

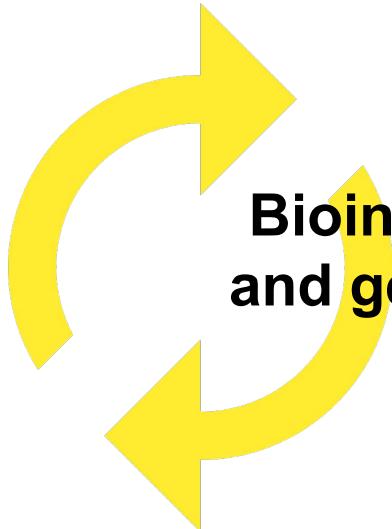


# Session de formation 2023



Alliance





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

comparative genomics  
phylogeny  
GWAS  
population genetics  
polyploidy

genome assembly  
transcriptome assembly  
metagenomics

SNP detection  
structural variation  
differential expression



Rice



Banana



Palm



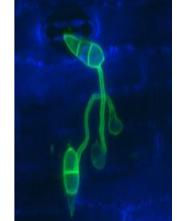
Sorghum



Coffee



Cassava



Magnaporthe



Charte



Larmande Pierre  
Sabot François  
Tando Ndomassi  
**Tranchant-Dubreuil**  
Christine



Comte Aurore  
Dereeper Alexis



**Orjuela-Bouniol Julie**



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



**Rouard Mathieu**  
Guignon Valentin  
Catherine Breton



**Mahé Frédéric**  
**Ravel Sébastien**



Sempere Guilhem



# South Green bioinformatics platform

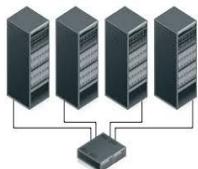
## Workflow manager

TOGGLE  
Toolbox for generic NGS analyses



Galaxy

## HPC and trainings....



## Genome Hubs & Information System



Gigwa

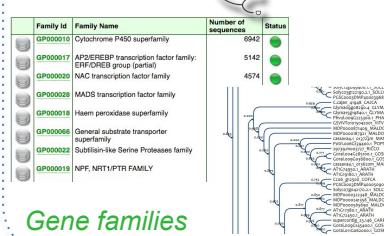
A screenshot of the Gigwa software interface, showing a grid of data entries with columns for ID, Name, and other parameters.

SNPs and Indels

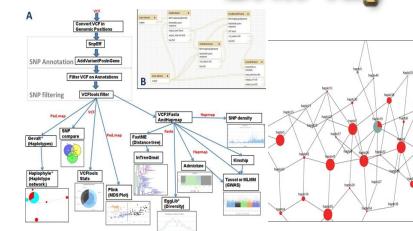
GreenPhyl

Family Id	Family Name	Number of Sequences	Status
GP000010	Cytochrome P450 superfamily	6942	green
GP000017	AP2/EREBP transcription factor family	5142	green
GP000020	EN-FOREB group (petal)	4574	green
GP000028	NAC transcription factor family		
GP000018	Heme peroxidase superfamily		
GP000006	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

# I-Trop

Plant & Health Bioinformatics Platform



<https://bioinfo.ird.fr/>



AURORE  
COMTE



JACQUES  
DAINAT



ALEXIS  
DEREOPER



BRUNO  
GRANOUILLAG  
ORJUELA-



JULIE  
NDOMASSI  
TANDO

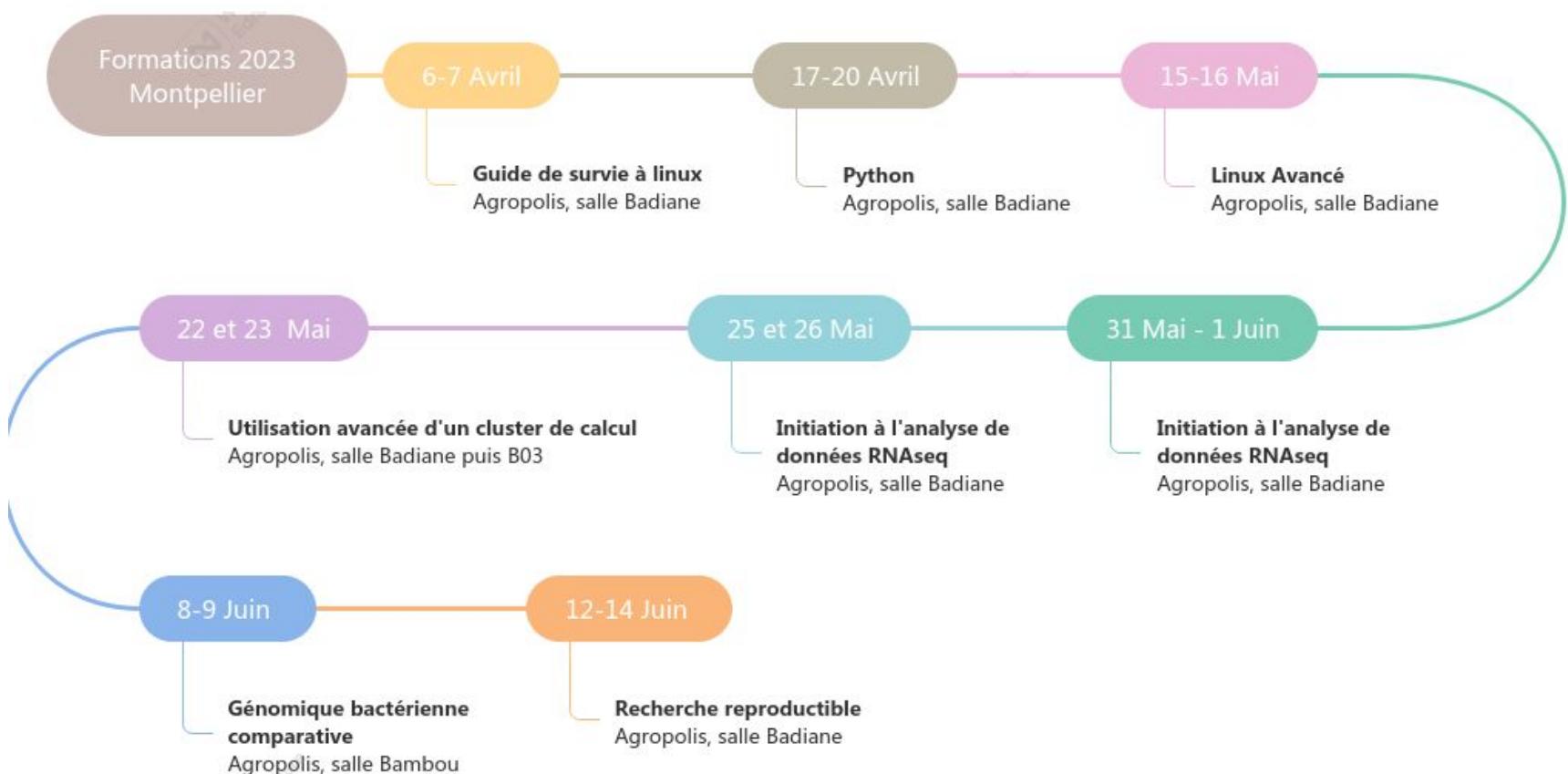


CHRISTINE  
TRANCHANT

[bioinfo@ird.fr](mailto:bioinfo@ird.fr)



@ItropBioinfo





# Modules de formation 2023

- Toutes nos formations :  
<https://southgreenplatform.github.io/trainings/>
- Topo & TP : [Linux For Dummies](#)
- Environnement de travail : [Logiciels à installer](#)





# Guide de survie à Linux



[www.southgreen.fr](http://www.southgreen.fr)

<https://southgreenplatform.github.io/trainings>



## The objectif!

Lancez vos analyses bioinformatiques sous Linux !



## Applications

Connaître les principales commandes sous Linux

- Se déplacer dans l'arborescence de fichier : *pwd, ls, cd, mkdir, ...*
- Se connecter à un serveur, transférer les données : *ssh, wget*
- Manipuler des fichiers : *head, tail, sort, cut, wc, grep*
- Lancer des logiciels bioinformatiques en ligne de commande

# Introduction



- **Système d'exploitation** réputé pour :
  - sa sécurité
  - ses mises à jour fréquentes
  - son prix et ses programmes gratuits
- Crée en 1991 par *Linus Torvalds*



- Basé sur l'OS propriétaire Unix (1969)
- Linux **gratuit** et **libre**
  - on peut avoir le code source, la “recette de fabrication”
  - on peut copier, modifier, redistribuer

- **Système robuste et multi-plateforme**



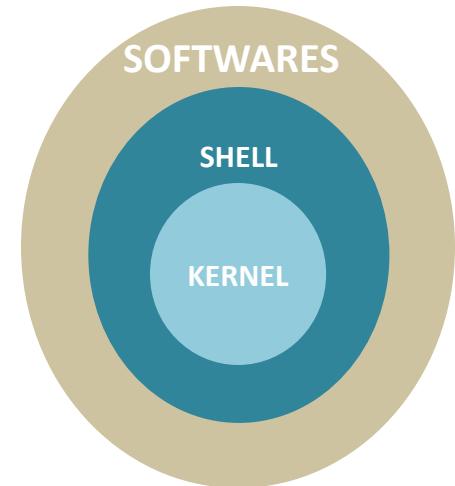
- **Système multi-utilisateurs**

Plusieurs utilisateurs peuvent travailler en même temps

- **Système multi-tâches (processus/programmes)**

Chaque utilisateur peut lancer plusieurs programmes en même temps

**Distribution** : Noyau/Coeur + logiciels/programmes



- 2 façons d'utiliser linux :

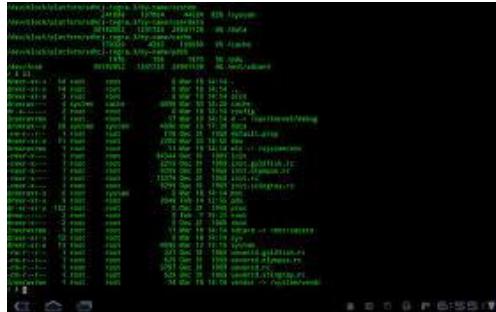
en *mode graphique*



- 2 façons d'utiliser linux :

en *mode graphique*

en *mode console*



# Pourquoi utiliser Linux ?



- Nombreux programmes rapides & puissants
- Facile de lier des commandes/programmes entre eux (workflow)
- Nombreux outils bioinformatique disponibles
- 90% des serveurs fonctionnent sous Linux

# Pourquoi utiliser Linux ?



- Nombreux programmes rapides & puissants
  - Facile de lier des commandes/programmes entre eux (workflow)
  - Nombreux outils bioinformatique disponibles
  - 90% des serveurs fonctionnent sous Linux

# Pas d'interfaces graphiques

## Convivialité de la ligne de commande ?





Nécessité de la pratique et de l'expérience

↔ **Investissement non négligeable pour de bons résultats rapidement**

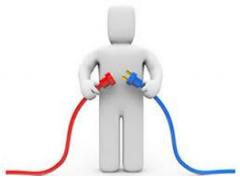


# Environnement de travail

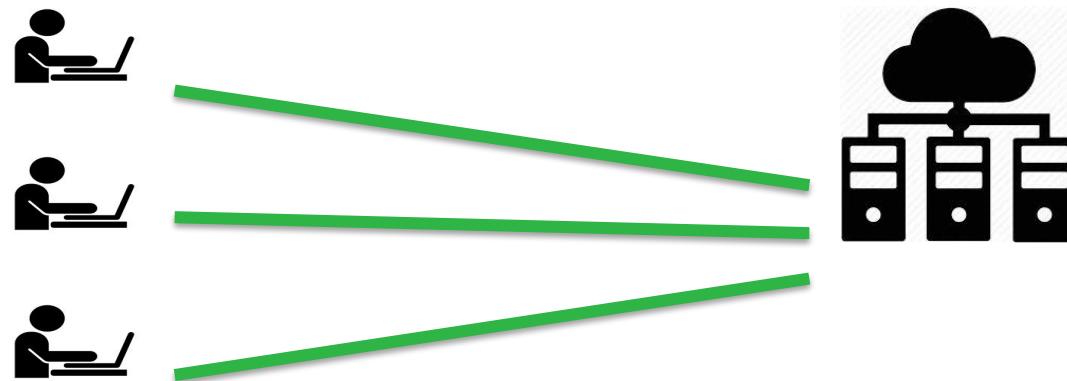
*Comment copier un fichier de son PC sur  
le serveur ?*



# *Copier un fichier de son PC sur le serveur ?*



- En se connectant sur un serveur linux distant de son ordinateur via le **protocole sftp**



HPC South Green

- itrop (IRD)

[bioinfo-nas.ird.fr](http://bioinfo-nas.ird.fr)



# Filezilla

FileZilla

Hôte : Identifiant : Mot de passe : Port : Connexion rapide

Site local : /U: Gestionnaire de Sites

Sélectionnez une entrée :

- ▼ Mes Sites
  - Nouveau site
  - bioinfo-inter
  - nas**

1 Nouveau Site    Nouveau Dossier  
Nouveau Favori    Renommer  
Supprimer    Dupliquer

2

3 Hôte : bioinfo-nas.ird.fr  
Protocole : SFTP - SSH File Trans...

Type d'authentification : Normale  
Identifiant : tranchant  
Mot de passe : \*\*\*\*\*

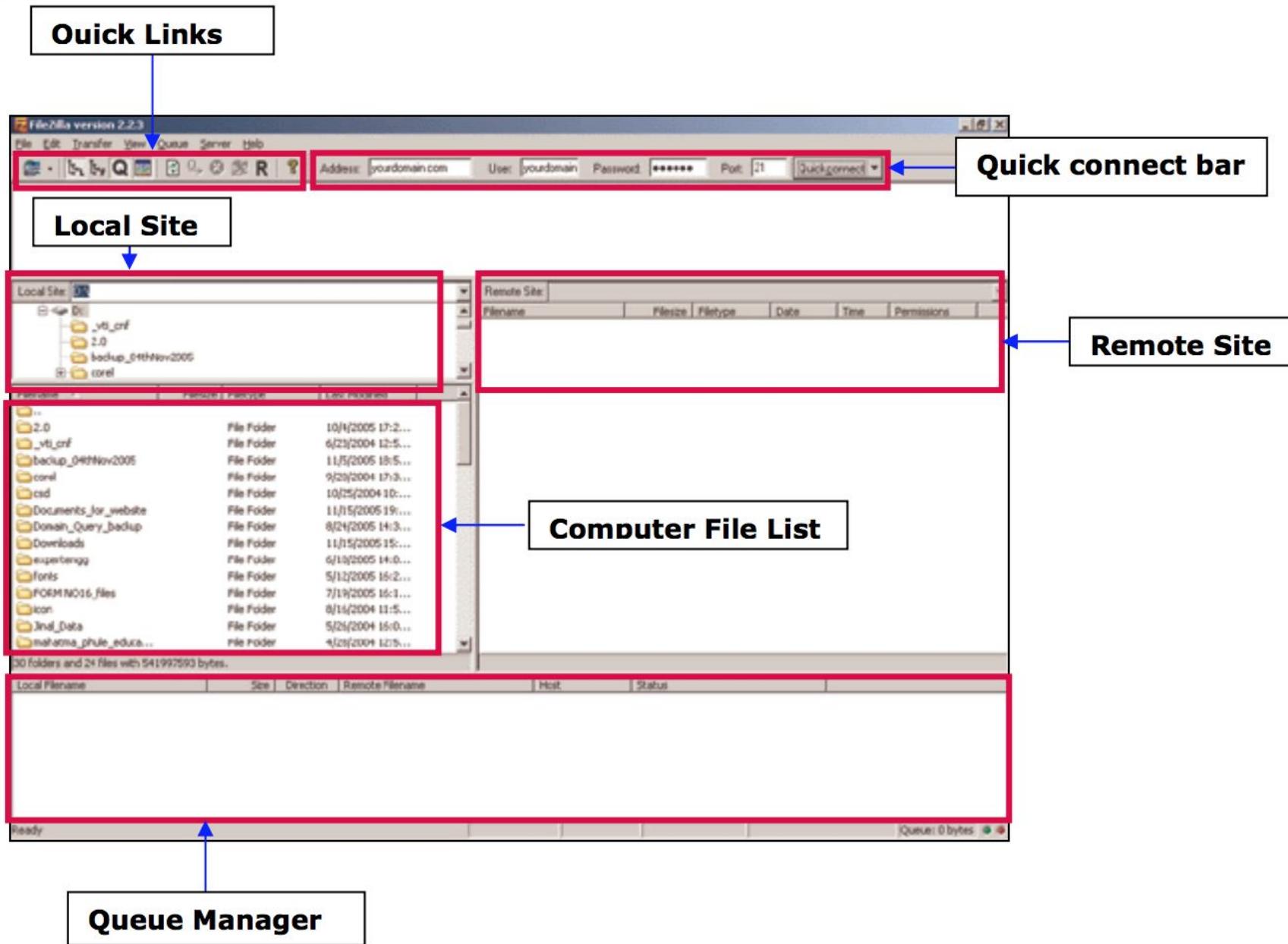
Couleur de fond : Aucune  
Commentaires :

4 Connexion

7396 jpeg-fichier 21.02.2018 15:41...



# Filezilla





# Practice

filezilla, sftp

1

Go to [Practice 1](#) on our github



# Environnement de travail

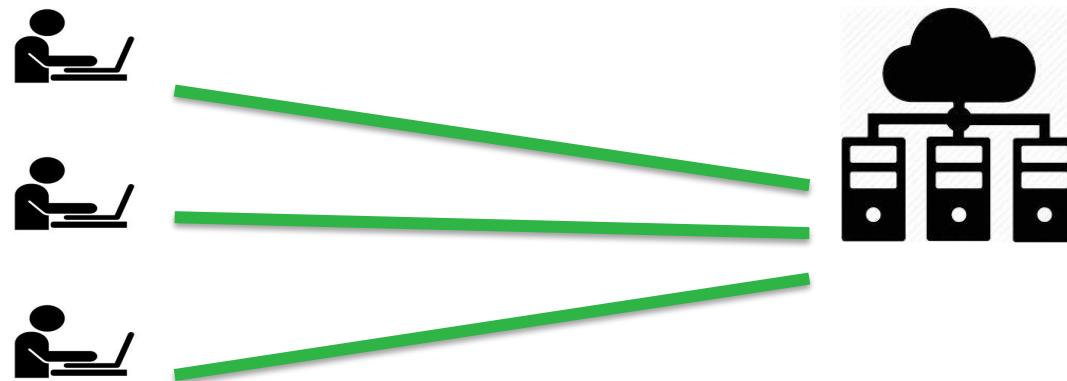
*Comment travailler sur le serveur ?*



# Comment travailler sur le serveur ?



- En se connectant sur un serveur linux distant de son ordinateur via le **protocole ssh**



HPC South Green

- itrop (IRD)

[bioinfo-inter.ird.fr](http://bioinfo-inter.ird.fr)



PuTTY



# Comment travailler sur le serveur ?

Sessions

Terminal Sessions View X server Tools Games Set

Session Servers Tools

Quick connect...

Saved sessions

SSH Telnet Rsh Xdmcp RDP VNC FTP SFTP Serial File Shell Browser Mosh

Basic SSH settings

Remote host \* unix.andrew.cmu.edu Specify username jezimmer Port 22

Advanced SSH settings Terminal settings Bookmark settings

Session name Andrew Unix Session Icon

Start session in Normal tab

Create a desktop shortcut to this session

OK Cancel

A yellow star icon is located on the right side of the session settings window.

The image shows a Linux desktop interface with a 'Session' menu open. The 'Session' tab is selected. A 'Session settings' dialog box is displayed over the menu. The dialog has tabs for 'Basic SSH settings' (selected), 'Advanced SSH settings', 'Terminal settings', and 'Bookmark settings'. In the 'Basic SSH settings' tab, fields include 'Remote host' (unix.andrew.cmu.edu), 'Specify username' (jezimmer), and 'Port' (22). Below these are fields for 'Session name' (Andrew Unix) and 'Session Icon'. A 'Create a desktop shortcut to this session' button is also present. At the bottom are 'OK' and 'Cancel' buttons. A yellow star icon is positioned to the right of the dialog. The desktop background features a penguin icon with question marks above its head.



# Comment travailler sur le serveur ?

```
● ● ● tranchan — CLUSTER — ssh bioinfo-inter.ird.fr -ltranchant — 130
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

MobaXterm,  
terminal, ssh

2

Go to [Practice 2](#) on our github

# Premiers Pas sur Linux



**Commandes relatives à l'arborescence de fichiers**

Toujours présent sur le terminal, juste avant de taper la commande

**Prompt**

```
[tranchant@node6 data]$ █
```

Toujours présent sur le terminal, juste avant de taper la commande

## Prompt

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

Nom  
utilisateur

Nom  
serveur

Répertoire  
courant

# Syntaxe d'une commande

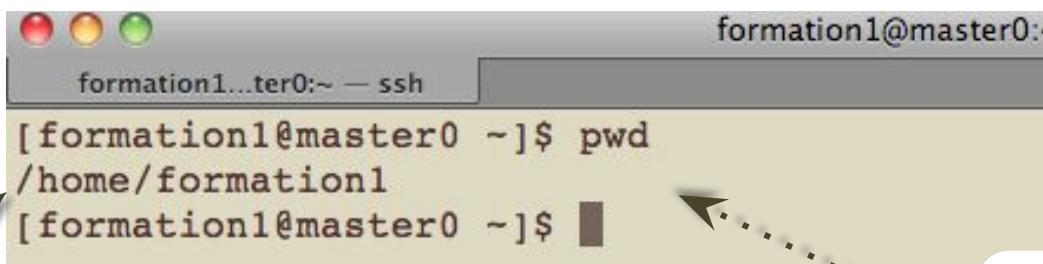
*commande [ -options ] [ arguments ]*

# 1ère commande

*pwd*

**pwd**  
Present Work Directory

*affiche nom du répertoire courant (et son chemin complet)*



```
formation1@master0:~$ formation1@master0:~ - ssh
[formation1@master0 ~]$ pwd
/home/formation1
[formation1@master0 ~]$
```

The screenshot shows a terminal window titled "formation1...ter0:~ - ssh". The command "pwd" is entered and its output "/home/formation1" is displayed. The terminal window has a standard OS X look with red, yellow, and green close buttons.

A callout bubble on the left points to the terminal window with the text "Affiche le nom du répertoire courant". A callout bubble on the right points to the command line with the text "Commande sans options et sans argument".

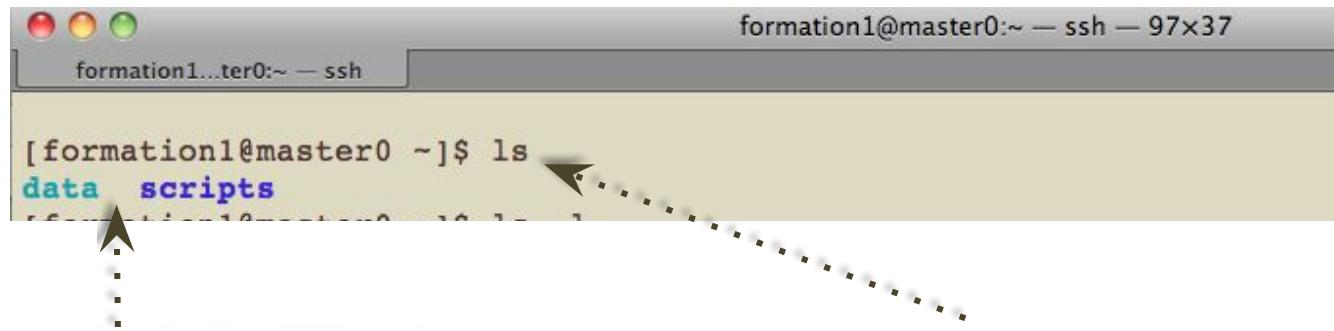
Affiche le nom du répertoire courant

Commande sans options et sans argument

**ls**

*list*

*liste fichiers & répertoires présents dans le répertoire courant*



A screenshot of a terminal window titled "formation1@master0:~ — ssh — 97x37". The window shows the command [formation1@master0 ~]\$ ls being run, and the output "data scripts" is displayed. A dotted arrow points from the word "scripts" in the terminal output towards the text "Liste des fichiers du répertoire courant (par défaut)".

Liste des fichiers du répertoire courant (par défaut)

Commande sans options et sans argument

**ls -l**  
*list long*

*liste les fichiers avec des informations pour chaque fichier/répertoire*

Commande avec l'option l et le nom d'un répertoire en 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
```

liste détaillée des fichiers

## Comment obtenir de l'aide sur une commande?

- avec l'option *--help* ou *-h*

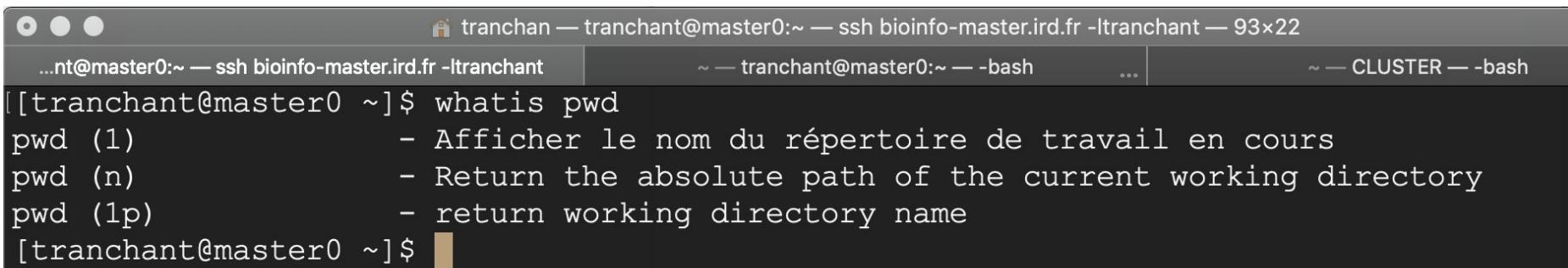
*ls --help*      *blastn -h*

- avec la commande *man*

*man ls*

- avec la commande *whatis*

*whatis ls*



```
tranchant — tranchant@master0:~ — ssh bioinfo-master.ird.fr -ltranchant — 93x22
...nt@master0:~ — ssh bioinfo-master.ird.fr -ltranchant           ~ — tranchant@master0:~ — -bash          ... — CLUSTER — -bash
[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 ~]$
```

## Arborescence linux

**pwd**

Affiche le chemin absolu

**ls**

Liste tous les fichiers/répertoires

**ls -l**

Affiche toutes les informations sur les fichiers



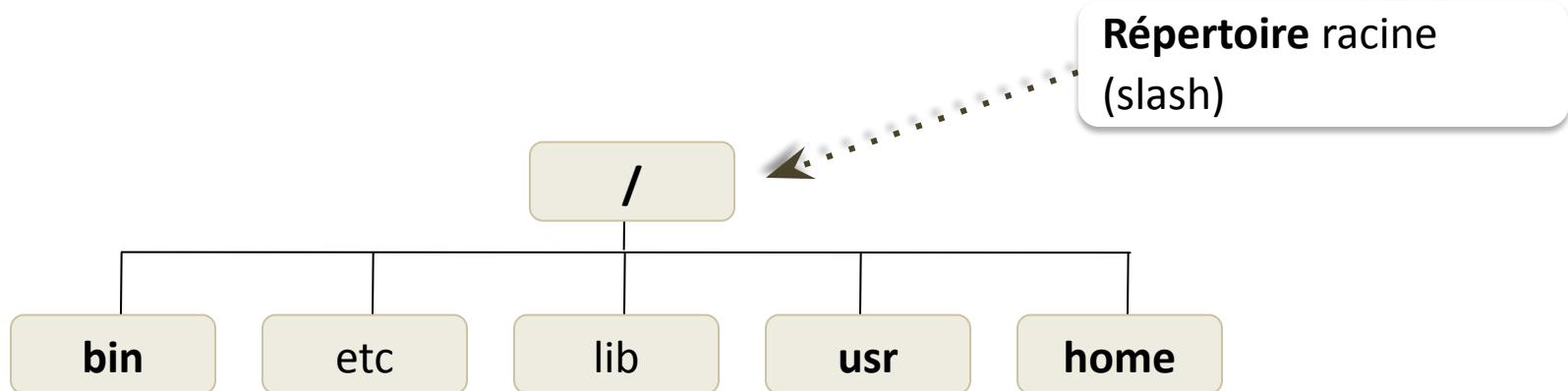
# Practice

prompt, pwd

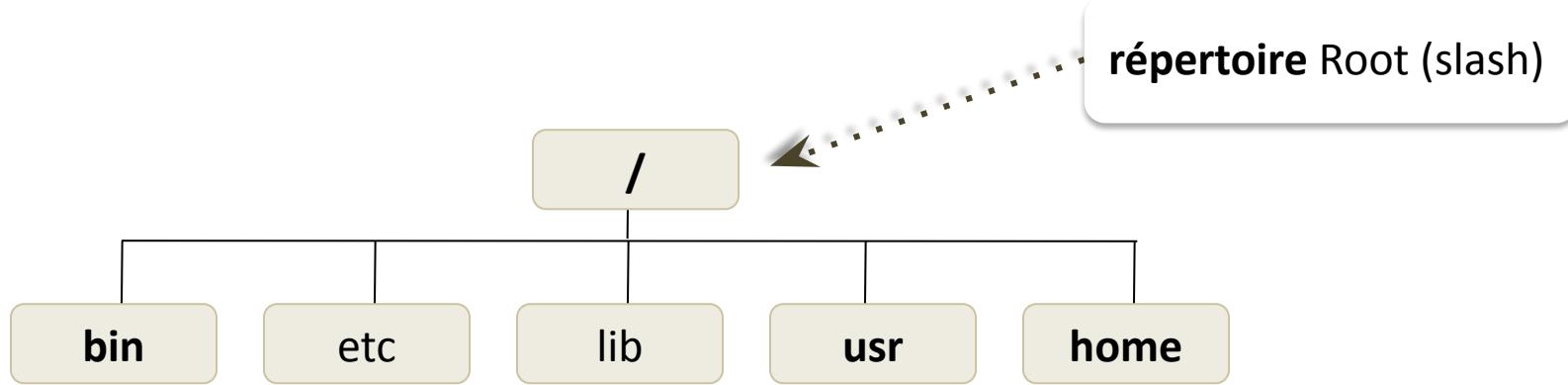
3

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

- Arborescence = système de fichiers
- Répertoire racine appelé “/”



## Principaux répertoires



<b>/bin</b>	principales commandes système
<b>/usr, /opt</b>	Applications et librairies ex : <b>/usr/local</b>
<b>/usr/bin</b>	Autres commandes
<b>/home</b>	répertoire des utilisateurs ( <b>1 par utilisateur, name=login</b> )

**Chemin (path)** : chemin d'accès à un fichier/répertoire

**Chemin (path)** : chemin d'accès à un fichier/répertoire

absolu

- chemin complet du fichier en partant du répertoire racine /

**Chemin** (path) : chemin d'accès à un fichier/répertoire

absolu

- chemin complet du fichier en partant du répertoire racine /
- *commence toujours par /*
- Toujours correct peu importe où on l'on travaille

**Chemin** (path) : chemin d'accès à un fichier/répertoire

**absolu**

- chemin complet du fichier en partant du répertoire racine /
- commence toujours par /
- Toujours correct peut importe où on l'on travaille

**relatif**

- chemin défini par rapport à où l'on est dans l'arborescence

**Chemin** (path) : chemin d'accès à un fichier/répertoire

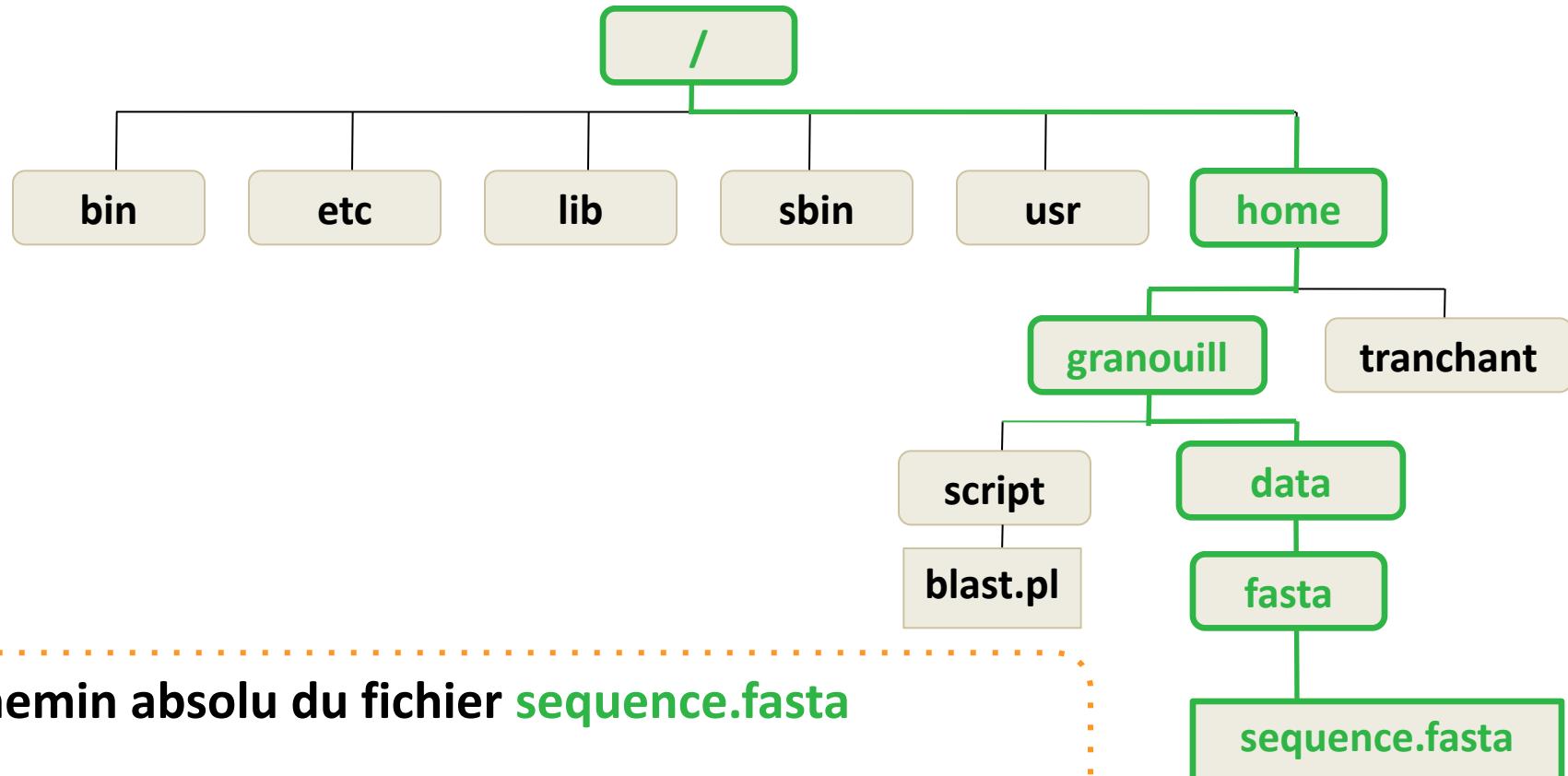
**absolu**

- chemin complet du fichier en partant du répertoire racine /
- commence toujours par /
- Toujours correct peut importe où on l'on travaille

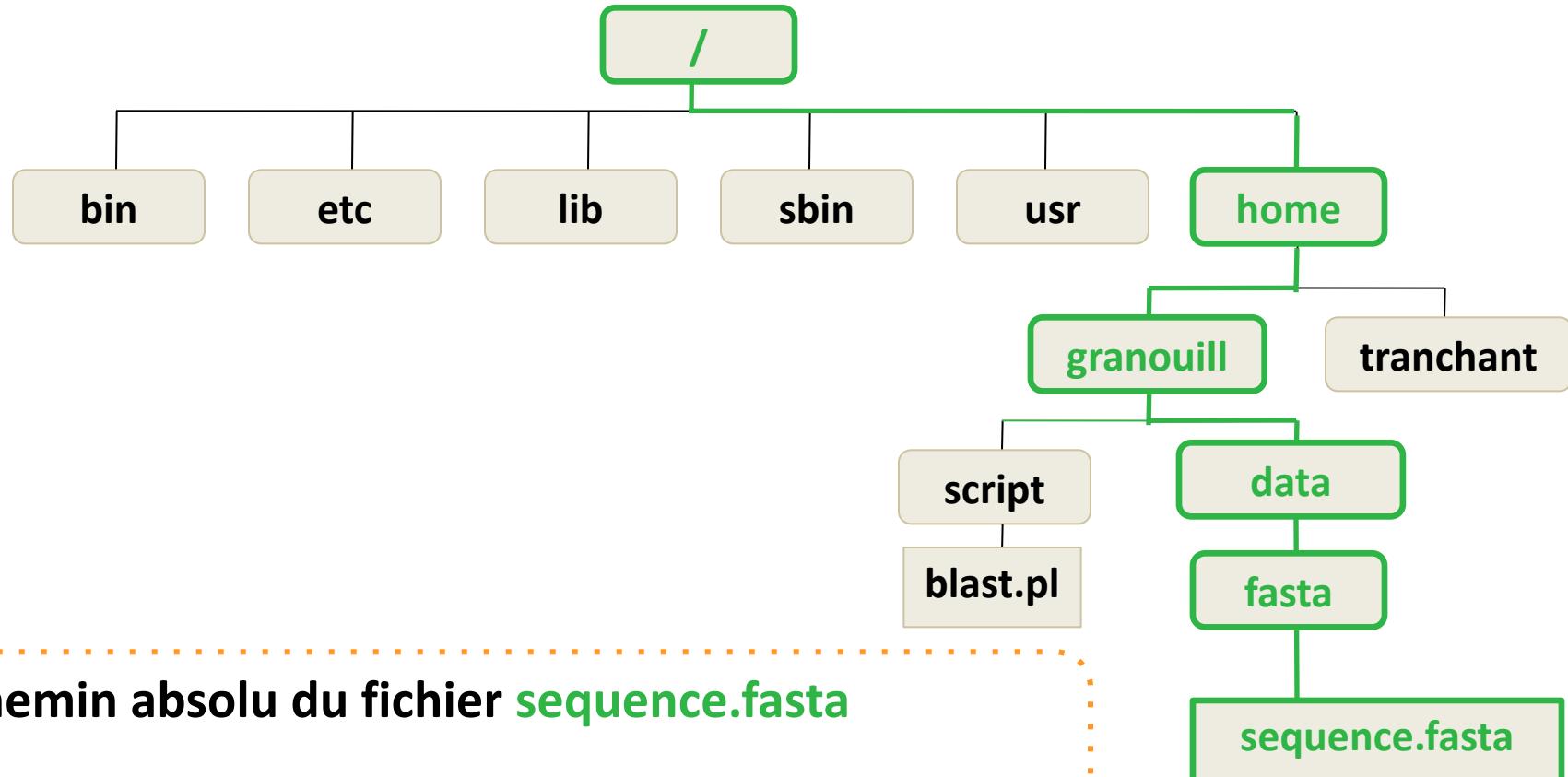
**relatif**

- chemin défini par rapport à où l'on est dans l'arborescence
- ***Ne commence jamais par /***
- ***Change selon où l'on travaille***

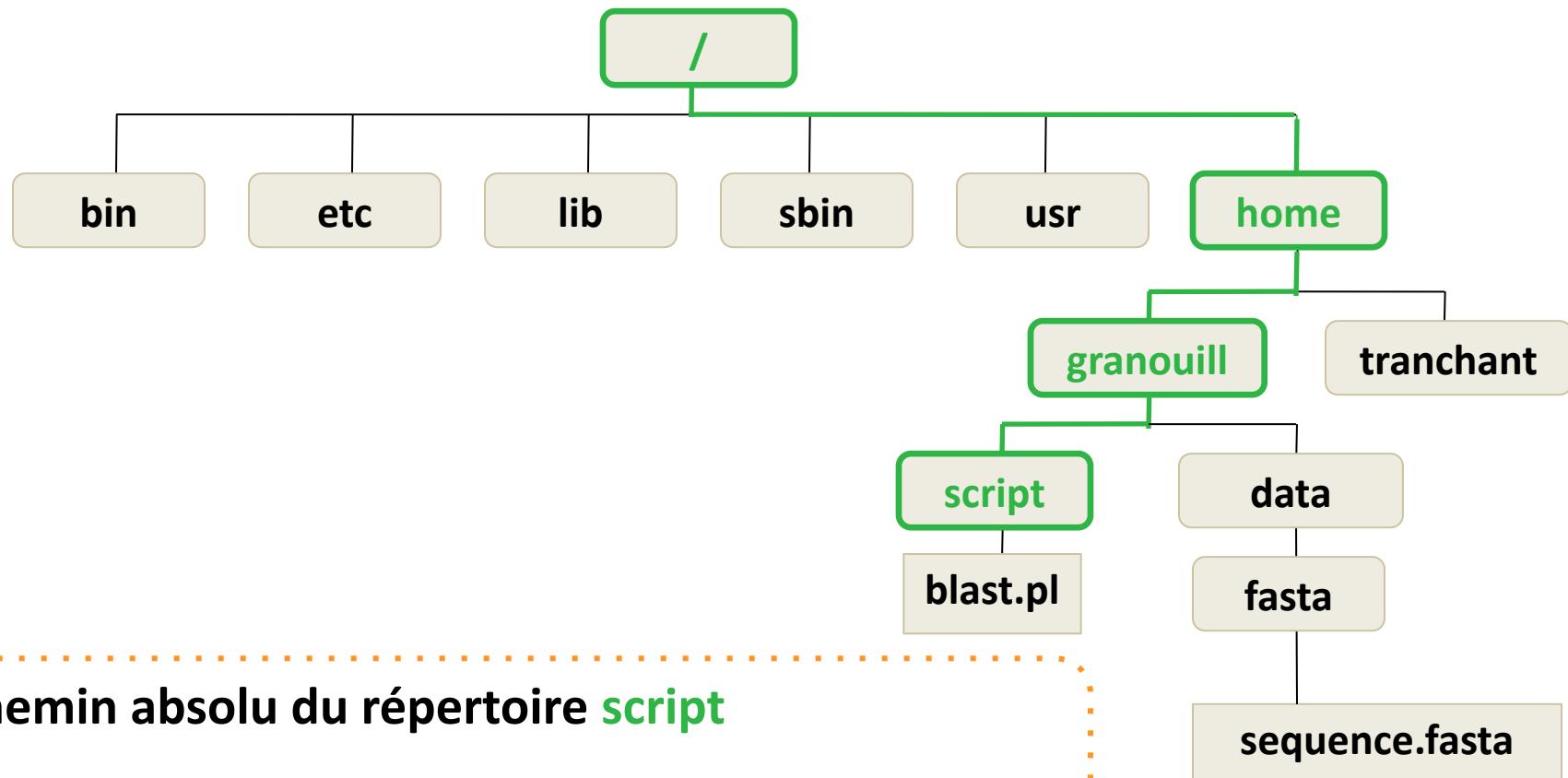
- Commence toujours par / (le répertoire racine)
- Toujours correct peu importe où l'on travaille



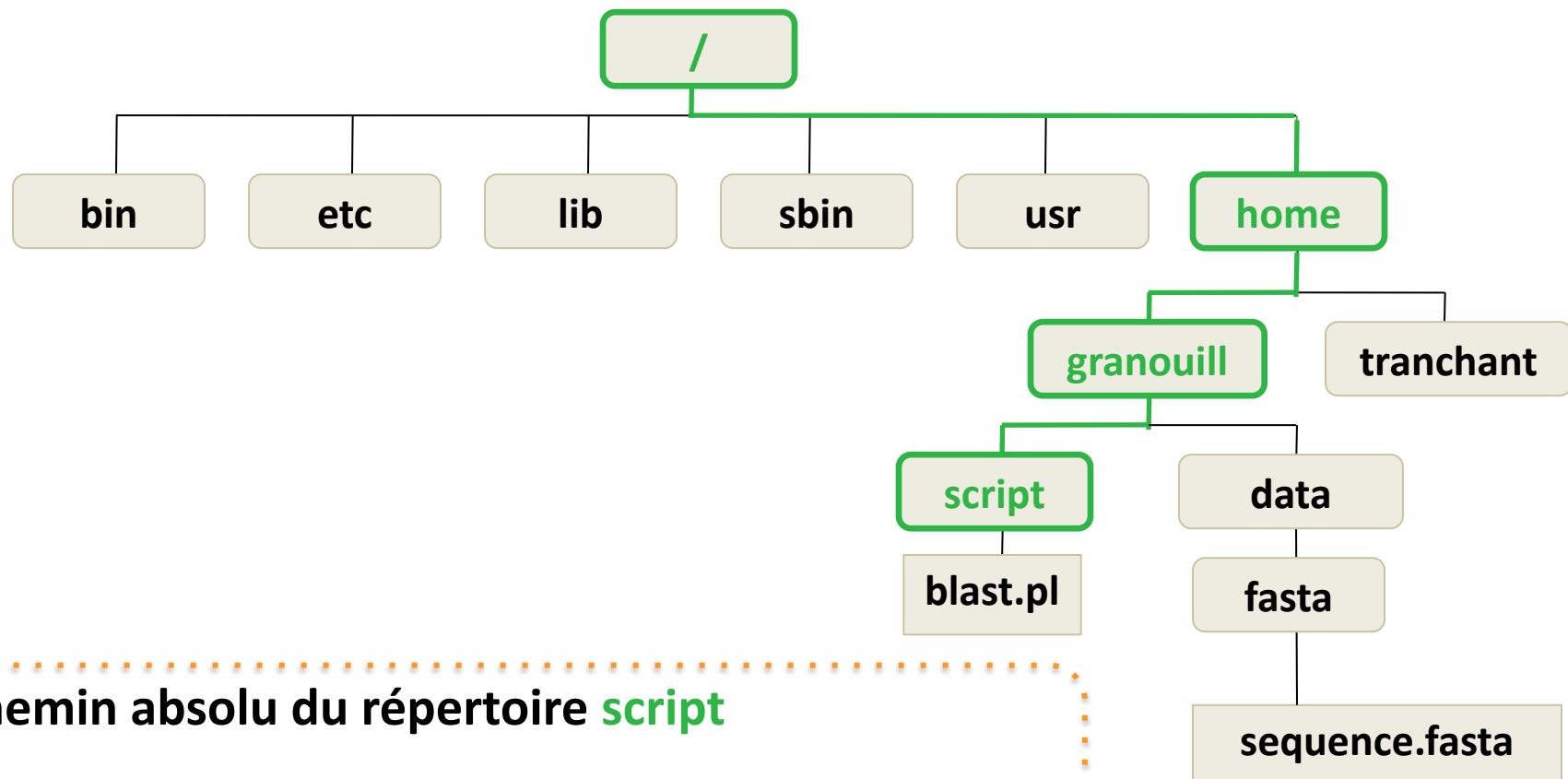
- Commence toujours par / (le répertoire racine)
- **Toujours correct peu importe où l'on travaille**



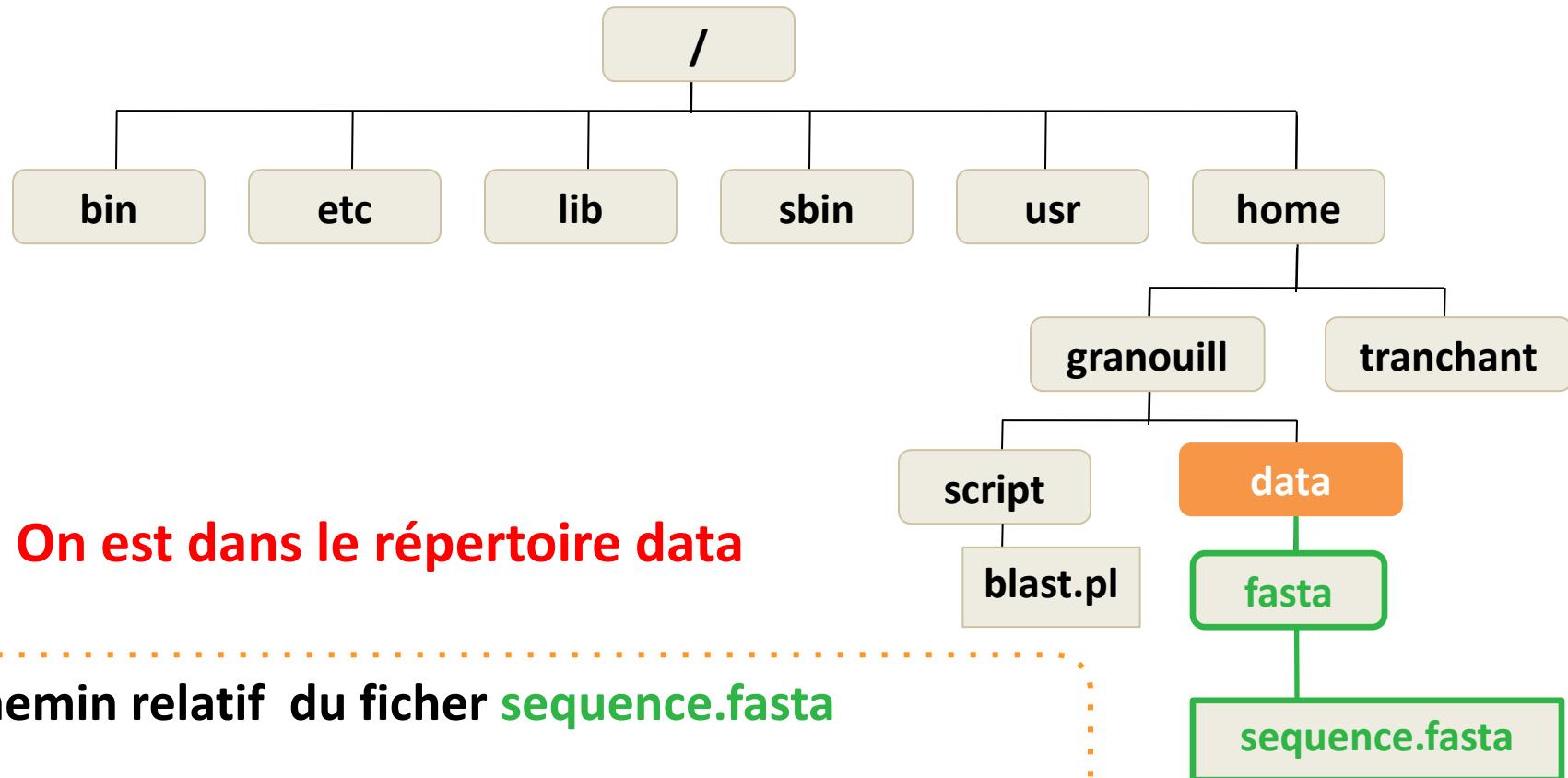
- Commence toujours par / (le répertoire racine)
- Toujours correct peu importe où l'on travaille



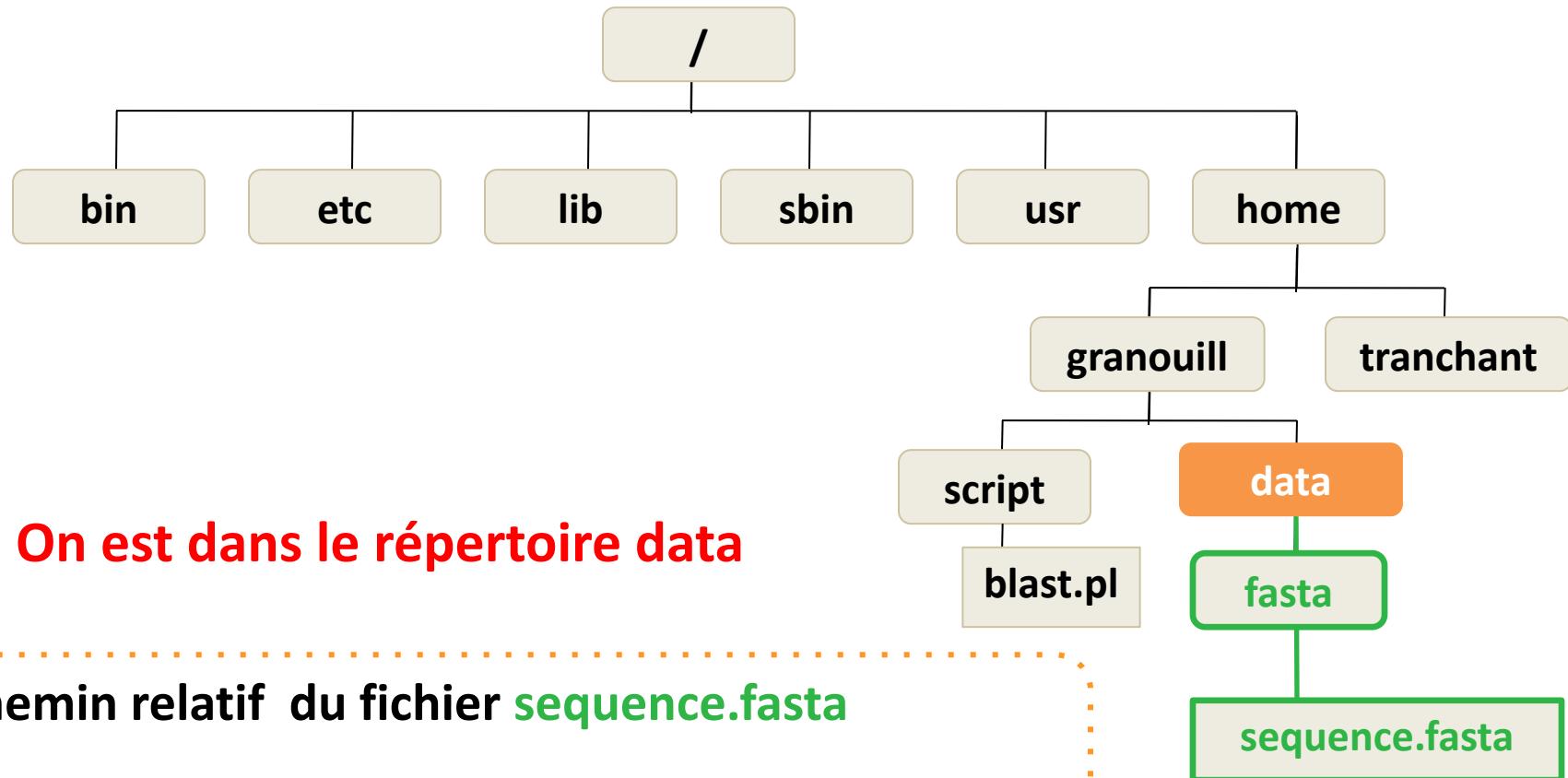
- Commence toujours par / (le répertoire racine)
- Toujours correct peu importe où l'on travaille



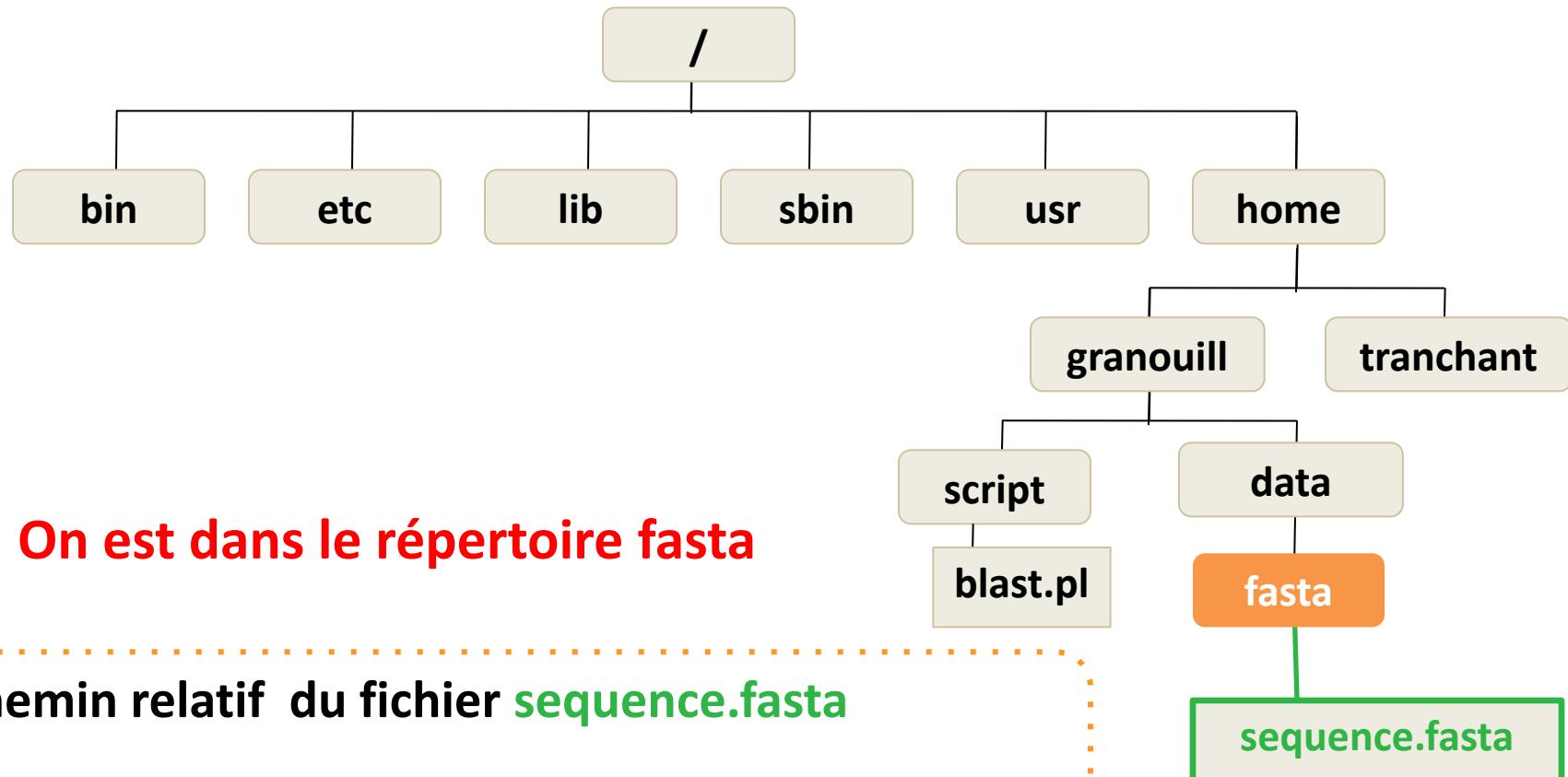
- Défini par rapport où on est dans l'arborescence
- Ne commence jamais par /



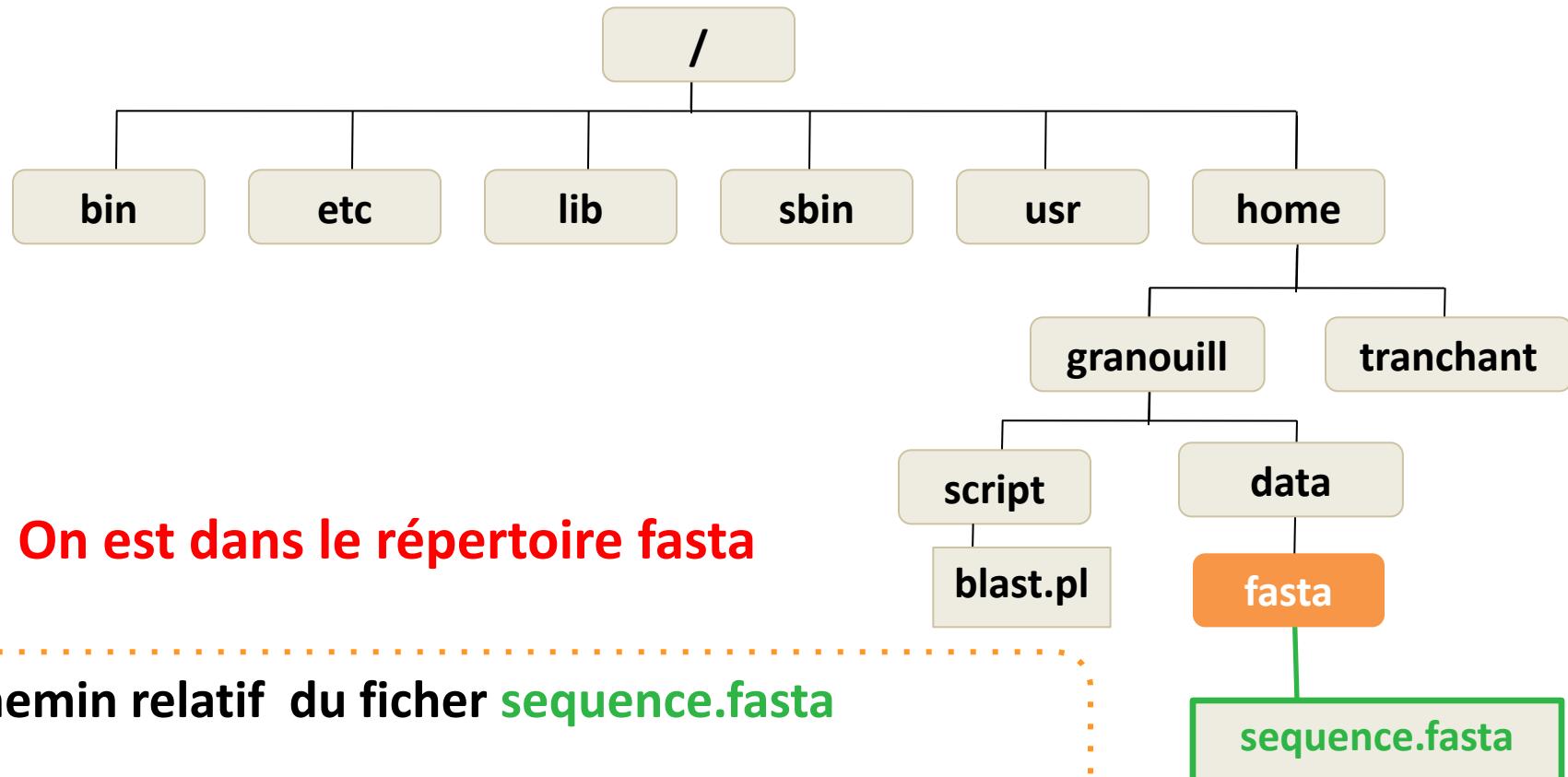
- Défini par rapport où on est dans l'arborescence
- Ne commence jamais par /



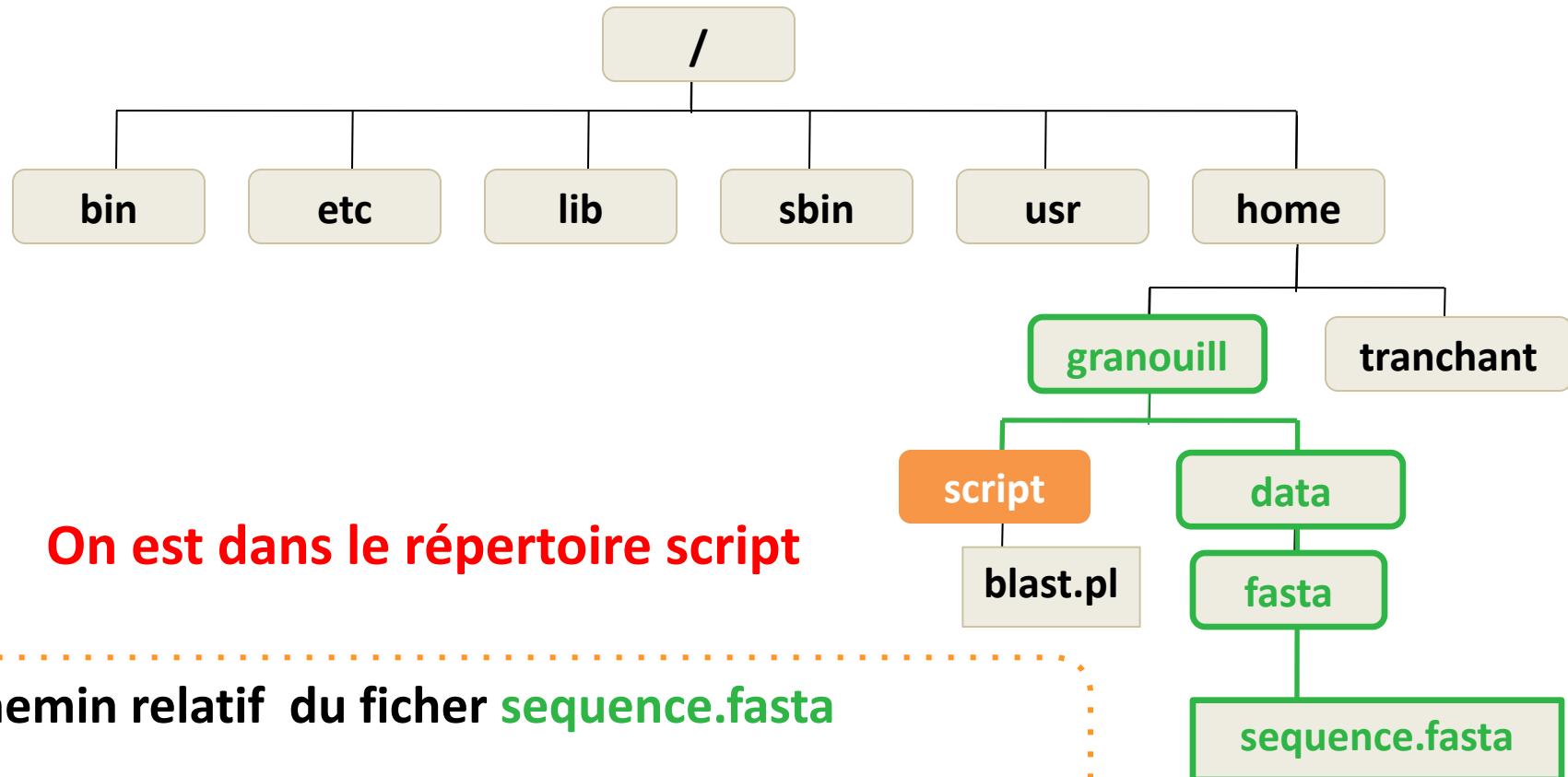
- Défini par rapport où on est dans l'arborescence
- Ne commence jamais par /



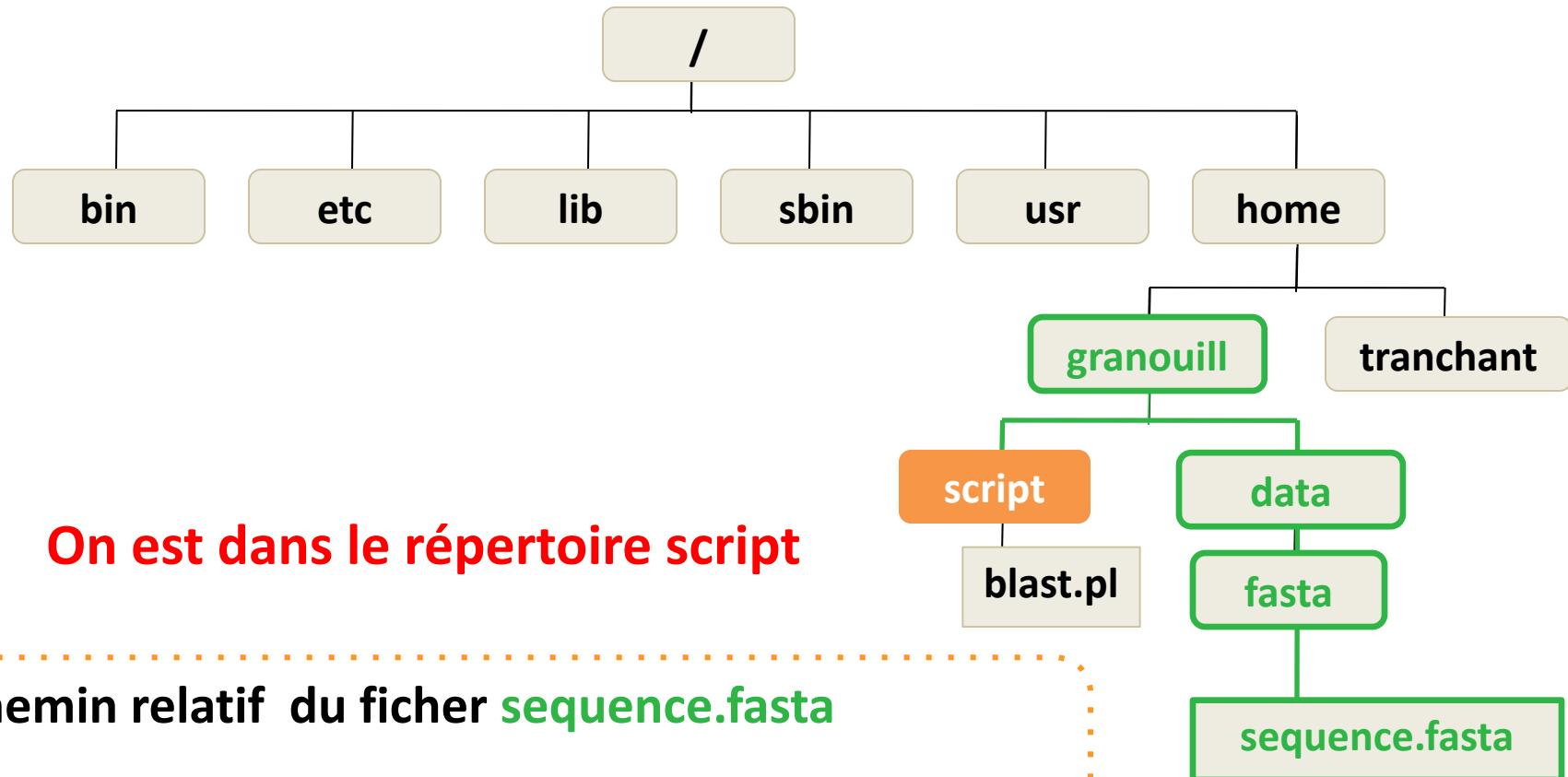
- Défini par rapport où on est dans l'arborescence
- Ne commence jamais par /



- Défini par rapport où on est dans l'arborescence
- Ne commence jamais par /



- Défini par rapport où on est dans l'arborescence
- Ne commence jamais par /





# Practice

ls

4

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

## Interagir avec les processus

- |                         |   |
|-------------------------|---|
| <b>&lt;Ctrl&gt; + C</b> | Arrêter le processus en cours sous le terminal      |
| <b>&lt;Ctrl&gt; + Z</b> | Suspendre le processus et le mettre en arrière plan |

## Tab completion

- |                               |   |
|-------------------------------|---|
| <b>&lt;Tab&gt;</b>            | Complète automatiquement le nom d'un fichier/<br>répertoire qui est en cours de saisie (choix unique) |
| <b>&lt;Tab&gt;&lt;Tab&gt;</b> | Affiche la liste des différentes possibilités si le<br>choix n'est pas unique                         |

## Interagir avec l'historique de commandes

**Flèche bas/haut**      - Afficher la commande précédente/suivante  
                          - Presser plusieurs fois pour naviguer dans  
l'historique

**<Ctrl> + R**      - Afficher la dernière commande qui contient  
les caractères saisis.  
                          - Presser les touches et commencer à taper la  
commande recherchée

- Linux est sensible à la casse

Sequence.fasta ≠ SEQUENCE.fasta ≠ sequence.fasta

- Utiliser uniquement des lettres, chiffres et caractères \_ . -

- PAS d'espaces, accents et caractères spéciaux

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

- Le suffixe des noms de fichiers (.txt, .fasta, .fa, .fq etc.) est optionnel



# Travailler avec les “jokers”

Métacaractères : \*, [ ]

# Qu'est ce un caractère "joker" ?

C'est un caractère qui peut être utilisé comme substitut de plusieurs caractères ou classes de caractère dans une recherche

Peut être utilisé avec de nombreuses commandes linux

# Comment utiliser les caractères jockers ?

\*

N'importe quelle chaîne de caractère



KYVF-01.R1.fastq	KYVF-02.R1.fastq	KYVF.sam	Z016.fastq
KYVF-01.R2.fastq	KYVF-02.R2.fastq	KYVF.bam	Z016.bam

```
ls *fastq
```

# Comment utiliser les caractères jockers ?

\*

N'importe quelle chaîne de caractère



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	

# Comment utiliser les caractères jockers ?

\*

N'importe quelle chaîne de caractère



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
```

# Comment utiliser les caractères jokers ?

\*

N'importe quelle chaîne de caractère



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

- \* N'importe quelle chaîne de caractère
- [ensemble] Tous les caractères dans l'ensemble
- [!ensemble] Tous les caractères pas dans l'ensemble



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 *. [sb]am
```

- \* N'importe quelle chaîne de caractère
- [ensemble] Tous les caractères dans l'ensemble
- [!ensemble] Tous les caractères pas dans l'ensemble



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

KYVF.sam ZO16.bam  
KYVF.bam



# Practice

ls, \*

5

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



# Commandes relative à l'arborescence de fichiers

commande cd

**cd**

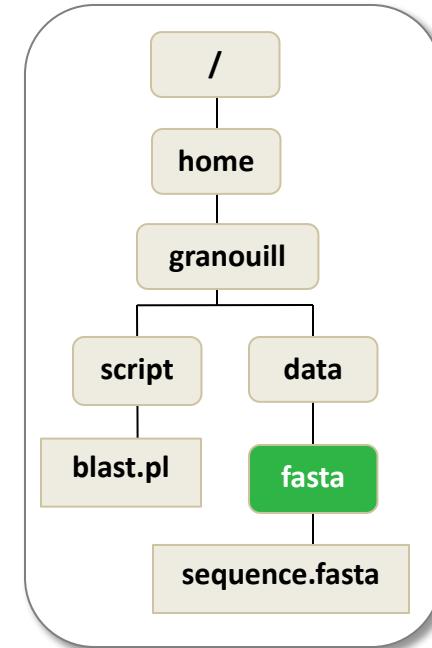
*pour se déplacer dans l'arborescence*

*Change Directory*

*cd nom\_repertoire (chemin absolu ou relatif)*

`cd nom_repertoire (chemin absolu ou relatif)`

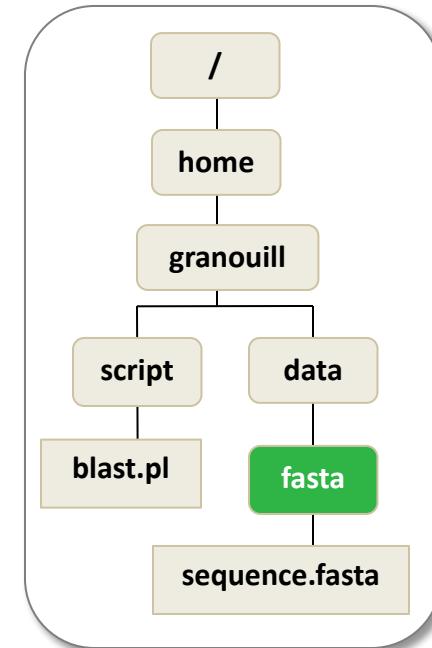
**Chemin absolu :**



`cd nom_repertoire (chemin absolu ou relatif)`

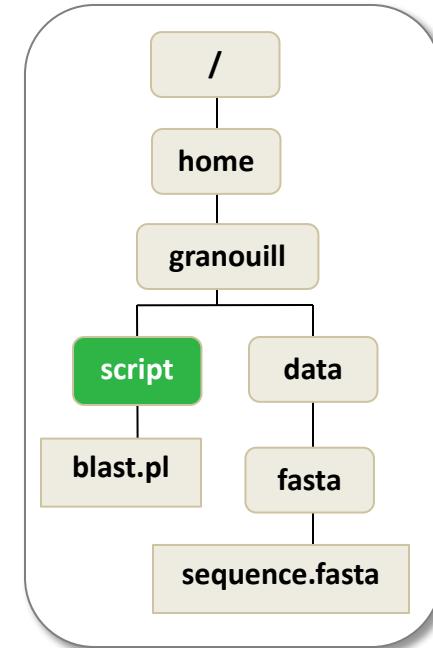
**Chemin absolu :**

`cd /home/granouill/data/fasta`



`cd nom_repertoire (chemin absolu ou relatif)`

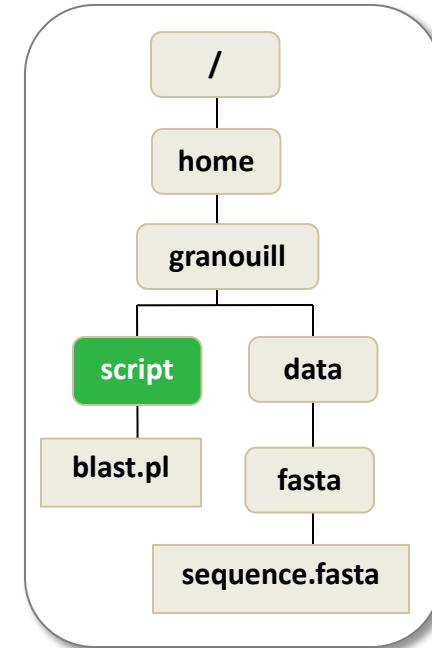
**Chemin absolu :**



`cd nom_repertoire (chemin absolu ou relatif)`

**Chemin absolu :**

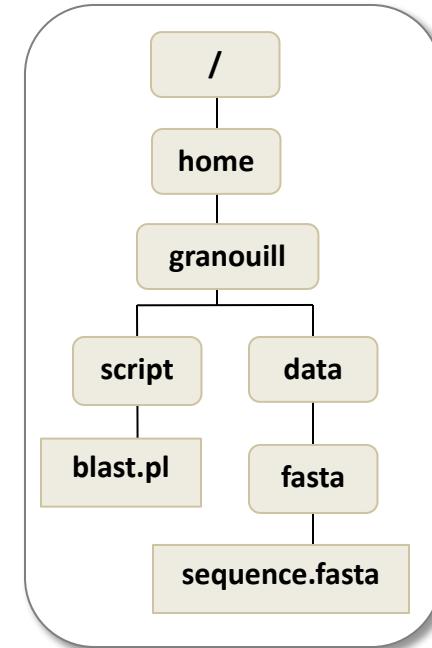
`cd /home/granouill/script/`



*cd nom\_repertoire (chemin absolu ou relatif)*

## Chemin relatif :

Commande	Se déplace dans	
cd directory_name	directory_name	Retour au répertoire de l'utilisateur courant home
cd	home directory	
cd ..	le répertoire parent	Remonte d'1 répertoire
cd ../../..	le répertoire parent du répertoire parent	Remonte de 2 répertoires



## Arborescence linux

**pwd**

Affiche le chemin absolu

**ls**

Liste tous les fichiers/répertoires

**cd rep\_name**

Se déplace dans rep\_name

**mkdir rep\_name**

Crée un répertoire

**rm nom\_fichier**

Supprime un fichier

**cp fichier\_source repertoire\_cible**

Copie un fichier dans un répertoire

**cp fichier\_source nouveau\_fichier**

Copie un fichier sous un nouveau nom

**cp fichier\_source repertoire\_cible/nouveau\_fichier**

Mix des 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



# Des commandes pour éditer des fichiers et les manipuler

# Commande cat

**cat nom\_fichier**

*Affiche le contenu d'un fichier sans pagination  
(à éviter s'il est volumineux!!)*

```
MacBook-Pro-de-Christine:Data tranchan$ cat Data/Fasta/sequence.fasta
>Gxbjbsjxbjs
CCACCCCTCTACAGTCTCACCAAATGTCTTTAAACTCCACCTAAAGTATCCAAAGA
CTCGAGAAATGCTGTGCCACAACCAGCTTGAGTCATCCATGACCGTTGATCTCCTT
GCCCCCAGAGTGGGCCTAGCACCATCTAGCTACTACTTGCCCTTCATAACCCATCATTGG
GATACCCCTGAATACCTATCTTATAAGTTCCATATGGCTTATTTCTAAGTAAGAGATGC
ACTTAGTAAGTGCATGTCGTCTGACTGTTACTCTAATGTATGATATTTATATCCC
TATAATATAGTGTACTAATATATGTTGGTATTGTGTAGACTCCATTGTACCATGGTGT
GCTAATTAGAAATAACATGCCAGCTTGCTATTGTGGTTGCAAGTAAAGTAAAAAAA
MacBook-Pro-de-Christine:Data tranchan$
```

# Commande less

**less nom\_fichier**

*Affiche le contenu d'un fichier*

<space> pour défilez

q pour quitter

/ suivi par le mot recherché

*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-XX-SH3-053-F02-EM, mRNA sequence
AATATTGGTGTACCGCTAACCTCTGCACAAAAGTAAGAGAAGGGTAACAAGTAAAGGG
AGCGAGACTATGGCTGCCACATCTGCTGAGACTAAATGGATTGGGCTCCCTCTGACTGGTGAA
GGAAAAGTCAGGCCCTGCTGGCTCACAGCTGCAGCTAGAGTCGGTGGTGTGCTGCTGTTGCTCAA
GAGATTAGTTGTGGTAGCTCGTCGCCCCAAGAAAGTCATGGATCCTGGTTAGAACTGGCCAGGC
TTCTTGACCCCGAGTATCTGATGGCTCGCTCCCTGGTGAACAGGGTTGATCCACTGGCTTGGCA
AGGATCCACCATTCGAATGGTATAGAGAAGCTGAGCTATTGATGGCAGATGGCAATGACTGCACT
TGGTGGAACTTTGGTGGCCAGGCATGGAGTGGTATCCCAGTGGTTCAAGCTGGCGTGAACCTGGTGC
ATTGCGCCATTCTCCTTGGTTCGCTCGGACTCAACTCCTGCTCA
>gi|315911381|gb|GT649552.1|GT649552 CC00-XX-SH3-077-D04-EM.F Coffea canephora SH3 Coffea canephora cDNA clone CC00-XX-SH3-077-D04-EM, mRNA sequence
ATTCATCGTATCTGCCCTTTCTTCATCTCTCTTGTAGTCAGTCAGCAGAGGCACCGGCAGC
TTTAAAAAGCTTATCACACACACACACATTCTTGTAGTCAGTCAGTCAGGCAACAAATTACTGAGTACT
ACTAGTTACTTATCGAACAAATTGATTAGCTTGTAGCTAGGAGGTACGGTTGGTTGACTTGTGTTG
AGTATAAATATTGATGGGATGGGATGCTGCGGAGTTAACATTGGAGGGAGACTGAAACAGCTAAACCA
GTCGAGCCTAATTCTCAGTCGGTAGCGGAGGGATGGAATTCACTCATATTGTAGCTTGG
Data/Fasta/EST-68566-Coffeacanephora.fasta
```

<b>head</b>	Affiche les premières lignes d'un fichier (n=10, 10 lignes par défaut)	<i>head -n 20 script.pl</i>
<b>tail</b>	affiche les dernières lignes d'un fichier (n=10 par défaut)	<i>tail -n 5 script.pl</i>
<b>wc</b>	Compter le nombre de lignes, mots ou caractères d'un fichier	<i>wc script.pl</i> <i>wc -l script.pl</i>

# Commande grep

**grep**

*pour rechercher un motif dans un fichier*

*grep [options] motif [file1, ... ]*

**Pour rechercher un motif, le mettre entre simple/double quote  
=> ex : ‘gene’ ou “gene”**

# Commande grep

**grep**

*pour rechercher un motif dans un fichier*

*grep [options] motif [file1, ... ]*

Option	Description
-i	Recherche le motif sans tenir compte de la casse
-c	Compte le nombre de lignes dans lesquelles le motif a été trouvées
-v	Affiche seulement les lignes sans le motif
-l	Affiche uniquement les noms de fichiers dans lesquels le motif a été trouvé
-P	Utilise les expressions régulières du langage Perl équivalent à <b>egrep</b>



# Practice

ls

8

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

# Commande *cut*

**cut**

Sélection de colonnes/champs d'un fichier

```
cut -d separateurColonne -f fieldNumber fileName
```

```
cut -d ":" -f 1,5 /etc/passwd
```

Sélection des champs 1 et 5 dans un fichier tabulé  
dont les colonnes sont séparées par un :

# Commande *sort*

## sort

Trier une ou plusieurs colonnes d'un fichier tabulé

```
sort -k2 fileName
```

sur la colonne 2 (tri alphanumérique)

```
sort -k2r fileName
```

sur la colonne 2 (décroissant)

```
sort -k2g -k1r fileName
```

sur les colonnes 2 (numérique) et 1 (décroissant)

```
sort -t: -k3g fileName
```

sur la colonne 3 (numérique), le délimiteur de colonne étant :



# Practice

9

Go to [Practice 9](#) on our github



# Les entrées / sorties

**pour sauvegarder la sortie d'une  
commande dans un fichier**

La sortie d'une commande peut être écrite  
dans un fichier avec les caractères :

>

>>

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

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

## Redirection

## Action

Command > file

- si le fichier n'existe pas : il sera créé
- si le fichier existe : efface le contenu

Command >> file

- si le fichier n'existe pas : il sera créé
- si le fichier existe : écrit à la fin du fichier

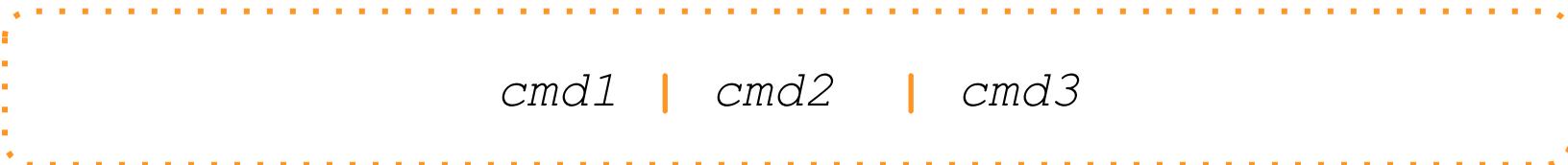


# Practice

10

Go to [Practice 10](#) on our github

- La sortie d'une 1ère commande peut être envoyée comme input d'une 2ème commande
- Pour connecter/combiner plusieurs commandes ensemble dans la même ligne de commande (sans fichier intermédiaire)
- Pipelines= *workflow*



*cmd1 | cmd2 | cmd3*

```
cut -d: -f1 /etc/passwd
```

Root

troot

iroot

ctroot

//

```
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
```

Root

troot

iroot

ctroot

//

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

abate

adm

adroott

ais

#albar

alvaro-wis

anthony

apache

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

```
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

11

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

# Autres commandes utiles



Taille (et espace libre) 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"*

Permettent d'attribuer un autre chemin d'accès à un fichier en pointant sur un nom de fichier.

Il s'agit d'un raccourci **In**

```
ln -s levrai lefaux
```

Exemple: ln -s /opt/jdk-7.01 /opt/jdk

Permettent de gagner de l'espace disque sur un système: seul le “vrai” fichier pèse



# Practice

12-13-14

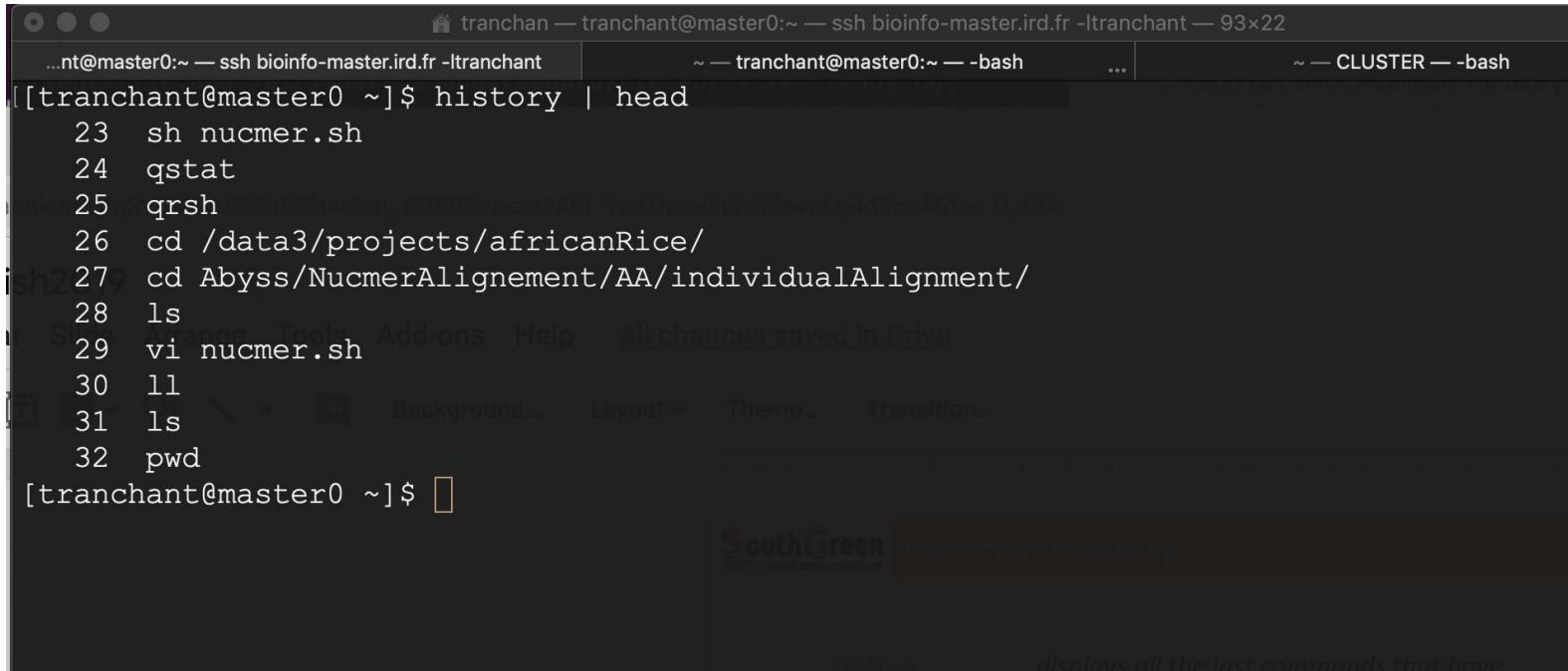
Go to [Practice 12 ,13 and 14](#) on our github

# Commande history

**history**

*afficher toutes les commandes enregistrées de la plus ancienne à la plus récente*

**L'historique est enregistré dans le fichier .bash\_history**



A screenshot of a terminal window titled "tranchant — tranchant@master0:~ — ssh bioinfo-master.ird.fr -ltranchant — 93x22". The window shows the command "history | head" being run, which displays the following command 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 ~]$ ]
```

The terminal window has a dark background and light-colored text. At the bottom, there is a toolbar with icons for "Slide", "Annotate", "Tools", "Add-ons", "Help", and a status message "All changes saved in Drive". The bottom right corner of the slide contains the SouthGreen logo.

**history**

*afficher toutes les commandes enregistrées de la plus ancienne à la plus récente*

**L'historique est enregistré dans le fichier .bash\_history**

## Filtering the History Output

`history | head -n 5`

Liste les commandes les plus anciennes

`history | tail`

Liste les commandes les plus récentes

`history | grep "blastn"`

Recherche les anciennes commandes “blastn”

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

## Compresser des fichiers **tar,gzip**

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

## Décompresser une archive **gunzip, tar**

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

## Afficher le contenu d'une archive **zcat**

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

## Recherche d'expression/motif dans une fichier compressé: **zgrep**

```
zgrep 'NM_000020' data.gz
```

# Renommer des fichiers

## rename

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

- **Guilhem Sempere**



- **Ndomassi Tando**



- **Alice Boizet**



- **Bruno Granouillac**



- **Christine Tranchant-Dubreuil**



Si vous utilisez les ressources du plateau i-Trop.

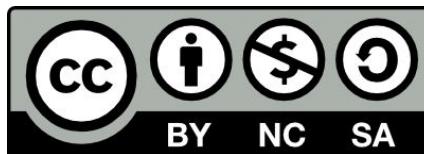
Merci de nous citer avec:

“ The authors acknowledge the ISO 9001 certified IRD i-Trop HPC (South Green Platform) at IRD montpellier for providing HPC resources that have contributed to the research results reported within this paper.

URL: <https://bioinfo.ird.fr/> - <http://www.southgreen.fr>”

- Pensez à inclure un budget ressources de calcul dans vos réponses à projets
- Besoin en disques dur, renouvellement de machines etc...
- Devis disponibles
- Contactez [bioinfo@ird.fr](mailto:bioinfo@ird.fr) : aide, définition de besoins, devis...

# Merci pour votre attention !



Le matériel pédagogique utilisé pour ces enseignements est mis à disposition selon les termes de la licence Creative Commons Attribution - Pas d'Utilisation Commerciale - Partage dans les Mêmes Conditions (BY-NC-SA) 4.0 International:

<http://creativecommons.org/licenses/by-nc-sa/4.0/>

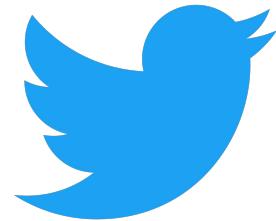


**Merci de prendre 5 min pour remplir  
l'enquête**

<https://itrop-survey.ird.fr/index.php/515725?lang=fr>



**SUIVEZ NOUS SUR TWITTER !**



South Green : [@green\\_bioinfo](#)



i-Trop : [@ItropBioinfo](#)



## N'oubliez pas de nous citer !

### Comment citer les clusters?

"The authors acknowledge the IRD i-Trop HPC at IRD Montpellier for providing HPC resources that have contributed to the research results reported within this paper. URL: <http://bioinfo.ird.fr/> "

"The authors acknowledge the CIRAD UMR-AGAP HPC (South Green Platform) at CIRAD montpellier for providing HPC resources that have contributed to the research results reported within this paper. URL:  
<http://www.southgreen.fr>"

## Commande ls -l

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

Permissions Proprio Groupe Taille Heure et date de la dernière modification

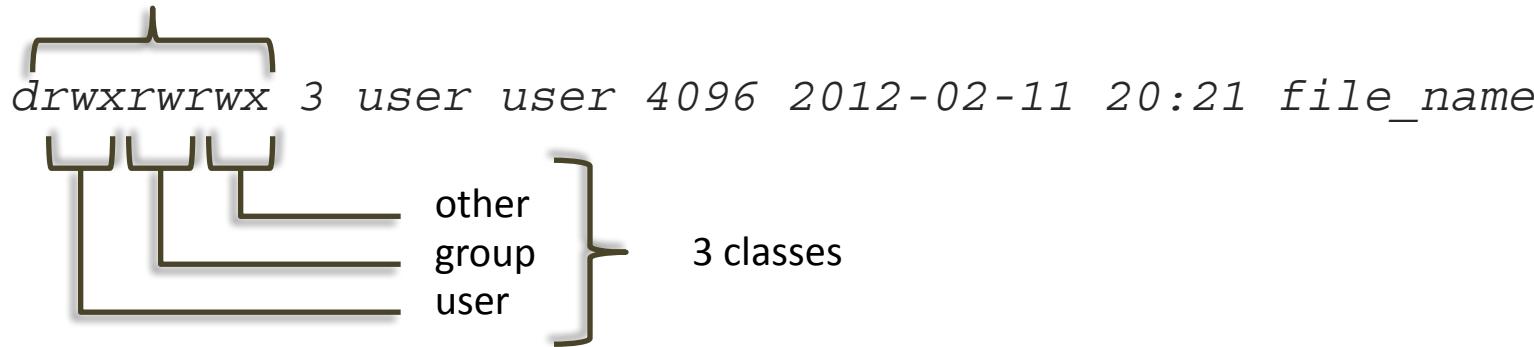
## Interprétation/Légendes des permissions

## Type

- : fichier normal
  - d : répertoire
  - l : lien symbolique

## ls -l command

Permissions



**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 valeur

R	4
W	2
X	1
none	0

## Exemple

```
chmod 740 script.sh  
chmod 755 script.sh
```

```
# Owner=rwx Group=r-- Other=---  
# 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

