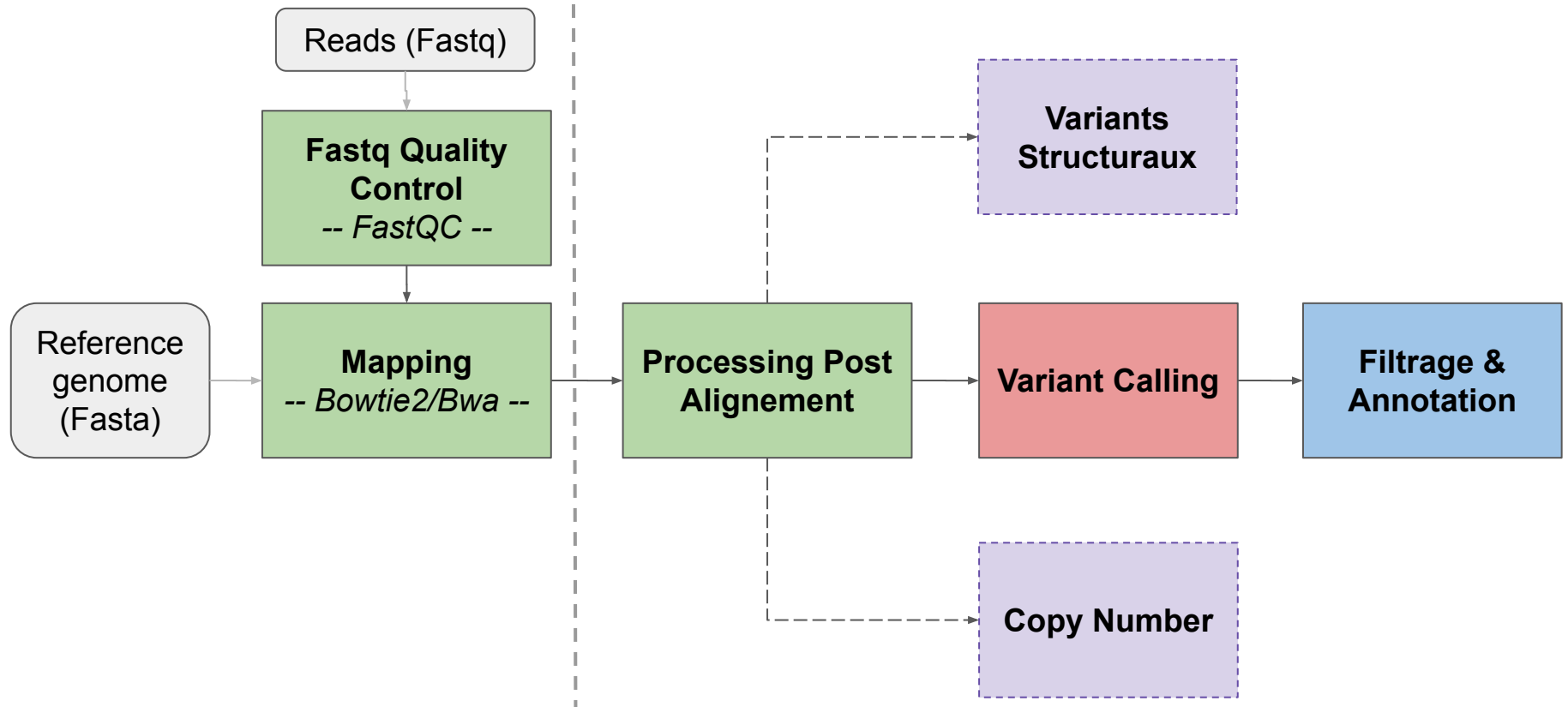




# Variant calling

Nadia Bessoltane - INRAE

# Workflow



# Définition

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

- **SNV** : Single Nucleotide Variant
- **INDEL** : INsertion ou DELetion d'une ou plusieurs bases
- **MNV** (Multi-Nucleotide Variant) : plusieurs SNVs et/ou INDELS dans un bloc
- **SV** (Structural Variant) : réarrangement génomique affectant > 50bp

AACGGCC T GTAAC  
AACGGCC A GTAAC

AACGGCC T GTAAC  
AACGGCC - GTAAC

AACGGCC T GTAAC  
AACGGCC AG C GTAAC

# SNV $\neq$ SNP

- **SNV (Single Nucleotide Variant)**
  - toute altération nucléotidique sans implication de fréquence populationnelle
- **SNP (Single Nucleotide Polymorphism)**
  - implique qu'un variant est partagée dans la population (> 1%)

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

# 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)

.fastq

.bam / .sam

.bcf / .vcf

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

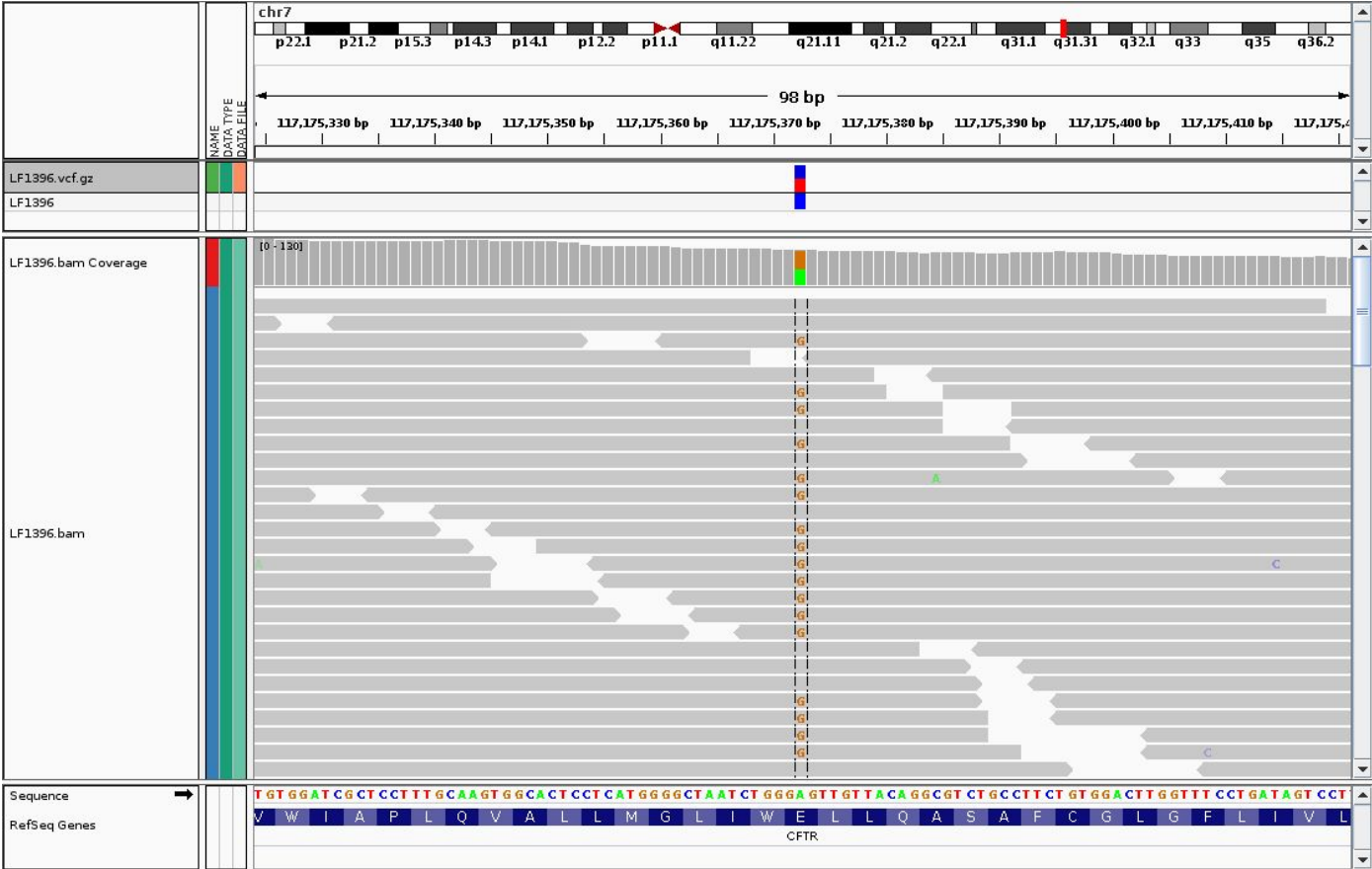
H5:1:H3T27BBXY:8:1110:4878:2035	83	Chr01	1568	60	136M	=	1495	-209	AAACCCTAAACCCTAAACCCTAAACCCTAA
H5:1:H3T27BBXY:8:1128:11657:35198	99	Chr01	1572	60	151M	=	1843	422	CCTAAACCCTAAACCCTAAACCCTAAACC
H5:1:H3T27BBXY:8:1217:6045:36200	163	Chr01	1575	60	115M	=	1575	126	AAACCCTAAACCCTAAACCCTAAACCCTAA
H5:1:H3T27BBXY:8:1217:6045:36200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

#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;Ex		
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

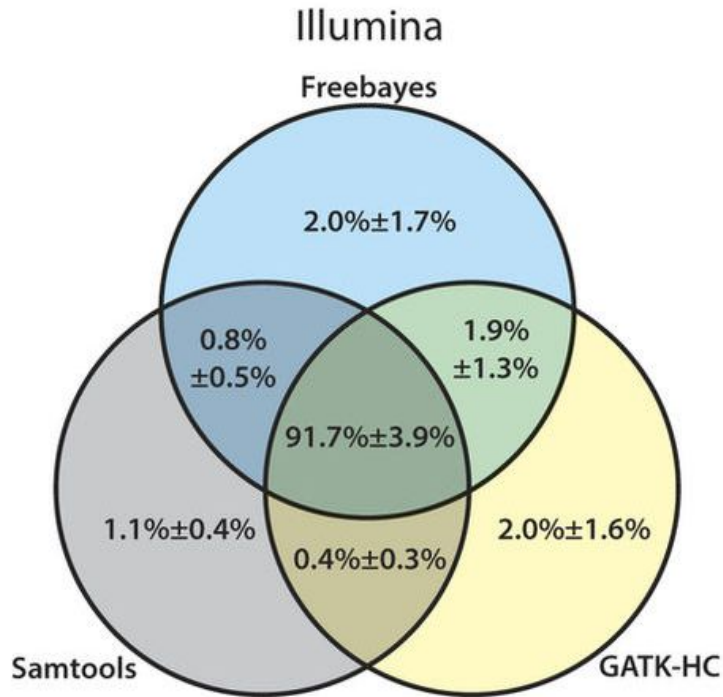


# Variant callers

- 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 Mutect (spécifique à la détection tumorale)
  - DiscoSnp (variant calling sans génome de référence)

→ **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 !/\\**



# 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, ... )
  - Difficultés d'**alignement** (régions d'ADN répétées)
  - **Erreurs de lecture** lors du “BaseCalling”
- Des algorithmes complexes de détection compliquent l'interprétation des résultats

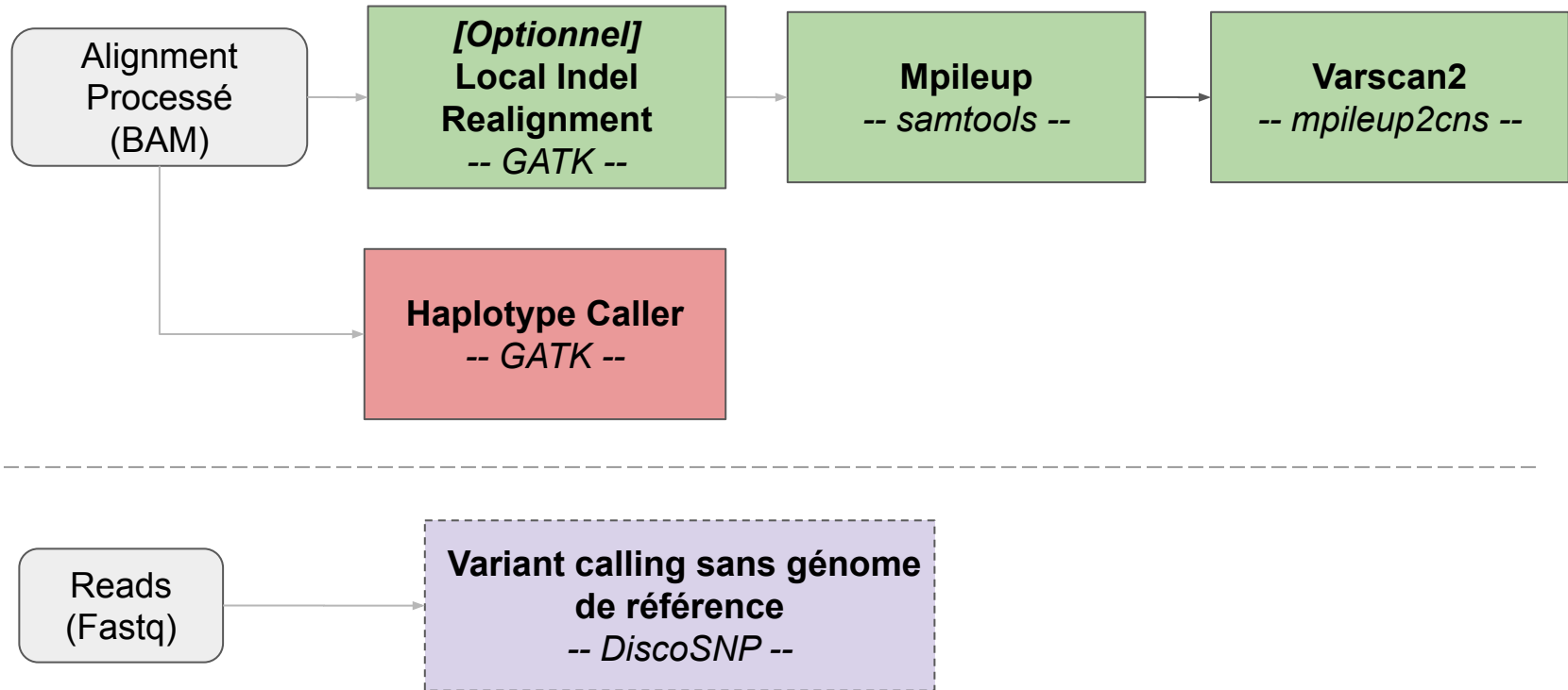
# En conclusion

- La détection de variant permet d'identifier 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

- Utilisation de deux outils : **GATK HaplotypeCaller** et **Varscan2**
- **1/ 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 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)
- **2/ Varscan2 :**
  - Temps d'exécution plus courts
  - Algorithme basé sur des heuristiques (utilise des seuils pour valider ou non les variants : fréquence allélique, couverture en read, score de qualité)

# Workflow - Variant Calling



# GATK HaplotypeCaller

```
$ # module load gatk4/4.1.7.0          # si vous ne l'avez pas déjà fait
$ gatk HaplotypeCaller --version        # affiche la version de GATK (v 4.1.7.0)
```

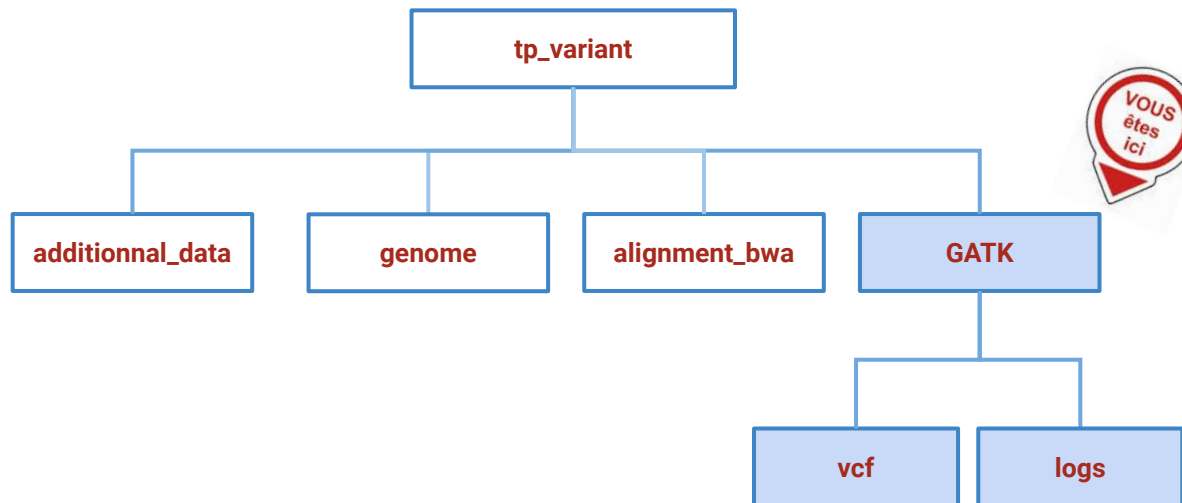
```
$ gatk HaplotypeCaller                  # affiche l'aide d'HaplotypeCaller

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 repertoire pour l'appel des variants  
$ mkdir -p ~/tp_variant/GATK/vcf  
$ mkdir -p ~/tp_variant/GATK/logs  
$ cd ~/tp_variant/GATK/
```



# 1/GATK HaplotypeCaller avec sortie VCF

## *Single-sample variant calling*

```
# Détection de variant GATK avec sortie VCF
$ sbatch -J HC_to_VCF -o logs/HC_to_VCF.out -e logs/HC_to_VCF.err --mem=8G --wrap=" \
    gatk HaplotypeCaller --java-options '-Xmx8G' \
    --input ~/tp_variant/alignment_bwa/SRR1262731_extract.sort.rg.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 logs/
$ 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 spec">
##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 necessarily sum to AN)">
##INFO=<ID=MLEAF,Number=A,Type=Float,Description="Maximum likelihood expectation (MLE) for the allele frequency (not necessarily sum to 1)">
##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 quality">
##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 bias">
##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

SNP

Insertion

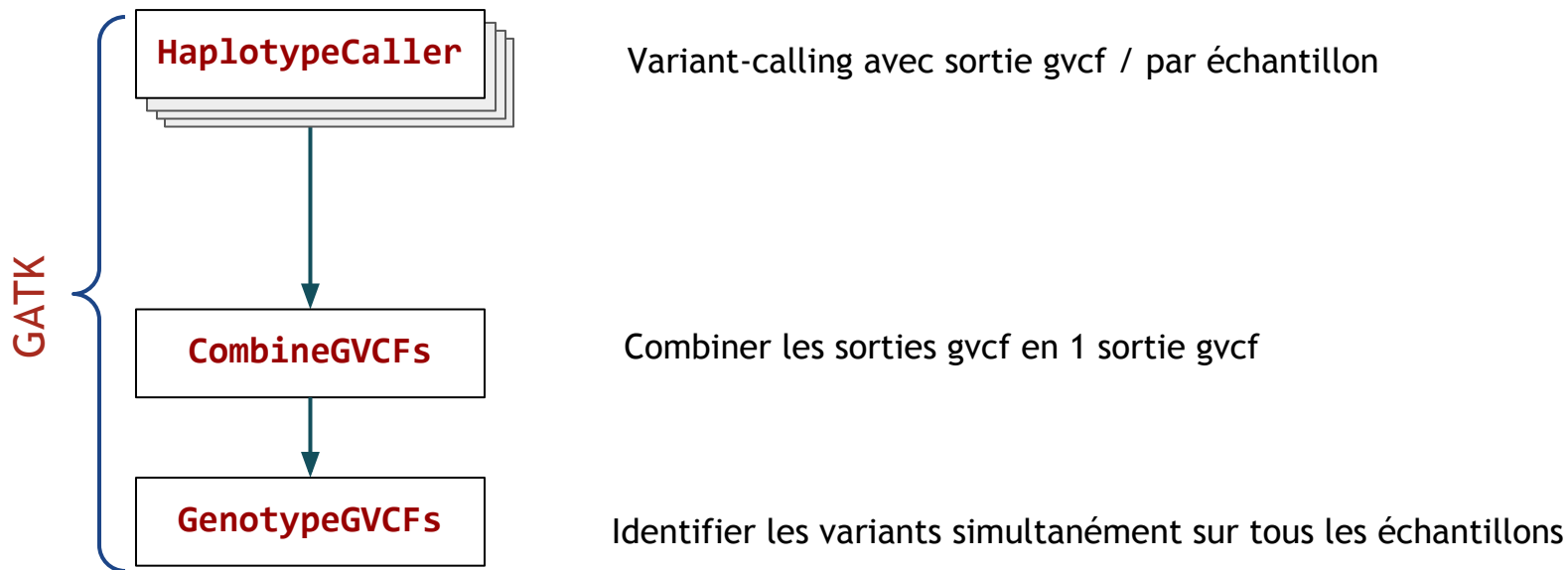
Deletion



# 1 / GATK HaplotypeCaller en mode GVCF

## *Multi-sample variant calling*

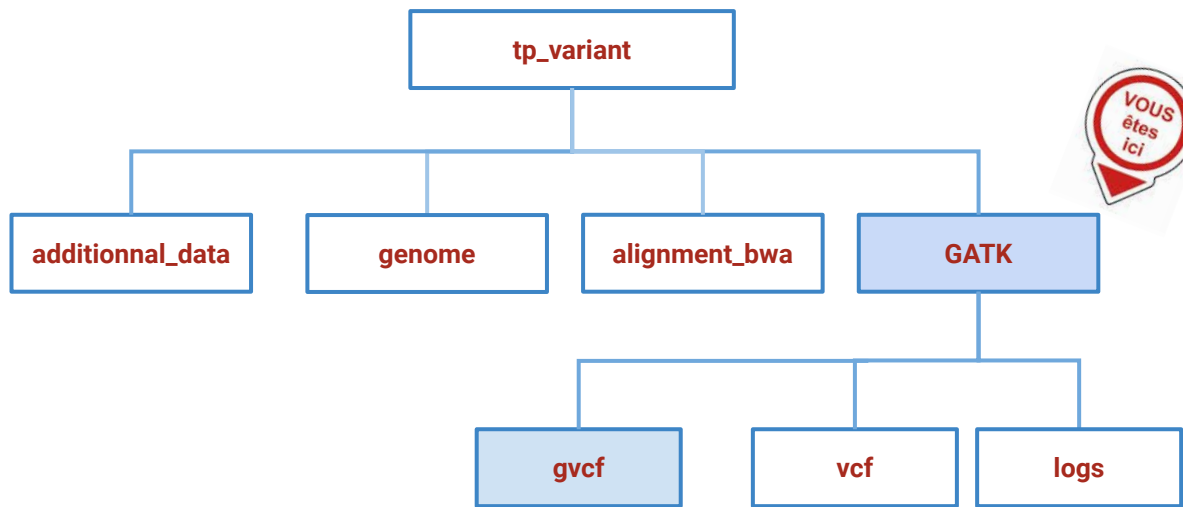
- En 3 étapes (=> 3 **outils**) :



# 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
```

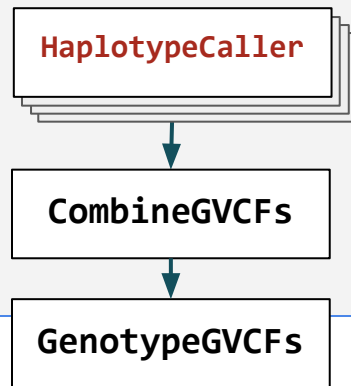


# 1/GATK HaplotypeCaller en mode GVCF

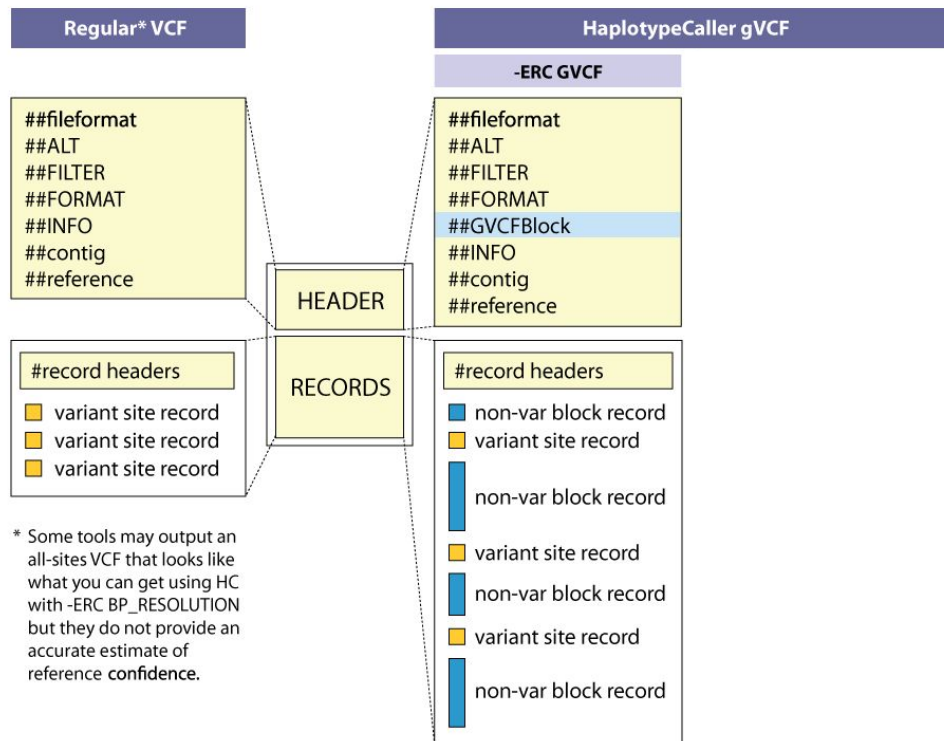
## *Multi-sample variant calling*

```
# 1.Détection de variants GATK avec sortie gVCF
$ sbatch -J HC_to_gVCF -o logs/HC_to_gVCF.out -e logs/HC_to_gVCF.err --mem=8G --wrap=" \
    gatk HaplotypeCaller --java-options '-Xmx8G' \
    --input ~/tp_variant/alignment_bwa/SRR1262731_extract.sort.rg.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 logs/
$ 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

### #record headers

- variant site record
- variant site record
- variant site record

### RECORDS

### #record headers

- non-var block record
- variant site record
- non-var block record
- variant site record
- non-var block record

\* Some tools may output an all-sites VCF that looks like what you can get using HC with -ERC BP\_RESOLUTION but they do not provide an accurate estimate of reference confidence.

# 1/GATK HaplotypeCaller en mode GVCF

## *Multi-sample variant calling*

```
# 2.Fusion des fichiers gVCFs en un seul gVCF
$ sbatch -J CombineGVCFs -o logs/CombineGVCFs.out -e logs/CombineGVCFs.err --mem=8G \
--wrap="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"
```

HaplotypeCaller



CombineGVCFs



GenotypeGVCFs

# 1/GATK HaplotypeCaller en mode GVCF

## *Multi-sample variant calling*

```
# 3.Détection de variants simultanée sur les 3 échantillons du gVCF
$ sbatch -J GenotypeGVCFs -o logs/GenotypeGVCFs.out -e logs/GenotypeGVCFs.err \
--mem=8G --wrap="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"

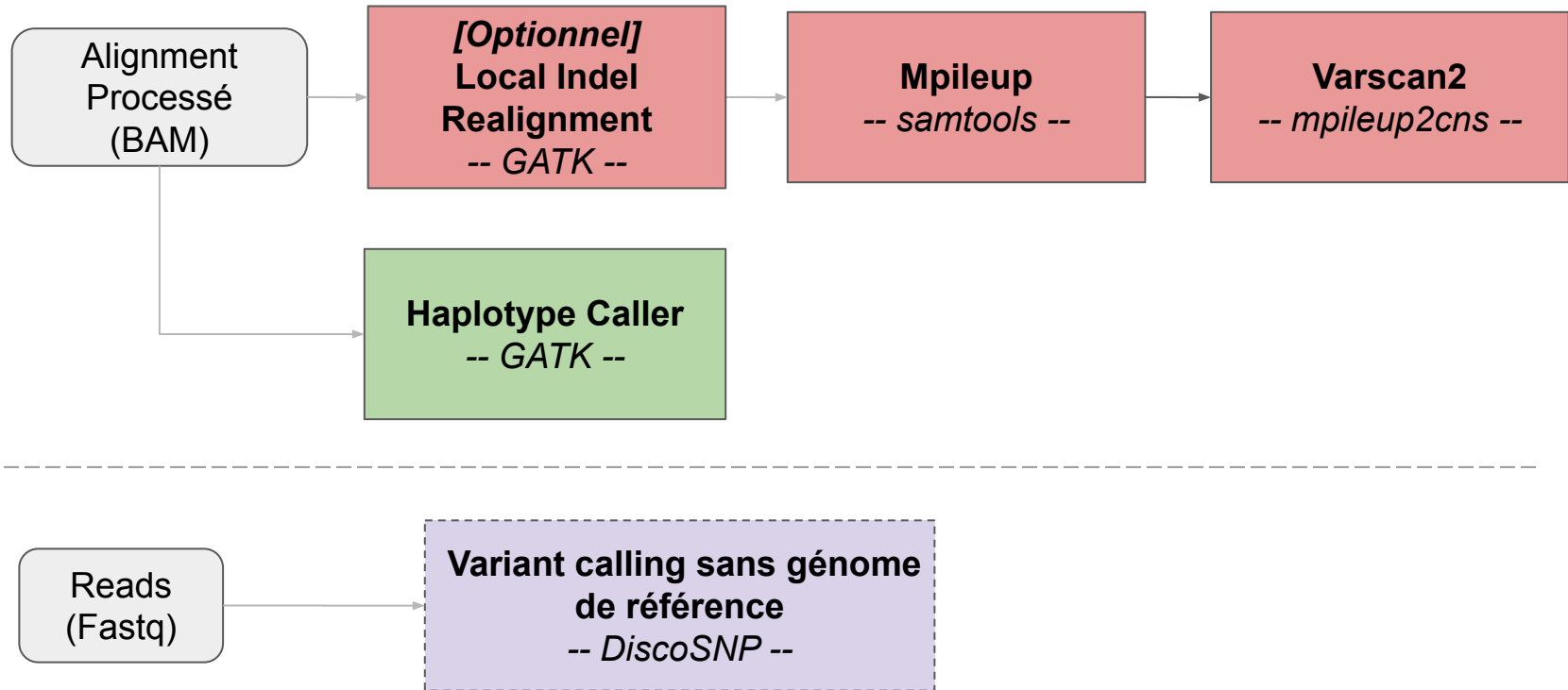
$ less -S vcf/pool_GATK.vcf
```

HaplotypeCaller

CombineGVCFs

GenotypeGVCFs

# Workflow - Variant Calling





## 2/ Samtools mpileup/Varscan2

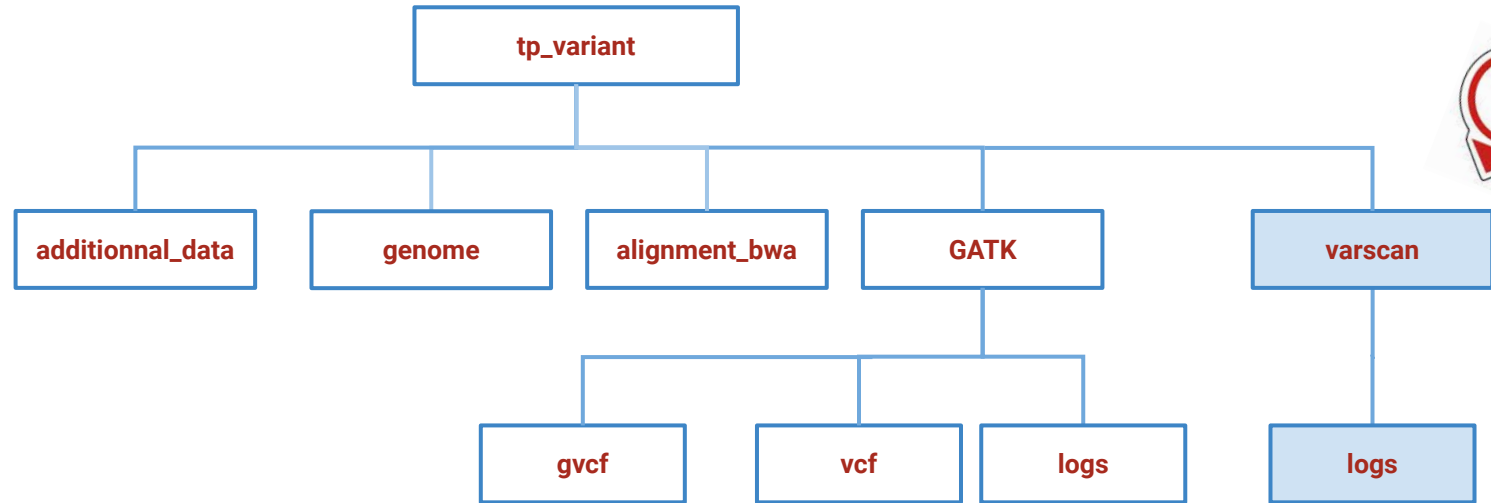
```
$ # module load samtools/1.10      # si vous ne l'avez pas déjà fait
$ samtools mpileup                  # Affichage de l'aide de samtools mpileup
Usage: samtools mpileup [options] in1.bam [in2.bam [...]]
-q, --min-MQ INT                    skip alignments with mapQ smaller than INT [0]
```

```
$ module load varscan/2.4.4
$ varscan mpileup2cns -h            # Affichage de l'aide de varscan mpileup2cns

USAGE: java -jar VarScan.jar mpileup2cns [pileup file] OPTIONS
pileup file - The SAMtools mpileup file
OPTIONS:
  --min-coverage  Minimum read depth at a position to make a call [8]
  --min-reads2    Minimum supporting reads at a position to call variants [2]
  --min-avg-qual  Minimum base quality at a position to count a read [15]
```

## 2/ Samtools mpileup/Varscan2

```
# Creation d'un nouveau dossier  
$ mkdir -p ~/tp_variant/Varscan/logs  
$ cd ~/tp_variant/Varscan
```



## 2/Samtools mpileup/Varscan2

### *Single-sample variant calling*

```
# Conversion du fichier d'alignement "bam" en format "mpileup"
# ajouter option -A pour garder les paires anormales
$ sbatch -J mpileup -o logs/mpileup.out -e logs/mpileup.err --mem=8G --wrap=" \
    samtools mpileup -q 30 -B -A -d 10000 \
    -f ~/tp_variant/genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa \
    ~/tp_variant/alignment_bwa/SRR1262731_extract.sort.rg.md.filt.onTarget.bam \
    > SRR1262731_extract.mpileup"

# Détection de variants avec Varscan
$ sbatch -J mpileup2cns -o logs/mpileup2cns.out -e logs/mpileup2cns.err --mem=8G \
--wrap=" varscan mpileup2cns SRR1262731_extract.mpileup \
    --output-vcf --variants --min-avg-qual 18 > SRR1262731_extract_Varscan.vcf"
```

## 2/Samtools mpileup/VarScan2

### *Multi-sample variant calling*

```
$ module load bcftools/1.10.2
$ bgzip ; tabix ; bcftools # v1.10.2

# Renommer l'échantillon dans le VCF
$ sed -i 's|Sample1|SRR1262731.VarScan|g' SRR1262731_extract_VarScan.vcf

# Compression et indexation du fichiers vcf
$ bgzip -c SRR1262731_extract_VarScan.vcf > SRR1262731_extract_VarScan.vcf.gz
$ tabix -p vcf SRR1262731_extract_VarScan.vcf.gz

# Merge des trois échantillons appelés avec VarScan
$ sbatch -J VarScanMerge -o logs/VarScanMerge.out -e logs/VarScanMerge.err --wrap=" \
    bcftools merge SRR1262731_extract_VarScan.vcf.gz \
    ~/tp_variant/additional_data/SRR1205992_extract_VarScan.vcf.gz \
    ~/tp_variant/additional_data/SRR1205973_extract_VarScan.vcf.gz \
    > pool_VarScan.vcf"
```

## 2/Samtools mpileup/Varscan2

### *Multi-sample variant calling*

```
# Correction du header ($ grep contig pool_Varscan.vcf)
$ sbatch -J updateSeqDict -o logs/updateSeqDict.out -e logs/updateSeqDict.err \
--wrap=" gatk UpdateVcfSequenceDictionary -I pool_Varscan.vcf \
        -O pool_Varscan_dict.vcf \
        -SD ~/tp_variant/genome/Bos_taurus.UMD3.1.dna.toplevel.6.dict"

$ less -S pool_Varscan_dict.vcf
```

# VCF Multi-échantillons

**VCF header**

```
##fileformat=VCFv4.0
##fileDate=20100707
##source=VCFtools
##reference=NCBI36
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality (phred score)">
##FORMAT=<ID=GL,Number=3,Type=Float,Description="Likelihoods for RR,RA,AA genotypes (R=ref,A=alt)">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##ALT=<ID=DEL,Description="Deletion">
##INFO=<ID=SVTYPE,Number=1,Type=String,Description="Type of structural variant">
##INFO=<ID=END,Number=1,Type=Integer,Description="End position of the variant">
```

**Mandatory header lines**

**Optional header lines** (meta-data about the annotations in the VCF body)

**Body**

#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	SAMPLE1	SAMPLE2
1	1	.	ACG	A,AT	.	PASS	.	GT:DP	1/2:13	0/0:29
1	2	rs1	C	T,CT	.	PASS	H2;AA=T	GT:GQ	0/1:100	2/2:70
1	5	.	A	G	.	PASS	.	GT:GQ	1/0:77	1/1:95
1	100	.	T	<DEL>	.	PASS	SVTYPE=DEL;END=300	GT:GQ:DP	1/1:12:3	0/0:20

**Reference alleles (GT=0)**

**Alternate alleles (GT>0 is an index to the ALT column)**

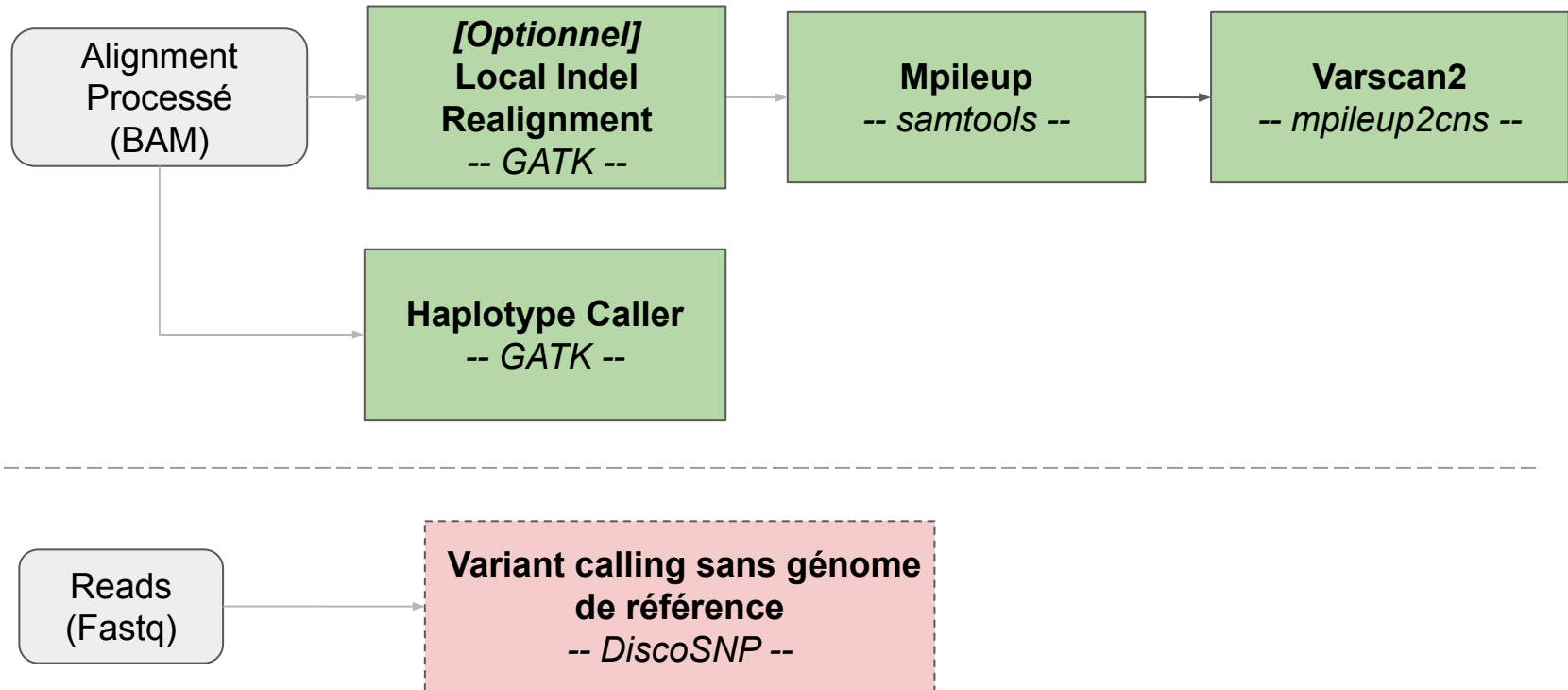
**Deletion**

**SNP**

**Insertion**

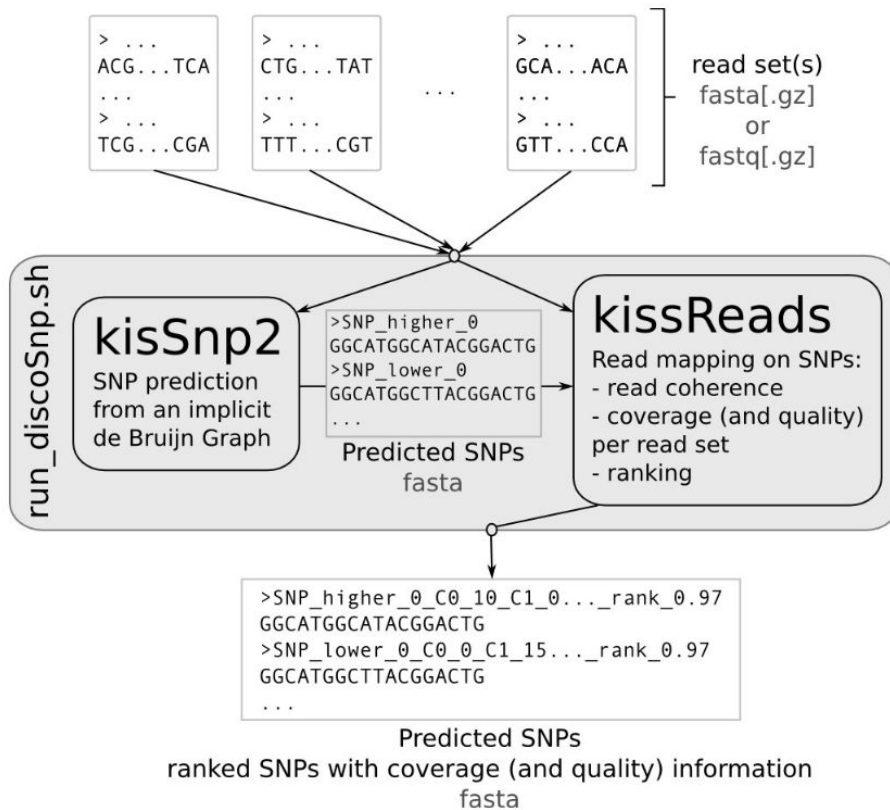
**Other event**

# Workflow - Variant Calling



# Variant Calling sans génome de référence

**Discosnp++** : **discovering**  
**Single Nucleotide Polymorphism**  
(SNP) and Indels from raw  
set(s) of reads





# Recall/Precision

		Reference variant set	
		Positive	Negative
Variants Called by the Algorithm	Positive	<b>True Positive (TP)</b> Correct variant allele or position call.	<b>False Positive (FP)</b> Incorrect variant allele or position call.
	Negative	<b>False Negative (FN)</b> Incorrect reference genotype or no call.	<b>True Negative (TN)</b> Correct reference genotype or no call.

## Recall (sensibilité)

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

$$\rightarrow TP / (TP + FN)$$

## Precision (spécificité)

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

$$\rightarrow TN / (TN + FP)$$

# Performance de la détection de SNVs/InDels par l'APR

