

# Les systèmes d'exploitation

Filière: Génie Informatique  
S2

## **Chapitre 2**

# **Introduction à Unix/Linux**

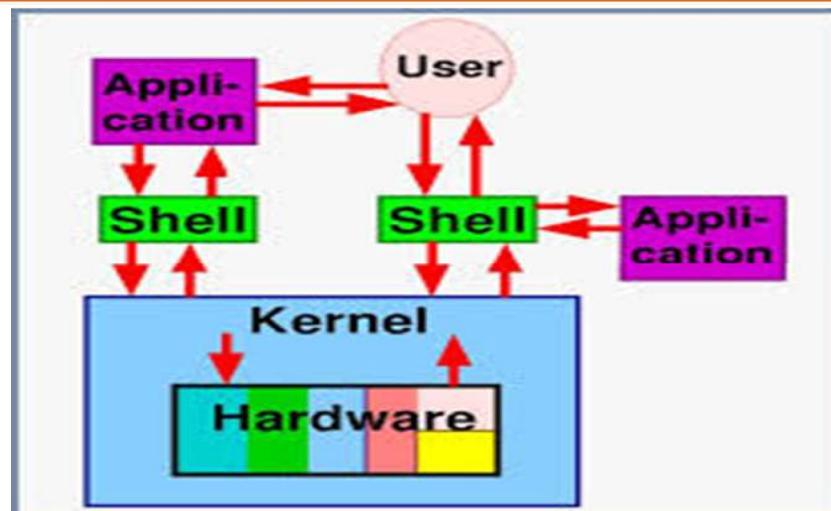
## Qu'est-ce qu'un Unix?

Le système d'exploitation Unix est un ensemble de programmes servant de lien entre l'ordinateur et l'utilisateur.

Les programmes informatiques qui allouent les ressources système et coordonnent tous les détails des composants internes de l'ordinateur sont appelés le système d'exploitation ou le noyau .

## Le Shell

Les utilisateurs communiquent avec le noyau via un programme appelé **shell**. Le shell est un interpréteur de ligne de commande. Il traduit les commandes entrées par l'utilisateur et les convertit dans un langage compris par le noyau.



## Points à retenir

- Unix a été développé en 1969 par un groupe d'employés d'AT&T Ken Thompson, Dennis Ritchie, Douglas McIlroy, et Joe Ossanna chez Bell Labs.
- Il existe différentes variantes Unix disponibles sur le marché. Solaris Unix, AIX, HP Unix et BSD en sont quelques exemples. Linux est également une version d'Unix disponible gratuitement.
- Plusieurs personnes peuvent utiliser un ordinateur Unix en même temps; Unix s'appelle donc un système multi-utilisateur.
- Un utilisateur peut également exécuter plusieurs programmes en même temps. Unix est donc un environnement multitâche.



## Les familles d'UNIX

Du point de vue de l'utilisateur, les divers UNIX se ressemblent beaucoup.

Du point de vue de l'administration, les divers UNIX ont chacun des spécificités (les commandes liées au hardware varient, on trouve des extensions propres à chaque constructeur).

On y voit plusieurs tentatives d'unification :

- ❑ *System V Interface Definition de AT&T (SVID, SVID2, SVID3 en 1989)*
- ❑ *IEEE POSIX (POSIX1003.1 en 1990)*
- ❑ *X/OPEN Portability Guide (XPG4 en 1993) du consortium X/OPEN (créé en 1984)*

**Mais...**

## Les familles d'UNIX

Il existe principalement deux versions de base d'Unix:

- AT&T System V**
- Berkeley Software Distribution (BSD).**

La grande majorité de toutes les variantes d'Unix est construite sur une de ces deux versions. Les principales différences entre les deux sont les utilitaires disponibles et la structure d'implémentation des fichiers..



FreeBSD



at&t

## La naissance de LINUX

En 1991 **Linus Torvalds**, un étudiant finlandais, a commencé à travailler sur un système Unix-like, appelé **Linux**.

Linux n'est qu'un noyau, tandis que les pièces avec laquelle la plupart des gens reconnaissent les outils, **Shell** et système de fichiers, sont les créations du **GNU**.

La force de Linux réside dans son homologation progressive, qui permet aux logiciels d'être librement distribuables sans condition. La seule exigence pour l'utilisateur final est que toutes les modifications apportées au logiciel de base, doivent être rendues disponibles pour d'autres gents de la Communauté, ce qui permet au logiciel d'évoluer à un rythme incroyablement rapide. La licence sous laquelle est distribué Linux s'appelle la **GNU Public License (GPL)**, disponible à <http://gnu.org/licenses/licenses.html>.



## UNIX VS LINUX

- UNIX est protégé par un copyright, seules les grandes entreprises sont autorisées à utiliser le nom UNIX, IBM AIX, Sun Solaris et HP-UX sont tous des systèmes d'exploitation UNIX. La plupart des systèmes UNIX sont de nature commerciale.
- Linux est un clone d'Unix écrite à partir de zéro par Linus Torvalds avec l'aide d'une équipe de hackers à travers le net. Linux est juste un noyau entouré par plusieurs applications et systèmes de gestion, par contre les systèmes d'exploitation UNIX sont considérés comme des systèmes d'exploitation complets.
- Linux est gratuit . Vous pouvez le télécharger à partir d'Internet ou de le redistribuer sous licences GNU. De l'autre côté. La plupart des systèmes UNIX d'exploitation ne sont pas libres .

## UNIX VS LINUX

Des exemples de systèmes UNIX:

- 1.HP-UX
- 2.IBM AIX
- 3.Solaris Sun
- 4.Mac OS X
- 5.IRIX



UNIX

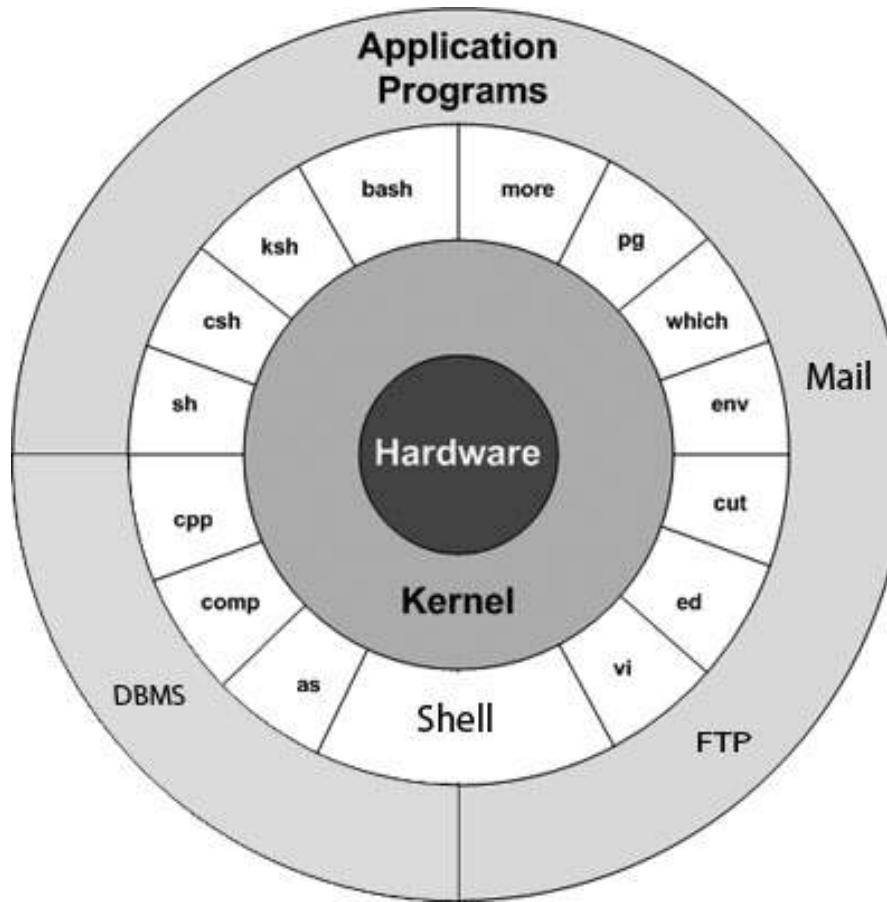


LINUX

Des exemples de distributions Linux:

- 1.Redhat Enterprise Linux
- 2.Fedora Linux
- 3.Debian Linux
- 4.Suse Enterprise Linux
- 5.Ubuntu Linux

## Architecture d'UNIX



## Architecture d'UNIX

Le concept principal qui réunit toutes les versions d'Unix est le suivant:

**Kernel** - Le noyau est le cœur du système d'exploitation. Il interagit avec le matériel et la plupart des tâches telles que la gestion de la mémoire, la planification des tâches et la gestion des fichiers.

**Shell** - Le shell est l'utilitaire qui traite vos requêtes. Lorsque vous tapez une commande sur votre terminal, le shell interprète la commande et appelle le programme souhaité. Le shell utilise une syntaxe standard pour toutes les commandes. C Shell, Bourne Shell et Korn Shell sont les coques les plus célèbres disponibles avec la plupart des variantes Unix.

## Architecture d'UNIX

Le concept principal qui réunit toutes les versions d'Unix est le suivant:

**Commandes et utilitaires** - Il existe diverses commandes et utilitaires que vous pouvez utiliser dans vos activités quotidiennes. cp , mv , cat et grep , etc. sont quelques exemples de commandes et d'utilitaires. Il y a plus de 250 commandes standard , plus nombreux autres fournis par 3 rd logiciels tiers. Toutes les commandes sont accompagnées de différentes options.

**Fichiers et répertoires** - Toutes les données d'Unix sont organisées en fichiers. Tous les fichiers sont ensuite organisés en répertoires. Ces répertoires sont ensuite organisés en une structure arborescente appelée système de fichiers .

## CLI - Interface en Lignes de Commandes -

Le Shell d'un système d'exploitation peut prendre deux formes distinctes :

- **Interface en ligne de commande** (Eng - **CLI : Command Line Interface en anglais**) : cela fonctionne à partir d'instructions en mode texte.
- **Shell** graphique fournissant une interface graphique pour l'utilisateur (Eng - **GUI : Graphical User Interface**).

Ce qui nous intéresse dans ce cours c'est le Shell en mode texte (CLI - Interface en ligne de commande).

Pourquoi !?...

## CLI - Interface en Lignes de Commandes -

- ❑ C'est l'outil natif pour piloter un système Linux
- ❑ C'est un outil puissant : On peut tout faire à partir des ligne de commandes, même avec des manipulations infaisables avec l'interface graphique.
- ❑ C'est peu gourmand en ressources : Cela ne demande pas de cartes graphiques performantes, ni d'extrêmes quantités de mémoire.
- ❑ Les tâches peuvent être automatisées dans des scripts de langage interprété.
- ❑ Communiquer des manipulations systèmes à distance serait plus facile et sera interprété d'une façon unique.
- ❑ ... il y en a plus.

## CLI - Interface en Lignes de Commandes -

Les Shells les plus répandus :

Shell	Program	Description
Bourne shell	sh	disponible sur toute plateforme Unix
C shell	csh	shell développé pour BSD
Korn shell	ksh	Bourne shell amélioré par AT&T
Bourne again shell	bash	Shell distribué avec linux ; version améliorée de sh et csh

Par défaut, le shell associé à un compte d'utilisateur dans Ubuntu est **Bourne-Again Shell (Bash)** ; toutes les commandes saisies par un utilisateur dans un émulateur de terminal sont traitées par **Bash**. Ubuntu inclut aussi l'interpréteur **Bourne Shell (sh)**. D'autres shells, comme **Z Shell (zsh)** ou **C-Shell (csh)**, peuvent être installés facilement à l'aide de son gestionnaire de paquets



## Les modes d'accès au Shell

Deux méthodes d'accès au Shell sont possibles en fonction des ressources matériels disponibles, du système d'exploitation ou du paramétrage utilisé :

- Le **mode console** qui affiche un Shell unique en plein écran, c'est l'interface homme-machine de base du système d'exploitation (accessible au démarrage du système).
- Le **mode terminal** qui émule une console et qui affiche en général le Shell dans une portion de l'écran.



## Le terminale Shell

Tous les Shells se présentent sous la même forme à l'écran lorsqu'ils fonctionnent :

- Une chaîne de caractères affiche que le Shell attend que l'utilisateur tape quelque chose au clavier; c'est le ***prompt***.
- Un curseur qui va se déplacer au fur et à mesure de la saisie des

The screenshot shows a terminal window with a dark grey header bar. In the top-left corner of the header bar are three small icons: a red circle with a white 'x', a grey square with a minus sign, and a grey square with a plus sign. To the right of these icons, the text 'smi@ubuntu: ~' is displayed in a light grey font. The main body of the terminal window has a yellow background. At the top of this area, the text 'smi@ubuntu:~\$' is shown in a black font, indicating the user's name, the machine name, and the current directory, followed by the prompt symbol '\$'. A thin orange border surrounds the entire terminal window.

**NB:** Raccourcis clavier pour lancer le terminal : **Alt + Ctrl + t**  
**smi** : c'est le nom de l'utilisateur actuel du Shell  
**ubuntu** : c'est le nom de la machine  
le symbole **~** indique que nous sommes dans le répertoire personnel  
Le symbole **\$** indique qu'on exécute des commandes en mode utilisateur et non en mode administrateur (le symbole serait alors **#**)



## Les commandes : Syntaxe

Une commande Unix = Un ensemble de mots séparé par des blancs, dont la syntaxe est la suivante :

<b>Commande</b>	<b>[Options]</b>	<b>[Paramètres]</b>
-----------------	------------------	---------------------

**Commande** : Le nom de la commande.

**Options** : Les options permettent de changer le comportement de la commande, il peut s'agir d'une seul option ou plusieurs en même temps.

**Paramètres** : Les informations nécessaires au fonction de la commande.

Les crochets [ et ] (qu'il ne faut pas tapés) signifient que les options ou les paramètres sont **optionnelles**. (dans certaines commandes, c'est obligatoire)



## Les commandes : Les options

Une option est introduite par le signe en rouge « - »

Elle est souvent constituée d'une seule lettre comme par exemple :

**\$ commande -a**

Si on veut exécuter une commande avec plusieurs options (a b et c) :

**\$ commande -a -b -c**

On peut réécrire plus simplement (on concaténant les options)

**\$ commande -abc**

Certains options sont verbeuses (sont écrites en entiers)

(exemple : **all**), ils vont être confondus avec les options d'une seul lettre!!!! **-all = -a -l -a**

Pour cela on exécute les options de cette façon **--all**

## Les commandes : Les options

Les lettres en majuscules sont différents des lettres minuscule

-a est une option

-A est une autre option différente de -a

Les options sont commutatif

**-a -b = -b -a = -ab =-ba**

## Les commandes : La documentation

Il existe une documentation électronique accessible pendant le fonctionnement du système : c'est l'aide en ligne.

La commande donnant l'aide est **man**. Elle donne accès aux pages de manuel des commandes Unix.

Syntaxe de la commande man :

**man [options] commande**

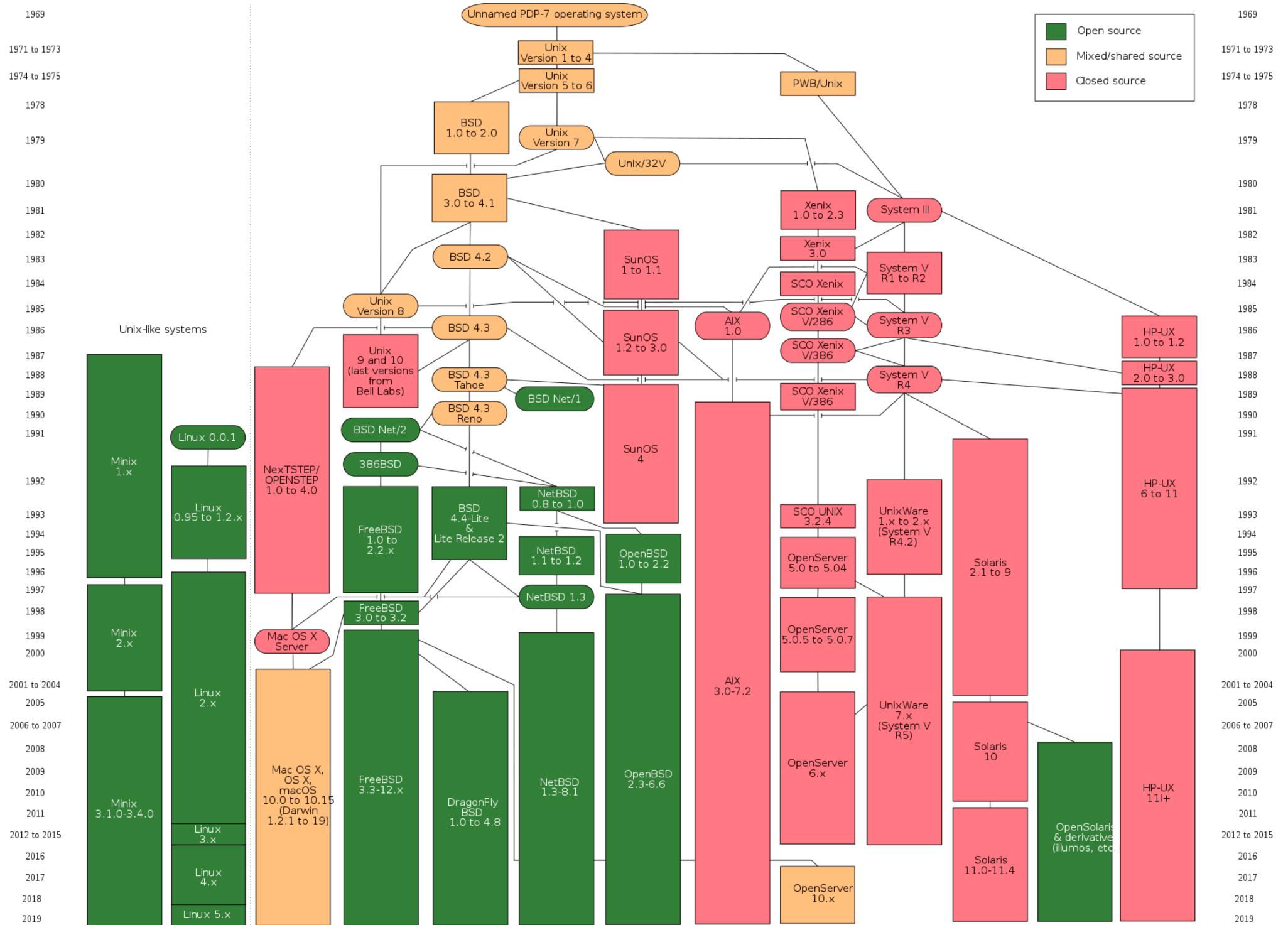
avec en particulier comme option :

**man [numero de section] commande**

**man [-s numero de section] commande**

Le numéro de section représente les rubriques de l'aide en ligne :

- 1: Commandes normales
- 2: Appels systèmes
- 3: fonctions de programmation C
- 4: périphériques et pilotes
- 5: format de fichiers
- 6: jeux
- 7: divers
- 8: Commandes de gestion du système



# Démonstration