Variant Analysis

FAIR Data Practices for Omics Analysis Workshop

University of Delaware

April 21 (Day 4)

An NIH sponsored program
Chemistry
Biology
Interface
at the University of Delaware

Center for
Bioinformatics &
Computational
Biology

How do we detect variants?

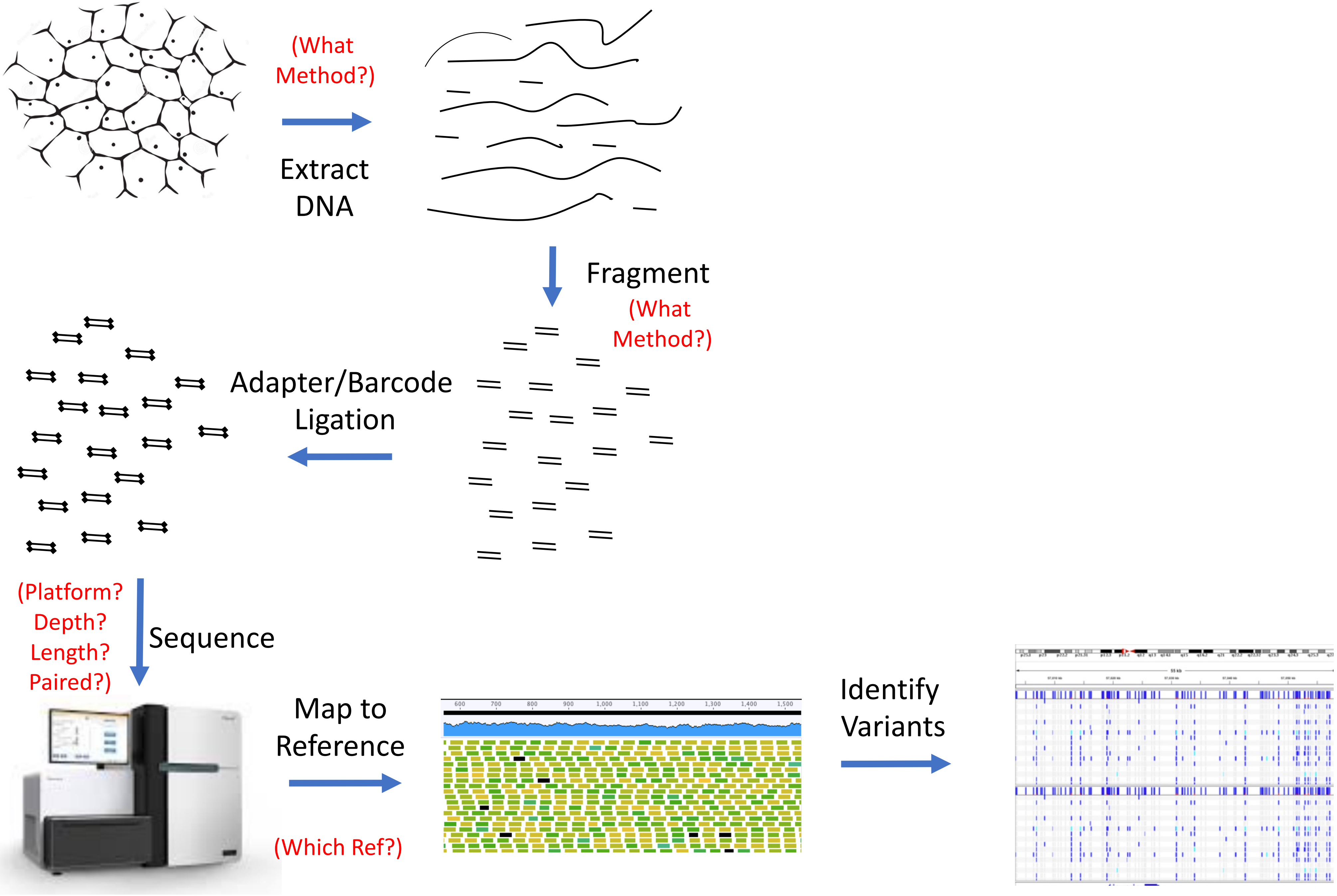
Genome Resequencing

- Detection of small genomic variants like SNV's and InDels is usually done through Genome Resequencing
(imprecisely referred to as DNA-Seq)
  - Start with a known reference genome
  - Sequence sample(s) of interest
  - Map Reads to reference
  - Identify differences

- To compare two samples to each other in this manner, you compare each to the reference genome and then look for differences

Typical Small Variant Analysis

- **Design Experiment**
- **Sample and Library Preparation**
- **Sequencing**
- Assess Sequence Quality
- Trim and Filter Reads
- Map Reads to Reference
- Identify Variants
- Explore and Verify Variants
  - Classify and Annotate Variants
  - Contextualize
  - Back to the lab: PCR and Sanger



(What Method?)

Extract DNA

Fragment

(What Method?)

Adapter/Barcode Ligation

(Platform? Depth? Length? Paired?)

Sequence

Map to Reference

(Which Ref?)

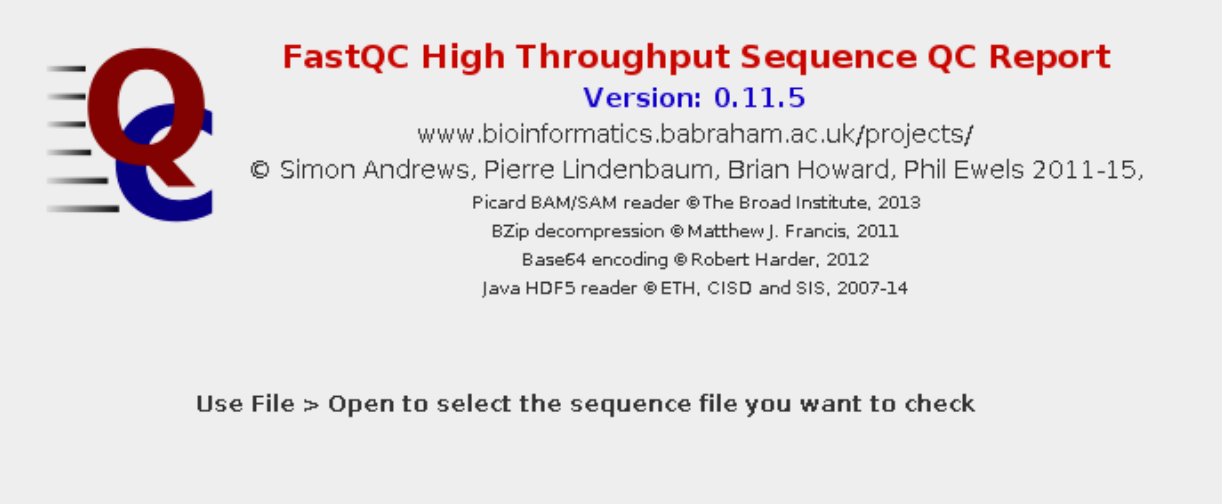
Identify Variants

Typical Small Variant Analysis

- Design Experiment
- Sample and Library Preparation
- Sequencing
- **Assess Sequence Quality**
- Trim and Filter Reads
- Map Reads to Reference
- Identify Variants
- Explore and Verify Variants
  - Classify and Annotate Variants
  - Contextualize
  - Back to the lab: PCR and Sanger

Quality Control

- FASTQ – similar to RNA-Seq
- GC should be a normal curve
- High number of duplicate reads is generally bad
- Not likely to see base composition skew at beginning



FastQC High Throughput Sequence QC Report

Version: 0.11.5

www.bioinformatics.babraham.ac.uk/projects/

© Simon Andrews, Pierre Lindenbaum, Brian Howard, Phil Ewels 2011-15,

Picard BAM/SAM reader ©The Broad Institute, 2013

BZip decompression ©Matthew J. Francis, 2011

Base64 encoding ©Robert Harder, 2012

Java HDF5 reader ©ETH, CISD and SIS, 2007-14

Use File > Open to select the sequence file you want to check

Typical Small Variant Analysis

- Design Experiment
- Sample and Library Preparation
- Sequencing
- Assess Sequence Quality
- **Trim and Filter Reads**
- Map Reads to Reference
- Identify Variants
- Explore and Verify Variants
  - Classify and Annotate Variants
  - Contextualize
  - Back to the lab: PCR and Sanger

Trimming and Filtering

- Even if your data is high quality as a whole, it may have reads of regions of reads which are lower quality
- Cutadapt/TrimGalore! and Trimmomatic are popular choices for trimming and filtering Illumina data
  - Remove low quality read ends
  - Remove adapter sequences
  - Filter out short sequences, sequences with overall low quality, or excessive ambiguous bases (e.g. N's)

Typical Small Variant Analysis

- Design Experiment
- Sample and Library Preparation
- Sequencing
- Assess Sequence Quality
- Trim and Filter Reads
- **Map Reads to Reference**
- Identify Variants
- Explore and Verify Variants
  - Classify and Annotate Variants
  - Contextualize
  - Back to the lab: PCR and Sanger

Reference Genome

- The completeness, quality, and annotations available for the reference genome can affect the approach (sequencing and bioinformatics)

- A perfect reference genome would represent the Major Alleles for the entire population . . . not always true (or for some genomes even usually)

- If no reference is available you can attempt a *de novo* assembly to make a reference, <u>very challenging </u>. . .

Reference Mapping

- Once reads are trimmed the next step is to map them onto the reference
- Similar software to RNA-Seq reference mapping . . . But doesn't need to be splice aware
- Bowtie2 is the genome mapping equivalent of TopHat2
- bwa (mem algorithm) is another popular mapper for DNA-Seq and has become the recommended standard for many workflows

- Structural variants: would need to allow disconcordant mapping

Typical Small Variant Analysis

- Design Experiment
- Sample and Library Preparation
- Sequencing
- Assess Sequence Quality
- Trim and Filter Reads
- Map Reads to Reference
- **Identify Variants**
- Explore and Verify Variants
    - Classify and Annotate Variants
    - Contextualize
    - Back to the lab: PCR and Sanger

Variant Detection

- Turns out this is a lot more complex than just comparing the read to the reference
- SAMtools and other packages can do that naïve comparison . . . But they will detect many, many, many false variants
  - Sequencing error
  - Bad mapping
  - InDels cause alignment issues
  - Incomplete penetrance of mutations
  - Mixed cell types
  - Ploidy and multi-copy genes

```
ACTAAT      ACTAAT
|||-|| OR   ||||-|
ACT-AT      ACTA-T

ACTAGT      ACTAGT
||||-| OR   |||-X|
ACTA-T      ACT-AT
```

Variant Detection

- Variant calling software needs to employ sophisticated strategies to overcome such issues and identify likely variants
  - Read depth masking (too much or too little is bad)
  - Allelic/multi-copy balance vs sequencing error
  - Insufficient coverage/masking vs no variant (comparing samples)

- Also such software usually also does double-duty as haplotype caller: 0/0, 0/1, 1/1, 1/2, 2/2, 0/2, 1/3, etc (can be separated like this 0|0)

- GATK (Genome Analysis Toolkit) has become the dominant variant detection software, especially for Eukaryotic organisms

GATK Best Practices

- GATK posts extensive information on best practices for performing variant identification: https://software.broadinstitute.org/gatk/best-practices/
- Best practices shift over time, so this is always a good resource to review before doing an analysis

Microbial Variant Detection

- For “Prokaryotes” the process is a little more straight forward? Why?

- We still have issues like sequencing error that need to be dealt with…but lack complexities of dealing with heterozygous alleles

- So we can use faster and more streamlined tools like Snippy…which we will use for our project

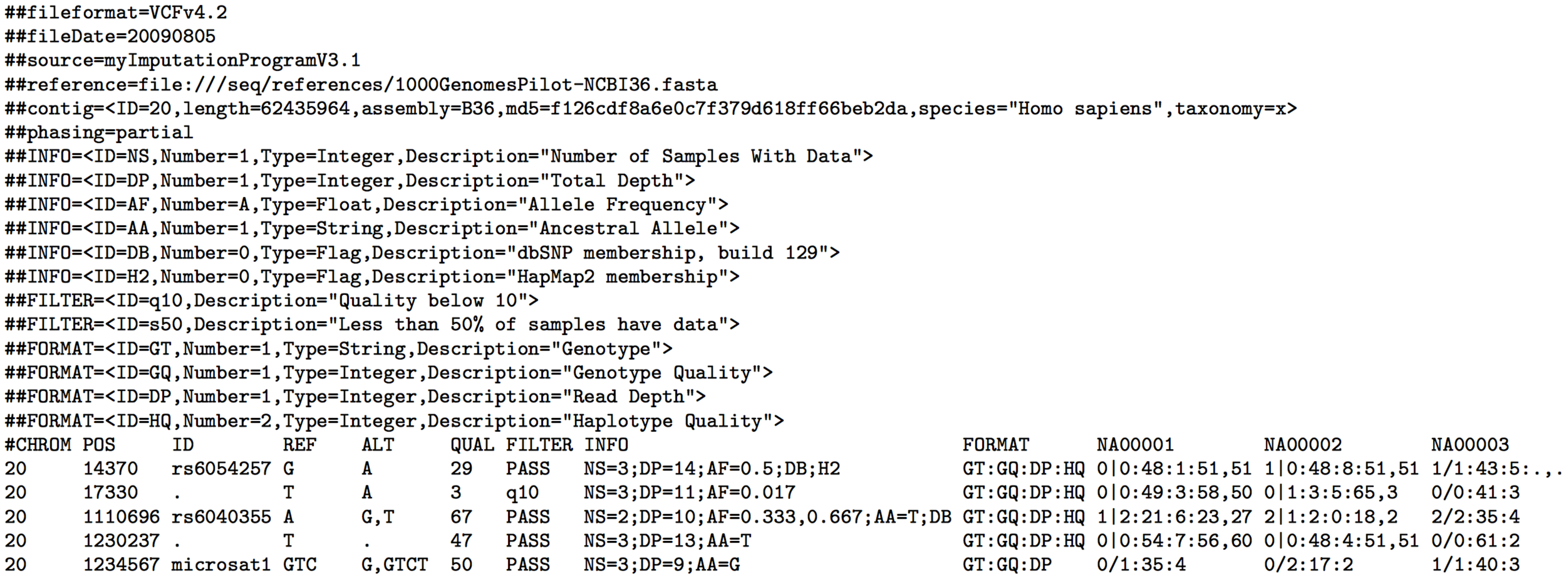
- Snippy wraps the steps of reference mapping, variant calling, and variant filtering into a single step!

```
ACTAAT      ACTAAT
|||-||  OR  ||||-|
ACT-AT      ACTA-T

ACTAGT      ACTAGT
||||-|  OR  |||-X|
ACTA-T      ACT-AT
```

VCF File Format

- https://samtools.github.io/hts-specs/VCFv4.2.pdf



```
##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
```