

UNIX

Cours 1: Introduction à UNIX

Filipe Vasconcelos¹

¹ESME, Lille, filipe.vasconcelo@esme.fr

- ① Syllabus**
- ② Prérequis – Séance 0**
- ③ Introduction à Unix**



① Syllabus

- 1.1 Objectif principal du cours
- 1.2 Les acquis d'apprentissage visés
- 1.3 Programme du module
- 1.4 Modalités d'évaluation
- 1.5 Ressources pédagogiques

② Prérequis – Séance 0

③ Introduction à Unix



1

Syllabus

1 Syllabus

- 1.1 Objectif principal du cours
- 1.2 Les acquis d'apprentissage visés
- 1.3 Programme du module
- 1.4 Modalités d'évaluation
- 1.5 Ressources pédagogiques

2 Prérequis – Séance 0

3 Introduction à Unix

- 3.1 Système d'exploitation
- 3.2 Le système d'exploitation UNIX
- 3.3 Système de fichiers
- 3.4 Commandes de base



Objectif principal du cours

À l'issue de ce cours, les étudiants pourront intéragir avec l'interpréteur de commandes **bash** largement associé aux systèmes d'exploitation du type **UNIX**.

1 Syllabus

- 1.1 Objectif principal du cours
- 1.2 Les acquis d'apprentissage visés
- 1.3 Programme du module
- 1.4 Modalités d'évaluation
- 1.5 Ressources pédagogiques

2 Prérequis – Séance 0

3 Introduction à Unix

- 3.1 Système d'exploitation
- 3.2 Le système d'exploitation UNIX
- 3.3 Système de fichiers
- 3.4 Commandes de base



Les acquis d'apprentissage visés

AAV1 : Identifier ou reconnaître un système d'exploitation de la famille Unix (Linux, macOS, BSD) ainsi que donner quelques exemples de ces systèmes d'exploitation. Installer et utiliser un interpréteur de commandes bash que ce soit par l'intermédiaire d'un émulateur, d'une machine virtuelle ou un système d'exploitation complet de la famille Unix

AAV2 : Reconnaître, décrire et utiliser les commandes de bases de l'interpréteur de commandes bash ou trouver efficacement et rapidement de la documentation sur ces commandes.

AAV3 : Utiliser et enchaîner, au moyen de tubes, les principales commandes de filtre pour manipuler des flux de données qu'ils proviennent ou non de fichiers

AAV4 : Produire, tester et vérifier le résultat des scripts SHELL (en bash) pour réaliser des tâches algorithmiques simples à complexes

AAV5 : Analyser et explorer l'arborescence d'un système de fichiers d'un OS du type Unix à l'aide des commandes de base, administrer les droits d'accès des répertoires et fichiers de l'arborescence et gérer les processus.

AAV6 : Concevoir, commenter et justifier l'interface en ligne de commande d'un script développé.

1 Syllabus

- 1.1 Objectif principal du cours
- 1.2 Les acquis d'apprentissage visés
- 1.3 Programme du module
- 1.4 Modalités d'évaluation
- 1.5 Ressources pédagogiques

2 Prérequis – Séance 0

3 Introduction à Unix

- 3.1 Système d'exploitation
- 3.2 Le système d'exploitation UNIX
- 3.3 Système de fichiers
- 3.4 Commandes de base



11 séances de 2h :

1. Introduction à UNIX (Chemin absolu/relatif)
2. Commandes de base, metacaractères du shell
3. Commandes grep et expressions régulières
4. Redirection d'entrées/sorties et les tubes
5. Gestion des utilisateurs et droits d'accès
6. Gestion des processus
7. Programmation SHELL (bash) (Partie 1)
8. Programmation SHELL (bash) (Partie 2)
9. Programmation SHELL (bash) (Partie 3)
10. Programmation SHELL (bash) (Partie 4)
11. Commandes de filtre
12. Revision – Préparation à l'examen



1 Syllabus

- 1.1 Objectif principal du cours
- 1.2 Les acquis d'apprentissage visés
- 1.3 Programme du module
- 1.4 Modalités d'évaluation**
- 1.5 Ressources pédagogiques

2 Prérequis – Séance 0

3 Introduction à Unix

- 3.1 Système d'exploitation
- 3.2 Le système d'exploitation UNIX
- 3.3 Système de fichiers
- 3.4 Commandes de base



$$\text{TOTAL} = 0.5\text{CC} + 0.5\text{EX}$$

CC : Contrôle continu

- Quiz Moodle hebdomadaires en début de séance (10 questions QCM).
- Les questions portent sur les deux séances précédentes.
- Le premier quiz (séance 2) est formatif.
- Deux jokers, moyenne sur les 8 meilleures notes. (Une absence = 0).

EX : Examen final

- Examen Moodle+SEB.
- Programmation de scripts bash.
- La dernière séance de cours est une séance de révision sur les annales.



1 Syllabus

- 1.1 Objectif principal du cours
- 1.2 Les acquis d'apprentissage visés
- 1.3 Programme du module
- 1.4 Modalités d'évaluation
- 1.5 Ressources pédagogiques

2 Prérequis – Séance 0

3 Introduction à Unix

- 3.1 Système d'exploitation
- 3.2 Le système d'exploitation UNIX
- 3.3 Système de fichiers
- 3.4 Commandes de base



❑ Moodle :

- ❑ Les pdf de cours, énoncés et corrigés des TD (peuvent être mis à jour au cours de l'année).
- ❑ Les quiz hebdomadaires
- ❑ Des ressources vidéos pour compléter les cours
- ❑ Des annales d'examen

❑ Github étudiant :

- ❑ Les pdf (mis à jour en début d'année)
- ❑ Les fichiers partagés pour les activités en séances.



- 1 Syllabus**
- 2 Prérequis – Séance 0**
- 3 Introduction à Unix**





2

Prérequis – Séance 0



Avez-vous installé un interpréteur de commandes bash ?
(WSL, VM ou un Linux ...?)



- ① Syllabus
- ② Prérequis – Séance 0
- ③ **Introduction à Unix**
 - 3.1 Système d'exploitation
 - 3.2 Le système d'exploitation UNIX
 - 3.3 Système de fichiers
 - 3.4 Commandes de base





3

Introduction à Unix



AAV1

Identifier ou reconnaître un système d'exploitation de la famille Unix (Linux, macOS, BSD) ainsi que donner quelques exemples de ces systèmes d'exploitation. Installer et utiliser un interpréteur de commandes bash que ce soit par l'intermédiaire d'un émulateur, d'une machine virtuelle ou un système d'exploitation complet de la famille Unix



1 Syllabus

- 1.1 Objectif principal du cours
- 1.2 Les acquis d'apprentissage visés
- 1.3 Programme du module
- 1.4 Modalités d'évaluation
- 1.5 Ressources pédagogiques

2 Prérequis – Séance 0

3 Introduction à Unix

- 3.1 Système d'exploitation
- 3.2 Le système d'exploitation UNIX
- 3.3 Système de fichiers
- 3.4 Commandes de base





Lancez l'activité wooclap de la séance: Cours01

Quels mots vous évoque le terme « système d'exploitation » ?



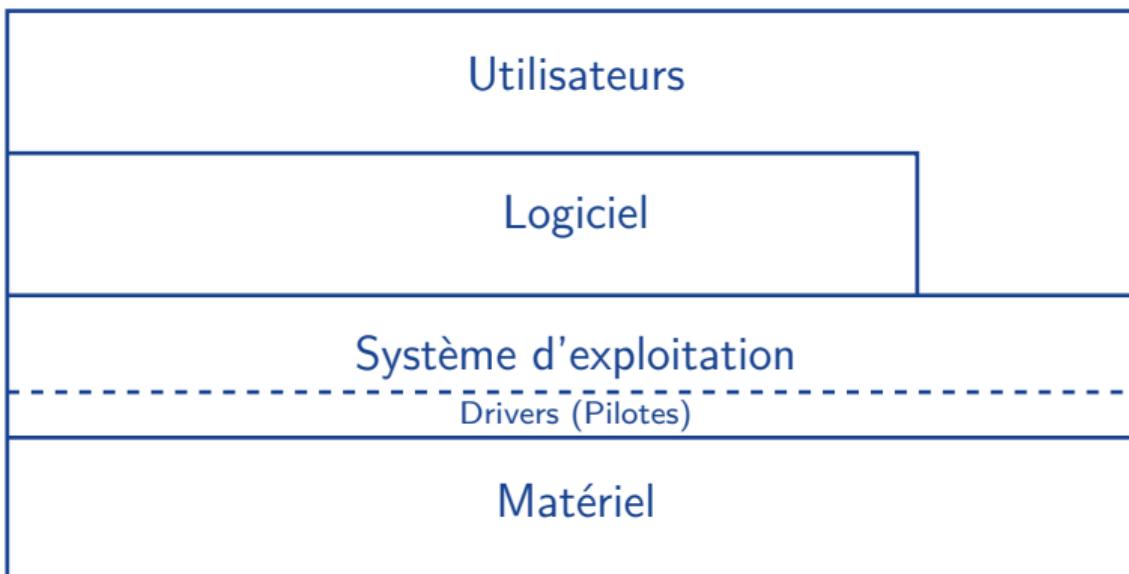
Operating System (OS) en Anglais

Un OS est un ensemble de programmes qui gère l'utilisation des ressources d'un ordinateur pour les logiciels applicatifs.

- ❑ Exemples de systèmes d'exploitation :
 - ❑ Desktop : Windows 10/8/7, MacOS, UNIX, Linux
 - ❑ Smartphone : ios, android
- ❑ Exemples d'utilisation quotidienne :
 - ❑ Naviguation Web : (Il pilote la carte réseau, active la transmission de donnée via internet, etc.)
 - ❑ Editer un fichier : (Il gère le système de fichier de l'ordinateur)



- Un système d'exploitation est une interface entre les couches utilisateur et matérielle d'un ordinateur. C'est un programme informatique lancé au démarrage permettant une gestion optimale des ressources physiques d'un ordinateur.



1 Syllabus

- 1.1 Objectif principal du cours
- 1.2 Les acquis d'apprentissage visés
- 1.3 Programme du module
- 1.4 Modalités d'évaluation
- 1.5 Ressources pédagogiques

2 Prérequis – Séance 0

3 Introduction à Unix

- 3.1 Système d'exploitation
- 3.2 Le système d'exploitation UNIX
- 3.3 Système de fichiers
- 3.4 Commandes de base





Lancez l'activité wooclap de la séance: Cours01

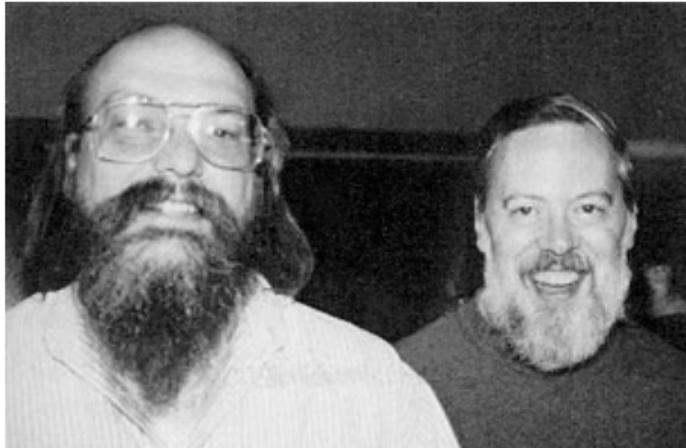
Quels mots vous évoque « Unix » ?



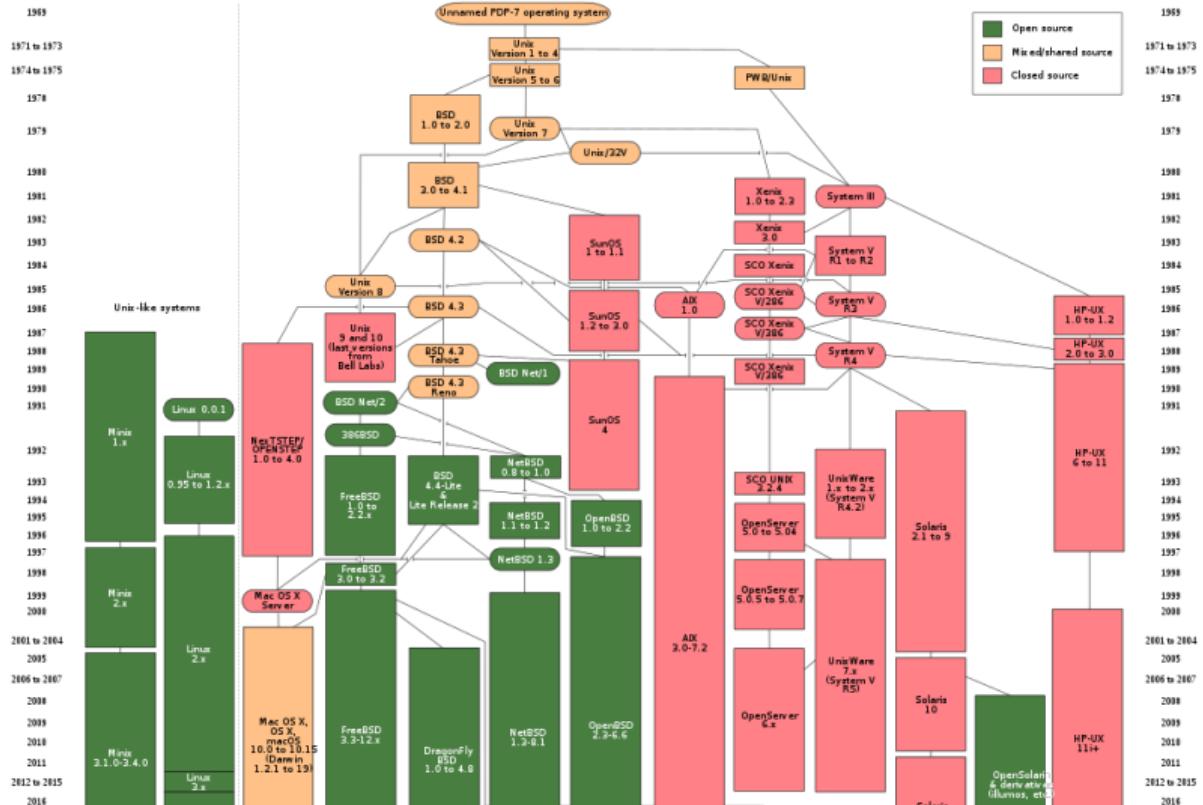
Histoire de UNIX

La première version de UNIX fut développé en 1969 par Ken Thompson, Dennis Ritchie au Bell Labs de l'entreprise AT&T.

- ❑ La société a ensuite distribué UNIX aux étudiants et professeurs des universités.
- ❑ UNIX était à l'époque distribué avec son code source ce qui a permis de nombreux développements et contributions indépendants.



La grande famille UNIX

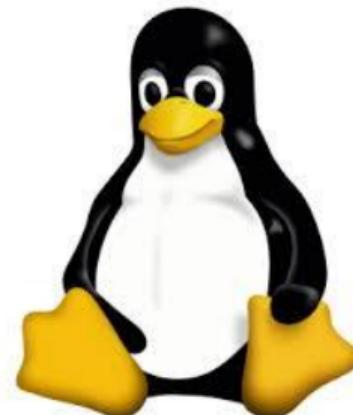


Le projet GNU

- ❑ Richard Stallman crée le projet GNU en 1984.
- ❑ GNU est un système d'exploitation de type UNIX. Cela signifie qu'il s'agit d'un ensemble de nombreux programmes : applications, bibliothèques, outils de développement.
- ❑ Le nom « GNU » est un acronyme récursif pour « GNU's Not UNIX ».
- ❑ Les programmes GNU sont distribués sous licence libre GPL.



- ❑ Développé à l'université de Helsinki en 1991 par Linus Torvalds.
- ❑ Linux est un noyau du système d'exploitation inspiré d'UNIX.
- ❑ Linux est un logiciel libre.



- ❑ Ces deux projets sont complémentaires : tandis que Richard Stallman créait les programmes de base (programme de copie de fichier, de suppression de fichier, éditeur de texte), Linus s'était lancé dans la création du « cœur » d'un système d'exploitation : le noyau.
- ❑ Le projet GNU (programmes libres) et Linux (noyau d'OS) ont fusionné pour créer GNU/Linux.
- ❑ C'est GNU/Linux qui est généralement simplement appelé Linux.



Utilisation de Linux

- Serveur (serveur Web Apache, serveur de base de donnée SQL, serveur PHP, etc.)
- Sécurité réseau (passerelle, routeur, proxy, pare-feu, etc.)
- Ordinateur central (centre de calculs dans les banques, sociétés de l'assurance, etc.)
- Grappe de serveur (cluster de Google, Amazon, Microsoft, etc.)
- Supercalculateurs (calcul intensif)
- Système embarqué (box d'internet, smartphone, décodeur TNT, PlayStation, Raspberry, etc.)



Distribution GNU/Linux

- ❑ Linux lui-même n'étant qu'un noyau, il nécessite d'être accompagné d'autres logiciels pour former un système d'exploitation complet.
- ❑ Plusieurs entreprises ou associations distribuent GNU/Linux accompagné d'un ensemble cohérent de logiciels : ce que l'on nomme une distribution Linux.



Distribution Linux

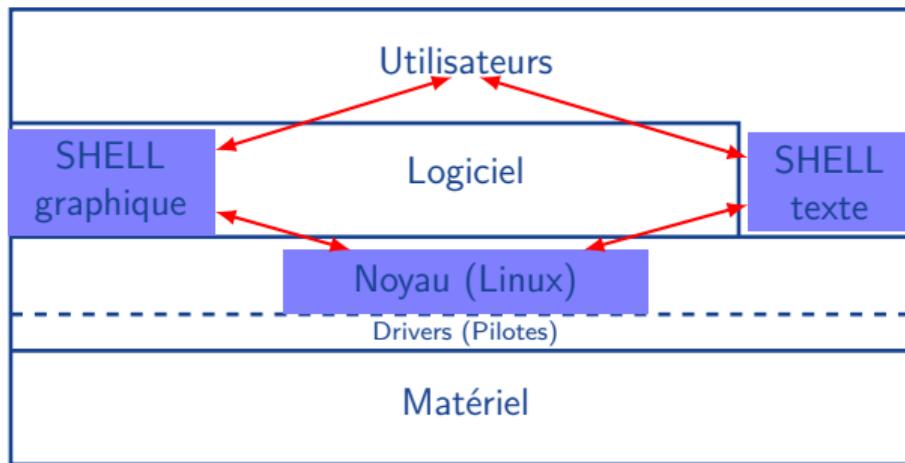


debian



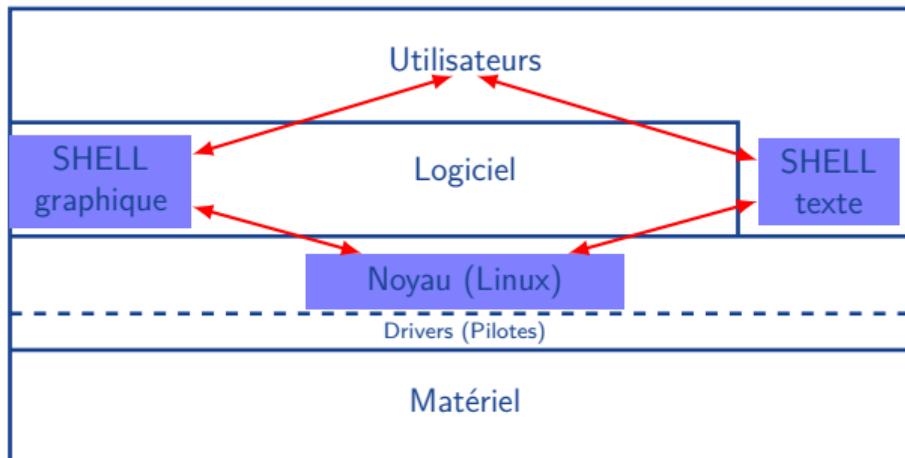
GUI : *Graphical User Interface*

- ❑ Il existe plusieurs environnements graphiques sous GNU/Linux :
 - ❑ GTK : Gnome, Unity, MATE, Xfce...
 - ❑ Qt : KDE, LXQt, ...
 - ❑ WSL (Windows) : Xming

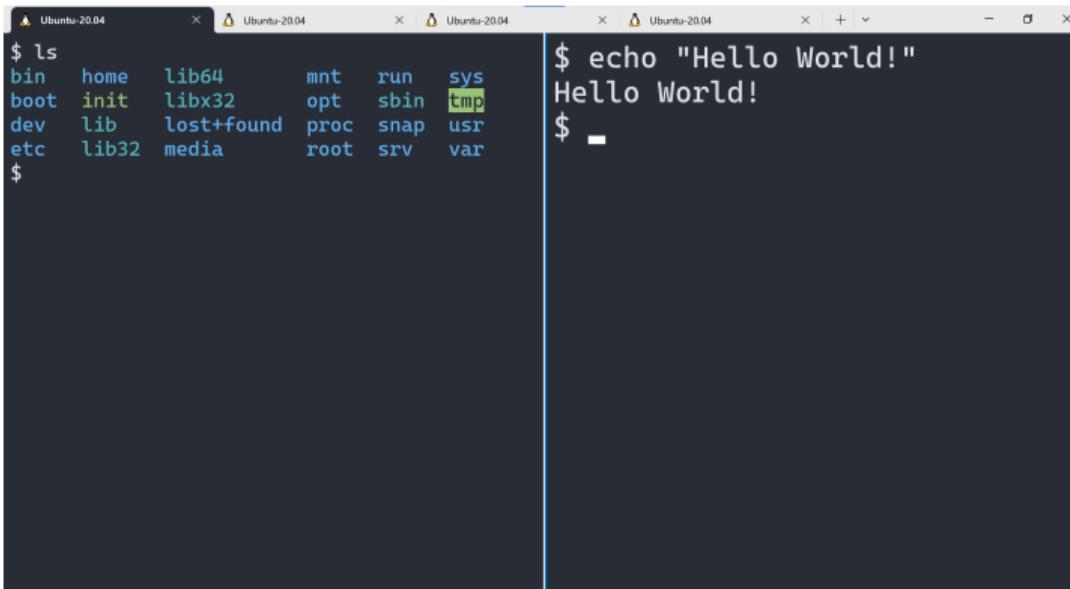


CLI : *Command line interface*

- ❑ Il existe différentes versions d'interpréteur de commande :
 - ❑ **bash** : Bourne Again Shell
 - ❑ sh : Bourne Shell
 - ❑ zsh : Z Shell



- ❑ Plus efficace, plus rapide pour donner les ordres au ordinateur.
- ❑ Elle consomme moins de ressource matérielle pour fonctionner.
- ❑ Elle permet d'automatiser une suite d'instructions à exécuter via le script (prochain cours).



The image shows three separate terminal windows side-by-side, all running the Ubuntu 20.04 LTS distribution. The first window on the left displays the output of the `ls` command, listing standard Linux system directories like `bin`, `home`, `lib64`, etc. The middle window shows the output of the `echo "Hello World!"` command, which prints `Hello World!` to the screen. The third window on the right is mostly blank, with only a single dollar sign prompt visible.

```
$ ls
bin  home  lib64      mnt  run  sys
boot init  libx32     opt  sbin  tmp
dev   lib  lost+found  proc  snap  usr
etc  lib32 media      root  srv  var
$
```

```
$ echo "Hello World!"
Hello World!
$ -
```

1 Syllabus

- 1.1 Objectif principal du cours
- 1.2 Les acquis d'apprentissage visés
- 1.3 Programme du module
- 1.4 Modalités d'évaluation
- 1.5 Ressources pédagogiques

2 Prérequis – Séance 0

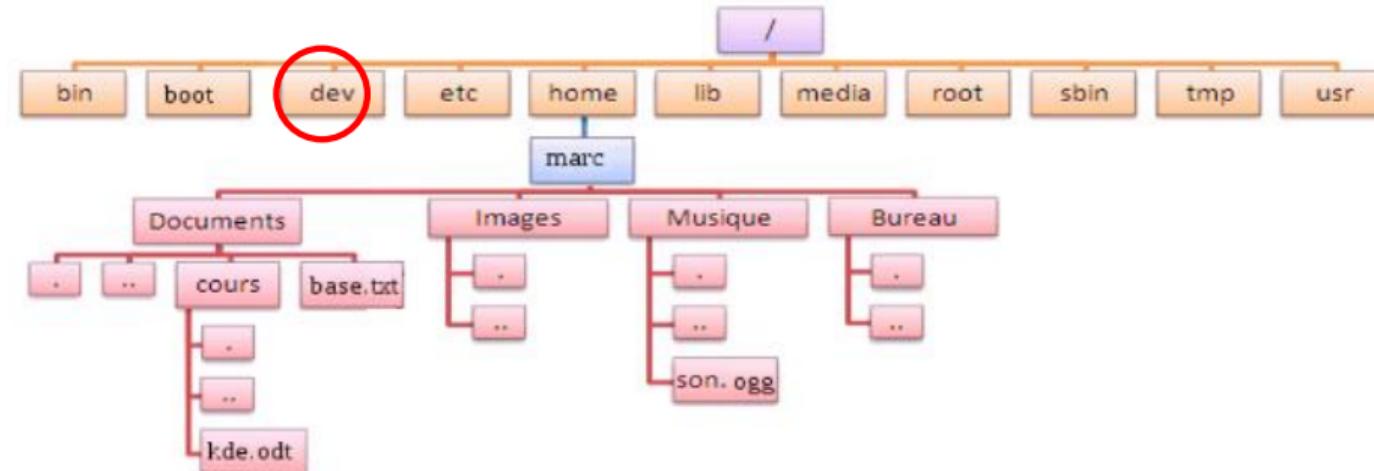
3 Introduction à Unix

- 3.1 Système d'exploitation
- 3.2 Le système d'exploitation UNIX
- 3.3 Système de fichiers
- 3.4 Commandes de base



Tout est fichier sous Linux

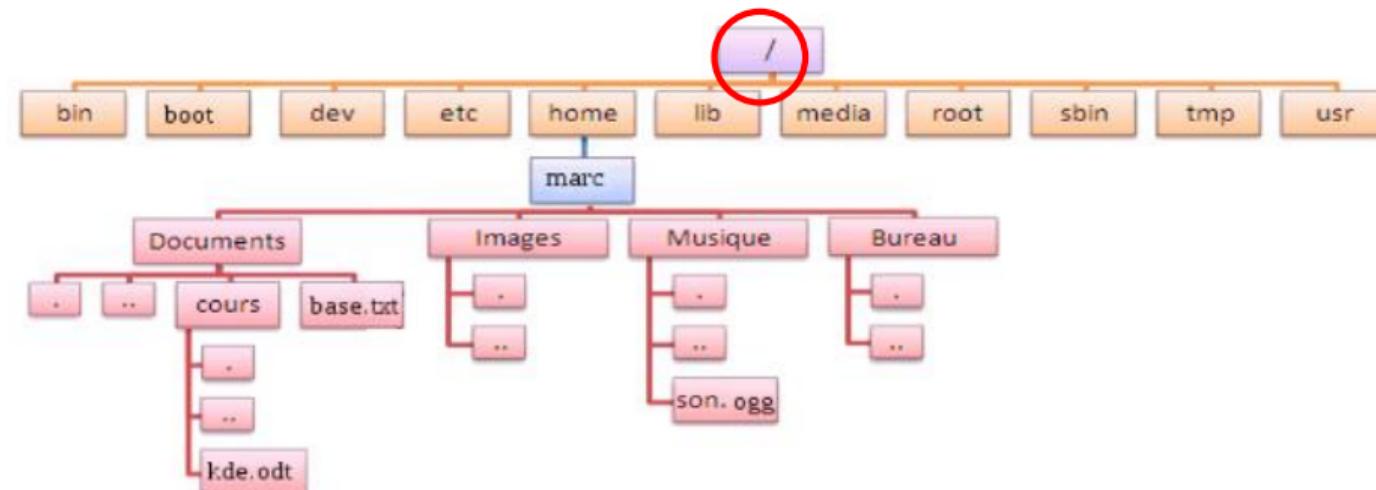
- ❑ Disque, clé USB, processeur, réseaux, carte graphique, carte de son, etc.
- ❑ L'organisation du matériel sous Linux est très différente de celle sous Windows
- ❑ Exemple : Sous Linux, brancher un périphérique s'apparente à créer un nouveau fichier dans /dev



Arborescence

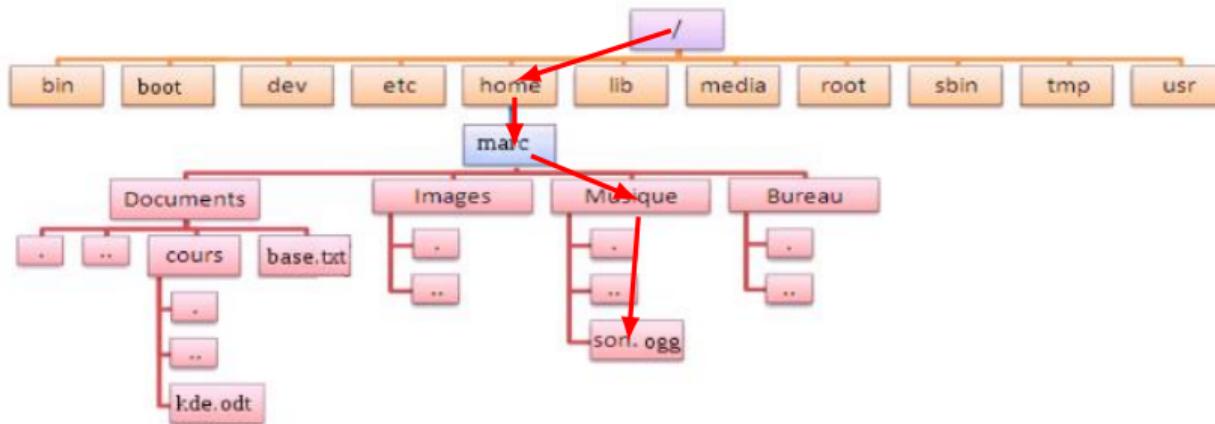
Les répertoires (« dossiers ») sont les noeuds d'un arbre

- ❑ La racine (« root ») est le répertoire en « haut » de la hiérarchie.
- ❑ Le système connaît la racine.
- ❑ Sous Linux, la racine est désignée par / : point de départ de l'arbre.



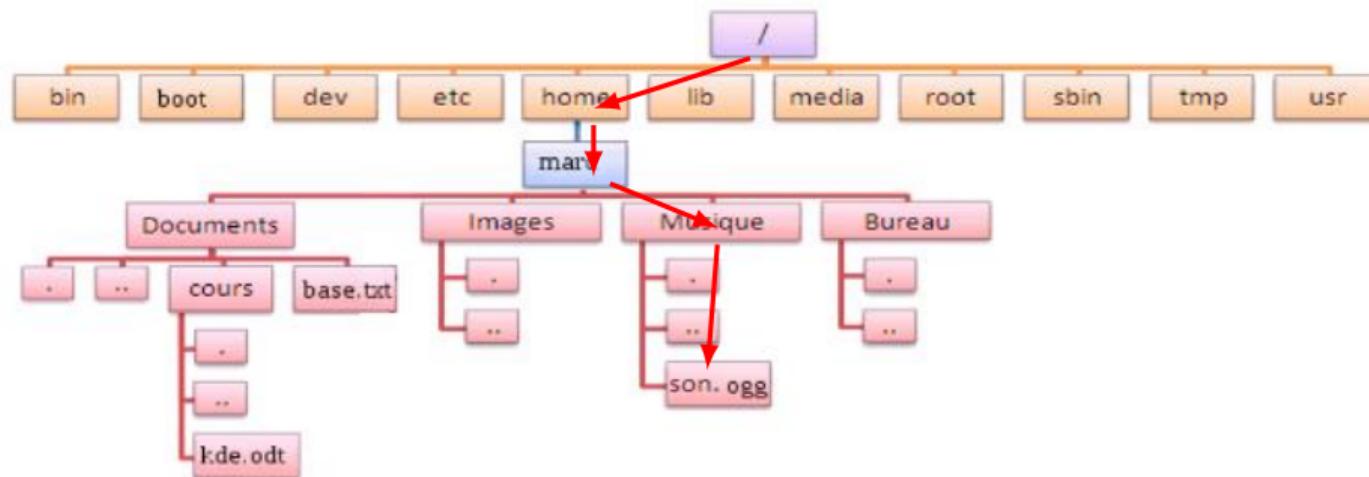
Comment repérer un fichier dans cette arborescence ?

- ❑ Un nom de fichier est composé : du chemin (e.g. noms des répertoires à traverser séparé par /) et du nom du fichier
- ❑ Exemple : /home/marc/Musique/son.ogg



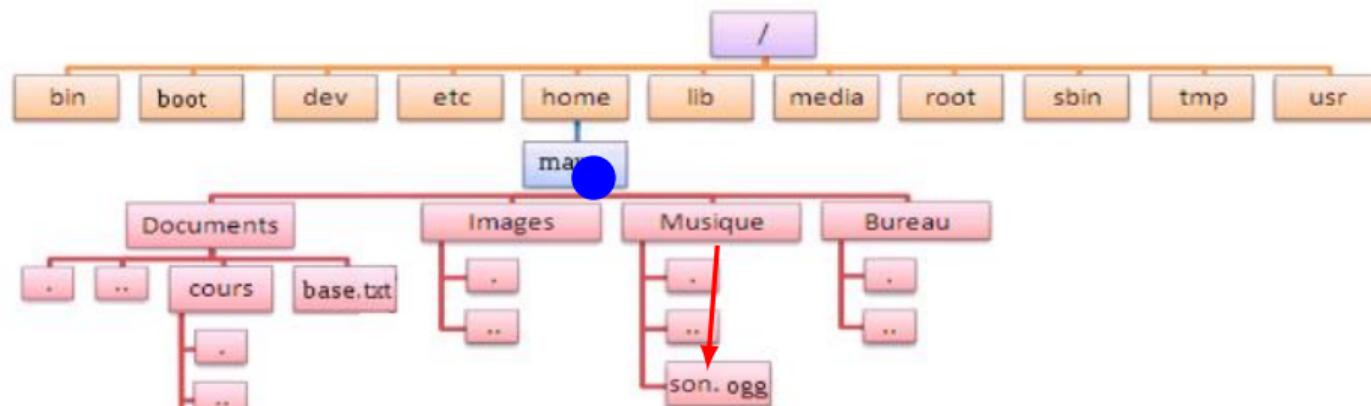
Chemin absolu

- ❑ Répertoire de départ : Répertoire racine
- ❑ Un nom absolu commence par /
- ❑ Exemple : /home/marc/Musique/son.ogg



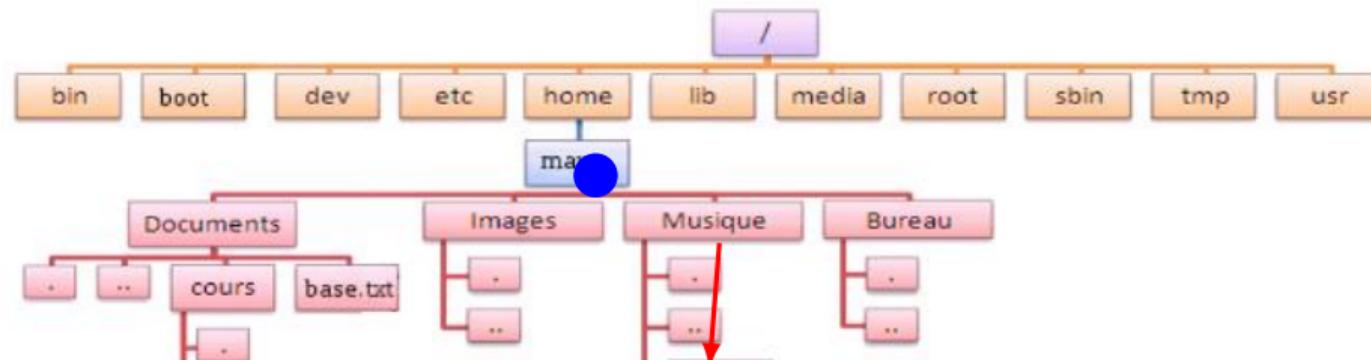
Chemin relatif

- ❑ Répertoire de départ : Répertoire courant
- ❑ Un chemin relatif commence par un nom de fichier ou de répertoire
- ❑ Exemple : si /home/marc est le répertoire courant, Musique/son.ogg est le chemin relatif vers son.ogg.



Chemin relatif (suite)

- ❑ Nom relatif du répertoire courant : ..
- ❑ Exemple : ./Musique/son.ogg est équivalent à Musique/son.ogg
- ❑ Nom relatif du père du répertoire courant : ...
- ❑ Exemple : ../marc/Musique/song.ogg est équivalent à Musique/son.ogg



Qu'est ce qu'un répertoire ?

- Un répertoire est la liste des fichiers qu'il contient.
- Un répertoire n'est jamais vide :
- Il contient au moins les informations sur lui-même : . et son père : ..
- Exemple : Si le répertoire courant est /home/marc/Musique on accède au fichier du répertoire Documents par le chemin relatif ../Documents.



Noms absous

- Utilisable quelque soit le contexte
- Pratiques dans les scripts : Indépendance au répertoire courant
- Pratiques pour désigner un fichier ou répertoire qui n'est dans le voisinage direct du répertoire courant. On n'a pas à compter les ../../../../

Noms relatifs

- Noms courts pour trouver le fichier ou répertoire de voisinage direct



Répertoires d'un UNIX-like

- ❑ / : Répertoire racine
- ❑ /root : Répertoire personnel de l'utilisateur root (administrateur)
- ❑ /home : Répertoire des utilisateurs
- ❑ /Users : Répertoire des utilisateurs (MacOS)
- ❑ /usr : Répertoire important du système Linux. Il contient des données et des programmes communs aux utilisateurs.



1 Syllabus

- 1.1 Objectif principal du cours
- 1.2 Les acquis d'apprentissage visés
- 1.3 Programme du module
- 1.4 Modalités d'évaluation
- 1.5 Ressources pédagogiques

2 Prérequis – Séance 0

3 Introduction à Unix

- 3.1 Système d'exploitation
- 3.2 Le système d'exploitation UNIX
- 3.3 Système de fichiers
- 3.4 Commandes de base





| File Commands | System Info |
|---|---|
| <code>ls</code> - directory listing | |
| <code>ls -al</code> - formatted listing with hidden files | |
| <code>cd dir</code> - change directory to <code>dir</code> | |
| <code>cd ..</code> - change to home | |
| <code>pwd</code> - show current directory | |
| <code>mkdir dir</code> - create a directory <code>dir</code> | |
| <code>rm file</code> - delete <code>file</code> | |
| <code>rm -r dir</code> - delete directory <code>dir</code> | |
| <code>rm -f file</code> - force remove <code>file</code> | |
| <code>rm -rf dir</code> - force remove directory <code>dir</code> * | |
| <code>cp file1 file2</code> - copy <code>file1</code> to <code>file2</code> | |
| <code>cp -r dir1 dir2</code> - copy <code>dir1</code> to <code>dir2</code> ; create <code>dir2</code> if it doesn't exist | |
| <code>mv file1 file2</code> - rename or move <code>file1</code> to <code>file2</code> if <code>file2</code> is an existing directory, moves <code>file1</code> into directory <code>file2</code> | |
| <code>ln -s file link</code> - create symbolic link <code>link</code> to <code>file</code> | |
| <code>touch file</code> - create or update <code>file</code> | |
| <code>cat > file</code> - places standard input into <code>file</code> | |
| <code>more file</code> - output the contents of <code>file</code> | |
| <code>head file</code> - output the first 10 lines of <code>file</code> | |
| <code>tail file</code> - output the last 10 lines of <code>file</code> | |
| <code>tail -f file</code> - output the contents of <code>file</code> as it grows, starting with the last 10 lines | |
| Process Management | Compression |
| <code>ps</code> - display your currently active processes | <code>tar cf file.tar files</code> - create a tar named <code>file.tar</code> containing <code>files</code> |
| <code>top</code> - display all running processes | <code>tar xf file.tar</code> - extract the files from <code>file.tar</code> |
| <code>kill pid</code> - kill process id <code>pid</code> | <code>tar czf file.tar.gz files</code> - create a tar with Gzip compression |
| <code>killall proc</code> - kill all processes named <code>proc</code> * | <code>tar xzf file.tar.gz</code> - extract a tar using Gzip |
| <code>bg</code> - lists stopped or background jobs; resume a stopped job in the background | <code>tar cjf file.tar.bz2</code> - create a tar with Bzip2 compression |
| <code>fg</code> - brings the most recent job to foreground | <code>tar xjf file.tar.bz2</code> - extract a tar using Bzip2 |
| <code>fg n</code> - brings job <code>n</code> to the foreground | <code>gzip file</code> - compresses <code>file</code> and renames it to <code>file.gz</code> |
| | <code>gzip -d file.gz</code> - decompresses <code>file.gz</code> back to <code>file</code> |
| File Permissions | Network |
| <code>chmod octal file</code> - change the permissions of <code>file</code> to <code>octal</code> , which can be found separately for user, group, and world by adding: | <code>ping host</code> - ping <code>host</code> and output results |
| <ul style="list-style-type: none">• 4 - read (r)• 2 - write (w)• 1 - execute (x) | <code>whois domain</code> - get whois information for <code>domain</code> |
| | <code>dig domain</code> - get DNS information for <code>domain</code> |
| | <code>dig -x host</code> - reverse lookup <code>host</code> |
| | <code>wget file</code> - download <code>file</code> |
| | <code>wget -c file</code> - continue a stopped download |
| Installation | |
| | Install from source: <code>./configure</code> |

Commande

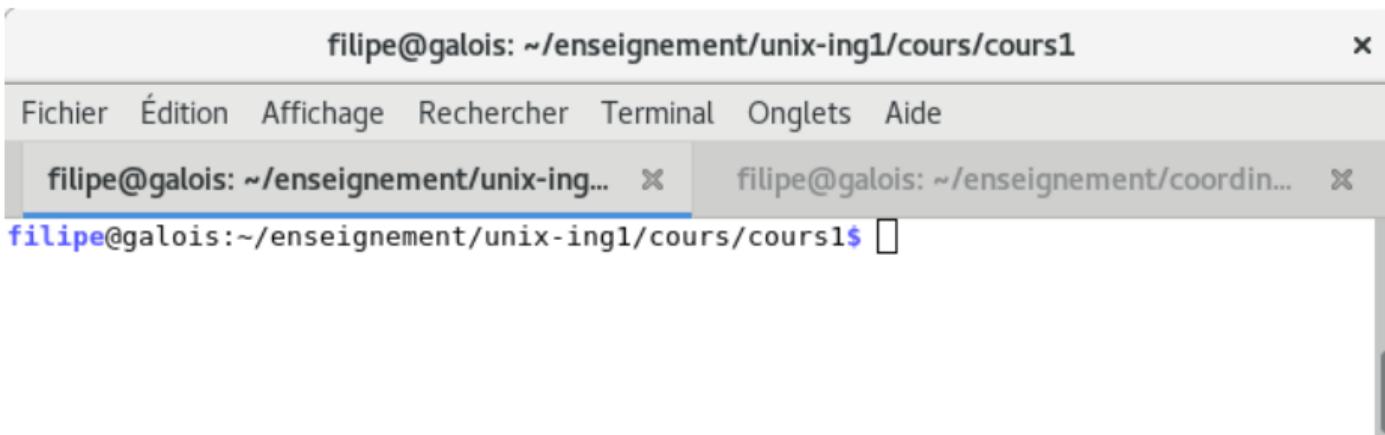
- ❑ Une commande comporte des séparateurs permettant d'isoler les commandes, les options, les arguments (Le caractère espace est l'un de ces séparateurs)
- ❑ Le premier élément de la commande est composé de soit du :
 - ❑ Nom d'une commande interne sous Linux
 - ❑ Nom d'un exécutable (e.g. script)



Premières commandes

Une ligne de commande est écrite devant un prompt

- ❑ prompt : Chaine de caractères indiquant que l'interpréteur attend une nouvelle commande

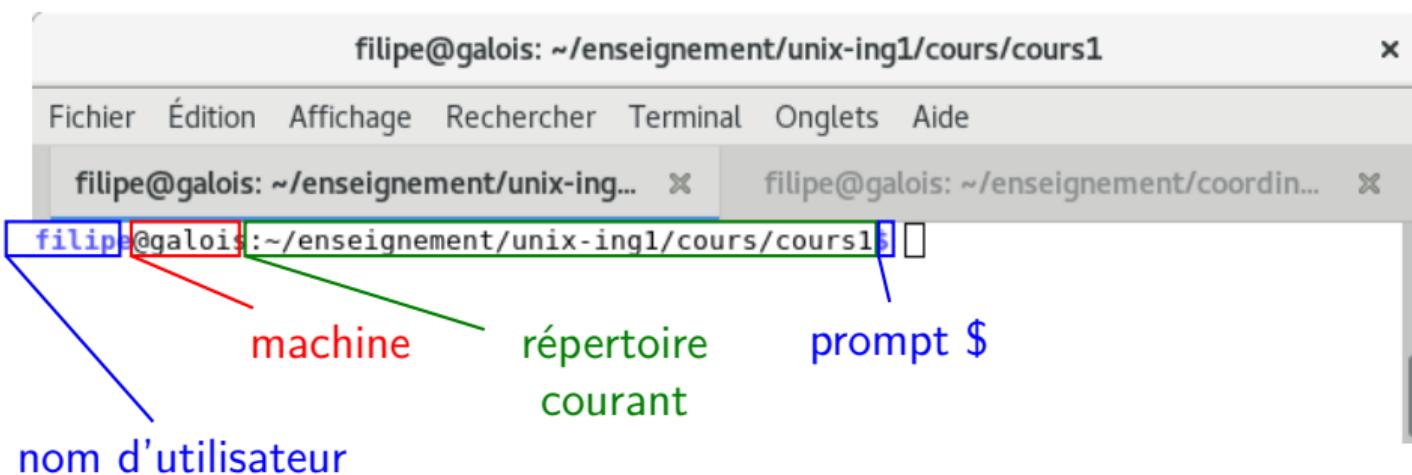


A screenshot of a terminal window with a blue header bar. The header bar contains the text "filipe@galois: ~/enseignement/unix-ing1/cours/cours1" and a close button (x). Below the header is a menu bar with French labels: Fichier, Édition, Affichage, Rechercher, Terminal, Onglets, Aide. The main area shows two tabs at the bottom: "filipe@galois: ~/enseignement/unix-ing..." and "filipe@galois: ~/enseignement/coordin...". The active tab has a grey background. In the center of the window, there is a command line input field with the text "filipe@galois:~/enseignement/unix-ing1/cours/cours1\$". To the right of the input field is a small square icon with a white border.

Premières commandes

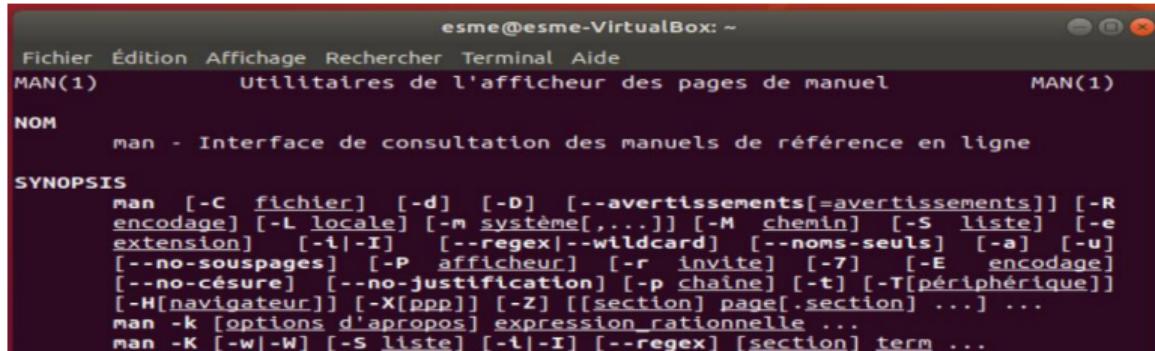
Une ligne de commande est écrite devant un prompt

- ❑ prompt : Chaine de caractères indiquant que l'interpréteur attend une nouvelle commande



man manual

- ❑ Usage : commande permettant d'accéder aux pages de manuel installées sur le système.
- ❑ Syntaxe : `man [-k] [section]` commande `-k` : Cherche la description courte des pages de manuel correspondant au mot-clé et affiche toutes celles qu'il trouve.
- ❑ Exemple : `man man`



```
esme@esme-VirtualBox: ~
Fichier Édition Affichage Rechercher Terminal Aide
MAN(1)          Utilitaires de l'afficheur des pages de manuel           MAN(1)

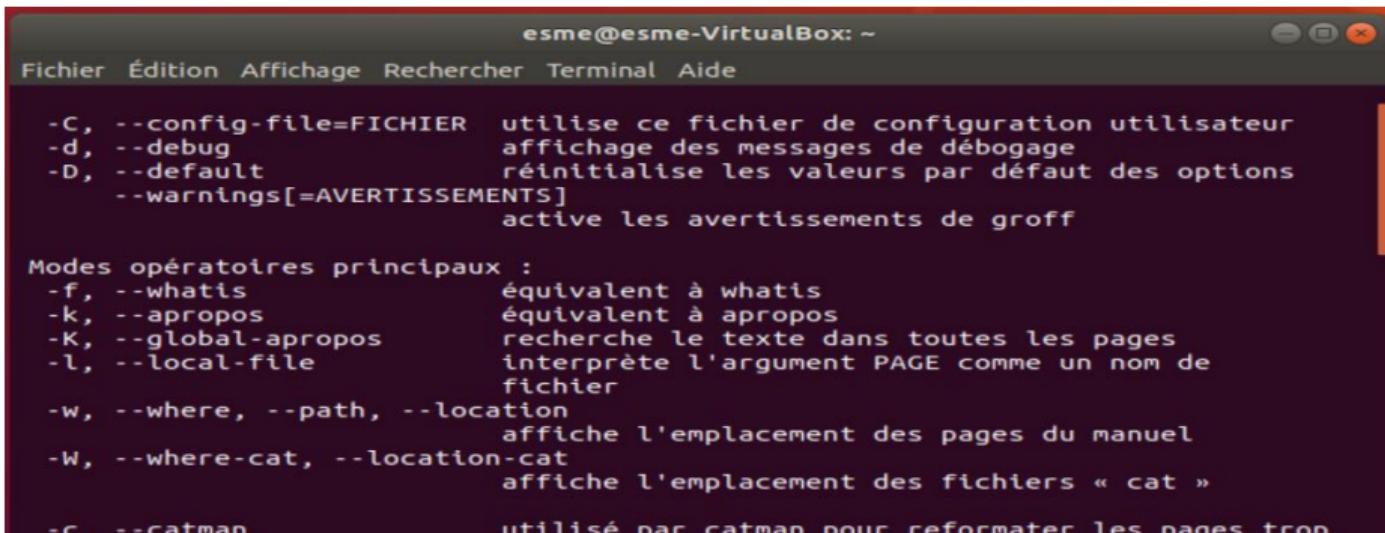
NOM
    man - Interface de consultation des manuels de référence en ligne

SYNOPSIS
    man [-C fichier] [-d] [-D] [--avertissements[=avertissements]] [-R
    encodage] [-L locale] [-m système[,...]] [-M chemin] [-S liste] [-e
    extension] [-t|-I] [-regex|-wildcard] [--noms-seuls] [-a] [-u]
    [--no-souspages] [-P afficheur] [-r invite] [-7] [-E encodage]
    [--no-césure] [--no-justification] [-p chaîne] [-t] [-T[périphérique]]
    [-H[navigateur]] [-X[ppp]] [-Z] [[section] page[...]] ...
    man -k [options d'apropos] expression rationnelle ...
    man -K [-w|-W] [-S liste] [-t|-I] [-regex] [section] term ...
```

Option -help

-help

- Usage : On peut aussi passer l'option -help ou -h à une commande pour avoir de l'aide
- Exemple : man -help



The screenshot shows a terminal window titled "esme@esme-VirtualBox: ~". The menu bar includes "Fichier", "Édition", "Affichage", "Rechercher", "Terminal", and "Aide". The main area displays the help documentation for the "man" command. The text is in French and describes various options and operational modes.

```
esme@esme-VirtualBox: ~
Fichier Édition Affichage Rechercher Terminal Aide

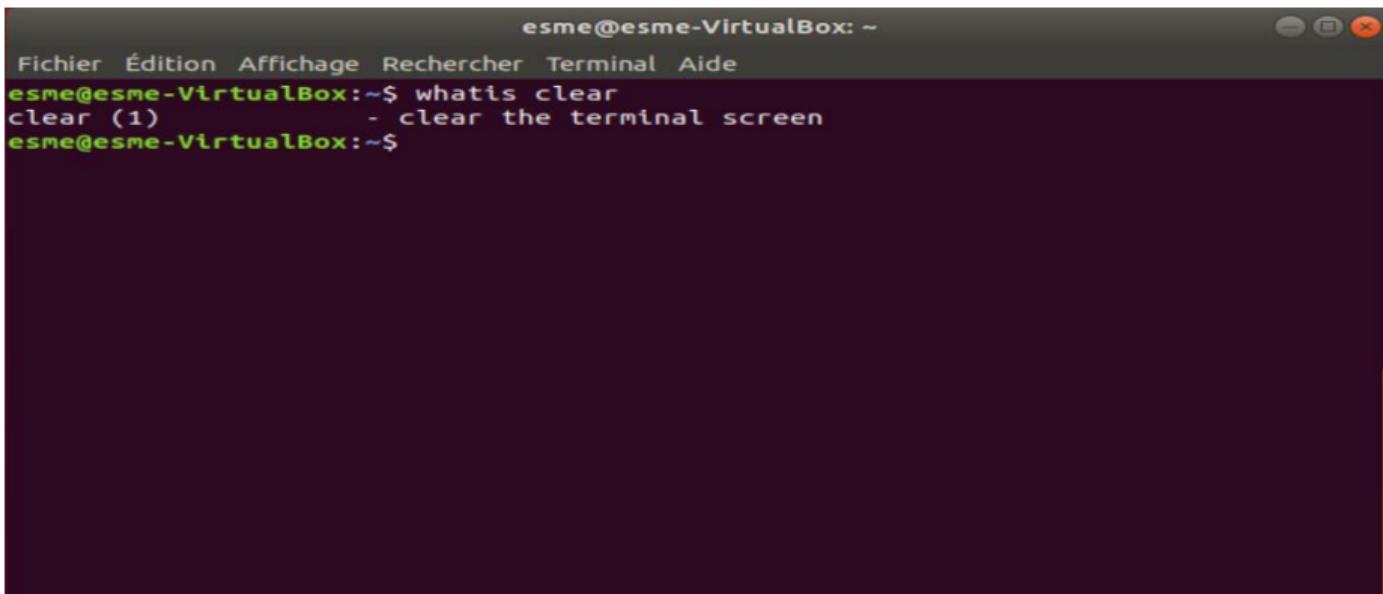
-C, --config-file=FICHIER utilise ce fichier de configuration utilisateur
-d, --debug                affichage des messages de débogage
-D, --default               réinitialise les valeurs par défaut des options
--warnings[=AVERTISSEMENTS] active les avertissements de groff

Modes opératoires principaux :
-f, --whatis            équivalent à whatis
-k, --apropos           équivalent à apropos
-K, --global-apropos    recherche le texte dans toutes les pages
-l, --local-file         interprète l'argument PAGE comme un nom de
                        fichier
-w, --where, --path, --location
-W, --where-cat, --location-cat
-s, --catman             utilisé par catman pour reformater les pages trop
                        longues
```

Commande whatis

whatis

- ❑ Usage : Afficher une ligne de description des pages de manuel.
- ❑ Exemple : whatis clear



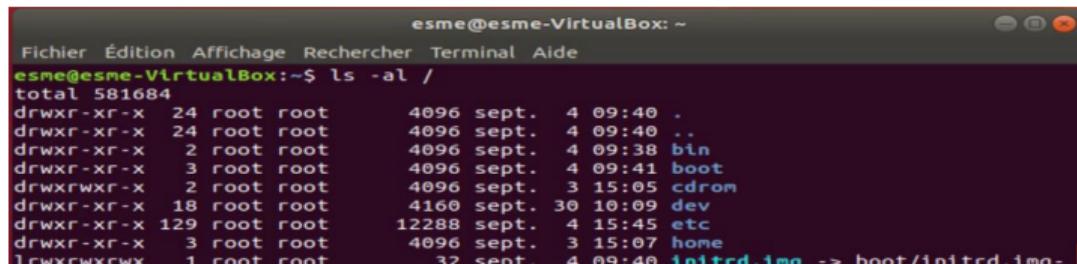
The screenshot shows a terminal window with a dark background and light-colored text. At the top, there is a menu bar with options: Fichier, Édition, Affichage, Rechercher, Terminal, and Aide. The title bar of the window reads "esme@esme-VirtualBox: ~". The terminal prompt is "esme@esme-VirtualBox:~\$". The user then types "whatis clear" and presses Enter. The output shows the man page entry for the "clear" command, which is described as "clear (1) - clear the terminal screen". The window has standard Linux-style window controls (minimize, maximize, close) in the top right corner.

```
esme@esme-VirtualBox: ~$ whatis clear
clear (1) - clear the terminal screen
esme@esme-VirtualBox: ~$
```

Commande ls

ls : list

- ❑ Usage : Permet de lister les fichiers et répertoires contenus dans un répertoire
- ❑ Syntaxe simplifiée :
 - ❑ ls [-a] [-l] [répertoire]
 - ❑ -a : affiche tous les fichiers et répertoires contenus dans répertoire : Y compris ceux qui commencent par '.' (cachés)
 - ❑ -l : affiche plus d'informations sous forme de liste
 - ❑ Exemple : ls -al /

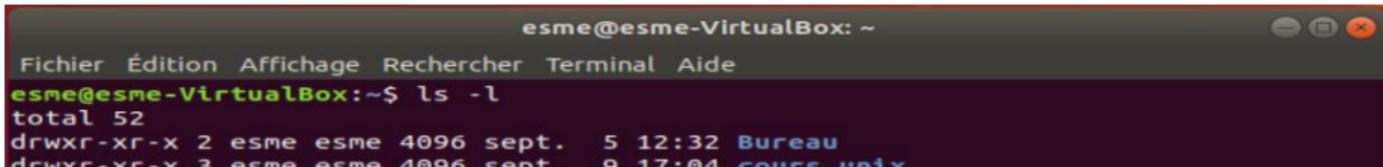


The screenshot shows a terminal window with the following content:

```
Fichier Édition Affichage Rechercher Terminal Aide
esme@esme-VirtualBox:~$ ls -al /
total 581684
drwxr-xr-x  24 root root      4096 sept.  4 09:40 .
drwxr-xr-x  24 root root      4096 sept.  4 09:40 ..
drwxr-xr-x   2 root root      4096 sept.  4 09:38 bin
drwxr-xr-x   3 root root      4096 sept.  4 09:41 boot
drwxrwxr-x   2 root root      4096 sept.  3 15:05 cdrom
drwxr-xr-x  18 root root     4160  sept. 30 10:09 dev
drwxr-xr-x 129 root root    12288 sept.  4 15:45 etc
drwxr-xr-x   3 root root      4096 sept.  3 15:07 home
lrwxrwxrwx   1 root root       32 sept.  4 09:40 initrd.img -> boot/initrd.img
```

De droite à gauche

- Nom du fichier ou répertoire
- Date avec heure de la dernière de modification
- Taille en octets
- Groupe d'appartenance
- Propriétaire du fichier/répertoire
- Le nombre de liens
- Droits d'accès (neuf caractères)
- Type de fichier (un caractère)

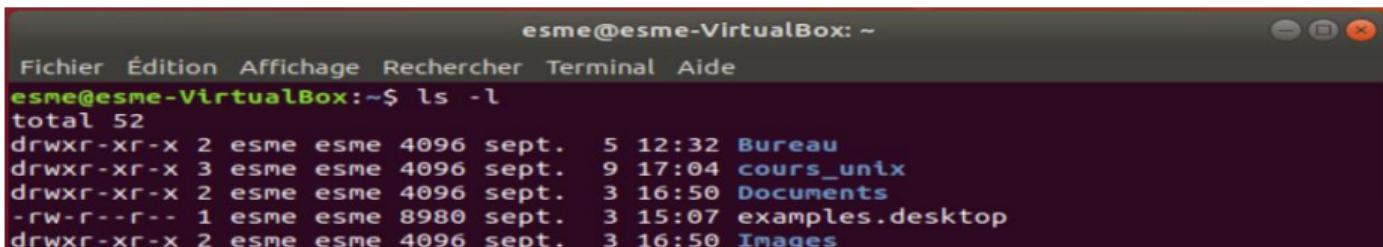


The screenshot shows a terminal window with the following content:

```
esme@esme-VirtualBox: ~
Fichier Édition Affichage Rechercher Terminal Aide
esme@esme-VirtualBox:~$ ls -l
total 52
drwxr-xr-x 2 esme esme 4096 sept. 5 12:32 Bureau
drwxr-xr-x 3 esme esme 4096 sept. 8 17:04 COURS_unix
```

Types de fichiers

- : fichier ordinaire
- d : fichier répertoire
- l : lien symbolique (peut être créé avec la commande ln)
- b : périphérique bloc (fichier spécial)
- c : périphérique caractère (fichier spécial)
- p : tubes nommés ou FIFO

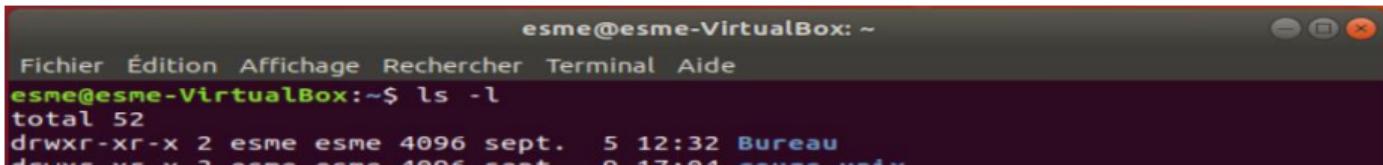


The screenshot shows a terminal window with a dark background and light-colored text. At the top, the window title is "esme@esme-VirtualBox: ~". Below the title is a menu bar with options: Fichier, Édition, Affichage, Rechercher, Terminal, Aide. The main area of the terminal displays the output of the "ls -l" command:

```
esme@esme-VirtualBox:~$ ls -l
total 52
drwxr-xr-x 2 esme esme 4096 sept.  5 12:32 Bureau
drwxr-xr-x 3 esme esme 4096 sept.  9 17:04 cours_unix
drwxr-xr-x 2 esme esme 4096 sept.  3 16:50 Documents
-rw-r--r-- 1 esme esme 8980 sept.  3 15:07 examples.desktop
drwxr-xr-x 2 esme esme 4096 sept.  3 16:50 Images
```

Droits d'accès

- Lire de gauche à droite, trois par trois
- De l'utilisateur (user) : u
- Des membres du groupe (group) : g
- Des autres utilisateurs (others) : o
- Types de droit :
 - r : permission en lecture
 - w : permission en écriture
 - x : permission d'exécution ou de traverser un répertoire
 - : permission refusée



```
esme@esme-VirtualBox: ~
Fichier Édition Affichage Rechercher Terminal Aide
esme@esme-VirtualBox:~$ ls -l
total 52
drwxr-xr-x 2 esme esme 4096 sept. 5 12:32 Bureau
drwxr-xr-x 2 esme esme 4096 sept. 5 12:32 Bureau
```

pwd : print working directory

- ❑ Usage : Affiche le nom absolu du répertoire courant.
- ❑ Syntaxe simplifiée : pwd



cd : change directory

- ❑ Usage : Changement de répertoire courant avec un chemin absolu ou relatif
- ❑ Syntaxe simplifiée : cd [répertoire]
- ❑ Exemple : cd /tmp
- ❑ Remarque : cd sans argument positionne le répertoire courant à la racine du home de l'utilisateur : Un raccourci vers le répertoire "home" de l'utilisateur courant est ~.



mkdir : make directory

- ❑ Usage : Création d'un répertoire
- ❑ Syntaxe simplifiée : `mkdir [-p] répertoire`
- ❑ `-p` : création des répertoires intermédiaires si nécessaire (Création de branches entières)
- ❑ Exemple : `mkdir rep`
- ❑ Remarque : répertoire ne doit pas correspondre à un nom déjà existant



Commande rmdir

rmdir : remove directory

- ❑ Usage : Suppression d'un répertoire si celui-ci est vide (autres que '.' et '..')
- ❑ Syntaxe simplifiée : `rmdir [-p] répertoire`
- ❑ `-p` : chaque répertoires présent dans répertoire est détruit (Destruction de branches entières)
- ❑ Exemple : `rmdir rep`



mv : move

- ❑ Usage : Renommer ou déplacer fichiers et/ou répertoires
- ❑ Syntaxe simplifiée : `mv nom_source nom_destination`
- ❑ Cela peut être un fichier ou un répertoire
- ❑ Exemple : `mv fic.txt /tmp`
- ❑ Remarque : répertoire ne doit pas correspondre à un nom déjà existant



cp : copy

- ❑ Usage : Copier des fichiers ou des arborescences.
- ❑ Syntaxe simplifiée : `cp [-r] nom_source nom_destination`
- ❑ Cela peut être un fichier ou un répertoire
- ❑ `-r` : copier une arborescence.
- ❑ Exemples :
 - ❑ `cp fic1.txt fic2.txt` : Une copie de `fic1.txt` est créée, nommée `fic2.txt`.
 - ❑ `cp fic1.txt fic2.txt rep` : `fic1.txt` et `fic2.txt` sont copiés dans le répertoire `rep`
 - ❑ `cp -r tmp tmp1` : Le répertoire `tmp` est intégralement copié dans le répertoire `tmp1`

rm : remove

- ❑ Usage : Effacer des fichiers ou arborescence.
- ❑ Syntaxe simplifiée : `rm [-r] fichier`
- ❑ Cela peut être un fichier ou un répertoire
- ❑ `-r` : effacer une arborescence.
- ❑ Exemple : `rm -r rep`
- ❑ **Attention à cette commande il n'y a pas de corbeille sous Linux**

