

# La ligne de commande du shell UNIX

mickael.hoerd@hesge.ch

# L'interface en ligne de commande

## Définition

Un interprète de ligne de texte qui permet de lancer les actions qui y sont décrites

## Deux modes d'utilisation

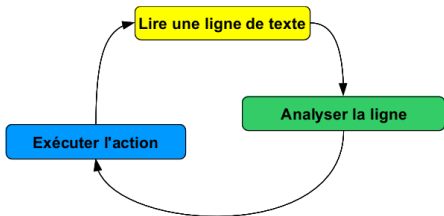
- Interactif
  - L'utilisateur tape au clavier les commandes dont il a besoin.
- Lecture d'un fichier de commandes
  - Cf. Cours sur les shell scripts.

# L'interface en ligne de commande

## Fonctionnement

un processus qui

- 1 lit une ligne de commande
- 2 l'analyse
- 3 exécute l'action indiquée
- 4 et recommence



On utilisera le shell `/bin/bash`

Plus d'informations sur les cli (Command Line Interface) :

[https://en.wikipedia.org/wiki/Command-line\\_interface](https://en.wikipedia.org/wiki/Command-line_interface)

# L'interface en ligne de commande

Une ligne de commande comporte :

- Le **nom d'un programme** à exécuter (premier mot de la ligne)
- Des **options** (pas toujours)
- Des **arguments** (pas toujours)
- Des **opérateurs du shell** (pas toujours)

Ces éléments sont séparés par des espaces

## Exemple

```
a2ps -r -Pbureau402 helloworld.c doc.txt
```

# L'interface en ligne de commande

## Exemple

```
a2ps -r -Pbureau402 helloworld.c doc.txt
```

## Détails expliqués

- **a2ps** : programme de formattage et d'impression
- **-r** : orientation "paysage"
- **-Pbureau402** : sur l'imprimante bureau402
- **helloworld.c doc.txt** : nom des fichiers à imprimer

# L'interface en ligne de commande

## Documentation intégrée (option -h ou -help)

```
a2ps --help
```

```
Usage: a2ps [OPTION]... [FILE]...
```

Convert FILE(s) or standard input to PostScript. By default, the output is sent to the default printer. An output file may be specified with -o.

Mandatory arguments to long options are mandatory for short options too. Long options marked with \* require a yes/no argument, corresponding short options stand for 'yes'.

```
--version          display version
--help             display this help
--guess           report guessed types of FILES
--which           report the full path of library files named FILES
--glob            report the full path of library files matching
--list=default    FILES display default settings and parameters
--list=TOPIC      detailed list on TOPIC (delegations, encodings, features,
variables, media, ppd, printers, prologues, styl
esheet options)
```

# L'interface en ligne de commande

## Pages de manuel (commande `man`)

```
A2PS(1)                                FSF                                A2PS(1
)

NAME
a2ps - format files for printing on a PostScript printer

SYNOPSIS
a2ps [OPTION]... [FILE]...

DESCRIPTION
Convert FILE(s) or standard input to PostScript.
By default, the output is sent to the default printer.
An output file may be specified with -o.

Mandatory arguments to long options are mandatory for short options too.
Long options marked with * require a yes/no argument,
corresponding short options stand for 'yes'.

Tasks :
--version
display version
```

# L'interface en ligne de commande

- Les pages de manuel UNIX sont réparties en 8 sections qui dépendent du type de documentation
  - 1 Commandes et applications.
  - 2 Appels systèmes.
  - 3 Fonctions de la librairie C.
  - 4 Fichiers spéciaux.
  - 5 Formats de fichiers.
  - 6 Jeux.
  - 7 Divers.
  - 8 Utilitaires d'administration système.

Exemple : page de manuel de l'appel système `open()`

```
man 2 open
```



# L'interface en ligne de commande

## Autres sources de documentation

- Commande "info"
- Documentations installée (/usr/share/doc)
- Sites pédagogiques
- Forums d'aide
- moteurs de recherche
- êtres humains

# Entrées et sorties des commandes

## Les commandes ont une entrée et une sortie "standard"

- *stdin* par défaut connectée à l'entrée du clavier.
- *stdout* par défaut connectée au terminal d'exécution de la commande.
- *stderr* par défaut aussi connectée au terminal d'exécution de la commande.

## Redirection de sortie vers un fichier : les opérateurs > et >>

- Ils redirigent la sortie standard vers un fichier donné en paramètre. L'opérateur > crée ou efface le fichier existant, l'opérateur >> continue à la fin du fichier sans l'effacer.
- **Exemple** : `a2ps -help > a2ps.txt` sauve l'aide du programme `a2ps` dans un fichier appelé `a2ps.txt`

# Entrées et sorties des commandes

## Redirection d'entrée depuis un fichier : les opérateurs < et <<

- l'opérateur < redirige l'entrée standard depuis un fichier donné en paramètre.
- L'opérateur << instruit le shell de lire l'entrée standard dans un buffer jusqu'au délimiteur fournit en paramètre, puis d'envoyer le tout dans l'entrée standard du programme.
- **Exemple** : `a2ps < a2ps.txt` imprime `a2ps.txt` car `a2ps` imprime ce qu'il reçoit sur son entrée standard.
- **Exemple** : `a2ps << abcd`  
bufferise l'entrée standard jusqu'à ce qu'il trouve "abcd", puis envoie le contenu du buffer dans le programme `a2ps`.

# Entrées et sorties des commandes

Relier la sortie d'un programme à l'entrée d'un programme :  
l'opérateur |

- Les commandes peuvent être reliées entre elles les unes à la suite des autres.
- Exemple** : `ps -cax | grep bash`  
Envoie la liste de tous les processus grâce la commande `ps` et la filtre sur l'apparence du mot "bash" grâce à la commande `grep`.
- Exemple** : `ps -cax | grep bash | wc -l`  
Fait la même chose qu'avant, mais en plus affiche le nombre de ligne contenant le mot bash. Grâce à la composition de trois processus, on obtient le nombre actuel de processus du programme bash sur la machine.

# Lancement de commandes en arrière plan

L'opérateur & à la fin d'une commande permet de lancer une commande en arrière plan

- Utile pour lancer des commandes qui prennent beaucoup de temps et faire autre chose pendant ce temps.
- Si vous l'avez oublié avant de lancer la commande : Ctrl+z et **bg** font l'équivalent.
- **exemple** : `sleep 10 &`  
Va attendre pendant 10 secondes, mais redonne la main pour l'interaction.

La commande **jobs** affichent les commandes courantes qui sont en arrière plan. La commande **fg** permet de passer une repasser une commmande en premier plan.

# Commandes UNIX indispensables

## Une liste à connaître impérativement

Nom	Paramètres de base	Fonction
mkdir	Répertoire	créé un répertoire
rmdir	Répertoire	supprime un répertoire
cd	Répertoire	change de répertoire de travail
pwd		Affiche le chemin du rép. de travail
ls		affiche le contenu du rép. de travail
cp	fichier... destination	copie fichier(s) dans destination
mv	fichier... destination	déplace fichier(s)
rm	fichier...	supprime une liste de fichier
chmod	permissions fichiers...	changer les droits d'accès
grep	pattern	Chercher un motif dans un fichier

# Commandes UNIX indispensables

## Une liste à connaître impérativement

Nom	Paramètres de base	Fonction
cat	fichier	afficher fichier
more	fichier	afficher fichier avec pause
head	-n fichier	Affiche n premières lignes
tail	-n fichier	Affiche n dernières lignes
wc	fichier	comptes octets, lignes et mots
ps	-x	afficher la liste des processus
kill	-signal pid	envoyer un signal au processus
jobs		afficher les tâches en arrière plan
passwd		changer de mot de passe
su		changer d'utilisateur
man	section commande	documentation sur une commande

# Gestion de l'édition de lignes

## Combinaison de touches spéciales

- CTRL+a revenir au début de la ligne.
- CTRL+e aller à la fin de la ligne.
- CTRL+w effacer le mot avant le curseur.
- CTRL+u effacer toute la ligne.
- CTRL+r retrouver une ligne déjà tapée.
- CTRL+c interrombre le processus courant.
- CTRL+z suspendre le processus courant.
- CTRL+s/CTRL+q stopper/reprendre l'affichage.
- CTRL+d envoyer EOF (end of file au programme).