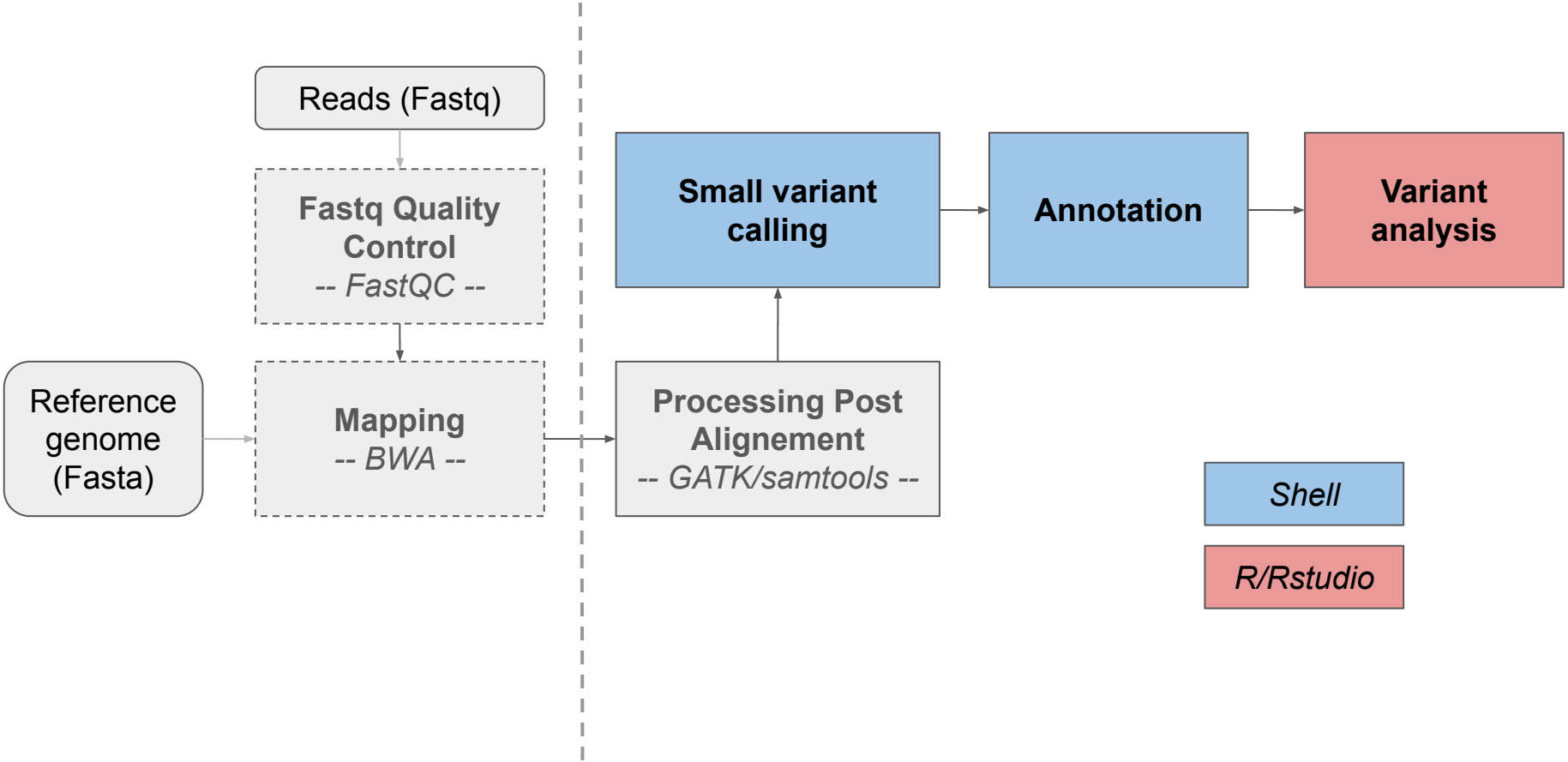
Variant calling

Vivien Deshaies - AP-HP

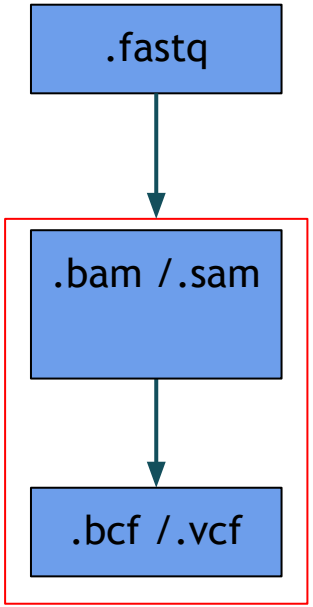
École de bioinformatique AVIESAN-IFB-INSERM 2022

Workflow


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

Qu'appelle t-on "Variant Calling"

Détection **automatisée** des variants (SNVs, Indels de petite taille) à partir d'un fichier contenant des données de séquençage alignées (BAM)




```
@H5:1:H3T27BBXY:8:1101:1955:1191/1
ATTNTTATAGATTCTAGGAAGTTGCTCGAGAAGTTTTCTAATTAGTAGAAGTTGTTGGAGAAGCGTCCTAGTTAGCGGAAGTAGCTCGAGAAGCTTCCTATTCAGTAATATATATAAGAGTCGAGG
+
AAA#FJJFJJJJFFJJJJJJJJFJJFJFJJJJJJ<<AJJJJJJJJJJJJJA<JJFJJJJJJJJJJFF<<JJJFJJJJFAJFJJ<JFJJJJJJJJF<FJJAJ-FFJFFAAAFJ<A-FJFJJ-7FFFJ
```

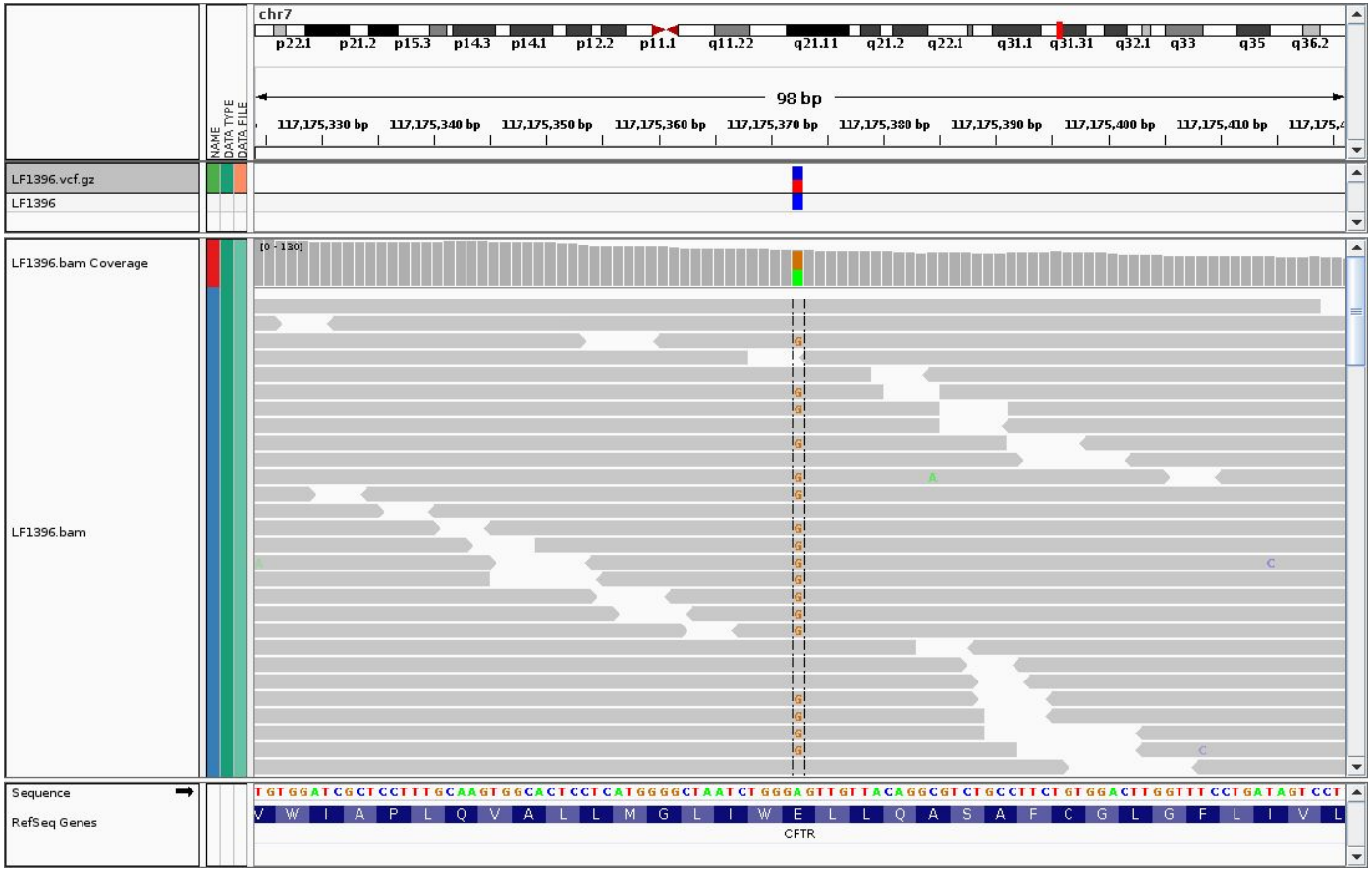
```
H5:1:H3T27BBXY:8:1110:4878:2035 83      Chr01   1568    60      136M    =       1495    -209    AAACCCTAAACCCTAAACCCTAAACCCTAA
H5:1:H3T27BBXY:8:1128:11657:35198       99      Chr01   1572    60      151M    =       1843    422     CCTAAACCCTAAACCCTAAACCC
H5:1:H3T27BBXY:8:1217:6045:36200        163     Chr01   1575    60      115M    =       1575    126     AAACCCTAAACCCTAAACCCTAA
H5:1:H3T27BBXY:8:1217:6045:36200        83      Chr01   1575    60      126M    =       1575    -126    AAACCCTAAACCCTAAACCCTAA
H5:1:H3T27BBXY:8:2227:16863:39963       83      Chr01   1582    60      89M     =       1560    -111    AACCCCTAACCCCTAAACCCTAA
```

```
#CHROM  POS     ID      REF     ALT     QUAL    FILTER  INFO    FORMAT  Ech-456
Chr2    1091    .       C       A       161.77  .       AC=1;AF=0.500;AN=2;BaseQRankSum=0.672;ClippingRankSum=0.567;DP=44;Exces
Chr2    1226    .       T       A       618.77  .       AC=1;AF=0.500;AN=2;BaseQRankSum=-6.233;ClippingRankSum=1.014;DP=201;Exc
Chr2    1708    .       G       A       133.77  .       AC=1;AF=0.500;AN=2;BaseQRankSum=0.000;ClippingRankSum=-0.720;DP=6;Exces
```

Qu'appelle t-on "Variant Calling"

.vcf

.bam /.sam



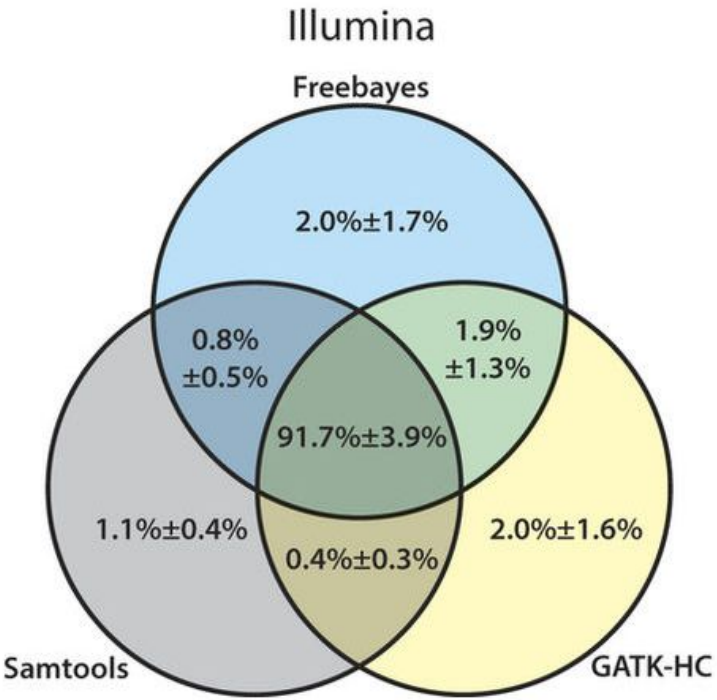
https://jchoigt.wordpress.com/2012/07/18/working-with-23andme-exome-data-my-cf-allele-and-the-need-for-verification/

Variant callers

- Choix du variant caller en fonction de la question biologique
- Utilisés classiquement par la communauté :
    - GATK Haplotype Caller
    - Samtools mpileup/Bcftools
    - Samtools mpileup/VarScan2
    - FreeBayes
    - GATK Mutect2 (spécifique à la détection tumorale)
    - DiscoSnp (variant calling sans génome de référence)
    - DeepVariant (???) (regions complexes, low depth)

→ Aucun outil n'est parfait : la qualité du calling dépend de l'ensemble du pipeline, des données analysées, et des paramètres utilisés pour filtrer les résultats

Concordance entre variant callers



- **Concordance de 91.7%** entre Freebayes, Samtools, GATK HC (Hwang et al., 2015)

- D'autres analyses montrent des taux plus bas :
  - **70%** (O'Rawe et al., Genome Med, 2013)
  - **57%** (Cornish et al., BioMed, 2015)

- La **sensibilité** et la **précision** diffèrent selon les outils et les paramètres utilisés

/!\ Existence de variants qui sont spécifiques aux différents callers /!\

Hwang et al., "Systematic comparison of variant calling pipelines using gold standard personal exome variants" Scientific Reports, vol. 5, pp. 17875, 2015.

Difficultés - Limitations

- De nombreux variants Faux Positifs peuvent survenir des étapes précédentes :
  - Artéfacts issus des **cycle PCR** pendant la préparation des échantillons
  - Artéfacts liés à **la technologie de séquençage** (PacBio, HiSeq, NextSeq, …)
  - Difficultées d'**alignement** (régions d'ADN répétées)
  - **Erreurs de lecture** lors du "BaseCalling"

- Des algorithmes complexes de détection compliquent l'interprétation des résultats

En conclusion

- La détection de variant permet d'i**dentifier des SNVs et petits Indels** à partir d'un fichier d'alignement au format BAM

- De nombreux outils existent pour la détection de variants, leur efficacité dépend de nombreux paramètres (mapping, qualité des données, paramètres de filtrage des résultats)

- La "**sensibilité**" et la "**précision**" permettent d'évaluer la qualité des résultats de détection de variant. Pour un même outil ces mesures varient selon les seuils de qualité utilisés.

Partie TP

- **GATK HaplotypeCaller :**
  - GATK (Genome Analysis ToolKit) est une suite d'outils développée par le Broad Institute
  - Bonne documentation (Best Practices)
  - Permet la gestion d'analyse de plusieurs échantillons (format gVCF)
  - Comporte une étape de réassemblage et réalignement local des indel.
  - Algorithme bayésien (modèles statistiques pour estimer la probabilité de chaque génotype possible, en prenant en compte les différents biais pouvant introduire du bruit dans les données)

Indexation du génome pour GATK

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

```
$ # se déplacer dans le dossier genome
$ cd ~/tp_variant/genome/


$ 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

$ cat Bos_taurus.UMD3.1.dna.toplevel.6.dict
```

GATK HaplotypeCaller

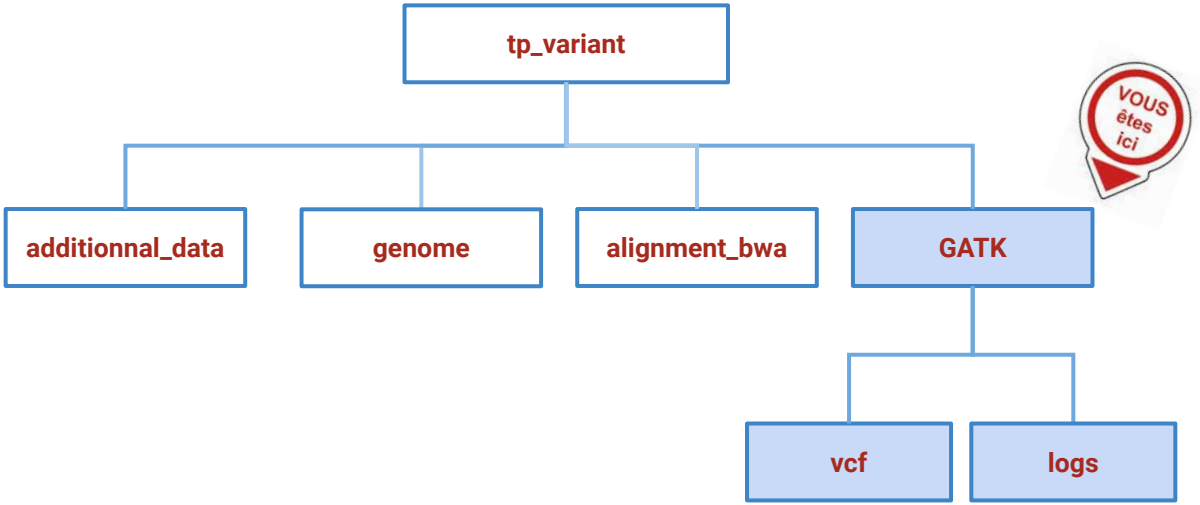
```
$ # module load gatk4/4.2.3.0              # si vous ne l'avez pas déjà fait
$ gatk HaplotypeCaller --version      # affiche la version de GATK (v 4.2.3.0)
```

```
$ gatk HaplotypeCaller                      # affiche l'aide d'HaplotypeCaller
Required Arguments:
--input,-I:String       BAM/SAM/CRAM file containing reads. This argument must be specified at least once.

--output,-O:String      File to which variants should be written  Required.

--reference,-R:String   Reference sequence file  Required.

--min-base-quality-score,-mbq:Byte
                        Minimum base quality required to consider a base for calling  Default value: 10.
...
--emit-ref-confidence,-ERC:ReferenceConfidenceMode
                        Mode for emitting reference confidence scores …
                        Default value: NONE. Possible values: {NONE, BP_RESOLUTION, GVCF}
```

1/GATK HaplotypeCaller avec sortie VCF

Single-sample variant calling

```
# Création d'un répertoire pour l'appel des variants
$ mkdir -p ~/tp_variant/GATK/vcf
$ cd ~/tp_variant/GATK/
```



1/GATK HaplotypeCaller avec sortie VCF

Single-sample variant calling

```
# Création d'un répertoire pour l'appel des variants
$ mkdir -p ~/tp_variant/GATK/vcf
$ cd ~/tp_variant/GATK/
# Détection de variant GATK avec sortie VCF
$ gatk HaplotypeCaller --java-options '-Xmx8G' \
    --input ~/tp_variant/alignment_bwa/SRR1262731_extract.sort.md.filt.onTarget.bam \
    --reference ~/tp_variant/genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa \
    --min-base-quality-score 18  \
    --minimum-mapping-quality 30 \
    --emit-ref-confidence "NONE" \
    --output vcf/SRR1262731_extract_GATK.vcf \
    --intervals ~/tp_variant/additionnal_data/QTL_BT6.bed

$ ls -ltrh vcf/
$ less -S vcf/SRR1262731_extract_GATK.vcf
```

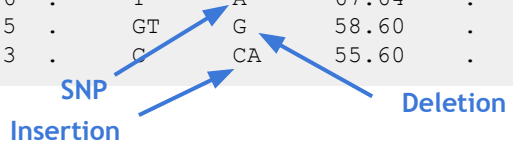
VCF (variant call format)

VCF header

```
##fileformat=VCFv4.2
##FILTER=<ID=LowQual,Description="Low quality">
##FORMAT=<ID=AD,Number=R,Type=Integer,Description="Allelic depths for the ref and alt alleles in the order listed">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Approximate read depth (reads with MQ=255 or with bad mates are filtered)"
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=PL,Number=G,Type=Integer,Description="Normalized, Phred-scaled likelihoods for genotypes as defined in the VCF s
##GATKCommandLine=<ID=HaplotypeCaller,CommandLine="HaplotypeCaller  --min-base-quality-score 18 --emit-ref-confidence NONE --
##INFO=<ID=AF,Number=A,Type=Float,Description="Allele Frequency, for each ALT allele, in the same order as listed">
##INFO=<ID=AN,Number=1,Type=Integer,Description="Total number of alleles in called genotypes">
##INFO=<ID=BaseQRankSum,Number=1,Type=Float,Description="Z-score from Wilcoxon rank sum test of Alt Vs. Ref base qualities">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Approximate read depth; some reads may have been filtered">
##INFO=<ID=ExcessHet,Number=1,Type=Float,Description="Phred-scaled p-value for exact test of excess heterozygosity">
##INFO=<ID=FS,Number=1,Type=Float,Description="Phred-scaled p-value using Fisher's exact test to detect strand bias">
##INFO=<ID=InbreedingCoeff,Number=1,Type=Float,Description="Inbreeding coefficient as estimated from the genotype likelihoods
##INFO=<ID=MLEAC,Number=A,Type=Integer,Description="Maximum likelihood expectation (MLE) for the allele counts (not necessari
##INFO=<ID=MLEAF,Number=A,Type=Float,Description="Maximum likelihood expectation (MLE) for the allele frequency (not necessar
##INFO=<ID=MQ,Number=1,Type=Float,Description="RMS Mapping Quality">
##INFO=<ID=MQRankSum,Number=1,Type=Float,Description="Z-score From Wilcoxon rank sum test of Alt vs. Ref read mapping qualiti
##INFO=<ID=QD,Number=1,Type=Float,Description="Variant Confidence/Quality by Depth">
##INFO=<ID=ReadPosRankSum,Number=1,Type=Float,Description="Z-score from Wilcoxon rank sum test of Alt vs. Ref read position b
##INFO=<ID=SOR,Number=1,Type=Float,Description="Symmetric Odds Ratio of 2x2 contingency table to detect strand bias">
##contig=<ID=6,length=119458736>
##source=HaplotypeCaller
```

Body

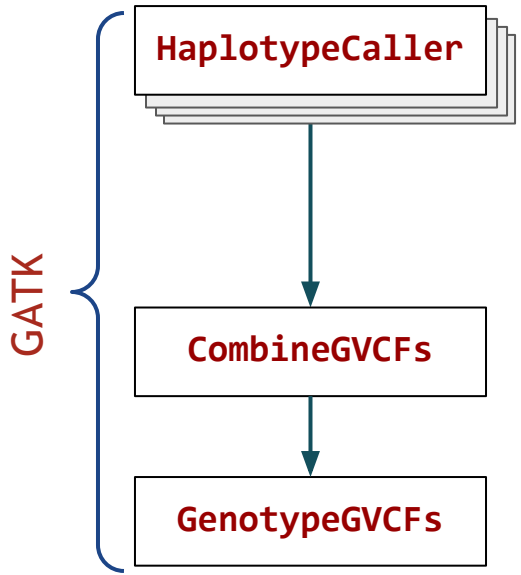
#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	SRR1262731
6	37913396	.	T	A	67.64	.	AC=1;AF=0.500;...	GT:AD:DP:GQ:PL	0/1:3,2:5:75:75,0,105
6	37916445	.	GT	G	58.60	.	AC=1;AF=0.500;...	GT:AD:DP:GQ:PL	0/1:1,2:3:28:66,0,28
6	37921683	.	C	CA	55.60	.	AC=1;AF=0.500;...	GT:AD:DP:GQ:PL	0/1:7,2:9:63:63,0,279



1/GATK HaplotypeCaller en mode GVCF

Multi-sample variant calling

- En 3 étapes (=> 3 outils) :



Variant-calling avec sortie gvcf / par échantillon

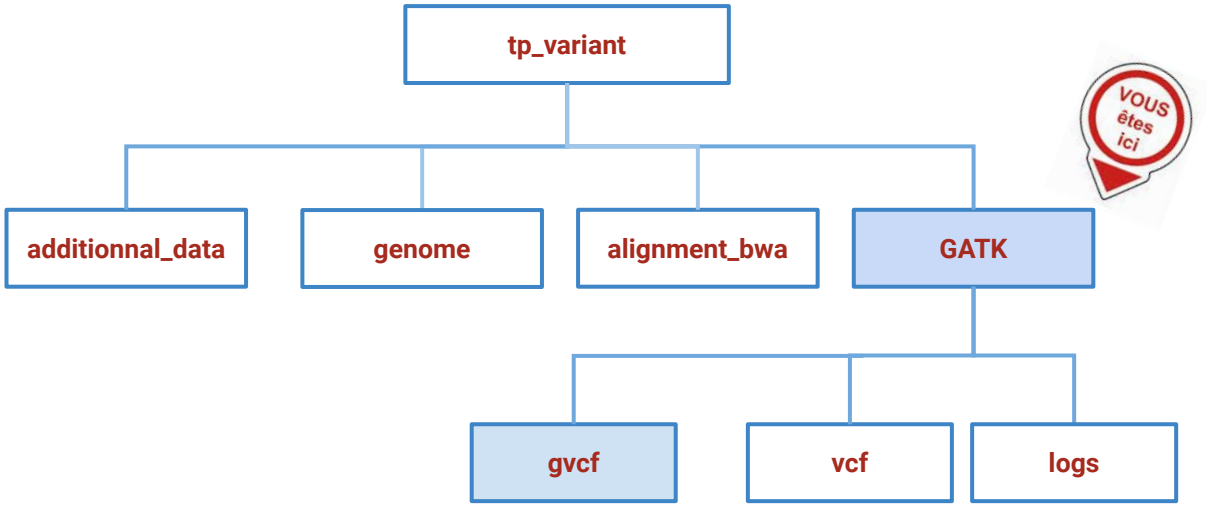
Combiner les sorties gvcf en 1 sortie gvcf

Identifier les variants simultanément sur tous les échantillons

1/GATK HaplotypeCaller en mode GVCF

Multi-sample variant calling

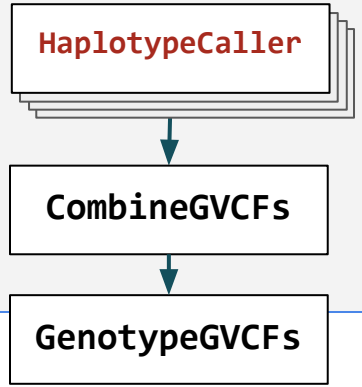
```
# Création d'un répertoire pour l'appel des variants
$ mkdir -p ~/tp_variant/GATK/gvcf
```



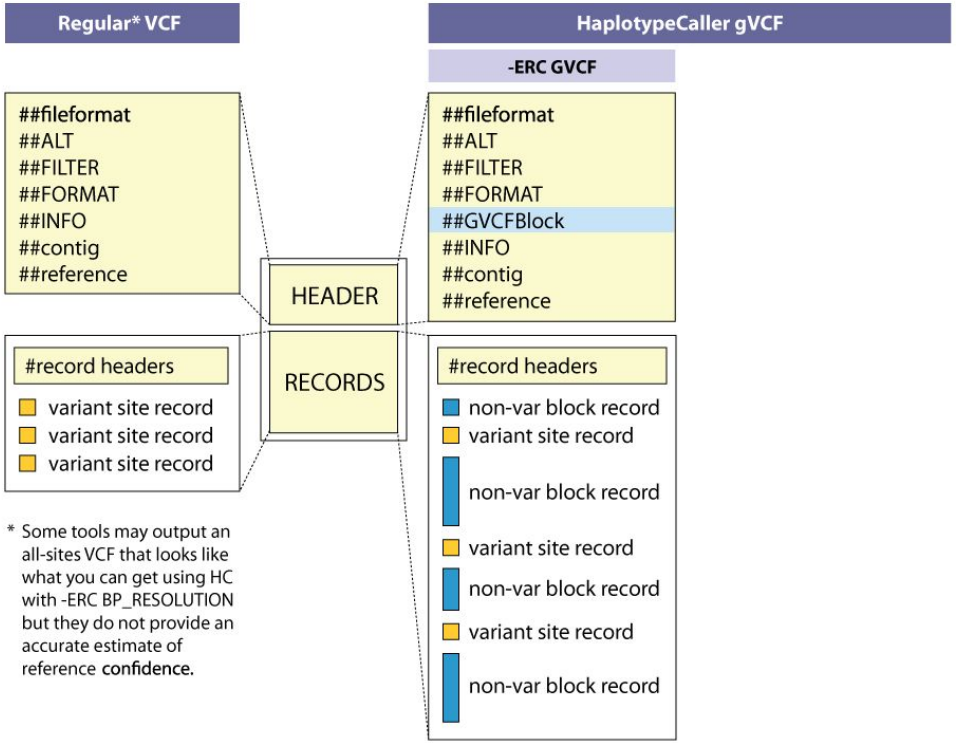
1/GATK HaplotypeCaller en mode GVCF

Multi-sample variant calling

```
# 1.Détection de variants GATK avec sortie gVCF
$ mkdir -p ~/tp_variant/GATK/gvcf
$ gatk HaplotypeCaller --java-options '-Xmx8G' \
    --input ~/tp_variant/alignment_bwa/SRR1262731_extract.sort.md.filt.onTarget.bam \
    --reference ~/tp_variant/genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa \
    --min-base-quality-score 18  \
    --minimum-mapping-quality 30 \
    --emit-ref-confidence "GVCF" \
    --output gvcf/SRR1262731_extract_GATK.g.vcf \
    --intervals ~/tp_variant/additionnal_data/QTL_BT6.bed

$ ls -ltrh gvcf/
$ less -S gvcf/SRR1262731_extract_GATK.g.vcf
```



Sorties VCF *vs.* gVCF (option -ERC)



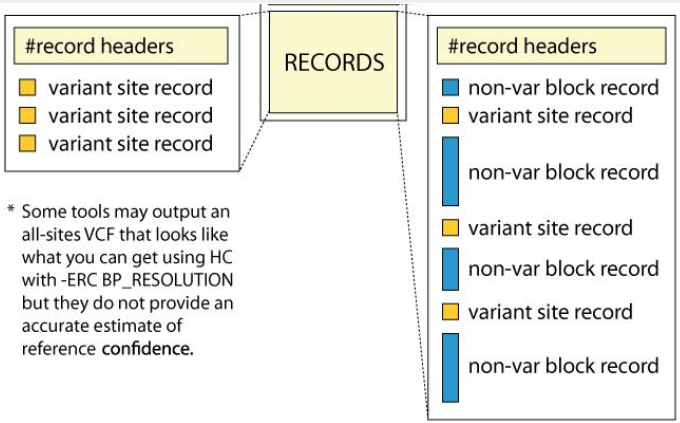
Sorties VCF *vs.* gVCF (option -ERC)

VCF

#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	SRR1262731
6	37913396	.	T	A	67.64	.	AC=1;AF=0.500;...	GT:AD:DP:GQ:PL	0/1:3,2:5:75:75,0,105
6	37916445	.	GT	G	58.60	.	AC=1;AF=0.500;...	GT:AD:DP:GQ:PL	0/1:1,2:3:28:66,0,28
6	37921683	.	C	CA	55.60	.	AC=1;AF=0.500;...	GT:AD:DP:GQ:PL	0/1:7,2:9:63:63,0,279

gVCF

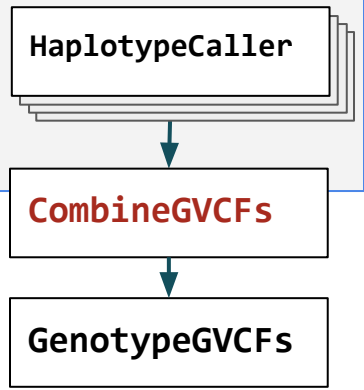
#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	SRR1262731
6	37913111	.	G	<NON_REF>	.	.	END=37913131	GT:DP:GQ:MIN_DP:PL	0/0:3:9:3:0,9,114
6	37913132	.	A	<NON_REF>	.	.	END=37913133	GT:DP:GQ:MIN_DP:PL	0/0:4:12:4:0,12,170
...									
6	37913394	.	T	<NON_REF>	.	.	END=37913395	GT:DP:GQ:MIN_DP:PL	0/0:5:12:5:0,12,180
6	**37913396**	**.**	**T**	**A,<NON_REF>**	**67.64**	**.**	**BaseQRankSum...**	**GT:AD:DP:GQ:PL:SB**	**0/1:3,2,0:5:75:75,..**
6	37913397	.	A	<NON_REF>	.	.	END=37913400	GT:DP:GQ:MIN_DP:PL	0/0:5:12:5:0,12,180



1/GATK HaplotypeCaller en mode GVCF

Multi-sample variant calling

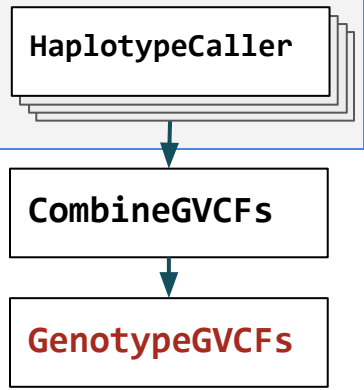
```
# 2.Fusion des fichiers gVCFs en un seul gVCF
$ gatk CombineGVCFs --java-options '-Xmx8G' \
    --variant gvcf/SRR1262731_extract_GATK.g.vcf \
    --variant  ~/tp_variant/additionnal_data/SRR1205992_extract_GATK.g.vcf \
    --variant  ~/tp_variant/additionnal_data/SRR1205973_extract_GATK.g.vcf \
    --reference ~/tp_variant/genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa \
    --intervals ~/tp_variant/additionnal_data/QTL_BT6.bed \
    --output gvcf/pool_GATK.g.vcf
```



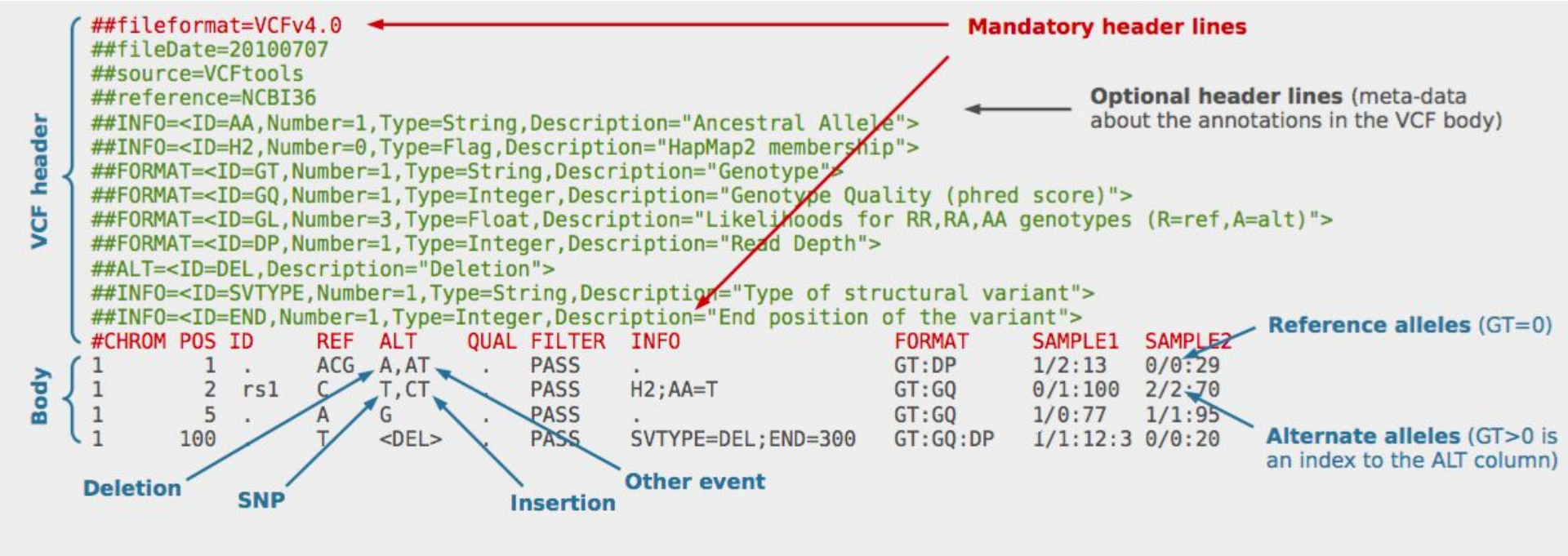
1/GATK HaplotypeCaller en mode GVCF

Multi-sample variant calling

```
# 3.Détection de variants simultanée sur les 3 échantillons du gVCF
$ gatk GenotypeGVCFs --java-options '-Xmx8G' \
     --variant  gvcf/pool_GATK.g.vcf \
     --reference ~/tp_variant/genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa \
     --output vcf/pool_GATK.vcf

$ less -S vcf/pool_GATK.vcf
```

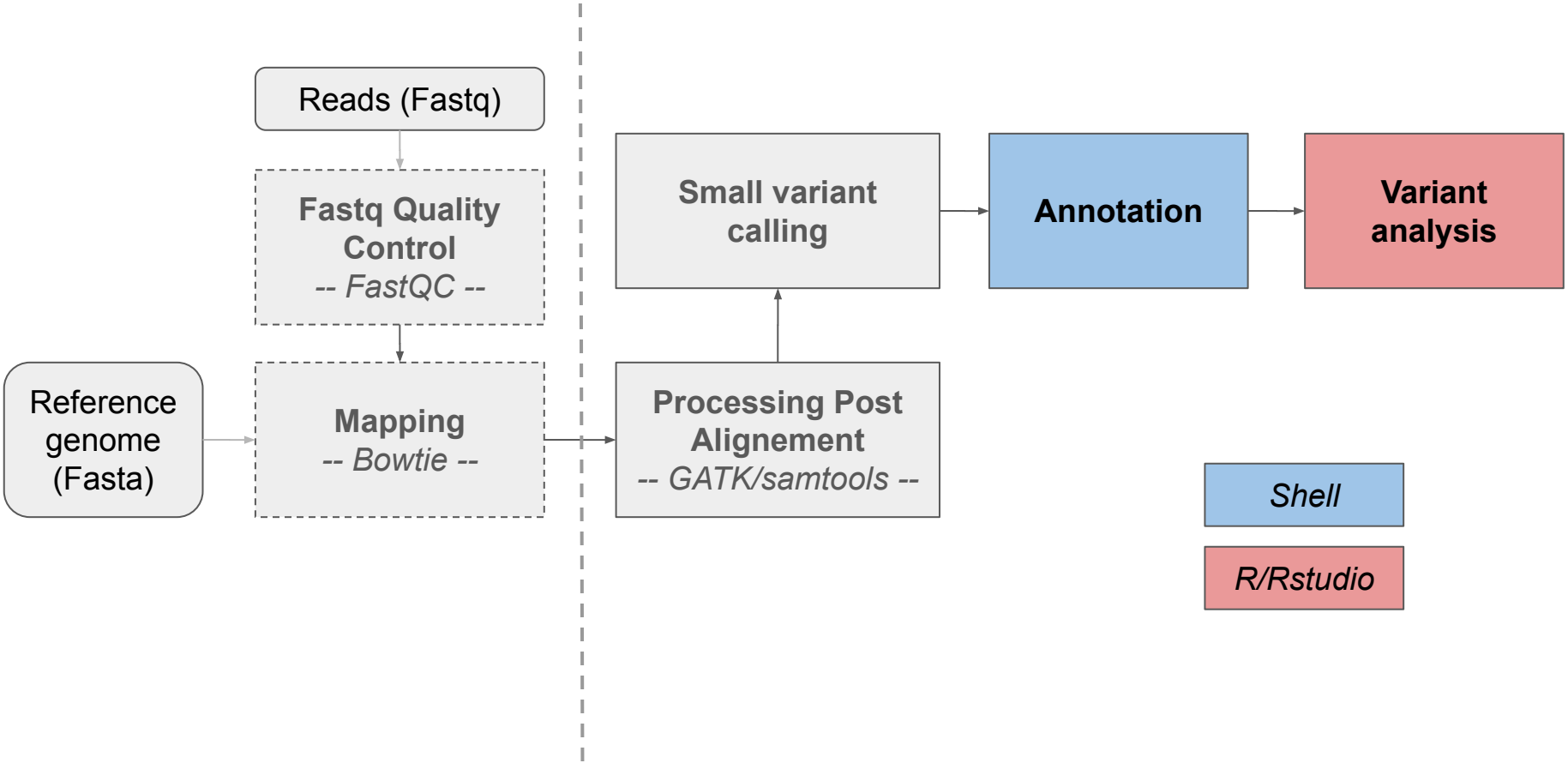


VCF Multi-échantillons



https://petridishtalk.com/2013/02/01/variant-discovery-annotation-filtering-with-samtools-the-gatk/

Workflow


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

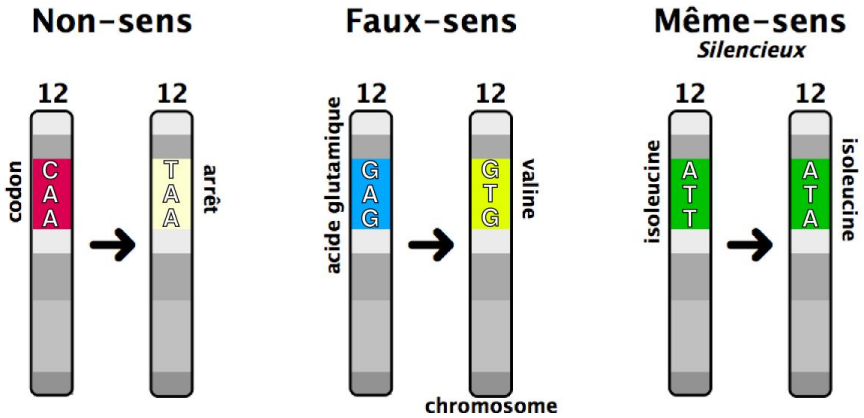
Annotation des variants

- Ajout d'informations biologiques pertinentes aux variants :

→ Est-ce que mes variants sont connus ?

→ Où se positionnent mes variants ?

→ Quel est l'effet d'une mutation sur le CDS qui le contient ?

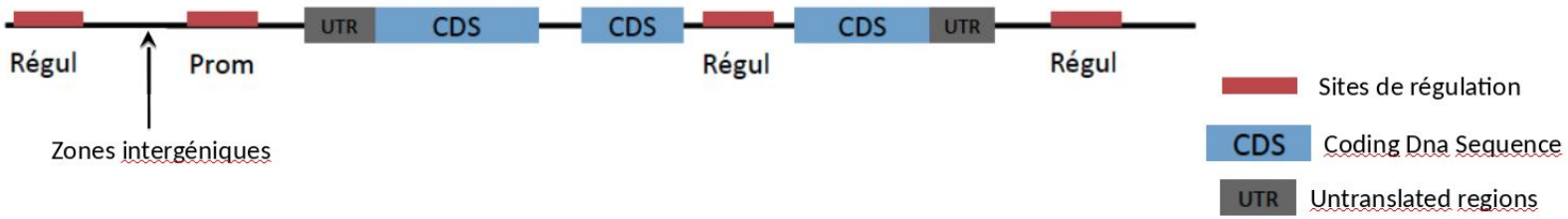


Annotation des variants

- Annotation structurale :
  → Mon variant se trouve-t-il dans un intron, un exon ?

- Annotation fonctionnelle :
  → Informations sur la région ? Exemple : CDS codant pour une protéine

- Impacts potentiels :
  → Dans le cas d'un CDS, protéine produite tronquée, allongée, décalée… ou silencieuse (redondance du code génétique)



Annotation des variants

- Nécessité d'avoir des bases de données associées aux organismes étudiés (Ensembl, Refseq...)

- Exemples d'**outils/algorithmes** :
  → SnpEff
  → VEP
  → Annovar
  → SIFT, POLYPHEN2, CADD…
  → dbNSFP,

Snpeff bases pré-construites

```
# récupération de la liste des bases pré-construites
$ module load snpeff
$ snpEff -version              # affiche la version (v4.3t)

$ mkdir -p ~/tp_variant/snpeff
$ cd ~/tp_variant/snpeff

$ snpEff databases > snpeff_databases.txt

$ grep -i -e "Bos.*taurus" snpeff_databases.txt
```

SnpEff

```
# Création de la base de données SnpEff

$ echo BosTaurus.genome > snpeff.config          # <genome_name>.genome
$ mkdir -p BosTaurus
$ cp ~/tp_variant/genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa BosTaurus/sequences.fa
$ cp ~/tp_variant/genome/Bos_taurus.UMD3.1.93.chromosome.6.gff3 BosTaurus/genes.gff
$ echo -e "BosTaurus\nSnpEff4.3t" > BosTaurus.db

$ snpEff build -c snpeff.config -gff3 -v BosTaurus -dataDir .
```

```
# Annotation avec notre base de données
$ snpEff eff -c snpeff.config -dataDir . BosTaurus -s snpeff_res.html \
~/tp_variant/GATK/vcf/pool_GATK.vcf > GATK.annot.vcf

$ less -S GATK.annot.vcf
```

SnpSift

```
$ module load snpsift/4.3.1t
$ SnpSift filter -h                    # affiche l'aide (v 4.3t)

# Garder les variants codant qui ne sont pas des synonymes :
$ cat GATK_varscan_inter.annot.vcf | SnpSift filter -Xmx8G \
 \"(ANN[*].EFFECT != 'synonymous_variant') && (ANN[*].BIOTYPE = 'protein_coding')\" \
   > GATK_varscan_inter.annot.coding.nosyn.vcf
```

```
# Sélectionner notre variant d'intérêt parmi les variants hétérozygotes ayant un
impact (missense)
$ cat GATK_varscan_inter.annot.coding.nosyn.vcf | SnpSift filter -Xmx8G \
 \"ANN[*].EFFECT = 'missense_variant' & isHet( GEN[2] ) & isVariant( GEN[2] ) \
 & isRef( GEN[0] ) & isRef( GEN[1] ) \" \
> GATK_varscan_inter.annot.coding.nosyn.filtered.vcf
```

Variant d'intérêt

- Quelle type de mutation est impliquée dans notre phénotype d'intérêt pour l'individu SRR1262731 ?
- Quel est son génotype ? Sur quel gène se situe-elle ?
- Qu'en est-il pour les autres individus ?

→ Le variant est **hétérozygote ALT (0/1)** pour l'individu SRR1262731, il comporte une mutation de type SNP (A → C) située sur le gène **ABCG2**, en position **38027010 du chromosome 6**.

→ Pour les deux autres individus, ils ne comportent pas cette mutation : il sont homozygote référence (GT: 0/0).

Zinder *et al.*, 2005

Filtres des variants

- De **nombreux filtres** peuvent être appliqués sur le VCF
  → type de variants à garder (SNVs seulement, Indels...)
  → région d'intérêt
  → seuils arbitraires : profondeur, génotype (0/1, 1/1), ratio allélique…

- Filtres difficilement transposables entre analyse :

  → dépendent de la **question biologique**
  → dépendent des outils utilisés

- GATK Bests Practices : recommendations selon des métriques spécifiques à GATK, différentes pour les SNVs des Indels

SelectVariants et Hard filtering

```
# Préparation d'un nouveau répertoire de résultats
$ mkdir -p ~/tp_variant/filter_and_annot/logs
$ cd ~/tp_variant/filter_and_annot

# Extraction des SNVs dans un fichier séparé pour GATK
$ sbatch -J GATK_SNP -o logs/GATK_SNP.out -e logs/GATK_SNP.err --mem=8G --wrap=" \
    gatk SelectVariants --java-options '-Xmx8G' \
    -R ~/tp_variant/genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa \
    -V ~/tp_variant/GATK/vcf/pool_GATK.vcf \
    --select-type SNP -O pool_GATK.SNP.vcf"

# Extraction des SNVs dans un fichier séparé pour Varscan
$ sbatch -J Varscan_SNP -o logs/Varscan_SNP.out -e logs/Varscan_SNP.err --mem=8G
--wrap="gatk SelectVariants --java-options '-Xmx8G' \
    -R ~/tp_variant/genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa \
    -V ~/tp_variant/Varscan/pool_Varscan_dict.vcf \
    --select-type SNP -O pool_Varscan.SNP.vcf"
```

SelectVariants et Hard filtering

- **QD** - QualByDepth : Score QUAL / AD [profondeur allélique]
- **FS** - FisherStrand :
- **SOR** - StrandOddsRatio:

} Score estimant un éventuel biais de brin

- **MQ** - MappingQuality : Qualité de mapping moyenne sur l'ensemble du read
- **MQRankSum** : Teste un biais de différence de qualité de mapping entre allèles
- **ReadPosRankSum** : Teste un biais de position des allèles le long du read

doc GATK

HowTo: Apply hard filters to a call set
I am unable to use VQSR (recalibration) to filter variants
how to understand and improve upon the generic hard filtering recommendations.

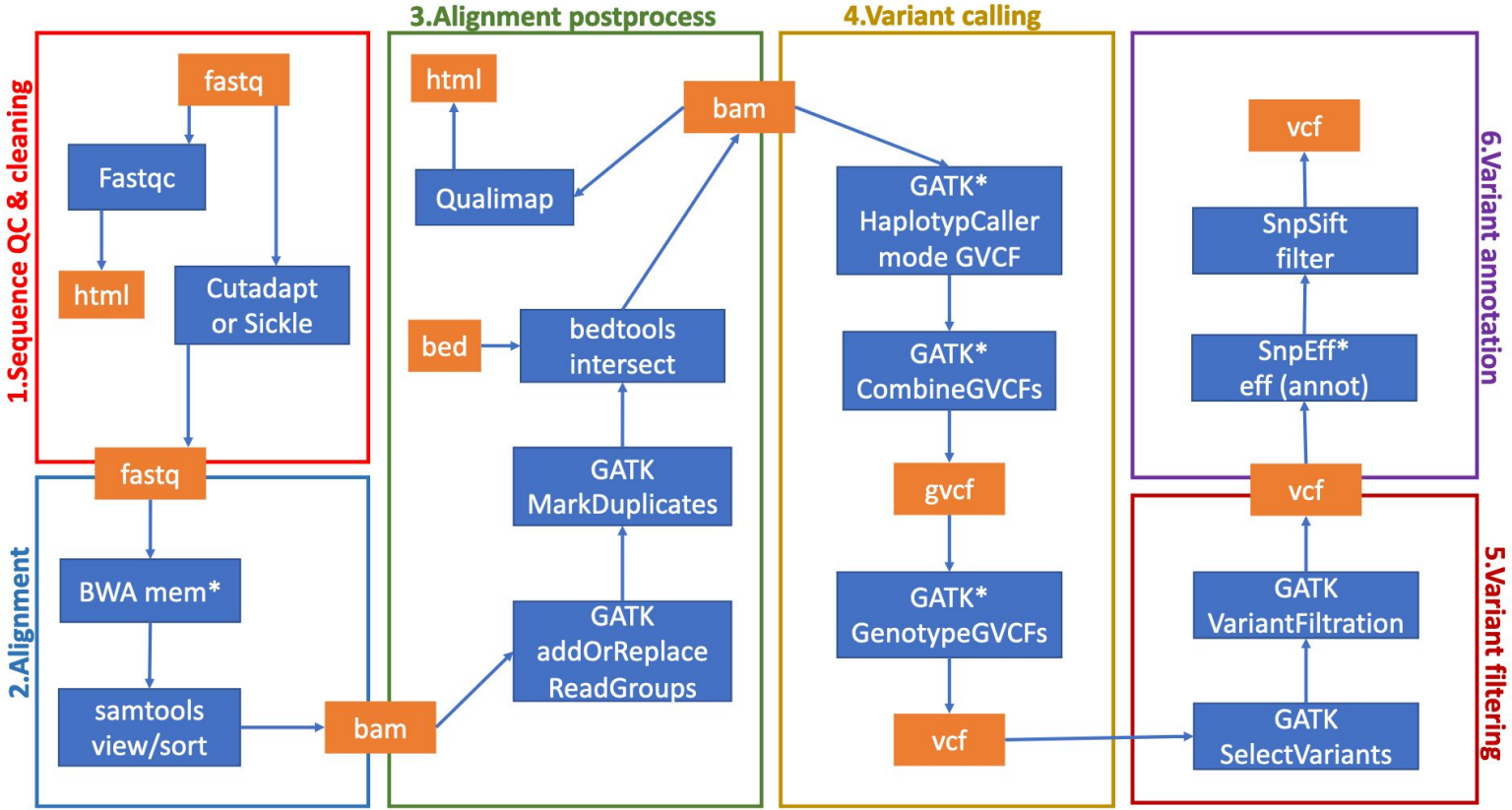
SelectVariants et Hard filtering

```
# Filtrage des SNVs selon les filtres recommandés par GATK
$ sbatch -J GATK_SNP_filter -o logs/GATK_SNP_filter.out -e logs/GATK_SNP_filter.err
--mem=8G --wrap="gatk VariantFiltration --java-options '-Xmx8G' \
  -R ~/tp_variant/genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa \
  -V pool_GATK.SNP.vcf -O pool_GATK.SNP.prefilt.vcf \
  -filter 'QD < 2.0' --filter-name 'QD2' -filter 'SOR > 3.0' --filter-name 'SOR3' \
  -filter 'FS > 60.0' --filter-name 'FS60' -filter 'MQ < 40.0' --filter-name 'MQ40' \
  -filter 'MQRankSum < -12.5' --filter-name 'MQRankSum-12.5' \
  -filter 'ReadPosRankSum < -8.0' --filter-name 'ReadPosRankSum-8'"

# Sélection des variants passant ce filtre
$ sbatch -J GATK_SNP_PASS -o logs/GATK_SNP_PASS.out -e logs/GATK_SNP_PASS.err
--mem=8G --wrap="gatk SelectVariants --java-options '-Xmx8G' \
    -R ~/tp_variant/genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa \
    -V pool_GATK.SNP.prefilt.vcf \
    --exclude-filtered \
    -O pool_GATK.SNP.filtered.vcf"
```

Intersection des résultats des variant callers

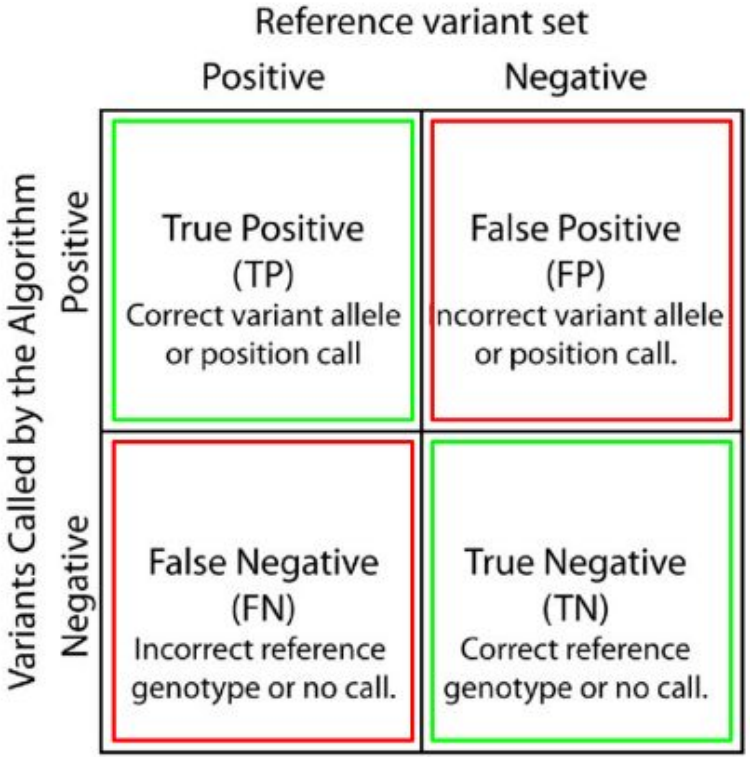
```
# Intersection des variants obtenus avec Varscan et avec GATK post filtering

# Compression et indexation des fichiers vcfs
$ bgzip -c pool_GATK.SNP.filtered.vcf > pool_GATK.SNP.filtered.vcf.gz
$ tabix -p vcf pool_GATK.SNP.filtered.vcf.gz

$ bgzip -c pool_Varscan.SNP.vcf > pool_Varscan.SNP.vcf.gz
$ tabix -p vcf pool_Varscan.SNP.vcf.gz

$ sbatch -J GATK_varscan_isec -o logs/GATK_varscan_isec.out \
    -e logs/GATK_varscan_isec.err --mem=8G --wrap=" \
    bcftools isec -f PASS -n +2 -w 1 -O v \
    pool_GATK.SNP.filtered.vcf.gz pool_Varscan.SNP.vcf.gz \
    > GATK_varscan_inter.vcf "
```

Rappel : du fastq au VCF



* need specific index

Recall/Precision



Recall (sensibilité)

→ Mesure la capacité de l'outil à détecter le maximum de véritables variants

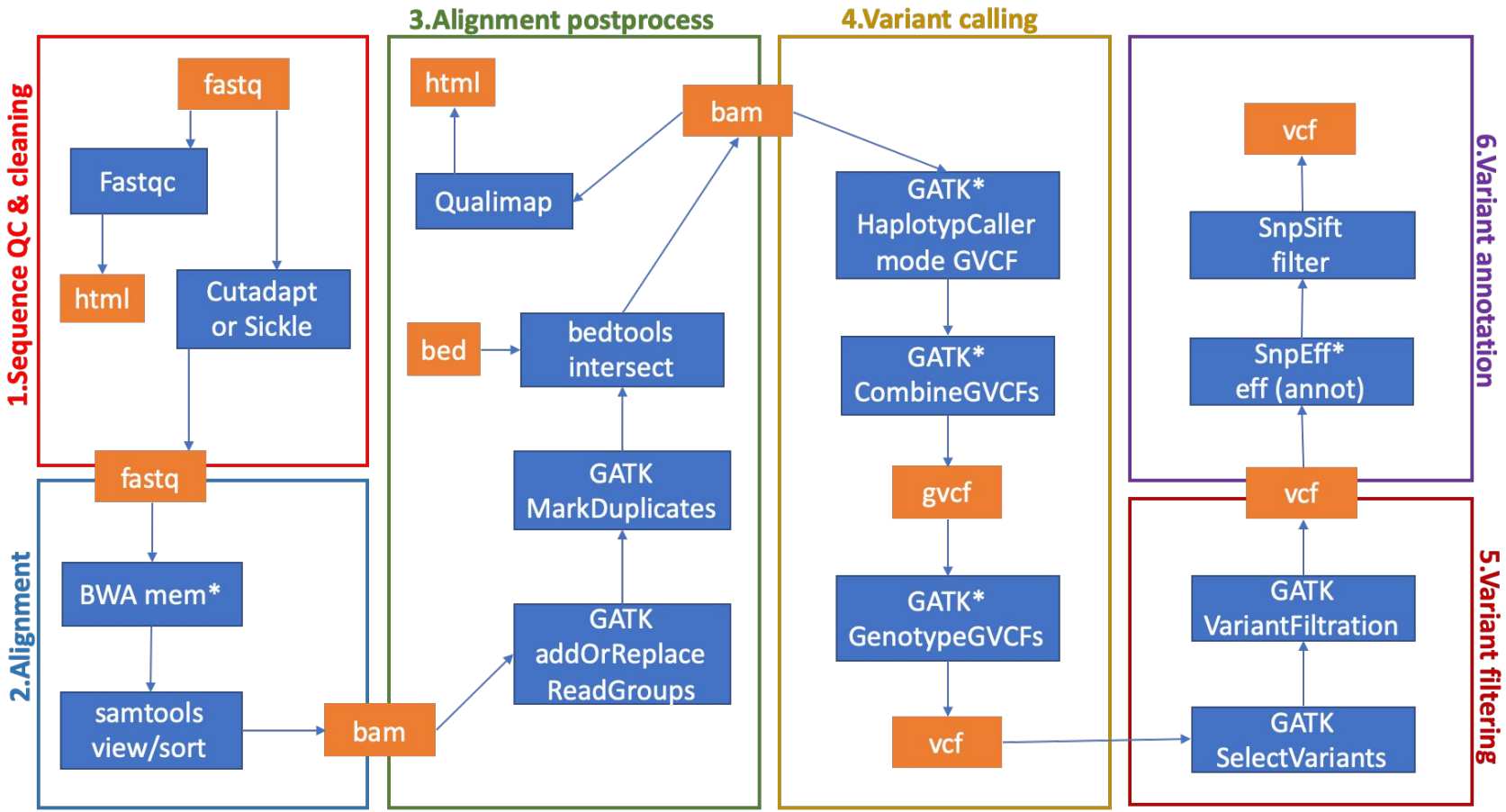
→ TP / (TP + FN)

Precision (spécificité)

→ Mesure la capacité de l'outil à ne pas détecter de faux variants

→ TN / (TN + FP)

Olson et al., Best practices for evaluating single nucleotide variant calling methods for microbial genomics, Frontiers in Genetics, 10.3389/fgene.2015.00235



* need specific index