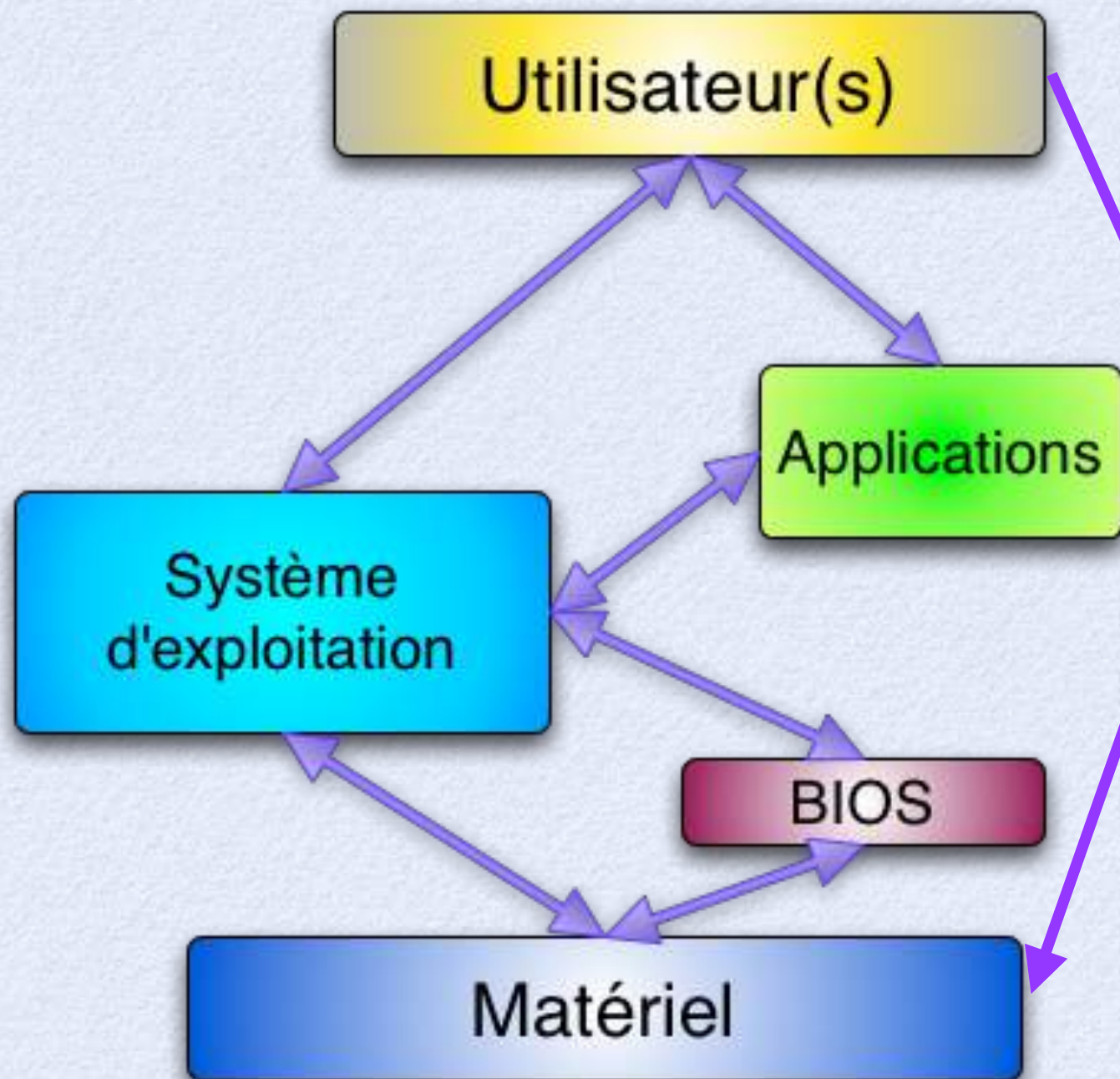


composants principaux

INTERMEDIATION PLACE DU S.E.

SPOILER

Quizz séance 3



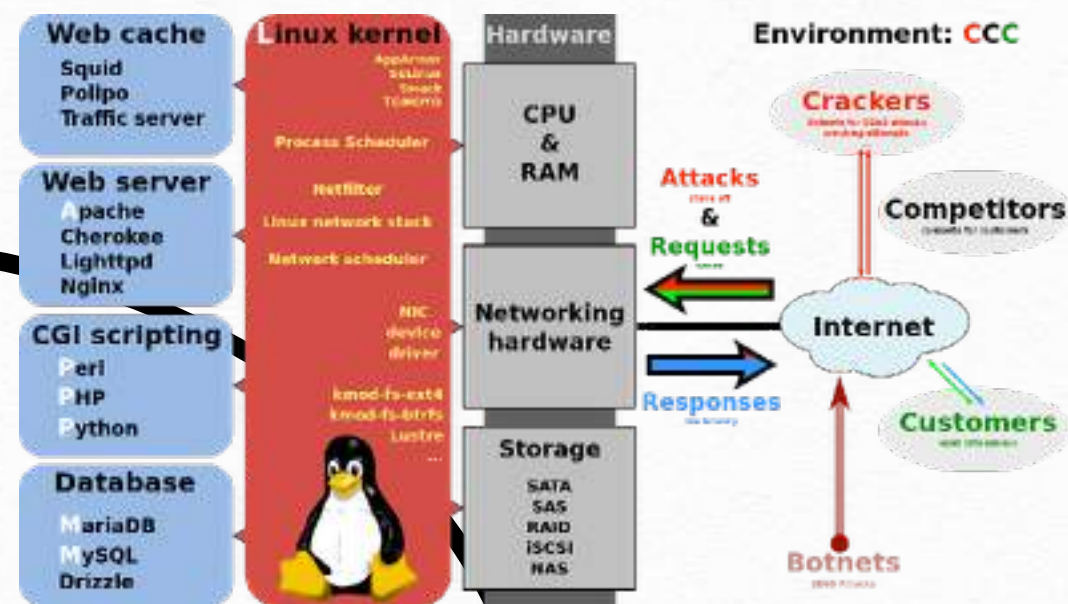


Architecture d'un système Unix

Pourquoi Unix ?



Calculateurs,
Data centers



Serveurs/Platerformes Web



Salles TP



Raspberry



UNIX

John

emacs

Applications

Shells

Tyrion

top

Appels système

Noyau

Matériel

open office

Cersei

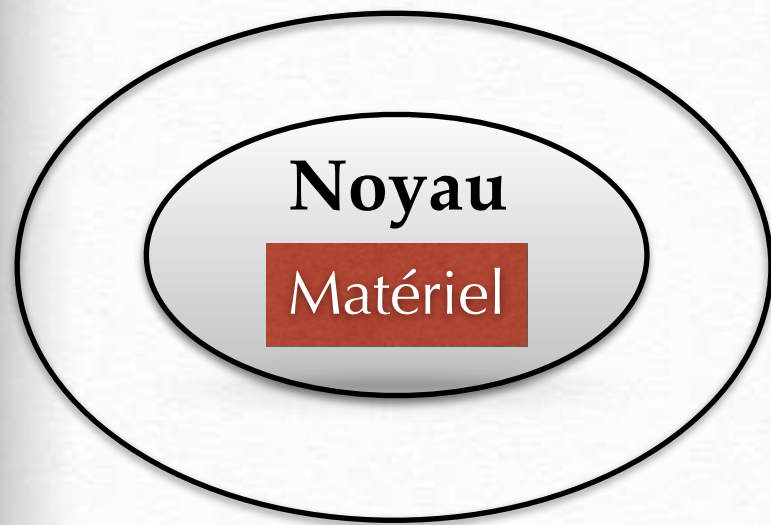
Bibliothèques systèmes

a.out

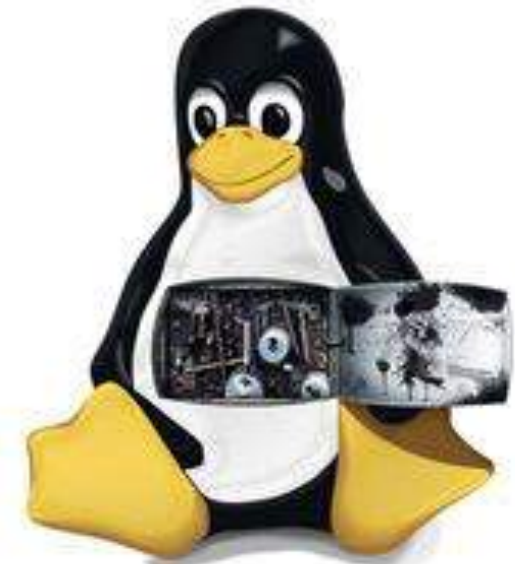
python

Arya

Daenerys



Le noyau



- Contrôle le matériel
- Sert d'intermédiaire pour accéder au matériel
- Gère les ressources :,,
- Propose des abstractions (structures de données) :
 -
 -
 -
- Assure la sécurité et la protection :
 - de contre
- Répond aux demandes des
(effectuées des appels systèmes)




```
Terminal — bash — 79x24
ls /usr/local/
ite          mysql
            mysql-5.1.28-rc-osx10.5-x86
lude        share
            texlive

find /usr/local/share -name README -print 2>/dev/null
r/local/share/ghostscript/8.57/doc/README
r/local/share/ghostscript/fonts/README
r/local/share/taxomanie/ol-taxo/README
r/local/share/taxomanie/taxomanie-1.0/README
grep -E ".htm" /usr/local/share/ghostscript/*/*/README
Deprecated.htm
Devices.htm
    Helpers.htm
Humor.htm
Ps2epsi.htm
    Ps2pdf.htm
Ps2ps2.htm
Readme.htm
Unix-lpr.htm
Use.htm
Changes.htm
Commprod.htm
Fonts.htm
```

Shell et terminal de commandes

Le terminal de commandes



Un **terminal de commande** est une application qui permet à l'utilisateur d'interagir avec le SE par le biais de **commandes** puissantes.

Cette application permet de lui demander des **traitements sophistiqués** que l'on ne peut réaliser par le cliquodrome de l'interface graphique.



Le terminal de commandes



- il exécute une boucle infinie répétant :
1. lire une **commande** de l'utilisateur
 2. exécuter la **commande**
 3. indiquer le résultat de la **commande**

The diagram illustrates the process of command execution in a terminal. It features a terminal window titled 'Terminal — bash — 66x14' showing the output of a 'find' command. To the left of the terminal, there are three numbered steps (1, 2, 3) with arrows pointing to the corresponding parts of the terminal output. Step 1 points to the prompt '~ >', step 2 points to the command 'find /usr/local/share -name README -print 2>/dev/null', and step 3 points to the list of file paths. A blue circle highlights the prompt '~ >' with an arrow pointing to the text 'invite de commande'.

```
Terminal — bash — 66x14
~ > find /usr/local/share -name README -print 2>/dev/null
/usr/local/share/ghostscript/8.57/doc/README
/usr/local/share/ghostscript/fonts/README
/usr/local/share/taxomanie/ol-taxo/README
/usr/local/share/taxomanie/taxomanie-1.0/README
```

invite de commande

Les premières commandes

Format des commandes :

★ **nomCde** **[-option(s)]** **[argument(s)]**

Exemples:

date

whoami

affiche le nom de l'utilisateur connecté

pwd

affiche le nom du
répertoire courant

ls -l

liste le contenu
d'un répertoire
de façon détaillée

man <cde>

manuel en ligne

espace

espace

espaces
entre chaque option,
entre chaque argument !

```
- > ls
Desktop      ExaChess Games  Music      Public
Documents    Library        Pictures   SVG
Downloads    Movies         Prog       Sites
- > ls -l
total 0
drwx-----+ 47 vberry  staff  1598  7 sep 22:20 Desktop
drwx-----@ 18 vberry  staff   612  8 août 19:50 Documents
drwx-----+ 21 vberry  staff   714  7 sep 21:34 Downloads
drwxr-xr-x  15 vberry  staff   510 27 jul 15:17 ExaChess
drwx-----+ 45 vberry  staff  1530  7 sep 22:25 Library
drwx-----+  8 vberry  staff   272  4 jul 16:14 Movies
drwx-----@  8 vberry  staff   272 26 août 13:20 Music
```


Le terminal de commandes

Commandes :

cat nom_d'un_fichier

affiche le contenu du fichier

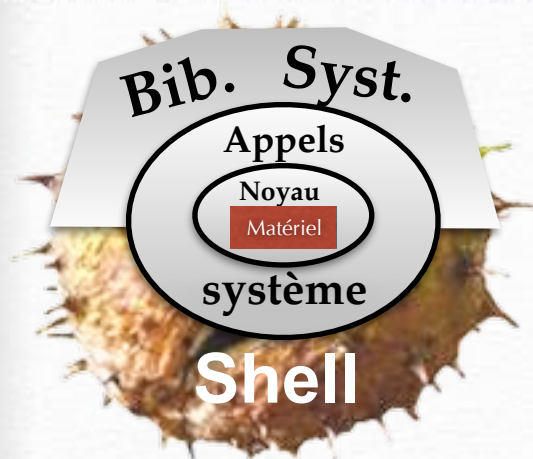
ls

liste le contenu d'un répertoire

- 5mn de détente : éprouvez cette sensation d'éternel recommencement due à la boucle infinie se déroulant dans un terminal :



Sur https://www.tutorialspoint.com/unix_terminal_online.php en lui demandant la date et en lisant le fichier README.txt



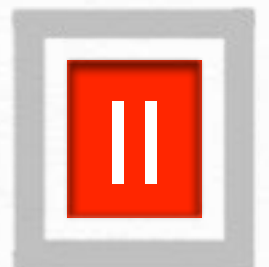
Shell

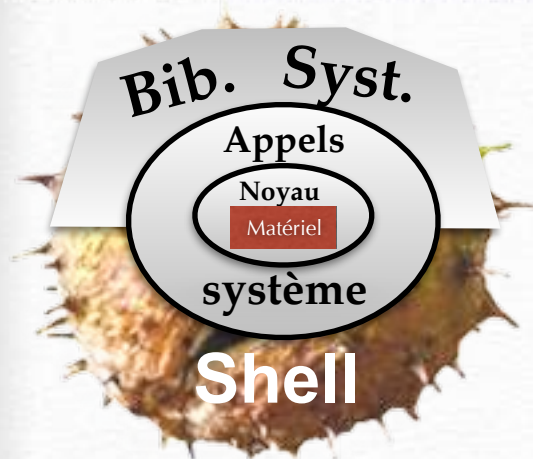


- ▶ Agit comme une interface entre les utilisateurs et le noyau du S.E.
- ▶ Permet à l'utilisateur de demander des services au S.E. sans avoir à réaliser de programme
- ▶ Interprète les commandes tapées par un utilisateur (dans un CLI (.....)) et s'arrange pour que le S.E. les réalise. Les commandes sont elles mêmes de petits programmes !
- ▶ Il existe de nombreux Shells: sh, bsh, csh, tcsh, bash, oh my zsh!
- ▶ Chaque utilisateur a un shell par défaut

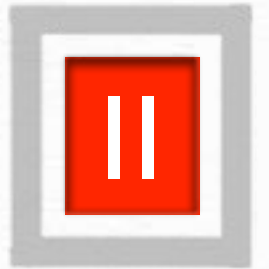


cat /etc/passwd pour voir quel shell est associé aux utilisateurs « root » et « cg » sur www.tutorialpoint.com





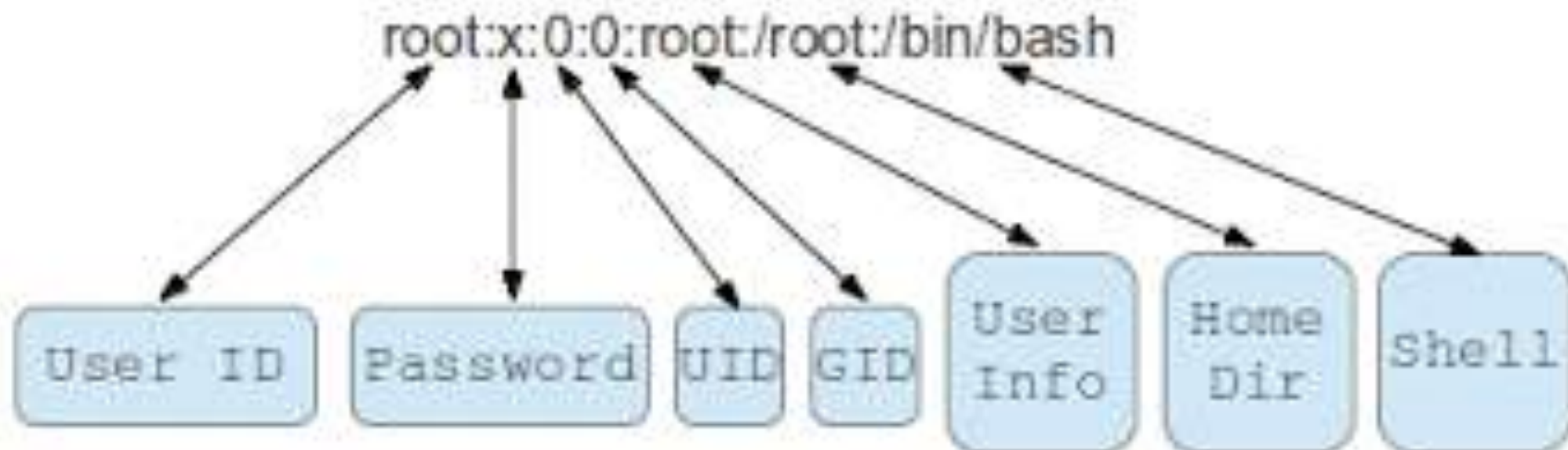
Shell



- Il existe de nombreux Shells: sh, bsh, csh, tcsh, bash, oh my zsh!
- Chaque utilisateur a un shell par défaut (celui qui pilotera son CLI).



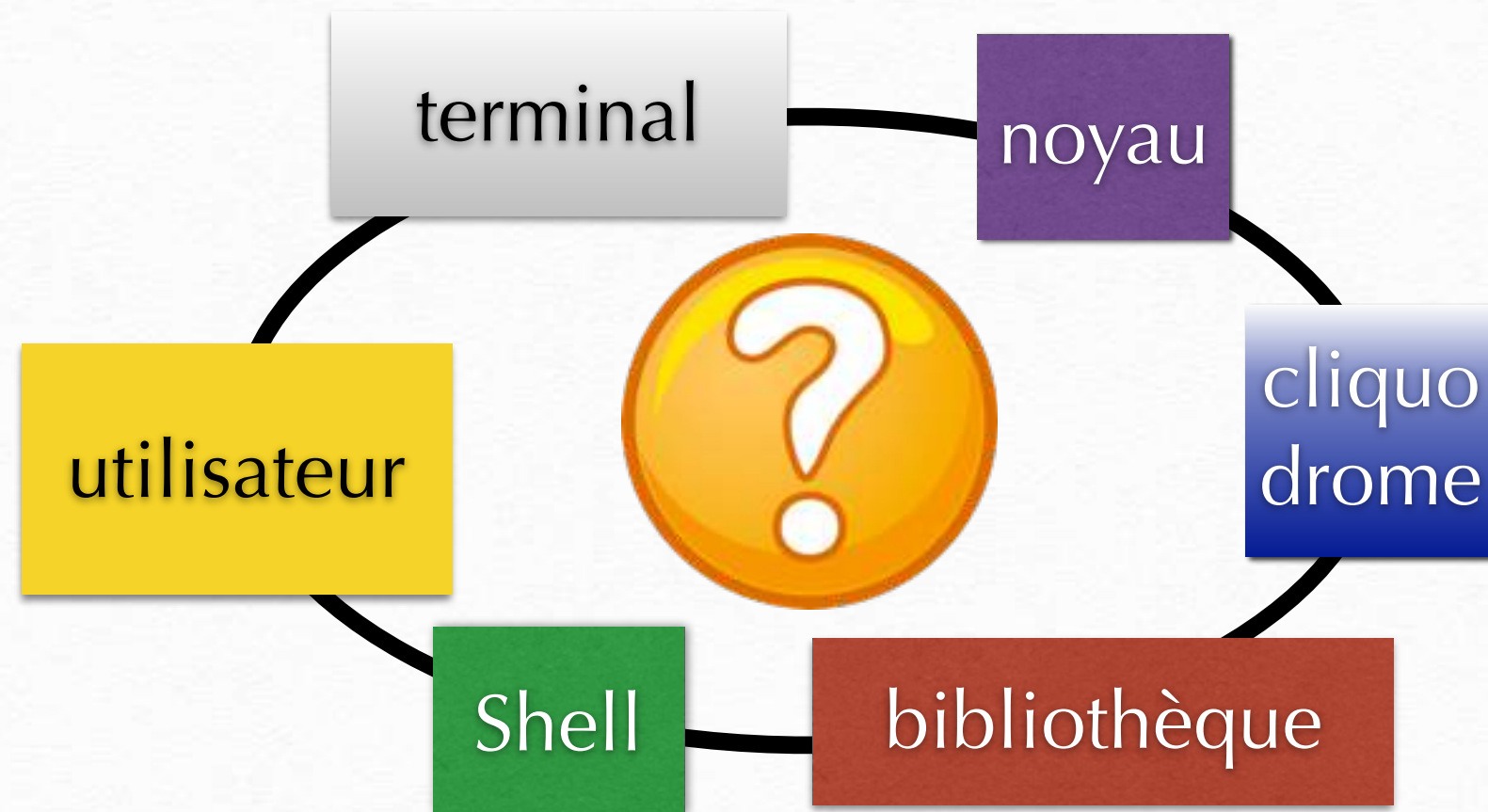
cat /etc/passwd pour voir quel shell est associé aux utilisateurs « root » et « cg » sur www.tutorialpoint.com



Vous suivez ?



► Qui interagit avec qui ?



► Classez les différentes interfaces qu'on peut utiliser pour demander un service au S.E. **du plus orienté utilisateur au plus orienté développeur**