Tutorial 1: Working with next-generation sequencing data - A short primer on QC, alignment, and variation analysis of next-generation sequencing data

Presenters: Thomas Keane and Jan Aerts

Schedule

Time	Presenter	Topics
9:00-10:30	Thomas	• Overview of next-generation sequencing technologies • Applications of next-generation sequencing • Quality Control • Next-gen data formats
10:30-11:00	BREAK	
11:00-12:30	Thomas	• Short read alignment algorithms and tools • Practical examples on use of short read aligners • Sequence assembly methods and tools • Case study: 1000 genomes project • Experimental Design
12:30-13:30	LUNCH	
13:30-15:00	Jan	• Overview of variation calling from next-generation sequence data • SNP and indel calling theory and tools • Practical examples of variation calling
15:00-15:30	BREAK	
15:30-17:00	Jan	• Introduction to structural variation • Summary of different types of structural variants • Algorithms and tools for calling structural variants • Visualisation of structural variants

About Me

Joined the Sanger Institute in 2006

2006-2008 Pathogen Genomics Group

- Genome assembly
  - Capillary based eukaryote pathogen projects: Malaria, Trypanosoma brucei, Babesia bigemina
- Assessment of early next-gen data
  - 25bp illumina reads – determining error rates, concordance with capillary data, sequence biases
  - 454 assembly assessment of bacterial and parasitic eukaryote genomes
  - Correcting 454 assemblies with Illumina data

2008-date Vertebrate Resequencing Informatics

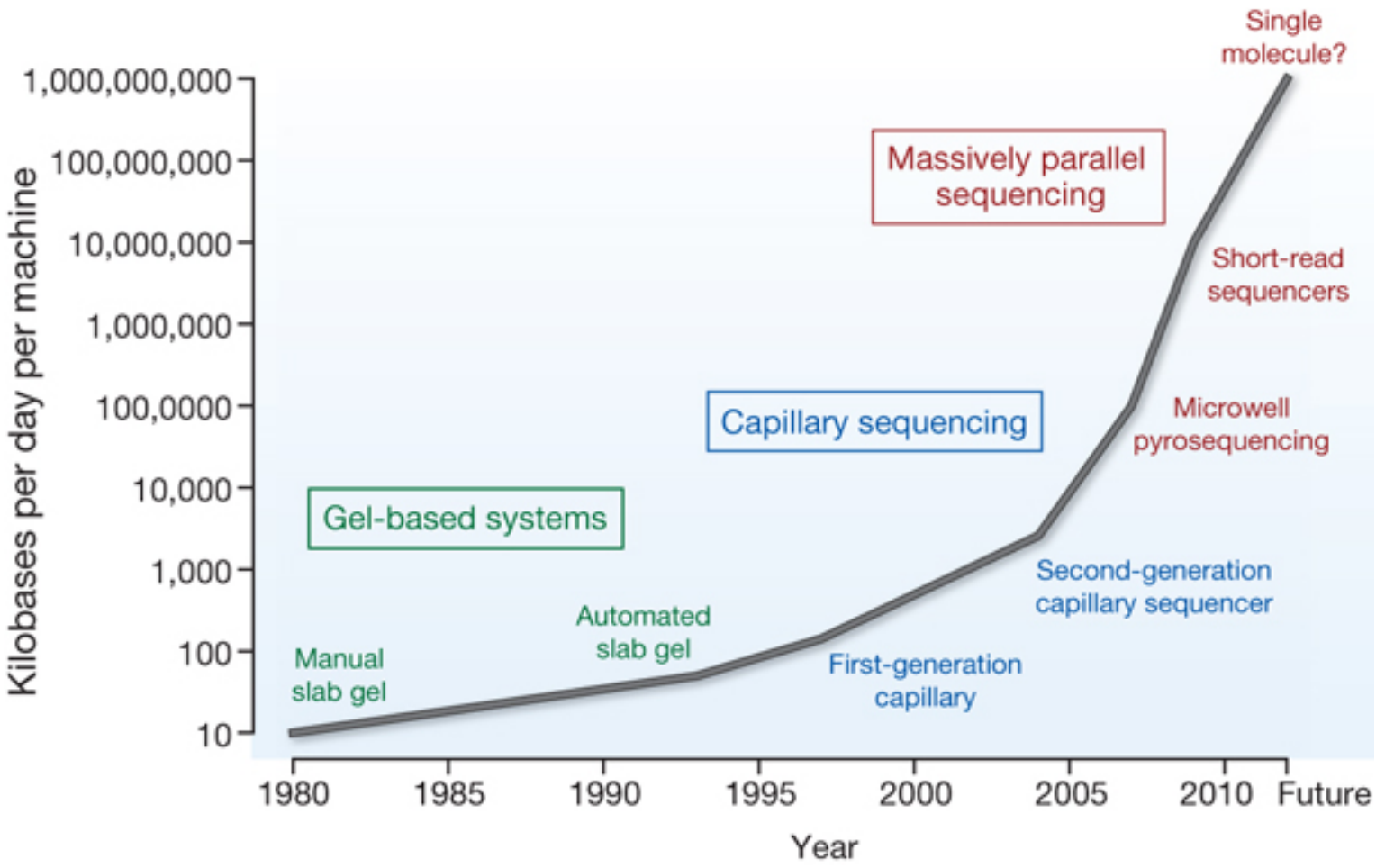
- Established group with Jim Stalker
- Initial projects
  - 1000 Genomes project (http://www.1000genomes.org)
    - Data releases, alignments, Sanger 1000G sequencing, data submission, aligner evaluation
  - Mouse Genomes Project (http://www.sanger.ac.uk/mousegenomes)
    - Sanger project, catalogue variants across 17 mouse strains, de novo assembly, data release and visualisation
- Upcoming projects
  - UK10K project
    - 4000 whole genomes at 6x, 6000 human exomes
    - Medical sequencing - Phenotype based study

Email: thomas.keane@sanger.ac.uk

Tutorial 1: Overview, Applications, QC and Formats

- Overview
- Quality Control
- Next-gen Data Formats
- Short Read Alignment
- Sequence Assembly
- Case Study: 1000 Genomes
- Experimental Design

Some Perspective



MR Stratton *et al. Nature* **458**, 719-724 (2009)

Previous Technology: Sanger Sequencing

1992-1999

ABI 373 / 377 Slab gel sequencers
2-3 runs per day, 36 - 96 samples per run
100kb per instrument per day 80 staff to operate

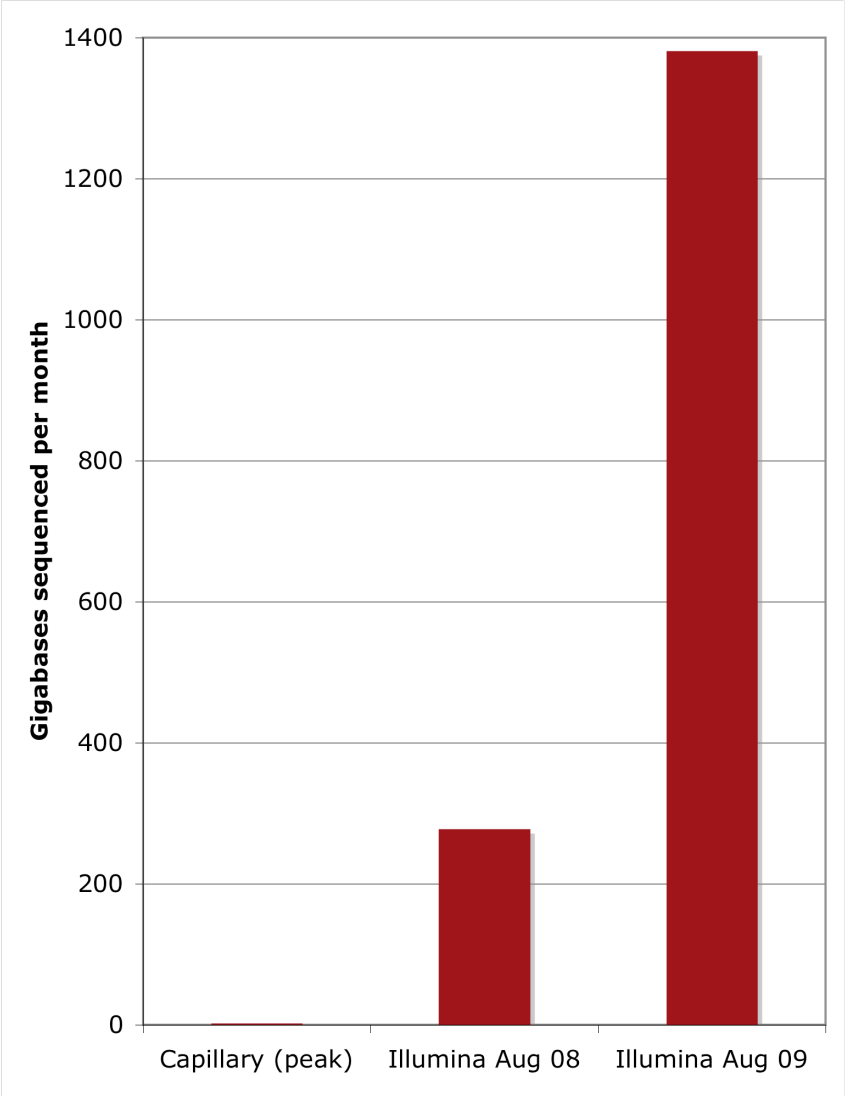
2000

ABI 3700 Capillary sequencer 8 runs per day, 96 samples per run
400kb per instrument per day 40 staff to operate

2004

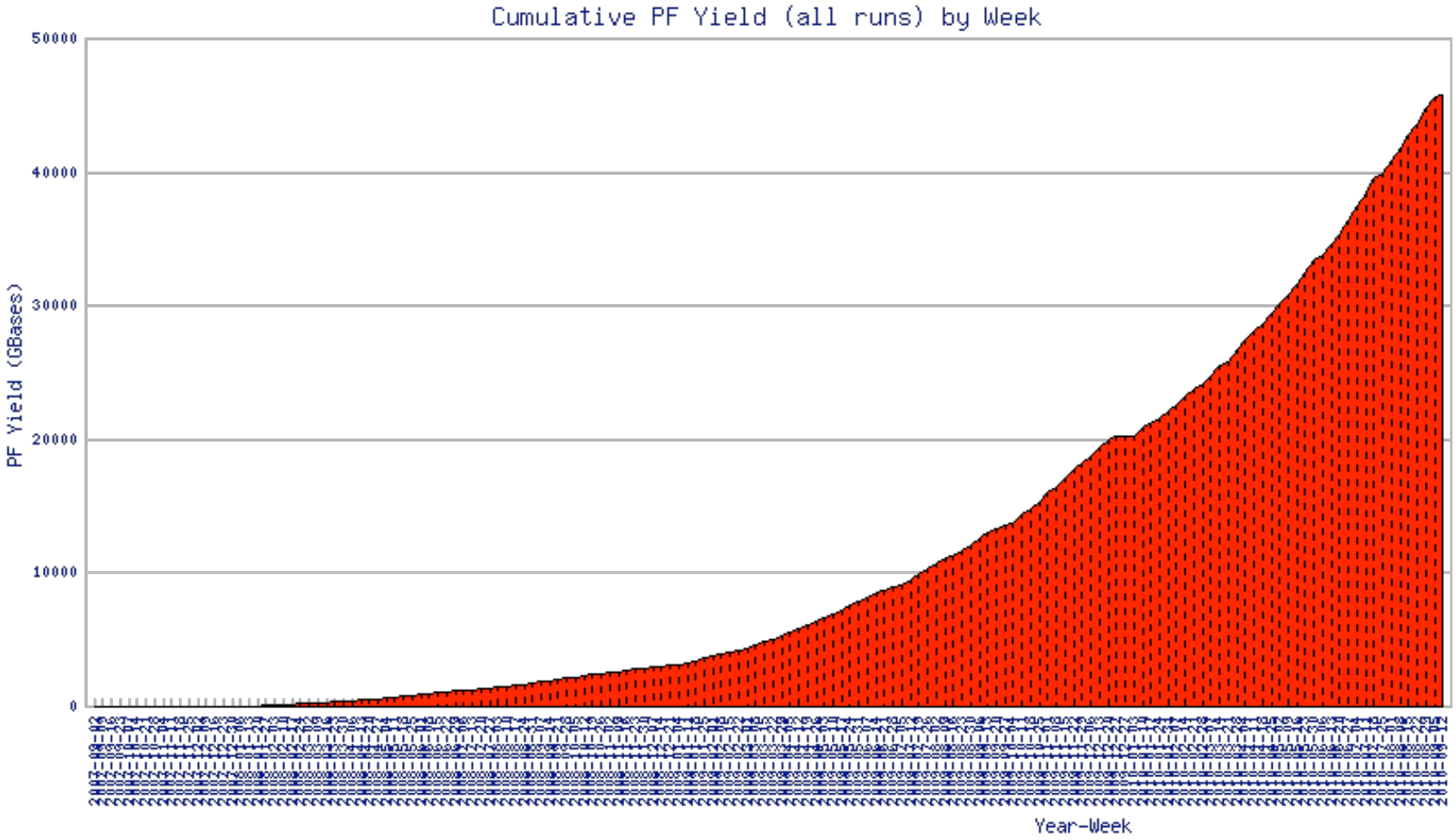
ABI 3730xl Capillary sequencers
15 - 40 runs per day x 96 samples
1-2Mb per instrument per day
4 staff to operate

A view from the Sanger



- 2000-2003 peak monthly capillary: ~2.7Gb
- August ‘08 production: ~278Gb
- August ‘09 production: ~1.38Tb
- Jan ‘10 Illumina
  - HighSeq 2000 – min 200Gb per run
  - ~60x human depth per run

A View from Sanger



Next-gen Sequencing Technologies

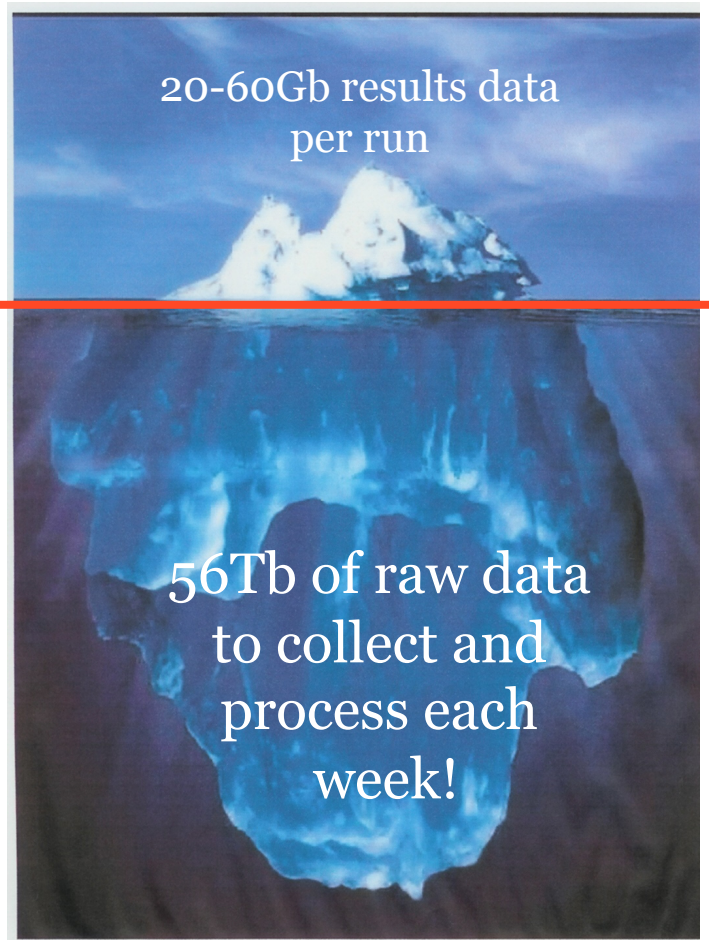
Table 1 Comparison of second-generation sequencing technologies

Sequencing platform	Sample requirements	Length of library prep/feature generation (days)	Method of feature generation	Sequencing chemistry	Read length (bases)	Run time	Throughput/ run (Gb)	Throughput/ day (Gb)
Roche 454 (FLX-Titanium)	1 μg for shotgun library, 5 μg for paired end	3–4	Bead-based/ emulsion PCR	Pyrosequencing	400–500	10 h	0.4–0.5	~1
Illumina Genome Analyzer (GAII)	<1 μg for single or paired-end libraries	2	Isothermal 'bridge amplification' on flowcell surface	Reversible terminator SBS	35–75	2 days for 36-cycle single-end run, 4 days for 36-cycle paired-end run	3–6	1.5
ABI SOLiD	<2 μg for shotgun library, 5–20 μg for paired end	2–4.5	Bead-based/ emulsion PCR	Ligation	25–75	6–7 days for fragment libraries, 8 days for 2 × 25 base paired-end libraries	10–20	1.7–2
Helicos tSMS	<2 μg, single end only	1	N/A (single molecule sequencing)	Virtual terminator SBS	25–50	8–9 days	21–28	2.5

Turner et al., 2009

The Sequencing Iceberg

1Tb of images, intermediate analysis files and tracking data per run



One sequencing run every 3 days (per instrument)

28 instruments

Storage And Compute

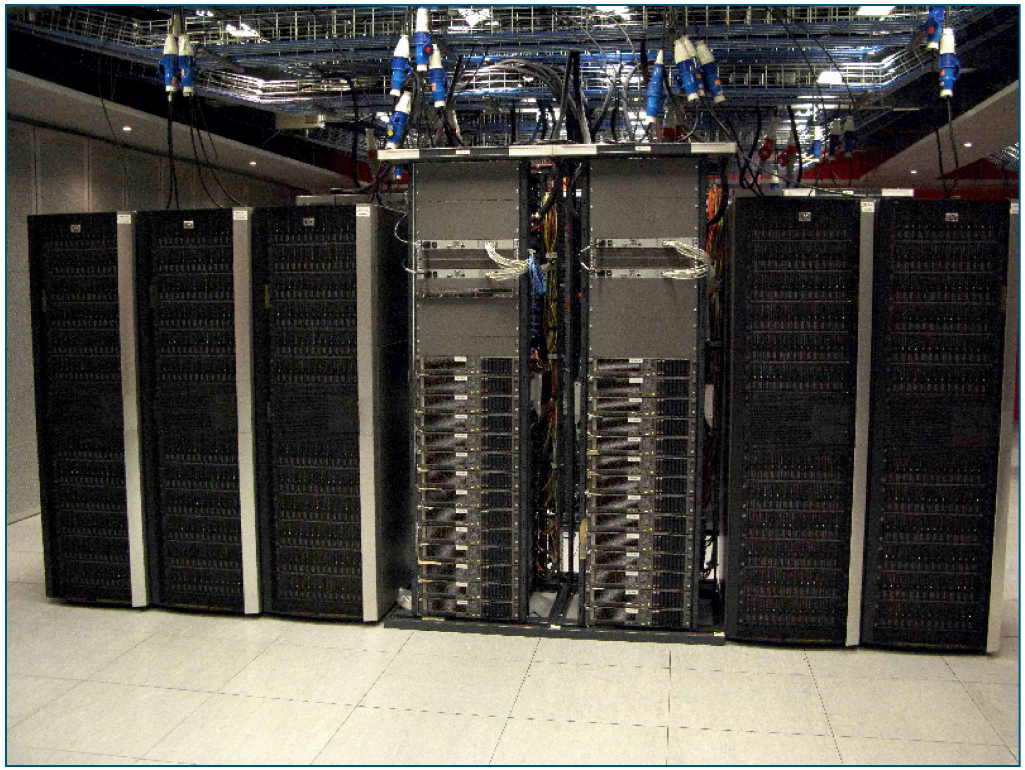
450Tb “staging” area to collect raw data

800 CPU compute farm

- primary data processing
- Secondary analysis
- Alignment

What do we store?

- SRF files for submission
  - 8-10 bytes per bp
- Fastq file of reads
  - 1 byte per bp
- BAM files of reads
  - 1.7 bytes per bp
  - Aligned and unaligned reads
  - Second best base call
  - Original + recalibrated base qualities



Next-gen Applications

Whole Genome Shotgun Sequencing (WGS)
- Randomly shear whole DNA and sequence fragments
- Paired end sequencing

Targeted/exome sequencing
- Randomly shear whole DNA
- Select fragments based on some pre-defined set of templates (“baits”)
- Exome sequencing targets the coding regions
- Requires pre-design of pulldown assay
  - Currently human, mouse, zebrafish

RNA-seq
- Start with whole RNA
- Reverse transcribe to cDNA, randomly shear and sequence
- Alignment depth is proportional to the relative abundance of the transcript
- Gene finding + alternative splicing
- No pre-design required

ChiP-seq
- ChIP is a powerful method to selectively enrich for DNA sequences bound by a particular protein in living cells
- Align + peak calling to detect active binding sites
- ChIP-Seq data can be used to locate the binding site within few tens of base pairs of the actual protein binding site

Large-scale sequencing projects

Capillary days

- Single individual, single genome projects
- Small amounts of variation based sequencing

Next-generation era

- Multiple individuals from multiple populations
  - 1000 Genomes Project
  - MalariaGEN - sequencing thousands of malaria isolates
  - ICGC – International Cancer Genome Consortium
  - Mouse Genomes Project – sequencing 17 common laboratory mouse strains
  - 1001 Genomes Project - *Arabidopsis* whole-genome sequence variation
  - UK10K – sequencing of 10,000 healthy and disease affected individuals

The Next-Generation Process

DNA Prep

↓

Library Prep

↓

Chip Prep

↓

Sequencing

↓

Raw Analysis

Randomly shear DNA + end repair + size select

Append sequencing adapters

Layout of library on sequencing slide or wells

For each library fragment – determine the order and identity of bases at either end of the fragment

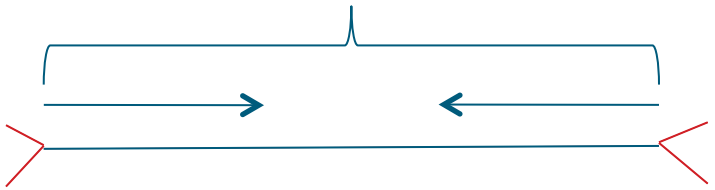


Image processing + base calling

- Base calls + associated quality: Fastq/BAM

Tutorial 1: Overview, Applications, QC and Formats

- Overview
- **Quality Control**
- Next-gen Data Formats
- Short Read Alignment
- Sequence Assembly
- Case Study: 1000 Genomes
- Experimental Design

Sequence Quality Control

Sequence quality control is essential

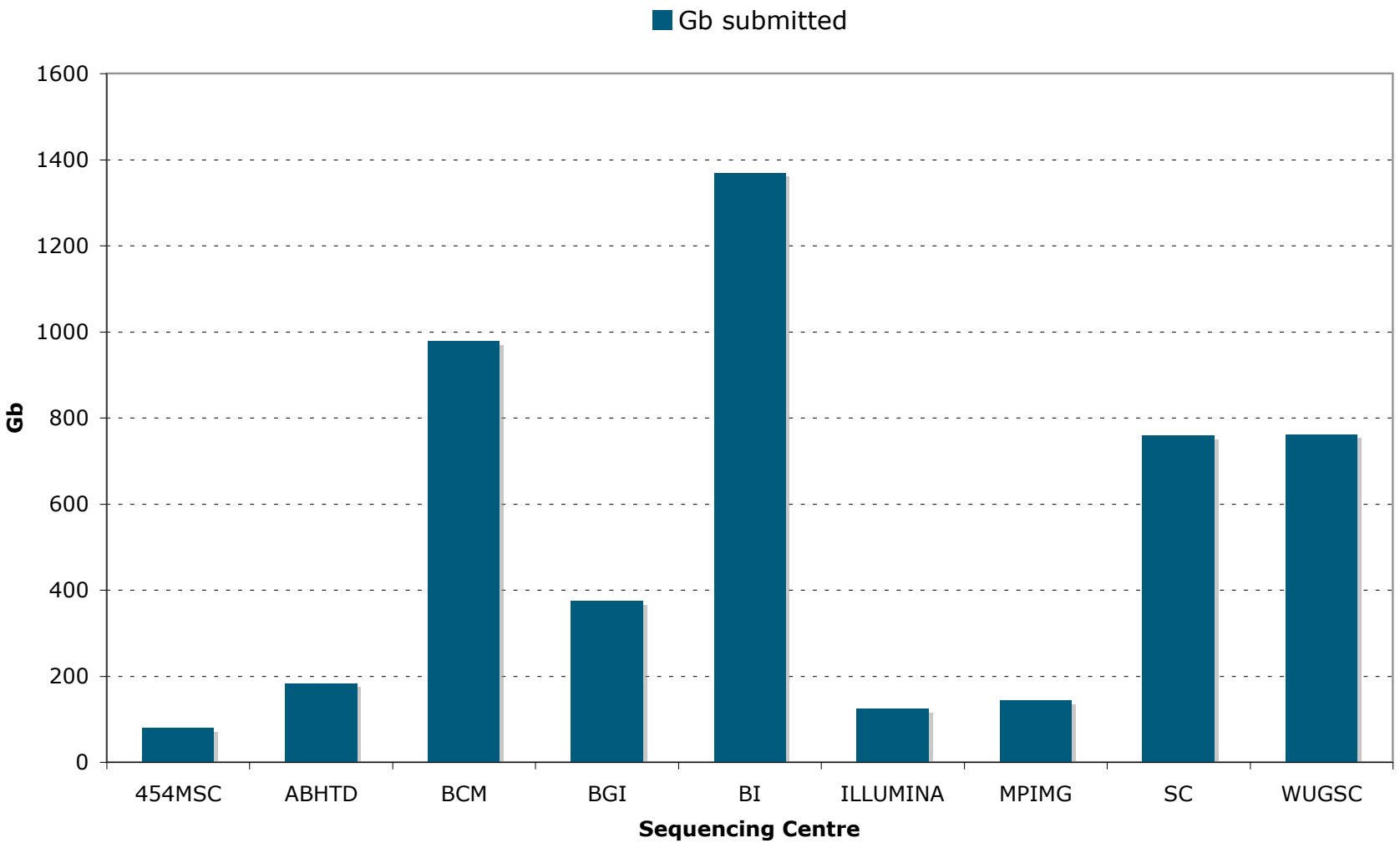
Why?

- Sequencing runs cost money
  - Sequencing a poor library on multiple runs – throwing money away!
- Cost of analysing data
  - CPU time required for alignment and assembly
  - Cost of storing raw sequence data
  - People time in analysing interpreting the results from poor data

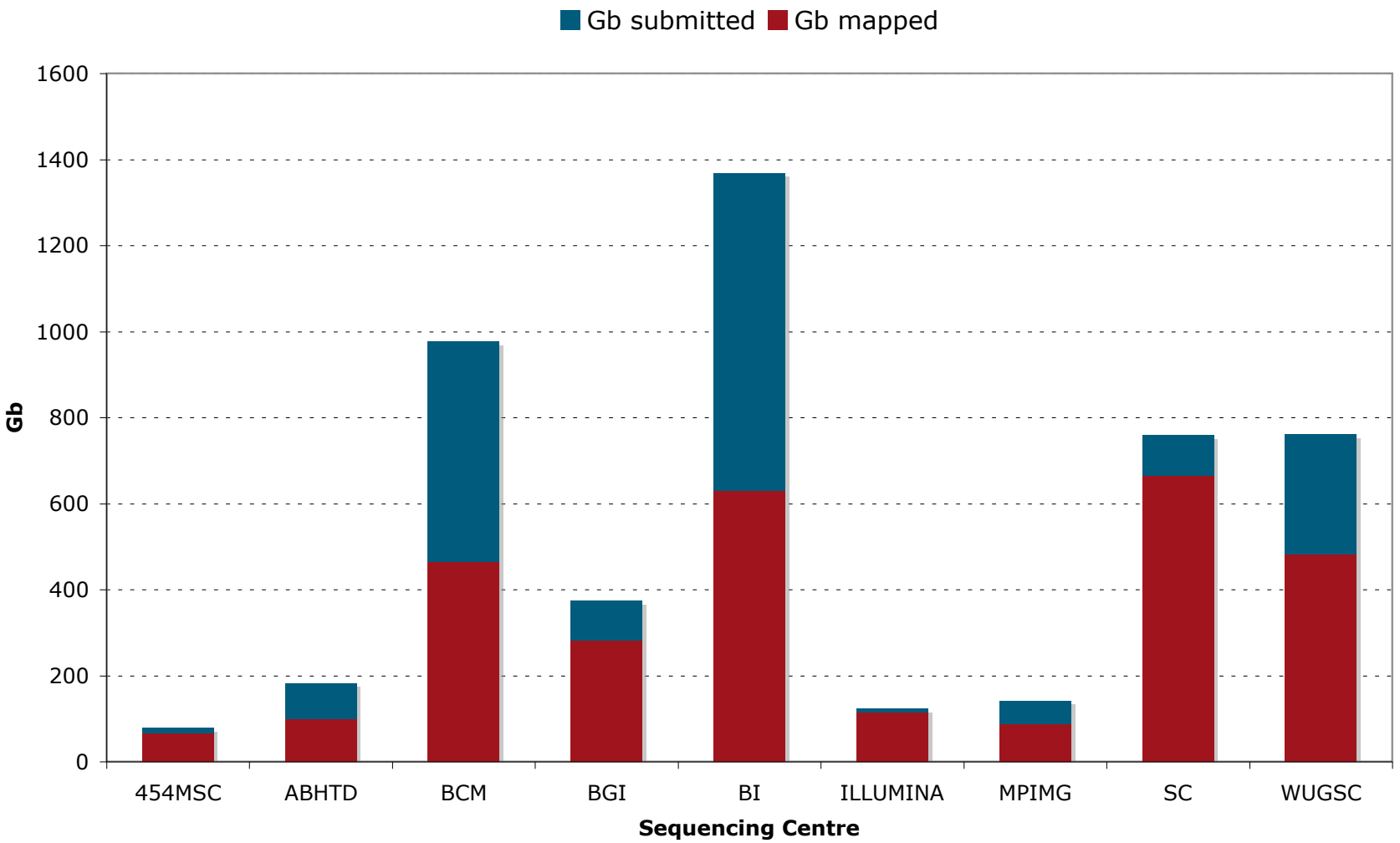
Two aspects of sequencing QC

- Run/Lane QC
  - Is the data off this lane useful?
- Library QC
  - Is this library worth sequencing more lanes off?
  - E.g. Have 2x but want another 8x in total
- Slightly different set of metrics

1000 Genomes Pilot Data



1000 Genomes Pilot Data



Raw Sequence QC

DNA sequencing is not an exact science!

Quality control on all sequence data/libraries is essential

1 test lane/plex per sequencing library

- Pass QC then sequence more from the library

Run our own QC pipeline within our group

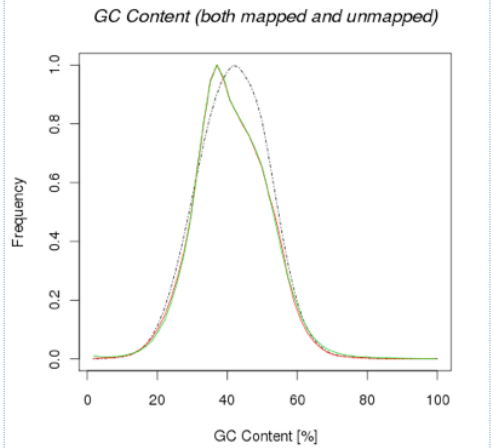
- GC plot vs. reference GC
- Adapter count
- Sample 100Mbp from the lane
  - Align to reference
  - Percent reads/bases mapping
  - Error rate
  - Insert size plot
  - GC vs. depth plot
  - Duplication rate
  - Clipping points

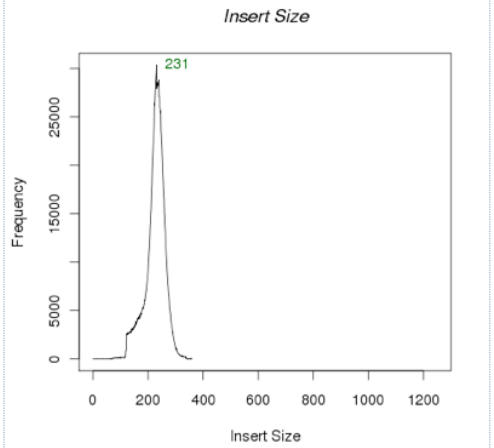
Lane QC

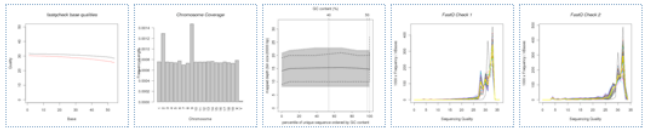
QC Grind

129S5 Mouse Genome : 129S5 : 129S5_SLX_200_NOPCR_3 : 2973_6 [npg]

QC Plots







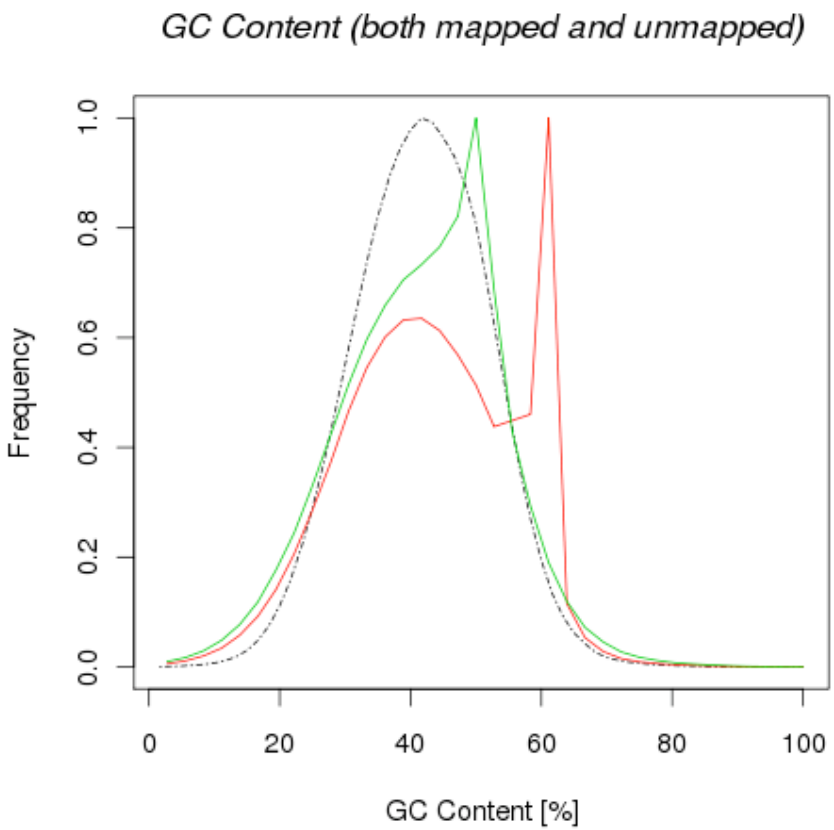
Mapping data

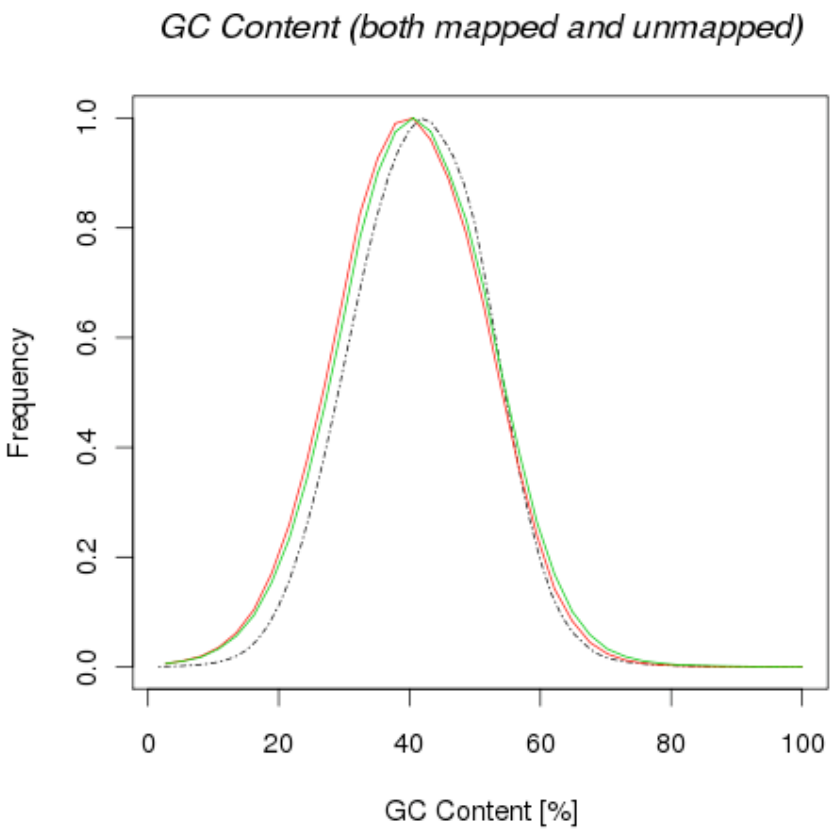
Reads:	2,298,678	Bases:	124,128,612	Cycles:	54
Reads w/adapter	59678 (2.6%)	Bases postclip:	121,776,614 (98.1% raw)	NPG QC:	fail
Reads mapped:	2,196,247 (95.5%)	Bases mapped:	116,928,621 (96.0%)	Error rate:	0.011
Reads paired:	1,897,140 (82.5%)			Genotype check:	confirmed (129S5:2.865)
Reads mapped (rmdup):	2,136,231 (92.9%)	Bases mapped (rmdup):	113,706,349 (93.4%)	Duplication rate:	0.0273

Lane QC : passed

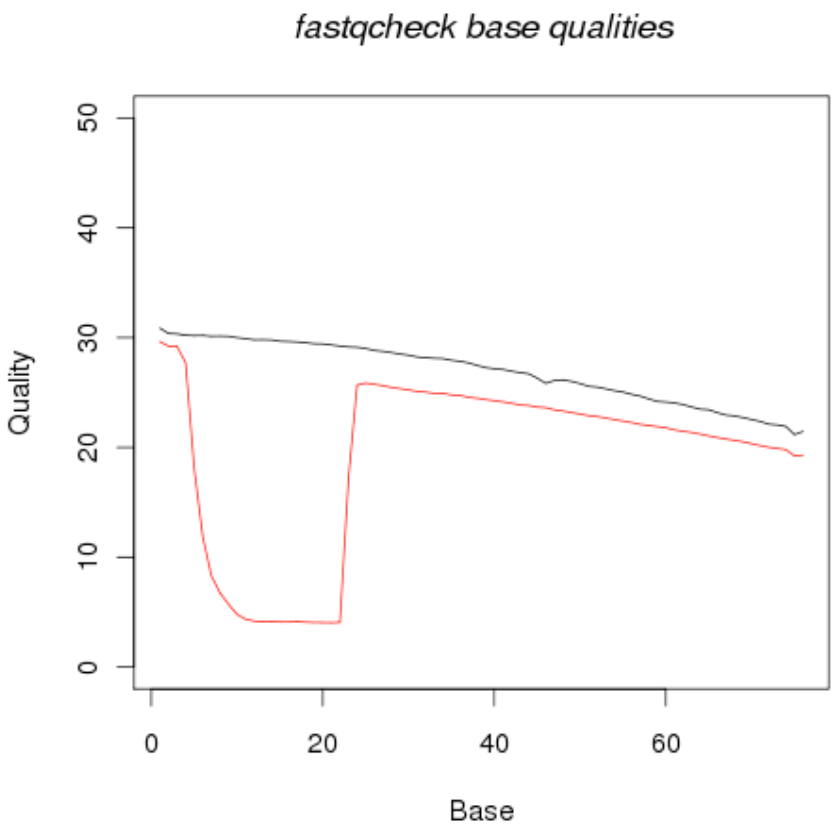
pending failed passed

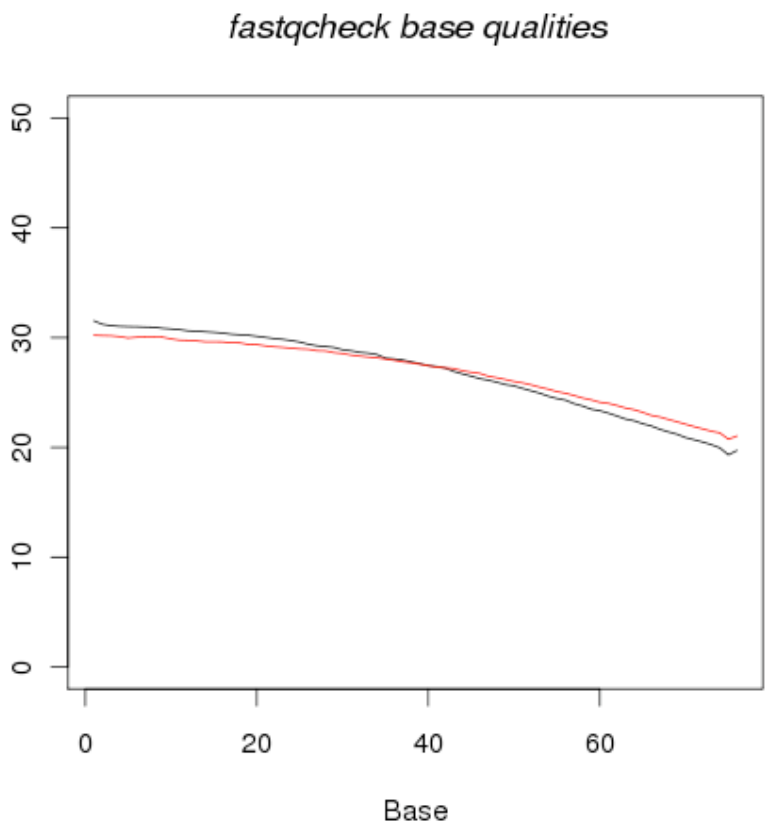
GC


GC Content (both mapped and unmapped)
Frequency
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100
GC Content [%]

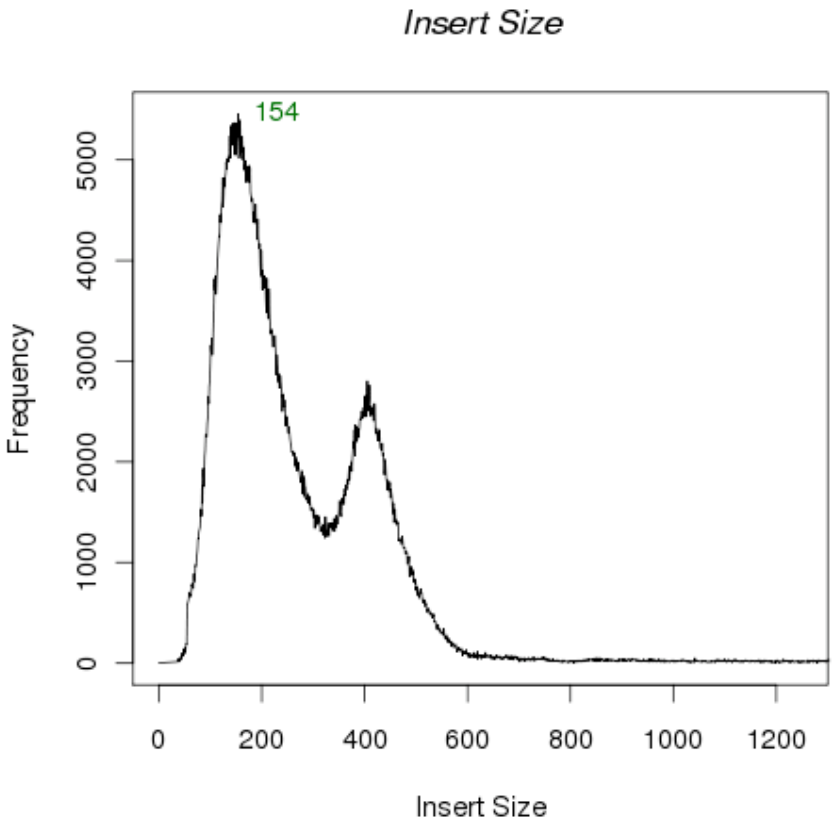

GC Content (both mapped and unmapped)
Frequency
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100
GC Content [%]

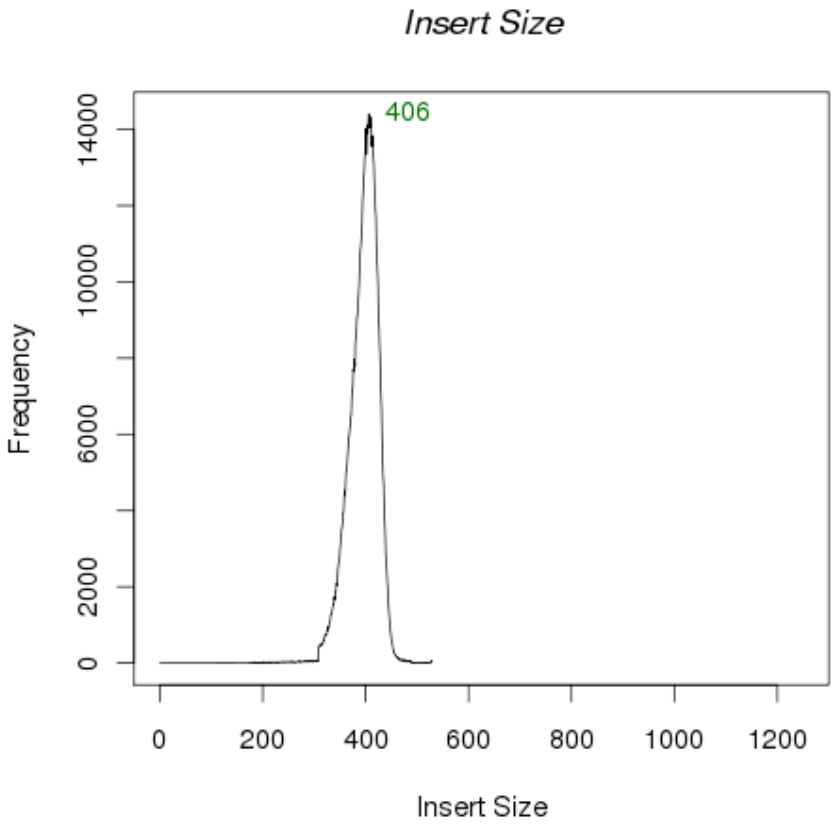
Average Base Quality



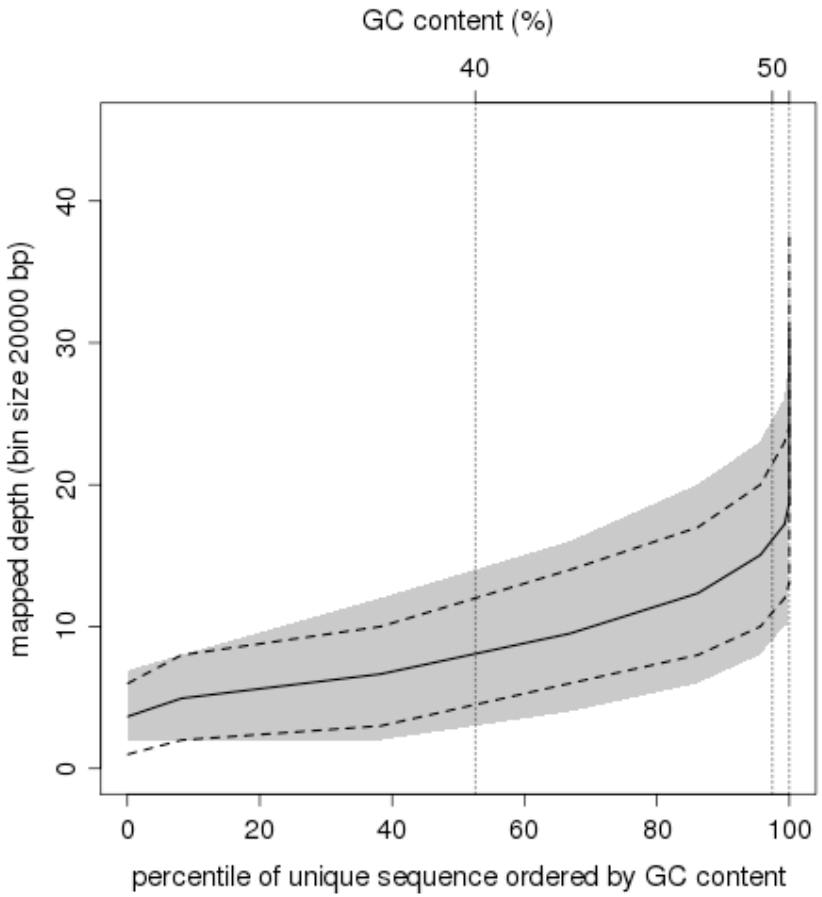


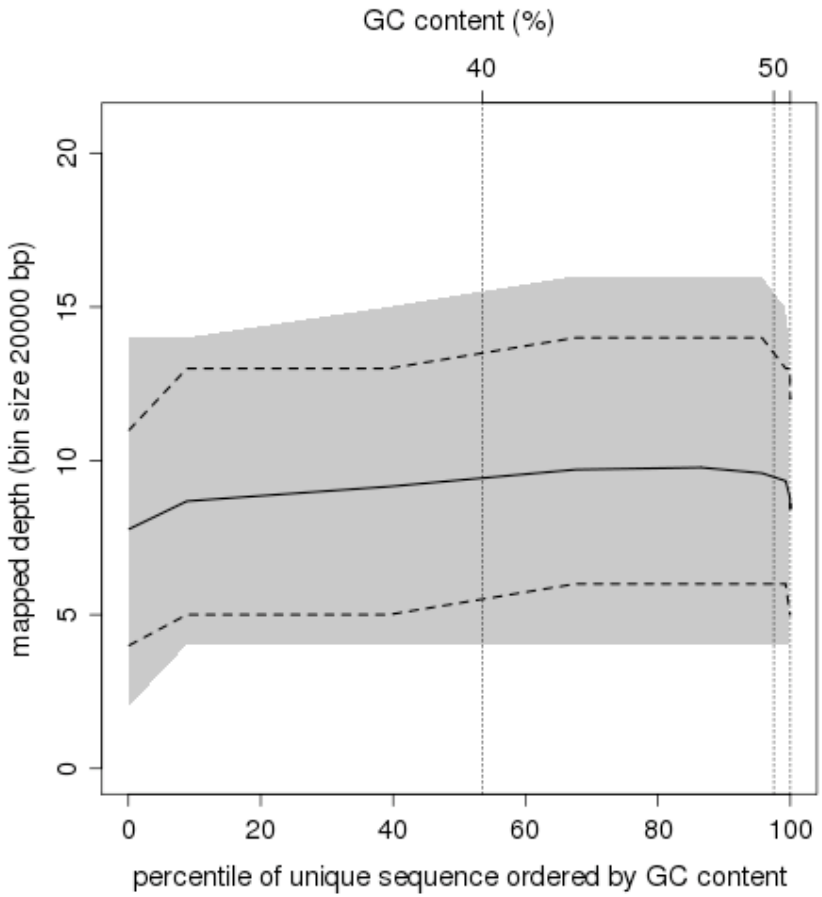
Insert size


Insert Size
154
Frequency
0
1000
2000
3000
4000
5000
0
200
400
600
800
1000
1200
Insert Size


Insert Size
406
Frequency
0
2000
6000
10000
14000
0
200
400
600
800
1000
1200
Insert Size

GC-Depth


GC content (%)
40
50
mapped depth (bin size 20000 bp)
0
10
20
30
40
0
20
40
60
80
100
percentile of unique sequence ordered by GC content


GC content (%)
40
50
mapped depth (bin size 20000 bp)
0
5
10
15
20
0
20
40
60
80
100
percentile of unique sequence ordered by GC content

Genotype checking

How do you know the data you get out is from the individual you meant to sequence? Mistakes happen:

- Original sample might be wrong individual
- Library might be made from wrong sample
- Lane might be loaded from wrong library

Check:

- Compare to known genotypes
  - samtools pileup + glf checkGenotype => likelihood that lane belongs to all samples
  - 3-4% of lanes found to be incorrect
- Compare to other lanes of same library
- Barcode libraries
  - Yield from single lane now ~1-1.5x
  - Allows for multiplexing of samples within a lane
    - Up to 24 libraries per lane with barcodes

Library Duplicates

All second-gen sequencing platforms are NOT single molecule sequencing

- PCR amplification step in library preparation
- Can result in duplicate DNA fragments in the final library prep.
- PCR-free protocols do exist – require large volumes of input DNA

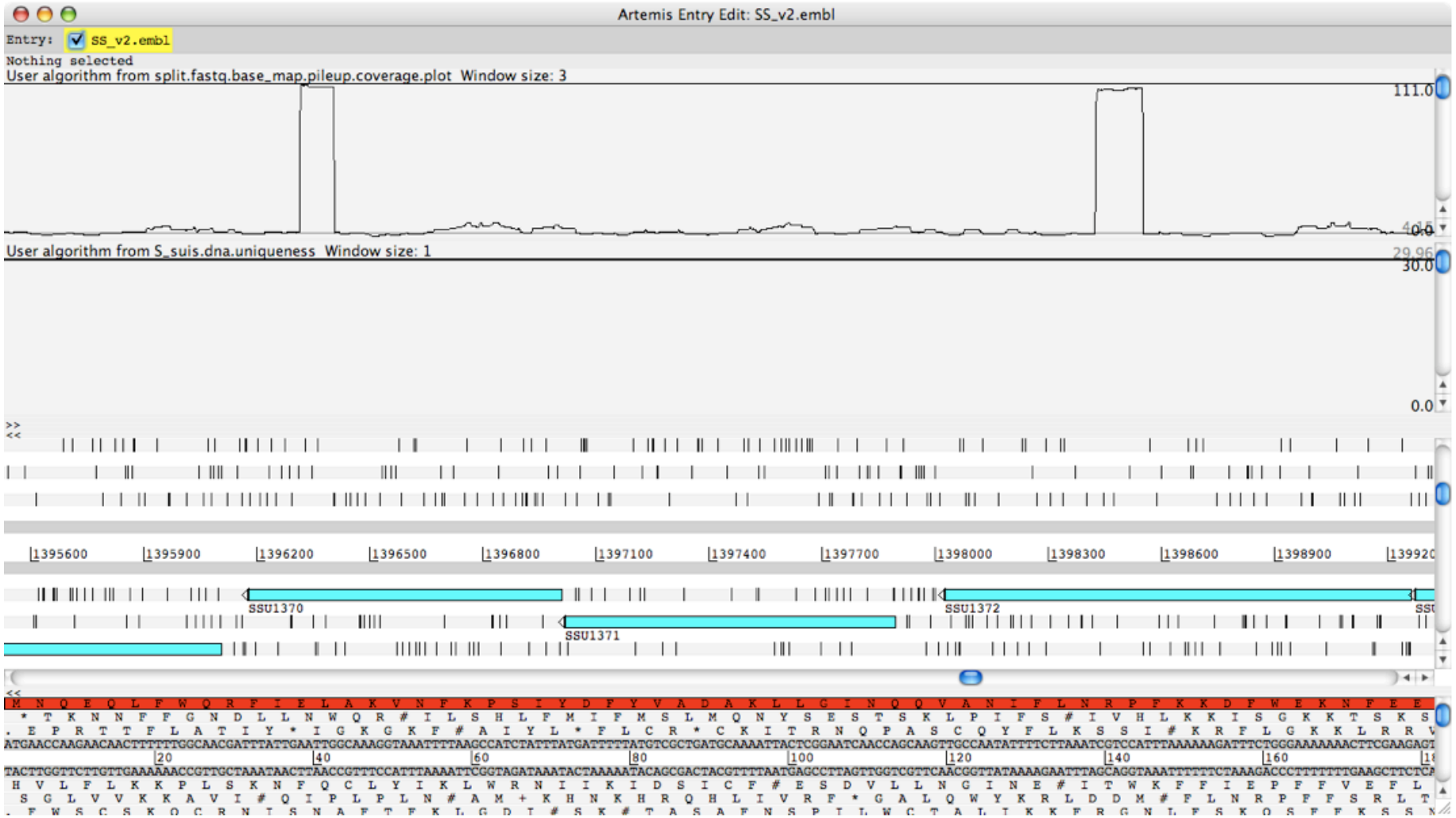
Generally low number of duplicates in good libraries (<3%)

- Align reads to the reference genome
- Identify read-pairs where the outer ends map to the same position on the genome and remove all but 1 copy
  - Samtools: samtools rmdup or samtools rmdupse
  - Picard/GATK: MarkDuplicates

Can result in false SNP calls

- Duplicates manifest themselves as high read depth support

Library Duplicates



Duplicates and False SNPs

NA12005 - chr20:8660-8790

Activity 1: Sequence QC

There are a set of human 1000 genomes lanes with QC information for each lane

Which lanes should pass/fail QC? And why?

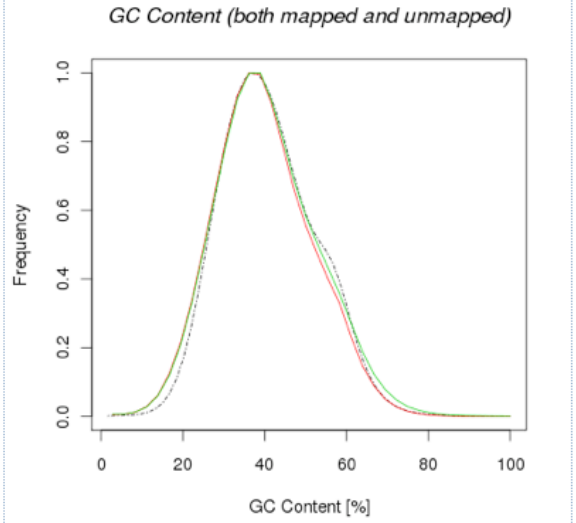
Pass criteria

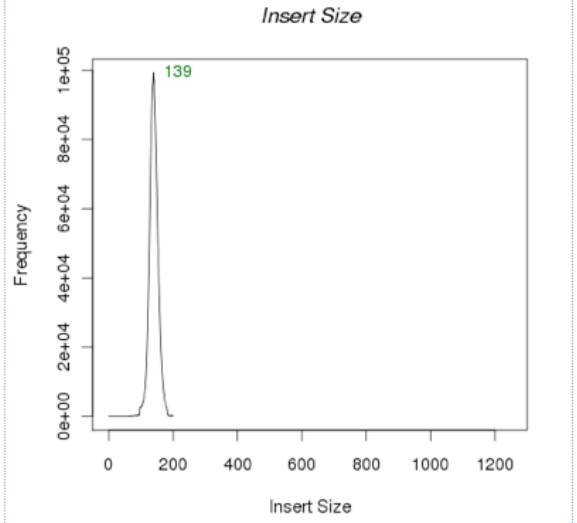
- >80% of reads mapping
- <5% duplicates
- <2% error rate
- Single fragment size peak
  - Automated measure: 80% of reads within 100bp of peak
- Single GC peak

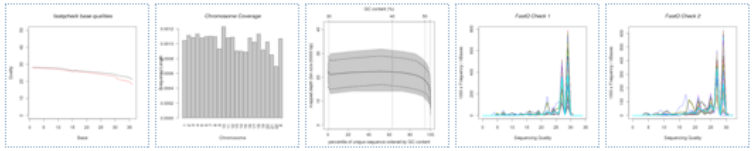
Lane 1

NA11995-CEU-1 : NA11995-CEU-1 : 391_6 [npg]

QC Plots







Mapping data

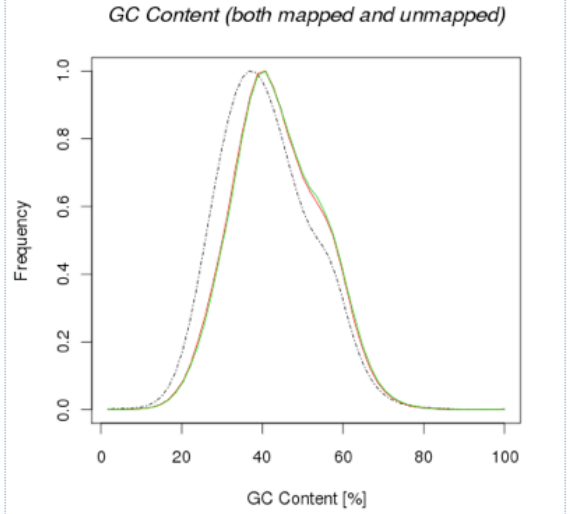
Reads:	3,380,758	Bases:	121,707,288	Cycles:	37
Reads w/adapter	16082 (0.5%)	Bases postclip:	120,368,581 (98.9% raw)	NPG QC:	pending
Reads mapped:	3,215,890 (95.1%)	Bases mapped:	114,512,178 (95.1%)	Error rate:	1.40%
Reads paired:	2,987,852 (88.4%)			Genotype check:	confirmed (NA11995:1.389)
Reads mapped (rmdup):	3,212,470 (95.0%)	Bases mapped (rmdup):	114,391,048 (95.0%)	Duplication rate:	0.11%

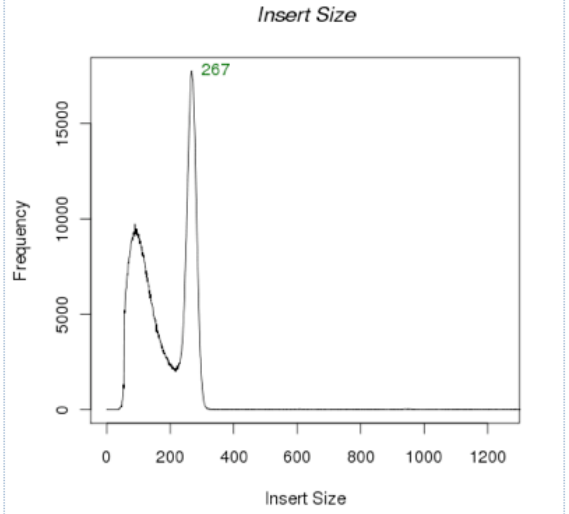
Lane QC : passed (Auto QC :)

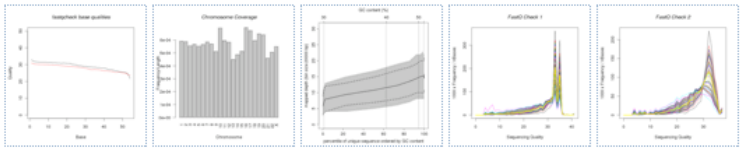
Lane 2

NA07056-CEU : NA07056-CEU-1 : 2060_1 [npg]

QC Plots







Mapping data

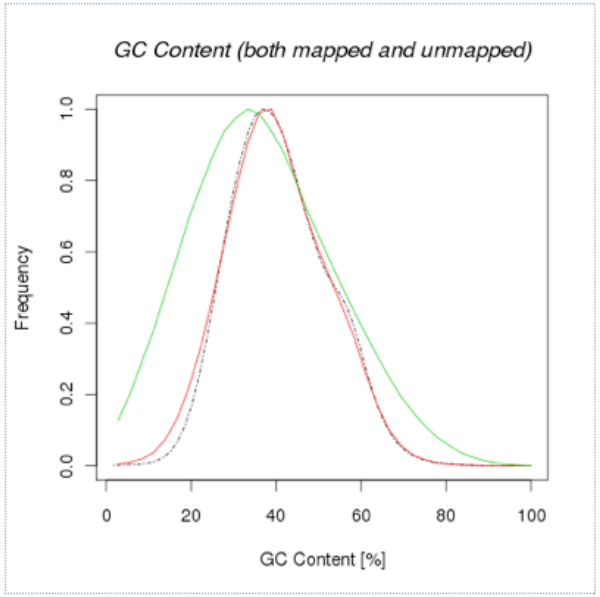
Reads:	1,914,642	Bases:	103,390,668	Cycles:	54
Reads w/adapter	2168 (0.1%)	Bases postclip:	100,595,422 (97.3% raw)	NPG QC:	pass
Reads mapped:	1,770,261 (92.5%)	Bases mapped:	93,171,650 (92.6%)	Error rate:	0.80%
Reads paired:	1,639,266 (85.6%)			Genotype check:	confirmed (NA07056:1.448)
Reads mapped (rmdup):	1,769,875 (92.4%)	Bases mapped (rmdup):	93,151,186 (92.6%)	Duplication rate:	0.02%

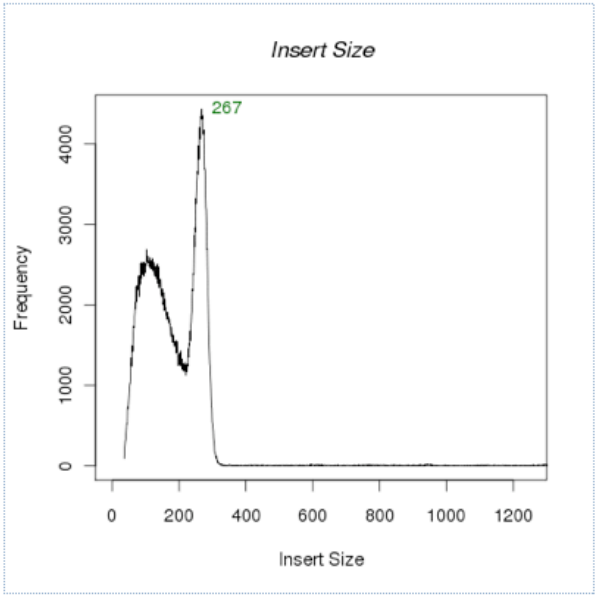
Lane QC : passed (Auto QC :)

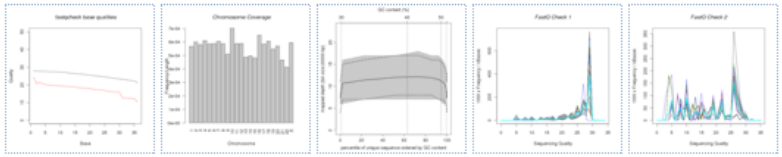
Lane 3

NA20508-TOS : NA20508-TOS 1 : 1444_3 [npg]

QC Plots







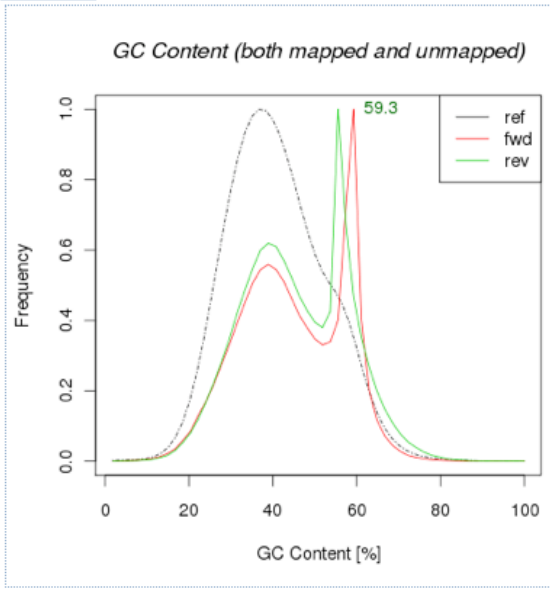
Mapping data

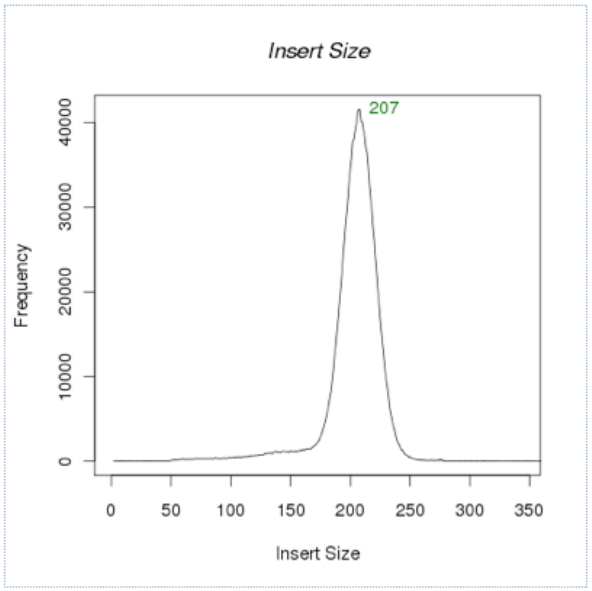
Reads:	3,012,698	Bases:	108,457,128	Cycles:	36
Reads w/adapter	1492 (0.0%)	Bases postclip:	107,123,848 (98.8% raw)	NPG QC:	pending
Reads mapped:	1,767,332 (58.7%)	Bases mapped:	63,276,180 (59.1%)	Error rate:	1.20%
Reads paired:	558,458 (18.5%)			Genotype check:	confirmed (NA20508:1.377)
Reads mapped (rmdup):	1,767,070 (58.7%)	Bases mapped (rmdup):	63,266,807 (59.1%)	Duplication rate:	0.01%

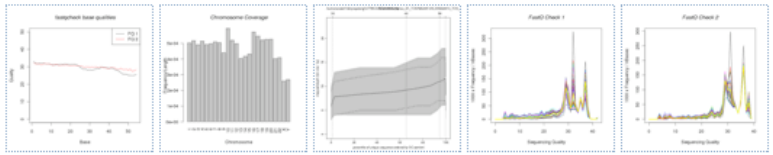
Lane 4

NA20513_TOS : NA20513-TOS 1 : 2658_1 [npg]

QC Plots







Mapping data

Reads:	1,948,762	Bases:	105,233,148	Cycles:	54
Reads w/adapter	1308850 (67.2%)	Bases postclip:	102,366,278 (97.3% raw)	NPG QC:	fail
Reads mapped:	1,500,098 (77.0%)	Bases mapped:	80,230,176 (78.4%)	Error rate:	0.80%
Reads paired:	1,451,952 (74.5%)			Genotype check:	confirmed (NA20513:1.600)
Reads mapped (rmdup):	1,498,195 (76.9%)	Bases mapped (rmdup):	80,129,087 (78.3%)	Duplication rate:	0.13%

Lane QC : failed (Auto QC : failed)

Library QC

QC Grid

Mouse : CAST_Ei Mouse Genome : CASTEi200A : CAST_Ei_SLX_200_NOPCR_2 [SS] [NPG]

Show Navigation

Library QC : passed

pending | failed | passed

8 lanes in library

Insert 175 Type No PCR

Lane data

Pass	Fail	Pend	Name	Auto QC	Cycles	Bases	Post-clip	Adapter	Mapped	Paired	Rmdup	Genotype
◉	○	○	1842_8	failed	37	101,328,052	99.1%	0.2%	95.6%	89.2%	95.1%	confirmed (14.399)
◉	○	○	2685_1	failed	54	100,093,860	96.3%	0.2%	96.1%	90.5%	95.8%	confirmed (14.258)
◉	○	○	2685_2	failed	54	105,559,092	96.7%	0.2%	96.1%	90.5%	95.8%	confirmed (15.258)
◉	○	○	2685_3	failed	54	102,666,636	96.2%	0.1%	95.9%	90.3%	95.6%	confirmed (10.190)
◉	○	○	2685_5	failed	54	102,552,156	95.6%	0.1%	95.8%	90.1%	95.6%	confirmed (12.666)
◉	○	○	2685_6	passed	54	103,556,016	96.8%	0.1%	96.3%	90.8%	96.0%	confirmed (17.003)

Real Example from 2009

Evidence
- Many sequencing lanes showing increasing GC levels
- GC-depth plot showing skew towards higher GC regions of the genome
  - Resulting in more false positives for structural variation calling

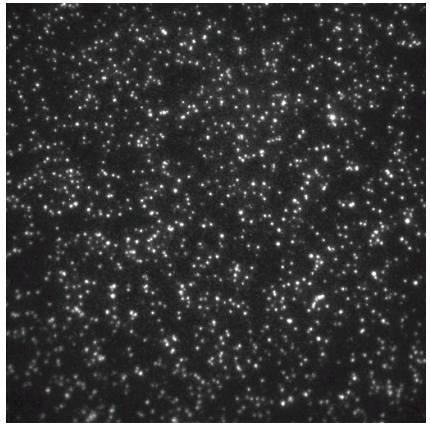
Possible sources
- Library preparation
  - Type of library prep. method
  - Library prep. materials changed
  - DNA shearing method
- Sequencing machine or chemistry issue
- Sample
  - DNA poor quality/biased

Key questions
- Are all library types affected?
- Are all runs affected?
- Are all DNA samples affected?

Final Proof
- An individual sequencing library sequenced in January, March, July showed increasing levels of GC in the lanes off the individual library

Explanation: Vendor had changed the chemistry on the machine that resulted in GC richer clusters on the slide

Base Quality Recalibration



Each base call has an associated base call quality

- What is the chance that the base call is incorrect?
  - Illumina evidence: intensity values + cycle
- Phred values (log scale)
  - Q10 = 1 in 10 chance of base call incorrect
  - Q20 = 1 in 100 chance of base call incorrect
- Accurate base qualities essential measure in variant calling

Rule of thumb: Anything less than Q20 is not useful data

Typically phred values max. out at Q35-40

Illumina sequencing

- Control lane used to generate a quality calibration table
- If no control lane – then use pre-computed calibration tables

Quality recalibration

- 1000 genomes project sequencing carried out on multiple platforms at multiple different sequencing centres
- Are the quality values comparable across centres/platforms given they have all been calibrated using different methods?

Base Quality Recalibration

Original recalibration algorithm

- Align subsample of reads from a lane to human reference
- Exclude all known dbSNP sites
  - Assume all other mismatches are sequencing errors
- Compute a new calibration table bases on mismatch rates per position on the read

Pre-calibration sequence reports Q25 base calls

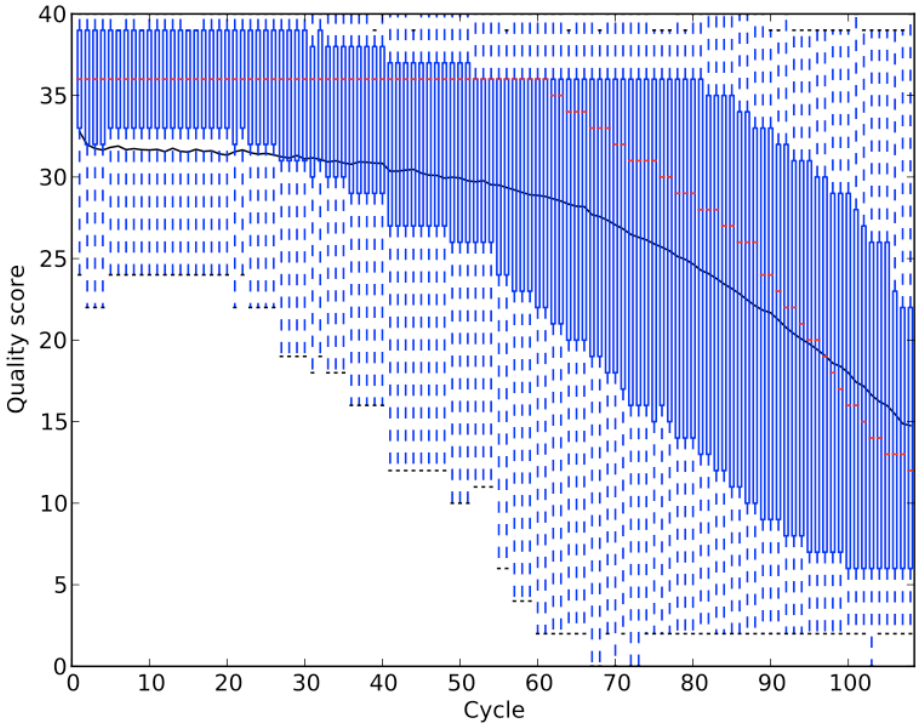
- After alignment - it may be that these bases actually mismatch the reference at a 1 in 100 rate, so are actually Q20

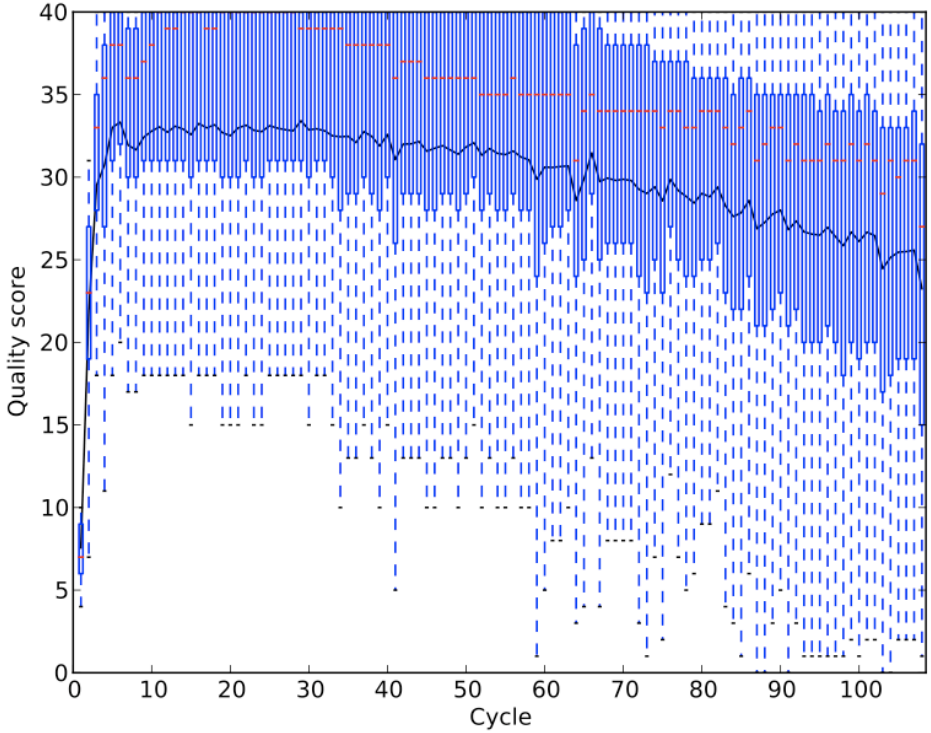
Recent improvements – GATK package

- Reported/original quality score
- The position within the read
- The preceding and current nucleotide (sequencing chemistry effect) observed by the sequencing machine
- Probability of mismatching the reference genome

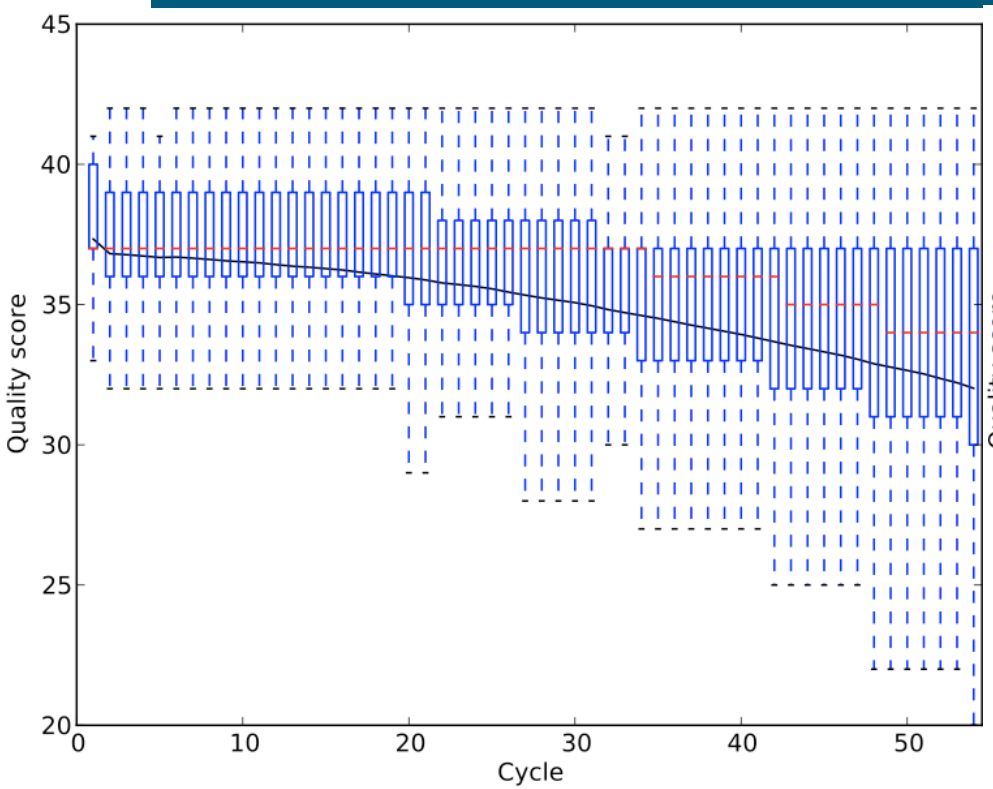
NOTE: requires a reference genome and good catalog of variable sites to exclude these sites

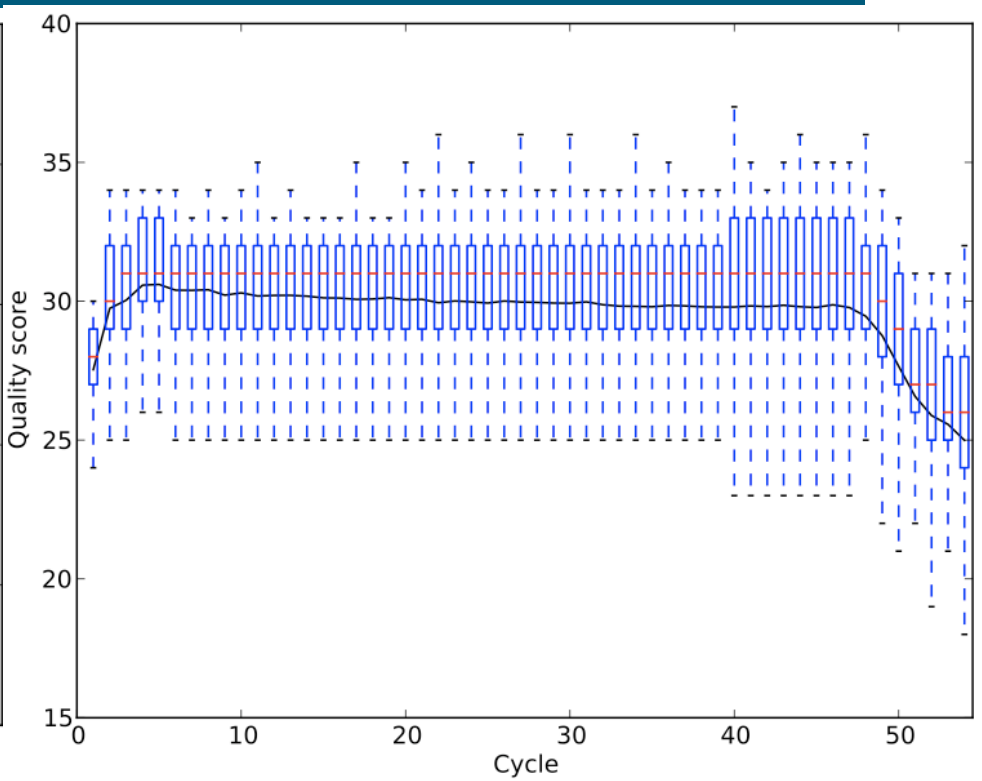
Base Quality Recalibration Effects





Base Quality Recalibration Effects





Potential Explanations

- an excess of mismatches and often at the beginning and end of the reads
- Illumina is now using a training procedure which treats the first five cycles in a read differently – some of this affect is real
- adapter sequence being seen at the ends of reads – high mismatches vs. human reference

Latest GATK modified to take these into account

- http://www.broadinstitute.org/gsa/wiki/index.php/Base_quality_score_recalibration

Tutorial 1: Overview, Applications, QC and Formats

- Overview
- Quality Control
- **Next-gen Data Formats**
- Short Read Alignment
- Sequence Assembly
- Case Study: 1000 Genomes
- Experimental Design

Next-gen Data Formats

Raw Data

- Initially stored raw images (tiff files)
  - Millions of images generated per sequencing run
  - Analyse and delete!
- SRF
  - Submission to archives (ERA/SRA)
  - Wrapper for the technology sequencing specific output files
- Fastq
  - Read sequences and base qualities
- SAM/BAM
  - Read sequences, qualities, and meta data
  - Multiple technologies, libraries, lanes
- VCF
  - Variant call Format
  - Store SNPs, short indels, and recently extended for structural variants

Fastq

FASTQ has emerged as a *de facto* file format for sharing sequencing read data

- Simple extension to the FASTA format:
- Sequence and an associated per base quality score

Original Sanger standard for capillary data

Format

- Subset of the ASCII printable characters
- ASCII 33–126 inclusive with a simple offset mapping
- perl -w -e "print (unpack('C', '%') - 33);"

	Range	Offset	Type	Range
Sanger standard				
`fastq-sanger`	33–126	33	PHRED	0 to 93
Solexa/early Illumina				
`fastq-solexa`	59–126	64	Solexa	−5 to 62
Illumina 1.3+				
`fastq-illumina`	64–126	64	PHRED	0 to 62

```
@SRR014849.1 EIXKN4201CFU84 length=93
GGGGGGGGGGGGGGGGCTTTTTTTGTTTGGAACCGAAAGG
GTTTTGAATTTCAAACCCTTTTCGGTTTCCAACCTTCCAA
AGCAATGCCAATA
+SRR014849.1 EIXKN4201CFU84 length=93
3+&$#""""""""""""7F@71,'";C?,B;?6B;:EA1EA
1EA5'9B:?:#9EA0D@2EA5':>5?:%A;A8A;?9B;D@
/=<?7=9<2A8==
```

@title and optional description
sequence line(s)
+optional repeat of title line
quality line(s)

SAM/BAM Format

Proliferation of alignment formats over the years: Cigar, psl, gff, xml etc.

SAM (Sequence Alignment/Map) format

- Single unified format for storing read alignments to a reference genome

BAM (Binary Alignment/Map) format

- Binary equivalent of SAM
- Developed for fast processing/indexing

Advantages

- Can store alignments from most aligners
- Supports multiple sequencing technologies
- Supports indexing for quick retrieval/viewing
- Compact size (e.g. 112Gbp Illumina = 116Gbytes disk space)
- Reads can be grouped into logical groups e.g. lanes, libraries, individuals/genotypes
- Supports second best base call/quality for hard to call bases

Possibility of storing raw sequencing data in BAM as replacement to SRF & fastq

Read Entries in SAM

No.	Name	Description
1	QNAME	Query NAME of the read or the read pair
2	FLAG	Bitwise FLAG (pairing, strand, mate strand, etc.)
3	RNAME	Reference sequence NAME
4	POS	1-Based leftmost POSition of clipped alignment
5	MAPQ	MAPping Quality (Phred-scaled)
6	CIGAR	Extended CIGAR string (operations: MIDNSHP)
7	MRNM	Mate Reference NaMe ('=' if same as RNAME)
8	MPOS	1-Based leftmost Mate POSition
9	ISIZE	Inferred Insert SIZE
10	SEQ	Query SEQuence on the same strand as the reference
11	QUAL	Query QUALity (ASCII-33=Phred base quality)

Heng Li , Bob Handsaker , Alec Wysoker , Tim Fennell , Jue Ruan , Nils Homer , Gabor Marth , Goncalo Abecasis , Richard Durbin , and 1000 Genome Project Data Processing Subgroup (2009) The Sequence Alignment/Map format and SAMtools, *Bioinformatics,* 25:2078-2079

Extended Cigar Format

Cigar has been traditionally used as a compact way to represent a sequence alignment

Operations include

- M - match or mismatch
- I - insertion
- D - deletion

SAM extends these to include

- S - soft clip
- H - hard clip
- N - skipped bases
- P – padding

E.g. Read: `ACGCA-TGCAGTtagacgt`

Ref: `ACTCAGTG--GT`

Cigar: 5M1D2M2I2M7S

What is the cigar line?

E.g. Read: `ACGCA-TGCAGTtagacgt`

Ref: `ACTCAGTG--GT`

Cigar: 5M1D2M2I2M7S

E.g. Read: `tgtcgtcACGCATG---CAGTtagacgt`

Ref: `ACGCATGCGGCAGT`

Cigar:

Read Group Tag

Each lane (or equivalent unit) has a unique read group (RG) tag

1000 Genomes

- Meta information derived from DCC

RG tags

- ID: SRR/ERR number
- PL: Sequencing platform
- PU: Run name
- LB: Library name
- PI: Insert fragment size
- SM: Individual
- CN: Sequencing center

Activity 2: Interpreting SAM/BAM files

From reading page 4 of the SAM specification, look at the following line from the header of the BAM file:

@RG ID:ERR001711 PL:ILLUMINA LB:g1k-sc-NA12878-CEU-1 PI:200 DS:SRP000032 SM:NA12878 CN:SC

What does RG stand for?

What is the sequencing platform?

What is the sequencing centre?

What is the lane accession number?

What is the expected fragment insert size?

1000 Genomes BAM File

```
@HD     VN:1.0  GO:none SO:coordinate
@SQ     SN:1    LN:249250621    AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:1b22b98cdeb4a9304cb5d48026a85128
@SQ     SN:2    LN:243199373    AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:a0d9851da00400dec1098a9255ac712e
@SQ     SN:3    LN:198022430    AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:fdfd811849cc2fadebc929bb925902e5
@SQ     SN:4    LN:191154276    AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:23dccd106897542ad87d2765d28a19a1
@SQ     SN:5    LN:180915260    AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:0740173db9ffd264d728f32784845cd7
@SQ     SN:6    LN:171115067    AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:1d3a93a248d92a729ee764823acbbc6b
@SQ     SN:7    LN:159138663    AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:618366e953d6aaad97dbe4777c29375e
@SQ     SN:8    LN:146364022    AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:96f514a9929e410c6651697bded59aec
@SQ     SN:9    LN:141213431    AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:3e273117f15e0a400f01055d9f393768
@SQ     SN:10   LN:135534747    AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:988c28e000e84c26d552359af1ea2e1d
@SQ     SN:11   LN:135006516    AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:98c59049a2df285c76ffb1c6db8f8b96
@SQ     SN:12   LN:133851895    AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:51851ac0e1a115847ad36449b0015864
@SQ     SN:13   LN:115169878    AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:283f8d7892baa81b510a015719ca7b0b
@SQ     SN:14   LN:107349540    AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:98f3cae32b2a2e9524bc19813927542e
@SQ     SN:15   LN:102531392    AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:e5645a794a8238215b2cd77acb95a078
@SQ     SN:16   LN:90354753     AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:fc9b1a7b42b97a864f56b348b06095e6
@SQ     SN:17   LN:81195210     AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:351f64d4f4f9ddd45b35336ad97aa6de
@SQ     SN:18   LN:78077248     AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:b15d4b2d29dde9d3e4f93d1d0f2cbc9c
@SQ     SN:19   LN:59128983     AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:1aacd71f30db8e561810913e0b72636d
@SQ     SN:20   LN:63025520     AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:0dec9660ec1efaaf33281c0d5ea2560f
@SQ     SN:21   LN:48129895     AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:2979a6085bfe28e3ad6f552f361ed74d
@SQ     SN:22   LN:51304566     AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:a718acaa6135fdca8357d5bfe94211dd
@SQ     SN:X    LN:155270560    AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:7e0e2e580297b7764e31dbc80c2540dd
@SQ     SN:Y    LN:59373566     AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:1fa3474750af0948bdf97d5a0ee52e51
@SQ     SN:MT   LN:16569        AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:c68f52674c9fb33aef52dcf399755519
@SQ     SN:GL000207.1   LN:4262 AS:NCBI37       UR:file:/lustre/scratch102/projects/g1k/ref/main_project/human_g1k_v37.fasta    M5:f3814841f1939d3ca19072d9e89f3fd7
@RG     ID:ERR001268    PL:ILLUMINA     LB:NA12878.1    PI:200  DS:SRP000032    SM:NA12878      CN:MPIMG
@RG     ID:ERR001269    PL:ILLUMINA     LB:NA12878.1    PI:200  DS:SRP000032    SM:NA12878      CN:MPIMG
@RG     ID:ERR001698    PL:ILLUMINA     LB:g1k-sc-NA12878-CEU-1 PI:200  DS:SRP000032    SM:NA12878      CN:SC
@RG     ID:SRR001114    PL:ILLUMINA     LB:Solexa-3620  PI:0    DS:SRP000032    SM:NA12878      CN:BI
@RG     ID:SRR001115    PL:ILLUMINA     LB:Solexa-3623  PI:0    DS:SRP000032    SM:NA12878      CN:BI
@PG     ID:GATK TableRecalibration.4    VN:v2.2.16      CL:Covariates=[ReadGroupCovariate, QualityScoreCovariate, DinucCovariate, CycleCovariate], use_original_quals=true,
default_read_group=DefaultReadGroup, default_platform=ILLUMINA, force_read_group=null, force_platform=null, solid_recal_mode=SET_Q_ZERO, window_size_nqs=5, homopolymer_nback=7,
exception_if_no_tile=false, pQ=5, maxQ=40, smoothing=1
@PG     ID:bwa  VN:0.5.5
```

samtools view –H my.bam

How is the BAM file sorted?
How many different sequencing centres contributed lanes to this BAM file?
What is the alignment tool used to create this BAM file?

How many different sequencing libraries are there in this BAM? Hint: RG tag

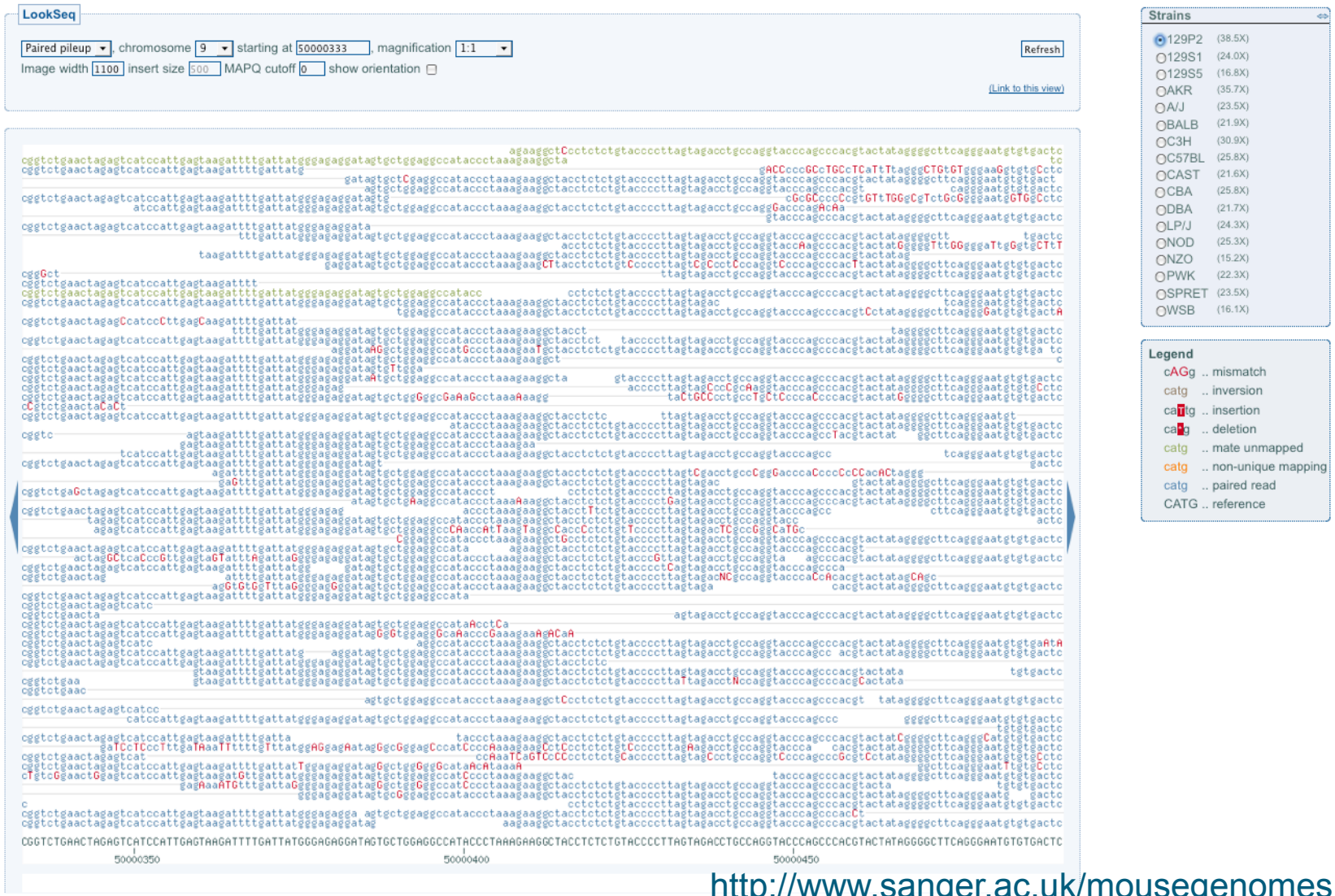
SAM/BAM Tools

Well defined specification for SAM/BAM

Several tools and programming APIs for interacting with SAM/BAM files

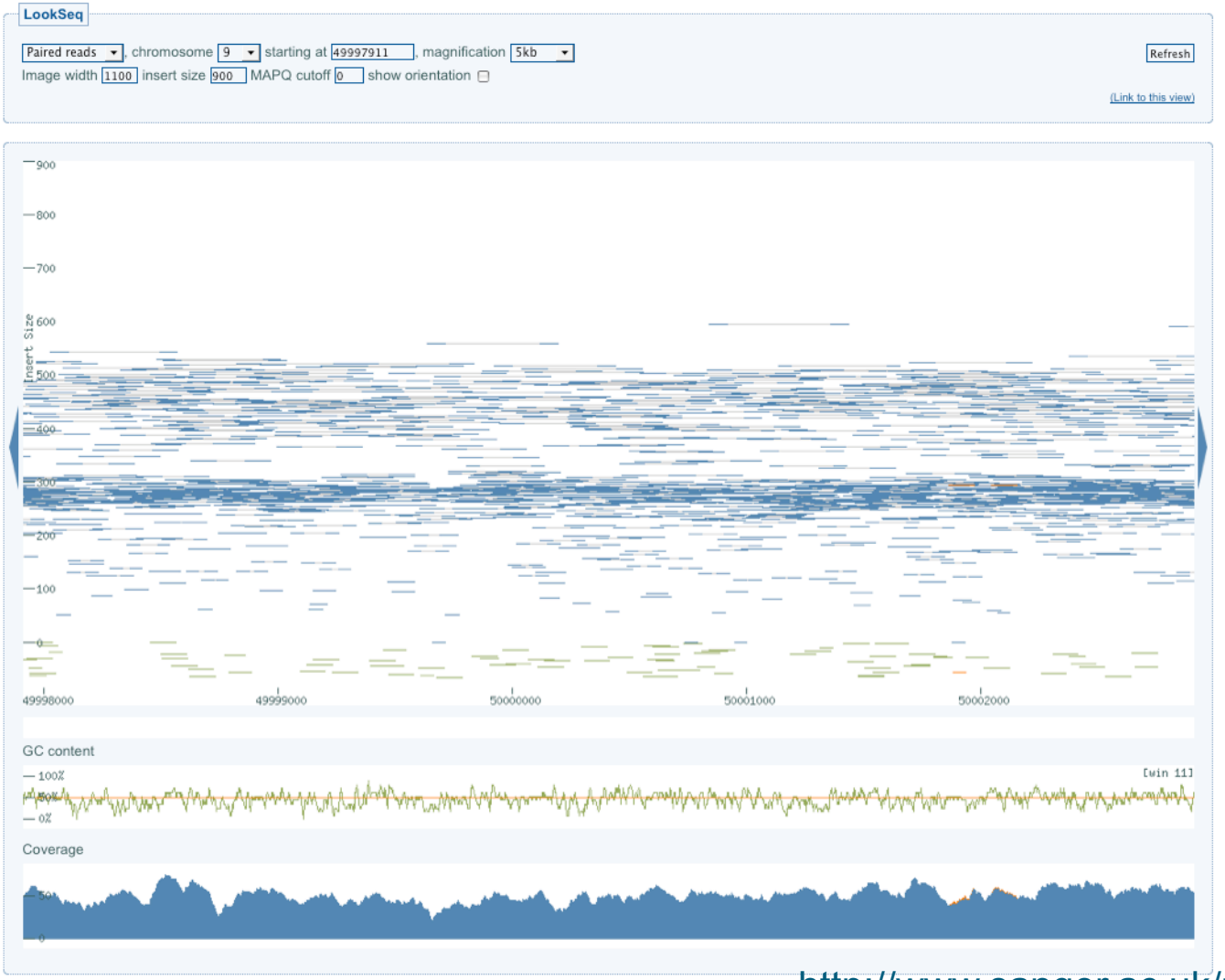
- Samtools - Sanger/C (http://samtools.sourceforge.net)
  - Convert SAM <-> BAM
  - Sort, index, BAM files
  - Flagstat – summary of the mapping flags
  - Merge multiple BAM files
  - Rmdup – remove PCR duplicates from the library preparation
- Picard - Broad Institute/Java (http://picard.sourceforge.net)
  - MarkDuplicates, CollectAlignmentSummaryMetrics, CreateSequenceDictionary, SamToFastq, MeanQualityByCycle, FixMateInformation…….
- Bio-SamTool – Perl (http://search.cpan.org/~lds/Bio-SamTools/)
- Pysam – Python (http://code.google.com/p/pysam/)

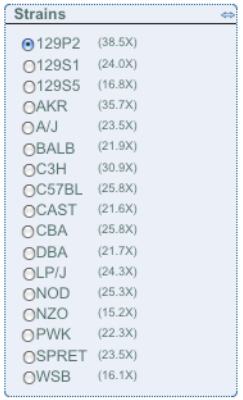
BAM Visualisation

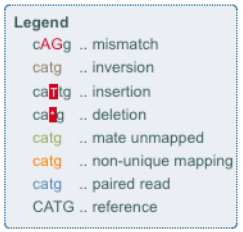


http://www.sanger.ac.uk/mousegenomes

BAM Visualisation







http://www.sanger.ac.uk/mousegenomes

BAM Visualisation - IGV

BAM Visualisation Tools

Requirements
- BAM files over http/ftp, Annotation data, multiple tracks
- Data on demand – only open a window of data at a time

LookSeq
- http://www.sanger.ac.uk/resources/software/lookseq/
- Perl/CGI/Samtools
- Web based
- Connects to BAM files over ftp
- Large genomes

IGV
- http://www.broadinstitute.org/igv/
- Java/Picard
- Standalone application
- Multiple tracks/BAM files simultaneously
- Annotation data via DAS
- Large genomes

Savant

IGB

Recommended Reading

Cock et al (2009) The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants, Nucleic Acids Research, 38 (6)1767-1771

Li *et al.* (2009) The Sequence Alignment/Map format and SAMtools, Bioinformatics, 25 (16):2078-2079

McKenna et al (2010) The Genome Analysis Toolkit: A MapReduce framework for analyzing next-generation DNA sequencing data, Genome Research, 20:1297-1303

Manske et al (2010) LookSeq: A browser-based viewer for deep sequencing data, Genome Research, 19:2125-2132

Savant: Genome Browser for High Throughput Sequencing Data
Marc Fiume; Vanessa Williams Andrew Brook; Michael Brudno
Bioinformatics; 2010 Aug 15;26(16):1938-44.

Tutorial 1: Overview, Applications, QC and Formats

- Overview
- Quality Control
- Next-gen Data Formats
- **Short Read Alignment**
- Sequence Assembly
- Case Study: 1000 Genomes
- Experimental Design

Short Read Alignment

Next-gen considerations

- Amount of data many orders of magnitude higher => memory + speed
- Different error profiles than the previous-generation technology
  - Roche 454 reads to have insertion or deletion errors at homopolymers
  - Illumina low quality calls can occur anywhere in a read
  - Increasing likelihood of sequence errors toward the end of the read for ABI SOLiD + illumina
  - Output of the SOLiD machine is a series of colors representing two nucleotides

Alignment itself is the process of determining the most likely source within the genome sequence for the observed DNA sequencing read

DNA based alignment mostly

- Very small evolutionary distances (human-human, strains of the reference genome)
- Assumptions about the number of expected mismatches
  - Allow for much faster processing

Short Read Alignment Algorithms

Large and ever growing number of implementations for short-read sequence alignment

Fundamental algorithms can be divided into two approaches

- Hash table based implementations
- Burrows-Wheeler transform (BWT)

Both approaches apply to sequence and colour space and all technologies

Heuristics to find potential locations on the genome

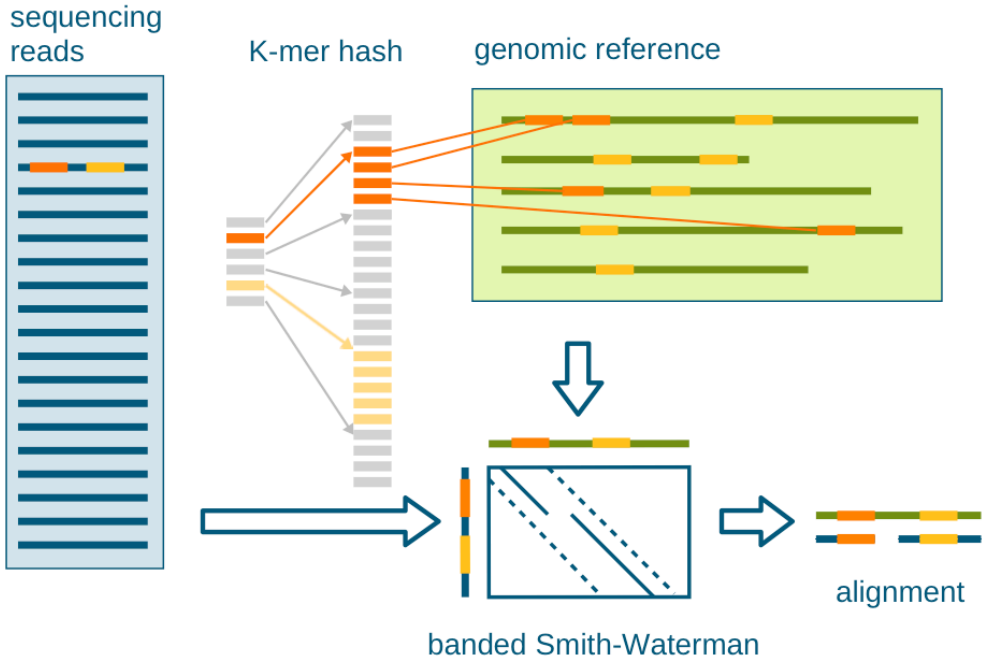
- Slower more accurate alignment run on a subset of potential locations
- Similar strategy to traditional read alignment algorithms: Blat, Blast, SSAHA2

Constant trade-off: speed vs. sensitivity

Guaranteed high accuracy will generally take longer

Hash Table Based Alignment

Hash table is a common data structure that is able to index complex and non-sequential data in a way that facilitates rapid searching



Hash of the reads: MAQ, ELAND, ZOOM and SHRiMP

- Smaller but more variable memory requirements

Hash the reference: SOAP, BFAST and MOSAIK

- Advantage: constant memory cost

Hash Table Based Alignment

Hash is typically built from a set of seeds spaced along the reference or read sequence

Seeds used in the hash table creation and the reads have been associated with the region of the genome

Specialized and accurate alignment algorithm is used to determine the exact placement

- Gapped and ungapped versions of Smith-Waterman
- Use the base qualities in order to determine the most likely alignment of a read
- Use information about where the mate pair aligns to
  - Maq: Require 1 mate to align ungapped and can do full Smith-Waterman to align its mate (e.g. indel in the mate)

Mapping quality

- A probability measure of the alignment being incorrect
- Single end mapping quality vs. paired end mapping quality
- Low complexity regions typically have low mapping qualities
- Typically represented as a phred score (i.e. Q10 = 1 in 10 incorrect, Q20 = 1 in 100 incorrect)

Mapping Qualities

http://www.sanger.ac.uk/mousegenomes

LookSeq viewer

BWT Based Alignment

New generation of short read aligners based on BWT

- BWA, SOAP2, BOWTIE

BWT commonly used in data compression – compressed suffix array

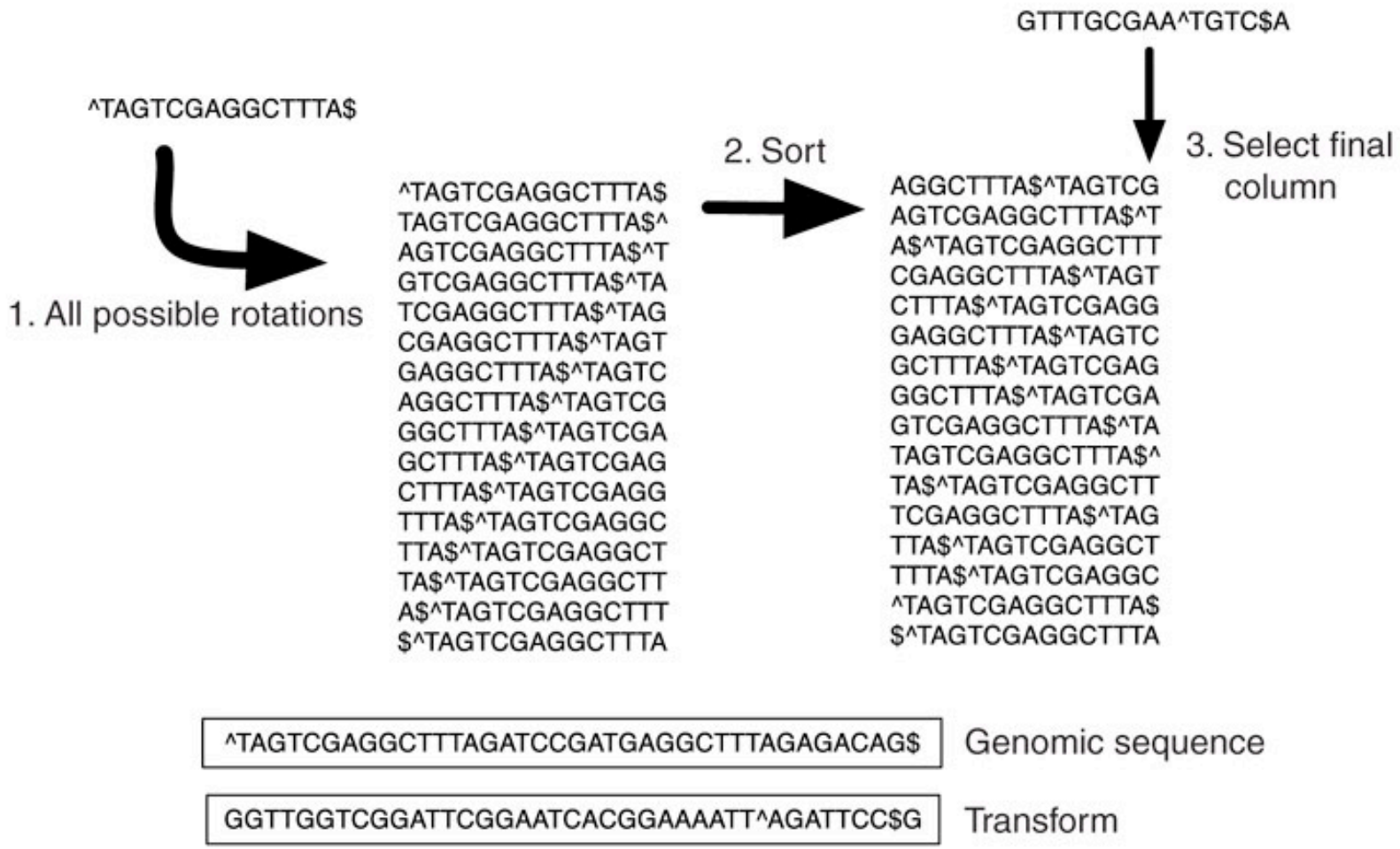
FM index data structure

- Ferragina and Manzini - suffix array is much more efficient if it is created from the BWT sequence
- FM index retains the suffix array's ability for rapid subsequence search
- Similar or smaller in size than input genome

Two step creation process

- Sequence order of the reference genome is modified using the BWT
- Final index is created; it is then used for rapid read placement on the genome

BWT Based Alignment



Li and Durbin (2009) Bioinformatics for further details

Which approach?

BWT implementations are much faster than their hash-based counterparts

- Several times faster still at slightly reduced sensitivity levels
- BOWTIE's reported 30-fold speed increase over hash-based MAQ with small loss in sensitivity
- Limitations to BWT approaches: BWA is only able to find alignments within a certain 'edit distance'
  - 100-bp reads, BWA allows 5 'edits
  - **Important to quality clip reads (-q in BWA)**
    - Non-A/C/G/T bases on reads are simply treated as mismatches

Hash based approaches are more suitable for divergent alignments

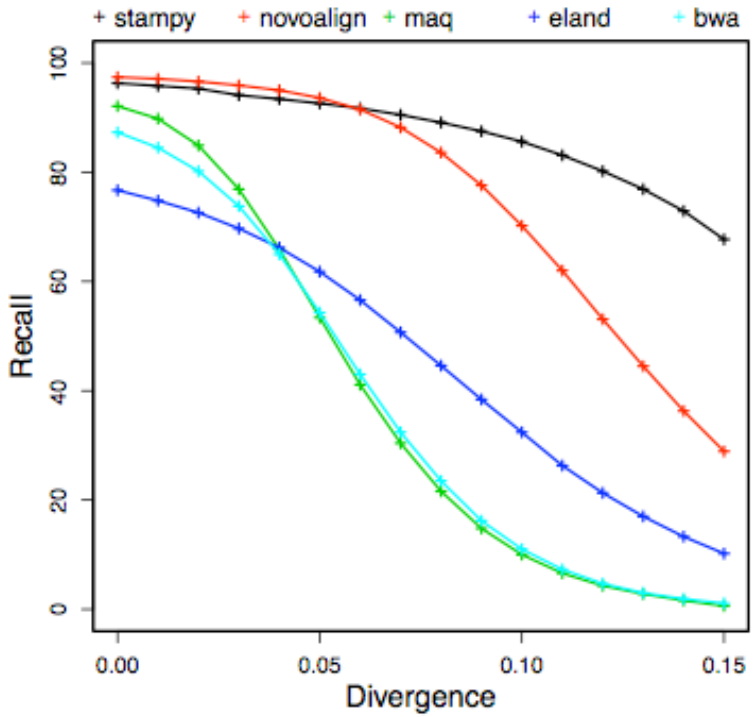
- Rule of thumb: <2% divergence -> BWT
  - E.g. human alignments
- >2% divergence -> hash based approach
  - E.g. wild mouse strains alignments

Aligner Comparisons

	Single-end			Paired-end		
Program	**Time (s)**	**Conf (%)**	**Err (%)**	**Time (s)**	**Conf (%)**	**Err (%)**
Bowtie-32	1271	79.0	0.76	1391	85.7	0.57
BWA-32	823	80.6	0.30	1224	89.6	0.32
MAQ-32	19797	81.0	0.14	21589	87.2	0.07
SOAP2-32	256	78.6	1.16	1909	86.8	0.78
Bowtie-70	1726	86.3	0.20	1580	90.7	0.43
BWA-70	1599	90.7	0.12	1619	96.2	0.11
MAQ-70	17928	91.0	0.13	19046	94.6	0.05
SOAP2-70	317	90.3	0.39	708	94.5	0.34
bowtie-125	1966	88.0	0.07	1701	91.0	0.37
BWA-125	3021	93.0	0.05	3059	97.6	0.04
MAQ-125	17506	92.7	0.08	19388	96.3	0.02
SOAP2-125	555	91.5	0.17	1187	90.8	0.14

0.09% SNP mutation rate, 0.01% indel mutation

BWA: Li and Durbin (2009) Bioinformatics



Varying mutation rates

Stampy: Lunter and Goodson, unpublished

Alignment Limitations

Read Length and complexity of the genome

- Very short reads difficult to align confidently to the genome
- Low complexity genomes present difficulties
  - Malaria is 80% AT rich – lots of low complexity AT stretches

Alignment around indels

- Next-gen alignments tend to accumulate false SNPs near true indel positions due to misalignment
- Smith-Waterman scoring schemes generally penalise a SNP less than a gap open
- New tools being developed to do a second pass on a BAM and locally realign the reads around indels and ‘correct’ the read alignments

High density SNP regions

- Seed and extend based aligners can have an upper limit on the number of consecutive SNPs in seed region of read (e.g. Maq – max of 2 mismatches in first 28bp of read)
- BWT based aligners work best at low divergence

Read Length

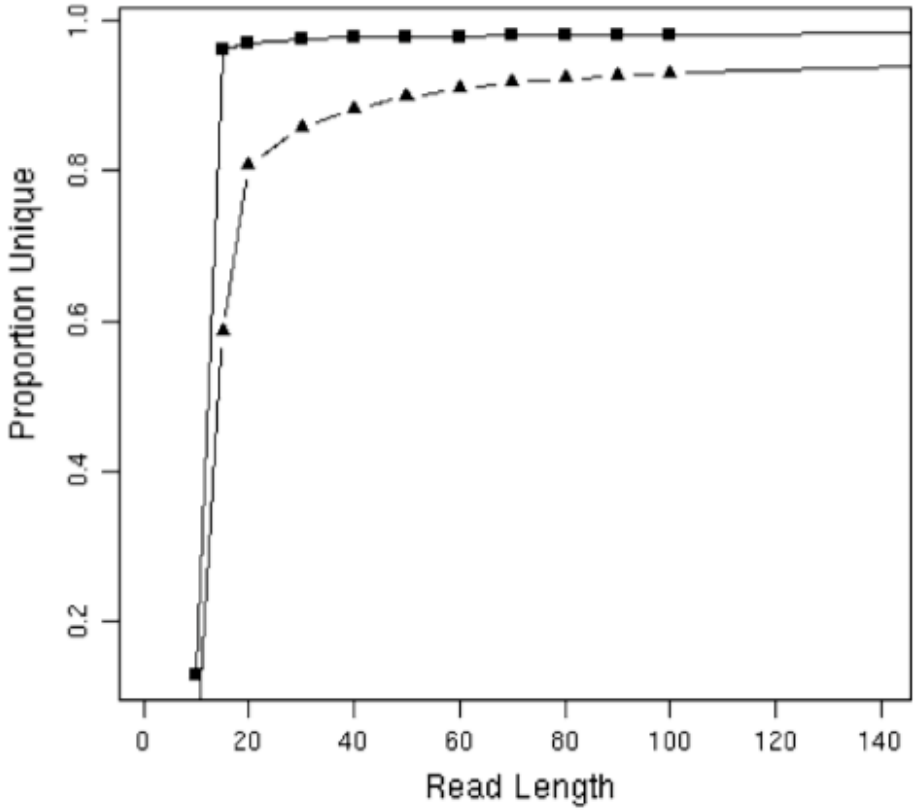
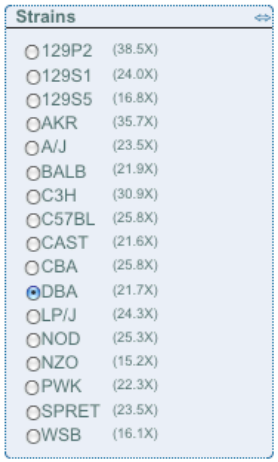
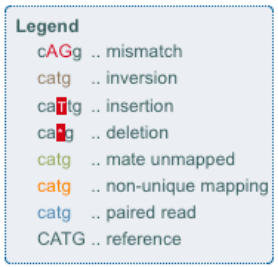


Fig. 1 The proportion of unique sequence in the *Streptococcus suis* (squares) and *Mus musculus* (triangles) genomes for varying read lengths. This graph indicates that read length has a critical affect on the ability to place reads uniquely to the genome

Example Indel





Using BWA

`bwa index [-a bwtsw|div|is] [-c] <in.fasta>`

- -a STR BWT construction algorithm: bwtsw or is
- bwtsw for human size genome, is for smaller genomes

`bwa aln [options] <prefix> <in.fq>`

- Align each single end fastq file individually
- Various options to change the alignment parameters/scoring matrix/seed length

`bwa sampe [options] <prefix> <in1.sai> <in2.sai> <in1.fq> <in2.fq>`

- sai files produced by aln step
- Produces SAM output

`bwa samse [-n max_occ] <prefix> <in.sai> <in.fq>`

- Unpaired reads – produces SAM output

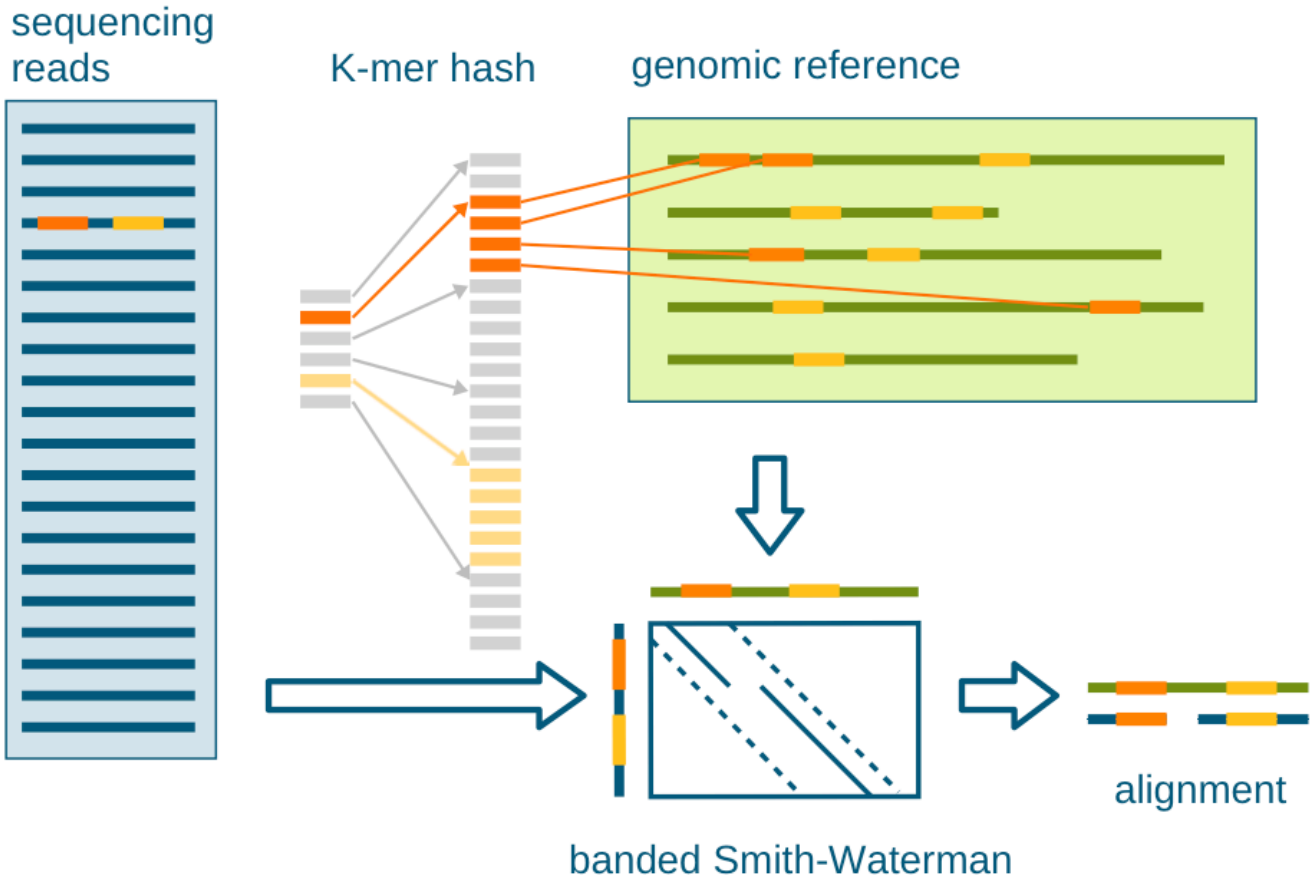
Use samtools to manipulate SAM file or convert to BAM

- `samtools view –bS in.sam > out.bam`

Using Smalt

Hash based aligner

- Range of sequencing platforms (Illumina/454/Capillary)



Ponstingl and Ning

Using Smalt

Index the genome

- `smalt index -k 13 -s 6 hs37k13s6 NCBI37.fasta`
- Produces an kmer index of the genome
- -k 13 specifies the length, -s 6 the spacing of the hashed words

Align the reads

- `smalt map -i 800 hs37k13s6 mate_1.fastq mate_2.fastq`

program	measured entity	100 bp				150 bp			
		0.5%	1%	2%	5%	0.5%	1%	2%	5%
SMALT	speed [10^6 pairs/h]	1.3	1.3	1.3	1.3	1.1	1.0	1.0	1.0
	fraction mapped [%]	96.7	96.7	96.6	95.8	97.4	95.6	94.6	95.1
	error rate [%]	0.01	0.03	0.06	0.22	<0.01	<0.01	<0.01	<0.01
BWA	speed [10^6 pairs/h]	2.8	1.9	1.2	–	2.0	1.2	0.6	–
	fraction mapped [%]	97.5	95.9	89.5	–	97.5	94.9	84.5	–
	error rate [%]	0.05	0.1	0.2	–	0.03	0.06	0.11	–
BOWTIE	speed [10^6 pairs/h]	6.3	5.3	–	–	4.8	4.1	–	–
	fraction mapped [%]	80.0	67.6	–	–	72.2	55.6	–	–
	error rate [%]	2.17	2.67	–	–	1.86	1.75	–	–

Parallelising Short Read Alignment

Simple parallelism by splitting data

- Split lane into 1Gbp chunks and align on different processors
  - BWA ~8 hours per 1Gbp chunk
- Merge chunk BAM files back into single lane BAM
  - Samtools merge command
- Merge lane BAM files for a library into single library level BAM
  - Run duplicate removal step (either samtools or MarkDuplicates in Picard)
  - Again check the BAM is not truncated!
- Merge sample libraries together into a single sample BAM
  - Run variant calling off sample level BAM

Computational issues to note

- Most aligners produce BAM files
- **Always** check for truncated BAM files
  - End of file marker
  - Or use samtools flagstat to count reads post alignment

Reading and Links

Flicek and Birney (2009) Sense from sequence reads: methods for alignment and assembly, Nature Methods, 6, S6 - S12

Samtools: http://samtools.sourceforge.net/

BWA: http://bio-bwa.sourceforge.net/bwa.shtml

Smalt: http://www.sanger.ac.uk/resources/software/smalt/

Local realignment

- http://www.broadinstitute.org/gsa/wiki/index.php/Local_realignment_around_indels

Tutorial 1: Overview, Applications, QC and Formats

- Overview
- Quality Control
- Next-gen Data Formats
- Short Read Alignment
- **Sequence Assembly**
- Case Study: 1000 Genomes
- Experimental Design

Sequence Assembly

A fundamental goal of DNA sequencing has been to generate large, continuous regions of DNA sequence

Whole-genome shotgun proven to be the most cost-effective and least labour intensive method of sequencing

- Human genome completed by a BAC-by-BAC strategy

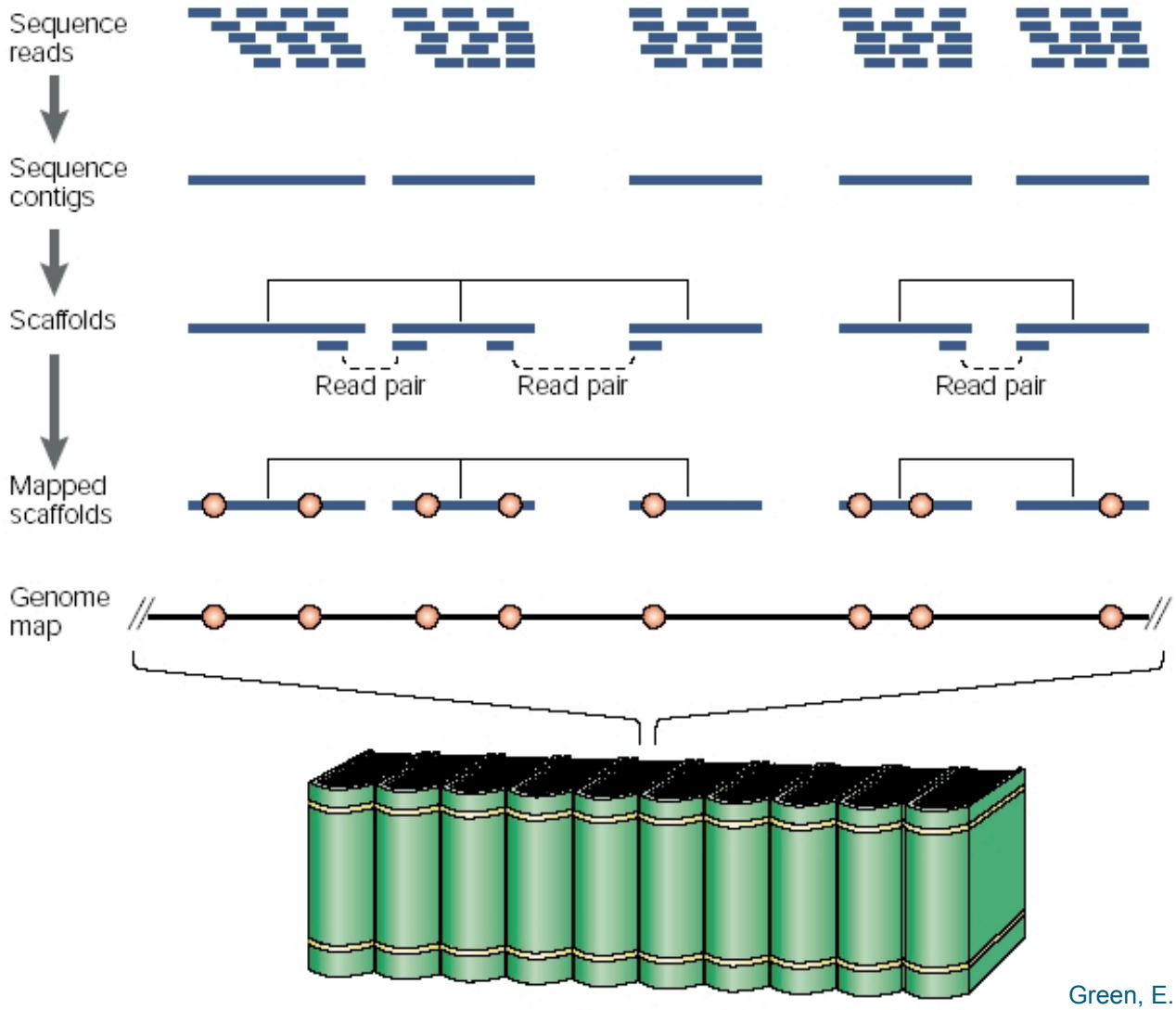
Capillary sequencing reads ~600-800bp in length

- Overlap based assembly algorithms (phrap, phusion, arachne, pcap…)
- Compute all overlaps of reads and then resolve the overlaps to generate the assembly

Volume + read length of data from next-gen sequencing machines meant that the read-centric overlap approaches were not feasible

- 1980's Pevzner *et al.* introduced an alternative assembly framework based on *de Bruijn* graph
- Based on a idea of a graph with fixed-length subsequences (k-mers)
- Key is that not storing read sequences – just k-mer abundance information in a graph structure

Sequence Assembly



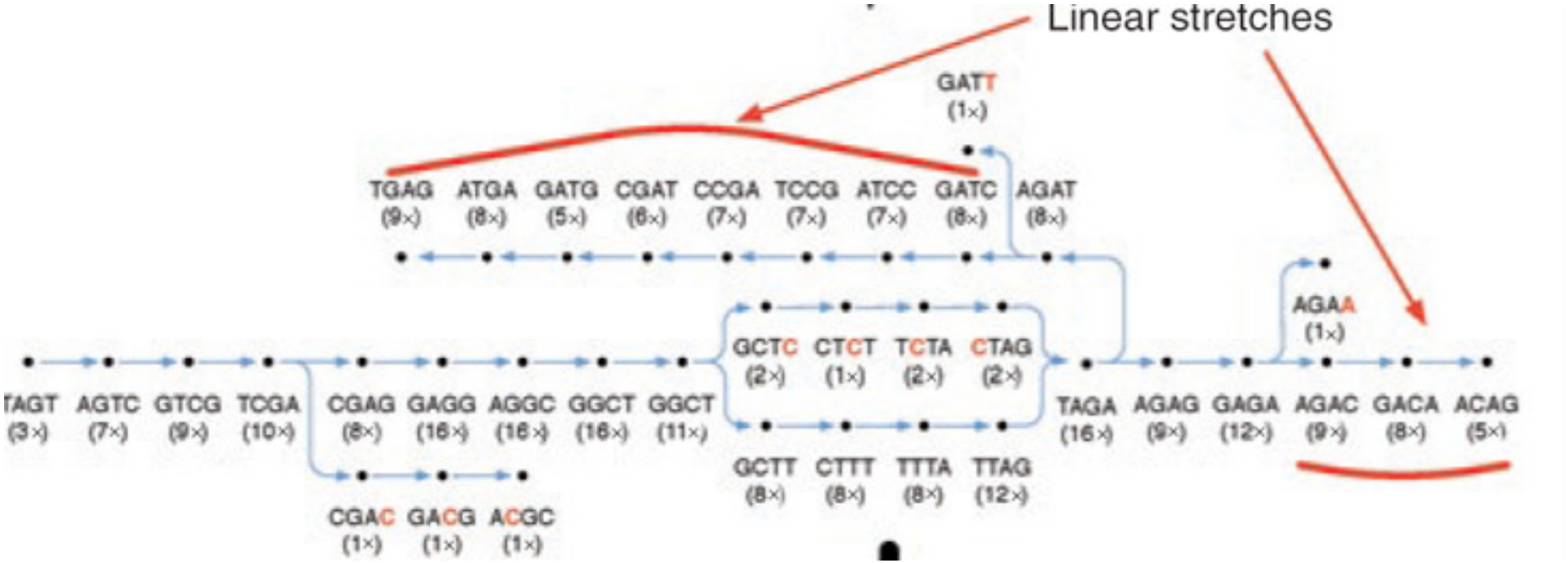
Green, E.D. (2001) NRG 2, 573-583

De Bruijn Graph Construction 1

Genome is sampled with random sequencing 7bp reads (e.g. Illumina or 454)
Note errors in the reads are represented in red

Flicek & Birney (2009) Nat Meth, 6

De Bruijn Graph Construction 2

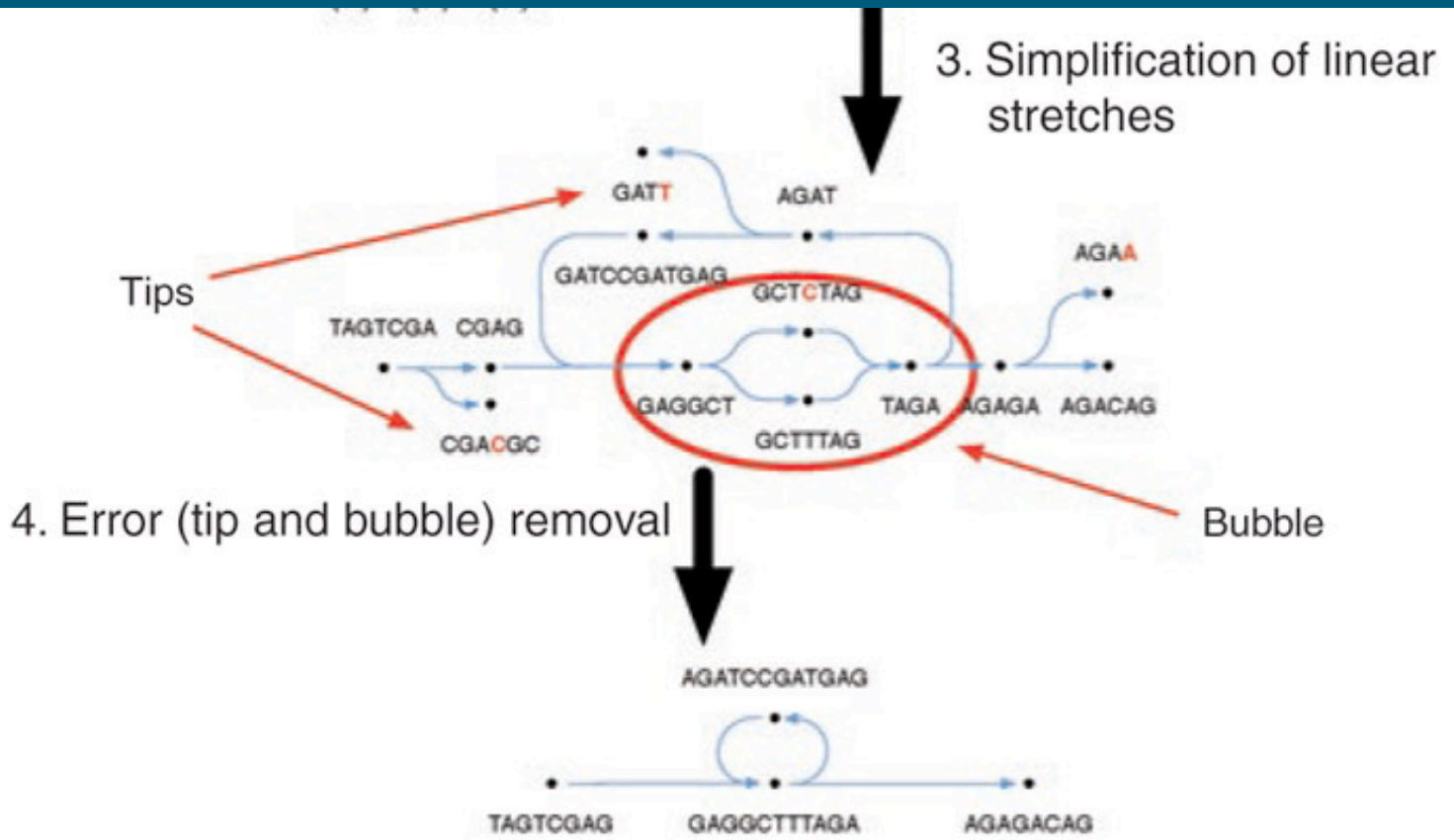


The *k*-mers in the reads (4-mers in this example) are collected into nodes and the coverage at each node is recorded (numbers at nodes)

Features

- continuous linear stretches within the graph
- Sequencing errors are low frequency tips in the graph

De Bruijn Graph Construction 3



Graph is simplified to combine nodes that are associated with the continuous linear stretches into single, larger nodes of various *k*-mer sizes

Error correction removes the tips and bubbles that result from sequencing errors

Final graph structure that accurately and completely describes in the original genome sequence

Next-gen Assemblers

First de Bruijn based assembler was Newbler

- Adapted it to handle main source of error in 454 data – indels in homopolymer tracts

Many de Bruijn assemblers subsequently developed

- SHARCGS, VCAKE, VELVET, EULER-SR, EDENA, ABySS and ALLPATHS
- Most can use mate-pair information

Few next-gen assemblers capable of assembling mammalian sized genomes out of the box

- ABySS – distributes de Bruijn graph over a network of computers using MPI protocol
- SOAP (BGI) and Cortex (unpublished)
  - Key: early removal of spurious errors from the data

Reads Pre-cleaning

Illumina sequencing error rate 1-2% depending on read length

many of the 25-mers will contain errors

Error correction before assembly for small data sets is less important

- Can be removed during the graph assembly

Large datasets

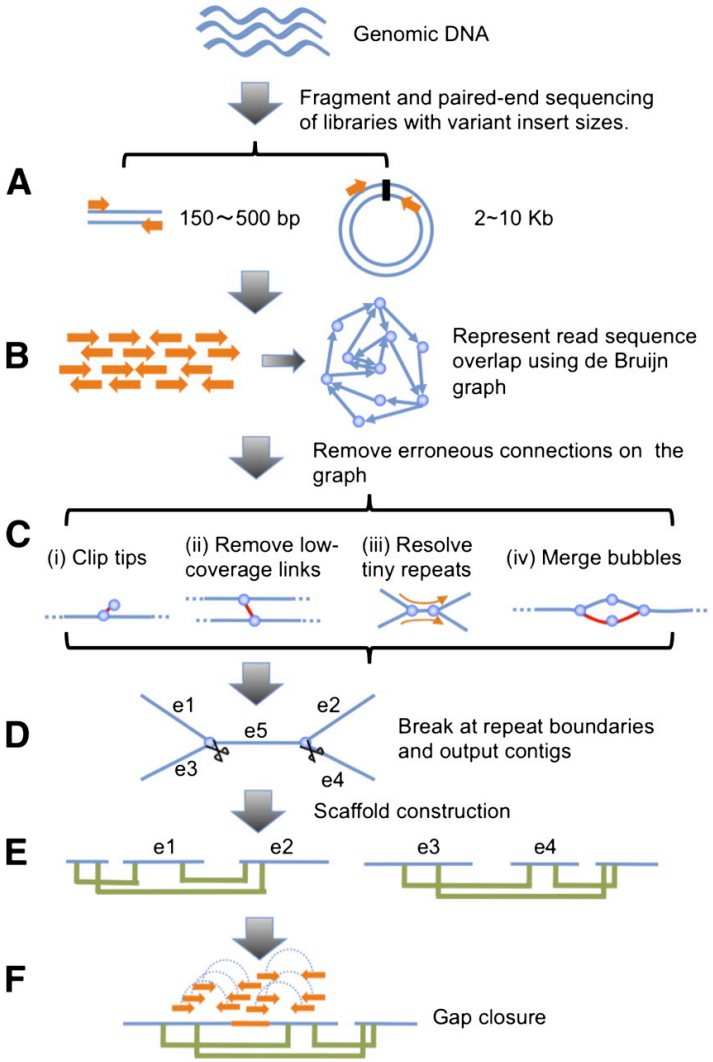
- Removal of singleton kmers is essential as will drastically reduce the memory footprint of the graph
- e.g. Asian human genome data, the total number of distinct 25-mers was reduced from 14.6 billion to 5.0 billion

Table 1. Summary of preassembly error correction in the Asian genome sequencing

	Total reads	Error-free reads (%)	25-mer no.
Original reads	4,083,271,441	60.1	14,551,534,812
After correction	3,312,495,883	74.0	4,966,416,149

Li et al (2009) Gen Res, 20

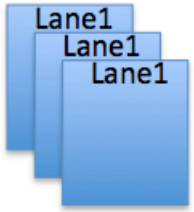
SOAP denovo Workflow



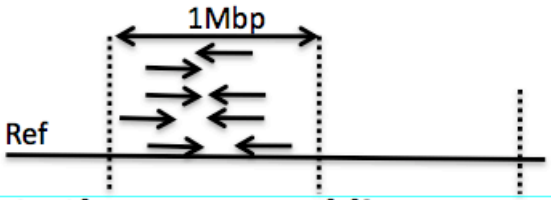
Li R et al. Genome Res. 2010;20:265-272

MouseGenomes Assembly – A low memory window based approach

1. Initial Read Alignment

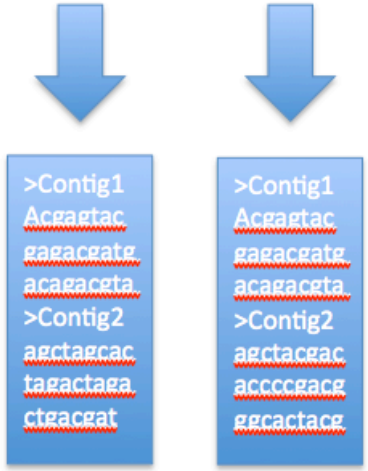


(Maq)

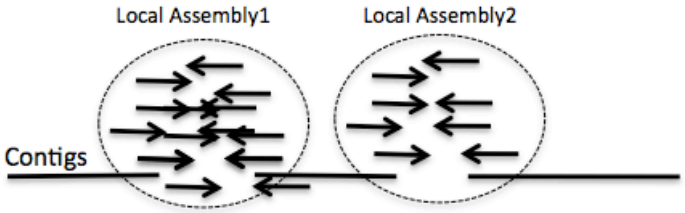


2. Cluster Assemblies

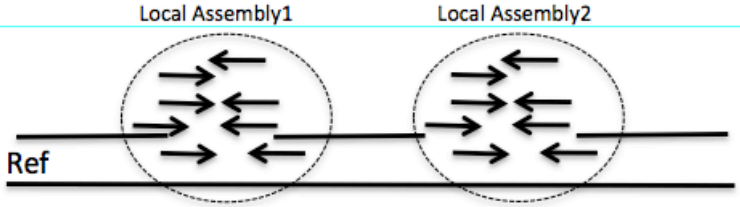
(Velvet)



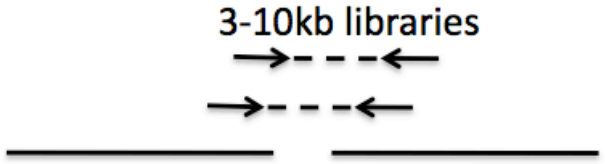
3. Gap filling



4. Reference based gap filling



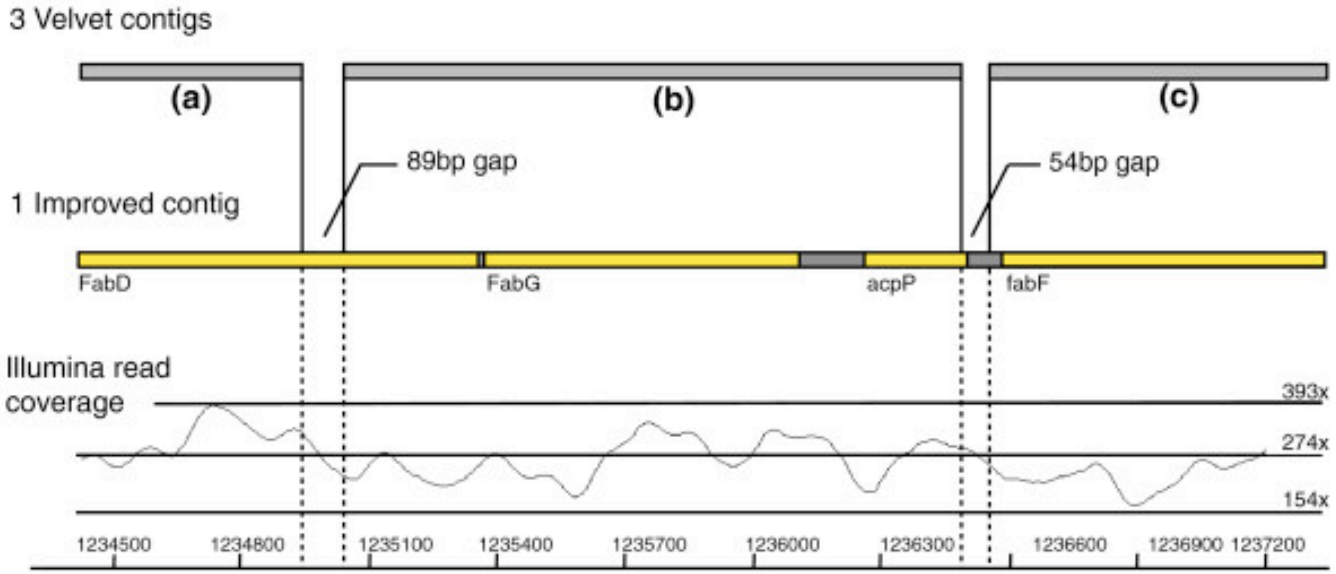
5. Scaffolding



Gap Filling

Assembly post processing step to close gaps between contigs
Gaps can be created by the assembler where there are changes in the read depth

- E.g. assembler might require a miniumum read depth at node in de Bruijn graph



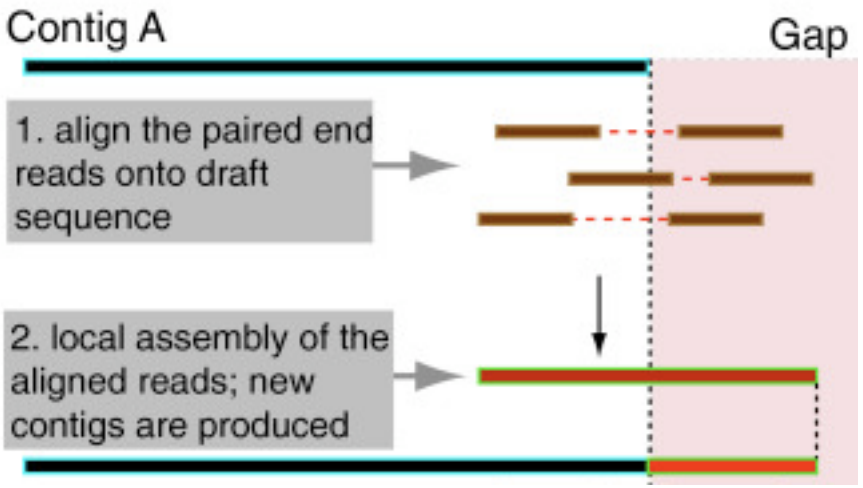
Closing gaps in *de novo* assembly comprising only Illumina reads. Schematic diagram showing the comparison of the original velvet assembly (3 contigs a, b and c) and the improved assembly in *Salmonella enterica*. The improved assembly was aligned to the reference sequence with 99.8% identity. The two closed gaps shown were 100% identical to the reference sequence. Contigs are indicated by grey bars; gene annotations are indicated by yellow boxes. Vertical lines highlight the gaps that are filled by IMAGE in the improved contigs. Below, a coverage plot showing the relatively even depth of coverage of realigned Illumina reads at the improved assembly, indicating no signature of misassembly.

Tsai et al. (2010) Gen Biol

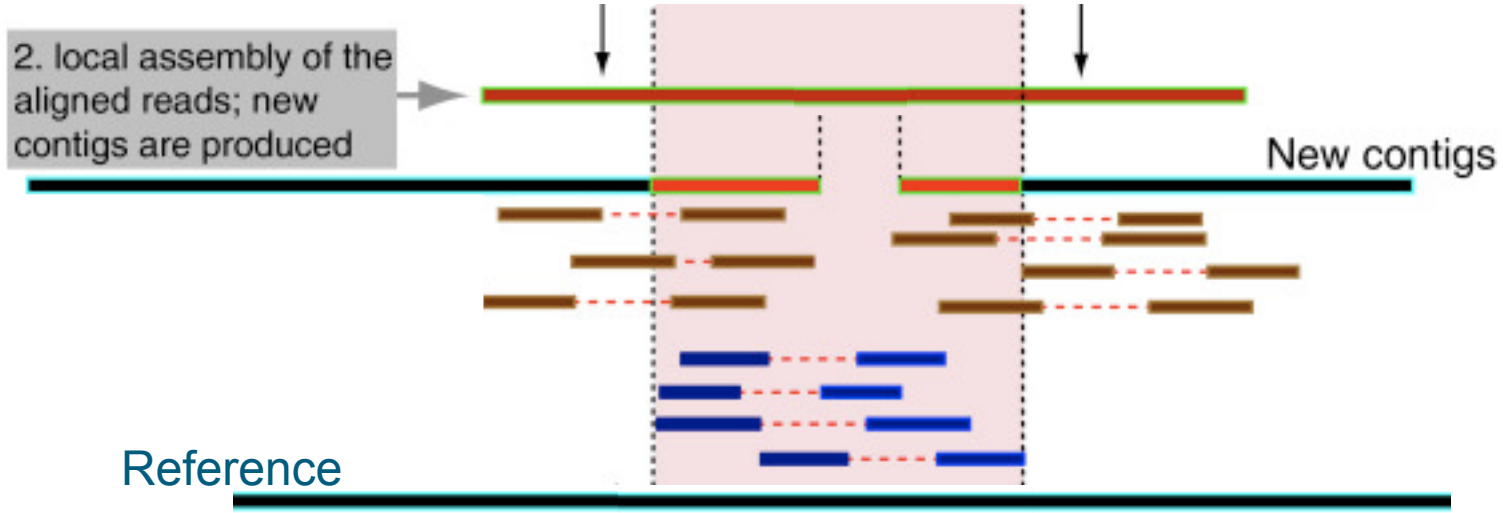
Gap Filling Algorithm

Non-reference based

- De novo contig extension



Reference guided



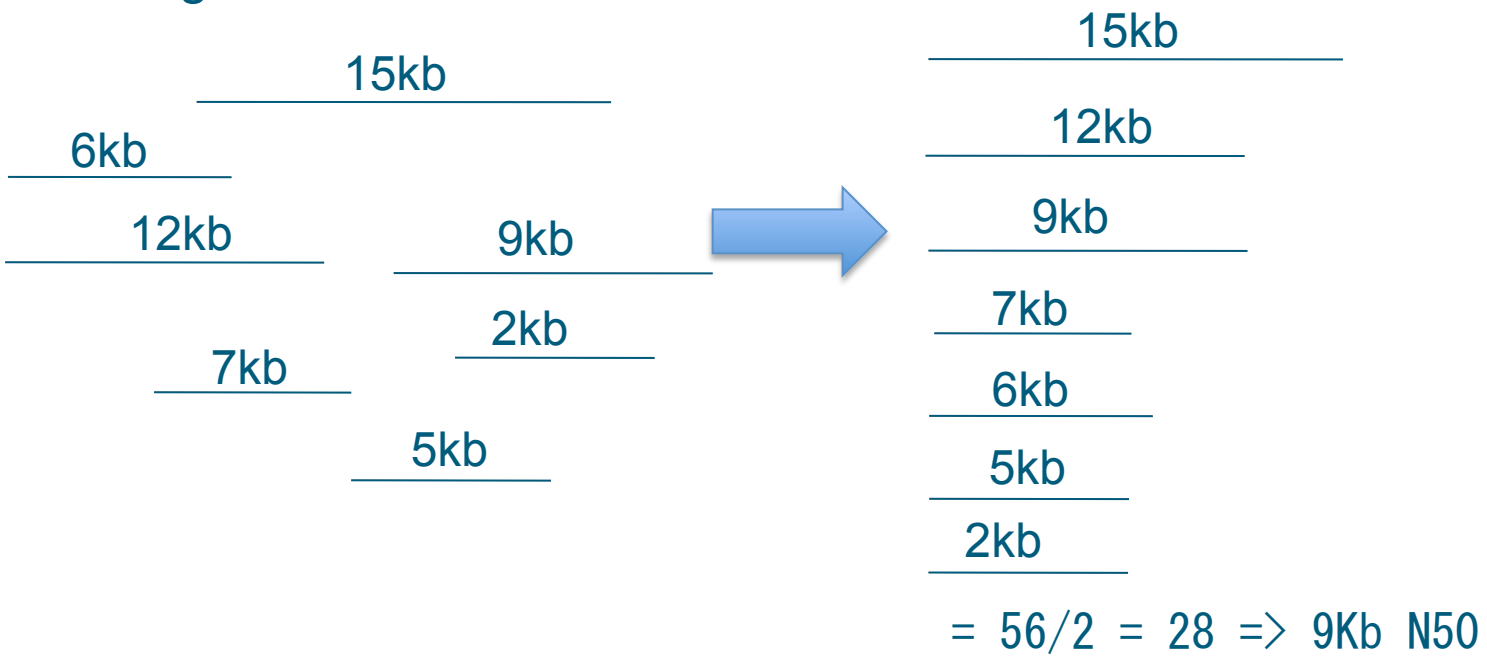
Assembly Evaluation – N50

N50 has traditionally been used to compare assemblies

If you order the set of contigs produced by the assembler by size

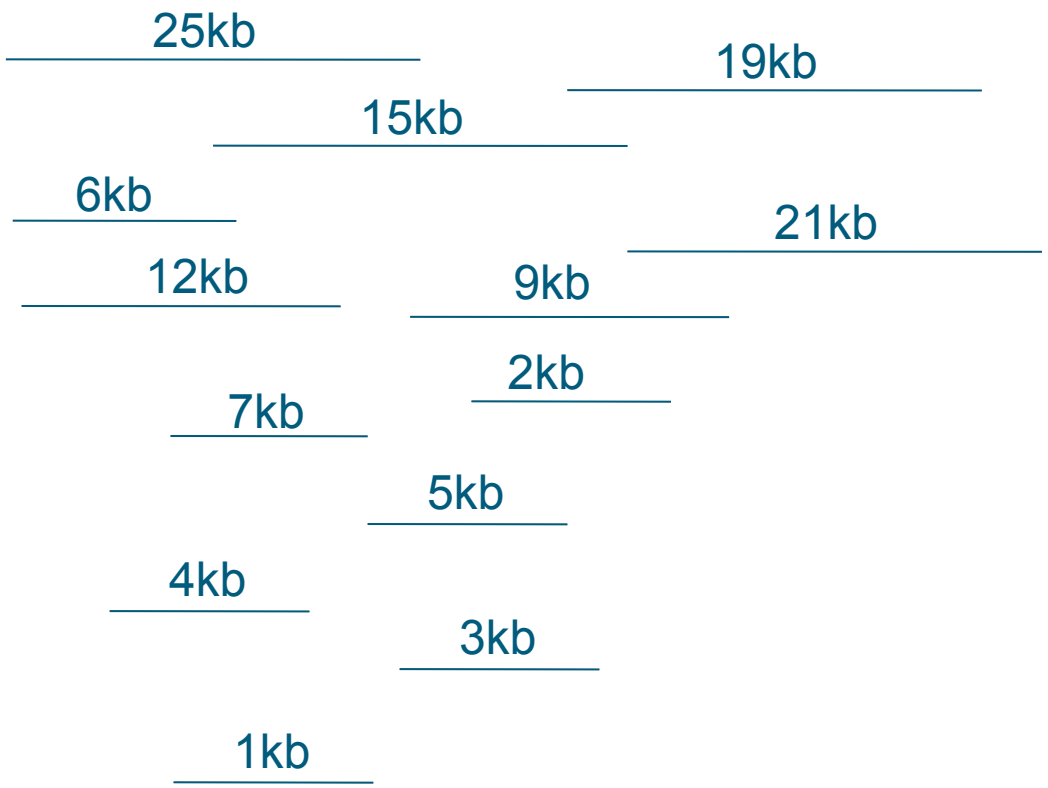
- N50 is the size of the contig such that 50% of the total bases are in contigs of equal or greater size

E.g.



Assembly Evaluation – N50

What is the N50?



Assembly Length vs. N50

Another informative measure is total length of the assembly

- Most genomes have an expected size prior to running assembly
- Assemblers assume diploid genome

Contig total length less than scaffold total length

- Scaffolds are contigs with runs of N's between the contigs

If you remove smaller contigs -> N50 increases :0)

- Total length decreases i.e. less of the genome sequence in the assembly :0(

Most assemblers will remove contigs less than 100bp or less than the read length

Assembly Evaluation Metrics

N50 just measures the continuity of the assembly

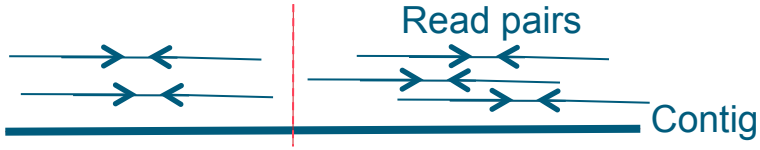
- Larger values are generally better

However it does not assess the quality of the assembled sequence

- E.g. if there are incorrect joins in the assembly the N50 could appear to be larger

Assembly quality measures

- Methods using contigs only:
  - N50
  - Total contig length
  - Number of contigs
- Metrics using an alignment of reads onto the contigs
  - Mapping Fraction (No. reads mapped/total reads) + pairing rate
  - Count the SNPs and indels
  - Misassemblies (regions not spanned by read pairs)



Which human assembly is better? Why?

	Assembly 1	Assembly 2		Assembly 1	Assembly 2
N50	51kb	42Kb		50Kb	20Kb
Total length	2.7Gb	2.69Gb		1.2Gb	2.7Gb
Avg. length	45Kb	39kb		40Kb	18Kb
Mapping rate	0.82	0.78		0.6	0.85
SNP rate	0.02	0.02		0.02	0.02
Indel rate	0.01	0.01		0.01	0.012
Pairing rate	0.8	0.9		0.9	0.88
Misassemblies	15	5		2	2

Which 1Mbp assembly is better? Why?

```
VELVET
======
Evaluation from contigs only:
-----------------------------
Number of contigs: 352
     Total length: 977050
       Max length: 27735
       Min length: 115
   Average length: 2775.71
              N50: 4721

Evaluation from read->contig mapping:
-------------------------------------
         Reads mapped: 325500 (25.32x)
          Reads total: 368472 (32.81x)
         Pairs mapped: 144579
          Pairs total: 144579
Read Mapping fraction: 0.88
Pair Mapping fraction: 0.78

    Read Length Total: 25317884
  Read Mismatch Total: 409353
     Read Indel Total: 23136

Evaluation from contig->reference mapping:
------------------------------------------
  Reference length: 1000000
Reference coverage: 87.506
       Indel count: 47
       Equiv count: 877022
    Mismatch count: 530
```

```
COLUMBUS
========
Evaluation from contigs only:
-----------------------------
Number of contigs: 189
     Total length: 996833
       Max length: 35370
       Min length: 111
   Average length: 5274.25
              N50: 9216

Evaluation from read->contig mapping:
-------------------------------------
         Reads mapped: 330187 (25.89x)
          Reads total: 368472 (32.81x)
         Pairs mapped: 147739
          Pairs total: 147739
Read Mapping fraction: 0.90
Pair Mapping fraction: 0.80

    Read Length Total: 25891260
  Read Mismatch Total: 393852
     Read Indel Total: 22509

Evaluation from contig->reference mapping:
------------------------------------------
  Reference length: 1000000
Reference coverage: 91.4934
       Indel count: 20
       Equiv count: 916483
    Mismatch count: 137
```

Recommended Reading

Li et al (2010) De novo assembly of human genomes with massively parallel short read sequencing, Genome Research, 20: 265-272

Velvet: algorithms for de novo short read assembly using de Bruijn graphs. D.R. Zerbino and E. Birney. Genome Research 18:821-829

Michael C Schatz, Arthur L Delcher and Steven L. Salzberg (2010) Assembly of large genomes using second-generation sequencing, 20(9):1165-73

Isheng J Tsai, Thomas D Otto and Matthew Berriman (2010) Improving draft assemblies by iterative mapping and assembly of short reads to eliminate gaps, Genome Biology, 11:R41

Tutorial 1: Overview, Applications, QC and Formats

- Overview
- Quality Control
- Next-gen Data Formats
- Short Read Alignment
- Sequence Assembly
- **Case Study: 1000 Genomes**
- Experimental Design

Case Study: 1000 Genomes Project Workflow

"Sequencing the genomes of at least a thousand people from around the world to create the most detailed and medically useful picture to date of human genetic variation"

International multi-institute project

- Sanger, Broad, WashU, MPIMG, BGI, BCM, Illumina, Roche, AB

Data co-ordination centre (DCC)

- Joint NCBI/EBI with the sequence archives (SRA/ERA)

1000 genomes pilot

- Pilot 1: 179 individuals (>2x per individual)
  - 4 populations
- Pilot 2: 2 high depth trios (40-60x per individual)
- Pilot 3: 697 individuals via exon pulldown sequencing
  - 7 populations

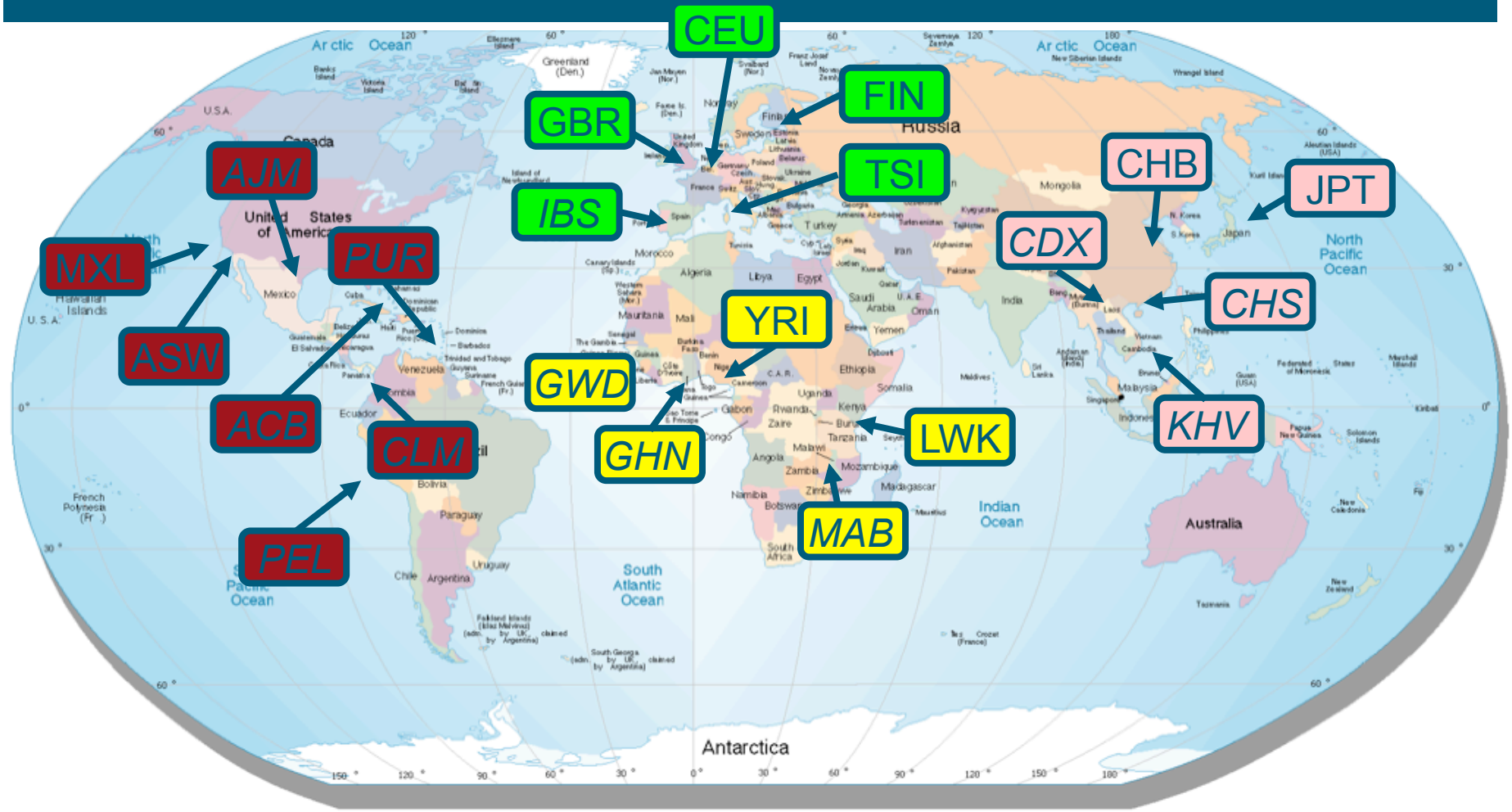
~4.8Tb data from ~12,000 lanes in ~26,000 fastq files

Main project

- ~1200 individuals at ~4x coverage
- 25 distinct populations

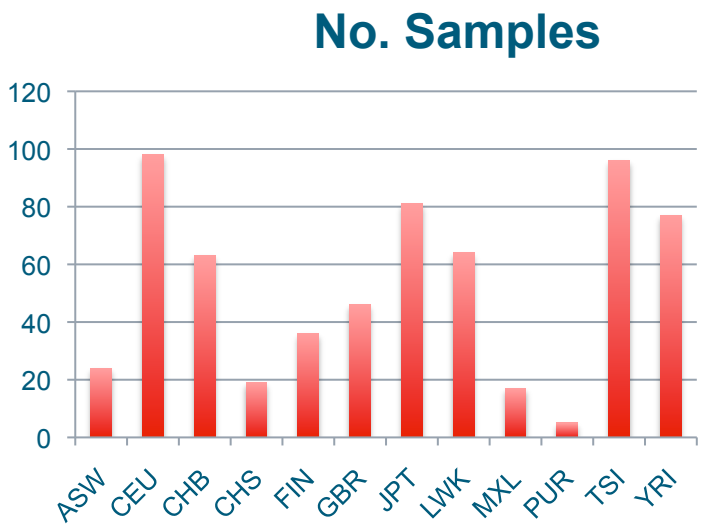
Dealing with the data….the early days of the 1000 genomes project

Main project: ~1200 samples at 4x in 2009/10



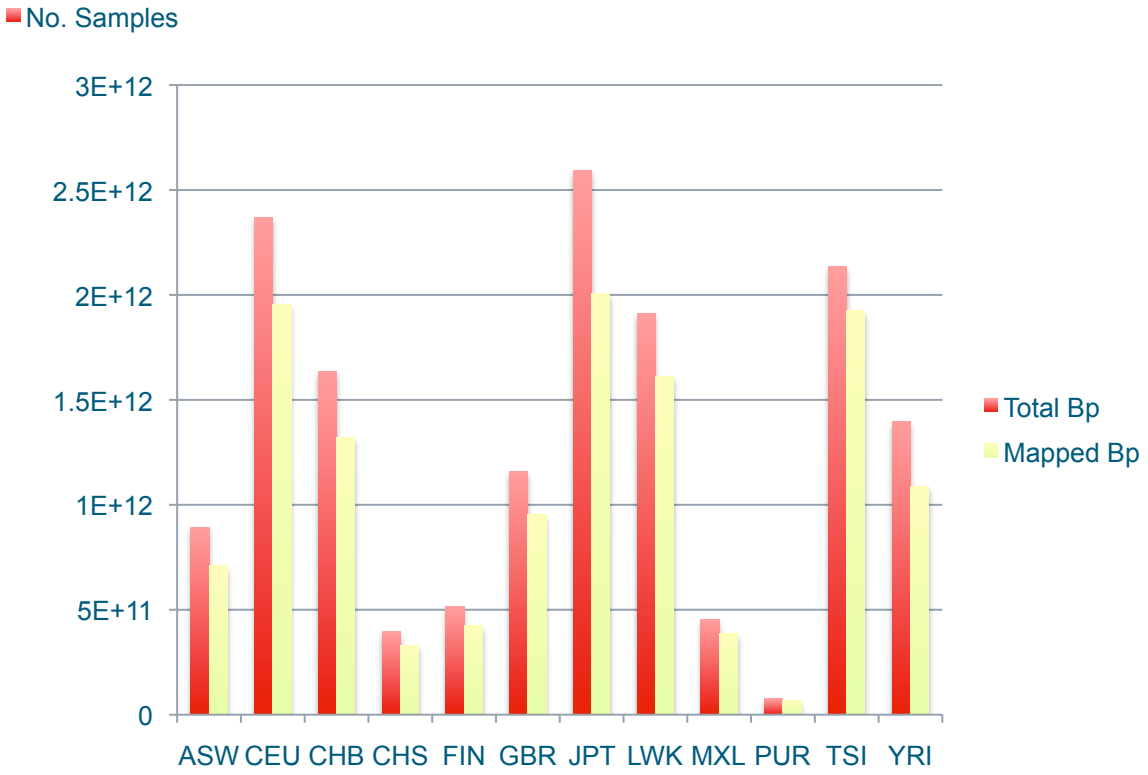
Major population groups comprised of subpopulations of ~100 each

August 2010 Release

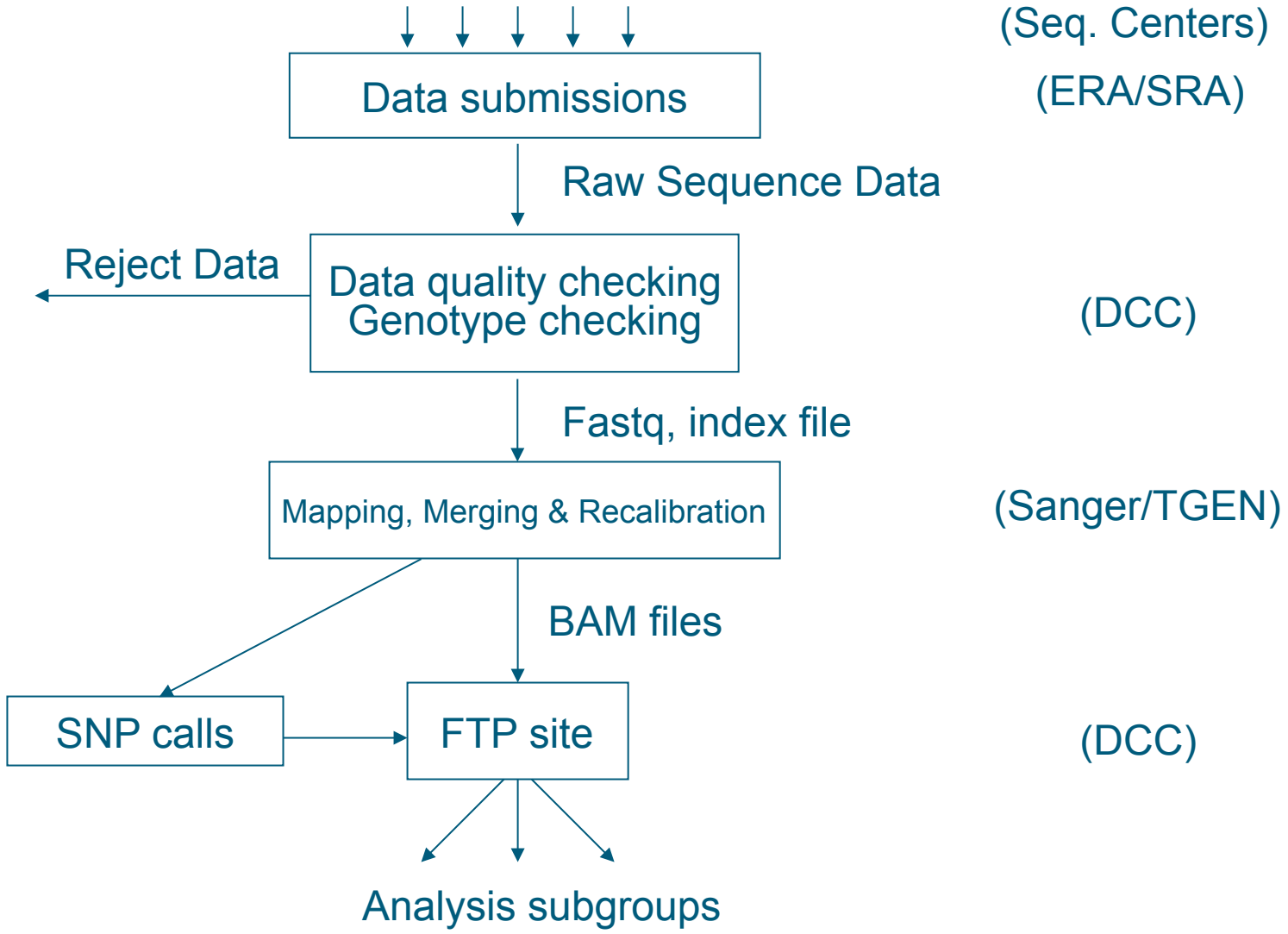


Total Samples: 627

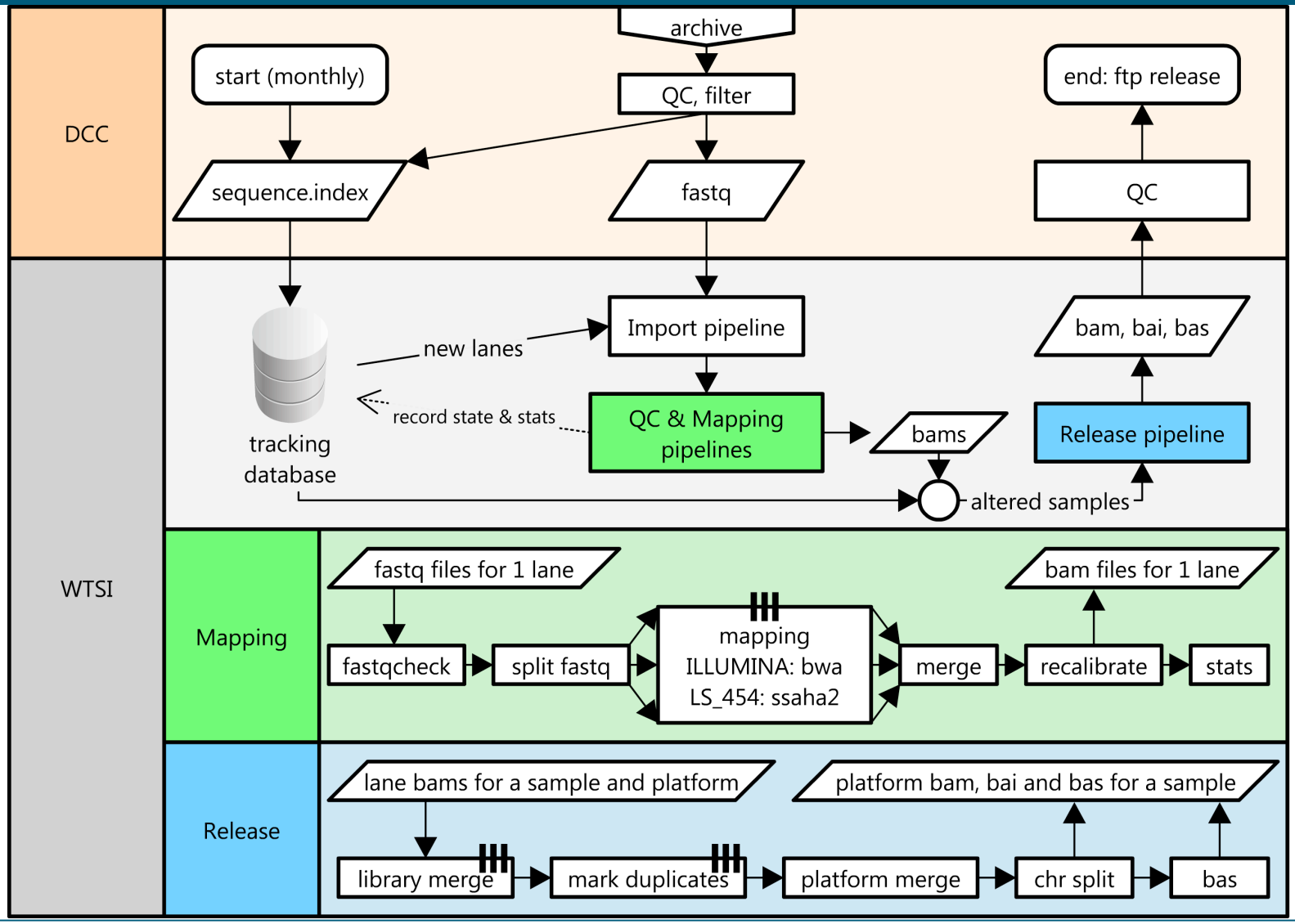
Total Bases: 15.6Tbp



1000 Genomes Pipeline



Sanger Pipeline

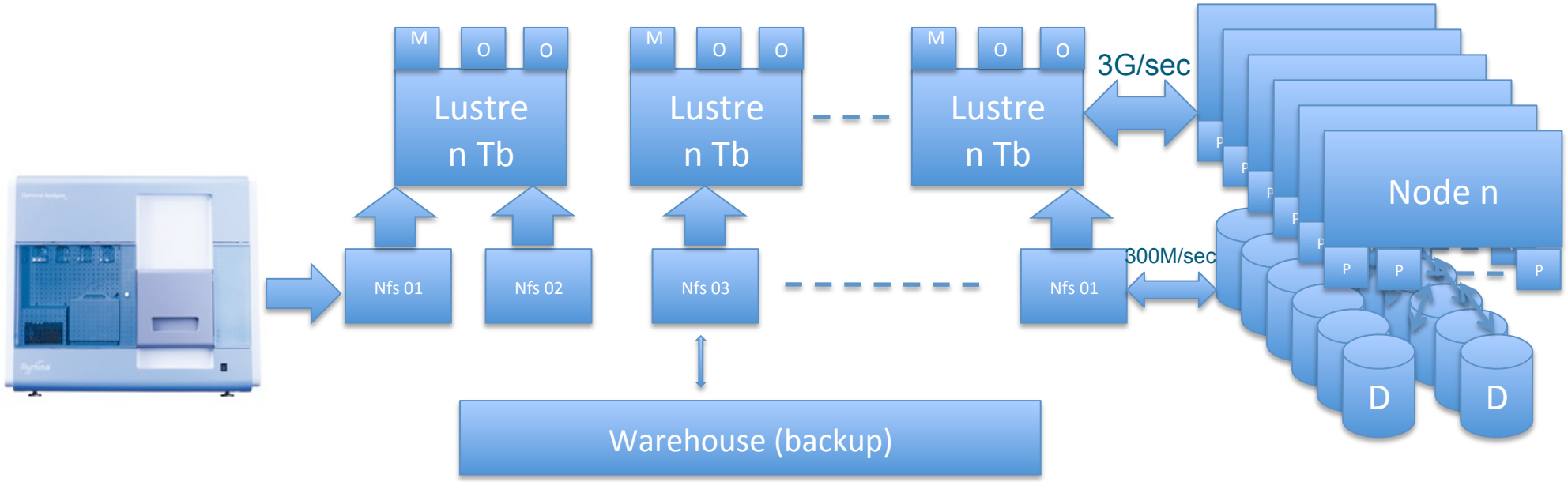


Vertebrate Resequencing Informatics Storage

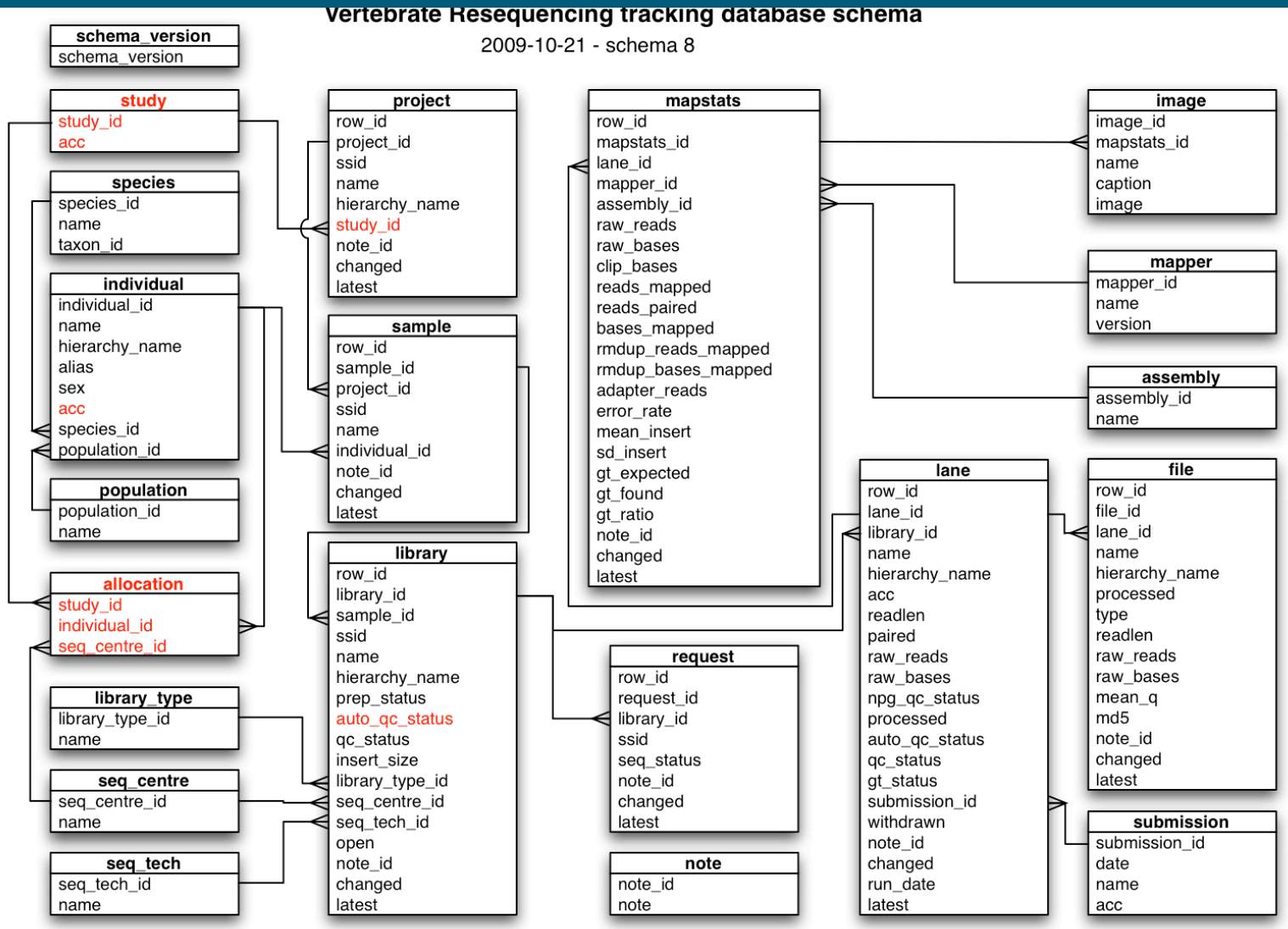
1.5Pbytes of total storage

3 Tiers of storage

- High-performance disk (33%)
  - BAM releases, current variant calls (VCF)
- Medium performance nfs disk (66%)
  - Lane BAMs
- Low performance backup disk (33%)
  - Backup of individual BAMs, old variant call releases (VCF)



Vertebrate Resequencing Tracking



Tutorial 1: Overview, Applications, QC and Formats

- Overview
- Quality Control
- Next-gen Data Formats
- Short Read Alignment
- Sequence Assembly
- Case Study: 1000 Genomes
- Experimental Design

Experimental Design

Choosing right sequencing technology to get optimal results for experiment

Experiment 1: *"I want to determine the genome of a new fungi species with no closely related reference genome"*

- Whole-genome sequencing
- De novo assembly with no reference
  - Longer reads might be more useful – 454?
- Mixture of fragment sizes
  - 200, 500, 3kb, 5kb, 10kb
  - Short range pairing information and long range information for scaffolding

Experiment 2: *"I want to measure the relative expression level differences of one yeast species under different environmental conditions"*

- Sequence the transcriptome (RNA-seq)
- Illumina or SOLiD sequencing for high depth
  - Multiplex the sequencing into a singlle lane
- Measure the relative expression levels by aligning
  - e.g. use Cufflinks to detect differential expression across the samples

Experimental Design

Experiment 4: *"I want to catalog all of the structual variants in a human cancer cell vs. the normal cell for as little cost as possible"*

- Fragment coverage vs. sequence coverage
- SVs are called from discordant read pairs – long range information
  - Sequence coverage not important
  - Require fragment coverage
- Sequence multiple paired libraries with short read length
  - E.g. 1000bp in total capacity
  - 100bp reads = 5RPs = 5 fragments x 500bp per fragment = 2.5Kbp fragment coverage
  - 40bp reads = 25RPs = 25 fragments x 500bp per fragment = 12.5Kbp fragment coverage
  - More fragments sequenced = more independent sources of evidence to call structural variants

Experiment 5: "*I have 3 patients with a rare condition and want to find the causitive variant*"

- High depth sequencing (20x?) per patient. Illumina or SOLiD or complete genomics
- Exome sequencing – 1 lane per patient
- SNPs + short indels
- Exclude all common variation (dbSNP + 1000Genomes)
- Is there a shared truncating variant? If not – is there a shared truncating structural variant?

Q&A

Questions from you?