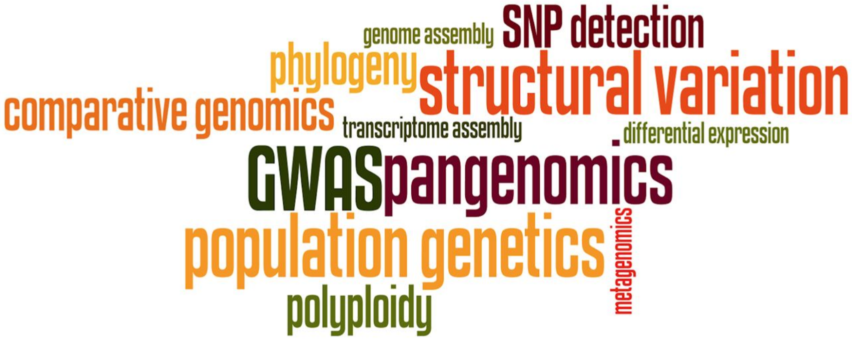
South Green
bioinformatics platform

Trainings 2019

SouthGreen bioinformatics platform

Bioinformatics platform dedicated to the genetics and genomics of tropical and Mediterranean plants and their pathogens




Rice

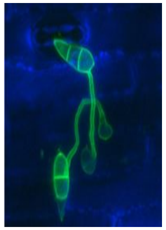

Banana


Palm


Sorghum


Coffee


Cassava


Magnaporthe

www.southgreen.fr

South Green bioinformatics platform

Charte

diade

Larmande Pierre
Sabot François
Tando Ndomassi
Tranchant-Dubreuil Christine

IPME

Comte Aurore
Dereeper Alexis

diade IPME

Orjuela-Bouniol Julie

agap

Bocs Stephanie
De Lamotte Fredéric
Droc Gaetan
Dufayard Jean-François
Hamelin Chantal
Martin Guillaume
Pitollat Bertrand
Ruiz Manuel
Sarah Gautier
Summo Marilyne

Bioversity International

Rouard Mathieu
Guignon Valentin
Catherine Breton

UMR BGPI

Mahé Frédéric
Ravel Sébastien

intertryp

Sempere Guilhem

Bioversity International
cirad
IRD Institut de Recherche pour le Développement FRANCE
INRA SCIENCE & IMPACT

South Green bioinformatics platform

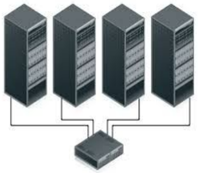
Workflow manager

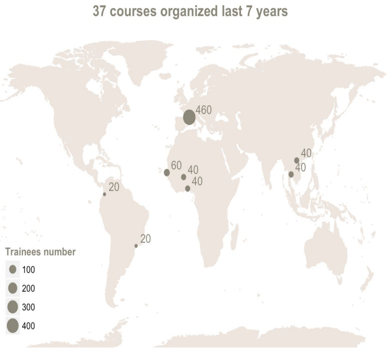
TOGGLe
Toolbox for generic NGS analyses

SNAKEMAKE

Galaxy

HPC and trainings….

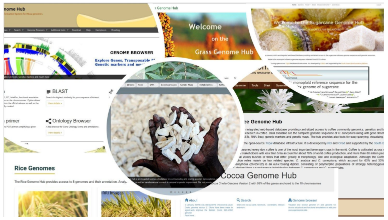




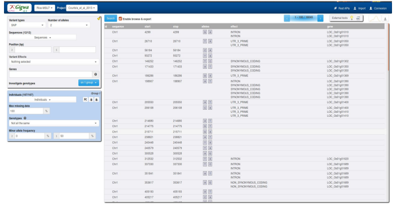
IRD
Institut de Recherche pour le Développement
FRANCE

cirad

Genome Hubs & Information System

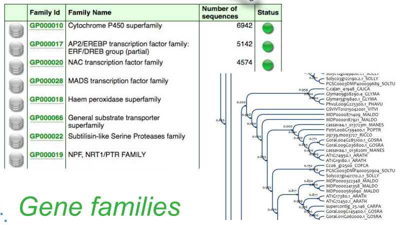


Gigwa



SNPs and Indels

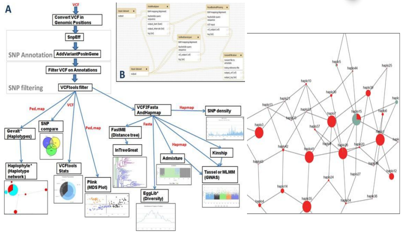
GreenPhyl



	Family Id	Family Name	Number of sequences	Status
	GP000010	Cytochrome P450 superfamily	6942	●
	GP000017	AP2/EREBP transcription factor family: ERF/DREB group (partial)	5142	●
	GP000020	NAC transcription factor family	4574	●
	GP000028	MADS transcription factor family		
	GP000018	Haem peroxidase superfamily		
	GP000066	General substrate transporter superfamily		
	GP000022	Subtilisin-like Serine Proteases family		
	GP000019	NPF, NRT1/PTR FAMILY		

Gene families

SNiPlay



https://github.com/SouthGreenPlatform

@green_bioinfo

The South Green portal: a comprehensive resource for tropical and Mediterranean crop genomics, Current Plant Biology, 2016

18-19/03 Guide de survie à Linux - IRD
21/03 Initiation à l'utilisation du cluster CIRAD – CIRAD
22/03 Initiation à l'utilisation du cluster itrop - IRD

15-16/04 Initiation au gestionnaires de workflow SG & Gigwa – IRD
18-19/04 Guide du Jedi en Linux & bash - CIRAD

13–16/05 Python - IRD
17/05 Initiation aux analyses de données transcriptomiques – IRD
21/05 Utilisation avancée du cluster IRD – IRD
23-24/05 Initiation aux analyses de données métagénomiques – IRD

6/06 Manipulation de données et figures sous R – CIRAD
26-28/06 Assemblage et annotation de transcriptomes - IRD

Trainings 2019

- **South Green Trainings** :
  https://southgreenplatform.github.io/trainings/
- Slides & Practices : Linux For Dummies
- Working environmant : Softwares to install

South Green
bioinformatics platform

Survival Guide to Linux

www.southgreen.fr

https://southgreenplatform.github.io/trainings

Bioversity International

cirad

INRA SCIENCE & IMPACT

IRD Institut de Recherche pour le Développement FRANCE

Montpellier SupAgro

Learning objectives

The objectif!

Run your own analysis using Linux !

After this course, you will be able to :

- Know the main Linux commands
- Move into the Linux file tree : ***pwd***, ***ls***, ***cd***, ***mkdir*** etc.
- Connect to a Linux server and transfer data : ***ssh***, ***scp***, ***wget***
- Work with text files: ***head***, ***tail***, ***sort***, ***cut***, ***wc***, ***grep***
- Chain and combine commands
- Run programs from the command line

Introduction

What is Linux?

- Operating system well known for :
  - its security and stability
  - its frequent updates
  - its (no) fees and openSource (mostly) softwares

- Created in 1991 by ***Linus Torvalds***

- Based on Unix (1969)

- Linux source code *opensource* and *free* : copy, modify, redistribute

What is Linux?

- **Robust et multi-plateform OS**

  (computer, server, android….)

- **Multi-users system**

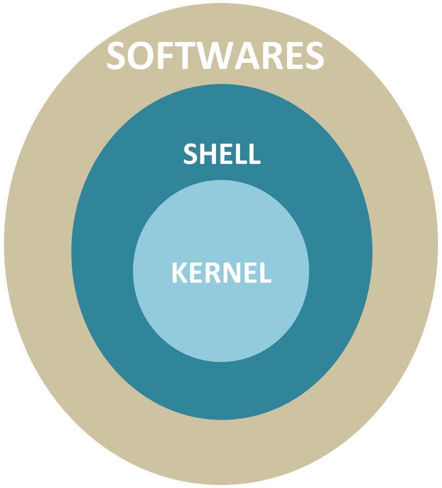
  Several users can work simultaneously

- **Multi-tasking system (processes/programs)**

Every user can run several programs at the same time

Linux Distribution

Distribution : Kernell + Shell + Softwares



How to use Linux?

- 2 ways:

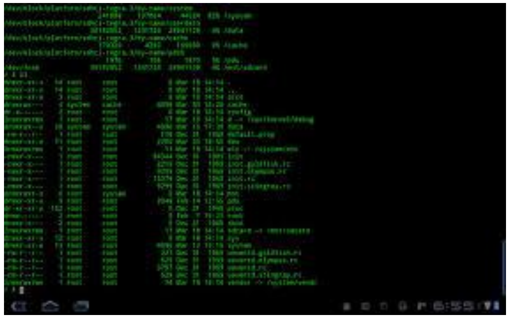
Graphical User Interface

How to use Linux?

- 2 ways :

Graphical User Interface

Command-Line Interface through a terminal

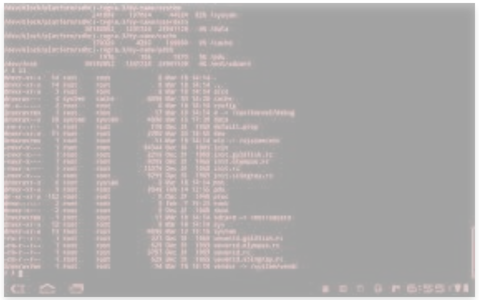


Why using Linux ?

- Numerous fast and powerful programs
- Easy to link commands and programs (workflow)
- Numerous bioinformatics softwares available
- 90% of servers on Linux

Why using Linux ?

- Numerous fast and powerful programs
- Easy to link commands and programs (workflow)
- Numerous bioinformatics softwares available
- 90% of servers on Linux



No graphical interface

Command line ergonomy ?

Why using Linux ?

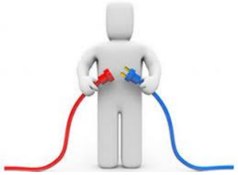
Need to practice

⬄ **Need important investissments to have good results quickly**

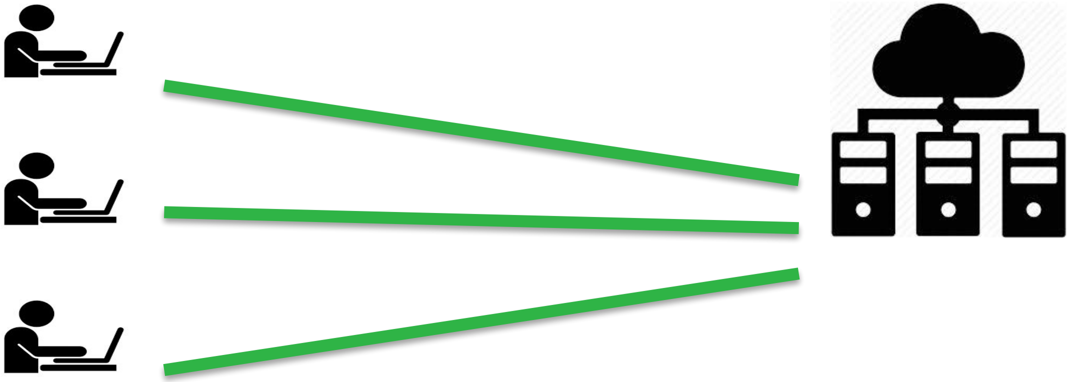
Working Environment

How to transfer files from your PC to the server ?

How to transfer files from a computer to a server?



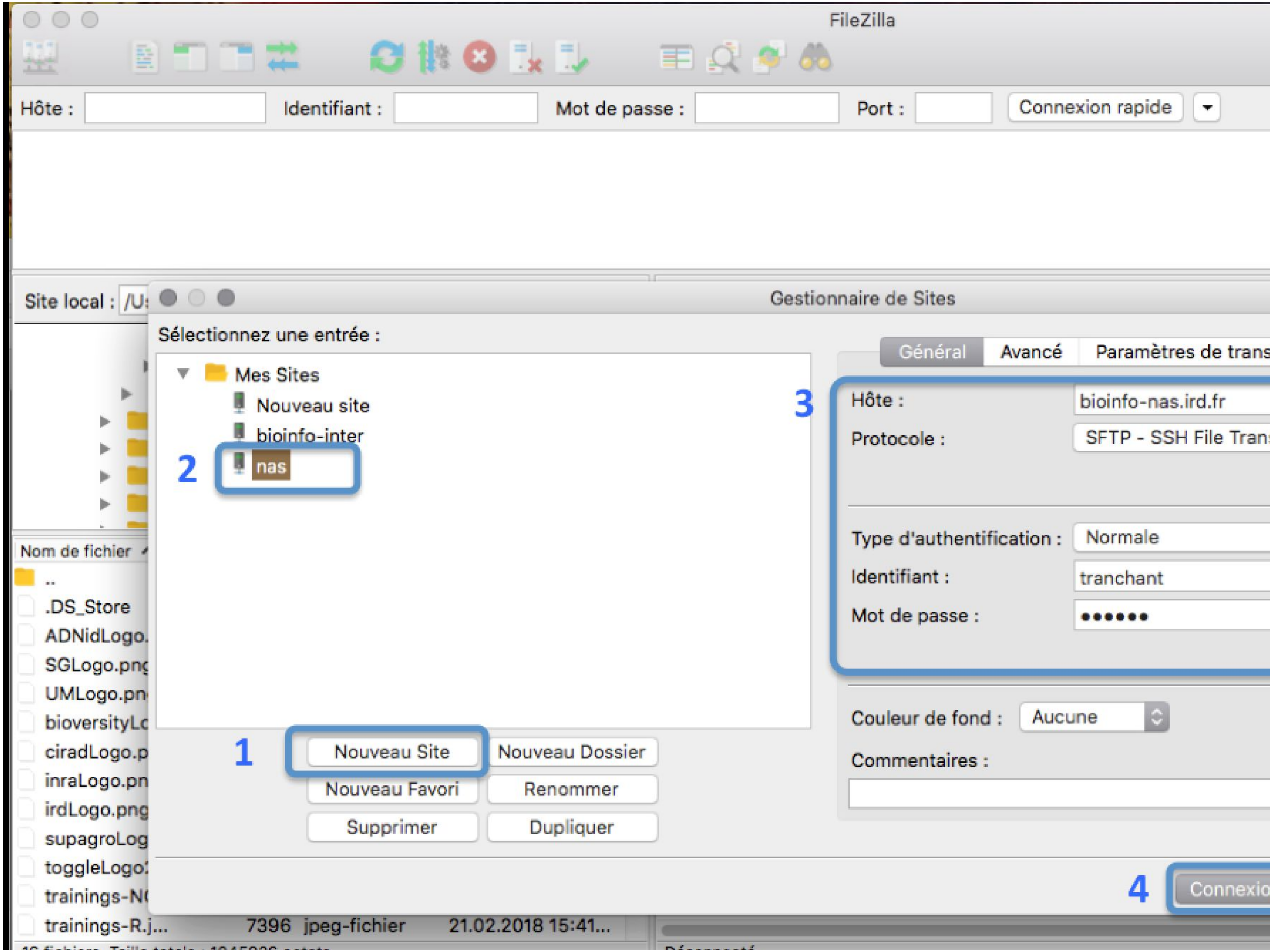
- Get connecting on a distant linux server from your computer by ***sftp protocol***
-



HPC South Green

- `itrop (IRD)` `bioinfo-nas.ird.fr`
- `HPC AGAP (CIRAD)` `cc2.login.cirad.fr`

Filezilla



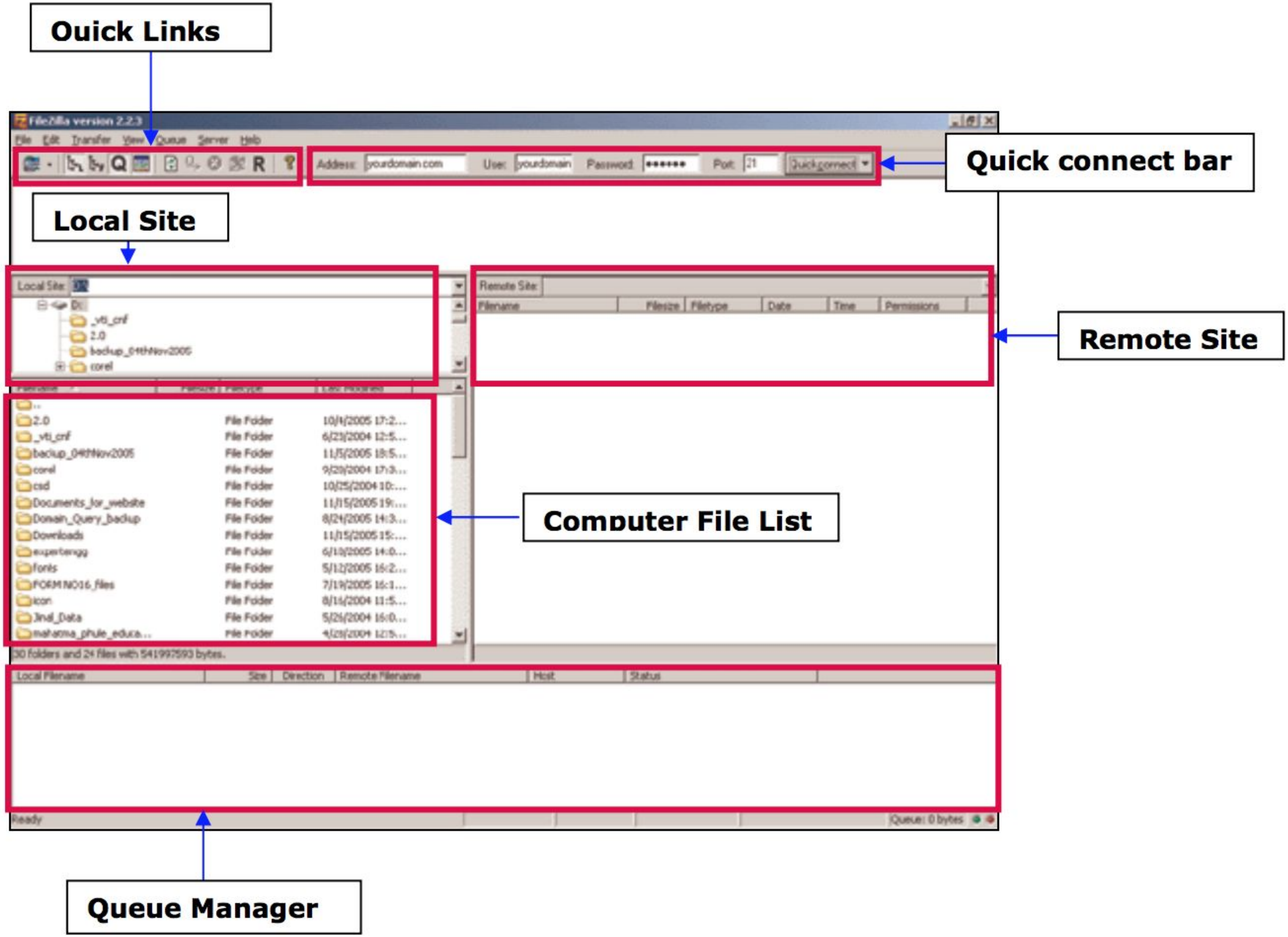
1 Nouveau Site

2 nas

3
Hôte : bioinfo-nas.ird.fr
Protocole : SFTP - SSH File Tran
Type d'authentification : Normale
Identifiant : tranchant
Mot de passe : ••••••

4 Connexio

Filezilla



Quick Links

Quick connect bar

Address: yourdomain.com User: yourdomain Password: ••••••• Port: 21

Local Site

Local Site: D:\

- D:
  - _vti_cnf
  - 2.0
  - backup_04thNov2005
  - corel

Remote Site

Computer File List

Filename	Filesize	Filetype	Last Modified
..			
2.0		File Folder	10/4/2005 17:2...
_vti_cnf		File Folder	6/23/2004 12:5...
backup_04thNov2005		File Folder	11/5/2005 18:5...
corel		File Folder	9/20/2004 17:3...
csd		File Folder	10/25/2004 10:...
Documents_for_website		File Folder	11/15/2005 19:...
Domain_Query_backup		File Folder	8/24/2005 14:3...
Downloads		File Folder	11/15/2005 15:...
expertengg		File Folder	6/10/2005 14:0...
fonts		File Folder	5/12/2005 16:2...
FORM NO16_files		File Folder	7/19/2005 16:1...
icon		File Folder	8/16/2004 11:5...
Jindal_Data		File Folder	5/26/2004 16:0...
mahatma_phule_educa...		File Folder	4/20/2004 12:5...

30 folders and 24 files with 541997593 bytes.

Queue Manager

Ready

Queue: 0 bytes

Practice

filezilla, sftp

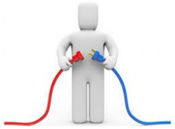
1

Go to Practice 1 *on our github*

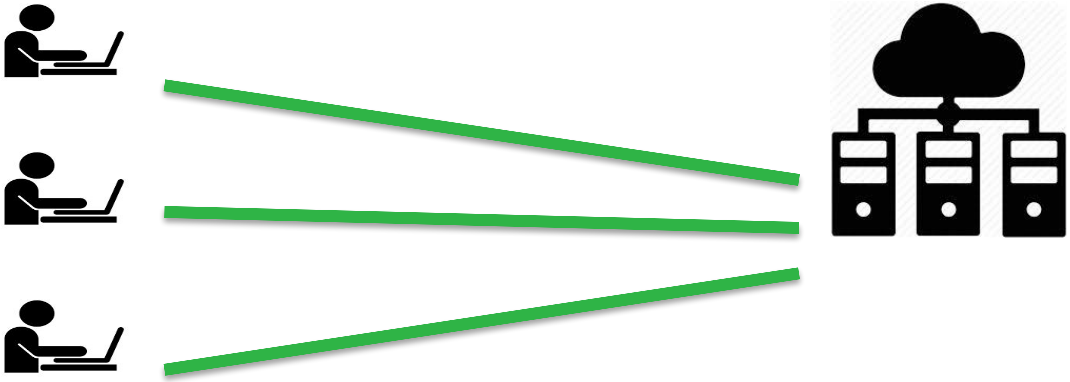
Working Environment

How to work on a distant server ?

How to work on a distant server ?



- Get connecting on a distant linux server from your computer by ***ssh protocol***

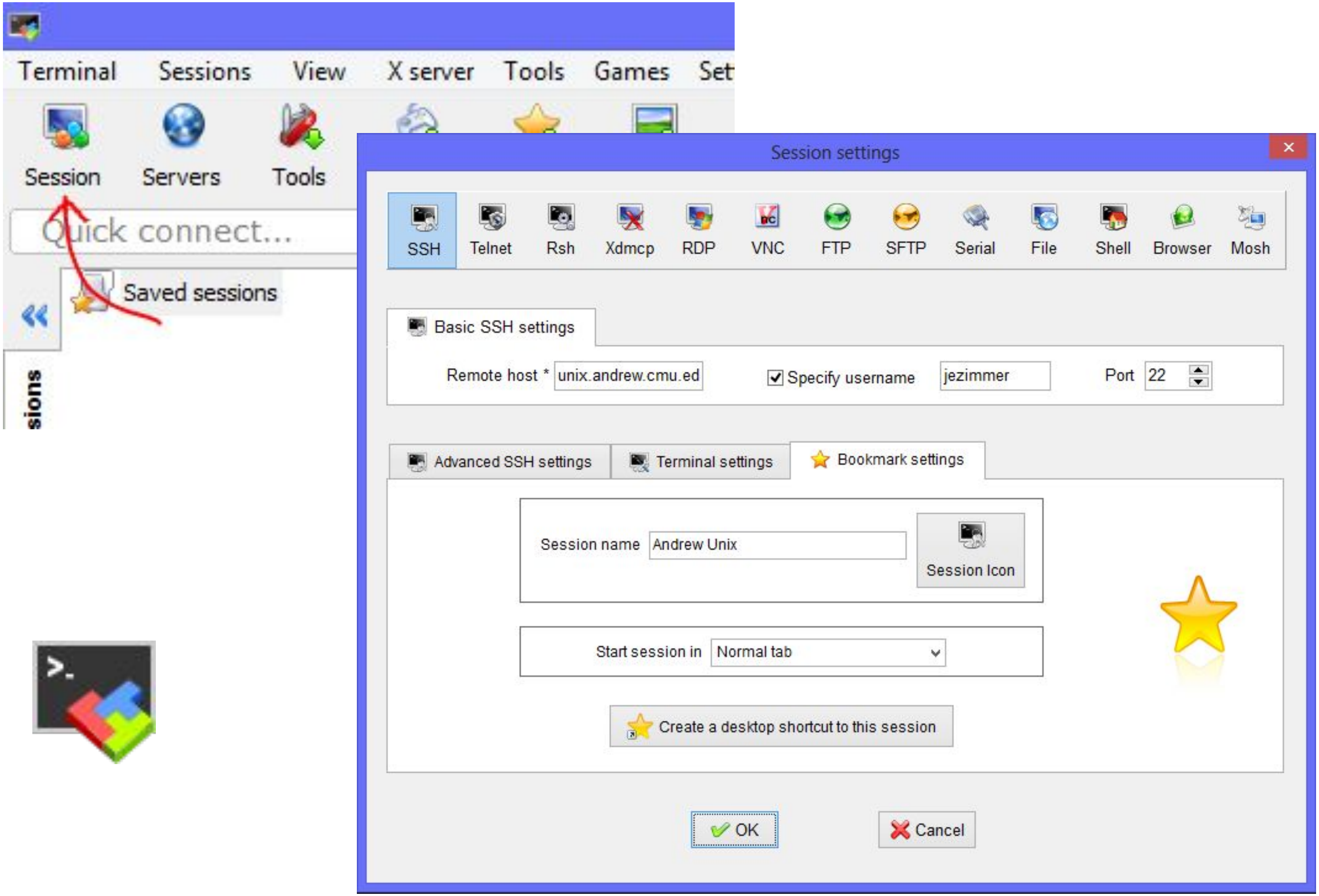


HPC South Green

● itrop (IRD)	bioinfo-inter.ird.fr
● HPC AGAP (CIRAD)	cc2-login.cirad.fr

PuTTY

How to work on a distant server ?



Remote host * unix.andrew.cmu.ed

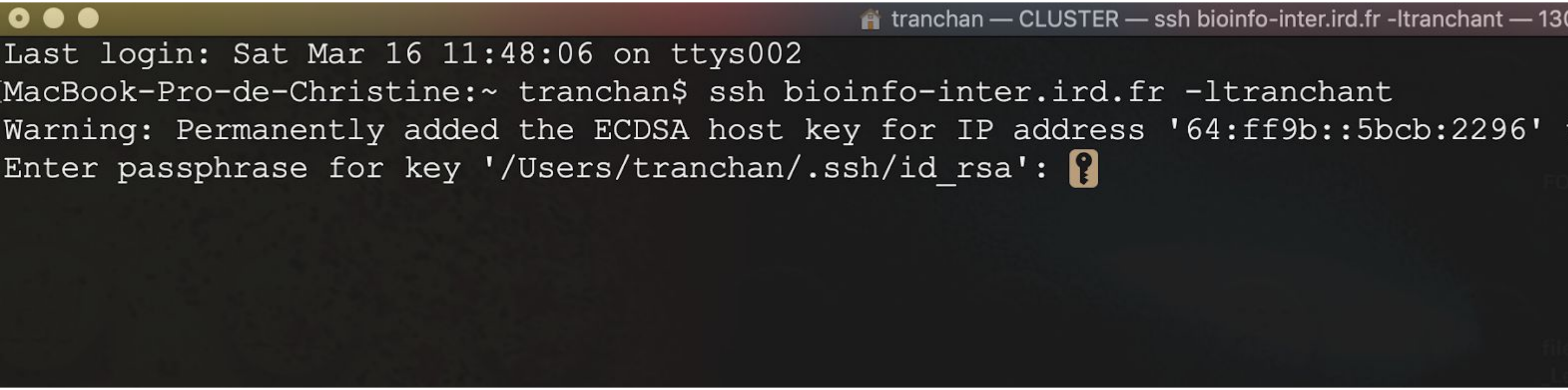
☑ Specify username jezimmer

Port 22

Session name Andrew Unix

Start session in Normal tab

Comment travailler sur le serveur ?



```
Last login: Sat Mar 16 11:48:06 on ttys002
MacBook-Pro-de-Christine:~ tranchan$ ssh bioinfo-inter.ird.fr -ltranchant
Warning: Permanently added the ECDSA host key for IP address '64:ff9b::5bcb:2296'
Enter passphrase for key '/Users/tranchan/.ssh/id_rsa':
```

Practice

putty,
terminal, ssh

2

Go to Practice 2 *on our github*

South Green bioinformatics platform

First steps on Linux

Commands for moving around the file system and manipulating files/folders

The Prompt

Always on the terminal, just before where user type commands

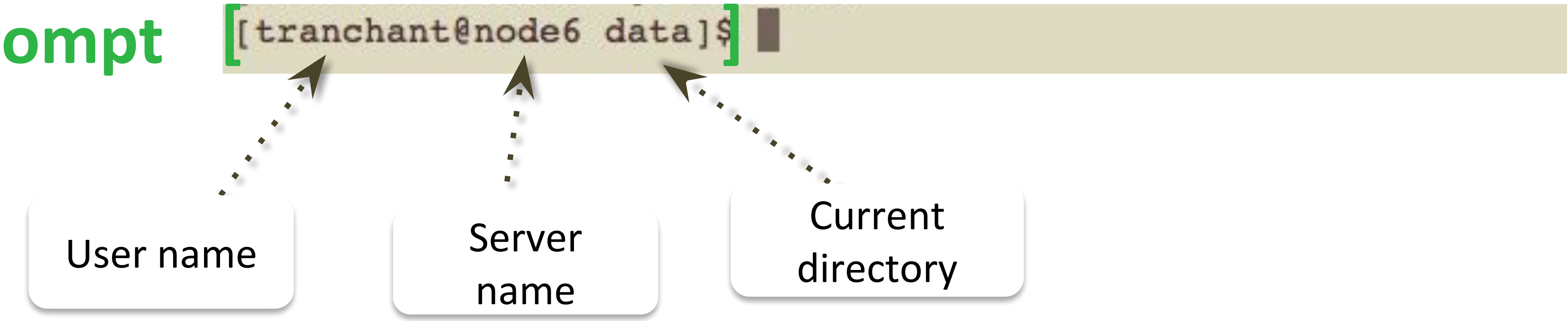
Prompt

```
[tranchant@node6 data]$
```

The Prompt

Always on the terminal, just before where user type commands

Prompt



```
[tranchant@node6 data]$
```

User name

Server name

Current directory

Command syntax

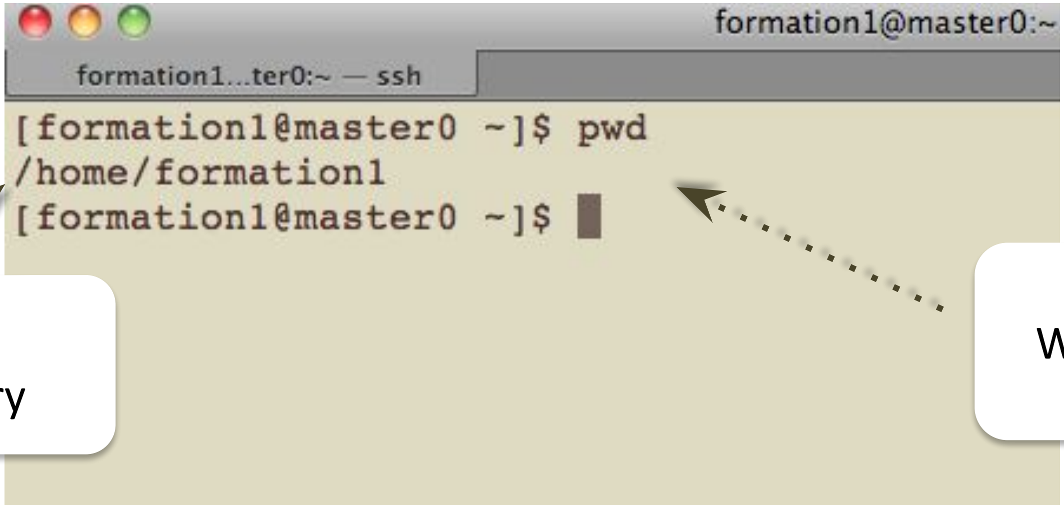
```
command [ -options ] [ arguments ]
```

Your First command *pwd*

pwd

__P__resent __W__ork __D__irectory

Print the name of the current directory (the full path)



```
[formation1@master0 ~]$ pwd
/home/formation1
[formation1@master0 ~]$ 
```

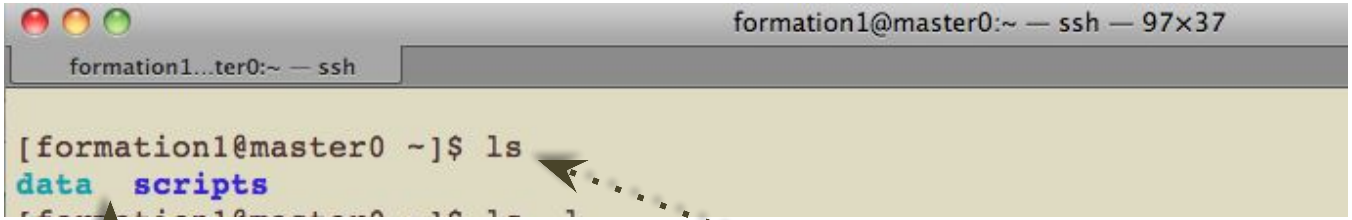
Name of the current directory

Command
Without option and argument

2nd commande *ls*

ls
list

List the content of the current directory



```
[formation1@master0 ~]$ ls
data   scripts
```

List all the files in the current directory (by default)

Command without option and argument

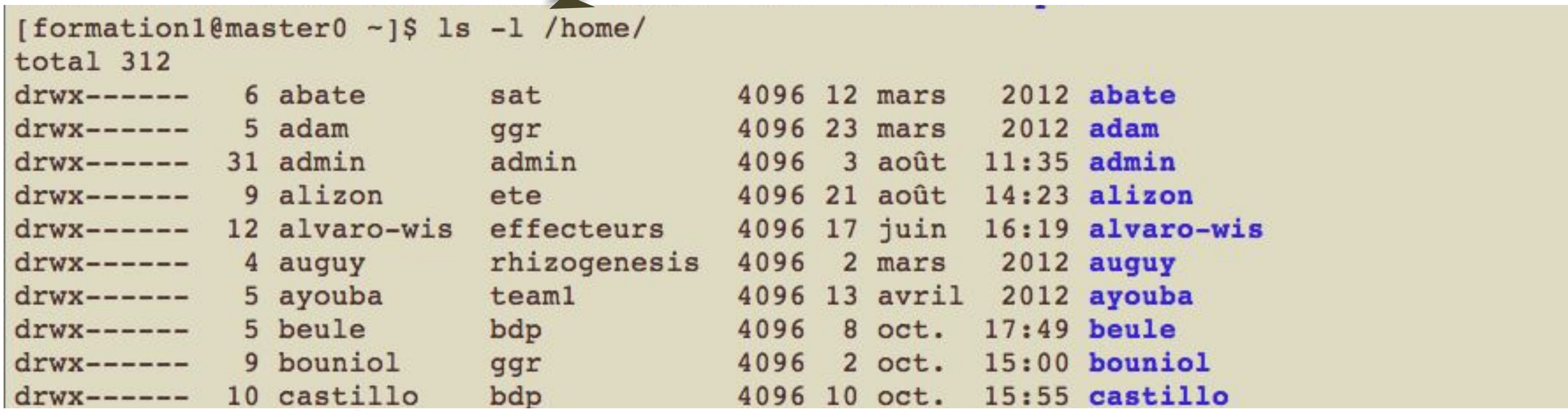
2nd command + option *ls -l*

ls -l

list long

ist files with more information about each file

Command with the option **–l** and a directory name gived as argument



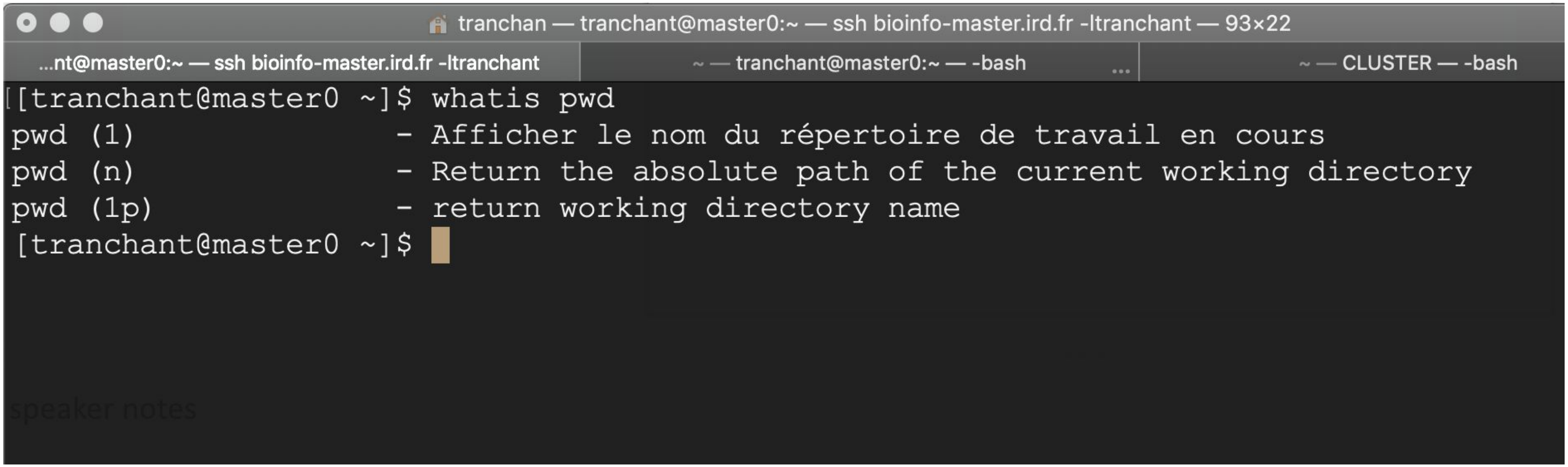
```
[formation1@master0 ~]$ ls -l /home/
total 312
drwx------   6 abate       sat          4096 12 mars   2012 abate
drwx------   5 adam        ggr          4096 23 mars   2012 adam
drwx------  31 admin       admin        4096  3 août  11:35 admin
drwx------   9 alizon      ete          4096 21 août  14:23 alizon
drwx------  12 alvaro-wis  effecteurs   4096 17 juin  16:19 alvaro-wis
drwx------   4 auguy       rhizogenesis 4096  2 mars   2012 auguy
drwx------   5 ayouba      team1        4096 13 avril  2012 ayouba
drwx------   5 beule       bdp          4096  8 oct.  17:49 beule
drwx------   9 bouniol     ggr          4096  2 oct.  15:00 bouniol
drwx------  10 castillo    bdp          4096 10 oct.  15:55 castillo
```

Display the long format listing of all files in the directory

A Few Basic Commands

How to get help about one command

- with the ’option *--help* ou *-h* — *ls* ***--help*** *blastn* ***-h***
- with the command *man* — ***man*** *ls*
- whith the command *whatis* — ***whatis*** *ls*



```
[[tranchant@master0 ~]$ whatis pwd
pwd (1)              - Afficher le nom du répertoire de travail en cours
pwd (n)              - Return the absolute path of the current working directory
pwd (1p)             - return working directory name
[tranchant@master0 ~]$ 
```

speaker notes

A Few Basic Commands

Basics

pwd	Display the full path of the current directory
ls	List all files/directories
ls –l	Display all files (Long listing)

Practice

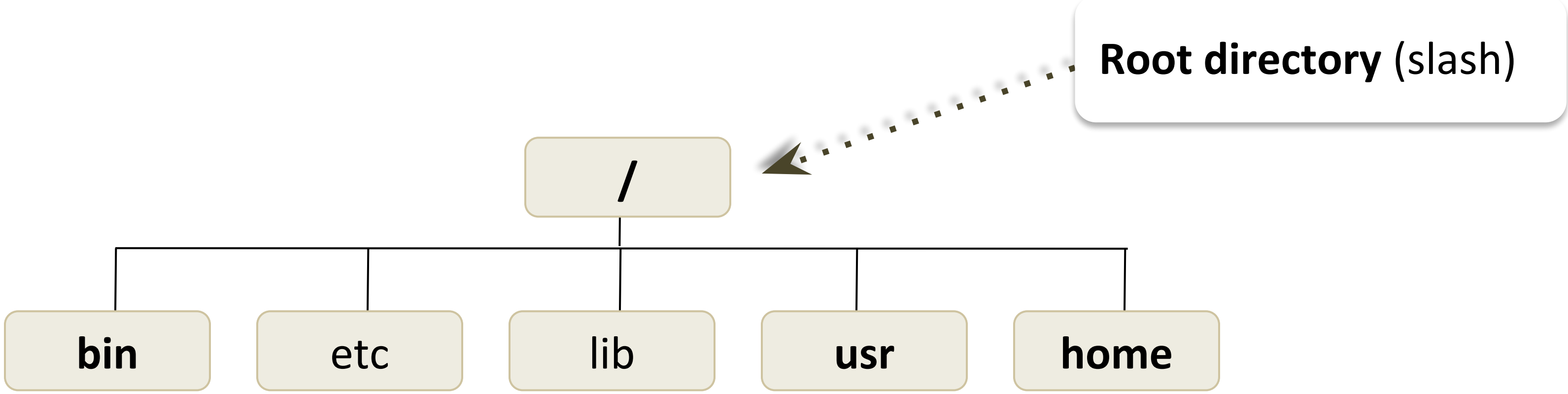
prompt, pwd

3

Go to [Practice 3](#) *on our github*

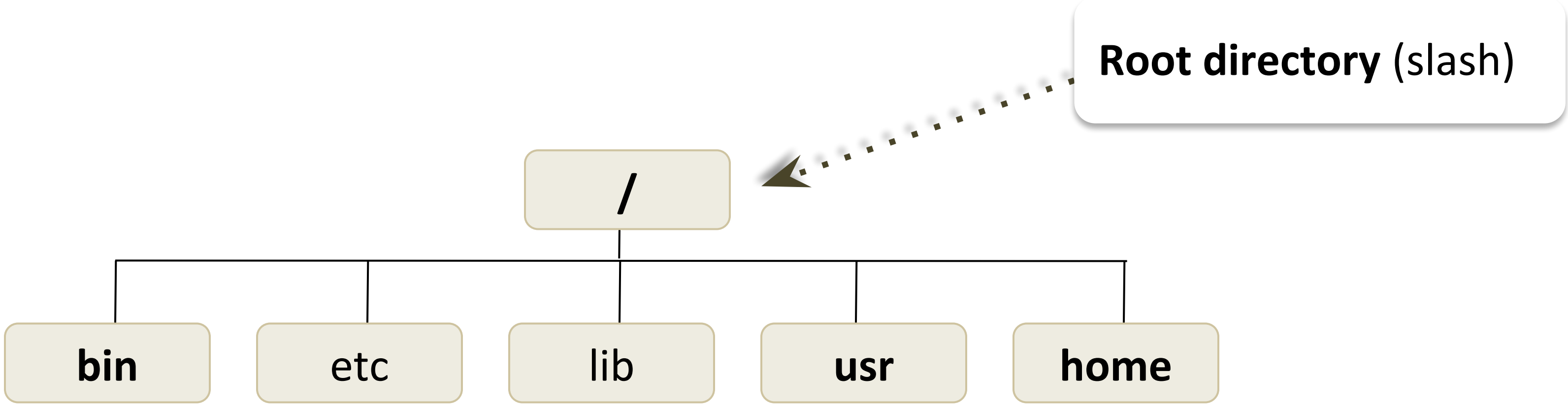
The file tree - Filesystem

- Directory structure starts at the root directory called “/” (slash)


Root directory (slash)

The file tree - Filesystem

Main directories



Root directory (slash)

/

bin

etc

lib

usr

home

/bin	**Main commands, shell, programs**
/usr, /opt	**Applications and user librairies**
/usr/bin	**Other commands**
/home	**User directory (one per user, name= login)**

Path of a file

Path : location to a file/directory in the file system

Path of a file

Path : location to a file/directory in the file system

absolute

- Complete path of a file starting from the root directory /

Path of a file

Path : location to a file/directory in the file system

absolute

- Complete path of a file starting from the root directory /

- **starts always with /**

- **always good wherever the user is working**

Path of a file

Path : location to a file/directory in the file system

absolute

- Complete path of a file starting from the root directory /
- **starts always with /**
- **always good wherever the user is working**

Path of a file

Path : location to a file/directory in the file system

absolu

- Complete path of a file starting from the root directory /
- ***starts always with /***
- **always good wherever user is working**

relative

- Path related to the present working directory (where the user is working)

Path of a file

Path : location to a file/directory in the file system

absolu

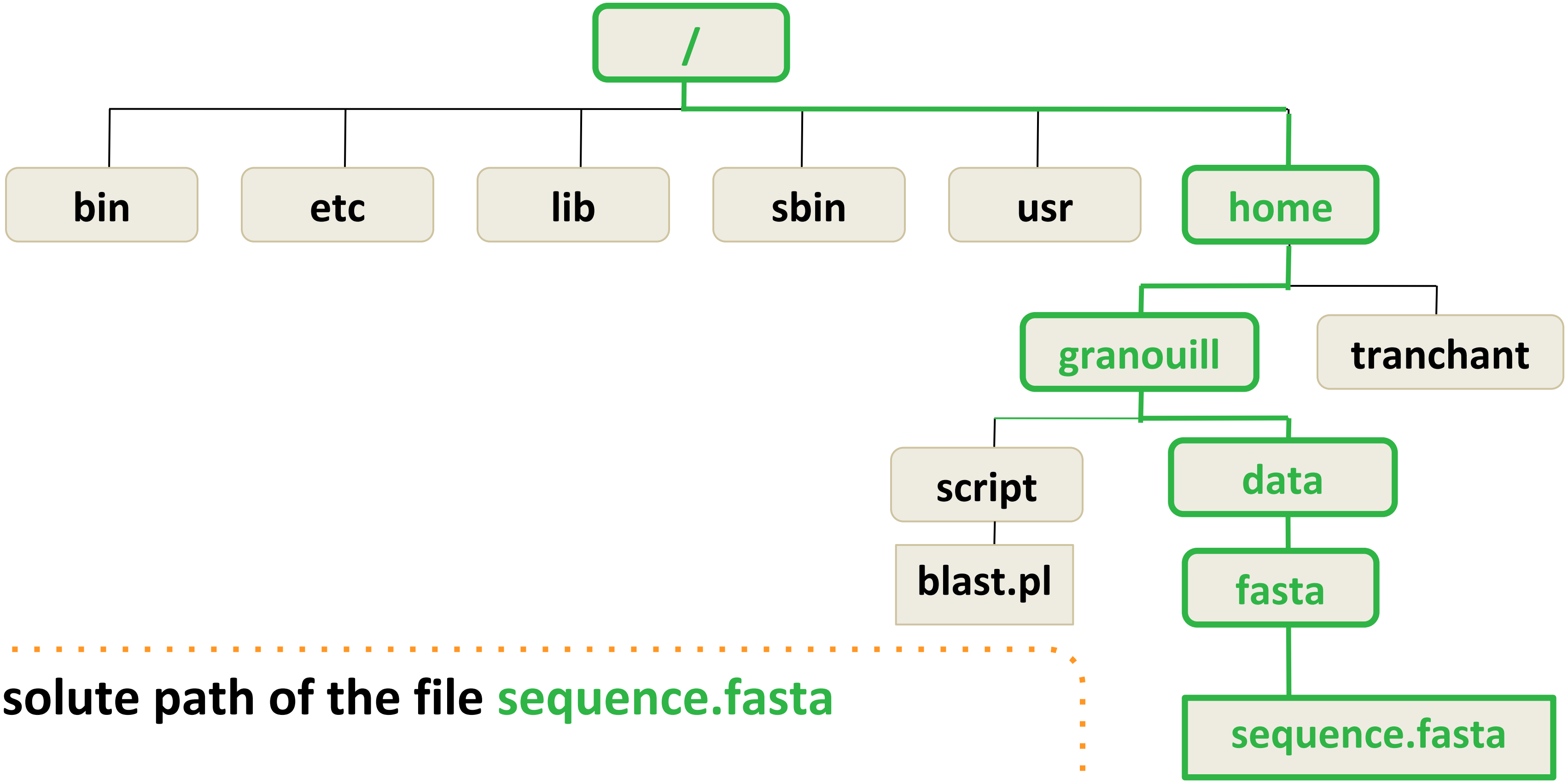
- Complete path of a file starting from the root directory /
- ***starts always with /***
- **always good wherever user is working**

relative

- Path related to the present working directory (where the user is working)
- ***Never starts with /***
- **Depends on where the user is working**

Absolute Path

- Always starts with / (root directory)
- Always works wherever user is working



/

bin

etc

lib

sbin

usr

home

granouill

tranchant

script

blast.pl

data

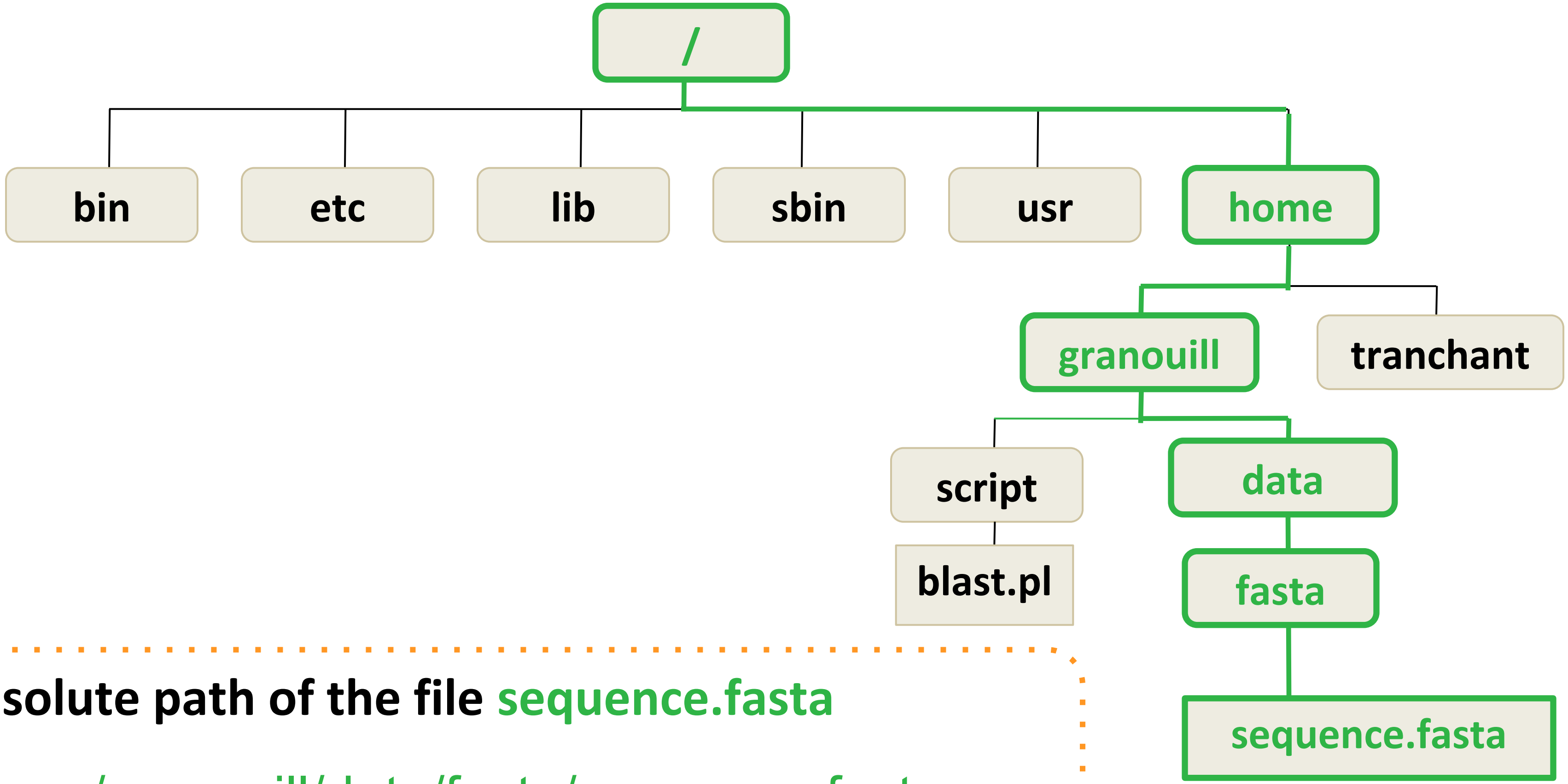
fasta

sequence.fasta

Absolute path of the file sequence.fasta

Absolute Path

- Always starts with /   (root directory)
- **Always works wherever user is working**

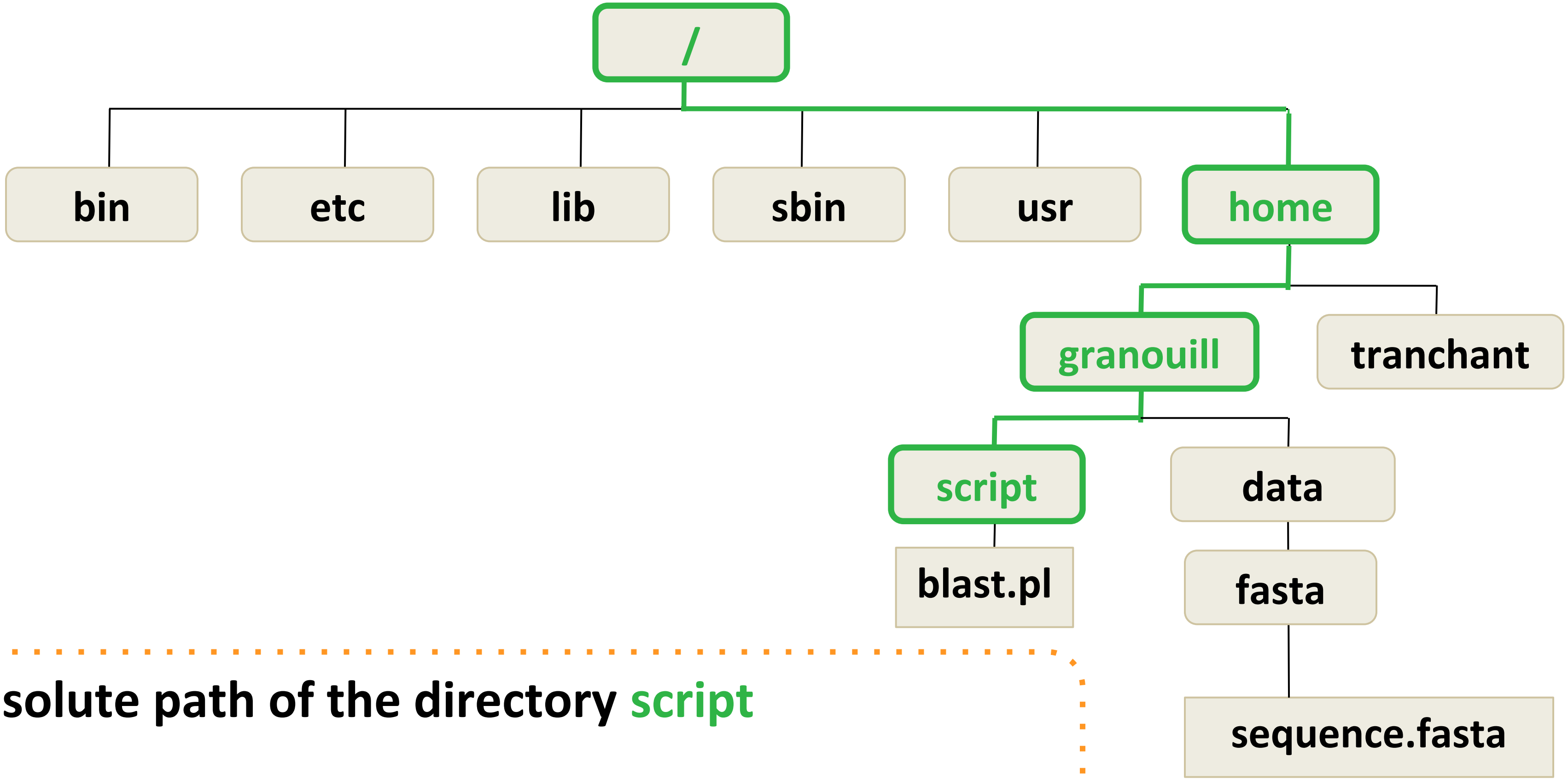


Absolute path of the file sequence.fasta

/home/granouill/data/fasta/sequence.fasta

Absolute Path

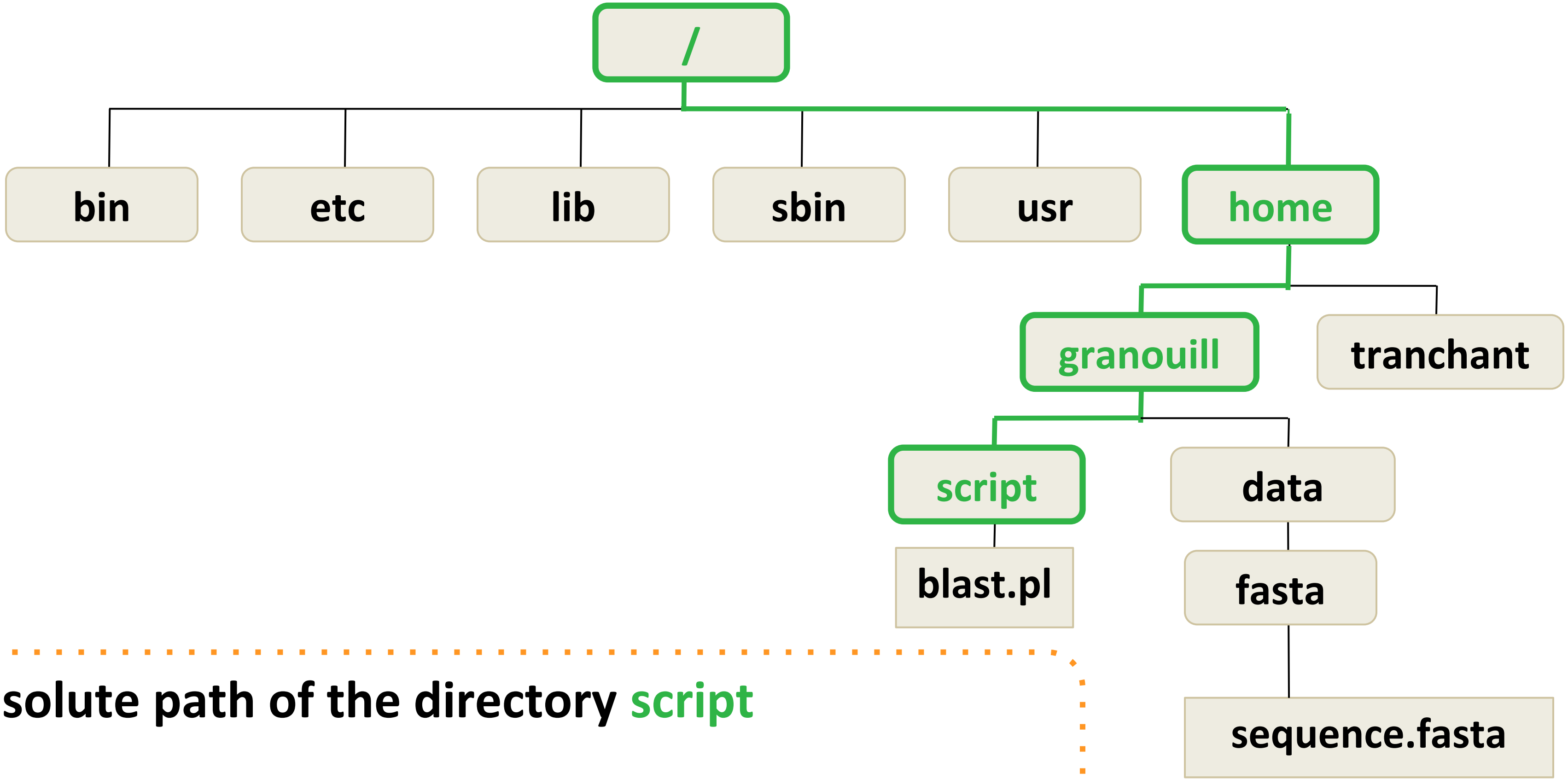
- Always starts with / (root directory)
- **Always works wherever user is working**



Absolute path of the directory script

Absolute Path

- Always starts with / (root directory)
- **Always works wherever user is working**

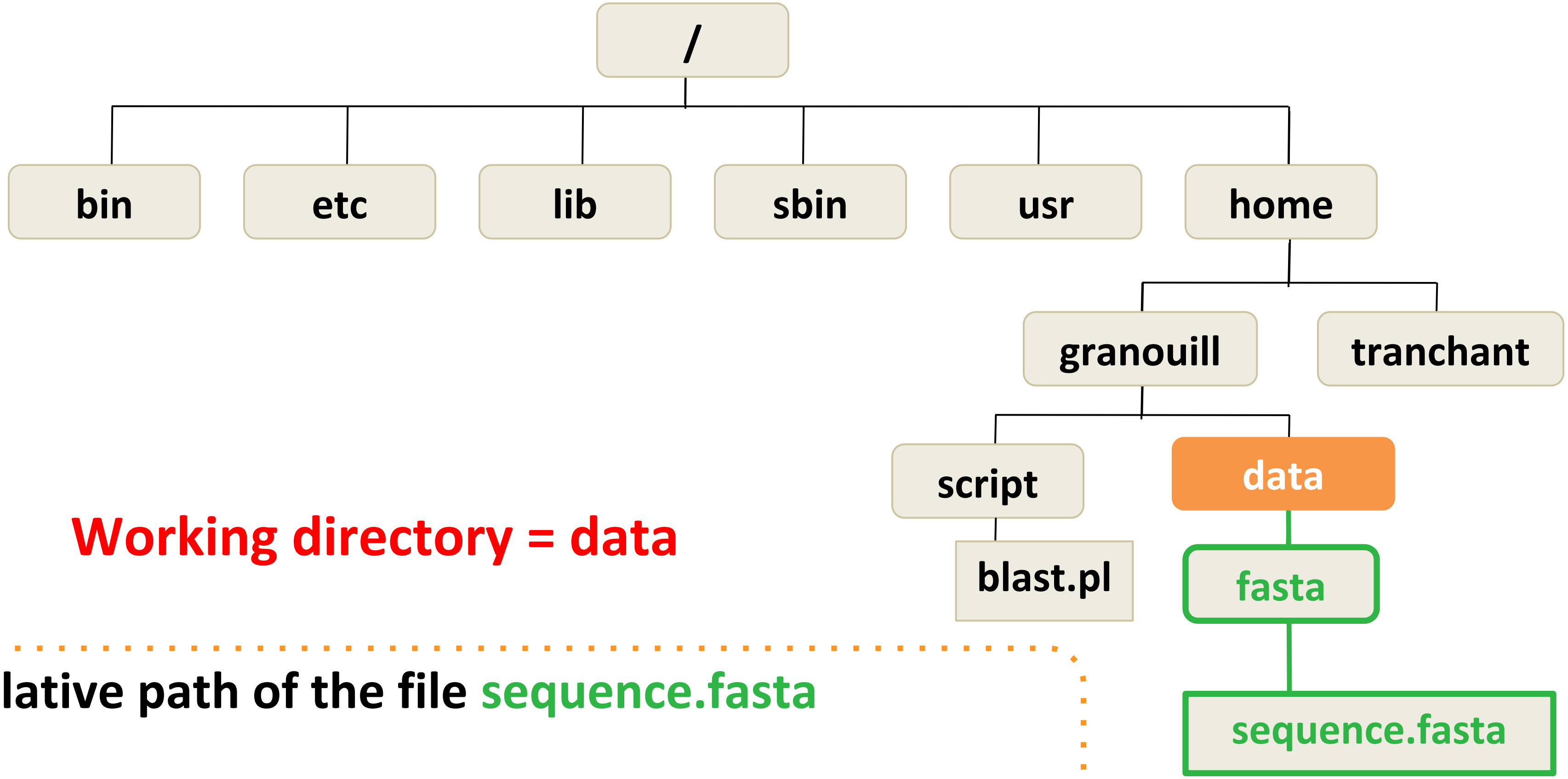


Absolute path of the directory script

/home/granouill/data/script

Relative Path

- Path related to the present working directory
- **Never starts with /**

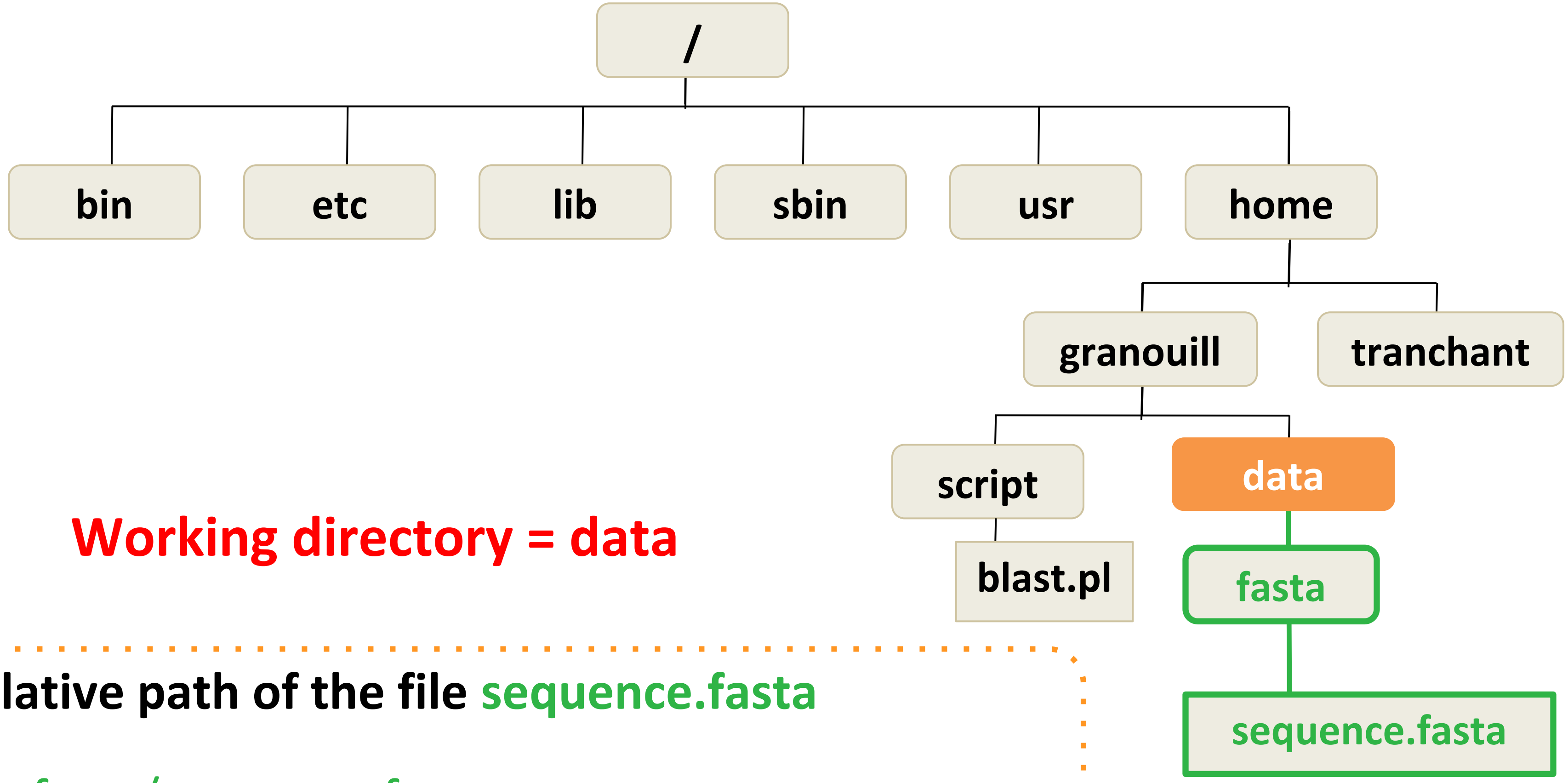


Working directory = data

Relative path of the file sequence.fasta

Relative Path

- Path related to the present working directory
- Never starts with /



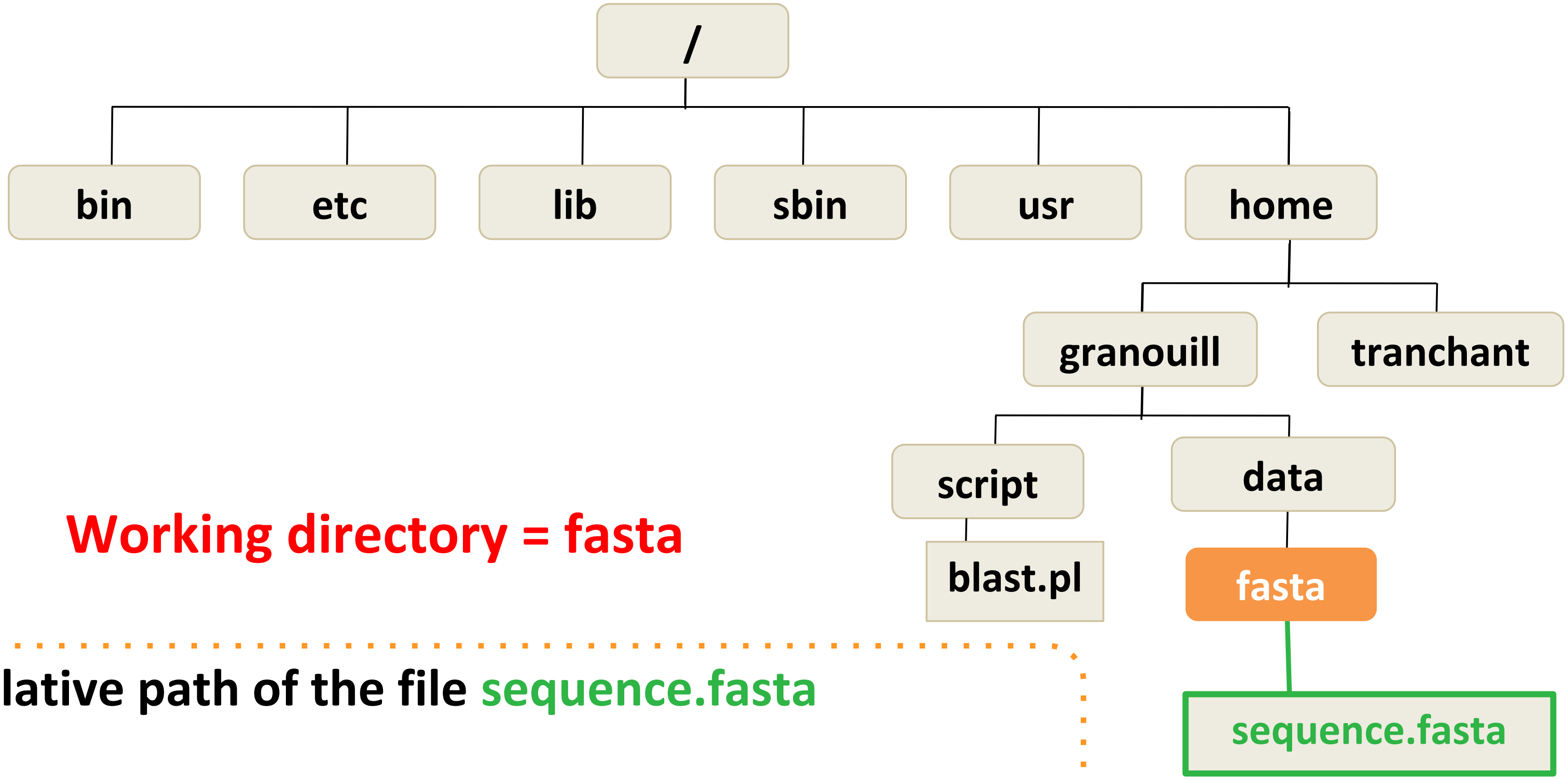
Working directory = data

Relative path of the file sequence.fasta

fasta/sequence.fasta

Relative Path

- Path related to the present working directory
- **Never starts with /**



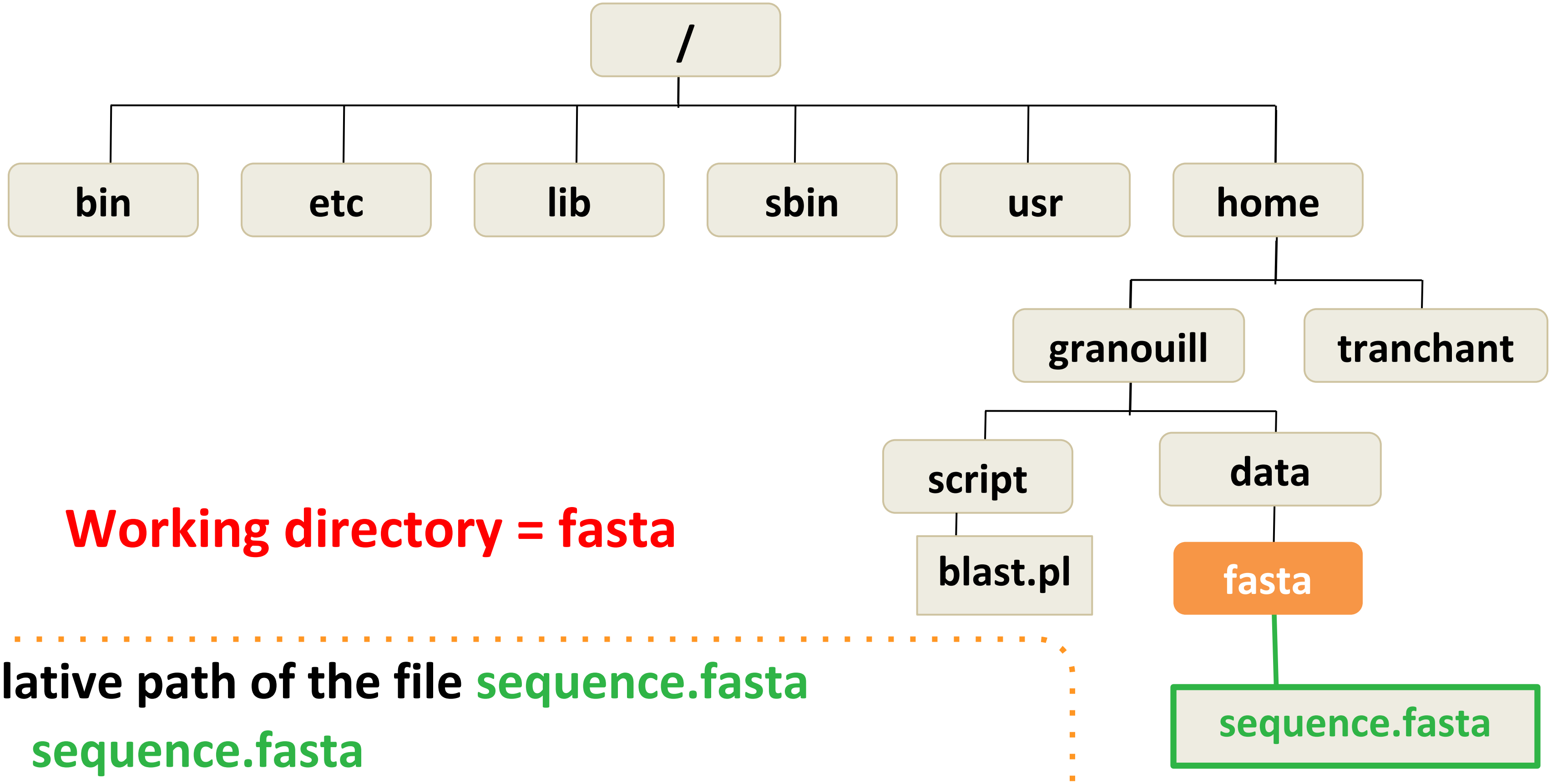
/
bin
etc
lib
sbin
usr
home
granouill
tranchant
script
data
blast.pl
fasta
sequence.fasta

Working directory = fasta

Relative path of the file sequence.fasta

Relative Path

- Path related to the present working directory
- **Never starts with /**



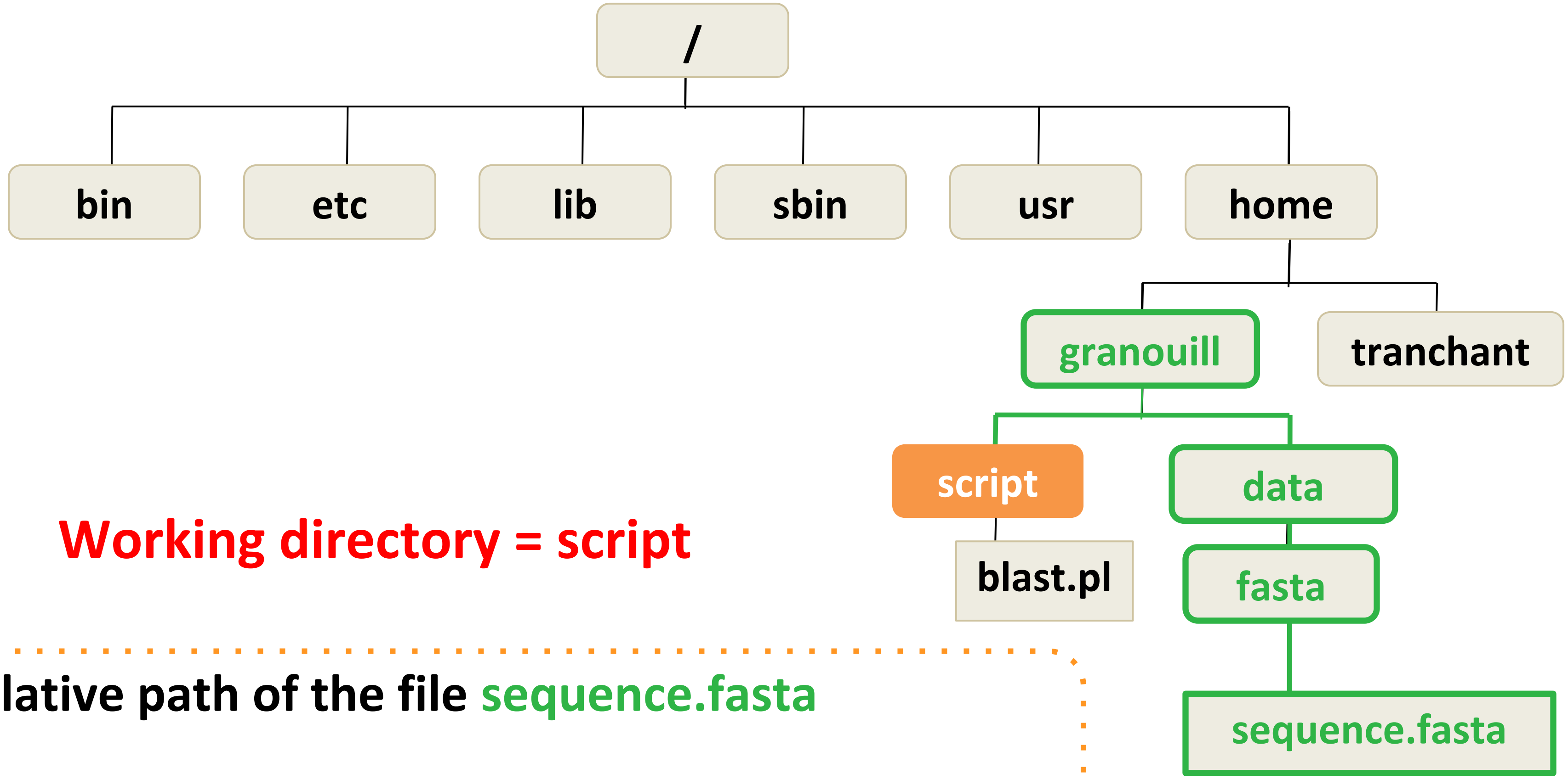
Working directory = fasta

Relative path of the file sequence.fasta

sequence.fasta

Relative Path

- Path related to the present working directory
- **Never starts with /**

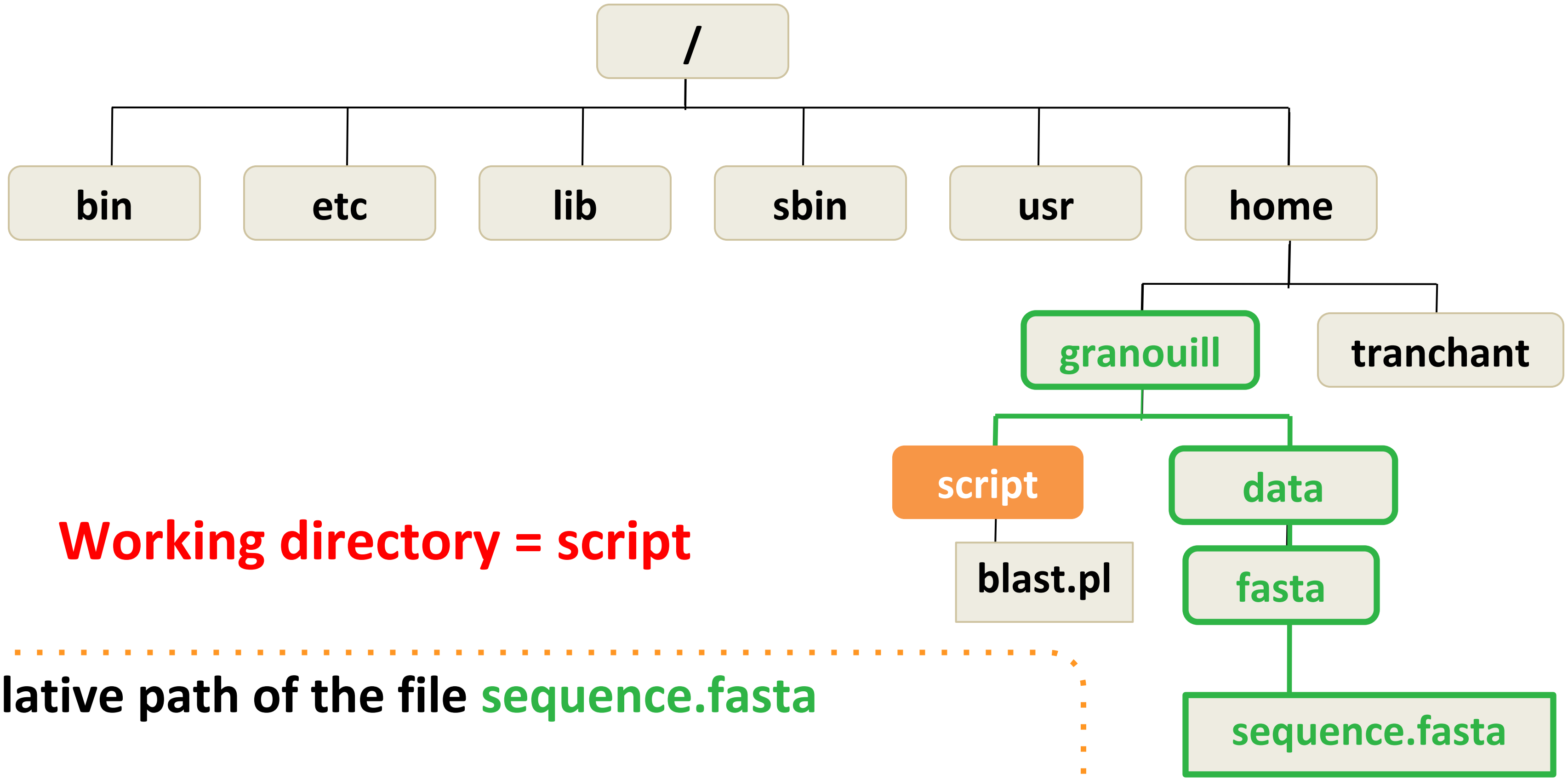


Working directory = script

Relative path of the file sequence.fasta

Relative Path

- Path related to the present working directory
- **Never starts with /**



/

bin etc lib sbin usr home

granouill tranchant

script data

blast.pl fasta

sequence.fasta

Working directory = script

Relative path of the file sequence.fasta

../data/fasta/sequence.fasta

Practice

ls

4

Go to [Practice 4](#) *on our github*

Useful Keyboard Shortcuts

Working with processes

<Ctrl> + C <Ctrl> + Z	Interrupt (kill) the current process running in the terminal

Tab completion

<Tab>	Automatically complete the file, directory or command you’re typing
<Tab><Tab>	if multiple files or directories, print the list of possibilities

Useful Keyboard Shortcuts

Working with the command history

Up/Down arrow Go to the previous / next command in the command history. Press it multiple times to walk back through the history

<Ctrl> + R Recall the last command matching the characters typed. Press this shortcut and start typing to search your bash history for a command.

Rules for naming file and directory names

- Linux is case sensitive

  Sequence.fasta ≠ SEQUENCE.fasta ≠ sequence.fasta

- Only ROMAN letters, numbers and _ -

- No space, accent or special symbol

  & ~ # ” ' { ([| ` \ ^ @)] } $ * % ! / ; , ?

- No need to use filename extension (.txt), just to improve readability of filenames.

Working with wildcard

Metacharacters : *, [ ]

What is a wildcard ?

It is a character that can be used as a substitute for any of a class of characters in a search

Can be used with numerous linux commands

How to use wildcards ?

Star wildcard
*****

can represent zero characters, all single characters or any string

KYVF-01.R1.fastq
KYVF-01.R2.fastq
KYVF-02.R1.fastq
KYVF-02.R2.fastq
KYVF.sam
KYVF.bam
ZO16.fastq
ZO16.bam

```
ls *fastq
```

How to use wildcards ?

Star wildcard
*****

can represent zero characters, all single characters or any string

```
KYVF-01.R1.fastq   KYVF-02.R1.fastq   KYVF.sam   ZO16.fastq
KYVF-01.R2.fastq   KYVF-02.R2.fastq   KYVF.bam   ZO16.bam
```

```
ls *fastq

KYVF-01.R1.fastq   KYVF-02.R1.fastq   ZO16.fastq
KYVF-01.R2.fastq   KYVF-02.R2.fastq
```

How to use wildcards ?

Star wildcard
*****

can represent zero characters, all single characters or any string

KYVF-01.R1.fastq
KYVF-01.R2.fastq
KYVF-02.R1.fastq
KYVF-02.R2.fastq
KYVF.sam
KYVF.bam
ZO16.fastq
ZO16.bam

```
ls KYVF*fastq
```

How to use wildcards ?

Star wildcard *****	can represent zero characters, all single characters or any string

KYVF-01.R1.fastq KYVF-02.R1.fastq KYVF.sam ZO16.fastq
KYVF-01.R2.fastq KYVF-02.R2.fastq KYVF.bam ZO16.bam

```
ls KYVF*fastq

KYVF-01.R1.fastq        KYVF-02.R1.fastq
KYVF-01.R2.fastq        KYVF-02.R2.fastq
```

How to use wildcards ?

Star wildcard ***** can represent zero characters, all single characters or any string

Square Brackets wildcards [ ]
can represent any of the characters enclosed in the brackets.

KYVF-01.R1.fastq
KYVF-01.R2.fastq
KYVF-02.R1.fastq
KYVF-02.R2.fastq
KYVF.sam
KYVF.bam
ZO16.fastq
ZO16.bam

```
ls *.[sb]am
```

How to use wildcards ?

Star wildcard can represent zero characters, all single
***** characters or any string

Square Brackets wildcards [ ]
can represent any of the characters enclosed in the brackets.

KYVF-01.R1.fastq
KYVF-01.R2.fastq
KYVF-02.R1.fastq
KYVF-02.R2.fastq
KYVF.sam
KYVF.bam
ZO16.fastq
ZO16.bam

```
ls *.[sb]am                    = ls *.[!f]*
```

KYVF.sam
KYVF.bam
ZO16.bam

Practice

ls, *

5

Go to [Practice 5](#) *on our github*

Commands for moving around the file system

Command cd

Command *cd*

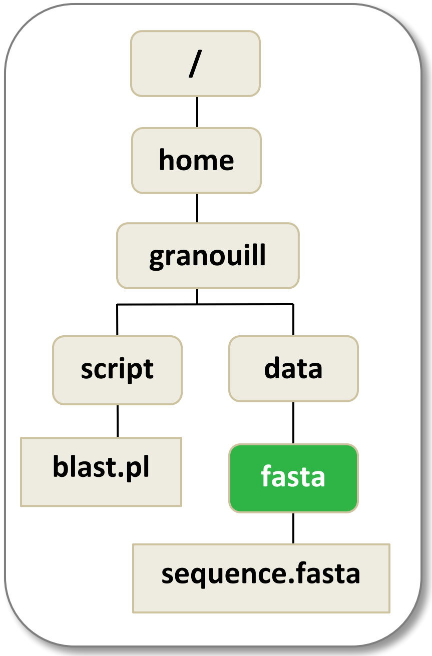
cd *Change **D**irectory*	*Move from the current directory into a new directory*

```
cd DIRECTORY_NAME(absolute or relative path)
```

Command *cd*

```
cd DIRECTORY_NAME (absolute or relative path)
```

Absolute path :

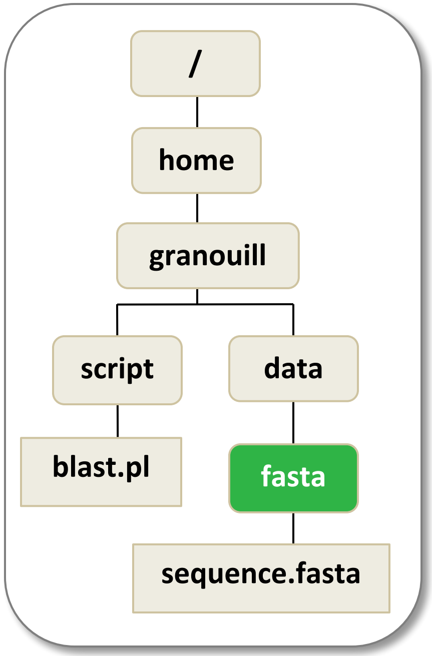

/
home
granouill
script
data
blast.pl
fasta
sequence.fasta

Command *cd*

```
cd DIRECTORY_NAME(absolute or relative path)
```

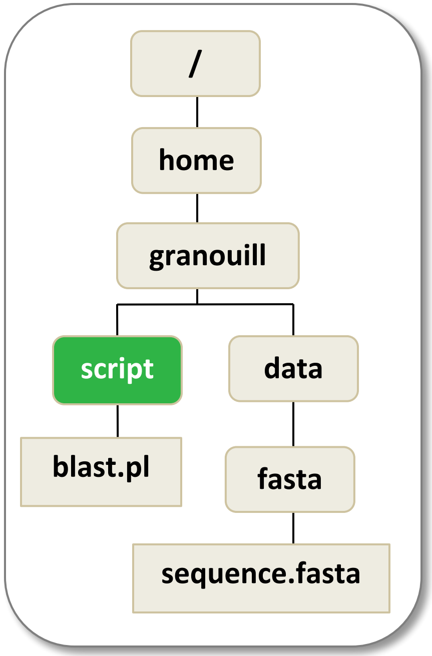
Absolute path :

```
cd /home/granouill/data/fasta
```



Command *cd*

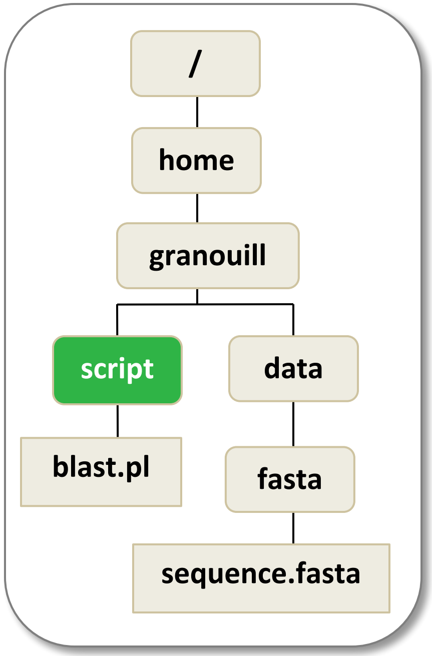
`cd DIRECTORY_NAME (absolute or relative path)`



Absolute path :

Command *cd*

`cd DIRECTORY_NAME(absolute or relative path)`

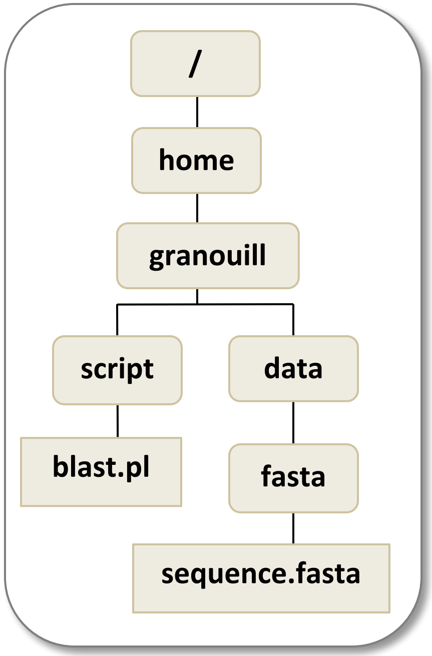


Absolute path :

```
cd /home/granouill/script/
```

Command *cd*

```
cd DIRECTORY_NAME (absolute or relative path)
```



Relative path :

Command	Go to
cd directory_name	directory_name
cd	*home directory*
cd ..	Parent directory
cd ../..	?

Go to home directory

One folder up

2 folders up

Some Commands

Linux file tree

pwd	Print the full path of the current directory
ls	Display the list of files in a directory
cd DIR_NAME	Change the working directory

mkdir rep_name	Create a new directory
rm nom_fichier	Remove a directory
cp file1 file2	Make a copy of file1 and calls it file2
cp FILE_NAME DIR_NAME	Copy the file FILE_NAME in the directory DIR_NAME, keeping the same name
cp FILE DIR/NEW_FILE	Mix of 2

Practice

pwd, cd, ls

6

Go to [Practice 6](#) *on our github*

Practice

mkdir, mv, ls, cp, rm

7

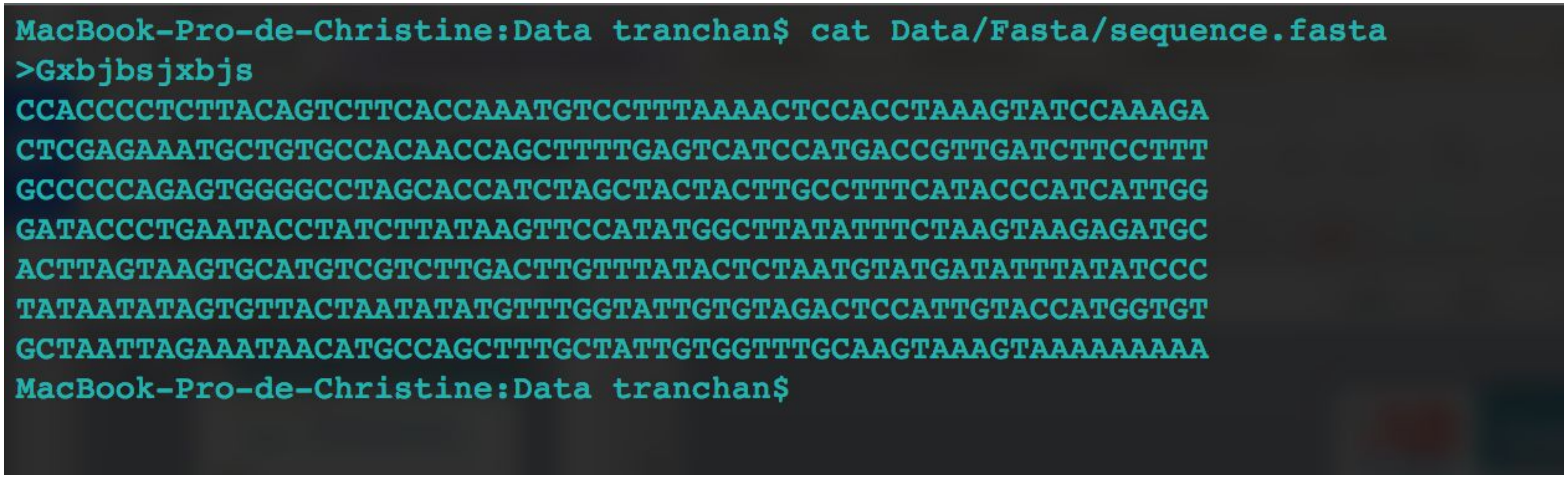
Go to Practice 7 *on our github*

Commands for displaying text files and searching word

Command cat

cat nom_fichier *Displays the content of a file on the screen*

(Don’t use it with big files!!)



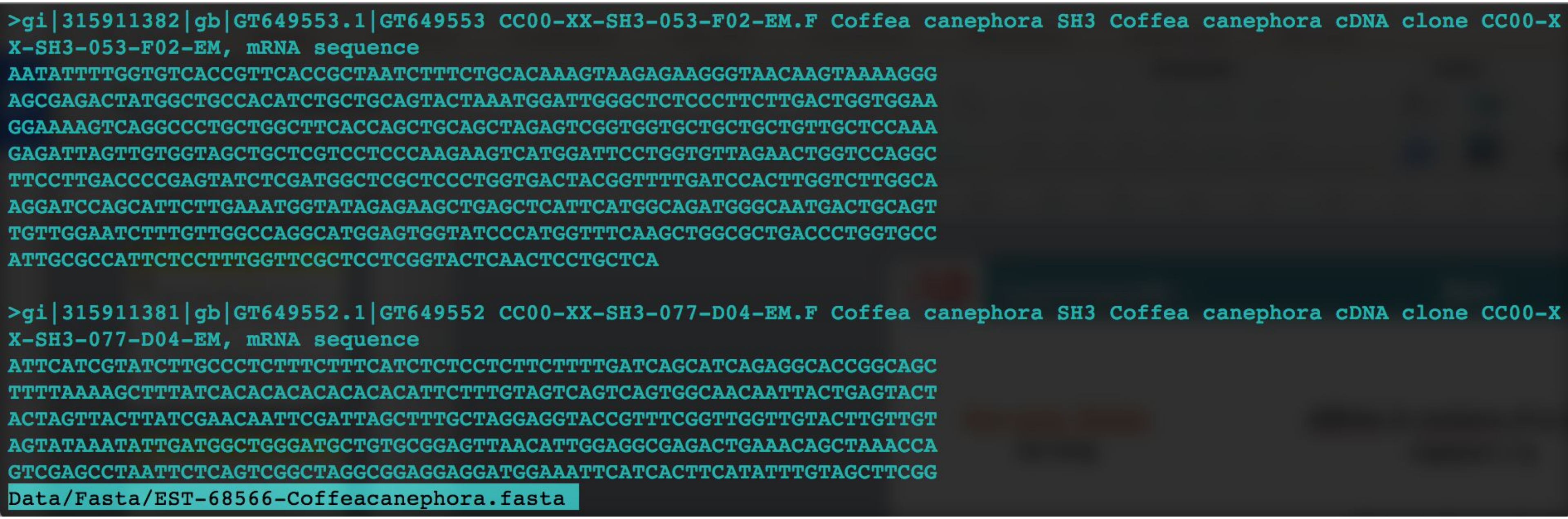
```
MacBook-Pro-de-Christine:Data tranchan$ cat Data/Fasta/sequence.fasta
>Gxbjbsjxbjs
CCACCCCTCTTACAGTCTTCACCAAATGTCCTTTAAAACTCCACCTAAAGTATCCAAAGA
CTCGAGAAATGCTGTGCCACAACCAGCTTTTGAGTCATCCATGACCGTTGATCTTCCTTT
GCCCCCAGAGTGGGGCCTAGCACCATCTAGCTACTACTTGCCTTTCATACCCATCATTGG
GATACCCTGAATACCTATCTTATAAGTTCCATATGGCTTATATTTCTAAGTAAGAGATGC
ACTTAGTAAGTGCATGTCGTCTTGACTTGTTTATACTCTAATGTATGATATTTATATCCC
TATAATATAGTGTTACTAATATATGTTTGGTATTGTGTAGACTCCATTGTACCATGGTGT
GCTAATTAGAAATAACATGCCAGCTTTGCTATTGTGGTTTGCAAGTAAAGTAAAAAAAAA
MacBook-Pro-de-Christine:Data tranchan$
```

Command less

less nom_fichier *writes the contents of a file a page at a time.*

[space-bar] to see another page
[q] to quit reading
[/] followed by the word to search

```
less Data/Fasta/EST-68566-Coffeacanephora.fasta
```



```
>gi|315911382|gb|GT649553.1|GT649553 CC00-XX-SH3-053-F02-EM.F Coffea canephora SH3 Coffea canephora cDNA clone CC00-X
X-SH3-053-F02-EM, mRNA sequence
AATATTTTGGTGTCACCGTTCACCGCTAATCTTTCTGCACAAAGTAAGAGAAGGGTAACAAGTAAAAGGG
AGCGAGACTATGGCTGCCACATCTGCTGCAGTACTAAATGGATTGGGCTCTCCCTTCTTGACTGGTGGAA
GGAAAAGTCAGGCCCTGCTGGCTTCACCAGCTGCAGCTAGAGTCGGTGGTGCTGCTGCTGTTGCTCCAAA
GAGATTAGTTGTGGTAGCTGCTCGTCCTCCCAAGAAGTCATGGATTCCTGGTGTTAGAACTGGTCCAGGC
TTCCTTGACCCCGAGTATCTCGATGGCTCGCTCCCTGGTGACTACGGTTTTGATCCACTTGGTCTTGGCA
AGGATCCAGCATTCTTGAAATGGTATAGAGAAGCTGAGCTCATTCATGGCAGATGGGCAATGACTGCAGT
TGTTGGAATCTTTGTTGGCCAGGCATGGAGTGGTATCCCATGGTTTCAAGCTGGCGCTGACCCTGGTGCC
ATTGCGCCATTCTCCTTTGGTTCGCTCCTCGGTACTCAACTCCTGCTCA

>gi|315911381|gb|GT649552.1|GT649552 CC00-XX-SH3-077-D04-EM.F Coffea canephora SH3 Coffea canephora cDNA clone CC00-X
X-SH3-077-D04-EM, mRNA sequence
ATTCATCGTATCTTGCCCTCTTTCTTTCATCTCTCCTCTTCTTTTGATCAGCATCAGAGGCACCGGCAGC
TTTTAAAAGCTTTATCACACACACACACACATTCTTTGTAGTCAGTCAGTGGCAACAATTACTGAGTACT
ACTAGTTACTTATCGAACAATTCGATTAGCTTTGCTAGGAGGTACCGTTTCGGTTGGTTGTACTTGTTGT
AGTATAAATATTGATGGCTGGGATGCTGTGCGGAGTTAACATTGGAGGCGAGACTGAAACAGCTAAACCA
GTCGAGCCTAATTCTCAGTCGGCTAGGCGGAGGAGGATGGAAATTCATCACTTCATATTTGTAGCTTCGG
Data/Fasta/EST-68566-Coffeacanephora.fasta
```

Other Useful Commands

head	writes the first ten lines of a file to the screen	`head -n 20 script.pl`
tail	writes the last ten lines of a file to the screen	`tail -n 5 script.pl`
wc	Count for word, lines, characters in a file	`wc script.pl` `wc -l  script.pl`

Practice

ls, head, tail, wc

8

Go to [Practice 8](#) *on our github*

Command grep

grep *searching a word, a pattern in a file*

```
grep [options] motif [file1, … ]
```

Command grep

grep *searching a word, a pattern in a file*

```
grep [options] motif [file1, … ]
```

To search for a phrase or pattern,

you must enclose it in single quotes

Command grep

grep *searching a word, a pattern in a file*

```
grep [options] motif [file1, … ]
```

Option	Description
-c	Count the number of lignes where *MOTIF* was found at least once
-l	Print only the name of files where *MOTIF* was found
-i	Case insensitive
-v	Print lines **WITHOUT** the *MOTIF*

Practice

9

Go to Practice 9 *on our github*

Command *cut*

cut Extracts columns/fields from a file

```
cut -d SEPARATOR -f fieldNumber fileName
```

```
cut –d “:” -f1,5 /etc/passwd
```

Picked un the FIRST and FIFTH columns of FILE, separated by :

Command sort

sort sort the content of a text file, line by line

`sort -k2 fileName`	Alphabetical sorting based on the second column
`sort -k2r  fileName`	Reverse Alphabetical sorting based on the 2nd col
`sort -k2g -k1r fileName`	Numeral sorting on the 2nd column then the 1st col
`sort -t: -k3g fileName`	-t option defines the field separator (by default :)

Practice

10

Go to Practice 10 *on our github*

INPUT/OUTPUT REDIRECTION

For saving the output of a command to a file

Saving the output of a command to a file

The output of commands can be written in a file

using the 2 operators :

>

>>

```
cut -d: -f1 /etc/passwd > userName.txt
```

Saving the output of a command to a file

```
cut -d: -f1 /etc/passwd > userName.txt
```

Redirection	Action
`Command > file1`	• redirect the output in a newly created file *file1* • **If *file1* exists, will overwrite it**
`Command >> file1`	• redirect the output at the end of *file1 (add)* • **If *file1* does not exist, will create it**

Practice

2

Go to [Practice 11](#) *on our github*

Connecting commands together

- The standard output of one command can be send as the standard input of another with the operator |
- To connect multiple commands together with what are called pipelines (without tempory files)
- Pipelines= *workflow*

```
cmd1 | cmd2 | cmd3
```

Connecting commands together

```
cut -d: -f1 /etc/passwd
Root
troot
iroot
ctroot
//
```

Connecting commands together

```
cut -d: -f1 /etc/passwd
Root
troot
iroot
ctroot
//

cut -d: -f1 /etc/passwd | sort
abate
adm
adroot
ais
#albar
alvaro-wis
anthony
apache
```

Connecting commands together

```
cut -d: -f1 /etc/passwd
Root
troot
iroot
ctroot
//

cut -d: -f1 /etc/passwd | sort
abate
adm
adroot
ais
#albar
alvaro-wis
anthony
apache

cut -d: -f1 /etc/passwd | sort | head
```

Connecting commands together

```
cut -d: -f1 /etc/passwd
Root
troot
iroot
ctroot
//

cut -d: -f1 /etc/passwd | sort
abate
adm
adroot
ais
#albar
alvaro-wis
anthony
apache

cut -d: -f1 /etc/passwd | sort > /etc/passwd.sort
```

Practice

3

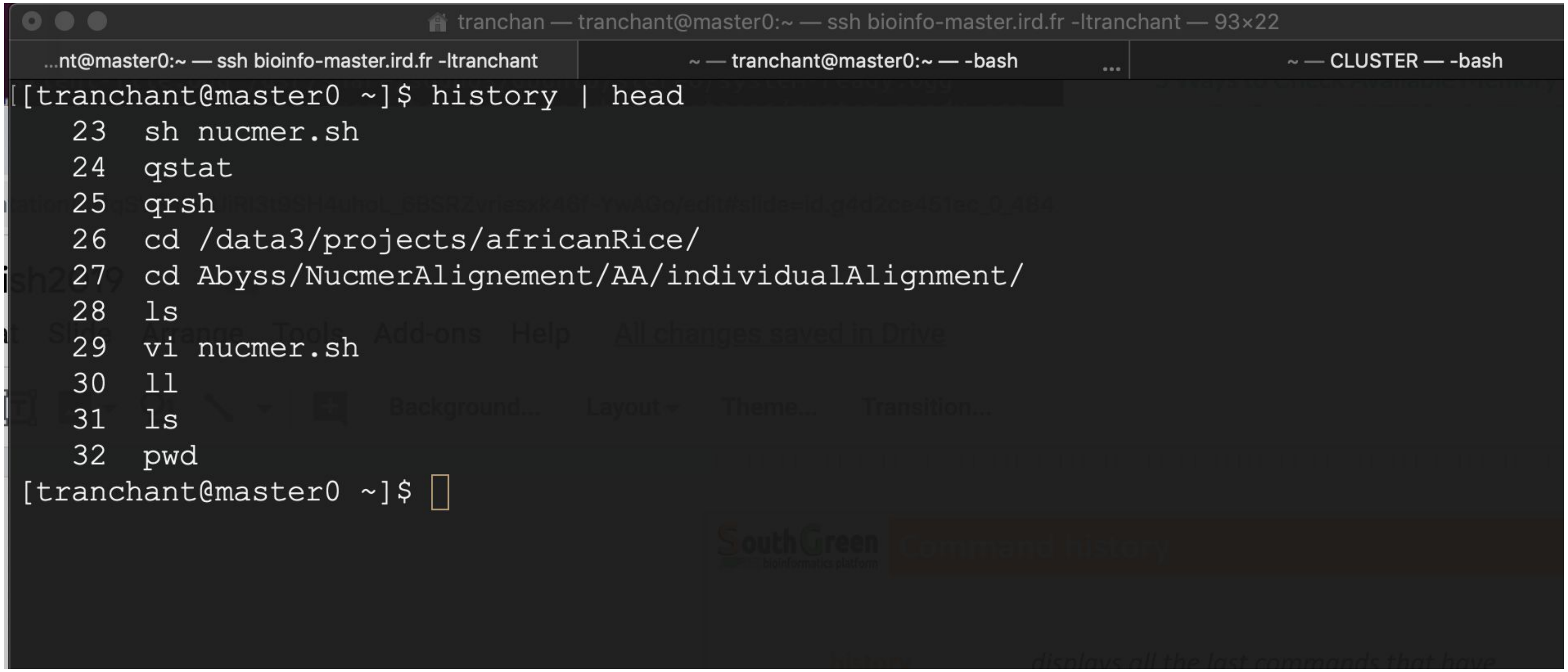
Go to [Practice 12](#) *on our github*

Autres commandes utiles

Command history

history *displays all the last commands that have been executed in all the previous sessions*

The entire history is saved into the file .bash_history



```
[tranchant@master0 ~]$ history | head
   23  sh nucmer.sh
   24  qstat
   25  qrsh
   26  cd /data3/projects/africanRice/
   27  cd Abyss/NucmerAlignement/AA/individualAlignment/
   28  ls
   29  vi nucmer.sh
   30  ll
   31  ls
   32  pwd
[tranchant@master0 ~]$ 
```

Command history

history *displays all the last commands that have been executed in all the previous sessions*

The entire history is saved into the file .bash_history

Filtering the History Output

```
history | grep "blastn"
```

displays only the commands including the search keyword “blastn”

```
history | tail
```

displays the commands recently used

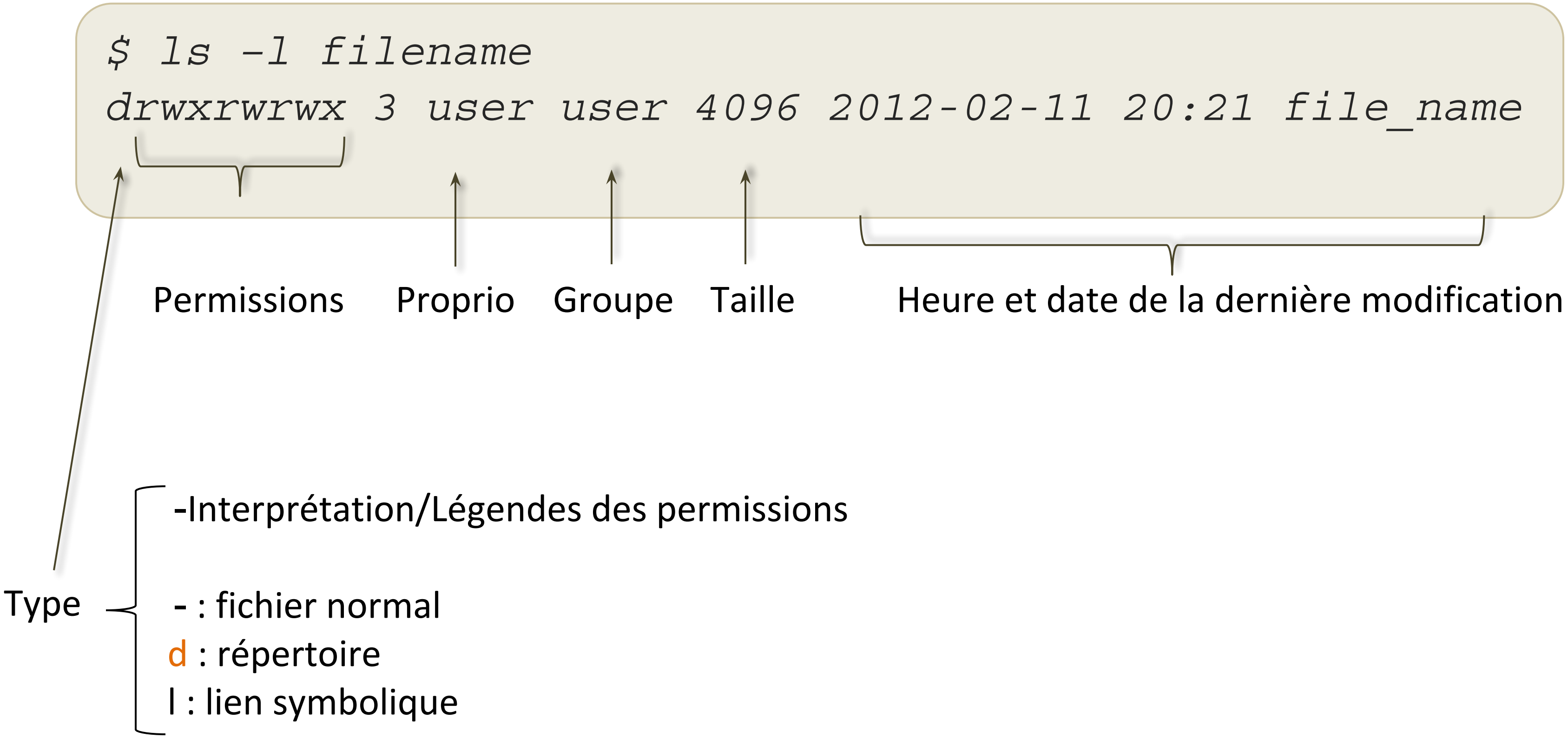
```
history | grep "blastn" | tail -n 5
```

```
history | head -n 5
```

displays the oldest commands

Attributs des fichiers / permission

Commande ls –l



```
$ ls -l filename
drwxrwrwx 3 user user 4096 2012-02-11 20:21 file_name
```

Permissions   Proprio   Groupe   Taille   Heure et date de la dernière modification

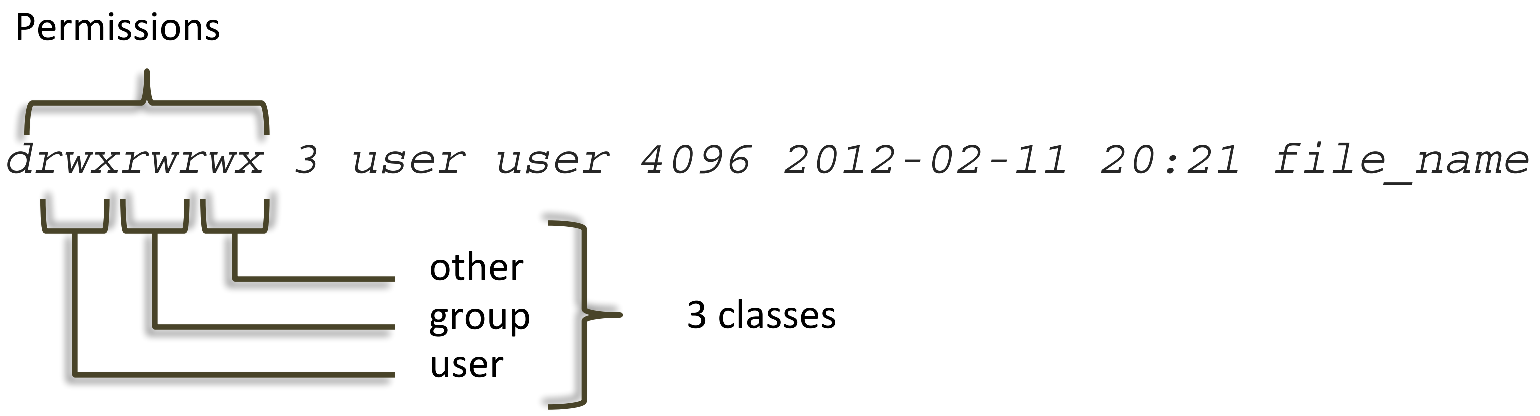
Type

-Interprétation/Légendes des permissions

- : fichier normal
d : répertoire
l : lien symbolique

Attributs des fichiers / permission

ls –l command



Permissions

```
drwxrwxrwx 3 user user 4096 2012-02-11 20:21 file_name
```

other

group

user

3 classes

3 types of permissions :

Permission	File	Directory
Read r	Ouverture et Lecture	Lister et copier les fichiers
Write w	Modifier et supprimer	Manipuler le contenu : copier, créer, modifier, écraser
Execution x	Executer le fichier	Accès seulement au fichier pour l’exécuter

Attributs des fichiers / permission

commande pour la gestion des permissions : chmod

```
chmod <perm> file_name
```

Chaque permission = 1 valeue

R	**4**
W	2
X	1
none	0

Exemple

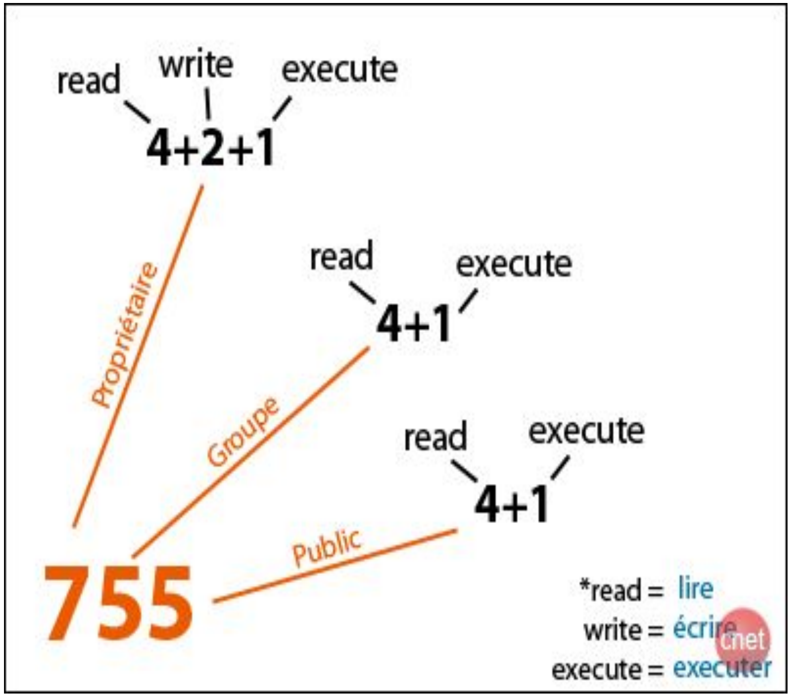
```
chmod 740 script.sh            # Owner=rwx Group=r-- Other=---
chmod 755 script.sh            # Owner=rwx Group=r-x Other=r-x
```

Visualiser et changer des droits

chmod, ls

Donner le nom du propriétaire, du groupe et les droits des fichiers contenus dans le répertoire “~/Data/454-projet1/raw”

Changer les droits du fichier
Scripts/blast.pl pour qu’il soit :
en lecture/écriture pour le groupe,
lecture/écriture/exécution
pour le propriétaire
et lecture au public



Manipuler des fichiers compressés

Compresser des fichiers **tar,gzip**

```
tar -zcvf tarfile.tar.gz dirToCompress
gzip fileToCompress
```

Décompresser une archive **gunzip, tar**

```
gunzip file.gzip
tar -xvf file.tar
tar -zxvf file.tar.gz
gzip -d file.gz
```

Afficher le contenu d’une archive **zcat**

```
zcat data.txt.gz
```

Rechercher une expression/motif dans une archive **zgrep**

```
zgrep ‘NM_000020’ data.gz
```

Renommer des fichiers

rename

Exemple	Description
`rename 's/.txt/.fasta/' *`	rename the extension of all files
`rename 'y/a-z/A-Z/' *`	rename files in uppercase

Espace disque et taille des fichiers

Taille du disque **df** disk free

`df`	taille espace occupé en octet
`df -h`	human-readable

Taille des répertoires **du** disk usage

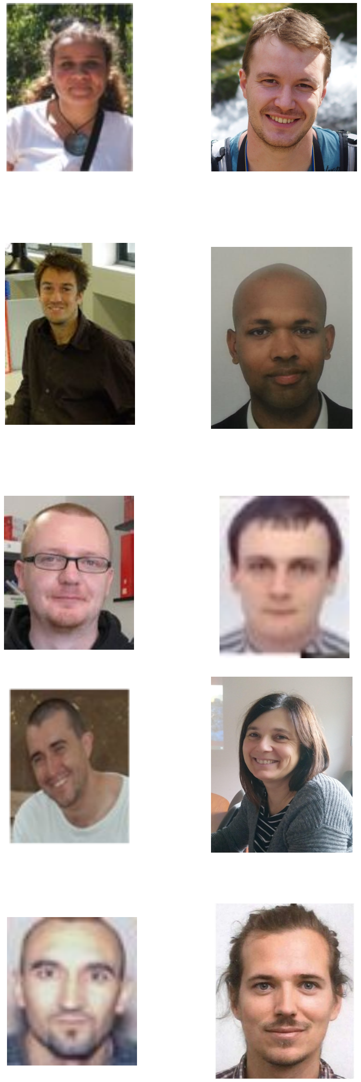
```
du
du -h
du -h *
```

Rechercher un fichier **find**

```
find -name “transcritsAssembly.fasta”
```

Formateurs itrop / South Green

- **Christine Tranchant-Dubreuil**
- Sebastien Ravel
- Alexis Dereeper
- **Ndomassi Tando**
- François Sabot
- Gautier Sarah
- Bruno Granouillac
- **Valérie Noël**
- **Guilhem Sempere**
- **Florentin Constancias**



Merci pour votre attention !