

Guide de survie à Linux

Christine Tranchant-Dubreuil
Équipe RICE, UMR DIADE

Alexis Dereeper
UMR IPME



The objectif!

Lancez vos analyses bioinformatiques sous Linux



Applications

Connaître les principales commandes sous Linux

- Manipuler des fichiers (sort, cut, wc, tr)
- Trier et filtrer les données (grep / sed / awk)
- Lancer des logiciels bioinformatiques en ligne de commande

- **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**

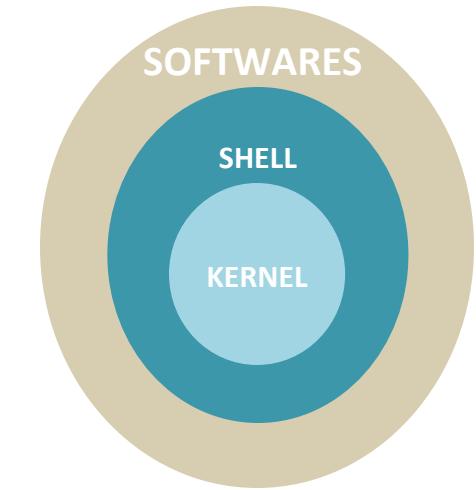
Plusieurs utilisateurs peuvent travailler en même temps

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

Chaque utilisateur peut lancer plusieurs programmes en même temps

Distribution sous Linux ?

Distribution : Noyau/Coeur + logiciels/programmes



- 2 façons d'utiliser linux :

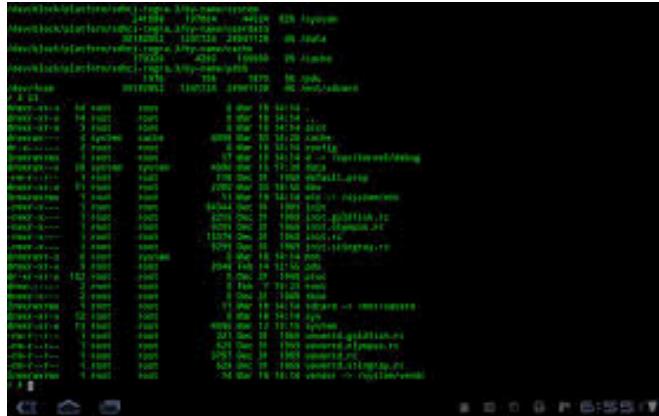
en *mode graphique* (équivalent windows)



- 2 façons d'utiliser linux :

en ***mode graphique*** (équivalent windows)

en ***mode console*** (équivalent DOS)

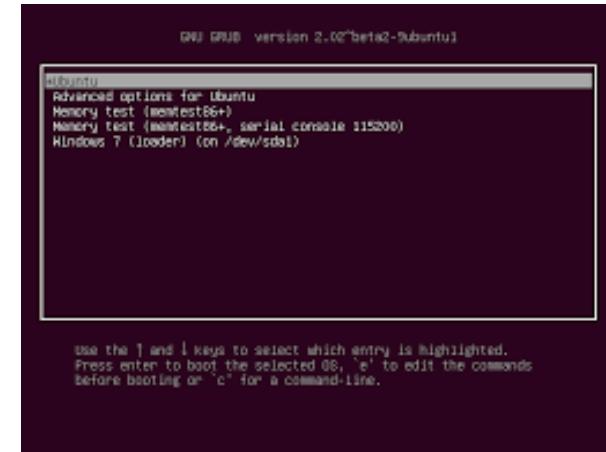


A screenshot of a Linux desktop environment. A terminal window is open, displaying a command-line interface with various system logs and commands. The desktop background is dark, and there are icons at the bottom of the screen.



Comment accéder à Linux ?

- Installer linux et windows sur son ordinateur et choisir au démarrage de son ordinateur

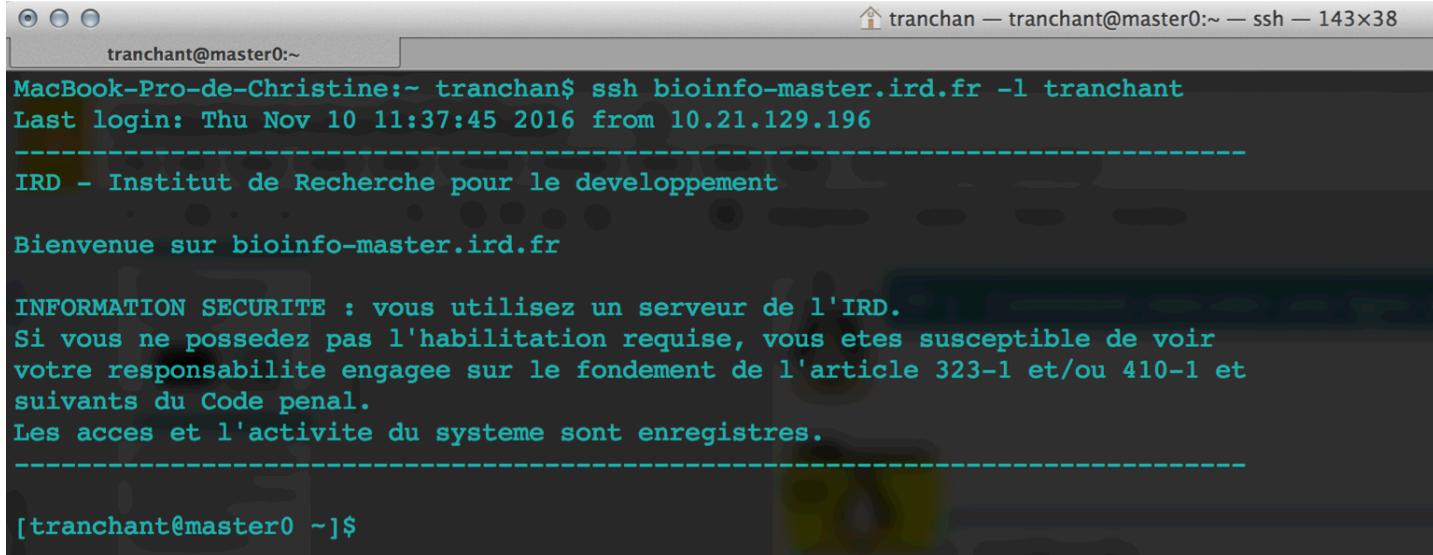


- Installer linux et windows **sur son ordinateur** et choisir au démarrage de son ordinateur
- **Sur son ordinateur**, à partir de windows ou mac os via une machine virtuelle ou docker



Comment accéder à Linux ?

- Installer linux et windows **sur son ordinateur** et choisir au démarrage de son ordinateur
- **Sur son ordinateur**, à partir de windows ou mac os via une machine virtuelle ou docker
- En se connectant sur un serveur linux distant de windows ou mac (via le protocole ssh)



The screenshot shows a terminal window with the following details:

- Title Bar:** tranchan — tranchant@master0:~ — ssh — 143x38
- User Prompt:** tranchant@master0:~
- SSH Connection Info:** MacBook-Pro-de-Christine:- tranchant\$ ssh bioinfo-master.ird.fr -l tranchant
- Last Login:** Last login: Thu Nov 10 11:37:45 2016 from 10.21.129.196
- Server Identification:** IRD - Institut de Recherche pour le développement
- Welcome Message:** Bienvenue sur bioinfo-master.ird.fr
- Security Information:** INFORMATION SECURITE : vous utilisez un serveur de l'IRD.
Si vous ne possédez pas l'habilitation requise, vous êtes susceptible de voir
votre responsabilité engagée sur le fondement de l'article 323-1 et/ou 410-1 et
suivants du Code penal.
Les accès et l'activité du système sont enregistrés.
- Terminal Prompt:** [tranchant@master0 ~]\$



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



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



Pas d'interfaces graphiques

Convivialité de la ligne de commande ?



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



Nécessité de la pratique et de l'expérience

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

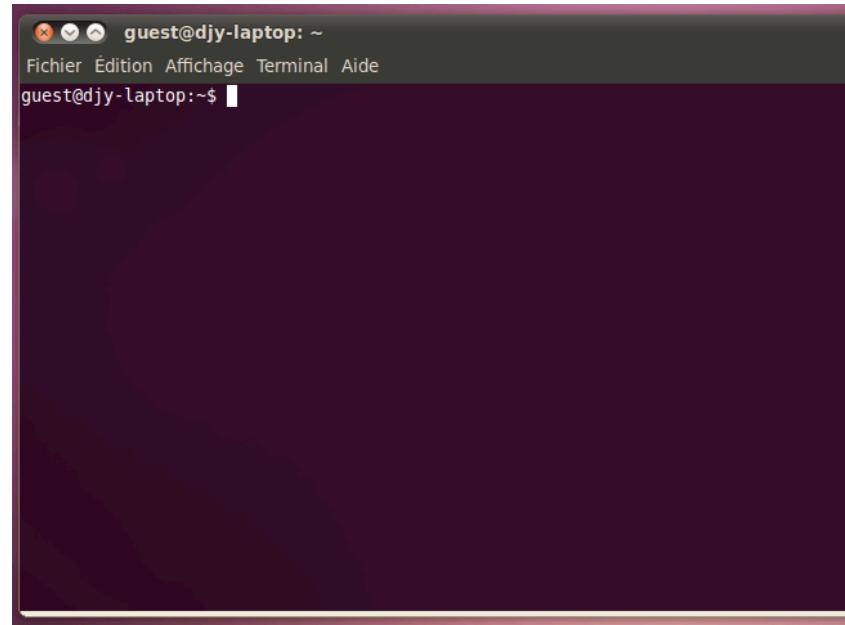


Se connecter sous une machine linux distante



- Installer putty
- Se connecter en SSH

Taper `ssh server_name -l YOUR_LOGIN`



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

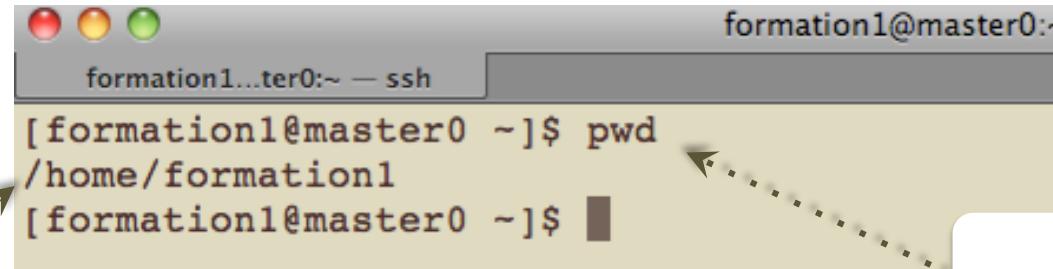
Répertoire
courant

1ère commande

pwd

Present Work Directory

affiche nom du répertoire courant (et son chemin complet)



A screenshot of a terminal window titled "formation1...ter0:~ - ssh". The window shows the command [formation1@master0 ~]\$ pwd followed by the output /home/formation1. The window has three colored buttons (red, yellow, green) at the top left and a status bar at the top right.

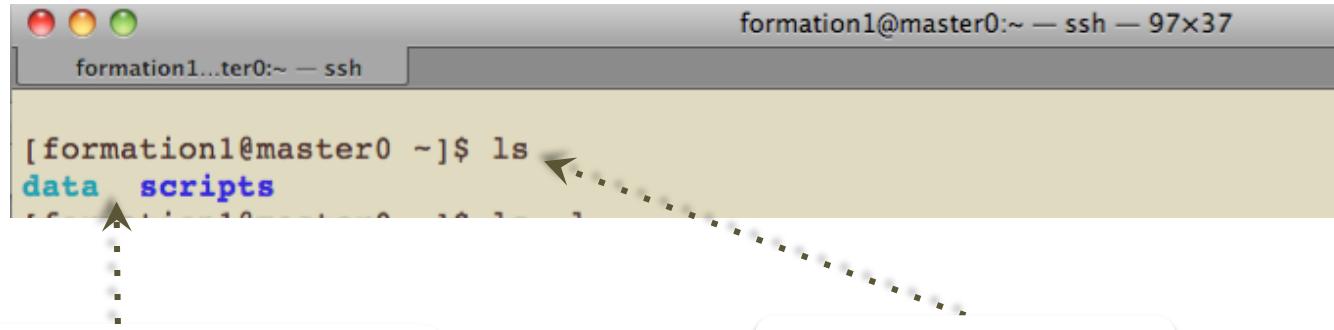
```
[formation1@master0 ~]$ pwd
/home/formation1
[formation1@master0 ~]$
```

Affiche le nom du répertoire courant

Commande

ls
list

liste les fichiers et répertoires présents dans un répertoire



A screenshot of a terminal window titled "formation1@master0:~ — ssh — 97x37". The window shows the command "[formation1@master0 ~]\$ ls" followed by "data scripts". A dotted arrow points from the word "scripts" in the terminal output to a callout box labeled "Commande". Another dotted arrow points from the word "scripts" to a callout box labeled "Liste des fichiers du répertoire courant (par défaut)".

Liste des fichiers du répertoire courant (par défaut)

Commande

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?

man ls pour obtenir le manuel d'aide sur une commande
ls --help

Comment obtenir de l'aide sur une commande?

man ls pour obtenir le manuel d'aide sur une commande
ls --help

Commandes

pwd Affiche le chemin absolu
ls Liste tous les fichiers/répertoires
ls -l affiche toutes les informations sur les fichiers



TP

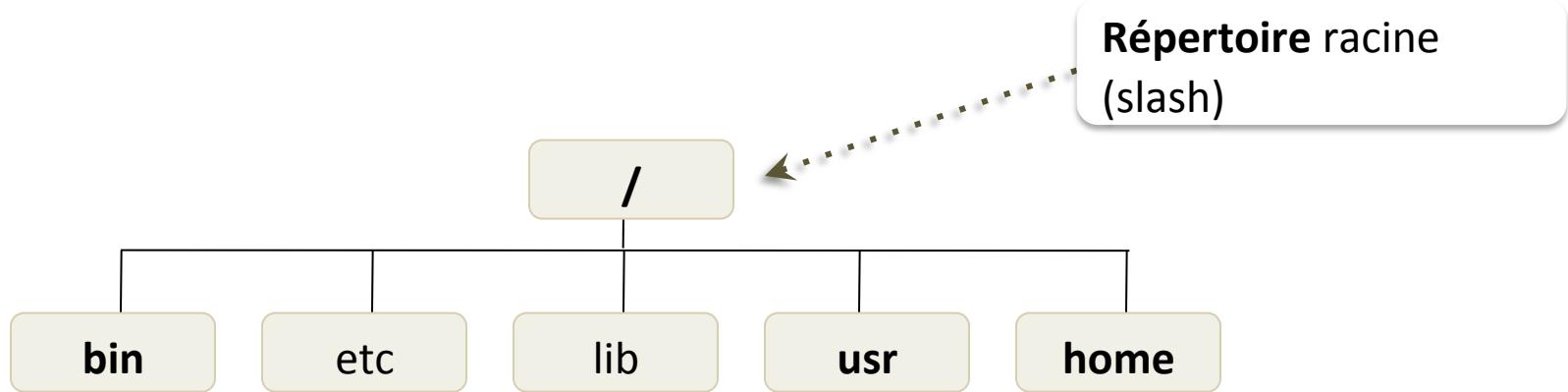
Exécuter ses premières commandes



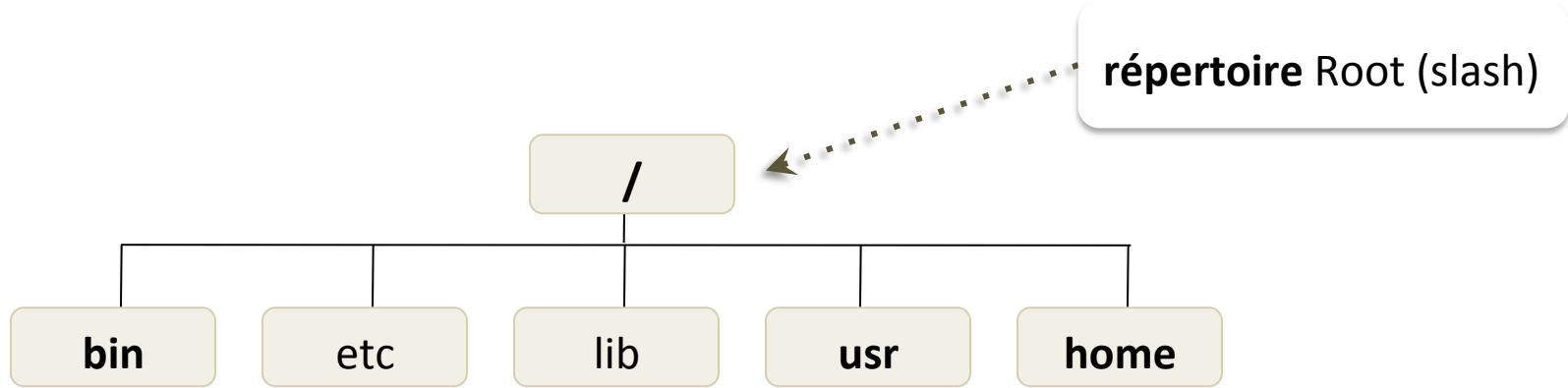
Sous un terminal :

- Quel est le répertoire courant/de travail affiché par le prompt?
- Vérifier avec la commande `pwd` le nom de répertoire de travail

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



Principaux répertoires



/bin	commandes principales
/usr, /opt	Applications et librairies
/usr/bin	Autres commandes
/home	répertoire des utilisateurs (1 par utilisateur/login)

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

- **Chemin (path)** : chemin d'accès à un fichier/répertoire
 - **Chemin absolu** : chemin complet du fichier en partant du répertoire racine /
commence toujours par /
Toujours correct peut importe où on l'on travaille

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

Chemin absolu : chemin complet du fichier en partant du répertoire racine /

commence toujours par /

Toujours correct peut importe où on l'on travaille

Chemin relatif : chemin du fichier défini par rapport où on est dans l'arborescence

Ne commence jamais par /

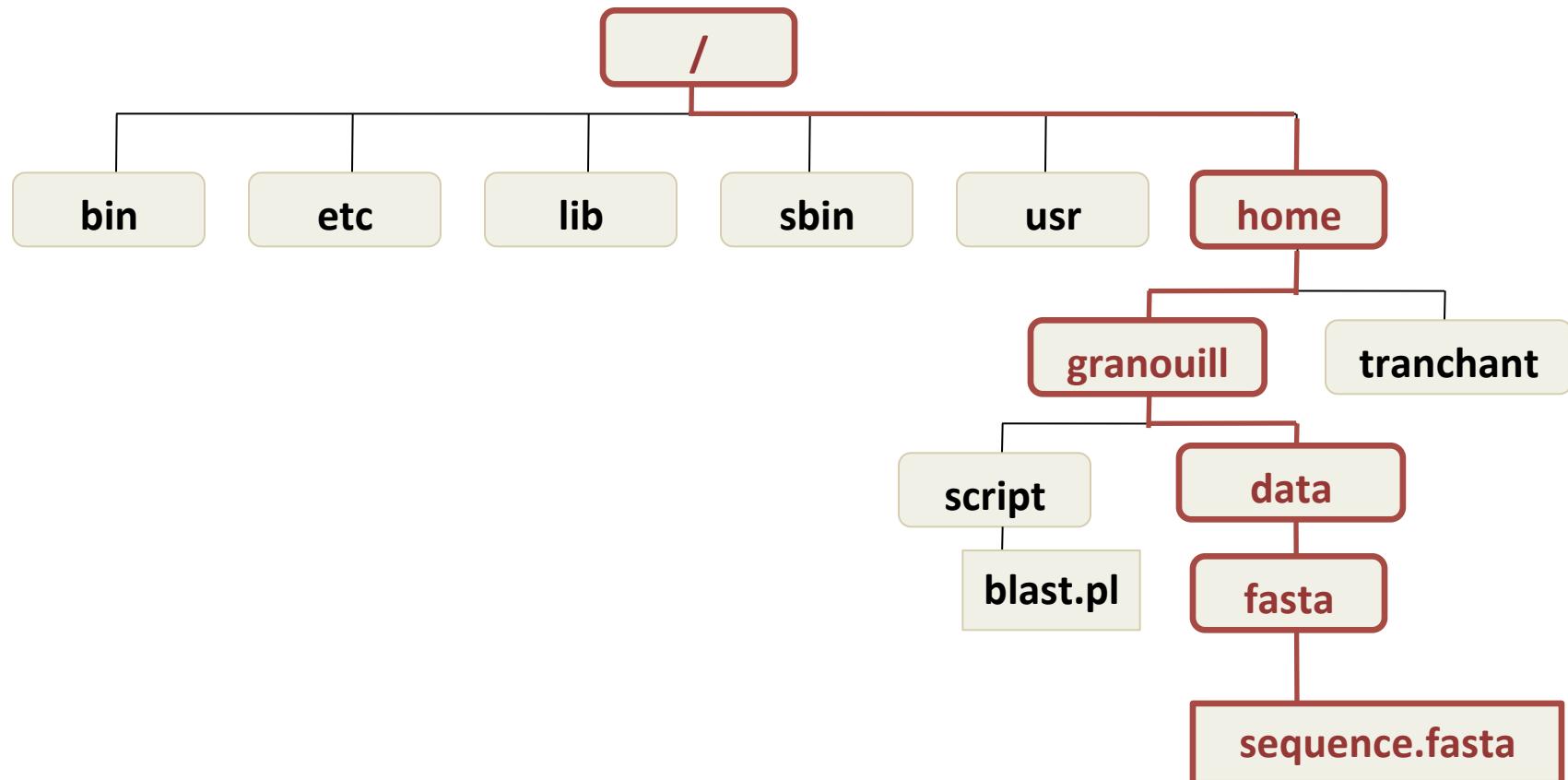
Change selon on l'on travaille

Chemin absolu d'un fichier

chemin complet du fichier en partant du répertoire racine /

↔ *commence toujours par /*

Toujours correct peut importe où on l'on travaille

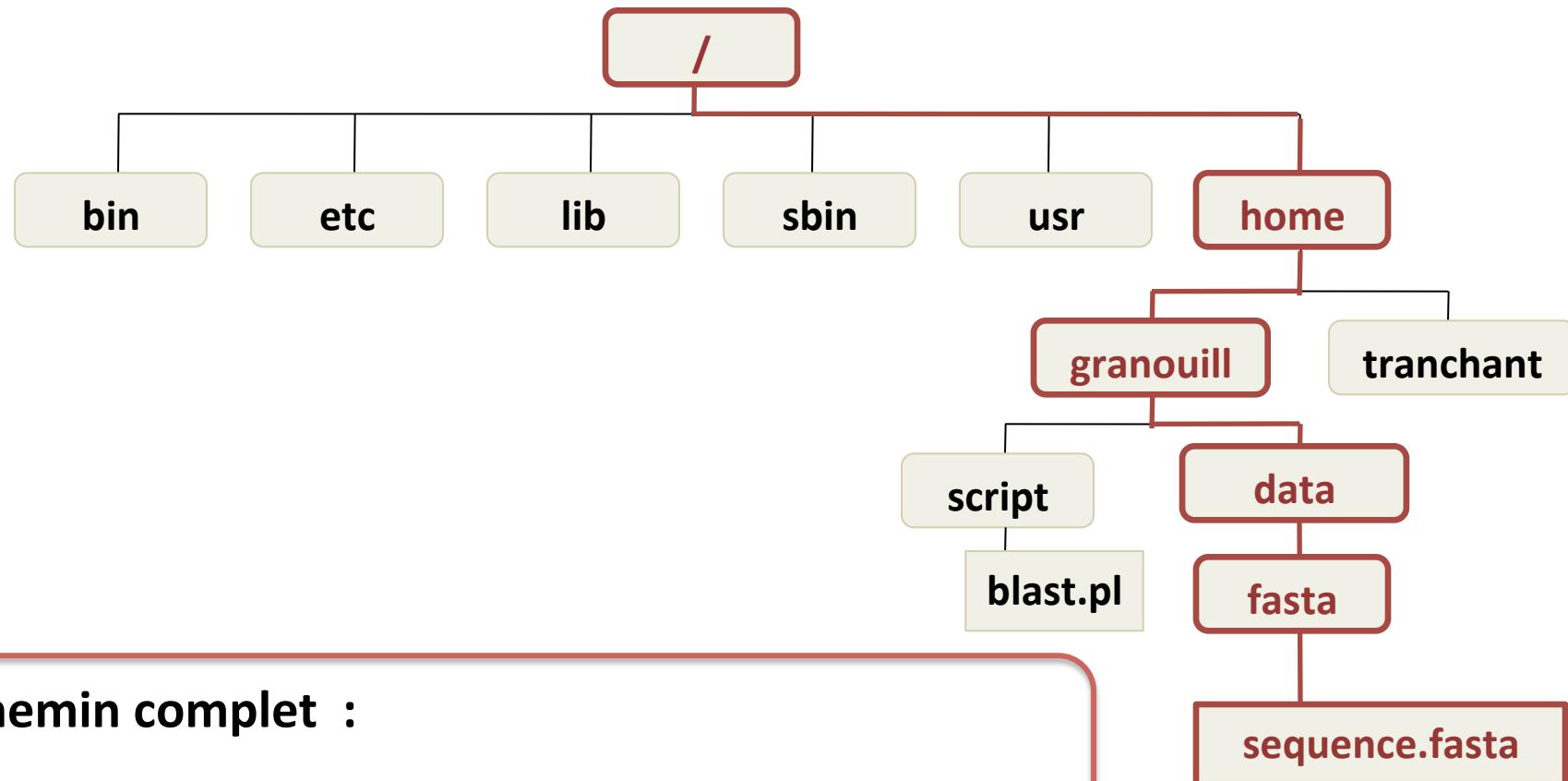


Chemin absolu d'un fichier

chemin complet du fichier en partant du répertoire racine /

↔ **commence toujours par /**

Toujours correct peut importe où on l'on travaille

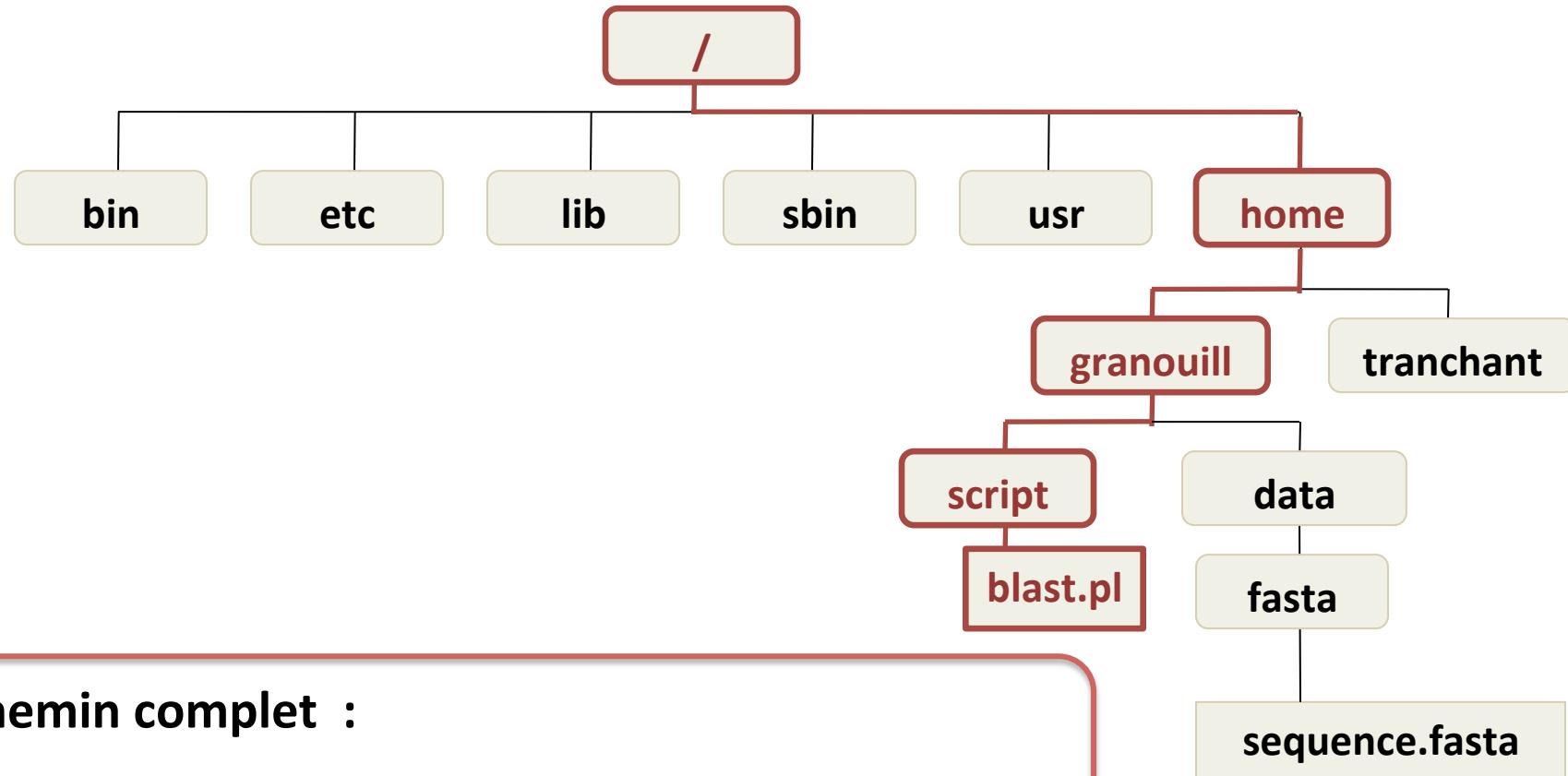


Chemin absolu d'un fichier

chemin complet du fichier en partant du répertoire racine /

↔ **commence toujours par /**

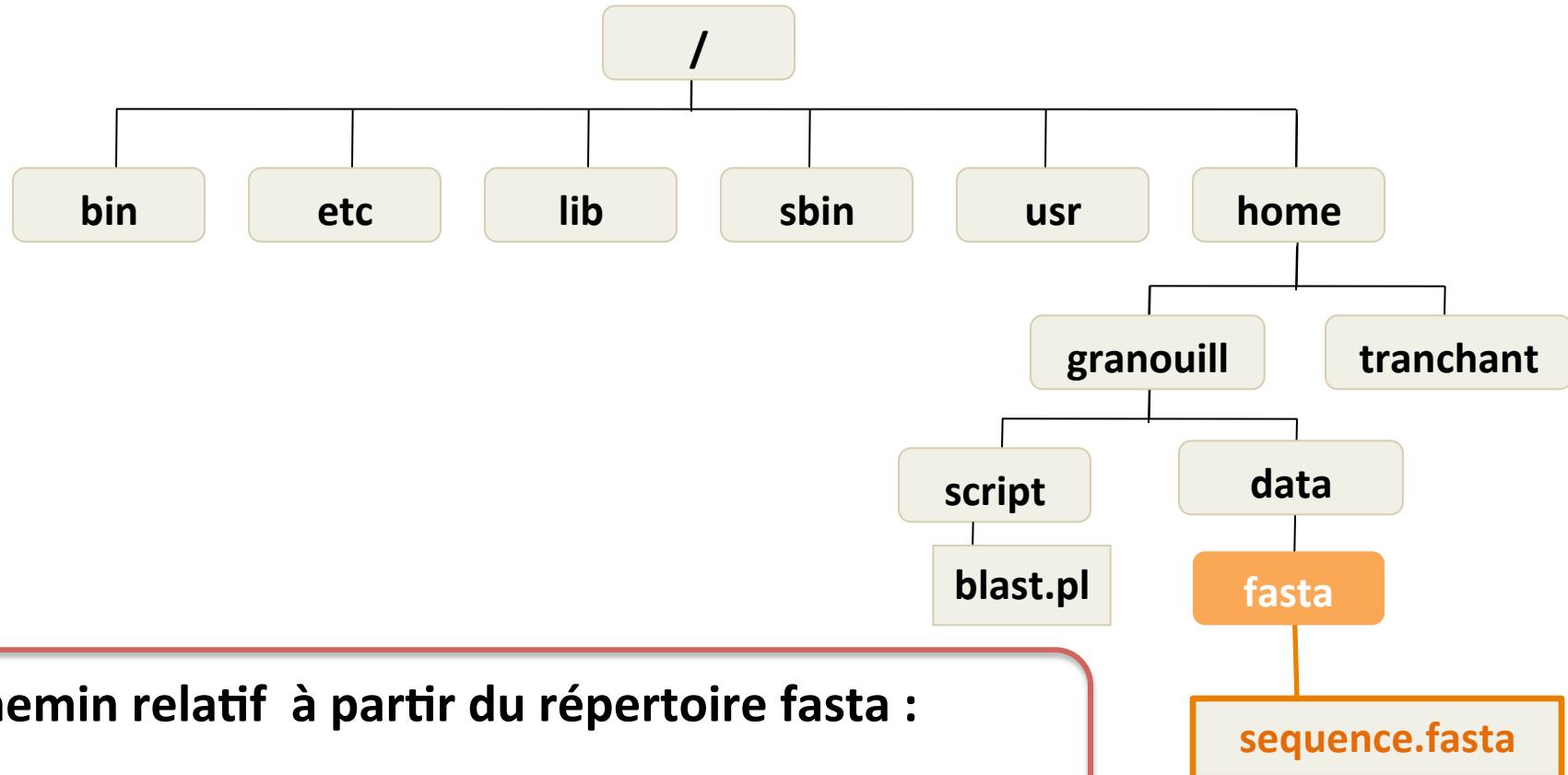
Toujours correct peut importe où on l'on travaille



chemin du fichier défini par rapport où on est dans l'arborescence ⇔

Ne commence jamais par /

Change selon on l'on travaille



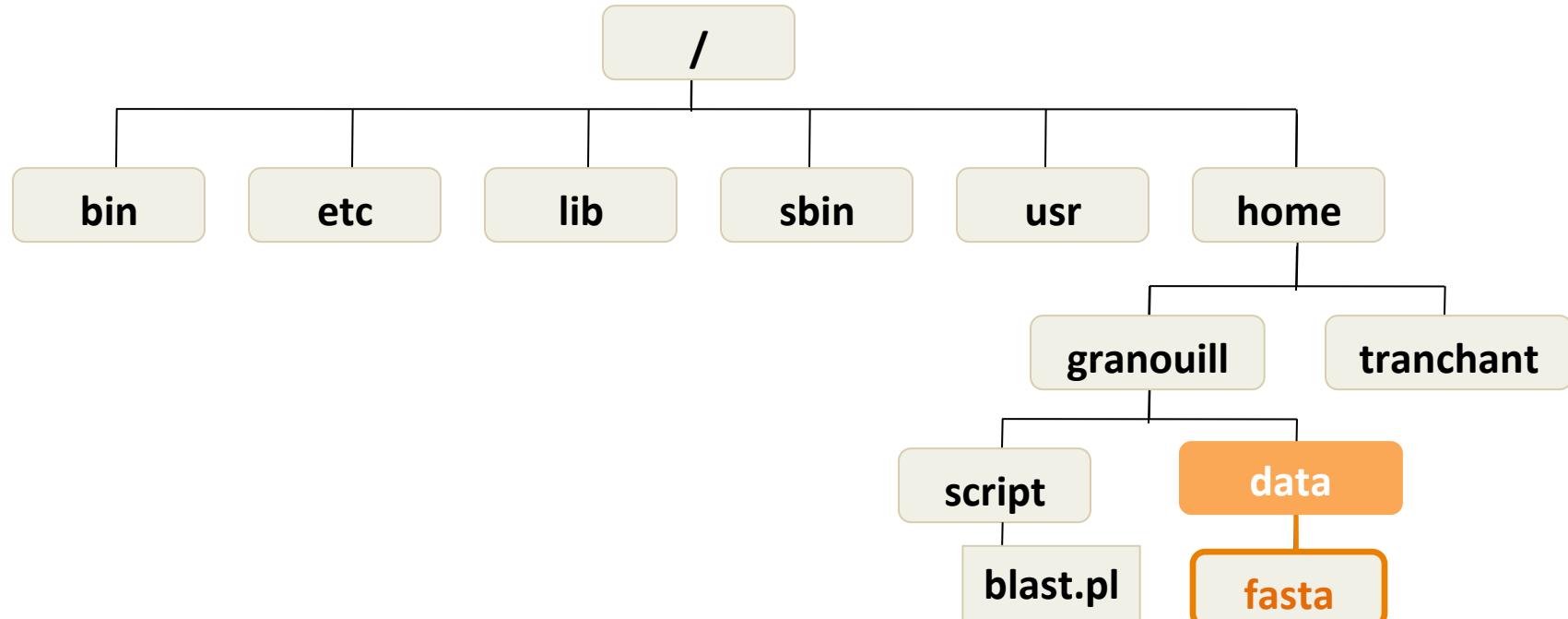
Chemin relatif à partir du répertoire **fasta :**

sequence.fasta

chemin du fichier défini par rapport où on est dans l'arborescence ⇔

Ne commence jamais par /

Change selon on l'on travaille

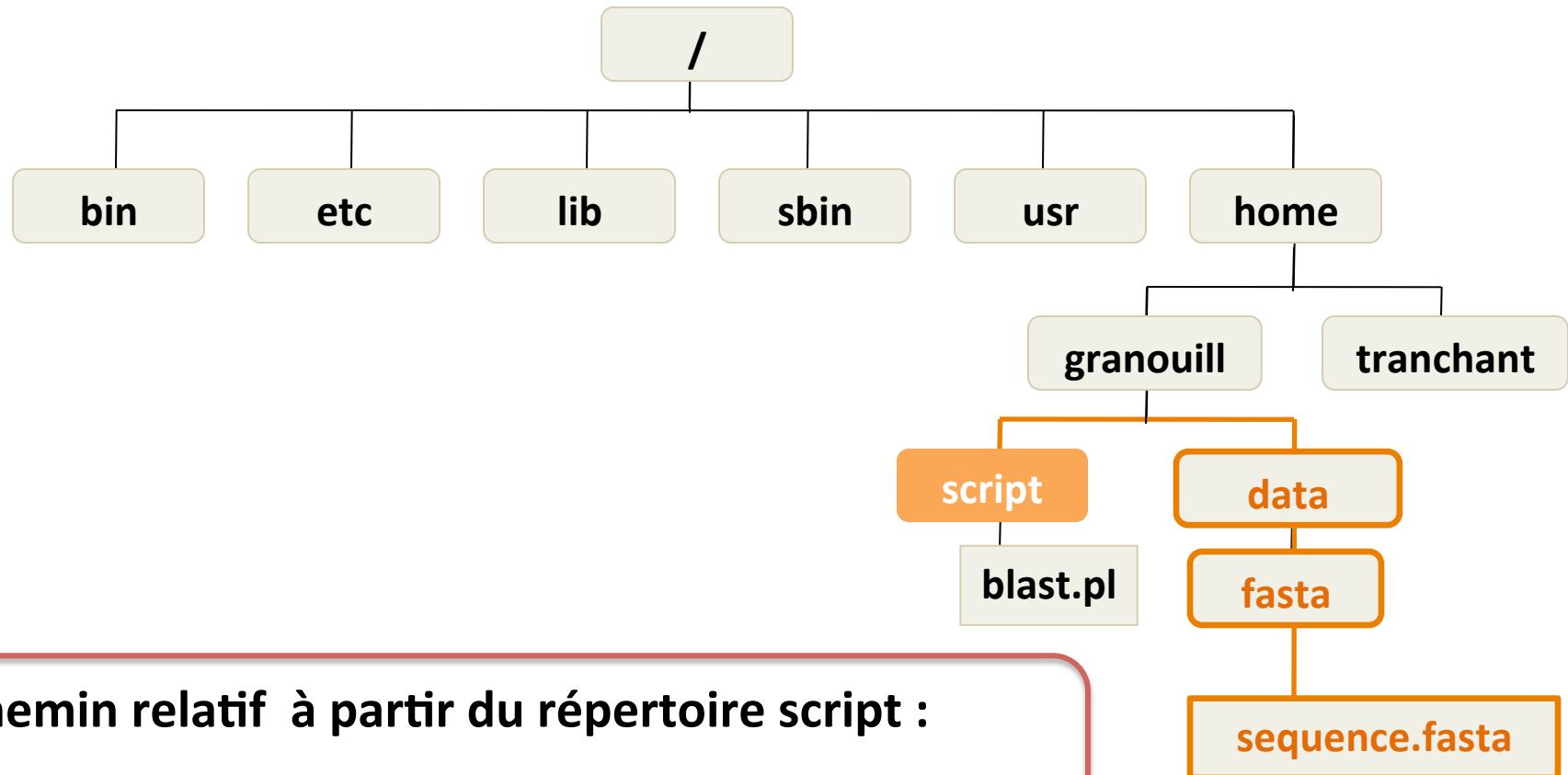


Chemin relatif à partir du répertoire data :

fasta/sequence.fasta

chemin du fichier défini par rapport où on est dans l'arborescence ⇔
Ne commence jamais par /

Change selon on l'on travaille



cd

pour se déplacer dans l'arborescence

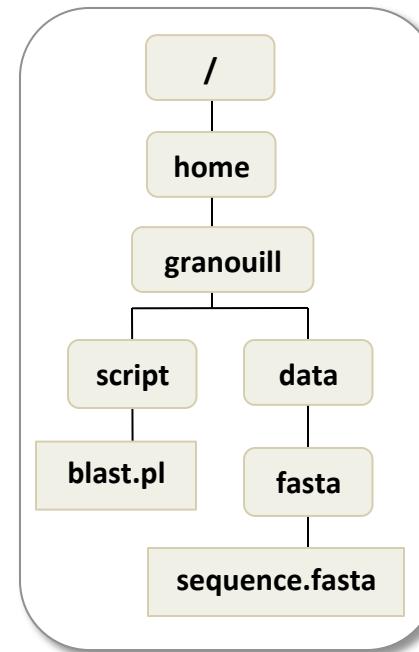
Change Directory

cd nom_repertoire(chemin absolu ou relatif)

cd nom_repertoire(chemin absolu ou relatif)

Chemin absolu :

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



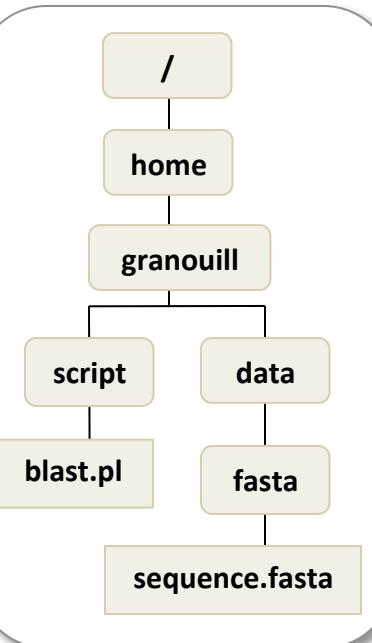
`cd nom_repertoire(chemin absolu ou relatif)`

Chemin absolu :

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

Chemin relatif :

Répertoire courant	Répertoire final	Chemin relatif
granouill	fasta	cd data/fasta
fasta	data	cd ..
fasta	granouill	cd ../../
?	granouill	cd



Remonte d'1 répertoire

Remonte de 2 répertoires

Retour au répertoire de l'utilisateur courant home

Commandes

pwd Affiche le chemin absolu

ls Liste tous les fichiers/répertoires

cd rep_name se déplace dans rep_name

mkdir rep_name Création d'un répertoire

rm nom_fichier Suppression du fichier en argument

cp source target Copie fichier source vers cible

- Linux est sensible à la case
- Utiliser uniquement des lettres, chiffres et caractères _ . -
- PAS d'espaces, accents et caractères spéciaux

Les caractères spéciaux ont des significations particulières

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

- Le suffixe des noms de fichiers (.txt) est optionnel et peut contenir des chiffres ou des lettres



Se déplacer dans l'arborescence linux

~, cd, pwd, ls, . (« point ») et .. (« point point »)



Aller dans le répertoire ~/Script/ et vérifier que vous avez bien changé de répertoire (commande pwd)

Lister le contenu du répertoire

Aller dans ~/Data/Fasta/ en utilisant ../

Revenir dans ~/Data/. Sans se déplacer, regarder le contenu du dossier ~/Data/Autres/

Lister le contenu du répertoire /Data/Sample_Data en utilisant l'option -R. Que contient ce répertoire?

Revenir dans le répertoire principal



Se déplacer dans l'arborescence linux

Commandes mkdir, cp, rm, cd



Créer le répertoire raw dans ~/Data/projetAB.

Déplacer les fichiers .fasta et .qual du répertoire ~/Data/projetAB/ dans le répertoire raw.

Lister le contenu du répertoire raw.

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

ls -l
list long

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

On peut utiliser la commande ls sur un jeu de fichiers dont on ne connaît pas les noms en utilisant les caractères spéciaux (Metacharacters)

```
[form
total
drwx-
drwx-
drwx-
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
```

Quelques caractères spéciaux

*

N'importe quelle chaîne de caractère

[ensemble]

Tous les caractères dans l'ensemble

[!ensemble]

Tous les caractères pas dans l'ensemble

Exemple :



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	

Quelques caractères spéciaux

*

N'importe quelle chaîne de caractère

[ensemble]

Tous les caractères dans l'ensemble

[!ensemble]

Tous les caractères pas dans l'ensemble

Exemple :



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

Quelques caractères spéciaux

*

N'importe quelle chaîne de caractère

[ensemble]

Tous les caractères dans l'ensemble

[!ensemble]

Tous les caractères pas dans l'ensemble

Exemple :



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
```

```
KYVF.sam ZO16.bam  
KYVF.bam
```



TP

Se déplacer dans l'arborescence linux

Commandes cp, mkdir, ls



Lister le contenu du répertoire ~/Data/Sample_Data/t-coffee. Y a t il que des fichiers fasta?

Lister uniquement les fichiers commençant par sample

Lister les fichiers fasta

Créer le répertoire ~/Data/Fasta/T-coffee-bis

Copier les fichiers commençant par sample de type fasta présents dans le répertoire ~/Data/Sample Data/t-coffee dans le répertoire que vous venez de créer



TP

Se déplacer dans l'arborescence linux

Commandes rm, cd



Détruire le fichier Script/string-array.pl

Détruire le répertoire ~/Data/Fasta/T-coffee-bis

Que faut-il faire pour détruire un répertoire ?

Détruire tout ce que contient ~/Data/Fasta/T-coffee-bis/

Détruire le répertoire ~/data/Fasta/T-coffee-bis

cat nom_fichier

Affiche le contenu d'un fichier

```
MacBook-Pro-de-Christine:Data tranchan$ cat Data/Fasta/sequence.fasta
>Gxbjbsjxbjs
CCACCCCTCTTACAGTCTTACCAAATGTCCTTAAACTCCACCTAAAGTATCCAAAGA
CTCGAGAAATGCTGTGCCACAACCAGCTTGAGTCATCCATGACCGTTGATCTCCTT
GCCCCCAGAGTGGGCCTAGCACCATCTAGCTACTACTTGCCCTTCATAACCCATCATTGG
GATAACCCCTGAATACTTATCTATAAGTTCCATATGGCTTATTTCTAAGTAAGAGATGC
ACTTAGTAAGTGCATGTCGTCTGACTGTTACTCTAATGTATGATATTATATCCC
TATAATATAGTGTACTAATATATGTTGGTATTGTGTAGACTCCATTGTACCATGGTGT
GCTAATTAGAAATAACATGCCAGCTTGCTATTGTGGTTGCAAGTAAAGTAAAAAAA
MacBook-Pro-de-Christine:Data tranchan$
```

less nom_fichier

*Affiche le contenu d'un fichier
<space> / q*

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
AATATTGTTGGTGTACCGTTCACCGCTAACATCTTCTGCACAAAGTAAGAGAAGGTAACAAGTAAAAGGG
AGCGAGACTATGGCTGCCACATCTGCTGCAGTACTAAATGGATTGGGCCTCCCTCTTGACTGGTGGAA
GGAAAAGTCAGGCCCTGCTGGCTTACCAAGCTGCAGCTAGAGTCGGTGGTGTGCTGCTGTTGCTCCAAA
GAGATTAGTTGTGGTAGCTGCTCGTCCCTCCAAGAAGTCATGGATTCTGGTTAGAACTGGTCCAGGC
TTCCTTGACCCCGAGTATCTCGATGGCTCGCTCCCTGGTACTACGGTTTGATCCACTGGTCTGGCA
AGGATCCAGCATTCTGAAATGGTATAAGAGAAGCTGAGCTCATTGAGATGGCAATGACTGCAGT
TGTGGAATCTTGTGGCCAGGCATGGAGTGGTATCCCATGGTTCAAGCTGGCCTGACCCCTGGTGCC
ATTGCCATTCTCCTTGGTCTCGCTCGTACTCAACTCCTGCTCA

>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
ATTCATCGTATCTGCCCTTTCATCTCCCTTTGATCAGCATCAGAGGCACCGGCAGC
TTTAAAAGCTTATCACACACACACACATTCTTGACTCAGTCAGTGGCAACAATTACTGAGTACT
ACTAGTTACTTATCGAACAAATTGATTAGCTTGCTAGGAGGTACCGTTGGTTGACTTGTGT
AGTATAAAATTGATGGCTGGGATGCTGTGCGGAGTTAACATTGGAGGCGAGACTGAAACAGCTAAACCA
GTCGAGCCTAATTCTCAGTCGGCTAGGGGAGGAGATGGAAATTCATCACTCATATTGTAGCTTCGG
Data/Fasta/EST-68566-Coffeacanephora.fasta
```

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



exo 7 : Blast de séquences

Suite à un séquençage d'ADNc, nous avons réalisé un assemblage des séquences obtenues.

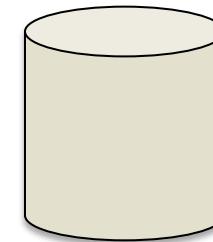
Nous souhaitons savoir si ces séquences correspondent à des séquences de gènes déjà identifiées sur notre modèle le cafier.



exo 7 : Blast de séquences

100 transcrits issus
de l'assemblage

blastn
→



Banque de séquences d'ESTs
(caféier) téléchargées de NCBI



exo 7 : Indexation de la banque blast

makeblastdb pour indexer une banque de séquences nouvelle

Nous avons téléchargé une banque de séquence d'ESTs à partir de NCBI; elle est stockée dans le répertoire ~/Bankcsous le nom de fichier All-EST-coffea.fasta.



makeblastdb -in banque_de_sequence -dbtype nucl ou prot -parse_seqids

Nom de la banque

Type de séquence

Indispensable pour utiliser le programme blastdbcmd



exo 7 : Indexation de la banque blast

makeblastdb pour indexer une banque de séquences nouvelle

Nous avons téléchargé une banque de séquence d'ESTs à partir de NCBI; elle est stockée dans le répertoire ~/Bank sous le nom de fichier All-EST-coffea.fasta.



- Indexer la banque de séquences d'ESTs avec le programme makeblastdb
- Lister le contenu du répertoire pour observer si les nouveaux fichiers index ont bien été créés



exo 7 : Blast de séquences

Blastn

Nous allons réaliser un blast du fichier `~/Data/100_transcrits/100transcrits.fa` contre la banque de séquence d'ESTs nouvellement indexée.



```
blastn -query fichier-fasta -db banque -outfmt [0-11] -out fichier_sortie
```

Fichier fasta avec
les séquences à blaster

Banque de
séquences

6 = format de sortie
tabulaire

Fichier de résultat



exo 7 : Blast de séquences

Blastn

Nous allons réaliser un blast du fichier ~/Data/100_transcrits/100transcrits.fa contre la banque de séquence d'ESTs nouvellement indexée.



Lancer le blast du fichier 100transcrits.fa et sauvegarder les résultats au format tabulé dans le fichier 100_transcrits.fasta.blastn



Exo 8 : Résultat du blast



Bon, le blast est fait!

Regardons ces résultats de plus près!



Exo 8 : Affichage de contenu de fichier

Commandes ls, head, tail, wc

Le fichier 100_transcrits.fasta.blastn a été généré par un blast de 100 transcrits contre une banque de séquence d'EST. Le fichier résultat a un format tabulé de 12 colonnes :

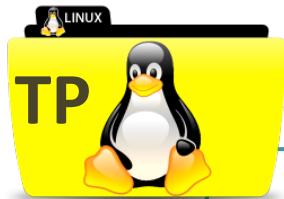


- | | | |
|---------------------|-------------------------------|------------------|
| 1. query id | 5. number of mismatched bases | 9. subject start |
| 2. subject id | 6. number of gap openings | 10. subject end |
| 3. percent identity | 7. query start | 11. expect value |
| 4. alignment length | 8. query end | 12. bit score |



Exo 8 : Affichage de contenu de fichier

Commandes ls, head, tail, wc



- Afficher les 10ères lignes du fichier de résultat blast
- Afficher les 15ères lignes du fichier de résultat blast
- Afficher les 15 dernières lignes du fichier de résultat blast
- Compter le nombre de ligne (résultat)

Une commande très utile...

Grep

grep : Recherche un motif dans un fichier

Syntaxe : grep [options] motif [file1 ...]

Ex : grep "AP1" *fasta

grep : Recherche un motif dans un fichier

Syntaxe : grep [options] motif [file1 ...]

Option	Description
-c	Affiche le nombre de lignes dans lesquelles le motif a été trouvé. Les lignes ne sont pas affichés en sortie.
-l	Affiche seulement le nom des fichiers dans lesquels le motif a été trouvé. Les lignes ne sont pas affichés en sortie.
-i	Recherche insensible à la case (pas de distinction entre minuscules et majuscules)
-v	Affichage toutes les lignes ne contenant pas le motif

grep : Quelques exemples d'utilisation de grep

Exemple

Description

grep "AP1" *fasta Recherche tous les motifs "AP1" dans les fichiers ayant une extension . fasta

grep -c ">" *fasta Compte le nombre de séquences des .fasta

ls -l | grep "^a" Affiche tous les fichiers commençant par 'a'

grep "^[a-d]"
adress_book.txt Affiche toutes les lignes commençant par a,b,c,d



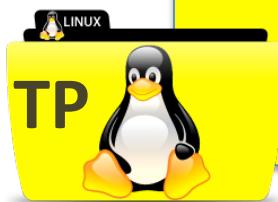
Ex. 10

Aller dans le dossier **Raw**

**Trouver dans quel fichier fasta est la séquence
GLX2YC304JYZ1P**

Visualiser l'un des fichiers fasta

Compter le nombre de séquences par fichier



sort	sort fichier	Trie un fichier
	sort -k2g,2g fichier	
	sort -k2gr,2gr fichier	
	sort -k2g,2g -k1r,1r fichier	
	sort -t: -k3g,3g fichier	

sort

sort fichier

Trie un fichier

sort -k2g,2g fichier

sort -k2gr,2gr fichier

sort -k2g,2g -k1r,1r fichier

sort -t: -k3g,3g fichier

cut

cut -d séparateur -f champs [fichiers]

Sélection de colonnes/champs d'un fichier

cut -d ":" -f1,5 /etc/passwd



Exo 8 : Visualiser le contenu d'un fichier

Commandes sort, cut

Le fichier 100_transcrits.fasta.blastn a été généré par un blast de 100 transcrits contre une banque de séquence d'EST. Le fichier résultat a un format tabulé de 12 colonnes.

Trier les lignes sur le 2nd champ (subject id) croissant puis décroissant

Trier les lignes du fichier de résultat blast sur la e-value (croissant) et longueur de l'alignement (décroissant)

Extraire uniquement les 4 premiers champs du fichier

Extraire l'identifiant des séquences (query et subject), la longueur de l'alignement, l'e-value.

Extraire l'identifiant des ESTs ayant un match avec les 2ères séquences



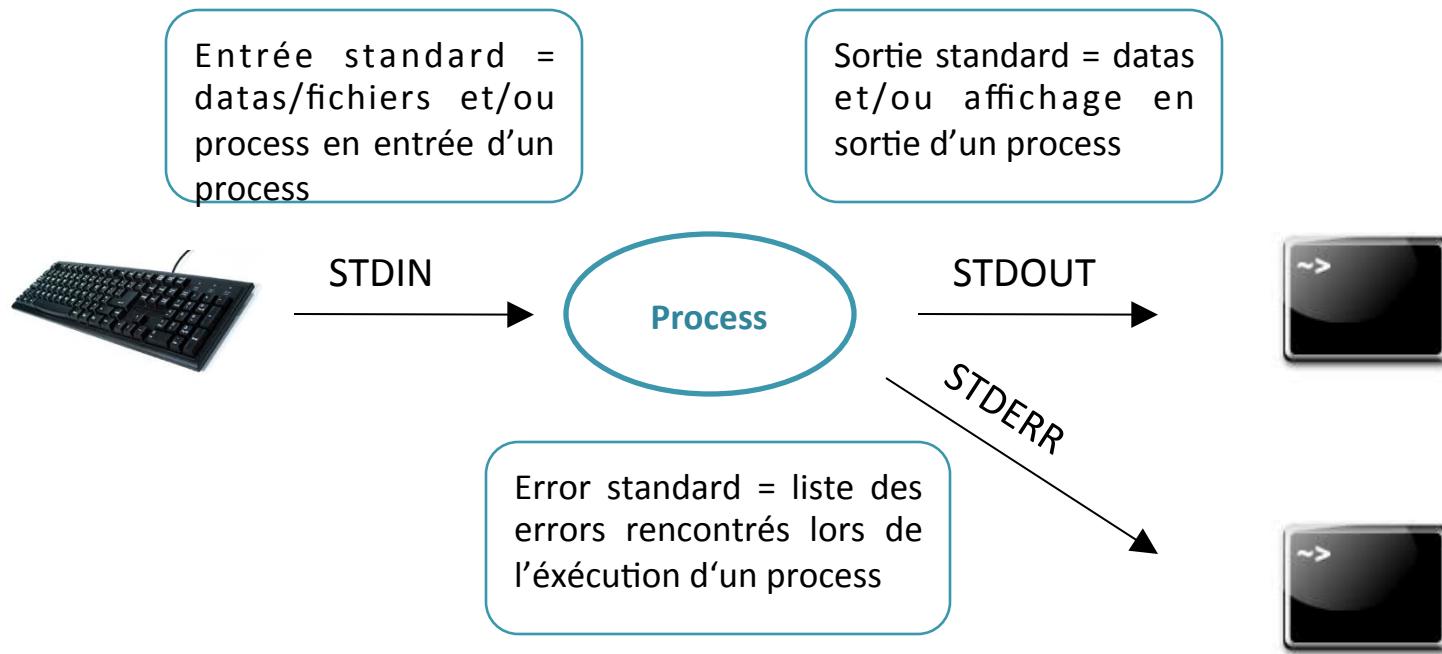
QUELQUES RACCOURCIS CLAVIER BIEN PRATIQUES

- | | |
|---|--|
| <Tab> | Complète automatiquement un nom s'il est unique |
| <Tab><Tab> | Affiche la liste des différentes possibilités si le choix n'est pas unique |
| <  > <  > | permet de passer en revue les commandes utilisées |
| <Ctrl> + r | Rechercher une commande précédemment tapée |
| <Ctrl> + c | Tue le processus en cours dans la console |



Est il possible de sauvegarder la sortie d'une commande dans un fichier au lieu de l'afficher sur l'écran?

Lors d'exécution d'une commande : 3 flux sont ouverts par le Shell



Lors d'exécution d'une commande : 3 flux sont ouverts par le Shell



```
$cut -d: -f1 /etc/passwd > user_name.list
```

Redirection	Action
Command > file	Redirige la sortie vers un nouveau fichier qui sera créé
Command >> file	Redirige la sortie : à la fin d'un fichier si le fichier existe déjà sinon crée un nouveau fichier



Exo 8

Commandes cut, >

Le fichier 100_transcrits.fasta.blastn a été généré par un blast de 100 transcrits contre une banque de séquence d'EST. Le fichier résultat a un format tabulé de 12 colonnes.



Extraire l'identifiant des ESTs ayant un match avec les 2ères séquences

Stocker le résultat de la commande précédente dans le fichier 2_ESTs_accession.list



exo 7 : Blast de séquences

blastdbcmd pour extraire des séquences d'un fichier fasta

```
blastdbcmd -db banque_sequence -entry_batch file_accession -out fichier_sortie
```

Nom de la banque contenant les séquences à extraire

Fichier contenant les accessions des séquences

Fichier fasta



Extraire les séquences fasta des 2 ESTs dont les accessions sont dans le fichier 2_ESTs_accession.list

Lors d'exécution d'une commande : 3 flux sont ouverts par le Shell

Entrée standard =
datas/fichiers et/ou
process en entrée d'un
process



Sortie standard = datas
et/ou affichage en
sortie d'un process



on peut rediriger la sortie
standard en entrée d'une autre
commande (workflow) avec le
caractère | (pipe)

l'exécution d'un process



- . Tube ou Pipe permet d'enchaîner l'exécution de 2 programmes
- . Redirection sans utilisation de fichier intermédiaire
- . Possibilité d'enchaîner les programmes/commandes :

cmd1 | cmd2 | cmd3

Les commandes sont liées avec le “pipe” symbol: | (AltGr+6)

The shell : Redirection tube

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

Root

troot

iroot

ctroot

//

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

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

abate

adm

adroot

ais

#albar

alvaro-wis

anthony

apache



Ex. 9 : Utiliser le pipe

Commandes cut, sort -u



Combien de contigs de C. x ont une homologie avec les ESTs de C. canephora?



Utiliser commande sort avec l'option –u

Les plus intéressantes, puissantes mais complexes aussi :

grep

sed

awk

Les plus intéressantes, puissantes mais complexes aussi :

grep

sed

awk

sed : rechercher et modifier une ligne

Selection de lignes dans un fichier vérifiant une expression régulière
ET applicant une modification ou un traitement

Sed "s/motif recherché/nouveau motif/" file

substitution

motif recherché

nouveau motif

fichier à parser

Sed : Quelques exemples

Example	Description
sed "s/linux/LINUX/" file	Change la 1ère occurrence de “linux” par “LINUX”
sed "s/linux/LINUX/3" file	Change la 3ème occurrence de “linux” par “LINUX”
sed "s/linux/LINUX/g" file	Change toutes occurrences de “linux” par “LINUX”

Sed : Autres exemples avec expression régulière

Example

```
sed s/>/>VS1-/g seq.fasta >  
new_seq.fasta
```

Description

Substitution de > par >VS1- dans toutes les Séquences

```
sed s/\|/-/g contigs_m_f_specif.fasta Substitution de | par -
```



exo 11 : Renommer les séquences

Commandes sed, cat, grep



Avant de générer un fichier unique contenant l'ensemble des séquences, on souhaite « taguer » les séquences par banque de la façon suivante :

File VS.MID1.clean.fasta before

```
>GLX2YC304JWSXM length=420 xy=3947_0232 region=4 run=R_2010_0...
CCACCCCTTTACAGTCTCACCAAATGTCC...
```

File VS.MID1.clean.fasta after

```
>VS1-GLX2YC304JWSXM length=420 xy=3947_0232 region=4 run=R_2010_0...
CCACCCCTTTACAGTCTCACCAAATGTCC....
```

Rq : une fois la commande sed testée sur un fichier, stocker le résultat dans un nouveau fichier nommé *RN-VS.MID1.clean.sff.fasta* ...

-Générer le fichier *RN-VS.ALLMID.clean.sff.fasta* contenant l'ensemble des séquences (MID 1 + MID 2)

- Compter le nombre de séquences dans chaque fichier fasta du répertoire.

Les plus intéressantes, puissantes mais complexes aussi :

grep

sed

awk

awk: Langage pour manipuler un fichier ligne par ligne

- Nom des auteurs : “Aho, Weinberger, and Kernighan”

awk: Langage pour manipuler un fichier ligne par ligne

- Nom des auteurs : “Aho, Weinberger, and Kernighan”
- Un langage de programmation qui permet facilement de manipuler des fichiers tabulés (blast, sam, vcf)et d’extraire une partie des données.
- Un langage utilisé pour rechercher des motifs et pour effectuer des opérations, des actions associées.

awk: Langage pour manipuler un fichier ligne par ligne

- Un langage de programmation qui permet facilement de manipuler des fichiers tabulés (blast, sam, vcf) et d'extraire une partie des données.
- Un langage utilisé pour rechercher des motifs et pour effectuer des opérations, des actions associées.

Principales caractéristiques d'awk

- Pour awk, le fichier en entrée est tabulé
- Comme tout langage de programmation, awk a des variables et peut appliquer des conditions
- awk peut faire des opérations sur les nombres et les chaînes de caractères
- Awk peut générer et afficher des données/rapports suite à des manipulations

awk: Langage pour manipuler un fichier ligne par ligne

Syntax : awk [-F] 'program' file

Option	Description
-F	Donne la nature des séparateurs de champs

awk: Langage pour manipuler un fichier ligne par ligne

Syntax : awk [-F] 'program' file

Option	Description
-F	Donne la nature des séparateurs de champs

Variables prédéfinies utilisées par awk

Variable	Description
\$0	ligne entière
NR	Numéro de la ligne lue
NF	Nombre de champs dans la ligne

awk: Langage pour manipuler un fichier ligne par ligne

awk voit le fichier en entrée comme des enregistrements et des champs

Helene	56	edu	hcyr@sun.com
jean	32	ri	jeanc@inexpress.net
julie	22	adm	juliem@sympatico.ca
michel	24	inf	michel@uqo.ca
richard	25	inf	r�aron@videotron.ca

File: contact.txt

awk: Langage pour manipuler un fichier ligne par ligne

Helene	56	edu	hcyr@sun.com
jean	32	ri	jeanc@inexpress.net
julie	22	adm	juliem@sympatico.ca
michel	24	inf	michel@uqo.ca
richard	25	inf	rcharon@videotron.ca

```
awk '{print $0}' contact.txt
```

```
Helene 56 edu hcyr@sun.com
jean 32 ri jeanc@inexpress.net
julie 22 adm juliem@sympatico.ca
michel 24 inf michel@uqo.ca
richard 25 inf rcharon@videotron.ca
```

Affiche chaque
ligne

awk: Langage pour manipuler un fichier ligne par ligne

Helene	56	edu	hcyr@sun.com
jean	32	ri	jeanc@inexpress.net
julie	22	adm	juliem@sympatico.ca
michel	24	inf	michel@uqo.ca
richard	25	inf	rcharon@videotron.ca

```
$awk '{print NR,$1,$2}' contact.txt
```

```
1 Helene 56
2 jean 32
3 julie 22
4 michel 24
5 richard 25
```

Affiche

le numéro de la ligne lue
Puis le 1^{er} champs
puis le 3^{ème} champs du
fichier tabulé

awk: Langage pour manipuler un fichier ligne par ligne

Helene	56	edu	hcyr@sun.com
jean	32	ri	jeanc@inexpress.net
julie	22	adm	juliem@sympatico.ca
michel	24	inf	michel@uqo.ca
richard	25	inf	r��aron@videotron.ca

```
$awk '{print $1,$2};'  
END { print NR « lignes lues en tout »; } contact.txt
```

Helene 56

Jean 32

Julie 22

Michel 24

Richard É(

5 lignes lues en tout

Instruction ex  cut  e une fois le fichier lu dans son int  gralit  

awk: Langage pour manipuler un fichier ligne par ligne

Helene	56	edu	hcyr@sun.com
jean	32	ri	jeanc@inexpress.net
julie	22	adm	juliem@sympatico.ca
michel	24	inf	michel@uqo.ca
richard	25	inf	rcharon@videotron.ca

```
$awk '{print $1,$3; somme+=$2}'  
END { print « Somme des ages égale à », somme; } contact.txt
```

```
Helene edu  
jean ri  
julie adm  
michel inf  
richard inf  
Somme des ages égale à 159
```

On ajoute l'âge (\$2) à la variable **somme** à chaque ligne lue

Puis on affiche la somme calculée à la fin de la lecture du fichier

awk: Langage pour manipuler un fichier ligne par ligne

Helene	56	edu	hcyr@sun.com
jean	32	ri	jeanc@inexpress.net
julie	22	adm	juliem@sympatico.ca
michel	24	inf	michel@uqo.ca
richard	25	inf	rcharon@videotron.ca

```
$awk '{somme+=$2}'  
END { print « Age moyen = », somme/NR; } contact.txt
```

Age moyen = 31,8

On ajoute l'âge (\$2) à la variable somme à chaque ligne lue

Puis on affiche la moyenne une fois le fichier lu

awk: Langage pour manipuler un fichier ligne par ligne

avec une liste d'instructions et **de conditions aussi**

Condition {Instr-1; Instr-2; ...; Instr-n}

```
awk ' $2>24 && $2<50{ print « Age de », $1, « compris entre 24 et 50, égal à », $2; }' contact.txt
```

L age d Helene est supérieur à 24 et égal à 56
L age d jean est supérieur à 24 et égal à 32
L age d richard est supérieur à 24 et égal à 25

Avec 2 conditions

awk: Langage pour manipuler un fichier ligne par ligne

```
awk '$3 == "inf" {print $0}' ' contact.txt
```

```
michel 24 inf michel@uqo.ca
richard 25 inf rcaron@videotron.ca
```

```
$awk '/j/ {print $0}' ' contact.txt
```

```
jean 32 ri jeanc@inexpress.net
julie 22 adm juliem@sympatico.ca
```

awk: Langage pour manipuler un fichier ligne par ligne

```
awk ' {print $1,$2-10}' contact.txt
```

```
Helene 46
Jean 12
Julie 12
Michel 14
Richard 15
```

```
awk '$2 > 30 && $3 == "ri" {print $0}' contact.txt
```

```
jean 32 ri jeanc@inexpress.net
```

Ces commandes peuvent être utilisées avec en entrée la sortie standard ou un fichier tabulé (comme .gff, fichier blast m8 , .vcf)

Pour linux : La pensée magique ne fonctionne pas

Il faut pratiquer !!!!

Compresser des fichiers - créer une archive .tar:

```
tar -cvf tarfile directory  
gzip file_name
```

Décompresser une archive:

```
gunzip file.gz  
tar -xvf file.tar
```

Afficher le contenu d'une archive:

```
zmore data.txt.gz
```

Rechercher une expression/motif dans une archive:

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

rename – renommer des fichiers

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

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

Type

-Interprétation/Légendes des permissions

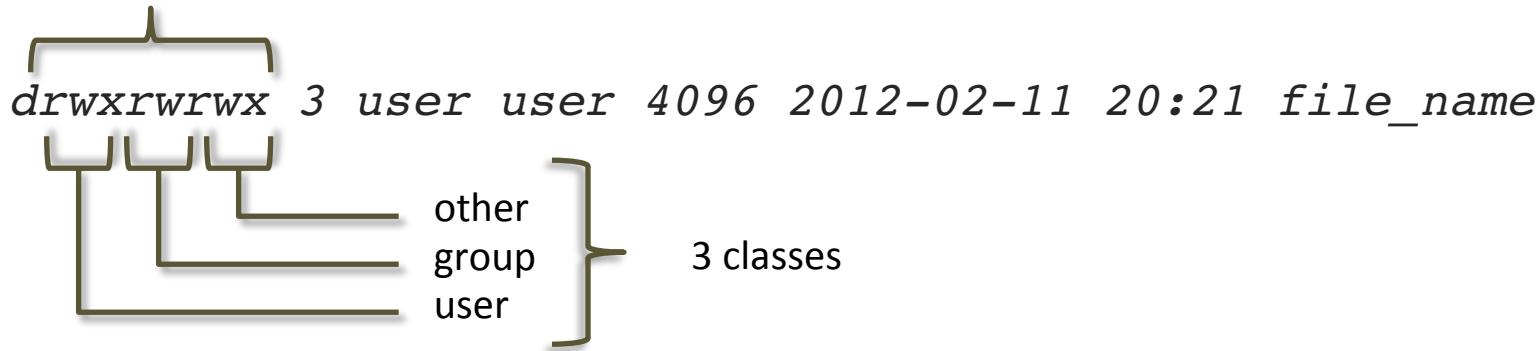
- : fichier normal

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

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

Owner=rwx

Group=r-- Other=---

chmod 755 script.sh

Owner=rwx Group=r-

x Other=r-x



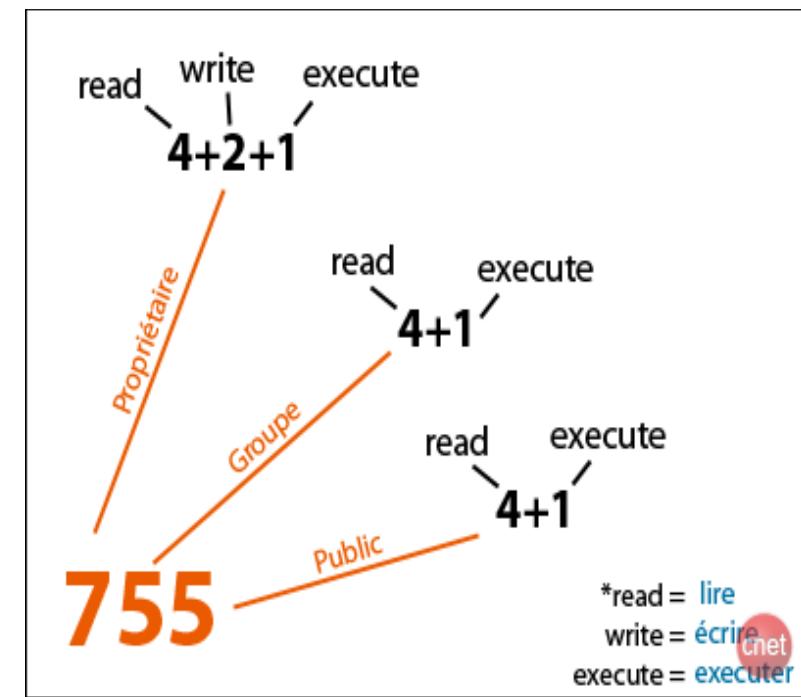
ex 5 : visualiser et changer des permissions

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





Ex. 6 : Manipuler des fichiers

Commandes nano, cat, more, ls

Créer un fichier monfichier.txt contenant une ou deux phrases dans ~/documents/.

Visualiser le contenu de monfichier.txt sans l'éditer.

Quelle est la taille de monfichier.txt ?

Éditez monfichier.txt. Que constatez-vous ?

Afficher le fichier ~/Data/454-projet1/Raw/VS.MID1.clean.sff.fasta page par page

Commande pour créer un fichier >file

Editor de texte (mode Terminal) : **nano**

nano filename

Ctr X : quit & save

Ctr k / ctrl u : copy paste

Ctrl w search

Ctrl Y V : page by page





The shell : other special characters

Characters	Meaning
~	Home directory
#	Comment
\$	Variable
&	Background process
>	Redirection of output
<	Redirection of input
/	Separator of folders in paths

More special characters : * ? () {} [] ; ‘’ !