

## Module Master M1

# Systèmes temps réel et Informatique Industrielle

## Chapitre 0 : Rappel sur le système d'exploitation GNU-Linux

Présenté par : Prof. Krolladi Mohamed-Khireddine  
 Département d'Informatique  
 Facultés des Sciences Exactes  
 Université Echahid Hamma Lakhdar d'El Oued  
 Tél. 0770314924  
 Email. kholladi@univ-eloued.dz et kholladi@yahoo.fr  
 Site Web. www.univ-eloued.dz  
<http://kholladi.doomby.com/> et <http://kholladi.e-monsite.com/>



### 0.0 - Introduction

En principe, un ordinateur est constitué de matériels (Hardware) et de programmes ou logiciels (software). Mais, il existe différents types d'ordinateurs, d'applications et de systèmes d'exploitation de différentes tailles:

Super calculateurs	Serveurs	Ordinateurs Individuels (ou PC)	Mobiles (ou Smartphones)
<ol style="list-style-type: none"> <li>1. Prévisions météo</li> <li>2. Prévisions financières</li> <li>3. Simulations physiques</li> <li>4. La recherche scientifique</li> </ol>	<ol style="list-style-type: none"> <li>1. Réseaux et stations de travail</li> <li>2. Les centres de données</li> <li>3. Les entreprises et les institutions</li> </ol>	<ol style="list-style-type: none"> <li>1. Bureautique</li> <li>2. Conception, Gestion</li> <li>3. Jeux, loisirs</li> </ol> <p><b>Applications de bureau</b></p>	<ol style="list-style-type: none"> <li>1. Les applications sur vos mobiles phones et tablettes</li> </ol> <p><b>Applications mobiles</b></p>
<ol style="list-style-type: none"> <li>1. Unix (open source)</li> <li>2. Windows server de Microsoft</li> </ol>	<ol style="list-style-type: none"> <li>1. Windows server de Microsoft</li> <li>2. Unix (open source)</li> </ol>	<ol style="list-style-type: none"> <li>1. Windows de Microsoft</li> <li>2. Mac OS X d'Apple</li> <li>3. Linux (open source)</li> </ol>	<ol style="list-style-type: none"> <li>1. Ios d'Apple</li> <li>2. Androïde de Google (open source)</li> </ol>

Ici, il s'agit de donner les principes essentiels à la compréhension du fonctionnement des ordinateurs sous les utilitaires GNU (Richard Stallman, 1983) et le noyau Linux (Linux Torvalds, 1991). Le concept du système d'exploitation GNU/Linux est en principe, un modèle de liberté reposant sur quatre règles essentielles des logiciels libres :

1. Assurer l'interopérabilité (formats ouverts, respect des normes) → Interopérabilité;
2. Viser le partage et l'échange → Communicabilité ;

3. Proposer une meilleure sécurité → Protection et Accessibilité;
4. Et pouvoir se déployer partout (serveurs, ordinateurs, supercalculateur, embarqué, etc.)  
→ Portabilité et robustesse.

### **a - Pourquoi utiliser GNU/Linux?**

1. Gratuité (mais pas toujours!);
2. Besoin;
3. Nouveauté;
4. Sécurité ;
5. Stabilité;
6. Et liberté.

### **b - Exemples de distribution GNU/Linux**

1. Bases : (Archlinux, Debian, Gentia, Red Hat, Slackware)
2. Dérivés (eOS, Fedora, HandyLinux, Mageia, Manjaro, Mint, openSUSE, Trisquel, Ubuntu)

### **c - Exemples de logiciels**

1. Libres (Trisquel, Firefox, VLC, Python, LibreOffice, etc.).
2. Non libres (Windows, Excel, Photoshop, Vegaspro, Steam, etc.).

Ce qu'il faut retenir:

1. Il existe d'autres systèmes que Windows ou Mac OS X;
2. Les systèmes GNU/Linux sont parfaitement accessibles;
3. Ils entendent respecter certains principes (open source, libre);
4. Ils doivent répondre uniquement aux besoins des utilisateurs;
5. L'univers GNU/Linux est loin d'être parfait;
6. Faire cohabiter le monde du libre avec le monde non libre est possible;
7. Etc.

Quelques ressources intéressants à voir:

1. Framasoft ⇒ <https://framasoftware.org>
2. Distrowatch ⇒ <https://distrowatch.com>
3. LinuxFr ⇒ <https://linuxfr.org>
4. Documentation Ubuntu ⇒ <https://doc.ubuntu-fr.org>

## 0.1 - Définition de système d'exploitation

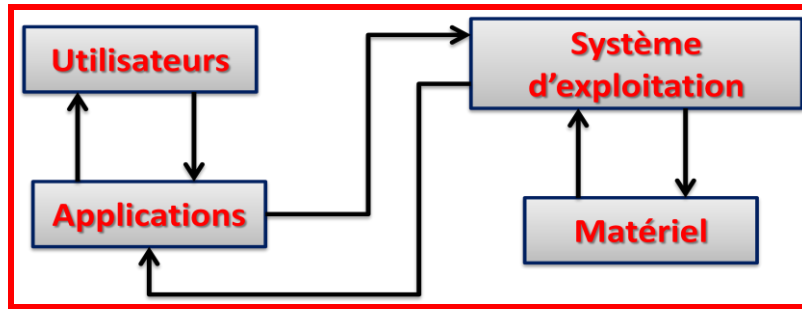
C'est un ensemble de programmes exécuté par le processeur central d'un outil informatique assurant la gestion de l'ordinateur et de ses périphériques. C'est-à-dire le démarrage et le bon fonctionnement de l'ordinateur. Il sert d'interface entre le matériel (hardware) et le logiciel (software). Le système d'exploitation va gérer (comme sur la figure 0.1) :

1. La mémoire;
2. Les accès aux périphériques;
3. Le processeur;
4. Le contrôle fiabilité du système;
5. Et la sécurité.



*Figure 0.1 – Exemple de système d'exploitation*

Le système d'exploitation est le premier programme exécuté lors du démarrage de la machine, car son rôle est de suivre des instructions. Du coup lors du démarrage, c'est le système d'exploitation qui va guider la machine pour exécuter les premières opérations et ce qu'il faut affiché (grâce essentiellement au programme amorce de la mémoire morte de la machine, qu'on appelle la ROM). C'est seulement qu'après que le système d'exploitation va donner la main à l'utilisateur pour utiliser la machine. Ensuite au-delà du démarrage, le système d'exploitation va assurer la liaison entre les ressources matérielles et les applications qui tournent dessus. Ainsi, le système d'exploitation est responsable de la bonne marche des applications via l'affectation des ressources nécessaires à leur bon fonctionnement. En fait, les utilisateurs interagissent avec des applications et celles-ci vont communiquer avec le système d'exploitation comme sur la figure 0.2.



*Figure 0.2 – Interaction Utilisateurs Machine via des applications et le système d'exploitation*

Néanmoins, il est possible pour les utilisateurs de solliciter directement des services au système d'exploitation via par exemple une interface graphique, des lignes de commande ou des interfaces de programmation comme sur la figure 0.3.



*Figure 0.3 – Interaction via Interface graphique et lignes de commande*

## 0.2 - Fonctionnalités d'un système d'exploitation

Le système d'exploitation permet :






1. D'assurer le bon fonctionnement de l'ordinateur et le contrôle de ses différents composants;
2. De faciliter la communication entre l'utilisateur et le matériel de l'ordinateur;
3. De mettre une liaison entre les logiciels installés et le matériel de l'ordinateur;
4. De gérer la mémoire et enregistrer les informations sur le disque dur;
5. Et d'installer, lancer et désinstaller des logiciels dans l'ordinateur.





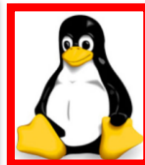





*Figure 0.4 – Portails d'entrée des systèmes d'exploitation*

### 03 - Exemples de systèmes d'exploitation

Il existe plusieurs exemples de systèmes d'exploitation:

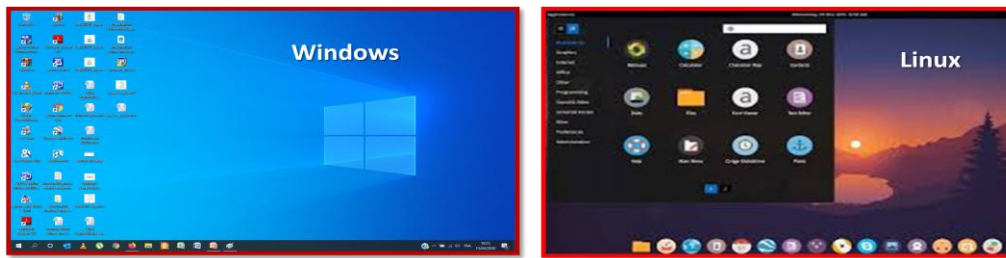
N°	Systèmes d'exploitation	Sigles
1	MS DOS (Microsoft Disk Operating System)	
2	Windows (95, 98, 2000, 2003, 7, 2010, 8, ...10)	
3	Linux	
4	Unix	
5	Mac OS X	
6	Etc.	

On a aujourd'hui :

Windows, Mac OS et Linux pour les ordinateurs	  
Windows server et Linux ou Unix pour les serveurs	  
iOS et Android pour les Smartphones et les tablettes	 

### 0.4 - Environnement d'un système d'exploitation

Au démarrage de l'ordinateur, le système d'exploitation affiche une interface graphique permettant la communication avec l'utilisateur qu'on appelle Bureau (Desktop). Le bureau, c'est l'écran d'accueil d'un système d'exploitation (comme sur la figure 0.5).



*Figure 0.5 – Exemple de bureaux de systèmes d'exploitation*

Le bureau contient plusieurs éléments :

1. Les icônes : ce sont les petites images affichées sur le bureau et qui représentent des fichiers, des dossiers ou des programmes;
2. Le bouton démarrer: c'est un bouton situé sur la barre des tâches et permet d'afficher un menu contenant toutes les informations enregistrées ou installées sur le disque.

Le bureau contient plusieurs éléments comme sur figure 0.6 :

- La barre des tâches : elle se trouve en bas de l'écran. Elle contient le bouton démarrer, la zone de notification et les boutons des programmes en cours d'exécution;
- La zone de notification : elle se trouve en bas à droite de l'écran et contient des petits programmes utiles pour l'utilisateur (la date, l'heure, le volume, la langue, etc.);
- Le menu démarrer : il s'affiche en cliquant sur le bouton "démarrer" et contient tous les documents et les programmes enregistrés sur le disque.



*Figure 0.6 – Bureau des systèmes d'exploitation avec menus et icônes*

## **0.5 - Les principaux composants d'un système d'exploitation**

Le Noyau (Kernel) gère les principales fonctions du système d'exploitation. C'est l'espace mémoire isolé regroupant les fonctions clés du système d'exploitation comme :

1. La gestion de la mémoire;
2. La gestion des processus;
3. Ou la gestion principale des entrées/sorties.

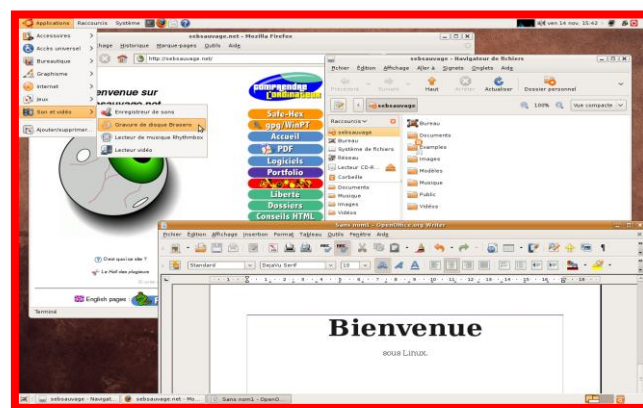
L'interpréteur de commandes (Shell): Il permet de communiquer avec le système d'exploitation par l'intermédiaire d'un langage de commandes. Le système de fichiers (File system) gère l'arborescence, l'écriture et la lecture des fichiers. Ainsi, il permet d'enregistrer des fichiers dans une arborescence mais aussi de les gérer et d'autoriser la lecture et l'écriture des fichiers. Les pilotes (Drivers) permettent la gestion des périphériques (imprimante, souris, webcam, etc.). En fait, chaque périphérique a ses propres instructions avec lesquelles, il peut être manipulé et évidemment le système d'exploitation devra en tenir compte.

## 0.6 - Définition de Linux

C'est un système d'exploitation de type Unix. C'est un système multitâche, multiutilisateur, multiprocesseur et multiplateforme. Il s'agit d'un logiciel open source (libre, ouvert et gratuit). Linux possède deux interfaces une interface graphique et une interface ligne de commande. On a l'habitude de dire que Linux est un système d'exploitation. En réalité, Linux, c'est uniquement le cœur (on dit le "noyau") du système d'exploitation GNU/Linux. (Les systèmes d'exploitation Microsoft Windows NT, 2000 et XP ont également un noyau, mais différent: le noyau NT.). Le noyau s'occupe des basses besognes : la gestion de la mémoire, l'accès aux périphériques (disque dur, lecteur de CD-Rom, clavier, souris, carte graphique, etc.), la gestion du réseau, le partage du temps microprocesseur entre les programmes (multitâche), etc. Contrairement à Windows dont l'interface graphique vous est imposée, il existe différentes interfaces graphiques sous Linux, les principales étant Gnome, KDE et XFCE. Il est même possible de faire fonctionner Linux sans interface graphique, ou même de ne lancer l'interface graphique que quand on le souhaite.

## 0.7 - Interface graphique de Linux

La figure 0.7 illustre l'interface graphique de Linux.



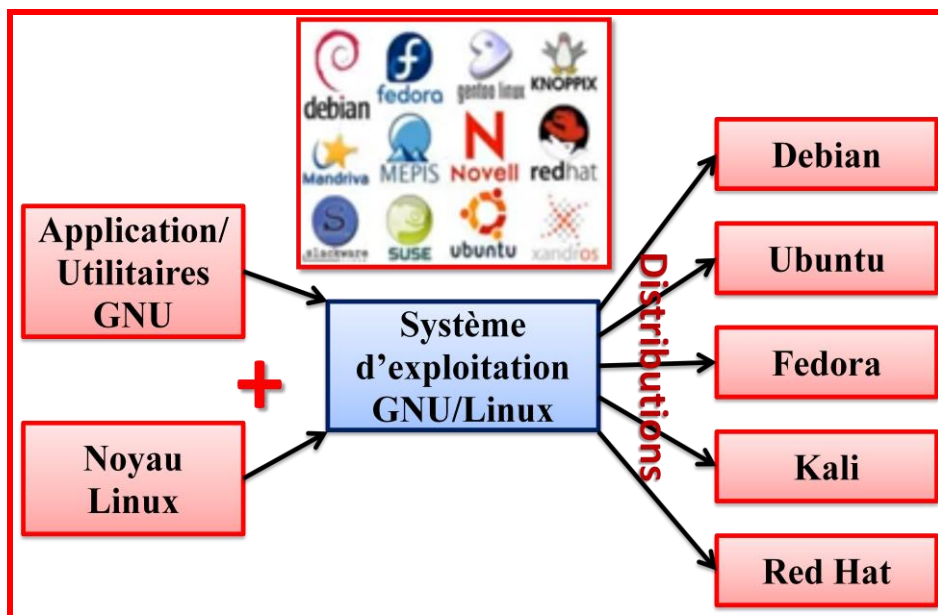
*Figure 0.7 – Interface graphique de Linux*

La figure 0.8 illustre les différentes interfaces de Linux selon les différentes versions de distribution du système d'exploitation linux.



*Figure 0.8 – Les différents types d'interface de Linux*

La figure 0.9 illustre les différentes versions de linux. En fait, Linux n'est pas un système d'exploitation mais plutôt le noyau, une fois associé aux utilitaires GNU, on aura à ce moment un véritable système d'exploitation GNU-Linux, Mais abusivement, on parle de système d'exploitation Linux.



*Figure 0.9 – Les différentes distributions du système Linux*

## 0.8 - Système de fichier sous Linux

Un système de fichier est un système d'organisation des données dans une mémoire de stockage (disque dur (DD), USB, CD, DVD, etc.). Sous Linux tous les fichiers sont organisés suivant une unique arborescence. Les types de fichier sous Linux sont :



- Fichier normal : Données stockées sur disque dure (fichier, image, vidéo, son, etc.),
- Répertoire : Ensemble d'information permettant l'accès à d'autres fichiers,
- Fichier spécial : Dispositifs d'E/S (terminal lecteur, etc.).

## 0.9 - Relation du chemin

Sous Linux, il existe deux manières de trouver le chemin d'un fichier:

- Chemin absolu : C'est la liste des répertoires traversés pour aller de la racine au fichier désiré séparer par "/".

Ex. /home/kholladi/musique

- Chemin relative : C'est la liste des répertoires traversés pour aller au répertoire courant (ou on se situe) le fichier.

Ex. musique/Andaloussi

## 0.10 - Notation spéciale

Répertoire	/
Personnel	~
Courant	.
Parent	..

La figure 0.10 illustre la ligne de commande de linux.

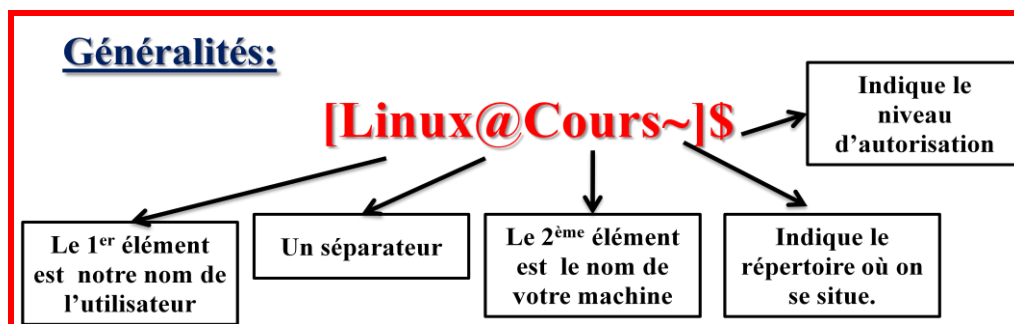


Figure 0.10 – Notation de la ligne Linux

## 0.11 - Commandes sous Linux

### 0.11.1 - La commande "cd" (change directory)

- cd . : Permet de revenir au répertoire personnel.
- cd .. : Permet d'aller au répertoire parent
- cd /user/lib : On se place dans le répertoire lib

- `cd image` : on se place dans le répertoire image.

### 0.11.2 - La commande "`pwd`"

Elle permet d'afficher le chemin absolu d'un répertoire.

### 0.11.3 - La commande "`mkdir`" (make directory)

Elle permet la création de répertoires ou de dossiers comme sur la figure 0.11.

Ex.

- `mkdir musique`
- `mkdir photo vidéo`
- `mkdir -p` : permet de créer des arborescences.

Ex. `mkdir -p cours/stage/docs cours/linux/server cours/maths`

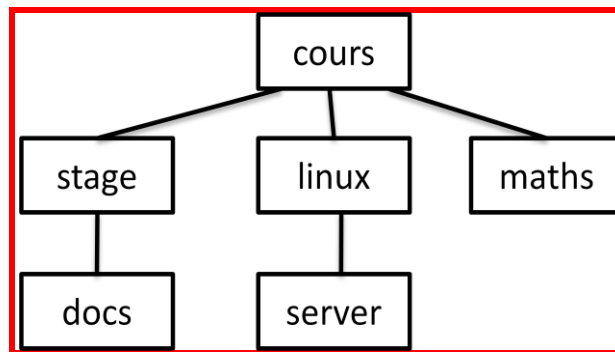


Figure 0.11 – Structure d'arborescence des répertoires de linux

### 0.11.4 - La commande "`ls`" de listage de fichiers

La commande "`ls`" permet d'afficher le contenu du répertoire.

Ex.

`ls nom du répertoire`

- La commande "`ls -l`" permet d'afficher les détails du contenu.
- La commande "`ls -a`" permet d'afficher tous les fichiers même les fichiers cachés.

### 0.11.5 - La commande "`rm`" remove pour supprimer des fichiers

- La commande "`rm`" permet de supprimer des fichiers.
- La commande "`rm -r`" permet de supprimer des répertoires.
- La commande "`rm -rf`" permet de supprimer des répertoires et des sous répertoires.

### 0.11.6 - La commande "mv" (move des fichiers ou des répertoires)

La commande "mv" permet de déplacer des fichiers ou des répertoires et il permet aussi de renommer un fichier.

Ex.

`mv linux cours` ⇒ Déplacer linux dans cours

### 0.11.7 - La commande "cp" (copier des fichiers)

La commande "cp" permet de copier des fichiers.

Ex.

- `cp` ⇒ Permet de copier les fichiers.
- `cp -r` ⇒ Permet de copier des répertoires et des sous répertoires.
- `cp -a` ⇒ Permet de copier en conservant la date, les droits et le propriétaire.

### 0.11.8 - La commande "wc" (calculer)

La commande "wc" permet de calculer le nombre de lignes, de mots et la taille du fichier.

Ex.

- `wc -l nom` ⇒ Permet de calculer le nombre de lignes du fichier nom.
- `wc -w` ⇒ Permet de calculer le nombre de mots.
- `wc -c` ⇒ Permet de calculer la taille du fichier.

### 0.11.9 - Les commandes "head", "tail" et "grep"

- `head -n` ⇒ Permet d'afficher les n premières lignes d'un fichier.
- `tail -n` ⇒ Permet d'afficher les n dernières lignes d'un fichier.
- `grep` ⇒ Permet de chercher dans un fichier.
- `grep -c` ⇒ Permet de compter le nombre de lignes contenant la chaîne.
- `grep -i` ⇒ Permet de ne pas faire de différence entre les majuscules et les minuscules.
- `grep -n` ⇒ Permet de numéroter les lignes.
- `grep -v` ⇒ Permet d'afficher les lignes ne contenant pas la chaîne.

## 0.12 - Les caractères génériques (spéciaux) sous Linux

- (`*` = étoile) permet d'afficher sauf les mots, les répertoires et les fichiers n'importe quelle suite de caractères spécifiée.

Ex. `ls f*` ⇒ Permet de lister sauf que les répertoires et les fichiers commençant avec le caractère "f".

- (? = point d'interrogation) représente n'importe quel caractère après le nom qui existe.

`ls fich?` ⇒ Permet de lister les répertoires et les fichiers commençant par fich.

- [...] ⇒ il représente un intervalle de caractères indiqué entre les crochets.

`ls [a.e]*` ⇒ Permet de lister les répertoires et les fichiers commençant par les caractères (a, b, c, d, e).

- [^,] ⇒ Il représente tous les caractères sauf que les caractères indiqués entre les crochets.

Ex. `ls [^b]*` ⇒ Permet de lister tous les répertoires et les fichiers sauf s'ils contiennent "b".

- (^) ⇒ Permet d'afficher les lignes qui débutent au caractère indiqué.

Ex.

`grep ^d fich1` ⇒ Permet d'afficher tous les lignes débutant par "d".

- (\$) ⇒ Permet d'afficher les lignes finissant au caractère indiqué.

Ex. `grep h$ fich1` ⇒ permet d'afficher tous les lignes finissant par "h".

- (>) ⇒ Permet la redirection de sortie standard.

### Exercice n°1 voir la figure 0.12.

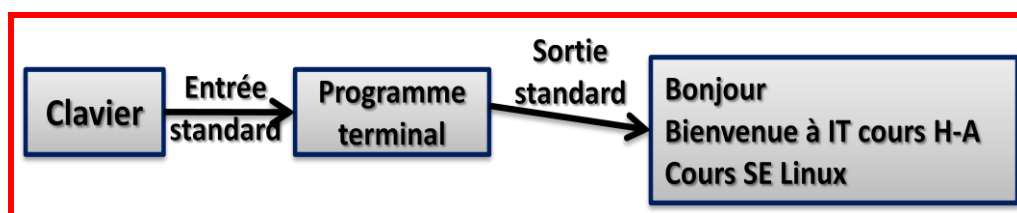
```
cat > fichier 1
```

Bonjour

Bienvenue à IT cours H-A

Cours SE Linux

Ctrl+d.



*Figure 0.12 – Exercice n°1*

- (<) ⇒ Permet la redirection de l'entrée standard.

**Exercice n°2 voir la figure 0.13**

cat < fichier1



*Figure 0.13 – Exercice n°2*

- (>>) ⇒ Permet d'écraser.

Ex. cat < fichier1 >> docs ⇒ Permet d'écraser le fichier1 dans docs.

- (: pipe) ⇒ Permet d'écraser.

Ex. head -4 fich1|wc -w ⇒ Permet de compter le nombre de mots dans les 4 lignes du fichier (fich1).