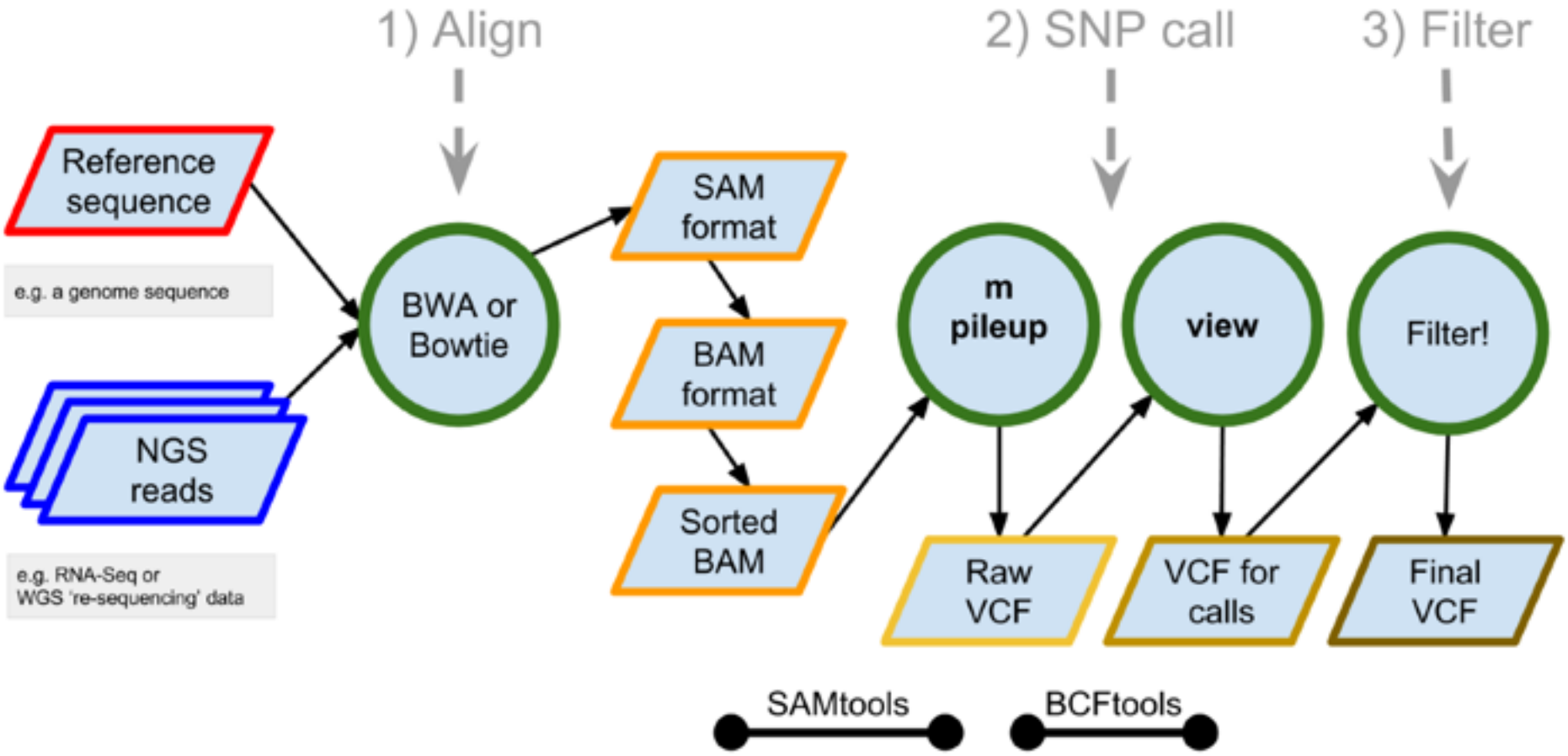
Mapping
Summary and Extension

Pipeline overview



1)Align reads to reference (using Bowtie)

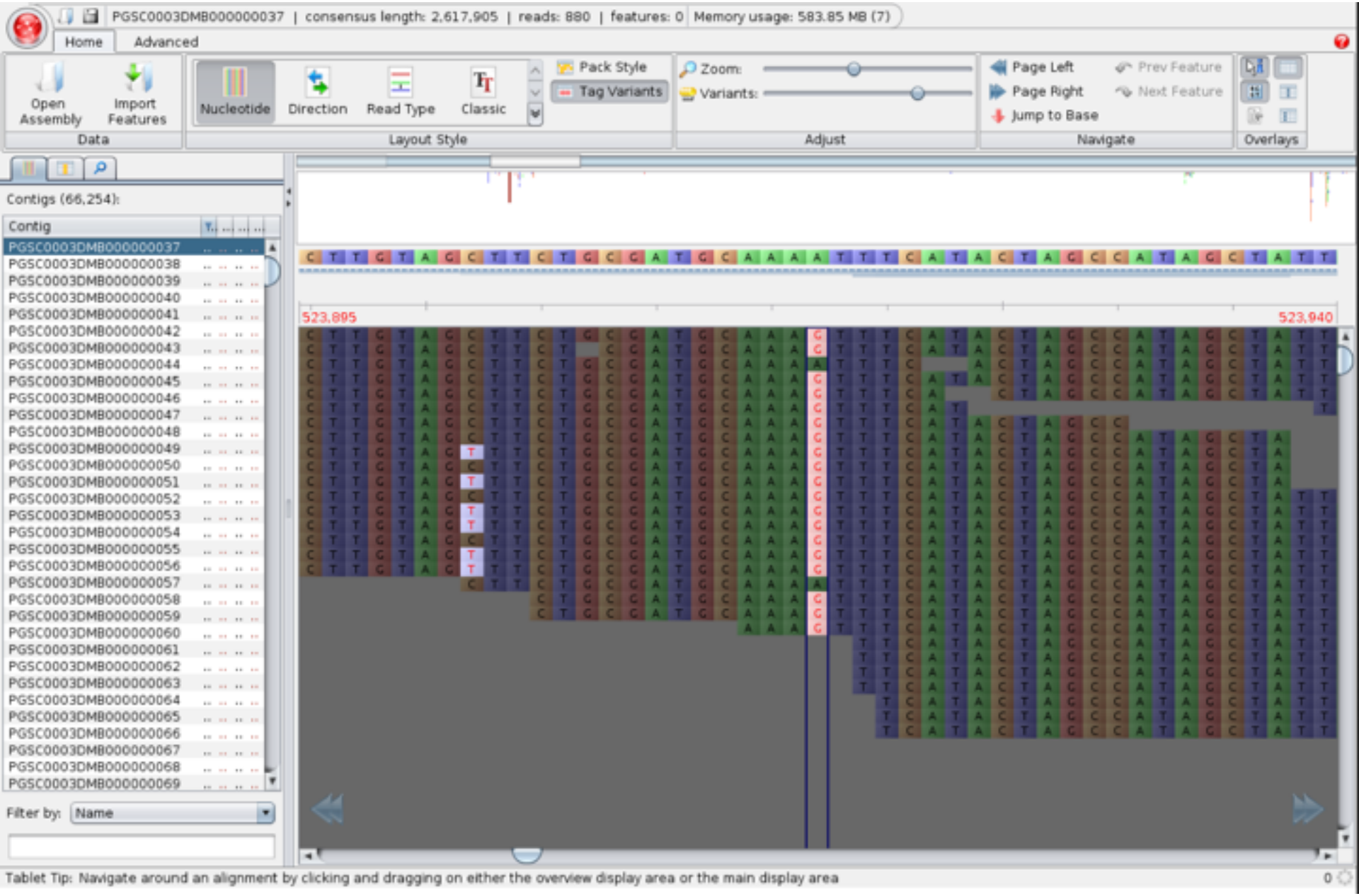
1. Index the reference sequence
    1. **bowtie2-build Clagr3_AssemblyScaffolds.fasta Clagr3_AssemblyScaffolds.fasta**

2. Mapping with bowtie2
    1. **bowtie2 -x Clagr3_AssemblyScaffolds.fasta**
    -1 Clagr3_AssemblyScaffolds.fasta.mod.art1.fq
    -2 Clagr3_AssemblyScaffolds.fasta.mod.art2.fq -S Clagr-mod.sam

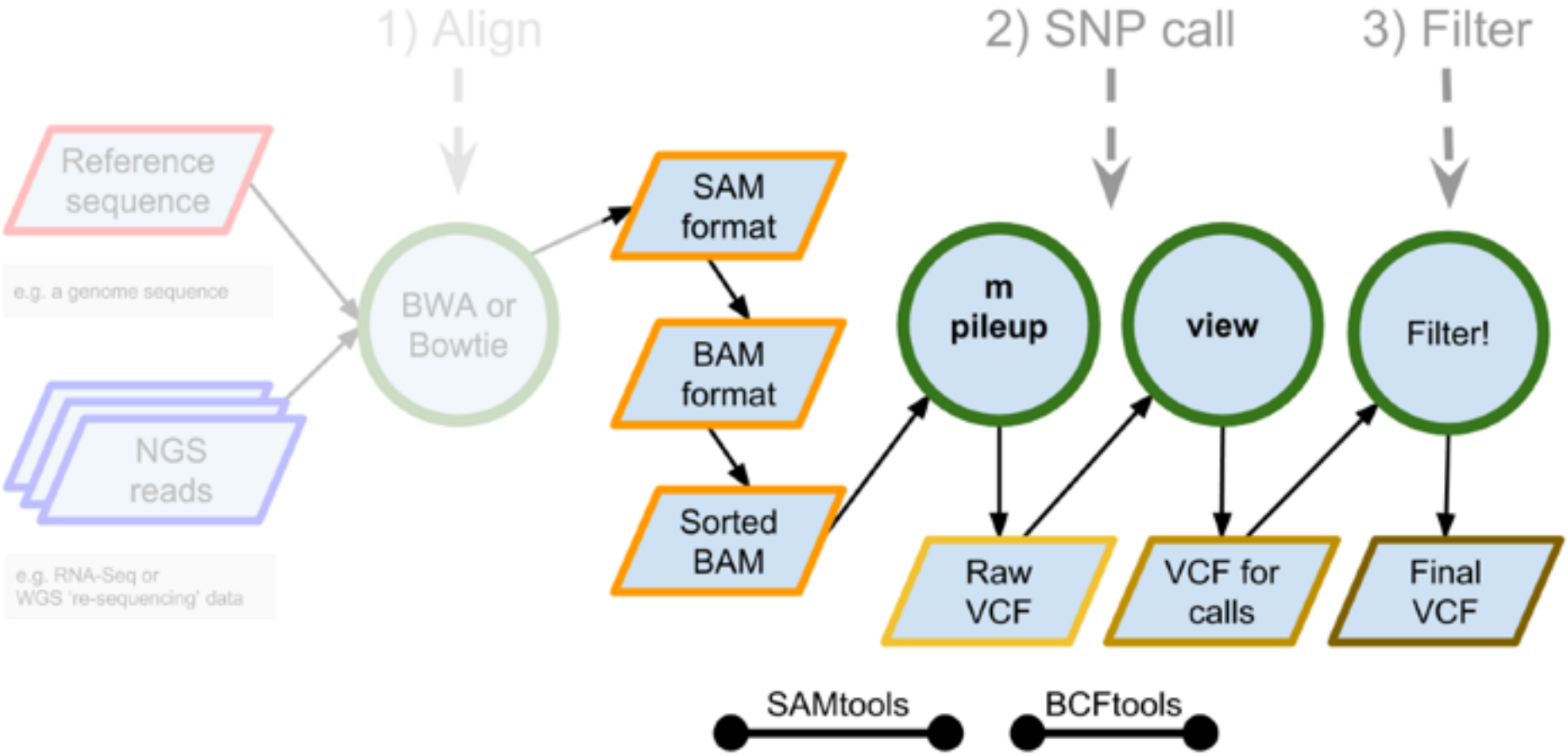
3. Convert Alignment in SAM format into binary format and sort
    1. **samtools view -bS Clagr-mod.sam |samtools sort - Clagr-mod.sorted**

4. Indexing of the sorted bam file
    1. **samtools index Clagr-mod.sorted.bam**

5. start up tablet and load the assembly


PGSC0003DMB000000037 | consensus length: 2,617,905 | reads: 880 | features: 0 Memory usage: 583.85 MB (7)
Home
Advanced
Open Assembly
Import Features
Data
Nucleotide
Direction
Read Type
Classic
Layout Style
Pack Style
Tag Variants
Zoom:
Variants:
Adjust
Page Left
Page Right
Jump to Base
Prev Feature
Next Feature
Navigate
Overlays
Contigs (66,254):
Contig
PGSC0003DMB000000037
PGSC0003DMB000000038
PGSC0003DMB000000039
PGSC0003DMB000000040
PGSC0003DMB000000041
PGSC0003DMB000000042
PGSC0003DMB000000043
PGSC0003DMB000000044
PGSC0003DMB000000045
PGSC0003DMB000000046
PGSC0003DMB000000047
PGSC0003DMB000000048
PGSC0003DMB000000049
PGSC0003DMB000000050
PGSC0003DMB000000051
PGSC0003DMB000000052
PGSC0003DMB000000053
PGSC0003DMB000000054
PGSC0003DMB000000055
PGSC0003DMB000000056
PGSC0003DMB000000057
PGSC0003DMB000000058
PGSC0003DMB000000059
PGSC0003DMB000000060
PGSC0003DMB000000061
PGSC0003DMB000000062
PGSC0003DMB000000063
PGSC0003DMB000000064
PGSC0003DMB000000065
PGSC0003DMB000000066
PGSC0003DMB000000067
PGSC0003DMB000000068
PGSC0003DMB000000069
Filter by:
Name
523,895
523,940
Tablet Tip: Navigate around an alignment by clicking and dragging on either the overview display area or the main display area
0

Alignment is done! Next, SNP calling!!



1)Remember: We have converted alignments to binary format (SAMtools) and sorted them

1. Convert SAM to BAM for sorting
   - **samtools view -S -b my.sam > my.bam**

2. Sort BAM for SNP calling
   - **samtools sort my.bam my-sorted**

The alignments are now in a format (BAM) suitable for long-term storage requiring less disk space

The alignments are now sorted to facilitate SNP calling

2)Call SNPs (using SAMtools)

1. Index the genome assembly (note, this is no longer an index based on Burrows-Wheeler transformation). See e.g. http://manpages.ubuntu.com/manpages/trusty/man5/faidx.5.html for details.
   - **samtools faidx my.fasta**

2. Run 'mpileup' to generate VCF/BCF format
   - **samtools mpileup -g -f my.fasta my-sorted-1.bam > my-raw.bcf**
   - **# NOTE: check samtools mpileup to see all options and the meaning of the individual flags**

All the has been done so far is indexing a reference sequence in fasta format to facilitate a rapid access (faidx). In addition we have converted a file in BAM format to a file in VCF/BCF format

3) Call SNPs (using bcftools)

1. Call SNPs…
   - **# Note: This call is for the bcftools version 0.x**
   - **bcftools view -bvcg my-raw.bcf > my-var.bcf**

   - **# for bcftools version 1.x use the following command**
   - **bcftools call -vc my-raw.bcf -o my-var.vcf**

A short recap:

samtools mpileup - Collects summary information in the input BAMs, computes the likelihood of the data given each possible genotype (if this option has been chosen with the flag -g), and stores the likelihoods in the BFC format.

bcftools view - applies the prior and does the actual SNP calling.

3) Optionally filter SNPs using vcfutils.pl

1. Filter SNPs…
   - **bcftools view my-var.vcf |vcfutils.pl varFilter - > my.var-final.vcf**
   - **# Note: vcfutils.pl is a perl script that helps filtering variants according to a certain set of parameters:**

Usage: vcfutils.pl varFilter [options] <in.vcf>
Options: -Q INT minimum RMS mapping quality for SNPs [$opts{Q}]
-d INT minimum read depth [$opts{d}]
-D INT maximum read depth [$opts{D}]
-a INT minimum number of alternate bases [$opts{a}]
-w INT SNP within INT bp around a gap to be filtered [$opts{w}]
-W INT window size for filtering adjacent gaps [$opts{W}]
-1 FLOAT min P-value for strand bias (given PV4) [$opts{1}]
-2 FLOAT min P-value for baseQ bias [$opts{2}]
-3 FLOAT min P-value for mapQ bias [$opts{3}]
-4 FLOAT min P-value for end distance bias [$opts{4}]
-e FLOAT min P-value for HWE (plus F<0) [$opts{e}]
-p print filtered variants

The VCF Format: An overview

```
##fileformat=VCFv4.2
##fileDate=20090805
##source=myImputationProgramV3.1
##reference=file:///seq/references/1000GenomesPilot-NCBI36.fasta
##contig=<ID=20,length=62435964,assembly=B36,md5=f126cdf8a6e0c7f379d618ff66beb2da,species="Homo sapiens",taxonomy=x>
##phasing=partial
##INFO=<ID=NS,Number=1,Type=Integer,Description="Number of Samples With Data">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##INFO=<ID=AF,Number=A,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">
#CHROM POS     ID        REF    ALT     QUAL FILTER INFO                              FORMAT      NA00001        NA00002        NA00003
20     14370   rs6054257 G      A       29   PASS   NS=3;DP=14;AF=0.5;DB;H2           GT:GQ:DP:HQ 0|0:48:1:51,51 1|0:48:8:51,51 1/1:43:5:.,.
20     17330   .         T      A       3    q10    NS=3;DP=11;AF=0.017               GT:GQ:DP:HQ 0|0:49:3:58,50 0|1:3:5:65,3   0/0:41:3
20     1110696 rs6040355 A      G,T     67   PASS   NS=2;DP=10;AF=0.333,0.667;AA=T;DB GT:GQ:DP:HQ 1|2:21:6:23,27 2|1:2:0:18,2   2/2:35:4
20     1230237 .         T      .       47   PASS   NS=3;DP=13;AA=T                   GT:GQ:DP:HQ 0|0:54:7:56,60 0|0:48:4:51,51 0/0:61:2
20     1234567 microsat1 GTC    G,GTCT  50   PASS   NS=3;DP=9;AA=G                    GT:GQ:DP    0/1:35:4       0/2:17:2       1/1:40:3
```

This example shows (in order): a good simple SNP, a possible SNP that has been filtered out because its quality is below 10, a site at which two alternate alleles are called, with one of them (T) being ancestral (possibly a reference sequencing error), a site that is called monomorphic reference (i.e. with no alternate alleles), and a microsatellite with two alternative alleles, one a deletion of 2 bases (TC), and the other an insertion of one base (T). Genotype data are given for three samples, two of which are phased and the third unphased, with per sample genotype quality, depth and haplotype qualities (the latter only for the phased samples) given as well as the genotypes. The microsatellite calls are unphased.

Source: VCFv4.2 (See course page)