

# UNIX

## Lecture 4 : Input/Output Redirection and Pipes

Filipe Vasconcelos<sup>1</sup>

<sup>1</sup>ESME, Lille, [filipe.vasconcelo@esme.fr](mailto:filipe.vasconcelo@esme.fr)

- ① Input/Output Redirection
- ② Pipes |



## ILO3

Use and chain together, through pipes, the main filter commands (grep, cut, head, tail, etc.) to manipulate data streams, whether or not contained in files.



① Input/Output Redirection

② Pipes |



# Streams Associated with a UNIX Command

Every Unix command has three default input/output streams:

- Standard input (stdin): Keyboard input
- Standard output (stdout): Displayed on the screen
- Standard error (stderr): Displayed on the screen

Each stream is identified by a number

- Standard input (stdin): 0
- Standard output (stdout): 1
- Standard error (stderr): 2



# Usage of Streams

## Standard input (stdin):

- STandard INput
- Input stream
- Allows the process to receive information

## Standard output (stdout):

- STandard OUTput
- Output stream
- Sends messages to the user, command execution results, information, etc.

## Standard error (stderr):

- STandard ERRor
- Output stream
- Displays errors if there are issues during command execution

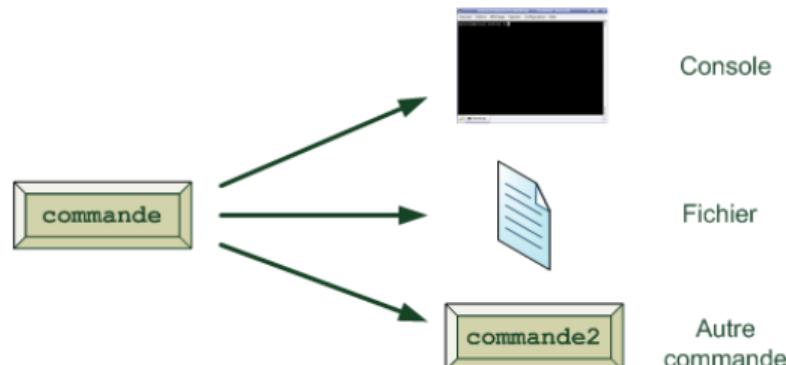


# Stream Redirection

Each of these streams associated with command execution can be redirected to:

- Keyboard, screen
- Regular files
- Streams of other commands

If the output stream is redirected to files or other commands, there will be no more display of information about the command execution result on the screen.



# Redirecting to a Regular File

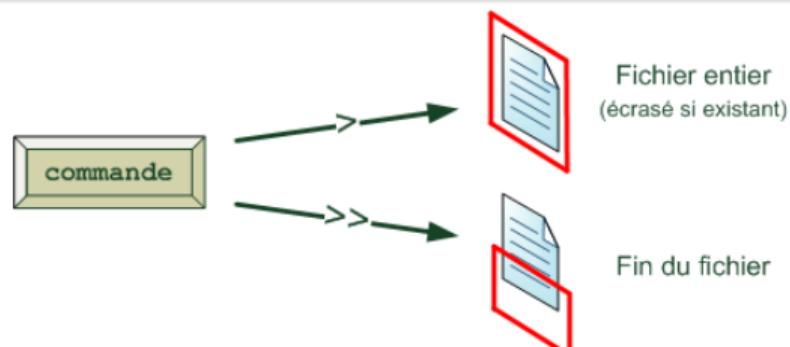
## Redirecting Standard Output (stdout)

command > file

- Standard output is redirected to the file
- If the file already exists, it will be destroyed and recreated with what stdout would have written to the screen

command >> file

- Standard output is redirected to the file
- If the file already exists, what stdout would have written to the screen is added to the end of the file



## command 2> file

- ❑ Standard error is redirected to the file
- ❑ If the file already exists, it will be destroyed and recreated with what stderr would have written to the screen

## command 2» file

- ❑ Standard error is redirected to the file
- ❑ If the file already exists, what stderr would have written to the screen is added to the end of the file

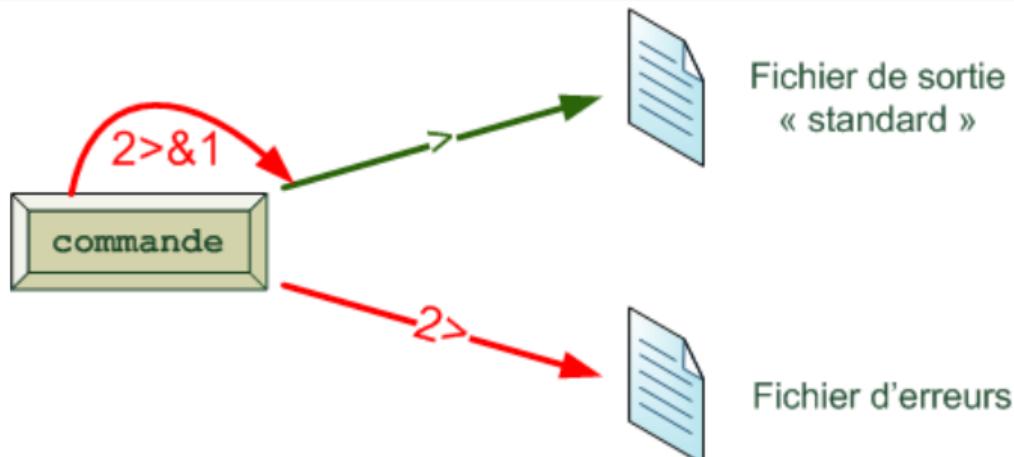


# Redirecting to a Regular File

Redirecting Standard Output (stdout) and Standard Error (stderr) Simultaneously to the Same File (Merging Outputs)

```
command > file 2>&1
```

- Standard output and standard error (stderr) are redirected to the same file simultaneously



# Redirection from/to a Regular File

## Redirecting Standard Input (stdin)

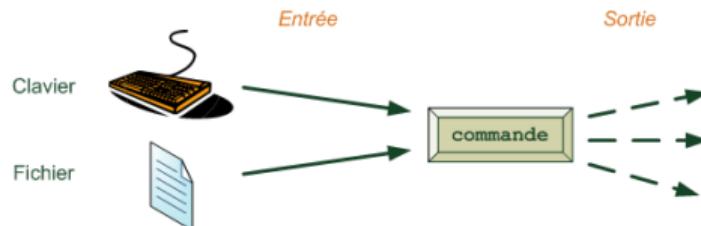
- command < file
- Data is read from the file by the command

## Discarding Outputs

- command > /dev/null
- The file /dev/null is a special system file that doesn't record anything. Any information written to it is discarded.

## Combining Redirections

- Example: Sort a file1 and save the result in file2:
- sort < file1 > file2



① Input/Output Redirection

② Pipes |



# Using Pipes to Chain Commands

- ❑ Pipes allow you to connect the output of one command to the input of another command.
- ❑ Everything that comes out of command1 is immediately sent to command2.

To use the output of one command with another:

- ❑ command1 | command2 | command3 ...
- ❑ The output of command1 is used as the input for command 2.
- ❑ Communication occurs through a buffer.

Examples

- ❑ cat file.txt | sort > sorted.txt
- ❑ env | grep -E HOME
- ❑ ls -l / | grep -E root

