Atelier Variant

Nadia Bessoltane - INRAE
Vivien Deshaies - AP-HP

École de bioinformatique AVIESAN-IFB-INSERM 2022

Programme de l’atelier Variants

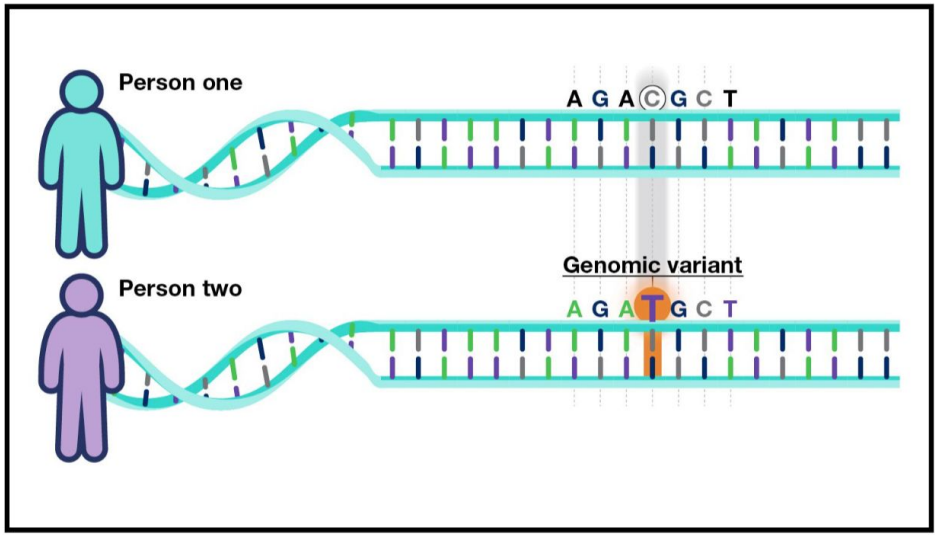
- Détection des petites variations génomiques
- Détection des variations structurales
- Manipulation des variants avec R
- Ecrire un script automatique

Introduction

Définition

1. Qu'est ce que c'est une variation génomique ?

Une variation génomique est un changement, d'une ou plusieurs bases nucléotides, dans une séquence d'ADN particulière en comparaison avec une séquence d'ADN (un génome) de référence (1). Les variations génomiques se distinguent en deux catégories : polymorphismes et mutations.



Il existe différents types de variations :

- **SNV** : **S**ingle **N**ucleotide **V**ariant
- **INDEL** : **IN**sertion ou **DEL**etion
- **SV** (**S**tructural **V**ariant)
- **CNV** (**C**opy **N**umber **V**ariation)

https://www.genome.gov/Health/Genomics-and-Medicine/Polygenic-risk-scores#one

Définition

Variant : variation génomique dans une séquence nucléotidique, en comparaison avec une séquence de référence

- **SNV** : **S**ingle **N**ucleotide **V**ariant
- **INDEL** : **IN**sertion ou **DEL**etion d'une ou plusieurs bases
- **MNV** (**M**ulti-**N**ucleotide **V**ariant) : plusieurs SNVs et/ou INDELS dans un bloc
- **SV** (**S**tructural **V**ariant) : réarrangement génomique affectant > 50bp

SNV ≠ SNP

- **SNV** (**S**ingle **N**ucleotide **V**ariant)

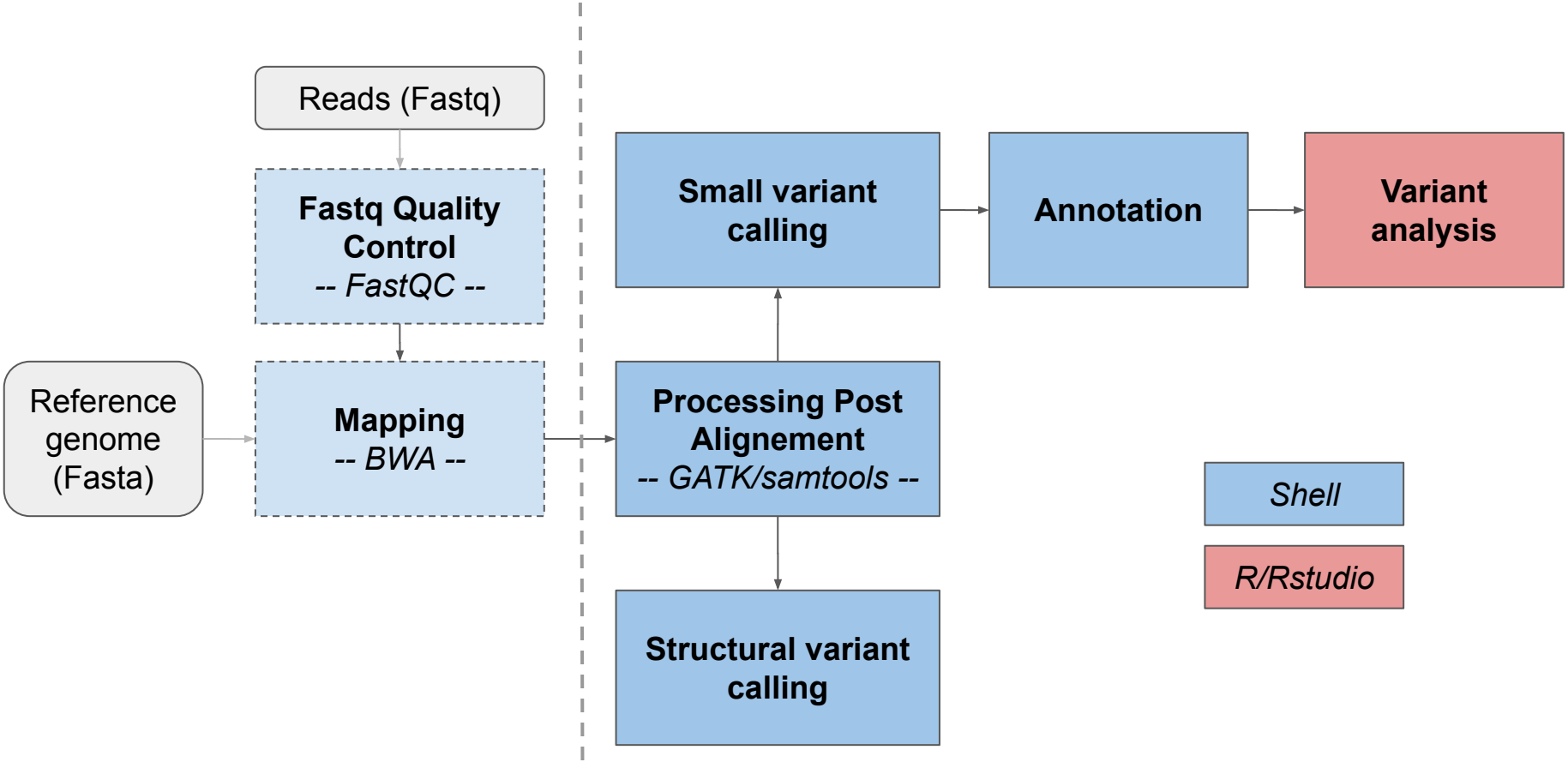
  → toute altération nucléotidique sans implication de fréquence populationnelle

- **SNP** (**S**ingle **N**ucleotide **P**olymorphism)

  → implique qu'un variant est partagée dans la population (> 1%)

/!\ l'amalgame SNPs est souvent fait pour qualifier les SNVs /!\

Workflow


Reads (Fastq)
Fastq Quality Control
-- FastQC --
Reference genome (Fasta)
Mapping
-- BWA --
Processing Post Alignement
-- GATK/samtools --
Small variant calling
Annotation
Variant analysis
Structural variant calling
Shell
R/Rstudio

Détection des petites variations génomiques

Vivien Deshaies - AP-HP

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 ?

Miri Cohen-Zinder et al, Genome Research 2005, DOI:10.1101/gr.3806705

Emplacement des données brutes

- Jeux de données #1 : SNVs/Indels

→ `/shared/projects/form_2022_32/atelier_variant/variants`

Cheatsheet :

→ Version **html** :

`/shared/projects/form_2022_32/atelier_variant/EBAII2021_variants.html`

Copie du jeu de données #1

```
# Listing des fichiers FASTQ, Genome et BAM
$ ls -lh /shared/projects/form_2022_32/atelier_variant/variants/fastq
$ ls -lh /shared/projects/form_2022_32/atelier_variant/variants/genome
```

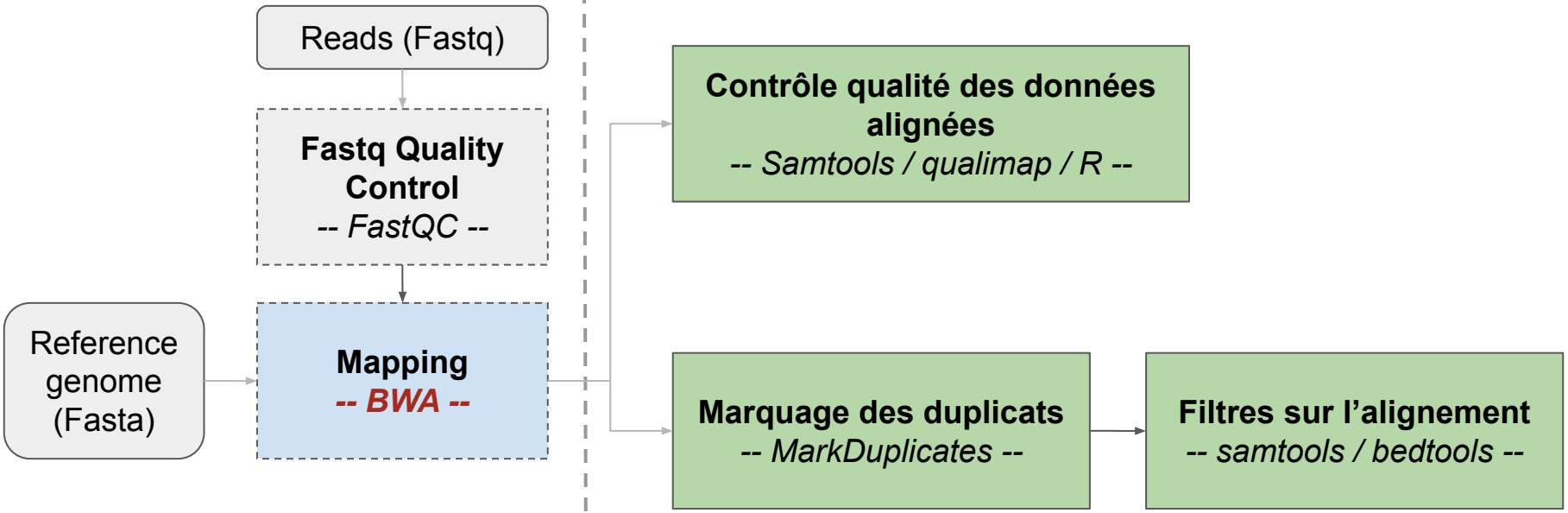
```
# Copie des fichiers dans notre home
$ mkdir -p ~/tp_variant
$ cp -r /shared/projects/form_2022_32/atelier_variant/variants/* ~/tp_variant/
$ ls -l
```

Détection des petites variations génomiques

Alignement et post-processing

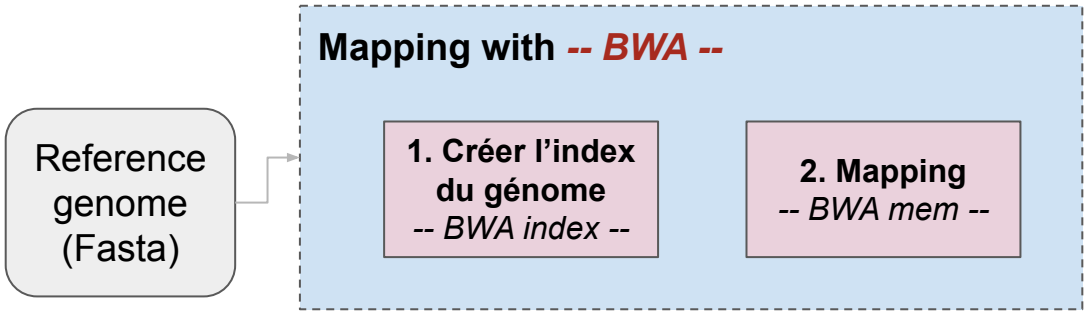
Workflow - Alignement & Processing Post Alignement

- Nécessité de préparer les données avant la détection des variants



Alignement des données avec l'outil BWA-mem

```
$ # charger BWA
$ module load bwa/0.7.17
$ bwa              # affiche la version et l'aide (v0.7.17-r1188)
```



Alignement des données avec l'outil BWA-mem

1/2 - Créer l'index du génome pour BWA

```
$ bwa index                  # affiche l'aide de l'algorithme index
$
$ cd ~/tp_variant/genome/ # aller dans le dossier genome
$ ls -l                      # voir le contenu du dossier
```

```
$ # créer les index : bwa index <fasta>
$ bwa index Bos_taurus.UMD3.1.dna.toplevel.6.fa
$
$ # voir le contenu dossier
$ ls -l
```

Alignement des données avec l'outil BWA-mem
2/2 - Mapping

```
$ bwa mem            # affiche l'aide de l'algorithme mem
$ cd ~/tp_variant/
```

```
$ # Exécuter l'alignement (bwa mem -t 4 -R <readGroup> genome fastq1 fastq2 > sam)
$ bwa mem -t 4 -R "@RG\tID:1\tPL:Illumina\tPU:PU\tLB:LB\tSM:SRR1262731" \
  genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa \
  fastq/SRR1262731_extract_R1.fq.gz \
  fastq/SRR1262731_extract_R2.fq.gz > SRR1262731_extract.sam
```

Alignement des données avec l'outil BWA-mem

2/2 - Mapping : trier et indexer l'alignement

```
$ module load samtools/1.13
$
$ # convertir le sam en bam
$ samtools view -Sh -bo SRR1262731_extract.bam SRR1262731_extract.sam
$
$ # On trie le fichier BAM par coordonnées
$ samtools sort -@ 4 -o SRR1262731_extract.sort.bam SRR1262731_extract.bam
$ # et on crée un index (.bai)
$ samtools index SRR1262731_extract.sort.bam
```

```
$ # Visualiser le contenu du BAM
$ samtools view -h SRR1262731_extract.bam | less -S
$ # supprimer le sam pour libérer de l'espace
$ rm SRR1262731_extract.sam
```

Ajout de la provenance des échantillons

- ReadGRoups (RG) : associe des **informations sur la provenance des reads**
  → Identité : run/échantillon
  → Séquençage, librairie…
- Nécessaire à la recherche de variants
- Plus d'informations

```
Mom's data:
@RG     ID:FLOWCELL1.LANE5      PL:ILLUMINA     LB:LIB-MOM-1 SM:MOM
@RG     ID:FLOWCELL1.LANE6      PL:ILLUMINA     LB:LIB-MOM-1 SM:MOM
@RG     ID:FLOWCELL1.LANE7      PL:ILLUMINA     LB:LIB-MOM-2 SM:MOM
@RG     ID:FLOWCELL1.LANE8      PL:ILLUMINA     LB:LIB-MOM-2 SM:MOM

Kid's data:
@RG     ID:FLOWCELL2.LANE1      PL:ILLUMINA     LB:LIB-KID-1 SM:KID
@RG     ID:FLOWCELL2.LANE2      PL:ILLUMINA     LB:LIB-KID-1 SM:KID
@RG     ID:FLOWCELL2.LANE3      PL:ILLUMINA     LB:LIB-KID-2 SM:KID
@RG     ID:FLOWCELL2.LANE4      PL:ILLUMINA     LB:LIB-KID-2 SM:KID
```

- Comment vérifier la présence de ReadGroups dans un fichier BAM?

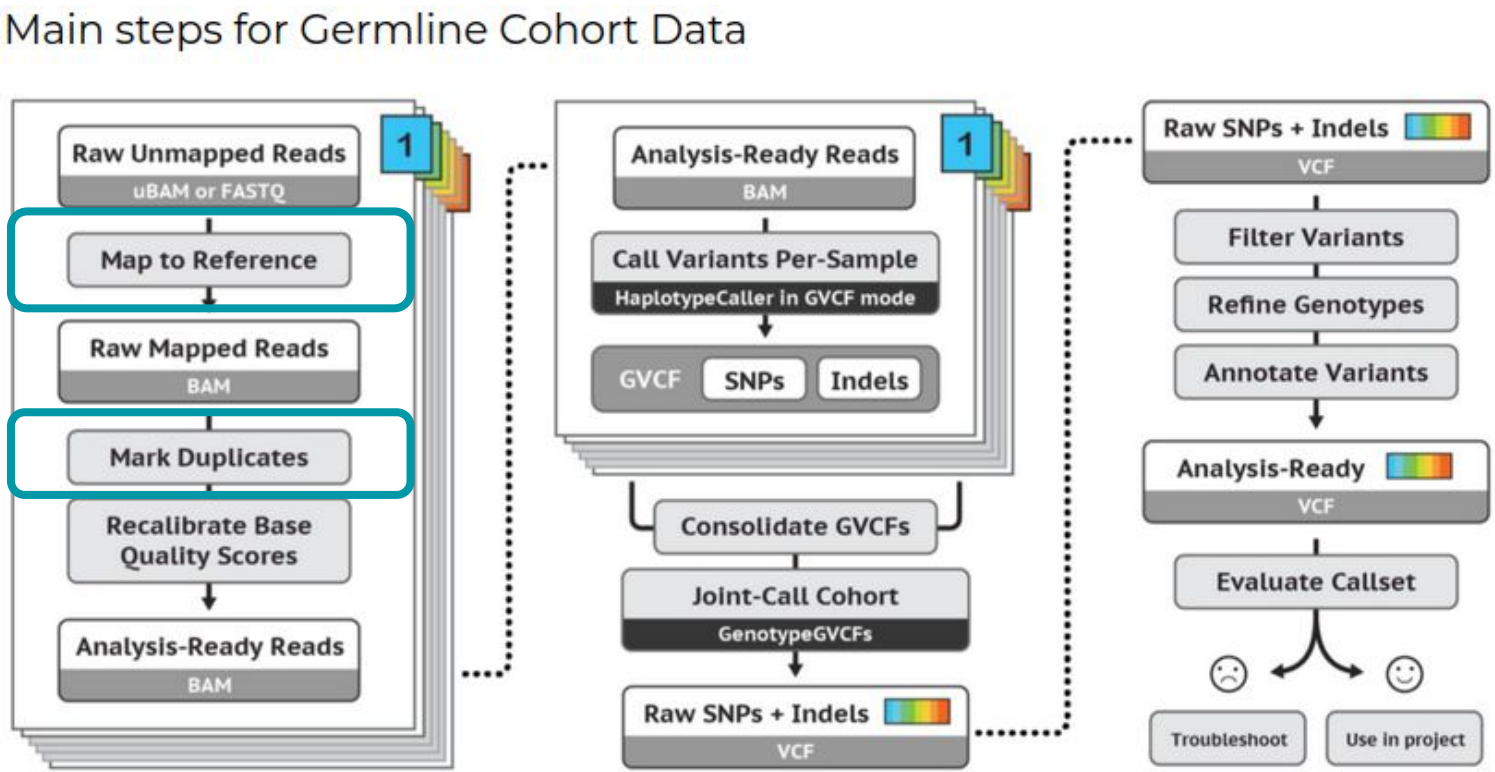
```
$ samtools view -H SRR1262731_extract.bam | grep "^@RG"
```

- Avec l'outil **AddOrReplaceReadGroups** de la suite **PicardTools** intégrée à **GATK4**

```
$ module load gatk4/4.2.3.0
$ gatk AddOrReplaceReadGroups --help
```

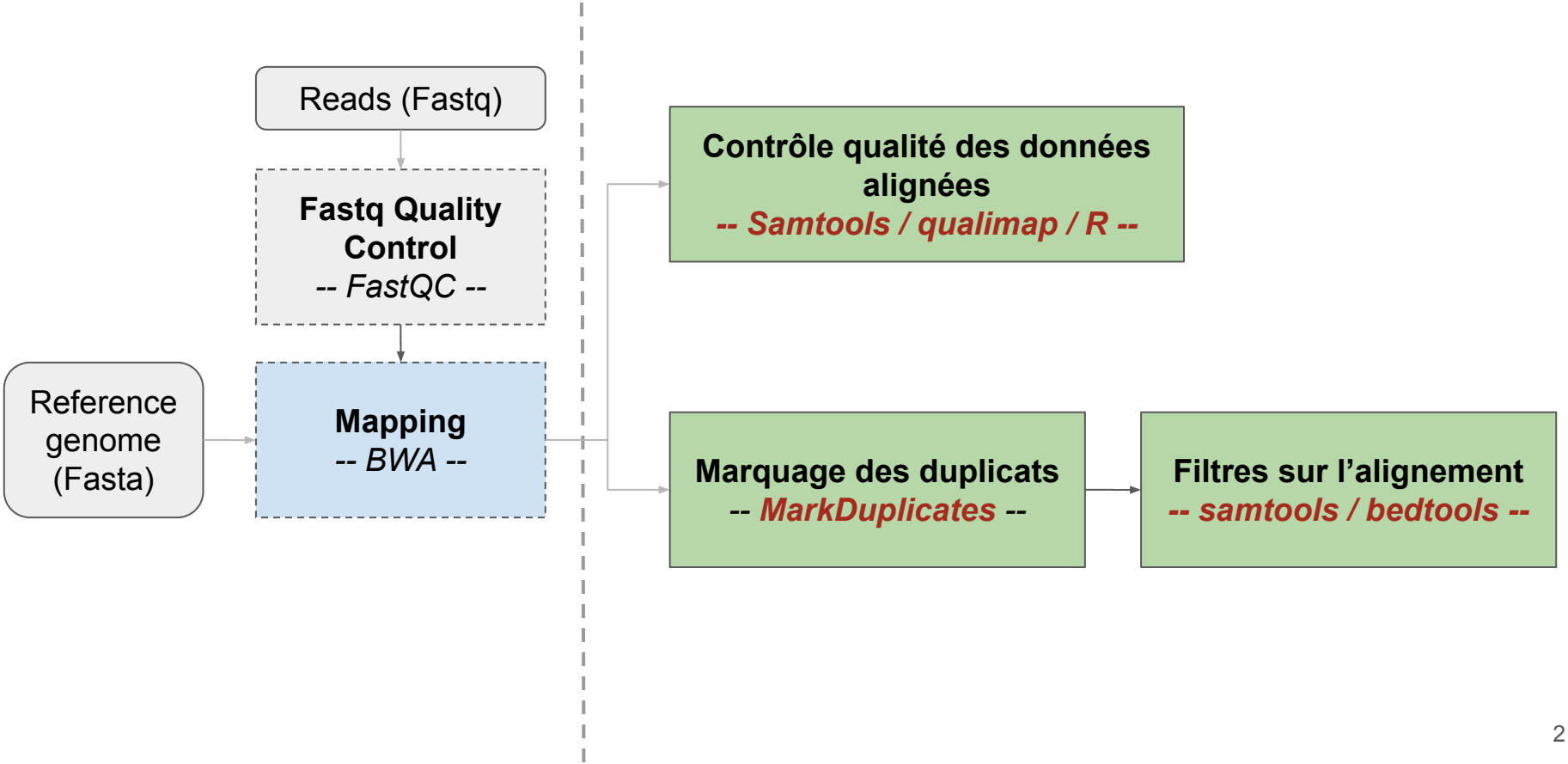
Workflow - Processing Post Alignement

- Nécessité de préparer les données avant la détection des variants



Workflow - Alignement & Processing Post Alignement

- Nécessité de préparer les données avant la détection des variants



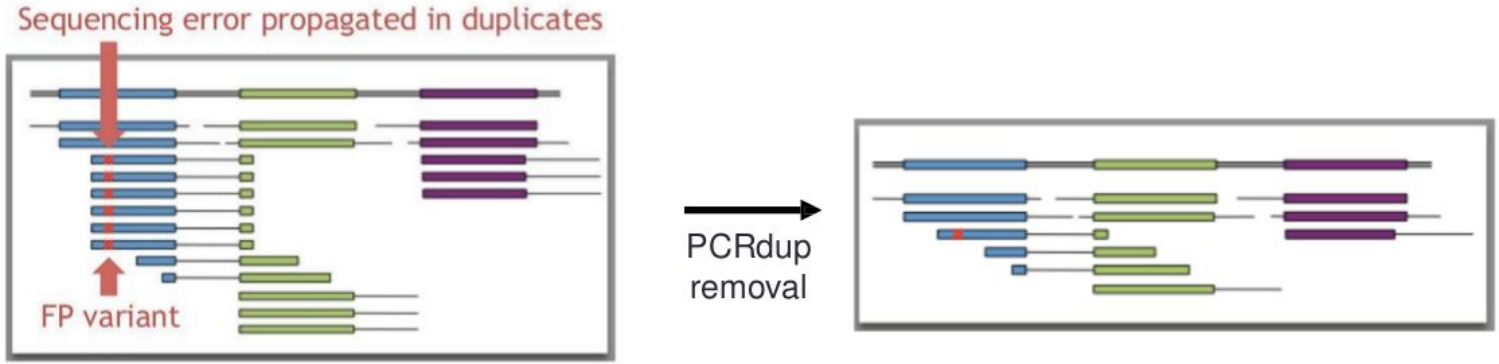
Indexation du génome pour BWA?

```
$ module load bwa/0.7.17
$ module load samtools/1.13
$ module load gatk4/4.2.3.0
```

```
$ # se déplacer dans le dossier genome
$ cd ~/tp_variant/genome/
$ # se déplacer dans le dossier genome
$ mkdir -p logs

$ bwa index Bos_taurus.UMD3.1.dna.toplevel.6.fa

$ samtools faidx Bos_taurus.UMD3.1.dna.toplevel.6.fa

$ gatk CreateSequenceDictionary --REFERENCE Bos_taurus.UMD3.1.dna.toplevel.6.fa
--OUTPUT Bos_taurus.UMD3.1.dna.toplevel.6.dict
```

Marquage des duplicats de PCR

- Identifier les **reads provenant d'une même molécule** issus de :
  → PCR duplicates : amplification PCR durant la préparation de la librairie
  → Optical duplicates : cluster illumina identifié comme deux clusters



Marquage des duplicats de PCR

→ Garder les duplicats : probabilité importante de confondre les duplicats avec des fragments biologiques issus du même locus
→ Marquer les duplicats mais les conserver dans le fichier BAM
→ Supprimer les duplicats du fichier BAM : certains outils les supprimeront par défaut (samtools, GATK…)

Avec l'outil **MarkDuplicates** de la suite **PicardTools** intégrée à la suite **GATK4**

```
$ module load gatk4
$ gatk MarkDuplicates --help          # affiche l'aide

$ gatk MarkDuplicates --java-options '-Xmx8G' \
    -I SRR1262731_extract.sort.bam --VALIDATION_STRINGENCY SILENT \
    -O SRR1262731_extract.sort.md.bam -M SRR1262731_extract_metrics_md.txt
```

Marquage des duplicats de PCR

→ Garder les duplicats : probabilité importante de confondre les duplicats avec des fragments biologiques issus du même locus
→ Marquer les duplicats mais les conserver dans le fichier BAM
→ Supprimer les duplicats du fichier BAM : certains outils les supprimeront par défaut (samtools, GATK…)

```
$ samtools flagstat SRR1262731_extract.sort.md.bam \
    > SRR1262731_extract.md.flagstat.txt

$ cat SRR1262731_extract.md.flagstat.txt                    # nombre de duplicats
$ grep -A1 "LIBRARY" SRR1262731_extract_metrics_md.txt      # % de pcrDup

# Bonus
$ grep -A1 "LIBRARY" SRR1262731_extract_metrics_md.txt | awk
'NR==2{printf("%.2f\n",$(NF-1)*100)}'
```

Filtres sur les alignements

Restreindre le fichier BAM en fonction de métriques d'alignements :

- qualité de mapping (MAPQ) suffisante
- retrait des reads non mappés

```
# Suppression des reads non mappés et filtre sur les reads avec MAPQ < 30
$ samtools view -bh -F 4 -q 30 SRR1262731_extract.sort.md.bam \
    > SRR1262731_extract.sort.md.filt.bam
```

Pour utiliser le paramètre -F : plus d'information sur les SAM Flags

```
$ samtools flagstat SRR1262731_extract.sort.md.filt.bam \
    > SRR1262731_extract.filt.flagstat.txt

$ cat SRR1262731_extract.filt.flagstat.txt
```

Filtres sur les alignements

Restreindre le fichier BAM en fonction de métriques d'alignements :

- alignements intersectant les régions d'intérêt
- en fonction du nombre de mismatchs, de la taille d'insert, de paires mappées sur des chromosomes différents…

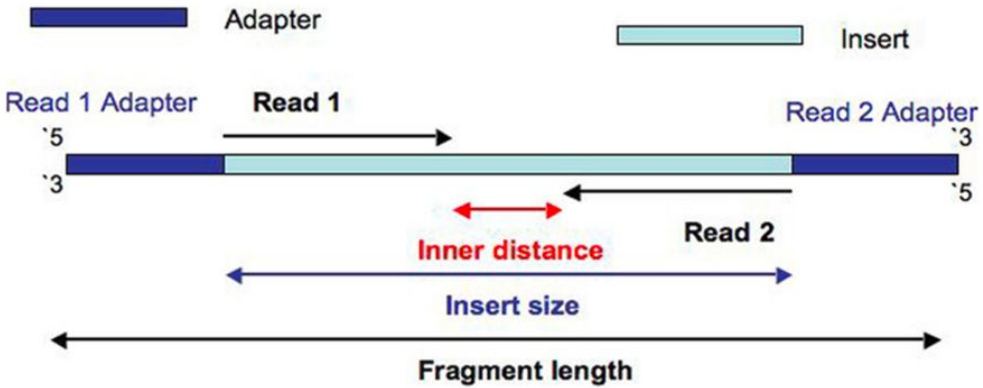
```
# Conservation des alignements dans les régions ciblées
$ module load bedtools/2.29.2
$ bedtools --version              # affiche la version (v2.29.2)
$ bedtools intersect --help       # affiche l'aide

$ bedtools intersect -a SRR1262731_extract.sort.md.filt.bam \
    -b ~/tp_variant/additionnal_data/QTL_BT6.bed \
    > SRR1262731_extract.sort.md.filt.onTarget.bam

$ samtools index SRR1262731_extract.sort.md.filt.onTarget.bam
```

Contrôle qualité des données alignées

- Quelles informations regarder une fois l'alignement effectué ?

  → Pourcentage total de reads alignés
  → Pourcentage de reads appariés **"correctement"**

- Quels outils ?
  - Samtools flagstat
  - Qualimap [optionnel]
  - MultiQC



Contrôle qualité des données alignées

```
# Lancement de samtools
$ samtools flagstat                  # affiche l'aide

$ samtools flagstat SRR1262731_extract.sort.bam > SRR1262731.flagstat.txt

$ cat SRR1262731.flagstat.txt    # visualisation du résultat

$ samtools stats                     # affiche l'aide

$ samtools stats SRR1262731_extract.sort.bam > SRR1262731.stats.txt

$ cat SRR1262731.stats.txt   # visualisation du résultat
```

Contrôle qualité des données alignées

- Visualisation des contrôles qualité

```
# Lancement de Multiqc
$ module load multiqc/1.11
$ multiqc -h                        # affiche l'aide

$ multiqc -f .

# Téléchargement du fichier html à partir de jupyterhub
```

Contrôle qualité des données alignées

```
# Lancement de Qualimap
$ module load qualimap/2.2.2b
$ qualimap -h                    # affiche les outils disponibles (+ version)
$ qualimap bamqc                 # affiche l'aide

$ qualimap bamqc -nt 4 -outdir SRR1262731_extract_qualimap_report \
    --java-mem-size=4G -bam SRR1262731_extract.sort.bam

# Création d'une archive zip
$ zip -r SRR1262731_extract_qualimap_report.zip \
SRR1262731_extract_qualimap_report

# Téléchargement du fichier zip à partir de jupyterhub
```

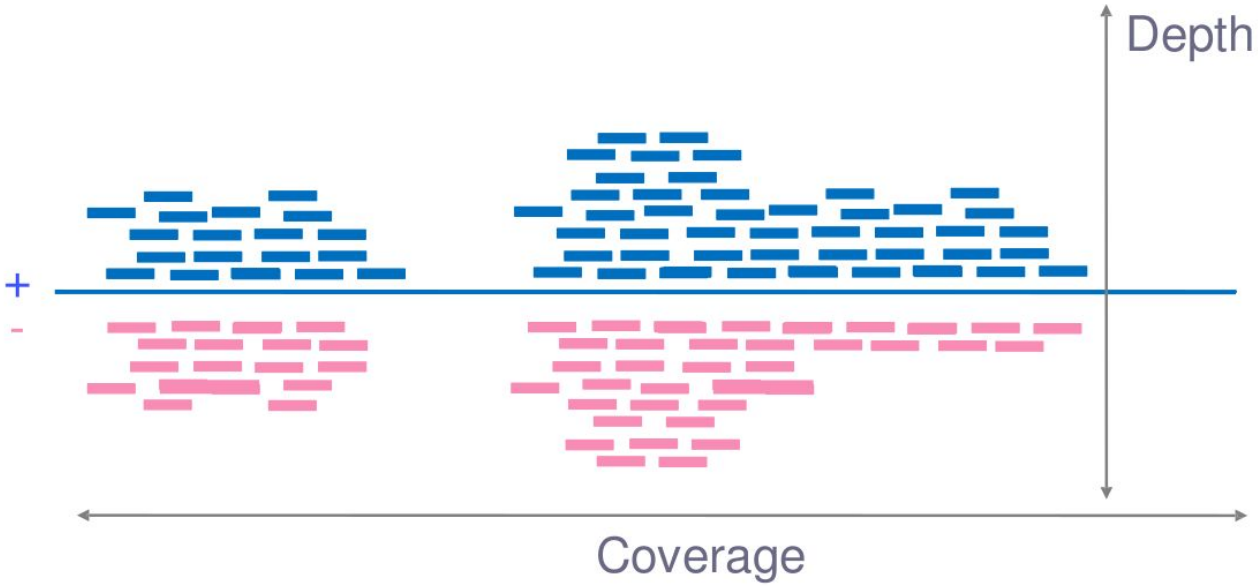
```
# Generation d'un rapport multiqc
$ multiqc -f .
```

Analyse de la couverture

Contrôle qualité de l'enrichissement de ma capture :

→ Est-ce que ma région est **couverte par suffisamment de reads** ?

→ Cette couverture est-elle homogène sur toute la région ?



Analyse de la couverture

Contrôle qualité de l'enrichissement de ma capture :

→ Est-ce que ma région est **couverte par suffisamment de reads** ?

→ Cette couverture est-elle homogène sur toute la région ?

```
# Calcul de la couverture avec samtools
$ samtools depth --help          # affiche l'aide

$ samtools depth -b ~/tp_variant/additionnal_data/QTL_BT6.bed \
    SRR1262731_extract.sort.md.filt.onTarget.bam \
    > SRR1262731_extract.onTarget.depth.txt

$ head SRR1262731_extract.onTarget.depth.txt

# Compter les position avec une profondeur inférieure à 3
$ awk '{if($3<3)print}' SRR1262731_extract.onTarget.depth.txt | wc -l
```