



Atelier Variant Introduction

Nadia Bessoltane - INRAE
Elodie Girard - Institut Curie
(Maria Bernard - INRAE)
(Olivier Rué - INRAE)

Olivier Quenez - INSERM
Mathieu Charles - INRAE
Odile Rogier - INRAE

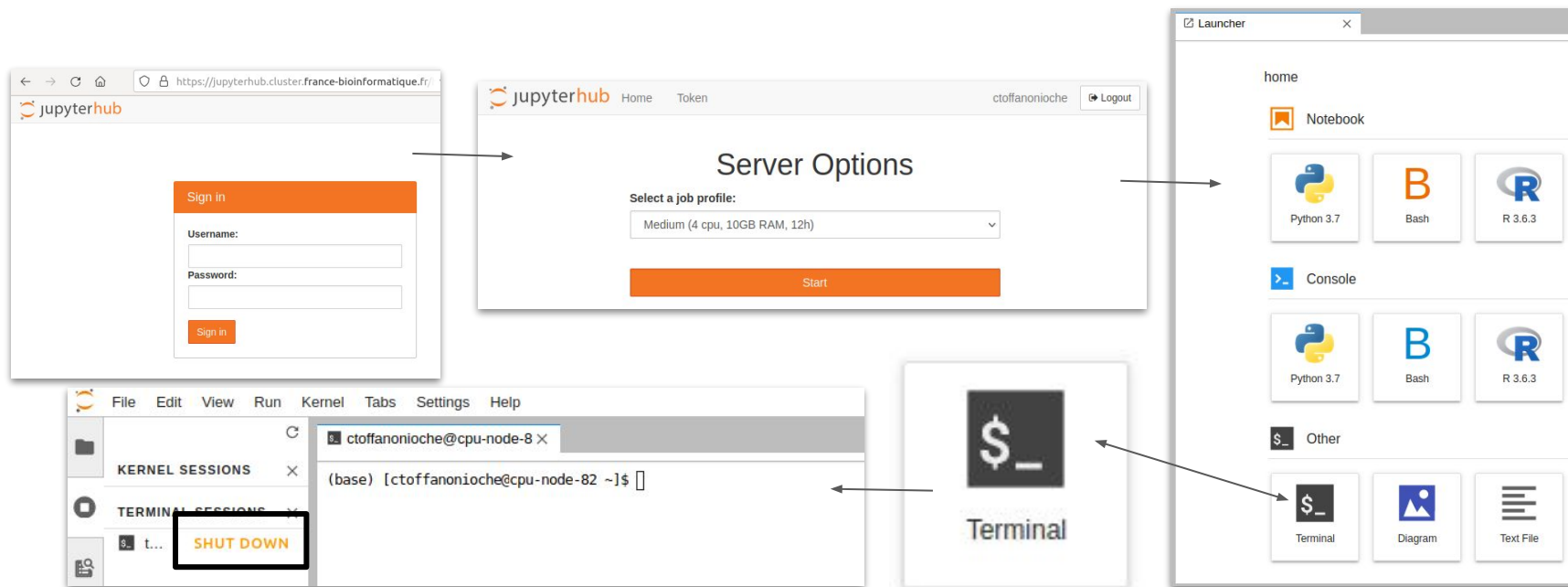
Objectifs

- Introduction, qualité de lectures et alignement (Elodie)
- Utilisation du visualiseur IGV (Olivier Q.)
- Pré- processing des alignements (Nadia)
- SNVs et indels de petite taille, à l'aide de 3 outils : GATK, Mpileup/VarScan et discoSNP (Nadia, Mathieu)
- Introduction à R (Nadia)
- Variations Structurales (SV) (Olivier Q.)
- Utilisation de R pour visualiser des métriques obtenus (Elodie)
- Workflow/Conclusion (Odile)

Cluster de l'IFB

JupyterHub @ IFB

<https://jupyterhub.cluster.france-bioinformatique.fr>



The image illustrates the JupyterHub workflow through five sequential screenshots:

- Sign-in:** A web browser window showing the JupyterHub sign-in page with fields for Username and Password, and a Sign in button.
- Server Options:** A page titled "Server Options" where a job profile is selected (Medium (4 cpu, 10GB RAM, 12h)) and a Start button is visible.
- Launcher:** A "Launcher" window showing the home page with Notebook and Console tabs. Under Notebook, there are icons for Python 3.7, Bash, and R 3.6.3. Under Console, there are icons for Python 3.7, Bash, and R 3.6.3. Under Other, there are icons for Terminal, Diagram, and Text File.
- Terminal:** A terminal window showing the prompt `(base) [ctoffanonioche@cpu-node-82 ~]$`. A "SHUT DOWN" button is highlighted in the bottom left corner.
- Terminal Icon:** A separate icon labeled "Terminal" with a "\$_" symbol.

Cluster de l'IFB



L'Institut Français de Bioinformatique met à disposition de la communauté un cluster de calculs

Your turn! Se connecter au cluster

Sous Windows avec MobaXterm

Session : ssh

Host : core.cluster.france-bioinformatique.fr

Specify username : coché et complété

Sous Mac avec Cyberduck

Open connexion : SFTP

Server : core.cluster.france-bioinformatique.fr

Username/Password : à compléter

Cluster de l'IFB

- **#!/ Connexion initiale : tout le monde sur le noeud maître sur lequel il ne faut pas travailler !/**
- Lancement de “jobs” ou d’une session interactive sur le cluster
- [Vidéo] : [The 5 minutes IFB Core Cluster tutorial](#)

Remember : Tous les jobs doivent être lancés sur un noeud du cluster !

```
# Chargement de l'environnement dédié à chaque outil (exemple pour varscan)
$ module avail -l | grep varscan
$ module load varscan/2.4.3          # ou module load varscan

# Nous aurons besoin au cours du TP de ressources CPU et mémoire (RAM)
$ sbatch --cpus=4 --mem=16G -J toolName_<user_name> --wrap="tool command line"

# Pour suivre vos “jobs” soumis sur le cluster, 2 solutions

$ squeue -u <user_name>                $ scontrol show job <job_id>
```

Jeux de données #1 : SNVs/Indels

Depuis que l'homme fait de l'élevage, il essaie de faire en sorte de toujours améliorer sa **production**, que ce soit en quantité ou en qualité.

Les technologies de génotypage permettent maintenant de **sélectionner les mâles reproducteurs en fonction du fond génétique** qu'ils vont pouvoir transmettre à leur descendance.

Chez le bovin, il existe un locus de caractères quantitatifs (QTL) lié à la production de lait, situé sur le **chromosome 6**, et plus exactement sur une région de 700 kb, composée de 7 gènes.



Jeux de données #1 : SNVs/Indels

Les échantillons **QTL+** sont caractérisés par une diminution de la production en lait et une augmentation des concentrations en protéine et lipide.

Vous aurez à votre disposition :

- Un extrait des données de séquences d'un échantillon du projet 1000 génomes bovins, phénotypé comme **QTL-** : **SRR1262731**
- Les résultats du variant calling pour deux échantillons phénotypés **QTL+** : **SRR1205992** et **SRR1205973**

Your turn !

Quelle mutation est responsable de ce QTL ?

Jeux de données #2 : SVs

Zymoseptoria tritici : Champignon ascomycète, pathogène du blé tendre, responsable d'une maladie foliaire (septoriose).

- Principale maladie du blé (jusqu'à 50% de perte de rendement).
- Haploïde, génome de 40 Mb séquencé en 2011 : 13 chromosomes essentiels + 8 chromosomes accessoires
- Souche séquencée avec **deux technologies** : Illumina et Minlon

Your turn !
Retrouvez les délétions de grande taille



Emplacement des données brutes

- Jeux de données #1 : SNVs/Indels

→

`/shared/projects/form_2021_26/data/atelier_variant/variants`

- Jeux de données #2 : SVs

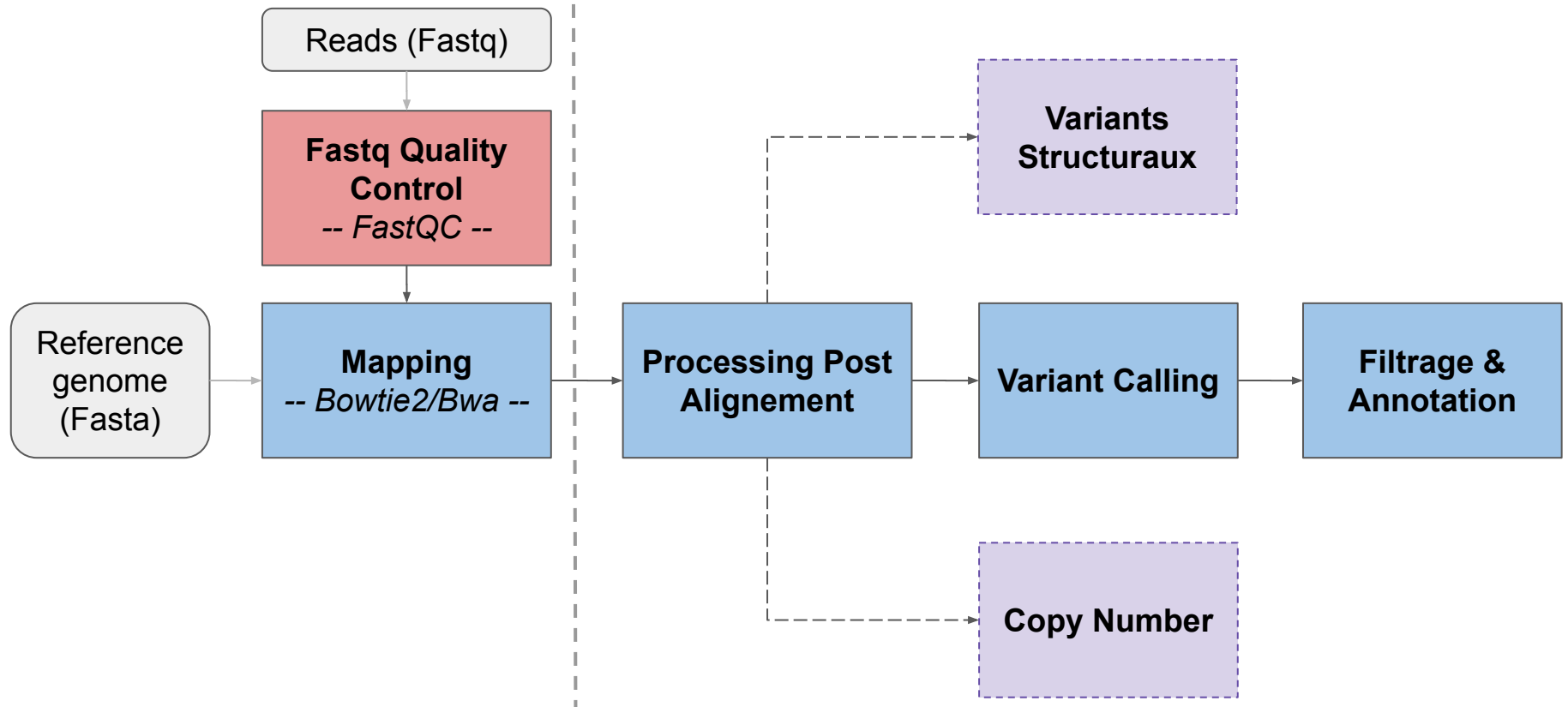
→ `/shared/projects/form_2021_26/data/atelier_variant/sv`

Cheatsheet :

→ Version [html](#) :

`/shared/projects/form_2021_26/data/atelier_variant/EBAII2021_variants.html`

Workflow



Copie du jeu de données #1

```
# Listing des fichiers FASTQ, Genome et BAM
```

```
$ ls -lh /shared/projects/form_2021_26/data/atelier_variant/variants/fastq
```

```
$ ls -lh /shared/projects/form_2021_26/data/atelier_variant/variants/genome
```

```
# Copie des fichiers dans notre home
```

```
$ mkdir -p ~/tp_variant
```

```
$ cp -r /shared/projects/form_2021_26/data/atelier_variant/variants/*  
~/tp_variant
```

```
# Se déplacer dans le dossier optional
```

```
$ mkdir -p ~/tp_variant/optional
```

```
$ cd ~/tp_variant/optional
```

Les lectures (raw reads) au format fastq

Header	@QSEQ32.249996 HWUSI-EAS1691:3:1:17036:13000#0/1 PF=0 length=36
Sequence	GGGGGTCATCATCATTTGATCTGGGAAAGGCTACTG
+ (optional header)	+
Quality	=.+5:<<<<>AA?0A>;A*A#####

```
# Vous pouvez utiliser la commande less pour visualiser le contenu du fichier
# q pour quitter
```

```
$ less -S ~/tp_variant/fastq/SRR1262731_extract_R1.fq.gz
```

Le score de qualité Sanger

- Une valeur de **score Sanger** est attribuée à chaque base séquencée
 - Basée sur p , la probabilité d'erreur (i.e. que la base soit fausse)
 - $Q_{Sanger} = -10 \cdot \log_{10}(p)$
 - $p = 0.1 \Leftrightarrow Q_{Sanger} 10$
 - $p = 0.01 \Leftrightarrow Q_{Sanger} 20$
 - $p = 0.001 \Leftrightarrow Q_{Sanger} 30$
 - ...
- Les scores sont encodés en ASCII 33
 - Objectif : compresser les données en diminuant le nombre de caractères utilisés pour encoder la qualité
- Le score de qualité Sanger varie entre 0 et 40

Le score de qualité Sanger

- ! correspond à 0
- “ correspond à 1
- # correspond à 2
- \$ correspond à 3
- ...
- I correspond à 40

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	Null	32	20	Space	64	40	@	96	60	`
1	01	Start of heading	33	21	!	65	41	A	97	61	a
2	02	Start of text	34	22	"	66	42	B	98	62	b
3	03	End of text	35	23	#	67	43	C	99	63	c
4	04	End of transmit	36	24	\$	68	44	D	100	64	d
5	05	Enquiry	37	25	%	69	45	E	101	65	e
6	06	Acknowledge	38	26	&	70	46	F	102	66	f
7	07	Audible bell	39	27	'	71	47	G	103	67	g
8	08	Backspace	40	28	(72	48	H	104	68	h
9	09	Horizontal tab	41	29)	73	49	I	105	69	i
10	0A	Line feed	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage return	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	47	2F	/	79	4F	O	111	6F	o
16	10	Data link escape	48	30	0	80	50	P	112	70	p
17	11	Device control 1	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	50	32	2	82	52	R	114	72	r
19	13	Device control 3	51	33	3	83	53	S	115	73	s
20	14	Device control 4	52	34	4	84	54	T	116	74	t
21	15	Neg. acknowledge	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	54	36	6	86	56	V	118	76	v
23	17	End trans. block	55	37	7	87	57	W	119	77	w
24	18	Cancel	56	38	8	88	58	X	120	78	x
25	19	End of medium	57	39	9	89	59	Y	121	79	y
26	1A	Substitution	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	59	3B	;	91	5B	[123	7B	{
28	1C	File separator	60	3C	<	92	5C	\	124	7C	
29	1D	Group separator	61	3D	=	93	5D]	125	7D	}
30	1E	Record separator	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	63	3F	?	95	5F	_	127	7F	□

Contrôle qualité des données brutes

```
$ module load fastqc/0.11.9
```

```
$ fastqc --version          # affiche la version (v0.11.9)
```

```
$ fastqc --help             # affiche l'aide
```

```
$ mkdir -p Fastqc/logs
```

```
$ cd ~/tp_variant/optional/Fastqc
```

```
$ sbatch -J FastQC_SRR1262731_R1 -o logs/FastQC_SRR1262731_R1.out -e  
logs/FastQC_SRR1262731_R1.err --cpus-per-task=2 --wrap=" \  
fastqc --threads 2 --outdir . ~/tp_variant/fastq/SRR1262731_extract_R1.fq.gz"
```

```
$ sbatch -J FastQC_SRR1262731_R2 -o logs/FastQC_SRR1262731_R2.out -e  
logs/FastQC_SRR1262731_R2.err --cpus-per-task=2 --wrap=" \  
fastqc --threads 2 --outdir . ~/tp_variant/fastq/SRR1262731_extract_R2.fq.gz"
```

```
# ouvrir les fichiers html via jupyter
```

Trimmer les lectures

- Une étape de **pré-processing**
 - Les reads en entrée sont rognés afin d'éliminer des extrémités de mauvaises qualités
 - En fonction de la capacité de l'outil à faire des alignements locaux ou globaux et de la qualité intrinsèque des données, cette étape peut être cruciale
 - **Risque:** peu de reads alignés
- Quelques logiciels existants
 - Sickle-trim (sliding window-based trimming)
 - FASTX-Toolkit (cut a defined number of nucleotides)
 - Trimmomatic
 - **Cutadapt**

Principe de cutadapt

- **Objectif :**
 - Supprimer les extrémités de mauvaise qualité
- **Solution:**
 - Parcourir le read avec un fenêtrage coulissant de droite à gauche. Calculer la qualité moyenne dans chaque fenêtrage
 - Si la valeur de qualité chute en dessous d'une valeur seuil q , supprimer l'extrémité 3'.
 - Si la taille restante du read est inférieure à une longueur seuil l , supprimer le read.

ACTCGCTCGCTGGTTAATCGATGATCGTGCAGTCGTACTCGTAGCTAGCTAGTCGTAACATAGCTAGTC



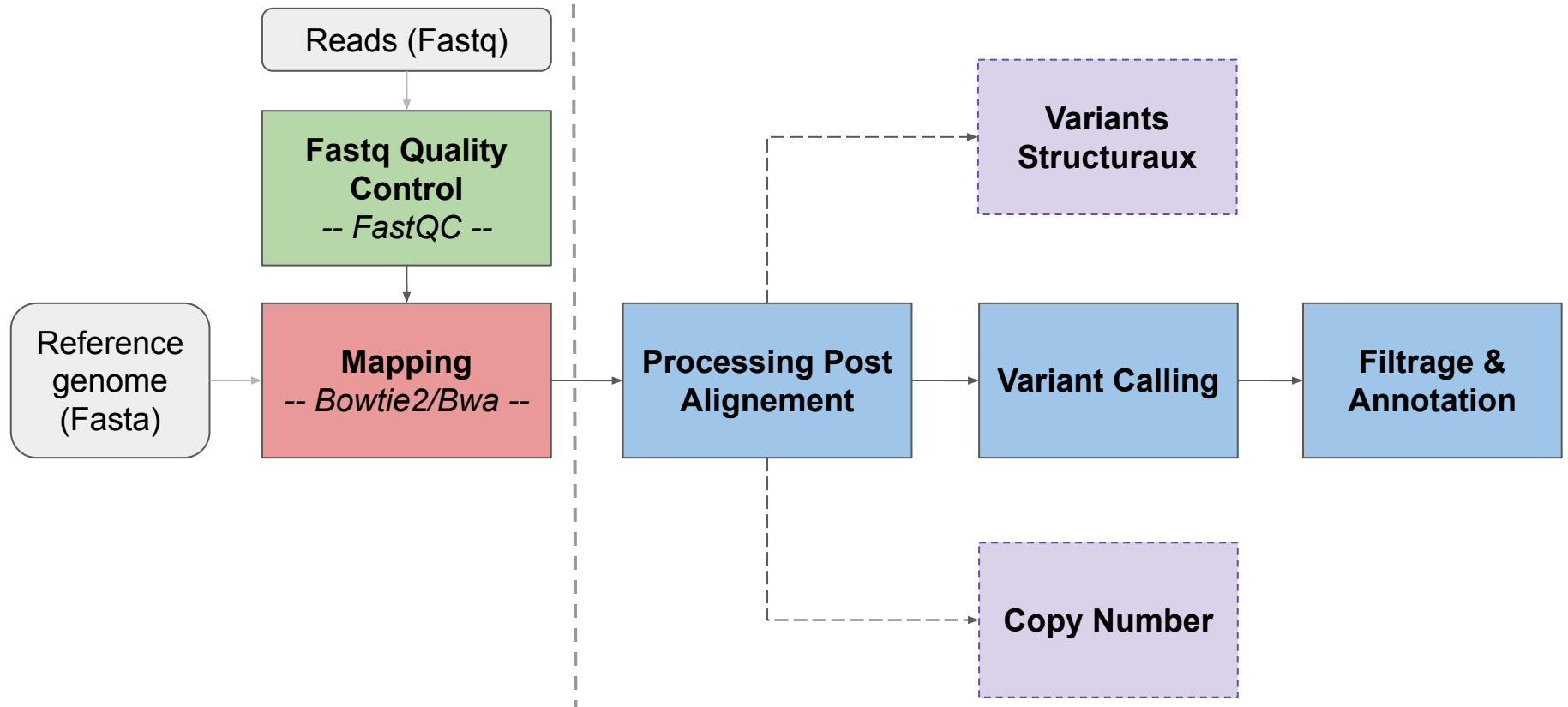
Retrait des séquences de mauvaises qualité

```
$ module load cutadapt/2.10
$ cutadapt --version          # affiche la version (v0.2.10)
$ cutadapt --help             # affiche l'aide

$ mkdir -p ~/tp_variant/optional/Cutadapt/logs
$ cd ~/tp_variant/optional/Cutadapt

$ sbatch -J Cutadapt_SRR1262731 -o logs/Cutadapt_SRR1262731.out -e
logs/Cutadapt_SRR1262731.err --cpus-per-task=2 --wrap=" \
cutadapt --cores 2 --trim-n --max-n 0.3 --error-rate 0.1 -q 30,30 \
--minimum-length 50 --pair-filter both \
--paired-output SRR1262731_extract_R2.trimmed.fq \
--output SRR1262731_extract_R1.trimmed.fq \
~/tp_variant/fastq/SRR1262731_extract_R1.fq.gz \
~/tp_variant/fastq/SRR1262731_extract_R2.fq.gz \
> SRR1262731_extract_trimming_stats.txt"
```

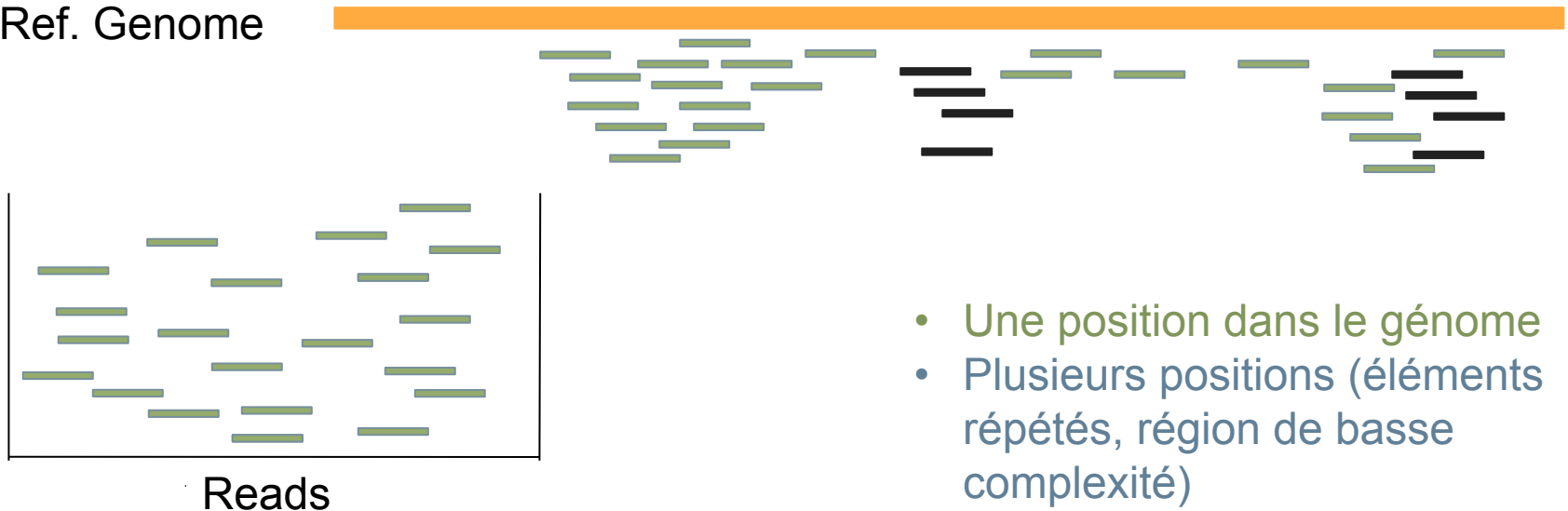
Workflow




Aligner les reads

- Objectif
 - Trouver la région du génome qui a produit les read
 - Trouver dans le génome le mot correspondant au read

Ref. Genome



L'approche *seed & extend*

- Une extrémité du read est interrogée (la graine = the **seed**) 
- On cherche ses régions correspondantes sur le génome (à l'aide d'un **index** créé initialement) avec ou sans mismatch

/!\ Ne faire qu'une seule fois par génome d'intérêt et version majeure /!

- On teste si le reste du read s'aligne avec la séquence
- Les données générées sont au format **SAM** ou **BAM**



L'ajout de *Read Group*

- Associe une identification de provenance à chaque read
 - Utile dans les analyses multi-échantillons ou de reséquençage
- Obligatoire pour utiliser certains outils (comme GATK)
- Un **Read Group** (RG) est défini par :
 - *ID* : Read group **ID**entifier (barcode)
 - *PU* : **P**latform **U**nit
 - *SM* : **S**ample **B**iological **N**aMe
 - *PL* : **P**latform/Technology utilisée (*e.g.*: Illumina)
 - *LB* : préparation de la **L**i**B**rary

Indexation du génome pour BWA

```
$ module load bwa/0.7.17  
$ module load samtools/1.10  
$ module load gatk4/4.1.7.0
```

```
$ cd ~/tp_variant/genome/  
$ mkdir -p logs
```

```
$ sbatch -J BWA_index -o logs/BWA_index.out -e logs/BWA_index.err --wrap="bwa  
index Bos_taurus.UMD3.1.dna.toplevel.6.fa"
```

```
$ sbatch -J samtools_index -o logs/samtools_index.out -e logs/samtools_index.err  
--wrap="samtools faidx Bos_taurus.UMD3.1.dna.toplevel.6.fa"
```

```
$ sbatch -J GATK_index -o logs/GATK_index.out -e logs/GATK_index.err --wrap=" \  
gatk CreateSequenceDictionary --REFERENCE Bos_taurus.UMD3.1.dna.toplevel.6.fa \  
--OUTPUT Bos_taurus.UMD3.1.dna.toplevel.6.dict"
```

Alignement des données

```
$ bwa                # affiche la version et l'aide (v0.7.17-r1188)
$ bwa mem            # affiche l'aide de l'algorithme mem
```

```
$ cd ~/tp_variant/optional/
$ mkdir -p alignment_bwa/logs
$ cd alignment_bwa
```

```
$ sbatch -J SRR1262731_mapping -o logs/SRR1262731_mapping.out -e
logs/SRR1262731_mapping.err --cpus-per-task=4 --mem=16G --wrap=" \
bwa mem -t 4 -R \"@RG\tID:1\tPL:Illumina\tPU:PU\tLB:LB\tSM:SRR1262731\" \
~/tp_variant/genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa \
~/tp_variant/fastq/SRR1262731_extract_R1.fq.gz \
~/tp_variant/fastq/SRR1262731_extract_R2.fq.gz \
| samtools view -Sh - -bo SRR1262731_extract.bam"
```

```
# Visualiser le contenu du BAM
```


Visualiser le contenu du BAM

```
# Visualiser le contenu du BAM
```

```
$ samtools view -h SRR1262731_extract.bam | less -S
```

```
@SQ SN:chr12 LN:133851895
```

```
@RG ID:Sample_ID LB:Sample_Library PL:ILLUMINA SM:Sample_Name PU:Platform_Unit
```

<u>Read name</u>	<u>Flag</u>	<u>Chr</u>	<u>5' pos</u>	<u>MAPQ</u>	<u>Cigar</u>	<u>paired</u>	<u>5' pos of the mate</u>	<u>Insert size</u>
ERR166338.1	99	chr12	82670685	23	101M	=	82670850	266

```
GCCCCTGGGGATGTTTTGCACCAAGCCACTGTCTCCAGCTGG sequence
```

```
BBC@GIIHGCFCIEHEAIEIFFGEONDNJFINIONHNGJNNNNKNJN Base quality
```

```
RG:Z:Sample_ID XT:A:U NM:i:0 X0:i:1 X1:i:1 XM:i:0 XO:i:0 XG:i:0 MD:Z:100 XA:Z tags
```

Group affiliation

Tri et indexage du BAM

```
# On trie le fichier BAM par coordonnées et on crée un index (.bai)
$ sbatch -J SRR1262731_mappingSort -o logs/SRR1262731_mappingSort.out -e
logs/SRR1262731_mappingSort.err --cpus-per-task=4 --mem=16G --wrap=" \
samtools sort -@ 4 --write-index \
-o SRR1262731_extract.sort.bam##idx##SRR1262731_extract.sort.bam.bai \
SRR1262731_extract.bam"
```

```
# On produit les statistiques d'alignement
$ sbatch -J SRR1262731_flagstat -o logs/SRR1262731_flagstat.out -e
logs/SRR1262731_flagstat.err --wrap=" \
samtools flagstat SRR1262731_extract.sort.bam > SRR1262731.flagstat.txt"
```

```
$ cat SRR1262731.flagstat.txt
```

```
[egirard@clust-slurm-client alignment_bwa]$ cat SRR1262731.flagstat.txt
2265873 + 0 in total (QC-passed reads + QC-failed reads)
0 + 0 secondary
46487 + 0 supplementary
0 + 0 duplicates
1700879 + 0 mapped (75.07% : N/A)
2219386 + 0 paired in sequencing
1109693 + 0 read1
1109693 + 0 read2
621472 + 0 properly paired (28.00% : N/A)
1229358 + 0 with itself and mate mapped
425034 + 0 singletons (19.15% : N/A)
0 + 0 with mate mapped to a different chr
0 + 0 with mate mapped to a different chr (mapQ>=5)
```