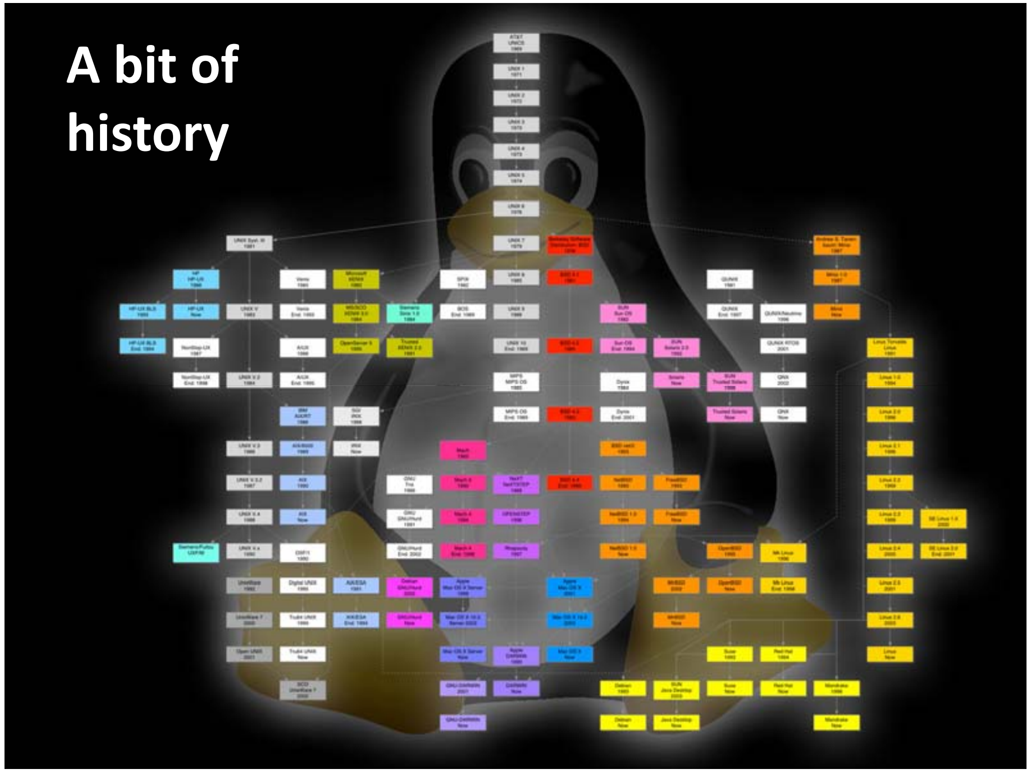
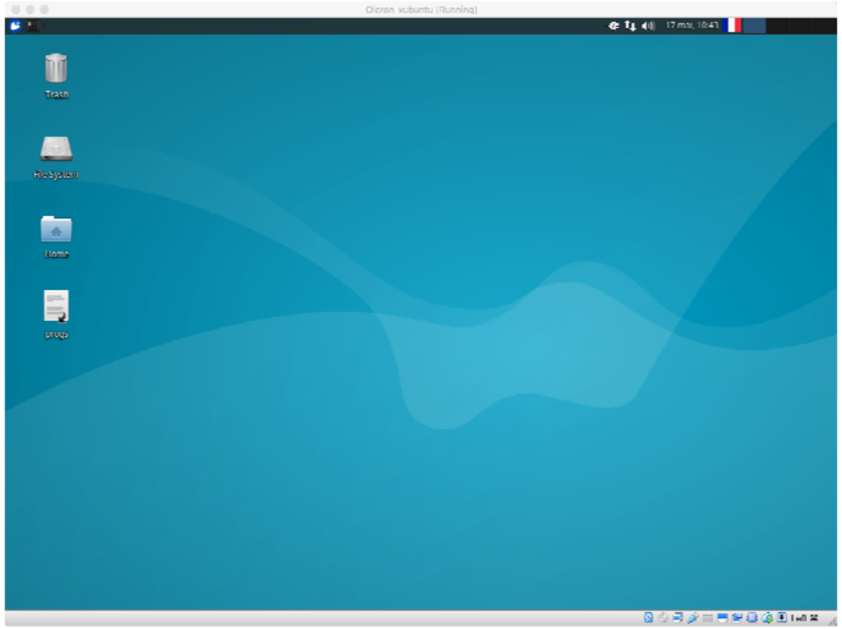
Linux FOR DUMMIES

Laurent Maveyraud, Oléron 2016

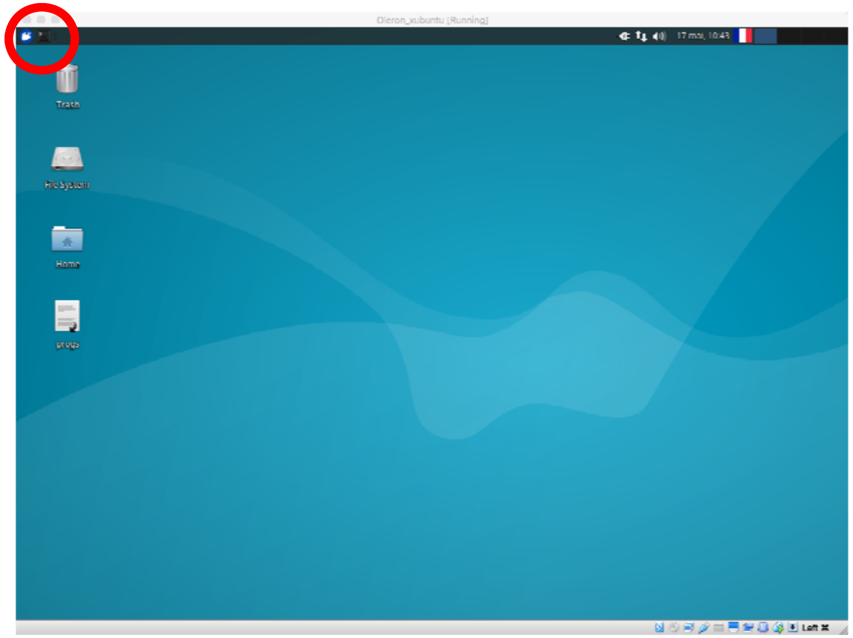
A bit of history



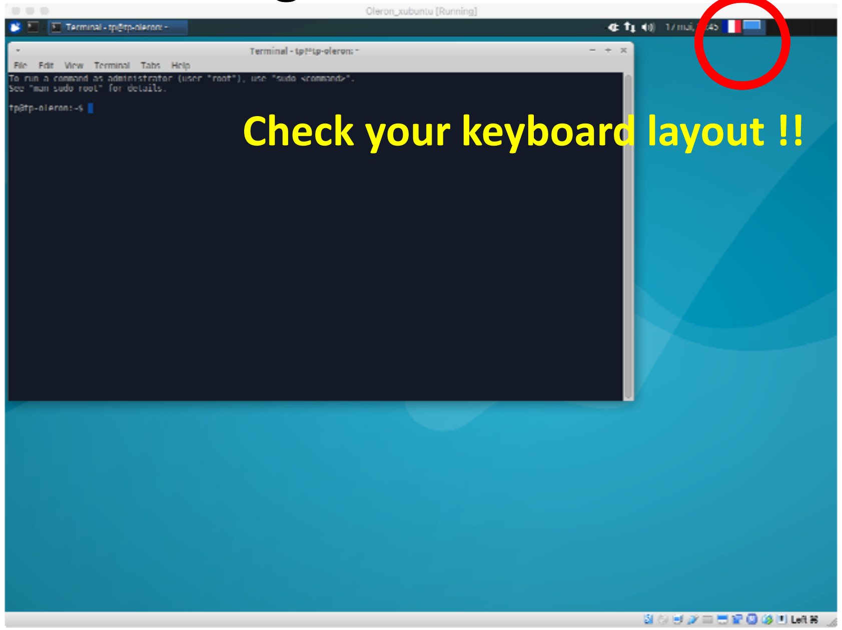
Do you know this GUI ?



Going for the terminal



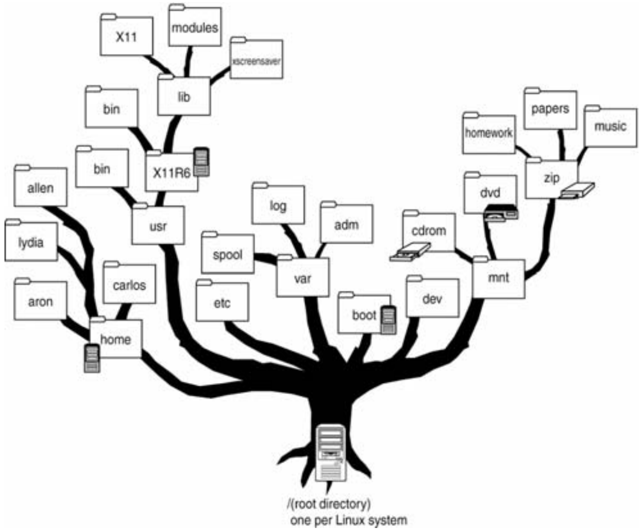
Going for the terminal


Oleron_xubuntu [Running]
Terminal - tp@tp-oleron: ~
Terminal - tp@tp-oleron: ~
File Edit View Terminal Tabs Help

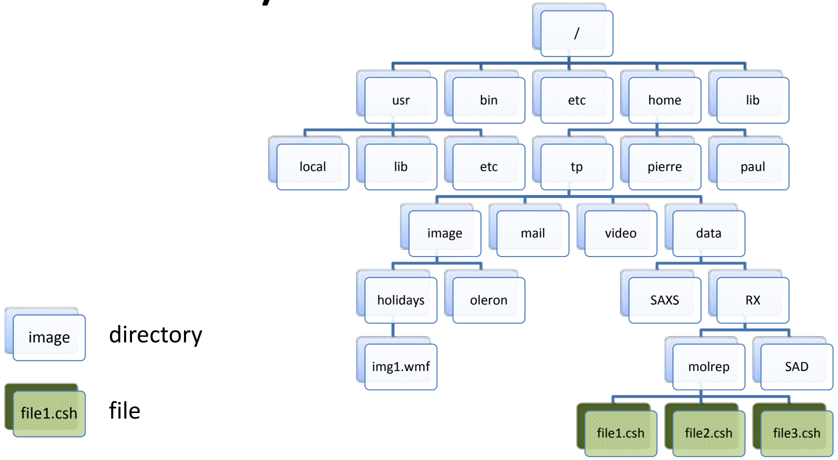
```
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

tp@tp-oleron:~$
```

Check your keyboard layout !!

Find your Path in the Tree…



Find your Path in the Tree…

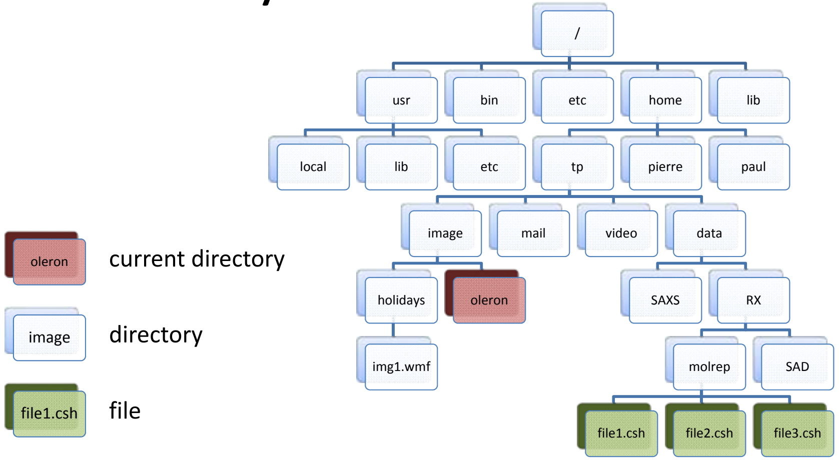


The **absolute** path for a file is the path you have to follow **from the root** to reach that file :

`/home/tp/data/RX/molrep/file1.csh`

It always starts with a / (i.e. from the root)

Find your Path in the Tree…

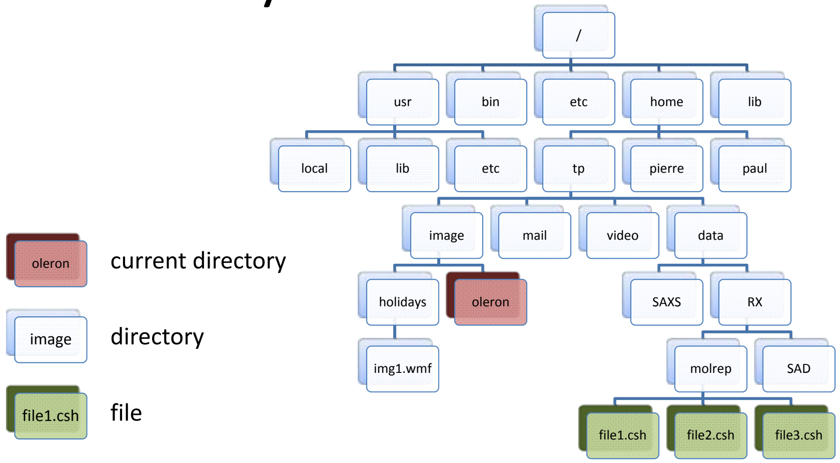


The **relative** path for a file is the path you have to follow **from the place where you are** :

`../../data/RX/molrep/file1.csh`

It never starts with a /

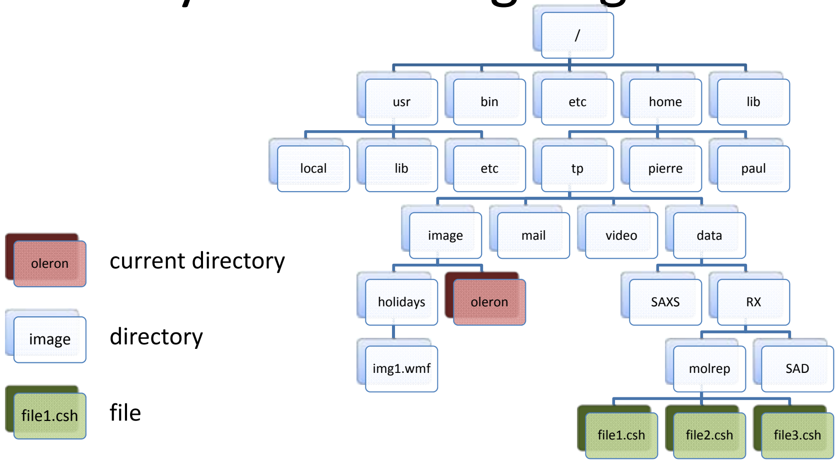
Find your Path in the Tree…



Useful commands :

```
pwd (Print Working Directory): shows where you are
cd (Change Directory): go somewhere else
ls (LiSt): list all files and folders
```

If you feel like going home…



The `homedir` is the place you are in the tree when you log in (`/home/tp`)

The shell in a nutshell

shell : the language for interacting with the system. Various flavors exist (sh, csh, tcsh, zsh, **bash**…). Mostly used in a **terminal**.

commands : tells the system what you want to do

```
command_name [options] arguments
command_name [options] arguments &
```

getting **help** about a command :

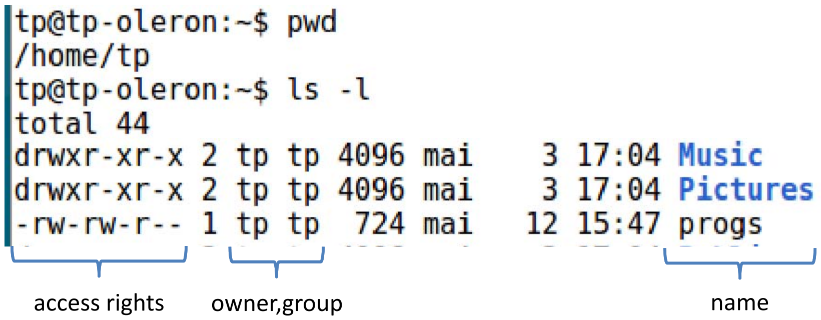
```
man command_name
```

Hit the TAB on the keybord : auto-completes commands or file/directory names.

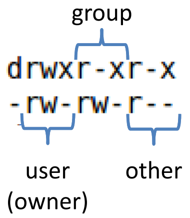
Handling files and directories

pwd : shows in which directory (folder) you are
cd : changes directory
mkdir: makes (creates) a directory
ls : lists files
ls -l: lists files with their attributes
ls -a: lists files, including hidden files
rm : removes files
rm -rf : removes files recursively, use with care !
rmdir : removes an empty directory
cp : copies file (`cp file1 file2`)
mv : renames file (`mv file1 file2`)
ln : creates link (`ln -s file_name link_name`)
. : the current working directory
.. : the directory above current working directory

Handling files and directories



```
tp@tp-oleron:~$ pwd
/home/tp
tp@tp-oleron:~$ ls -l
total 44
drwxr-xr-x 2 tp tp 4096 mai   3 17:04 Music
drwxr-xr-x 2 tp tp 4096 mai   3 17:04 Pictures
-rw-rw-r-- 1 tp tp  724 mai  12 15:47 progs
```



r: read access
w: write access
x: execute

```
chmod u+w filename
chmod 755 filename
```

Some useful commands

top: dynamics lists of running processes

du / df: disk usage, available disk space

less: displays a file (`less filename`)

find: searches for a file (`find . -name "*.img"`)

gedit: simple graphical text file editor

grep: find a word in a file

Sending data in the pipe|

Commands can be chained: the result of one command is used as the input of the next

grep toto file : display lines in file that contain toto

grep toto file | grep tata: among the lines that contain toto, display only those containing also tata

grep toto file | grep tata | grep titi: among the lines that contain toto and tata, display only those containing also titi

Sending data into > files

Normally, the result of a command is written to the screen (aka standatd output)

grep toto file : display on the screen lines in file that contain toto

grep toto file > result: do not display the result on the screen but place themn in a new file called result

grep toto file >> result: do not display the result on the screen but add them in the file called result