HapMap3: large-scale genotyping for *Zea mays*

Robert Bukowski
Bioinformatics Facility
(aka Computational Biology Service Unit)
Cornell University

4/28/2014

Objective:

Identify polymorphisms in *Zea mays* from WGS data

- Fits within the broad context of maize diversity research (Buckler Lab + others) aimed at identifying functional polymorphisms
- continuation of HapMap1, HapMap2
- sequence data from ~1,000 lines representing most of maize diversity

Challenge

- Maize is a very diverse species (10-20 times more diverse than human), yet only one reference genome available (B73)
    - Hard to align
    - A lot of false polymorphisms expected from misalignments
    - Specialized variant filtering strategy needed

Outline of the talk:

Datasets and size of the project

Pipeline overview

- Alignment
- Pileup
- Genotyping
- Variant filtering (release HMP v3.0)

Challenges and outlook

HapMap3: datasets

- Inbred maize lines
- teosinte lines (19)
- landraces (~100)

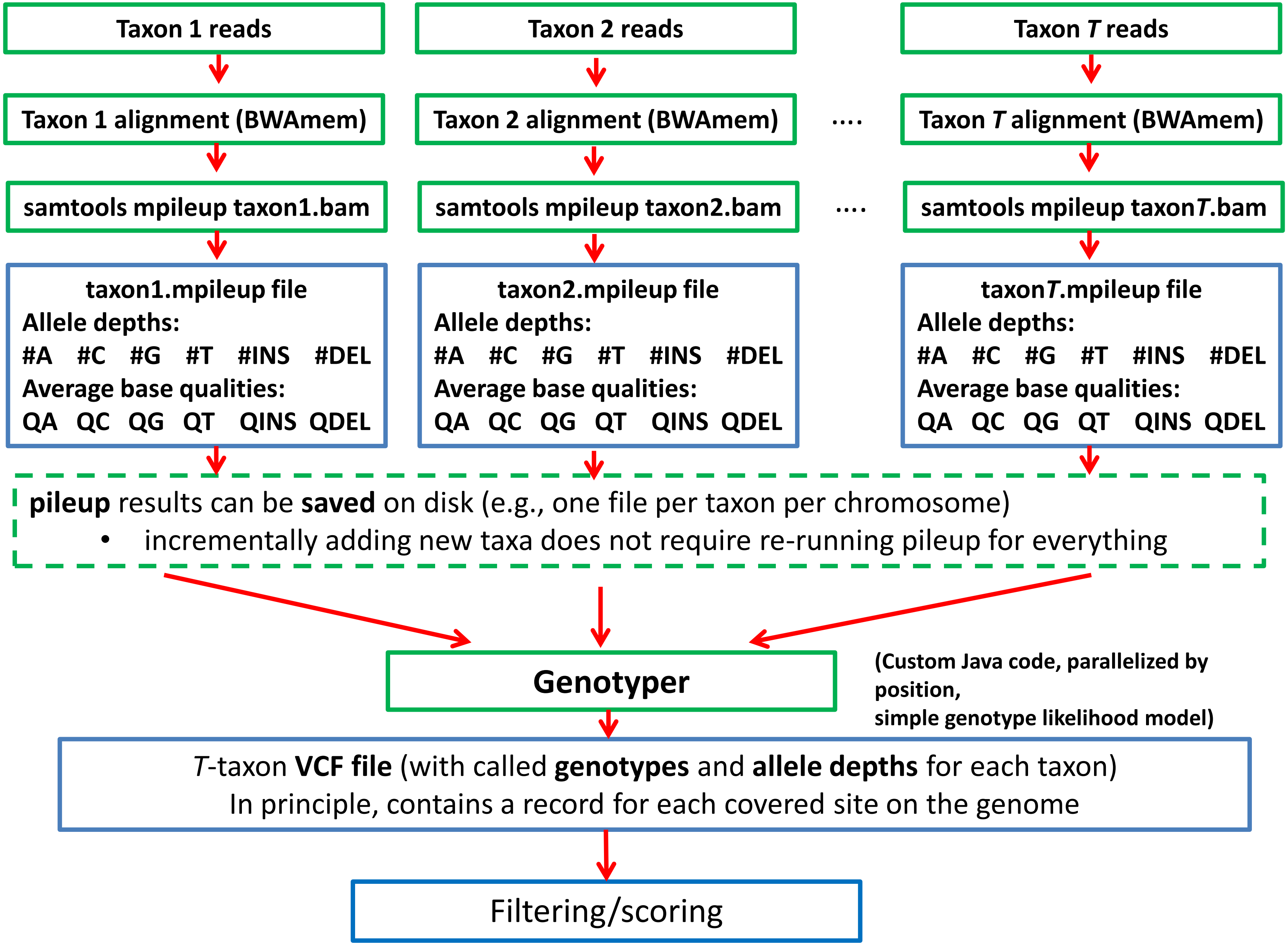
Taxa subset	# read pairs (including newly re-sequenced)	# taxa	Coverage
HapMap2	8,299,545,502	104	5-27x
Chinese Agricultural University	14,520,759,887	714	1-40x (most 1-3x)
Other (TIL25, RIMMA0438, …)	1,048,952,874	8	10-40x
CIMMYT-BGI*	12,852,099,345	89	10-21x
Total	36,721,357,608	915	

* Aligned, but not yet included in the current release HMP v3.0

Size of the project

	HapMap 3	HapMap 2
Number of taxa	915	104
Fastq files	1,758 pairs	565
Number of reads	72,221,586,212	11,393,537,138
Bases sequenced	6,826 Gbase	972.4 Gbase
Size of fastq files (compressed)	5,694 Gbytes	628 Gbytes
Size of BAM files	5,929 Gbytes	840 Gbytes
Size of extracted depths (in HDF5)	~5,000 GBytes	
Total size on disk	~17 TBytes	

Pipeline



Taxon 1 reads

Taxon 1 alignment (BWAmem)

samtools mpileup taxon1.bam

taxon1.mpileup file
Allele depths:
#A #C #G #T #INS #DEL
Average base qualities:
QA QC QG QT QINS QDEL

Taxon 2 reads

Taxon 2 alignment (BWAmem)

samtools mpileup taxon2.bam

taxon2.mpileup file
Allele depths:
#A #C #G #T #INS #DEL
Average base qualities:
QA QC QG QT QINS QDEL

....

Taxon *T* reads

Taxon *T* alignment (BWAmem)

samtools mpileup taxon*T*.bam

taxon*T*.mpileup file
Allele depths:
#A #C #G #T #INS #DEL
Average base qualities:
QA QC QG QT QINS QDEL

pileup results can be **saved** on disk (e.g., one file per taxon per chromosome)

- incrementally adding new taxa does not require re-running pileup for everything

Genotyper

(Custom Java code, parallelized by position,
simple genotype likelihood model)

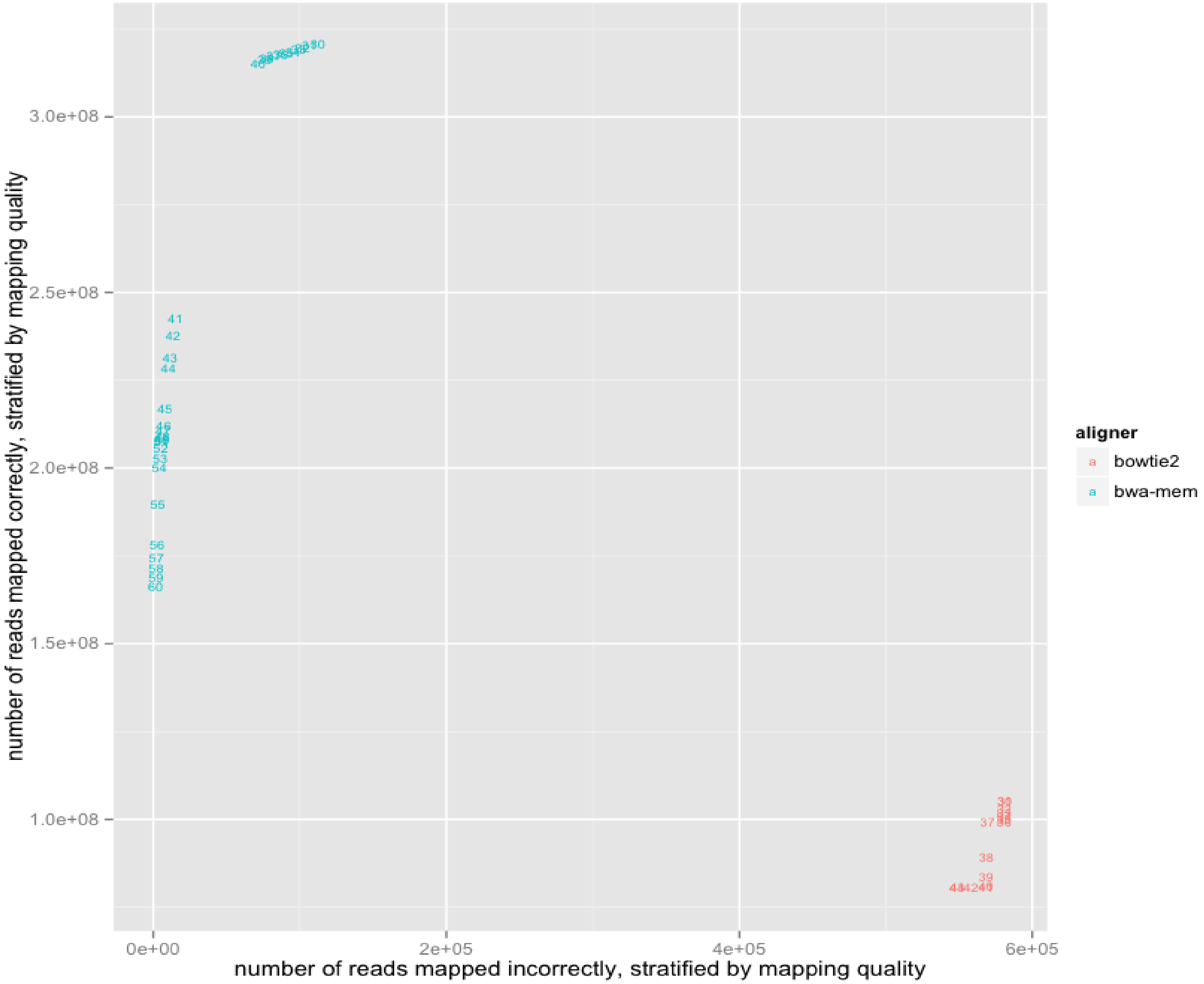
T-taxon **VCF file** (with called **genotypes** and **allele depths** for each taxon)
In principle, contains a record for each covered site on the genome

Filtering/scoring

Alignment

- Reference: **B73 AGPv2**
  - Good in algorithm development stage (e.g., comparisons with GBS variants)
  - Eventually, HapMap3 variants will be called on **v3**

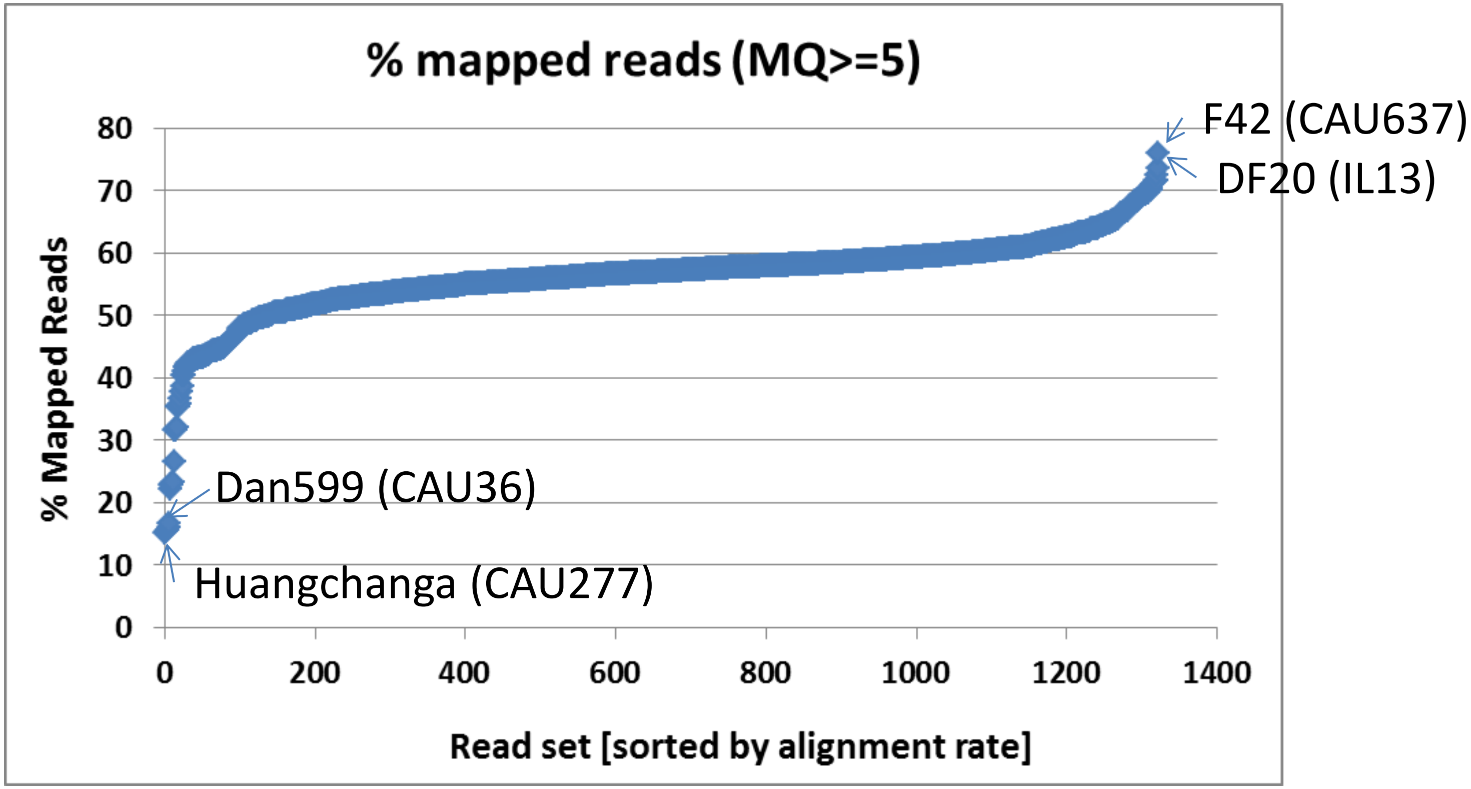
- Aligner: **BWA mem**:
  - shown to be more accurate than Bowtie2 (Vince Buffalo, UC Davies, based on simulated reads)
  - Produces about 2 times less heterozygous genotypes on inbred lines (in our own tests)

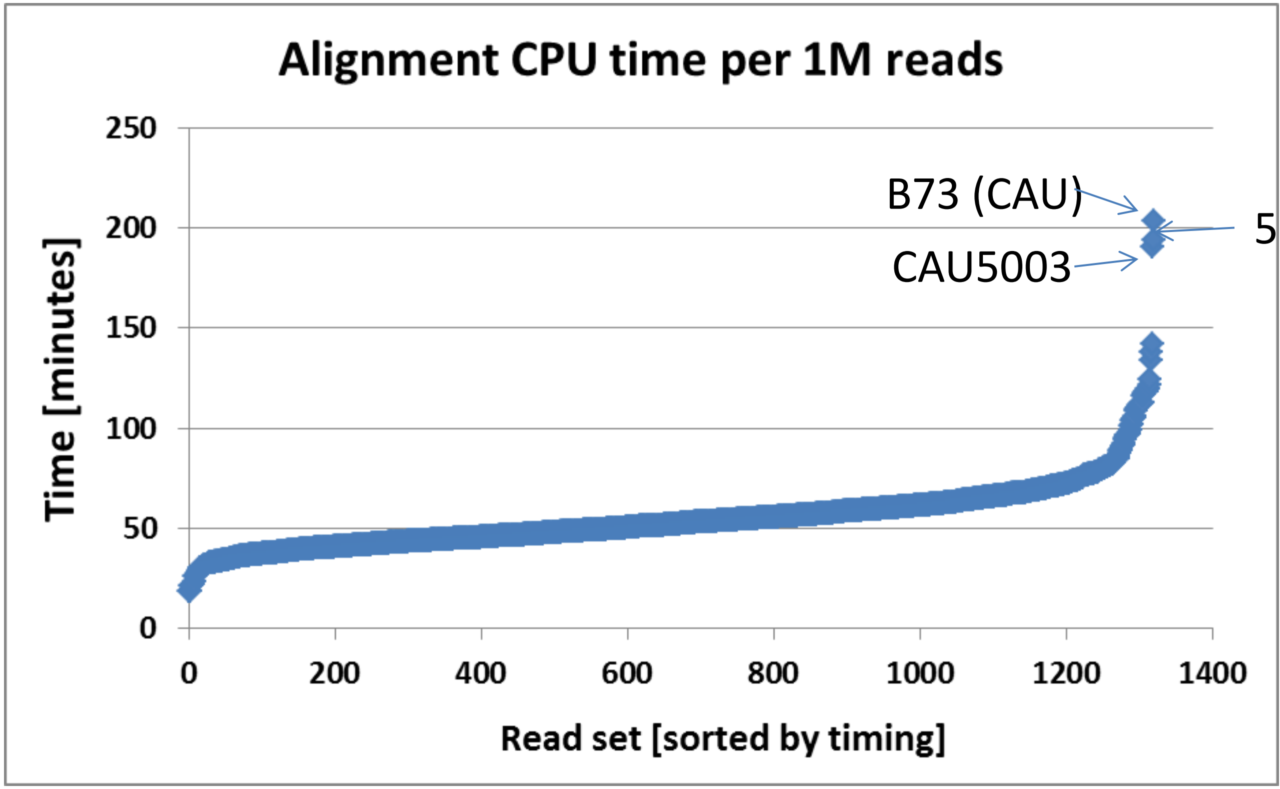


From simulation by **Vince Buffalo, UC Davis**

BWA mem (and Bowtie2) report > 95% reads as “aligned”. However, only about 50-60% align with **non-zero mapping quality**.

Only these alignments are used in variant calling.





B73 (CAU)

5003 (CAU04)

CAU5003

Overall alignment time for all taxa:
37,273 hours CPU → 373 hours on 100 CPUs = **15 days (+10 days for data transfer, pre- and post-processing, mishaps, etc.)**
Used 5 machines, two 10-thread BWA mem jobs on each

pileup

```
samtools  mpileup   taxon.bam
```

→

taxon.mpileup:
For each position, collect and store

#A #C #G #T #+ #-	allele depths
QA QC QG QT Q+ Q-	average base qualities

Position on chromosome

	1	2	3	4	5	6
A	0	3	1	0	0	....
C	3	0	1	0	0	....
G	0	1	2	0	0	....
T	0	0	0	0	0	....
I	0	0	1	0	0	....
D	0	0	0	0	0	....

Values: allele depths or average base qualities

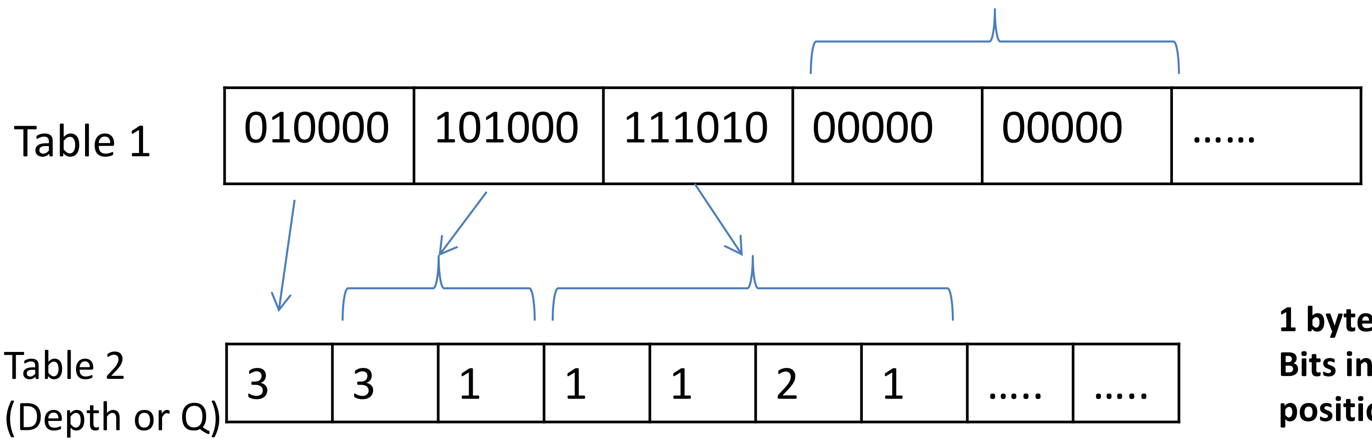
most sites are like these

a lot of contiguous coverage gaps

	1	2	3	4	5	6
A	0	3	1	0	0	....
C	3	0	1	0	0	....
G	0	1	2	0	0	....
T	0	0	0	0	0	....
I	0	0	1	0	0	....
D	0	0	0	0	0	....

Replace this….

with this….



Zero-coverage sites take NO bytes in Table 2

Table 1

010000	101000	111010	00000	00000	……

Table 2 (Depth or Q)

3	3	1	1	1	2	1	…..	…..

6 bits per position (could even use a full byte)

1 byte per nonzero depth
Bits in Table 1 determine which position and which allele the bytes correspond to

3.5 – 6 GB per taxon → 6 TB for 1000 taxa
Less effective if coverage high and/or data dirty

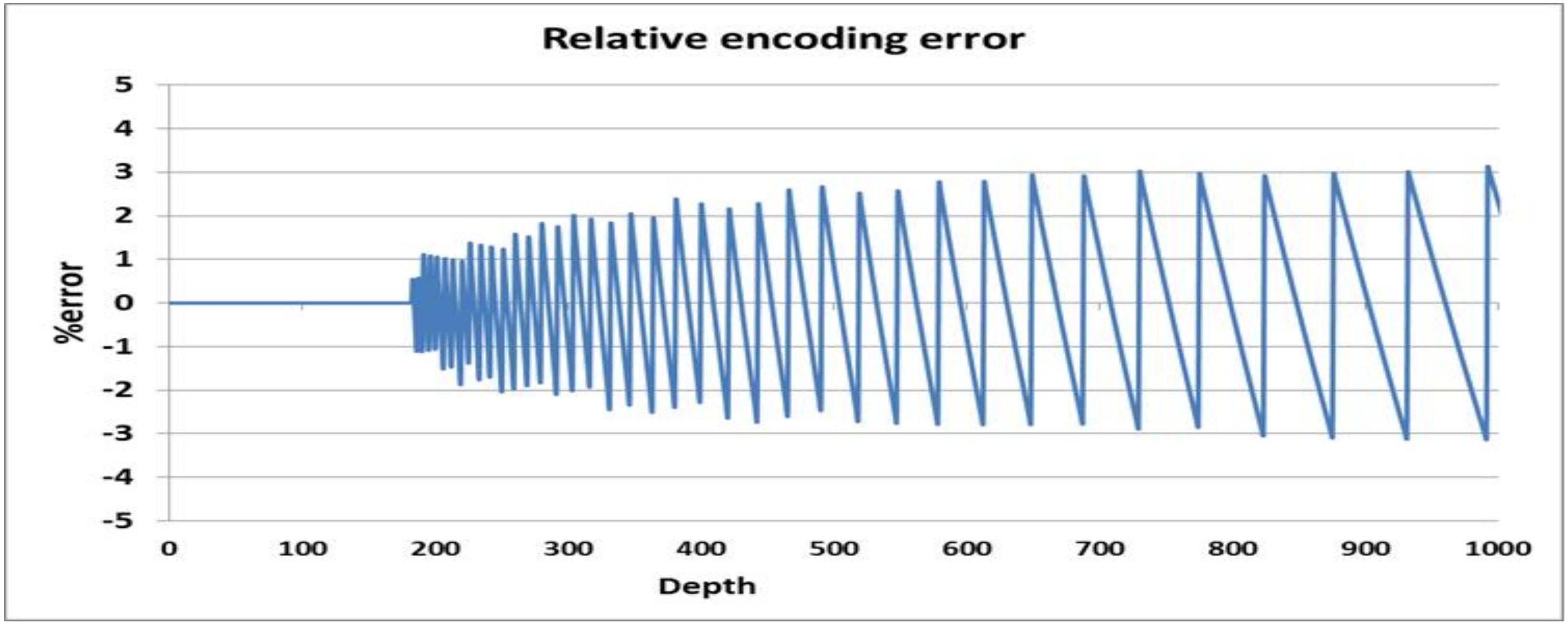
Byte representation of read depths

$$B = \begin{cases} I & \text{for } I \le 127 \\ 127 - I & \text{for } 127 < I \le M \\ \max\ [-\log_b(I - o), -128] & \text{for } I > M \end{cases}$$

$$I = \begin{cases} B & \text{for } B \ge 0 \\ 127 - B & \text{for } 0 > B \ge 127 - M \\ o + b^{0.5-B} & \text{for } B < 127 - M \end{cases}$$

where $M = 182, \quad o = 126, \quad b = 1.0746$

Depths up to 10,482 can be represented



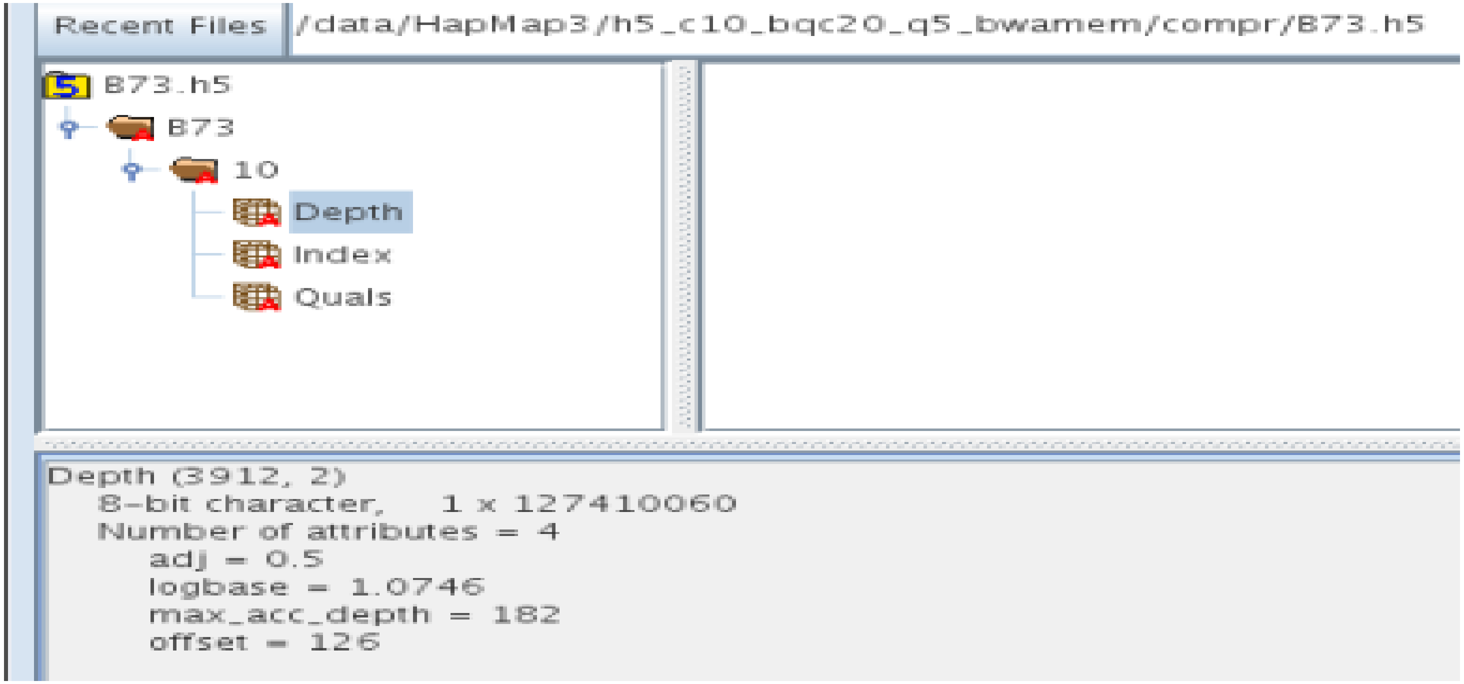
Storing pileup results

Need format/tool which

- Can bundle multiple tables with different kinds of data
- Allows some metadata (e.g., parameters of byte encoding)
- **Provide fast direct access to slices of data** (e.g., subsets of genomic positions)
- Has convenient API to use in our codes (Java

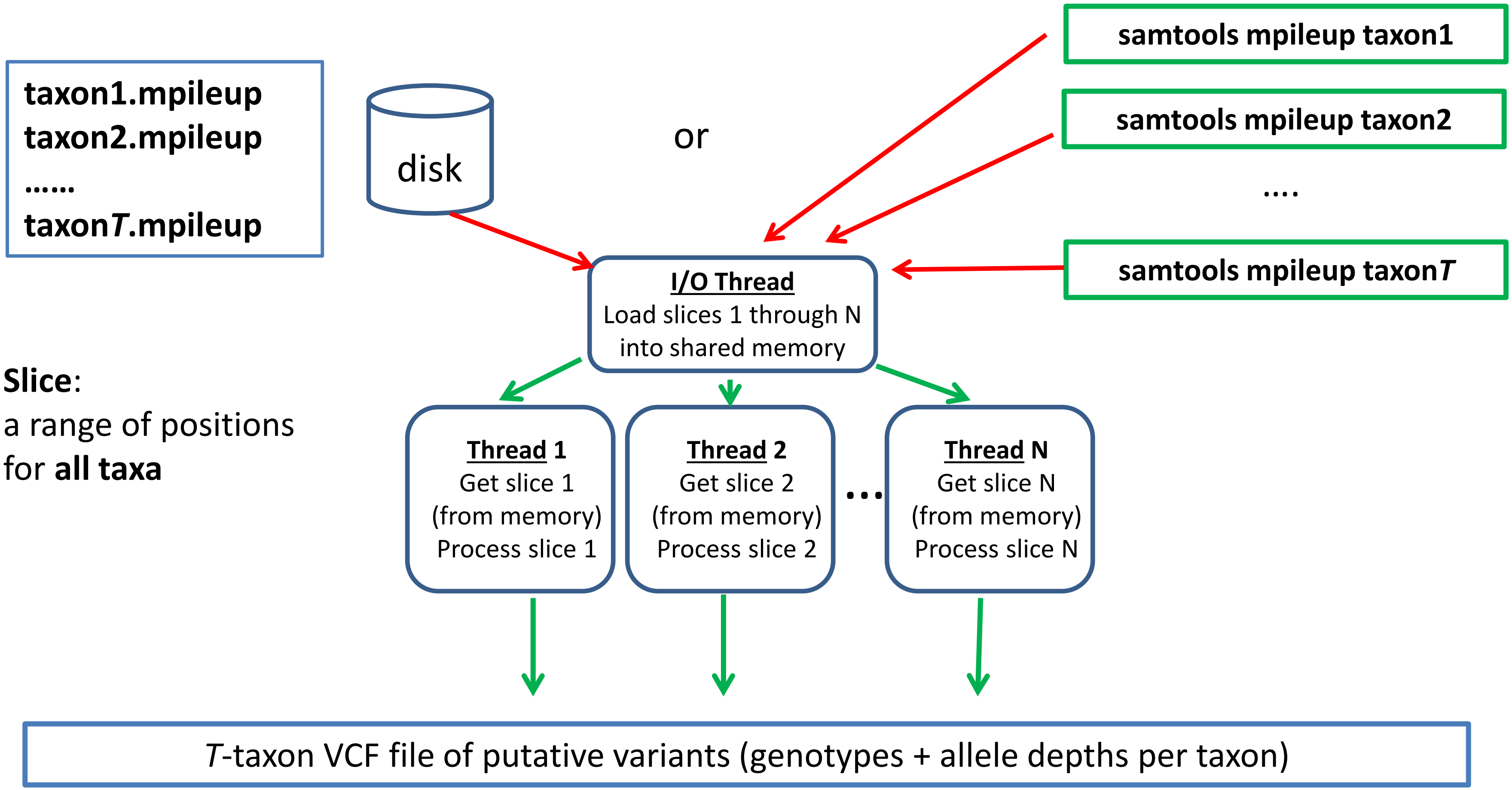
Answer: **HDF5 file format** (http://www.hdfgroup.org/HDF5/)

HDF = Hierarchical Data Format



```
Depth (3912, 2)
    8-bit character,    1 x 127410060
    Number of attributes = 4
        adj = 0.5
        logbase = 1.0746
        max_acc_depth = 182
        offset = 126
```

Genotyper



taxon1.mpileup
taxon2.mpileup
……
taxon*T*.mpileup

disk

or

samtools mpileup taxon1

samtools mpileup taxon2

….

samtools mpileup taxon*T*

I/O Thread
Load slices 1 through N
into shared memory

Slice:
a range of positions
for **all taxa**

Thread 1
Get slice 1
(from memory)
Process slice 1

Thread 2
Get slice 2
(from memory)
Process slice 2

…

Thread N
Get slice N
(from memory)
Process slice N

T-taxon VCF file of putative variants (genotypes + allele depths per taxon)

VCF format not practical if **all genomic positions** are required

Estimated VCF file size: 20 TB

Genotyper

What the Gemotyper DOES NOT do:

- Call SNPs

What the Genotyper DOES:

- For each position, summarize allele depth data over all taxa in terms of genotype calls and parameters derivable from them, do some rudimentary filtering, but be inclusive

Genotyper algorithm:

For each position on the genome:

- Skip if less than 10 taxa with coverage (optional)
- Compute depth of each allele in each taxon
- Skip if no variation detected (optional)
- Compute all genotype likelihoods for each taxon, assign genotype (0/0, 0/1, 1/1, etc)
  - Likelihoods based on a multinomial model with fixed overall error rate, independent of type of mismatch, position on read, genome, etc.
- Skip if only reference homozygotes detected (optional)
- Sort alleles according to frequency across taxa

Genotyper: simple 1-parameter genotype likelihood model

$$L(XY) = \sum_{Z=A,C,G,T,I,D} N_Z \log P(Z|XY)$$

$$P(Z|XY) = \frac{1}{2}[P(Z|X) + P(Z|Y)]$$

$$P(Z|X) = \begin{cases} 1 - \frac{5e}{6} & for\ Z = X \\ \frac{e}{6} & for\ Z \neq X \end{cases}$$

Currently: $e = 0.01$

Pick genotype X/Y with largest (most positive) $L(XY)$

Storage/distribution of genotyping results (in the future)

VCF format not practical if all genomic positions are required (20TB)

Better idea:

- Genotypes in Tassel-like bit representation
  - 3D bit table (#sites × #taxa × #alleles_per_site)
  - for 1,000 taxa and up to 2 alleles per site: 350 GB (50GB for largest chromosome)

- Depths in our compressed HDF5 format
  - one file per taxon (or per taxon, chromosome)
  - about 3.5 TB for 1,000 taxa

- Efficient extraction of data slices

- VCFtools-like tool to read/process genotypes/depths stored this way
  - TASSEL already has tools to process genotype data (but not depths)

Compilation of timing estimates (all sequence, whole genome, all taxa)

Alignment:

- 37,273 hours CPU → 373 hours on 100 CPUs = **15 days (+10 days for data transfer, pre- and post-processing, mishaps, etc.)**

Pileup:

- **About 0.32 min/taxon/1Mbp → 400 CPU-days for ~850 taxa and whole genome** parallelized over taxa and/or chromosomes

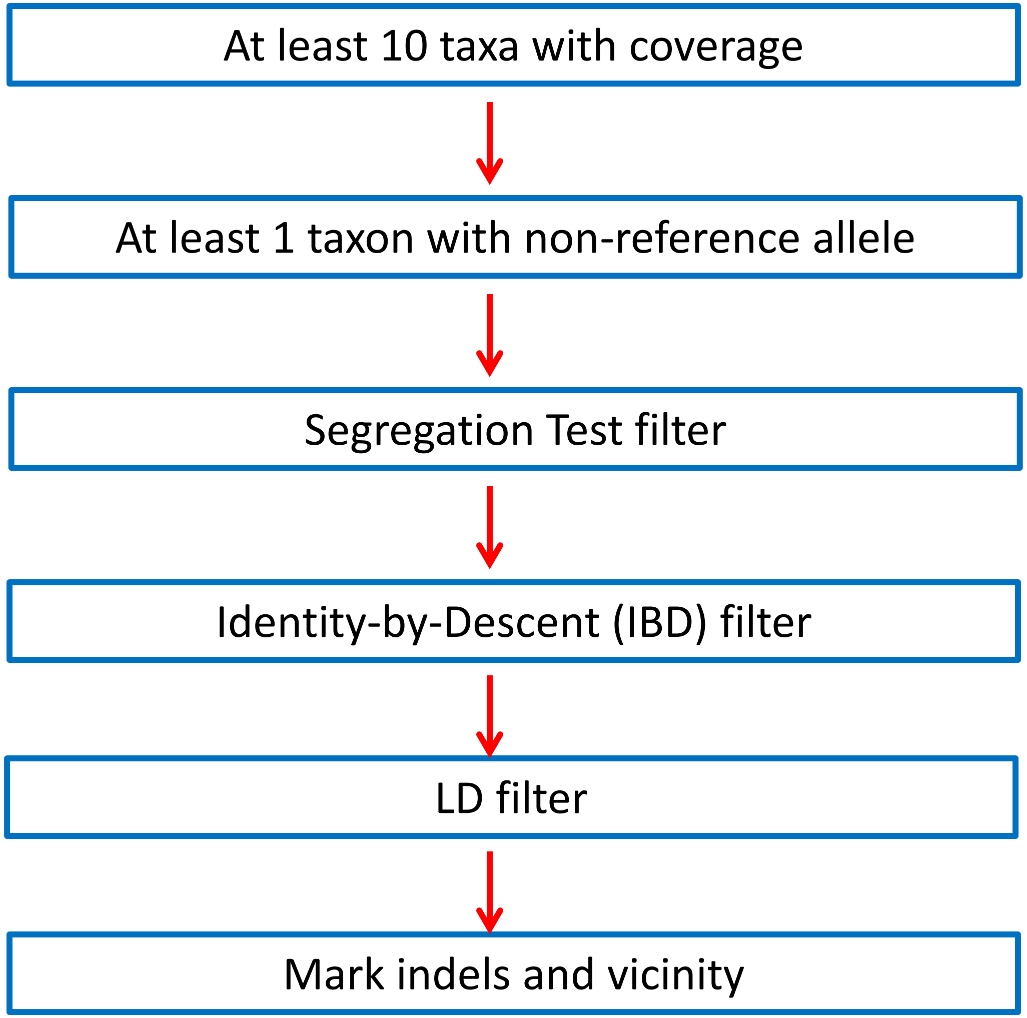
Genotyping:

- 4.5 days of HDF5 file reading (not parallelizable)
- 830 days of genotyping (17 days on 50 processors)

most of it goes into Segregation Test p-value calculation

Polymorphism filtering

(done for each site)



At least 10 taxa with coverage

At least 1 taxon with non-reference allele

Segregation Test filter

Identity-by-Descent (IBD) filter

LD filter

Mark indels and vicinity

GBS anchor

HapMap3 genotypes of 826 taxa on 955,120 GBS v2.7 SNP sites

- GBS SNPs considered reliable
- HapMap3 genotypes agree well with GBS ones (on overlapping taxa)
- Can be used to define **IBD** (Identity-by-Descent) regions within 826 taxa, or as an anchor for **LD test**

Segregation Test

For each site, construct contingency table of major/minor allele counts per taxon, e.g.,

	Taxon1	Taxon2	Taxon3	….	Taxon 826
Depth major allele	2	10	5	…	13
Depth minor allele	0	1	1	…	0

Calculate p-value of χ2 of the table given fixed row and column totals

use χ2 test first (fast)
if p-value form χ2 test >=0.2 – reject site
otherwise, run a 1000-step simulation to get a more accurate value

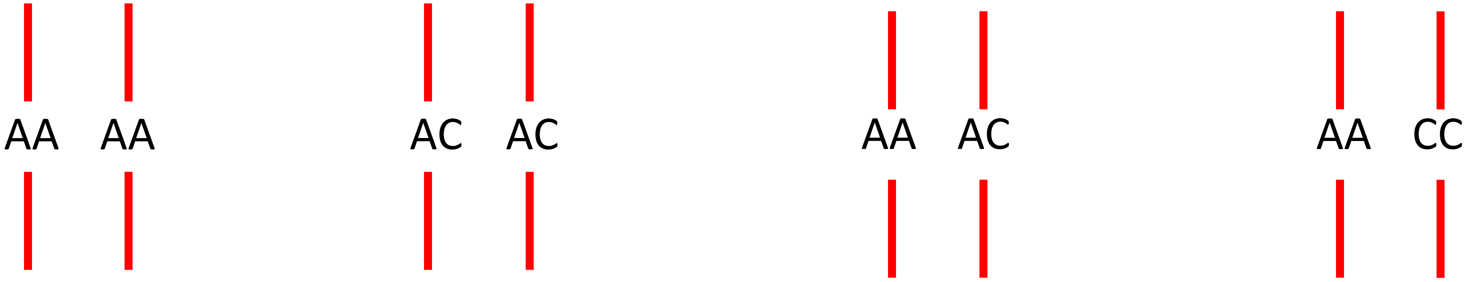
keep sites with p-value <= 0.01

ST filter tends to eliminates hets with poor read support

IBD filtering

Use **GBS anchor** to determine IBD pairs in the 826-taxa set in windows of about 5 Mb (2,000 GBS sites)

Idea: For each allele present at a given site, count number of times this allele matches between two taxa in an IBD pair and how many times it does not match; sum these over IBD pairs, e.g.,



AA AA AC AC AA AC AA CC

Match(A)	=	4	+	2	+	3	+	0	=	9
Mismatch(A)	=	0	+	0	+	0	+	2	=	2
Match(C)	=	0	+	2	+	0	+	0	=	2
Mismatch(C)	=	0	+	0	+	1	+	2	=	3

Keep sites for which

□ Match/Mismatch >=2 for both alleles

or

□ Only one allele present in all IBD pairs

LD filter

❑ For **each site**, (try to) compute LD with **each site of the GBS anchor map** (all chromosomes)

❑ LD measure used: p-value from Fisher Exact Test on a 2 X 2 table of taxa counts corresponding to 4 haplotypes (AB,Ab,aB,ab)

❑ hets treated as minor allele homozygotes

❑ Site pair tested only if

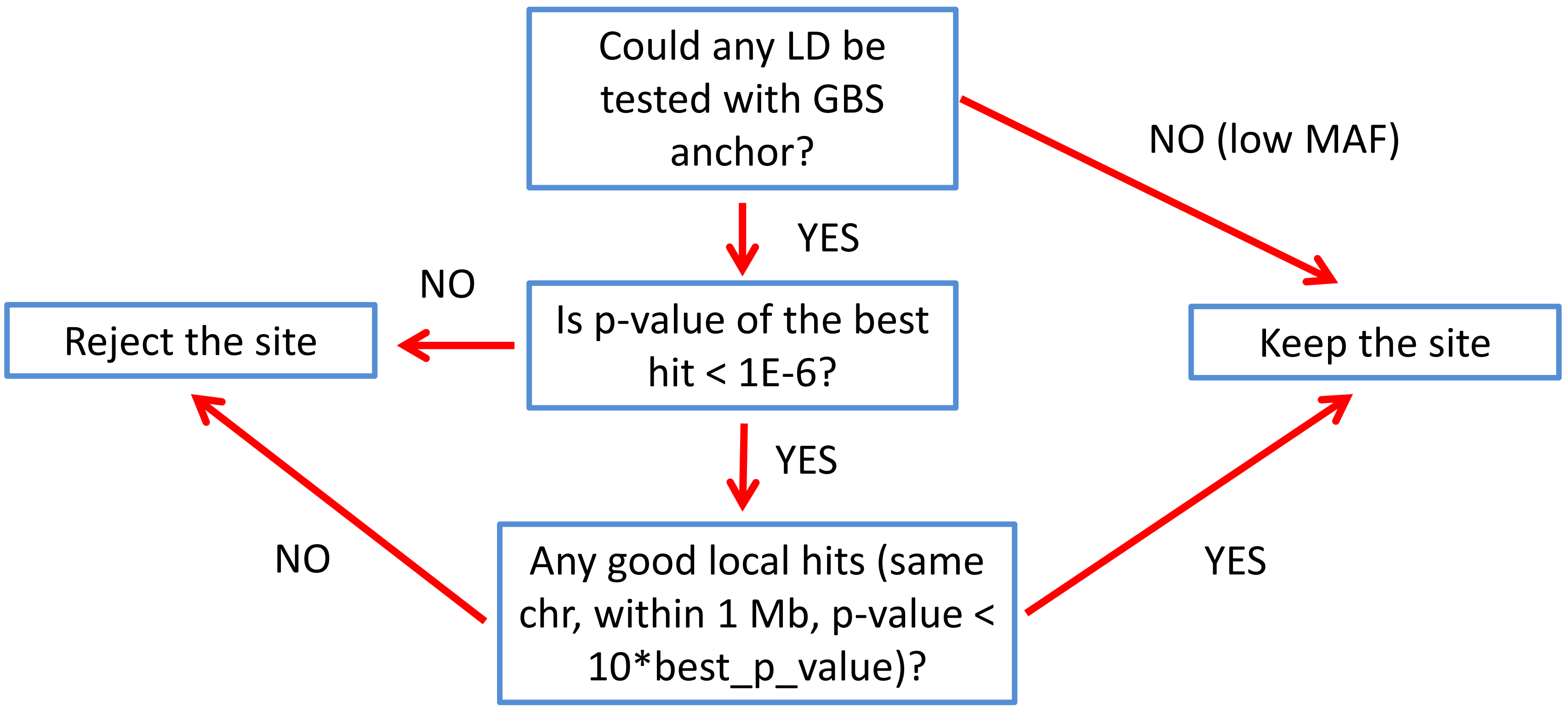
  ❑ the two sites at least 2,500 bp apart

  ❑ at least 40 taxa present with non-missing genotypes at both sites

  ❑ at least 2 taxa with minor allele present at each site

❑ Collect a number of best hits (p-values, R2, locations) for each site

LD filter



Could any LD be tested with GBS anchor?

YES

Is p-value of the best hit < 1E-6?

YES

Any good local hits (same chr, within 1 Mb, p-value < 10*best_p_value)?

NO (low MAF)

NO

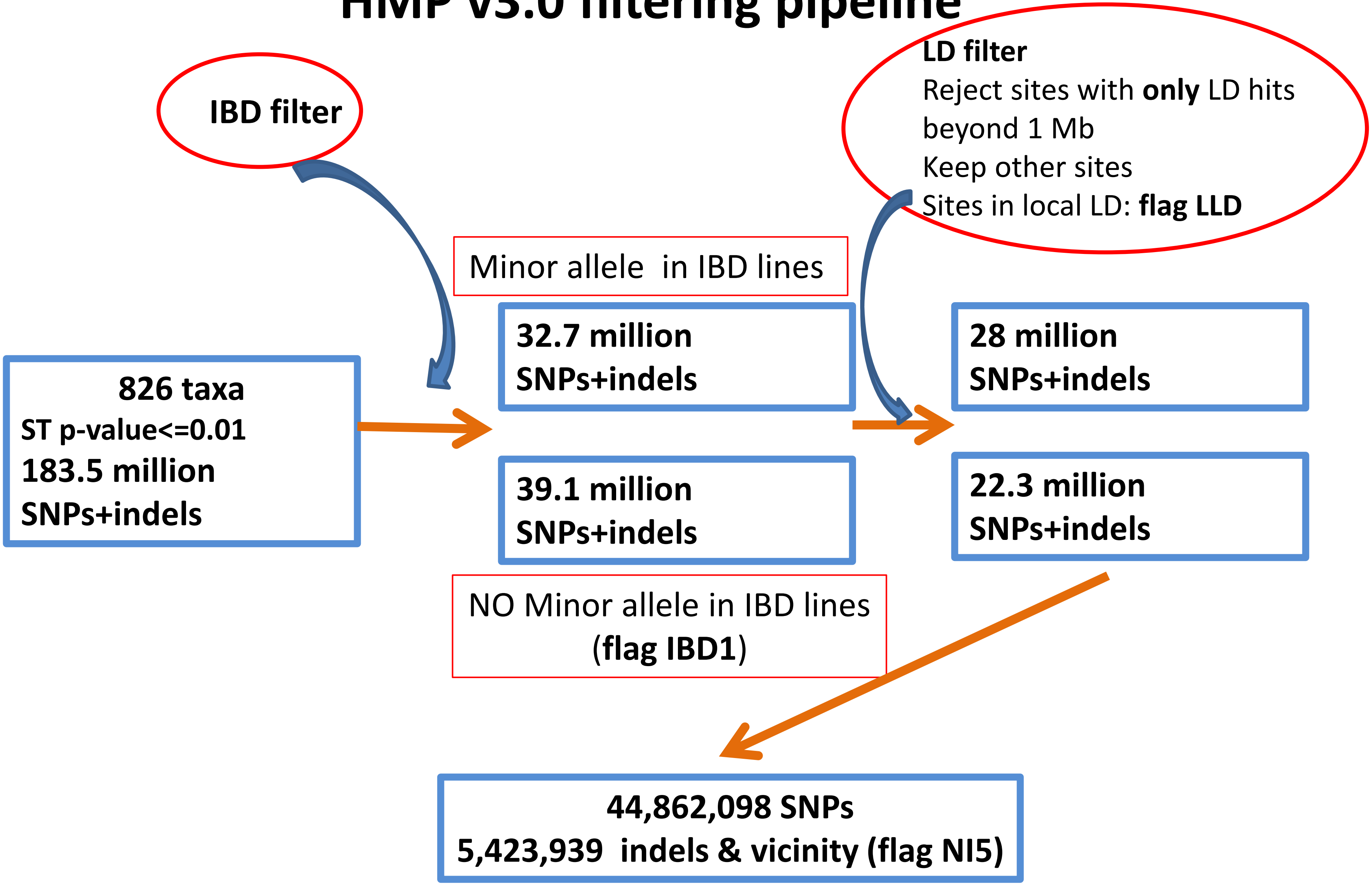
Reject the site

NO

YES

Keep the site

HMP v3.0 filtering pipeline



IBD filter

LD filter
Reject sites with **only** LD hits beyond 1 Mb
Keep other sites
Sites in local LD: **flag LLD**

826 taxa
ST p-value<=0.01
183.5 million SNPs+indels

Minor allele in IBD lines

32.7 million SNPs+indels

39.1 million SNPs+indels

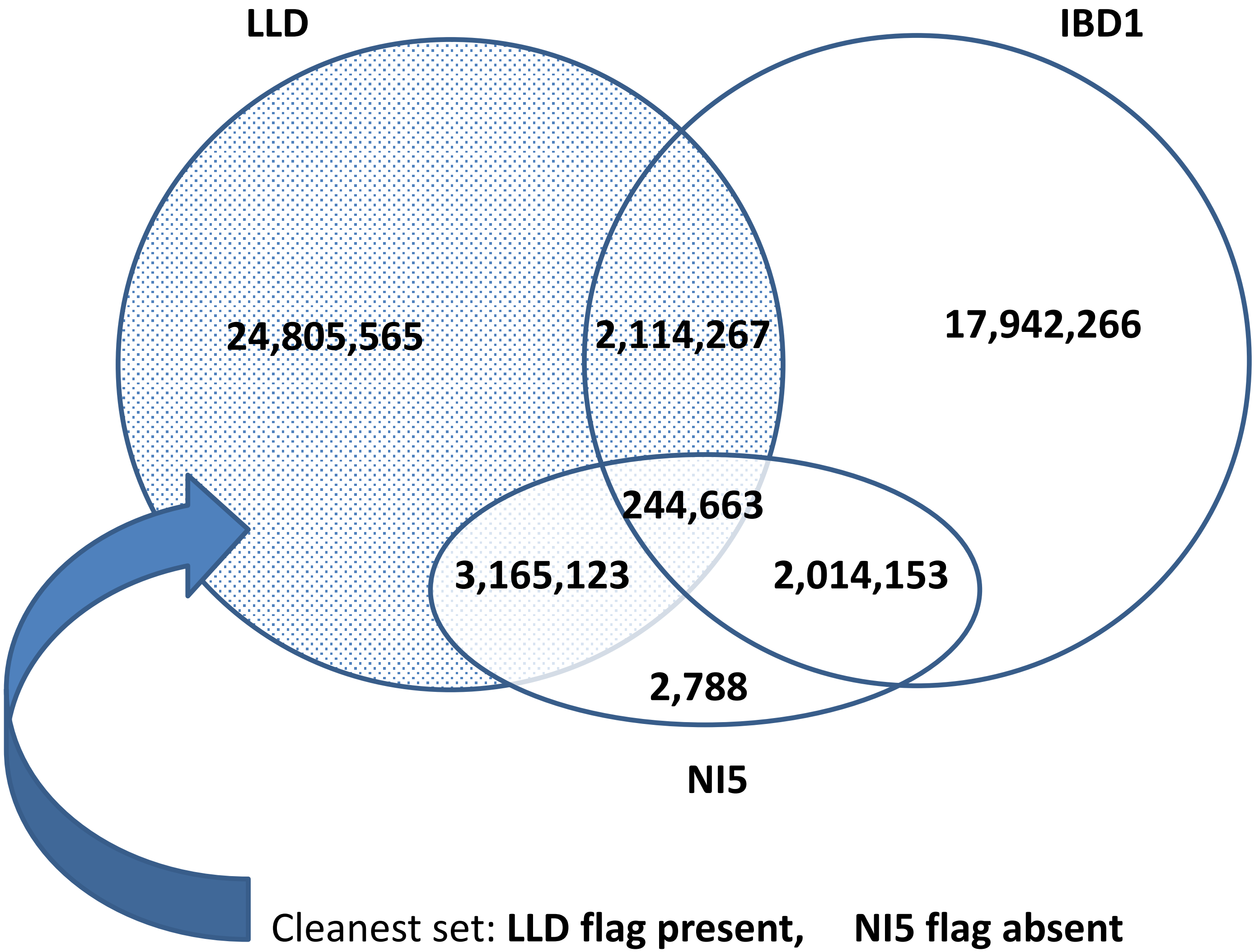
NO Minor allele in IBD lines
(**flag IBD1**)

28 million SNPs+indels

22.3 million SNPs+indels

44,862,098 SNPs
5,423,939 indels & vicinity (flag NI5)

HMP v3.0 polymorphisms

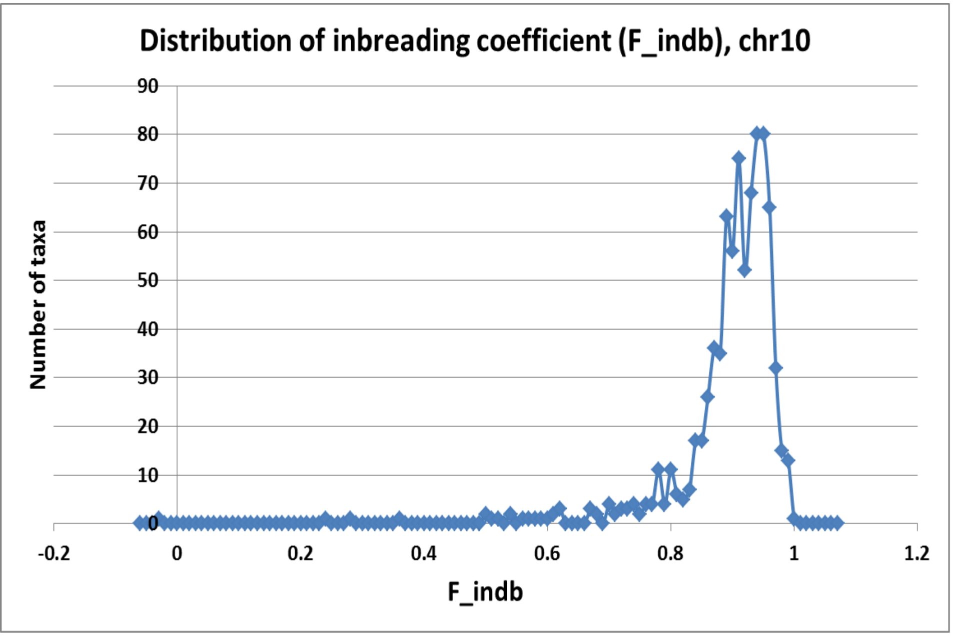


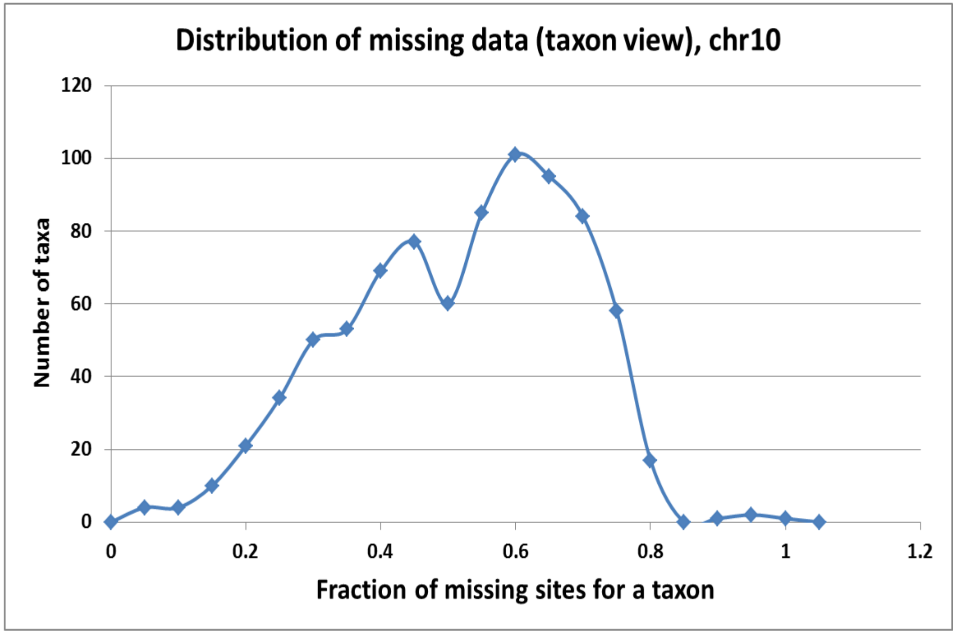
Cleanest set: **LLD flag present,   NI5 flag absent**

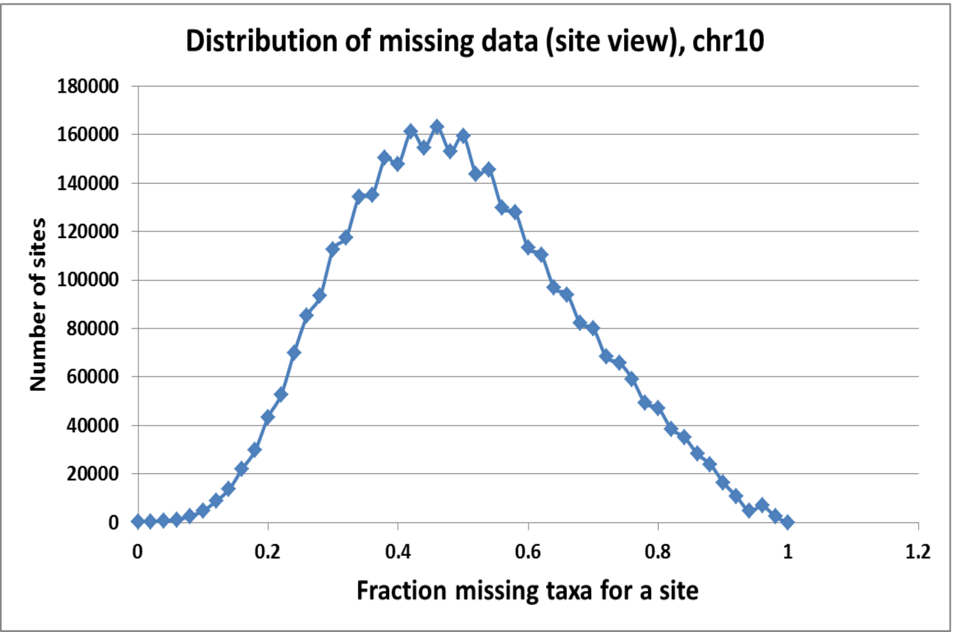
HMP v 3.0 VCF header

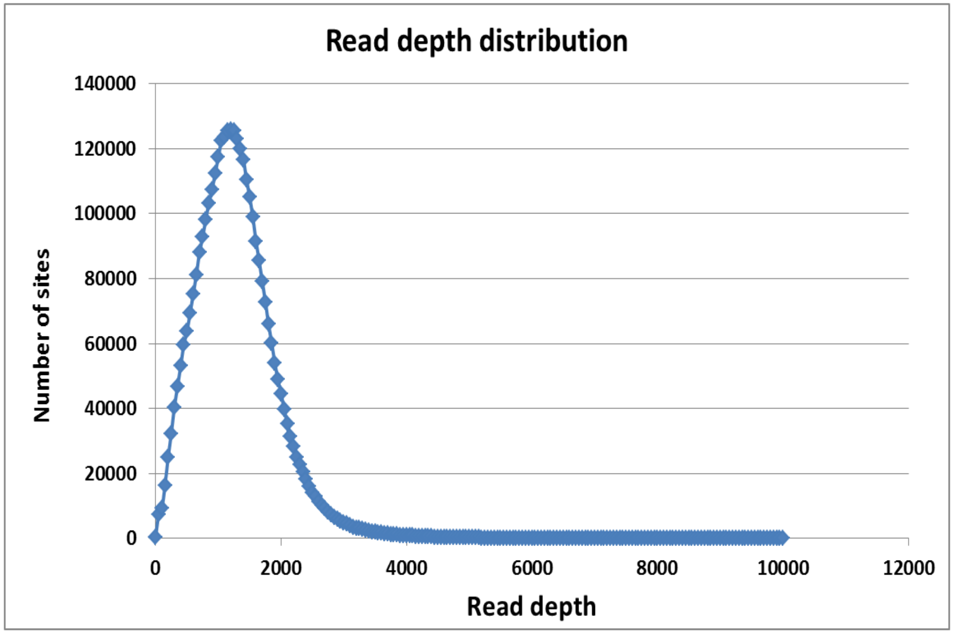
```
##fileformat=VCFv4.1
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##INFO=<ID=NZ,Number=1,Type=Integer,Description="Number of taxa with data">
##INFO=<ID=AD,Number=.,Type=Integer,Description="Total allelelic depths in order listed">
##INFO=<ID=AN,Number=.,Type=Integer,Description="Total number of alleles in order listed">
##INFO=<ID=AQ,Number=.,Type=Integer,Description="Average phred base quality for alleles">
##INFO=<ID=GN,Number=.,Type=Integer,Description="Number of taxa with genotypes AA,AB,BB">
##INFO=<ID=HT,Number=1,Type=Integer,Description="Number of heterozygotes">
##INFO=<ID=EF,Number=1,Type=Float,Description="Ed factor">
##INFO=<ID=PV,Number=.,Type=Float,Description="p-value from segregation test">
##INFO=<ID=IBD1,Number=0,Type=Flag,Description="only one allele present in IBD contrasts">
##INFO=<ID=LLD,Number=0,Type=Flag,Description="Site in local LD with GBS map">
##INFO=<ID=MAF,Number=1,Type=Float,Description="Minor allele frequency">
##INFO=<ID=NI5,Number=0,Type=Flag,Description="Site with 5bp of a putative indel">
##ALT=<ID=DEL,Description="Deletion">
##ALT=<ID=INS,Description="Insertion">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=AD,Number=.,Type=Integer,Description="Allelic depths for the ref and alt alleles">
##HapMapVersion="3.0"
```

Some stats for HMP v3.0

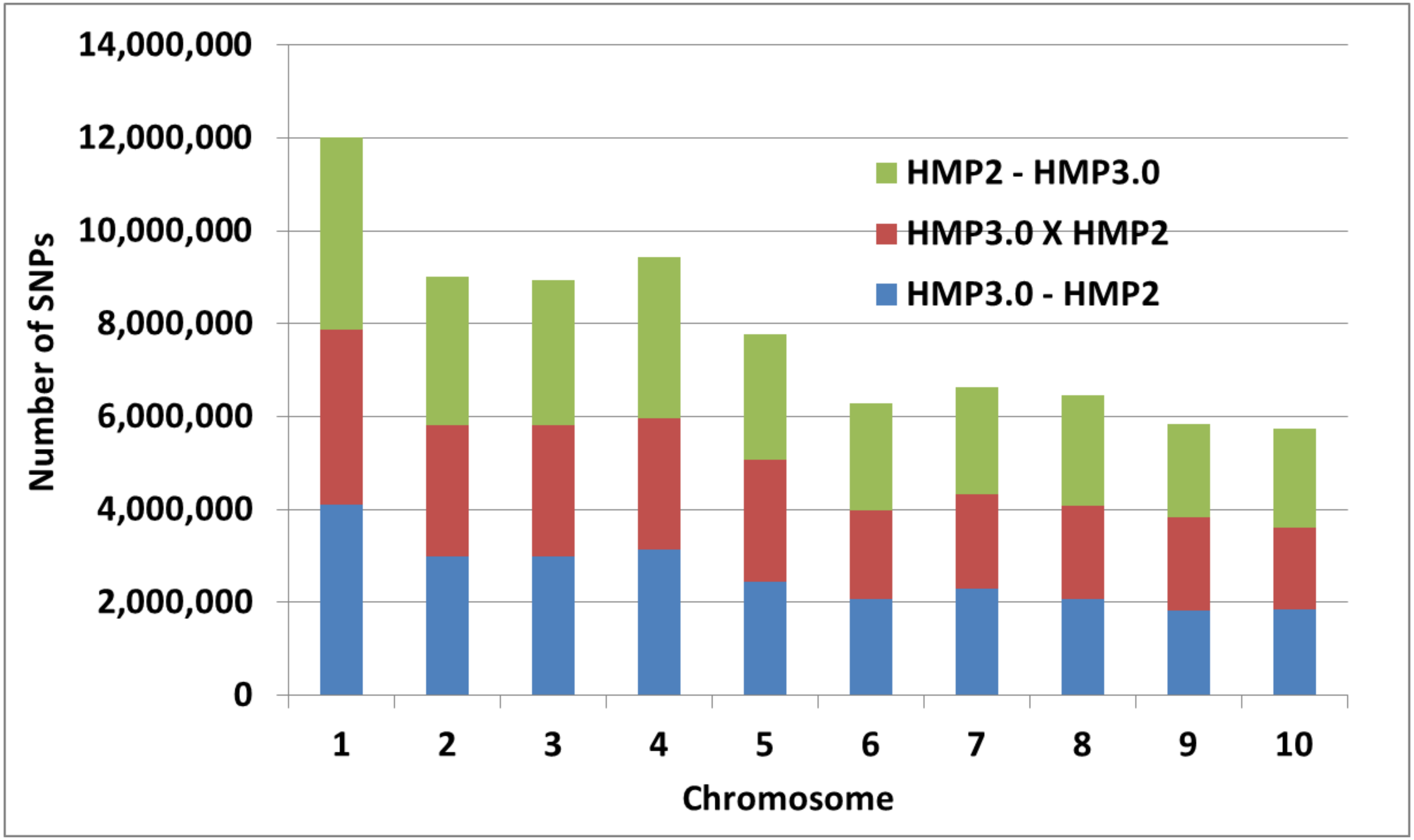

Distribution of inbreading coefficient (F_indb), chr10


Distribution of missing data (taxon view), chr10


Distribution of missing data (site view), chr10


Read depth distribution

Overlap between HMP v3.0 and HMP2 SNP sets



Overall:

HMP3.0 – HMP2	HMP3.0 X HMP2	HMP2 – HMP3.0
25,765,774	24,557,289	27,777,753

Outlook

- Develop a Machine Learning model for scoring sites based on attributes like MAF, heterozygosity, IBD and LD parameters, etc.
    - Training set needed

- Indels
    - Read re-alignment around indels needed

- Extend reference genome

Acknowledgments:
Buckler Lab
Vince Buffalo (UC Davis)
Yinping Jiao (CSHL)
Jeff RI (US Davis)
Doreen Ware (CSHL)
CBSU