Mapping

Isheng Jason Tsai

Introduction to NGS Data and Analysis
Lecture 3 v2020

BRCAS
中央研究院生物多樣性研究中心

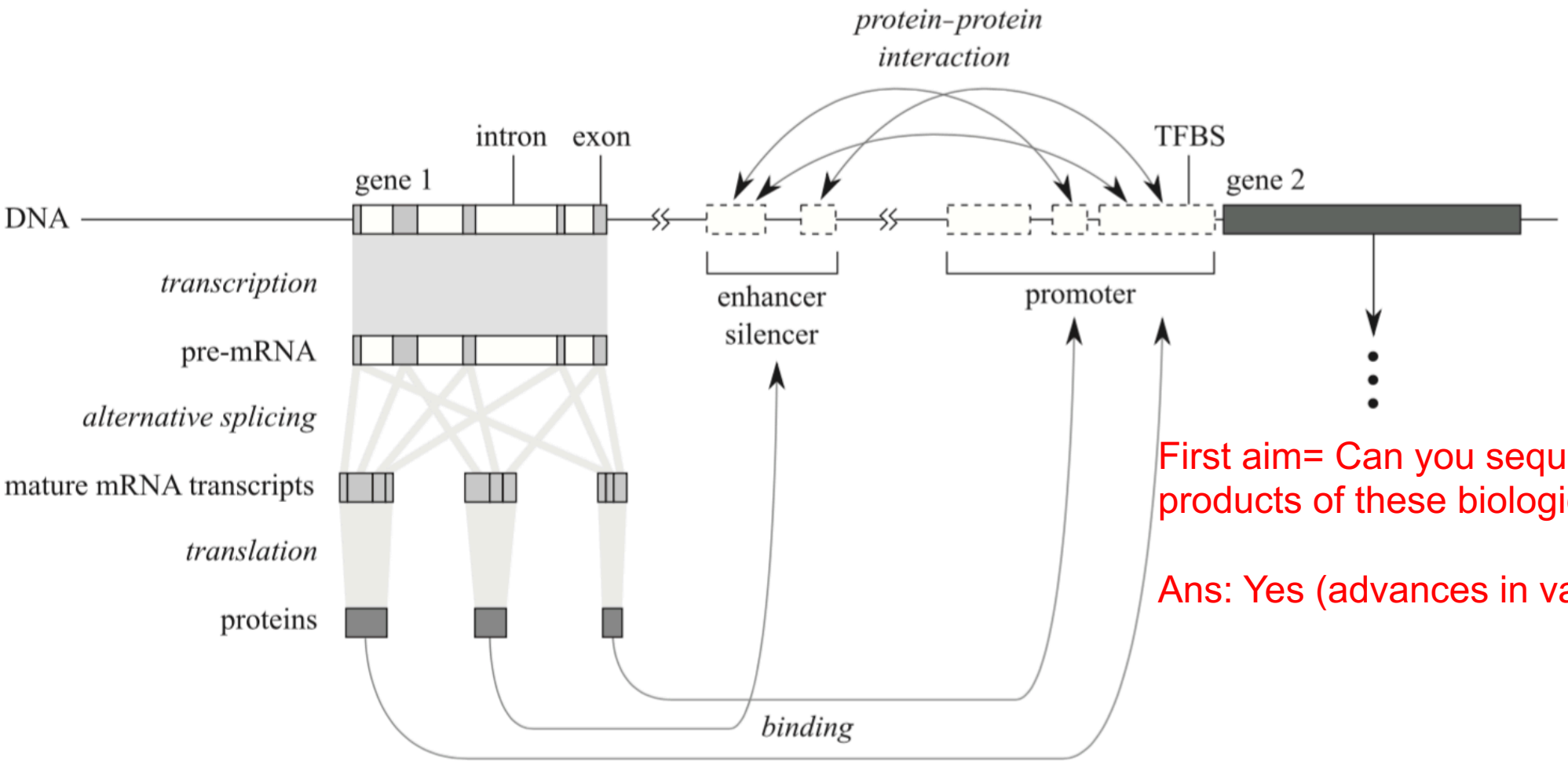
中央研究院
ACADEMIA SINICA
1928

Lecture outline

1. Background
2. Mapping algorithms
3. Mapping processes
4. Variant calling

Background

High throughput sequencing applications



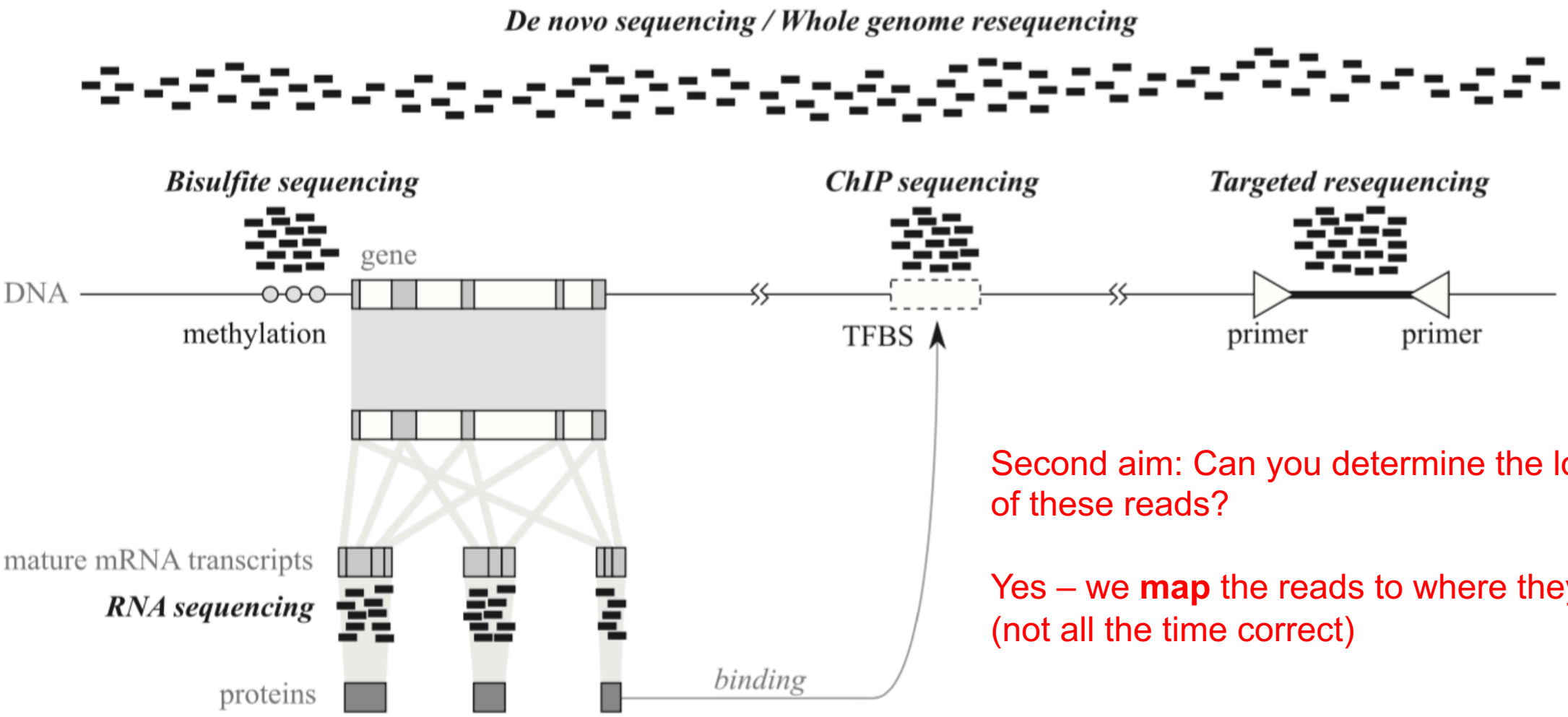
First aim= Can you sequence the products of these biological processes?

Ans: Yes (advances in various experiments)

Figure 1.1 A schematic illustration of the central dogma. Gene 1 has three alternatively spliced transcripts. The relative expression of such transcripts affects the regulatory modules of gene 2, and eventually its expression. Definitions are given in Section 1.1.

Makinen *et al*

High throughput sequencing applications



Second aim: Can you determine the location of these reads?

Yes – we **map** the reads to where they belong (not all the time correct)

Figure 1.2 A schematic summary of high-throughput sequencing applications. Details are described in Section 1.3.

Makinen *et al*

De novo vs mapping approach

Mapping is less complicated and more intuitive

Can gather lots of information from many individuals given a good reference

But, information on repeats/ gene families / *de novo* genes / large structural variants are more difficult to detect

Assembly is powerful but also computationally demanding

And is your question worth the trouble to assemble 100 strains?

In practice, people do a combination of both approaches

In humans, *de novo* genomes of references and cancer cells are being generated. In butterflies, many assemblies to reveal super gene

Recommended paper

Technology dictates algorithms: Recent developments in read alignment

Mohammed Alser, Jeremy Rotman, Kodi Taraszka, Huwenbo Shi, Pelin Icer Baykal, Harry Taegyun Yang, Victor Xue, Sergey Knyazev, Benjamin D. Singer, Brunilda Balliu, David Koslicki, Pavel Skums, Alex Zelikovsky, Can Alkan, Onur Mutlu, Serghei Mangul

(Submitted on 28 Feb 2020)

”…Our review provides a survey of **algorithmic foundations and methodologies across alignment methods for both short and long reads**. We provide rigorous experimental evaluation of 11 read aligners to demonstrate the effect of these underlying algorithms on speed and efficiency of read aligners. We separately **discuss how longer read lengths produce unique advantages and limitations to read alignment techniques**. We also **discuss how general alignment algorithms have been tailored to the specific needs of various domains in biology, including whole transcriptome, adaptive immune repertoire, and human microbiome studies**. “

https://arxiv.org/ftp/arxiv/papers/2003/2003.00110.pdf

Note: **a preprint**

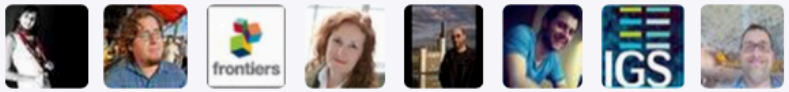
Preface

Nick Loman @pathogenomenick · Mar 11

Got a talk at ECCMID entitled: "So you've sequenced your (bug) genome ... what now?" Crowdsourcing best answers please, will acknowledge!

RETWEETS 7

FAVORITES 2



11:56 AM - 11 Mar 2015 · Details

https://twitter.com/pathogenomenick/status/575626319616176128

We all know…

Alan McNally @alanmcn1 · Mar 11
@pathogenomenick @biomickwatson in that case "give it to someone who knows what they are doing!"

1 1

Nicki Fawcett @DrNJFawcett · Mar 11
@alanmcn1 @pathogenomenick Clinician thirding/fourthing 'Give it to someone who knows what they're doing'. #ooohYersiniaInEverything

1

Mick Watson @BioMickWatson · Mar 11
@pathogenomenick ah. Clinician clinicians? Give the data to someone who knows what to do with it, then ;-)

2

azizipeasie @AzizAboobaker · Mar 11
@pathogenomenick send it to your bioinformation friend and give them a week to send back a paper with themselves as a middle author.

3 6

https://twitter.com/pathogenomenick/status/575626319616176128

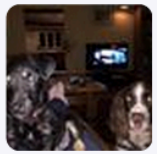
Logical answer

azizipeasie @AzizAboobaker · Mar 11
@pathogenomenick sequence some more while your thinking.

1

Esther Robinson @ilovechocagar · Mar 11
@pathogenomenick first law of doing a lab test: don't unless you know what your question is

4 2

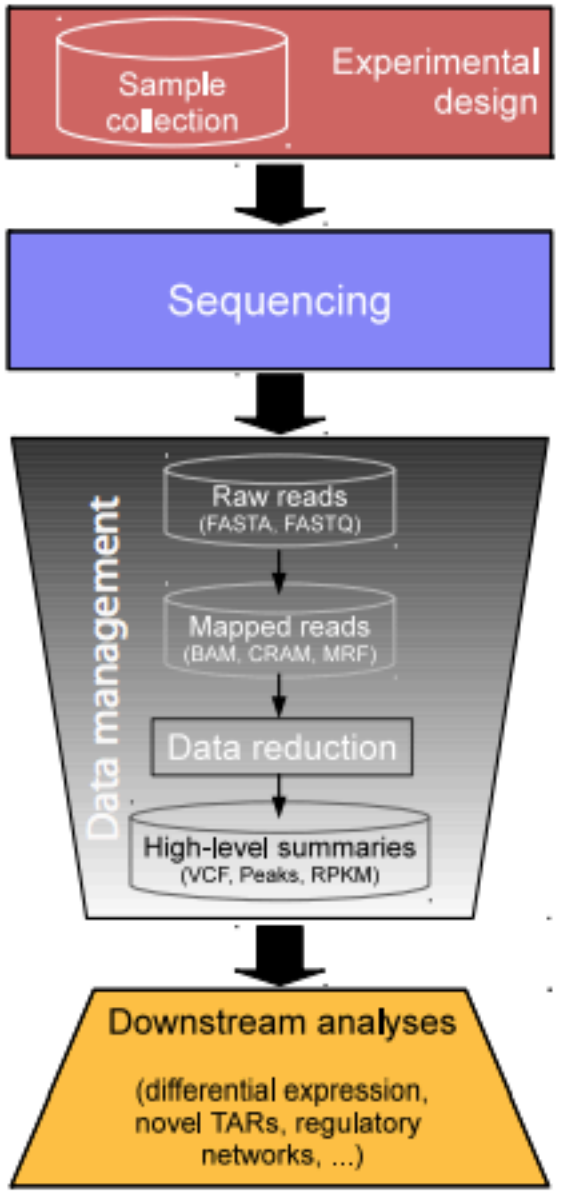


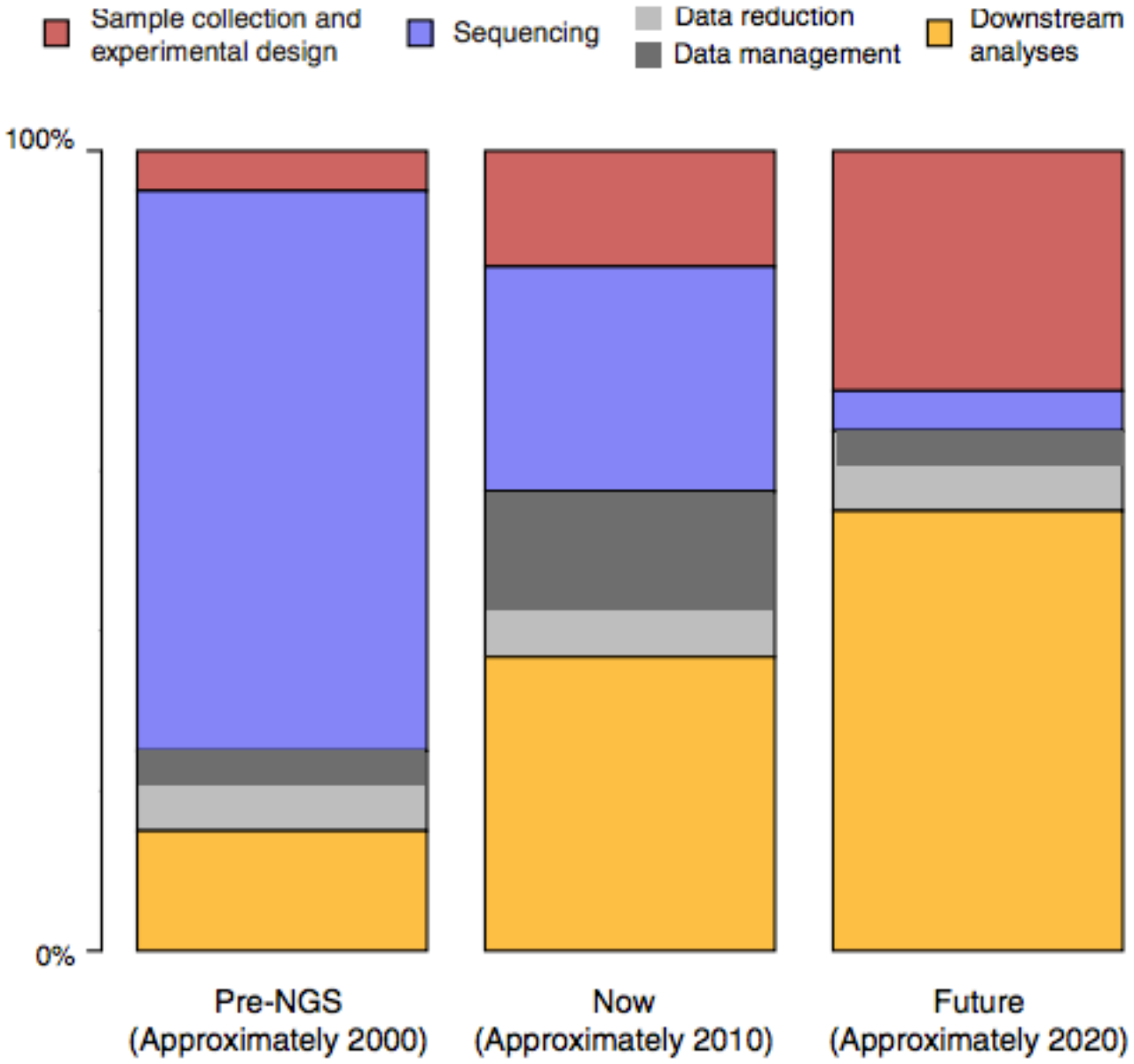
ruth massey @bowsermassey · Mar 11
@WvSchaik @pathogenomenick determine ID, resistance profile and dare I say it....virulence potential!

Bill Hanage @BillHanage · Mar 11
@pathogenomenick you've had many good suggestions but it completely depends on what you are interested in. Resistance? Epi? Something else?

https://twitter.com/pathogenomenick/status/575626319616176128

The real cost of sequencing

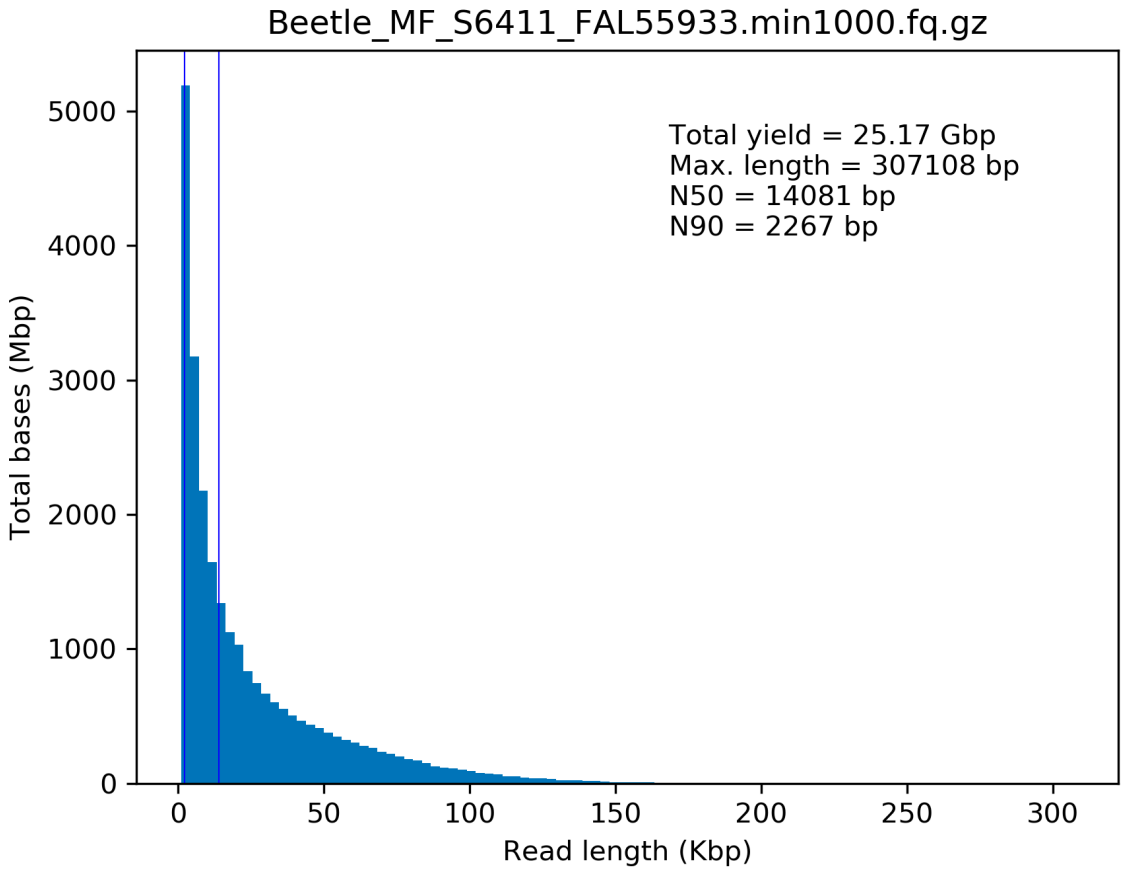




Long reads are now common

- Most users should and will be mainly analyzing Paired-end Illumina reads (typically 150 bases)

- Pacific Biosciences or Oxford Nanopore (long reads) are increasingly very common



Mapping

Mapping is **aligning** the read to where **the most likely origin** within the reference/assembly

Sequence alignment has not changed and will remain a classic problem
Tradeoffs of speed, accuracy and sensitivity

Sequence data we want to map:
- Mostly nucleotide
Very short evolutionary distances (human to reference, isolate/strain to reference, 'slightly diverged' strain will result in less mapped reads)
A lot of reads– needs faster processing per read (BLAST is too slow!)

There are some assumptions to make alignment process faster
(like allows most 2 mismatches)

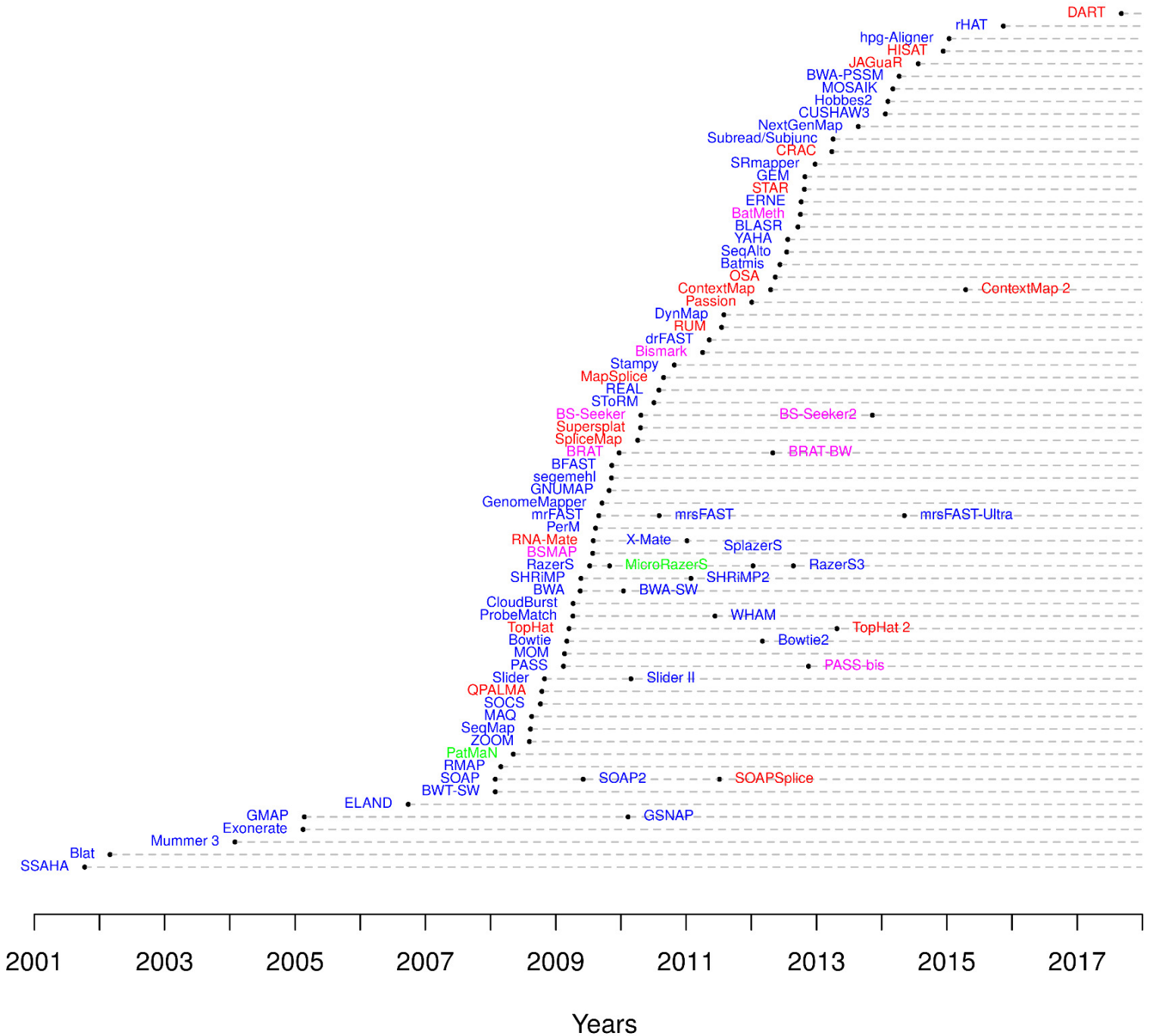
How to map billions of short reads onto genomes

Cole Trapnell & Steven L Salzberg

Mapping the vast quantities of short sequence fragments produced by next-generation sequencing platforms is a challenge. What programs are available and how do they work?

https://www.nature.com/articles/nbt0509-455.pdf

All the mappers!



http://www.ebi.ac.uk/~nf/hts_mappers/
(link no longer working; last updated 2018)

Mapper	Data	Availability	Version	O.S.	Number Citations	Seq.Plat.	Input	Output
BWA	DNA	OS	0.6.2	Linux,Mac,Windows	**13341**	I,So,4,Sa,P	FASTA/Q	SAM
Bowtie	DNA	OS	0.12.7	Linux,Mac,Windows	**11207**	I,So,4,Sa,P	(C)FAST(A/Q)	SAM TSV
Bowtie2	DNA	OS	2.0beta5	Linux,Mac,Windows	**8586**	I,4,Ion	FASTA/Q	SAM TSV
Blat	DNA	OS	34	Linux,Mac	**6252**	N	FASTA	TSV BLAST
TopHat	RNA	OS	1.4.1	Linux,Mac	**3764**	I	FASTA/Q GFF	BAM
BWA-SW	DNA	OS	0.6.2	Linux,Mac,Windows	**3494**	I,4,Sa,Hel,Ion,P	FASTA/Q	SAM
MAQ	DNA	OS	0.7.1	Linux,Mac	**2592**	I,So	(C)FAST(A/Q)	TSV
Mummer 3	DNA	OS	3.23	Linux,Mac	**2446**	N	FASTA	TSV
SOAP2	DNA	OS	2.21	Linux	**1655**	I	FASTA/Q	SAM TSV
SOAP	DNA	OS	1.11	Linux,Mac	**1284**	I	FASTA/Q	TSV
GSNAP	DNA	OS	2012-04-27	Linux,Unix,Mac,Windows	**1156**	I,4,Sa,Hel,Ion,P	FASTA/Q	SAM Native
TopHat 2	RNA	OS	2.0.8	Linux,Mac	**1102**	I	FASTA/Q	BAM
Exonerate	DNA	OS	2.2	Linux,Mac	**918**	N	FASTA	TSV
Bismark	Bisulfite	OS	0.7.3	Linux,Mac	**887**	I	FASTA/Q	SAM
SSAHA2	DNA	Bin	2.5.5	Linux,Mac	**874**	I,4,Sa	FASTA/Q	SAM
SSAHA	DNA	OS	3.1	Linux,Mac	**874**	N	FASTA/Q	TSV
GMAP	DNA	OS	2012-04-27	Linux,Unix,Mac,Windows	**868**	I,4,Sa,Hel,Ion,P	FASTA/Q	SAM GFF Native
CloudBurst	DNA	OS	1.1	Linux,Mac,Windows	**650**	N	FASTA	TSV
MapSplice	RNA	OS	1.15.2	Linux	**610**	I	FASTA/Q	SAM BED
STAR	RNA	OS	2.3.0	Linux,Unix,Mac	**602**	I,4,Sa,Ion,P	FASTA/Q	SAM
mrFAST	DNA	OS	2.5.0.1	Linux,Unix	**602**	I	FASTA/Q	SAM DIVET
SHRiMP	DNA	OS	1.3.2	Linux,Mac	**573**	I,So,4,Hel	(C)FAST(A/Q)	TSV
BFAST	DNA	OS	0.7.0	Linux,Mac	**553**	I,So,4, Hel	(C)FAST(A/Q)	SAM TSV
HISAT	RNA	OS	1	Windows, Linux, Unix, Mac	**480**	I	FASTA/Q	SAM

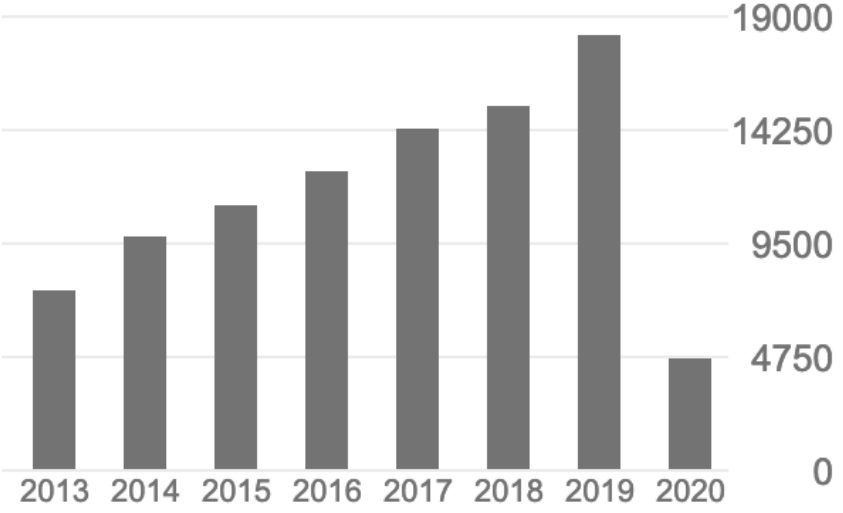
Heng Li has contributed a dozen of those mappers

Heng Li

Dana-Farber Cancer Institute & Harvard University
在 jimmy.harvard.edu 的電子郵件地址已通過驗證 - 首頁

Computational Biology Bioinformatics Genomics

	全部	自 2015 年
引文	108019	76196
H 指數	53	50
i10 指數	68	62



Author of maq, bwa, bwa-mem, wtdbg2 and minimap2…

http://www.liheng.org/

How?

Brute force comparison
Smith-Waterman
Suffix Tree
Burrows-Wheeler Transform

Brute force

```
TCGATCC
       ↘ ?
GACCTCATCGATCCCACTG
```

1.

```
TCGATCC
x
GACCTCATCGATCCCACTG
```

2.

```
 TCGATCC
 x
GACCTCATCGATCCCACTG
```

3.

```
     TCGATCC
     | | x
GACCTCATCGATCCCACTG
```

4.

```
        TCGATCC
        |||||||
GACCTCATCGATCCCACTG
```

Credit: Mike Zody

Exact matching

What's a simple algorithm for exact matching?

P: word

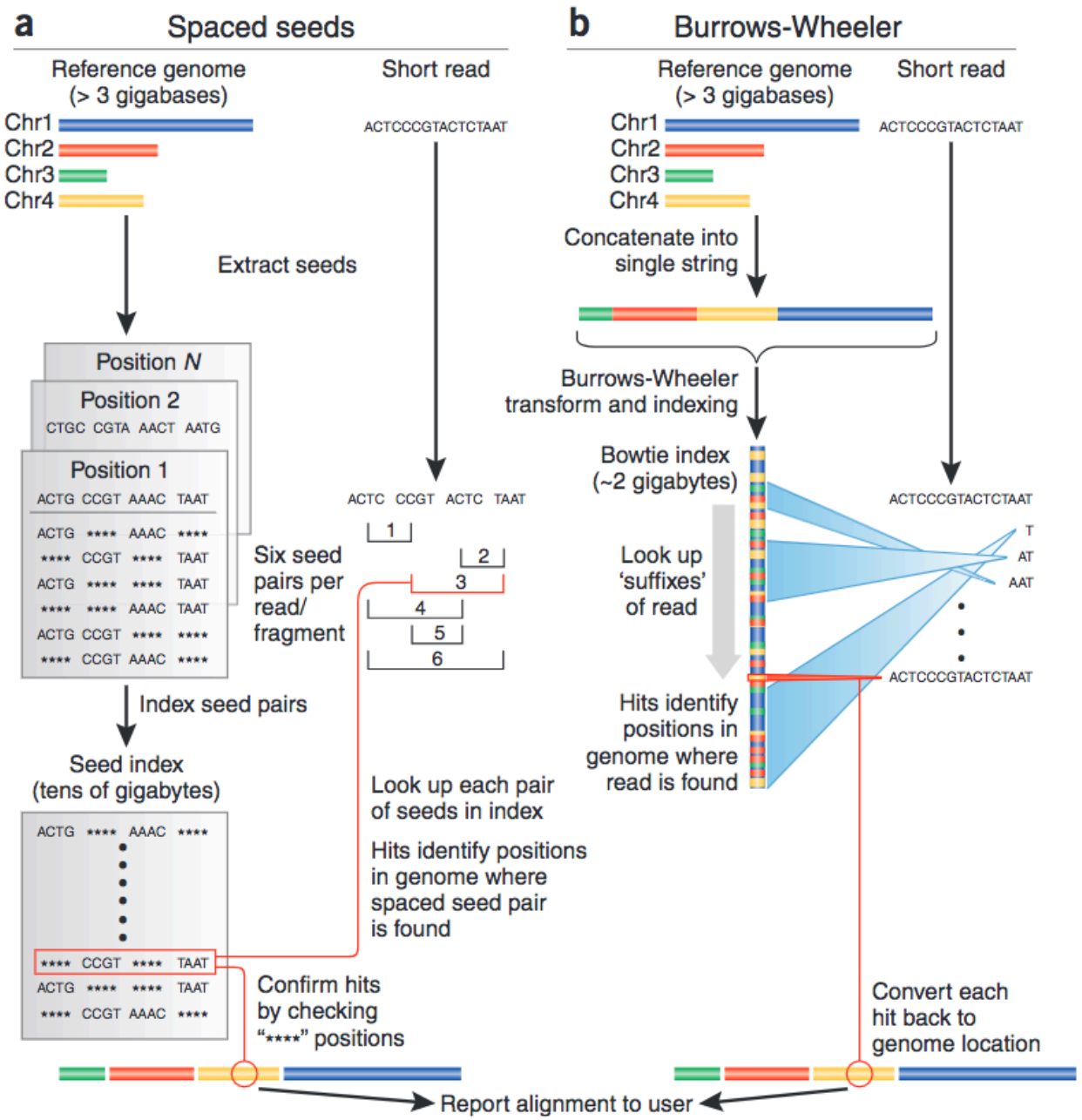
T: There would have been a time for such a word

```
word word word word word word word word word word
 word word word word word word word word word
  word word word word word word word word word
   word word word word word word word word word
    word word word word word word word word word
```

One occurrence

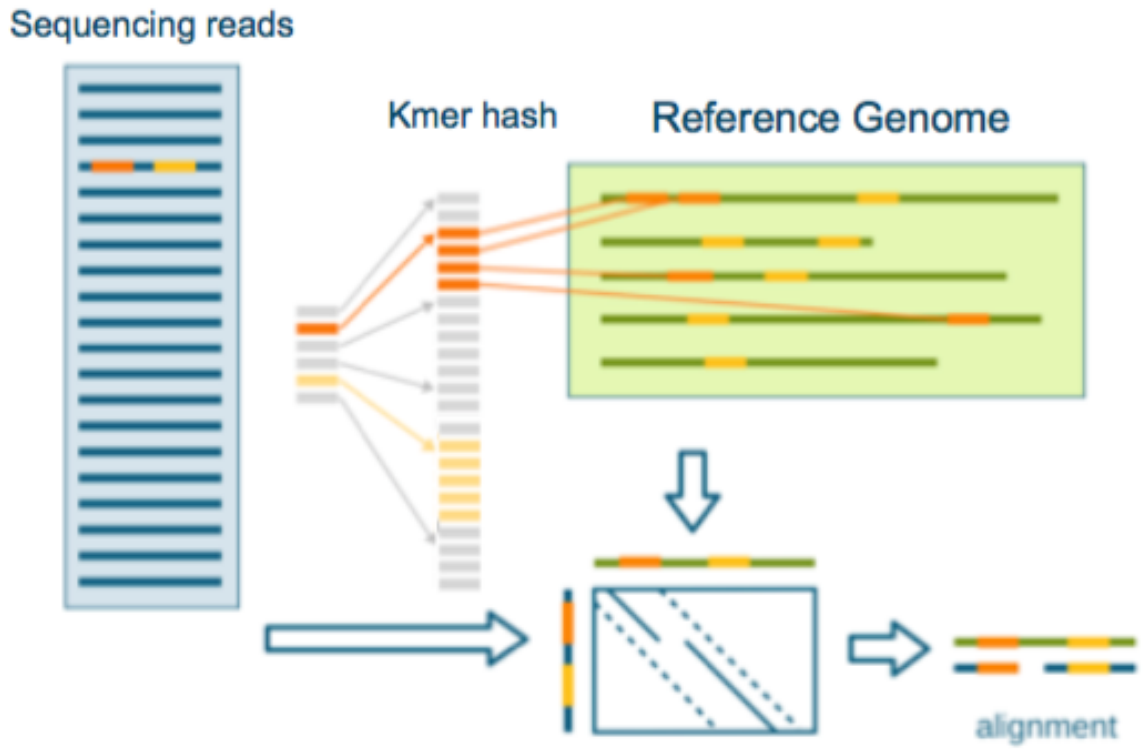
Try all possible alignments. For each, check whether it's an occurrence. "Naïve algorithm."

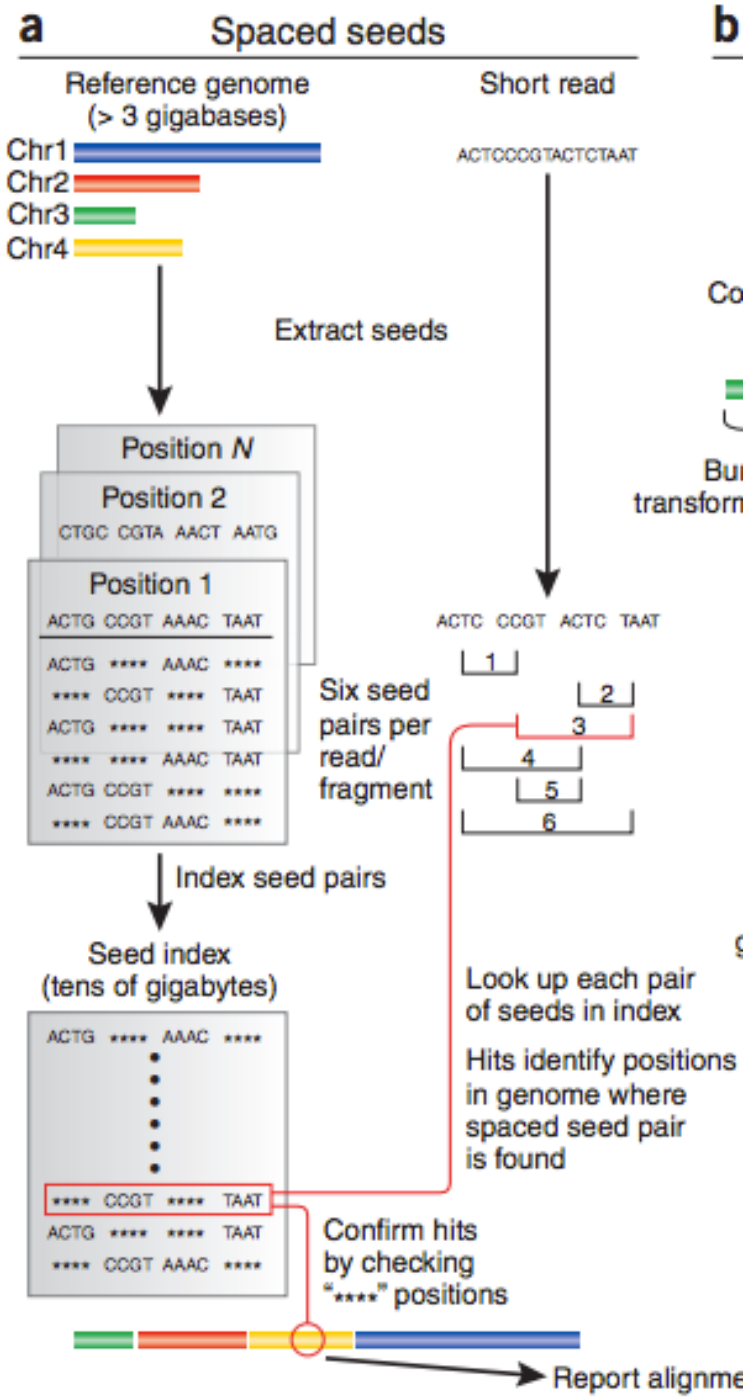
Ben Langmead



https://www.nature.com/articles/nbt0509-455.pdf

Mapping (hash table)





Identify all the seeds in the index
Determine the most likely location
Perform Smith-Waterman alignment to fully align
Output (important)

Example: BLAST, MAQ (Heng Li 2008)

Suffix tree

GACCTCATCGATCCCACTG

A				C								G			T			
C		T		A			C		G	T		A		$		C		G
C	T	C		C	T	A	C	T	A	C	G	C	T		A	C	G	$
T	G	C	G	T	C	C	A	C	T	A	$	C	C		T	C	A	
C	$	C	A	G	G	T	C	A	C	T		T	C		C	A	T	
A		A	T	$	A	G	T	T	C	C		C	C		G	C	C	
T		C	C		T	$	G	C	C	G		A	A		A	T	C	
C		T	C		C		$	G	A	A		T	C		T	G	C	
G		G	C		C			A	C	T		C	T		C	$	A	
A		$	A		C			T	T	C		G	G		C		C	
T			C		A			C	G	C		A	$		C		T	
C			T		C			C	$	C		T			A		G	
C			G		T			C		A		C			C		$	
C			$		G			A		C		C			T			
A					$			C		T		C			G			
C								T		G		A			$			
T								G		$		C						
G								$				T						
$												G						
												$						

But suffix can be very very big if data structure not considered carefully!

Credit: Mike Zody ; Trapnell *et al* (2009)

Burrows-Wheeler Transform

A transformation that will result in many repeated characters

This means it's easy to compress

And an elegant way to search!

Transformation				
Input	All Rotations	Sorting All Rows into Lex Order	Taking Last Column	Output Last Column
^BANANA|	^BANANA| |^BANANA A|^BANAN NA|^BANA ANA|^BAN NANA|^BA ANANA|^B BANANA|^	**A**NANA|^B **A**NA|^BAN **A**|^BANAN **B**ANANA|^ **N**ANA|^BA **N**A|^BANA ^BANANA| |^BANANA	ANANA|^**B** ANA|^BA**N** A|^BANA**N** BANANA|^ NANA|^B**A** NA|^BAN**A** ^BANANA| |^BANAN**A**	BNN^AA|A

Wiki

Original sequence	All permutations	Alphabetical ordering of rows	Output of last column
>BONOBO*	>BONOBO*	BONOBO*>	>
	*>BONOBO	BO*>BONO	O
	O*>BONOB	NOBO*>BO	O
	BO*>BONO	OBO*>BON	N
	OBO*>BON	ONOBO*>B	B
	NOBO*>BO	O*>BONOB	B
	ONOBO*>B	>BONOBO*	*
	BONOBO*>	*>BONOBO	O

https://link.springer.com/chapter/10.1007/978-3-319-54064-1_6

Inverse transformation using Burrows-Wheeler transform			
Add cycle 1	Sort cycle 1	Add cycle 2	Sort cycle 2
>	B	>B	BO
O	B	OB	BO
O	N	ON	NO
N	O	NO	OB
B	O	BO	ON
B	O	BO	O*
*	>	*>	>B
O	*	O*	*>
Add cycle 3	Sort cycle 3	Add cycle 4	Sort cycle 4
>BO	BON	>BON	BONO
OBO	BO*	OBO*	BO*>
ONO	NOB	ONOB	NOBO
NOB	OBO	NOBO	OBO*
BON	ONO	BONO	ONOB
BO*	O*>	BO*>	O*>B
*>B	>BO	*>BO	>BON
O*>	*>B	O*>B	*>BO

https://link.springer.com/chapter/10.1007/978-3-319-54064-1_6

GACCTCATCGATCCCACTG$
ACCTCATCGATCCCACTG$G
CCTCATCGATCCCACTG$GA
CTCATCGATCCCACTG$GAC
TCATCGATCCCACTG$GACC
CATCGATCCCACTG$GACCT
ATCGATCCCACTG$GACCTC
TCGATCCCACTG$GACCTCA
CGATCCCACTG$GACCTCAT
GATCCCACTG$GACCTCATC
ATCCCACTG$GACCTCATCG
TCCCACTG$GACCTCATCGA
CCCACTG$GACCTCATCGAT
CCACTG$GACCTCATCGATC
CACTG$GACCTCATCGATCC
ACTG$GACCTCATCGATCCC
CTG$GACCTCATCGATCCCA
TG$GACCTCATCGATCCCAC
G$GACCTCATCGATCCCACT
$GACCTCATCGATCCCACTG

Sort

ACCTCATCGATCCCACTG$G
ACTG$GACCTCATCGATCCC
ATCCCACTG$GACCTCATCG
ATCGATCCCACTG$GACCTC
CACTG$GACCTCATCGATCC
CATCGATCCCACTG$GACCT
CCACTG$GACCTCATCGATC
CCCACTG$GACCTCATCGAT
CCTCATCGATCCCACTG$GA
CGATCCCACTG$GACCTCAT
CTCATCGATCCCACTG$GAC
CTG$GACCTCATCGATCCCA
GACCTCATCGATCCCACTG$
GATCCCACTG$GACCTCATC
G$GACCTCATCGATCCCACT
TCATCGATCCCACTG$GACC
TCCCACTG$GACCTCATCGA
TCGATCCCACTG$GACCTCA
TG$GACCTCATCGATCCCAC
$GACCTCATCGATCCCACTG

Credit: Mike Zody

TCGATCC

?

GACCTCATCGATCCCACTG

```
GAC
CAC
GAT
CAT
CCA
TCA
CCC
TCC
ACC
TCG
CCT
ACT
$GA
CGA
TG$
CTC
ATC
ATC
CTG
G$G
```

- Start with the transform column
- My read starts with a T, so I want rows with Ts in them
- This column gives me all the single nucleotide counts
- Sort the single nucleotide counts to get the alphabetically first column
- Now these two columns give me all the dinucleotide counts
- Sort those to get the alphabetically first two columns
- Now there is only one place my read can match

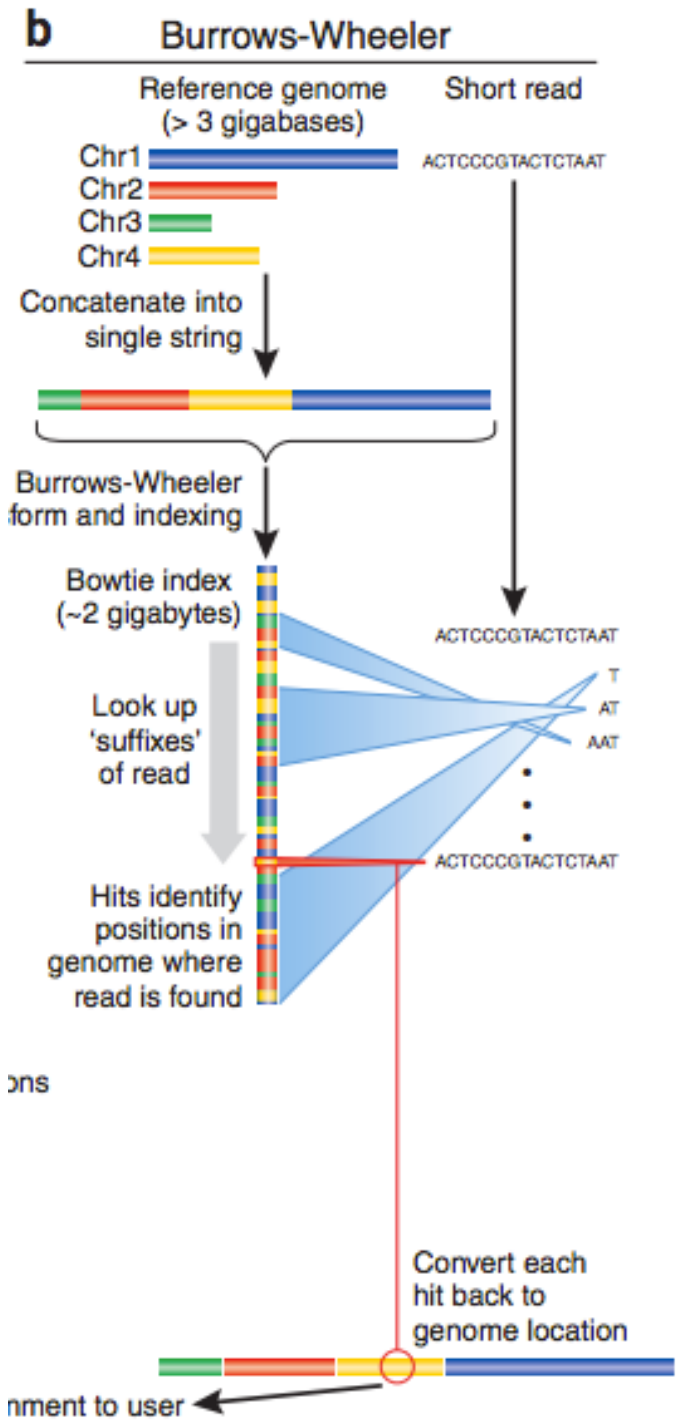
BWT – a summary

Stores all possible suffixes to enable fast string matching

Much smaller memory footprint than hash table
(hash table need to store all different kmers)

Examples:
MUMMER, bwa, bowtie2

Still need local alignment in final step



Trapnell *et al* (2009)

Hash table vs. BWT

BWT

Hash table

Table 1 A selection of short-read analysis software

Program	Website	Open source?	Handles ABI color space?	Maximum read length
Bowtie	http://bowtie.cbcb.umd.edu	Yes	No	None
BWA	http://maq.sourceforge.net/bwa-man.shtml	Yes	Yes	None
Maq	http://maq.sourceforge.net	Yes	Yes	127
~~Mosaik~~	~~http://bioinformatics.bc.edu/marthlab/Mosaik~~	~~No~~	~~Yes~~	~~None~~
~~Novoalign~~	~~http://www.novocraft.com~~	~~No~~	~~No~~	~~None~~
~~SOAP2~~	~~http://soap.genomics.org.cn~~	~~No~~	~~No~~	~~60~~
~~ZOOM~~	~~http://www.bioinfor.com~~	~~No~~	~~Yes~~	~~240~~

Trapnell *et al* (2009)

Hash table vs. BWT strengths and weaknesses

Burrows-Wheeler, e.g. bwa, bowtie
– Fast, esp. (multiple) exact matches
– High sensitivity at repetitive regions
– less robust at high genomic variation (because you need to retry with a substitution)

Hashing (overlapping k-mer words, e.g SMALT, Stampy)
– Slower (more memory hungry)
– Less sensitivity at repetitive regions
– tolerate high genomic variation
– partial alignments (junction reads) easier
– Flexible (multiple sequencing platforms)

Ponstingl and Ning

Choose an mapper/ aligner

Hash based approaches are more suitable for divergent alignments

General rule:

<2% divergence -> BWT

E.g. human samples

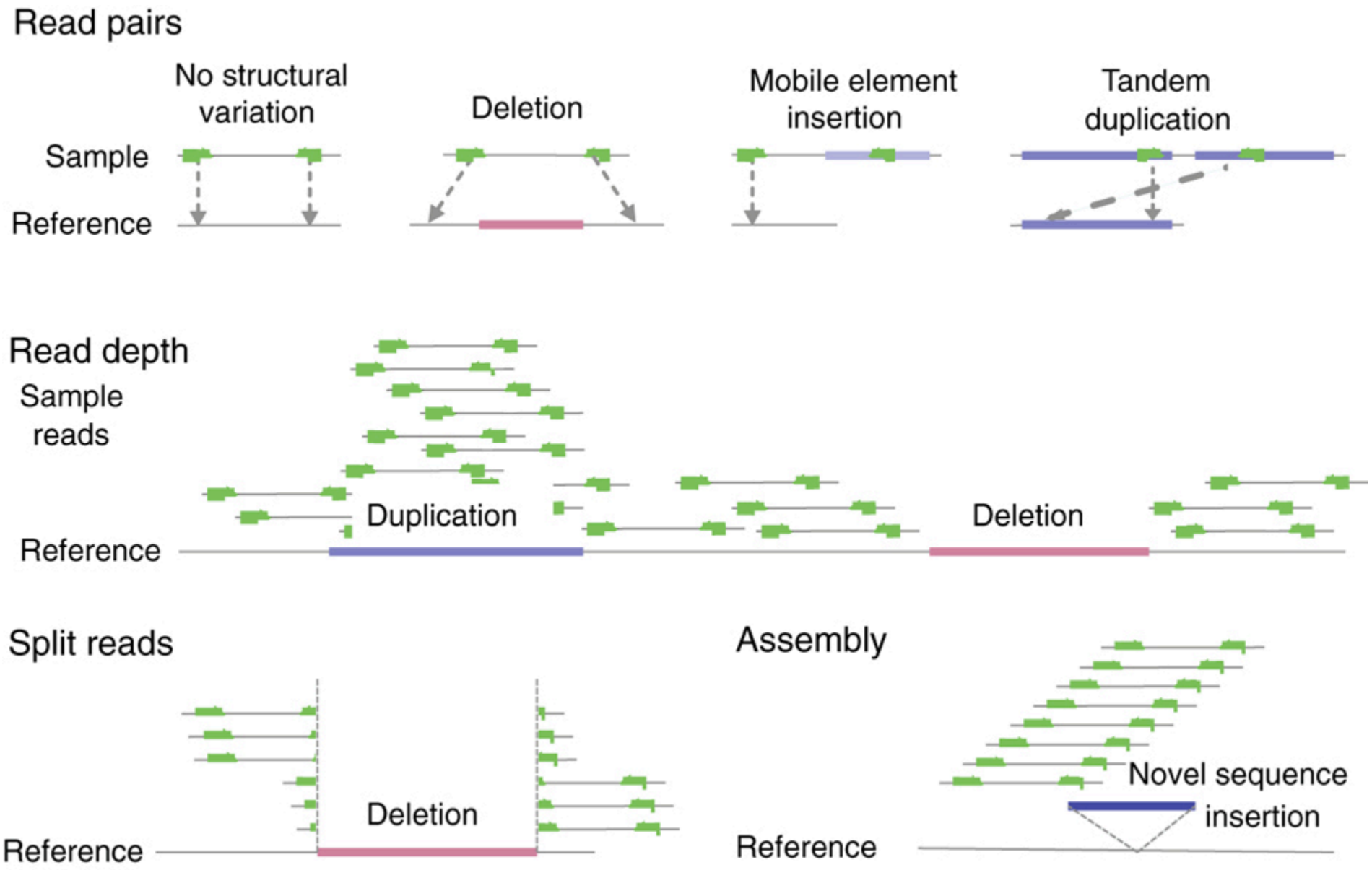
>2% divergence -> hash based approach

E.g. wild sample alignments ;

Watch out for latest advancement ; and don’t stay at one for too long

Credit: Golden Helix inc

Detecting structural variations (ideally assembly is probably better)



Baker (2012)

What to do with repetitive (multi) reads?

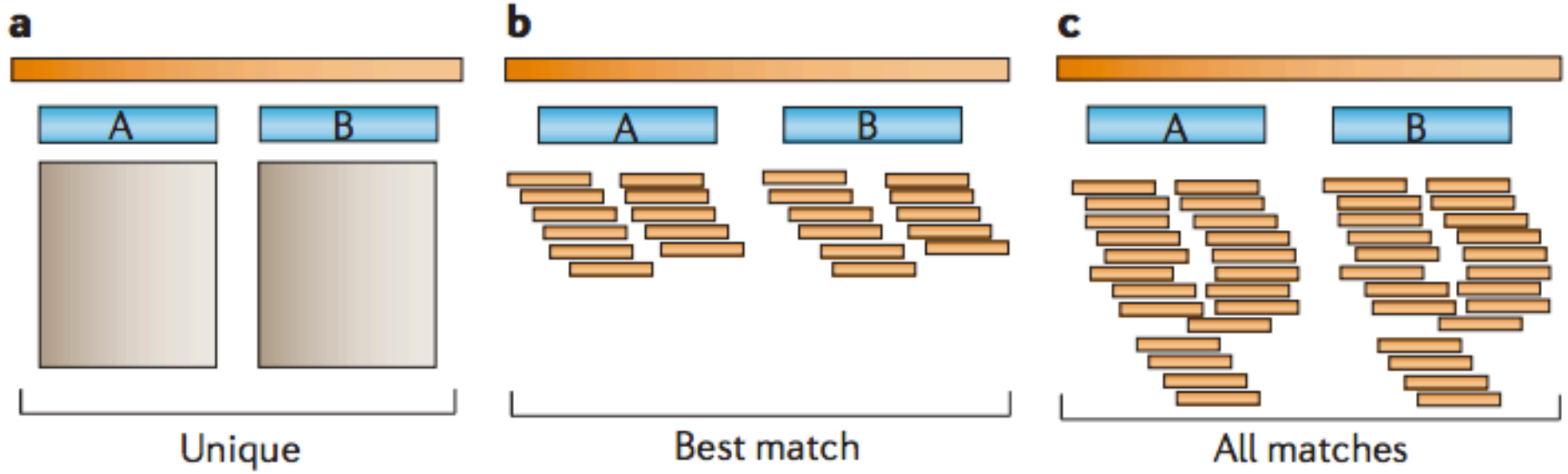
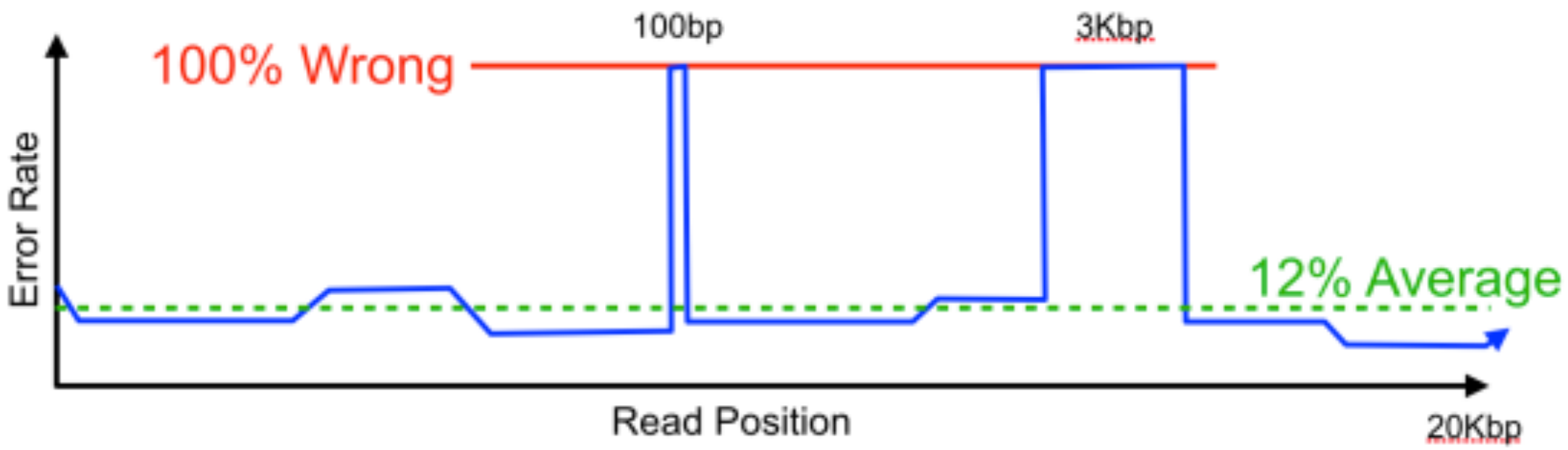


Figure 2 | **Three strategies for mapping multi-reads.** The shaded rectangles at the top represent intervals along a chromosome. The two blue rectangles below each region represent an identical two-copy repeat containing the paralogous genes A and B. The small orange bars represent reads aligned to specific positions. **a** | The 'unique' strategy reports only those reads that are uniquely mappable. Because A and B are identical, no alignments are reported. **b** | The 'best match' alignment strategy reports the best possible alignment for each read, which is determined by the scoring function of the alignment algorithm. In the case of ties, this strategy randomly distributes reads across equally good loci, as shown here. **c** | The 'all matches' strategy simply reports all alignments for each multi-read, including lower-scoring alignments.

Treangen *et al* (2012)

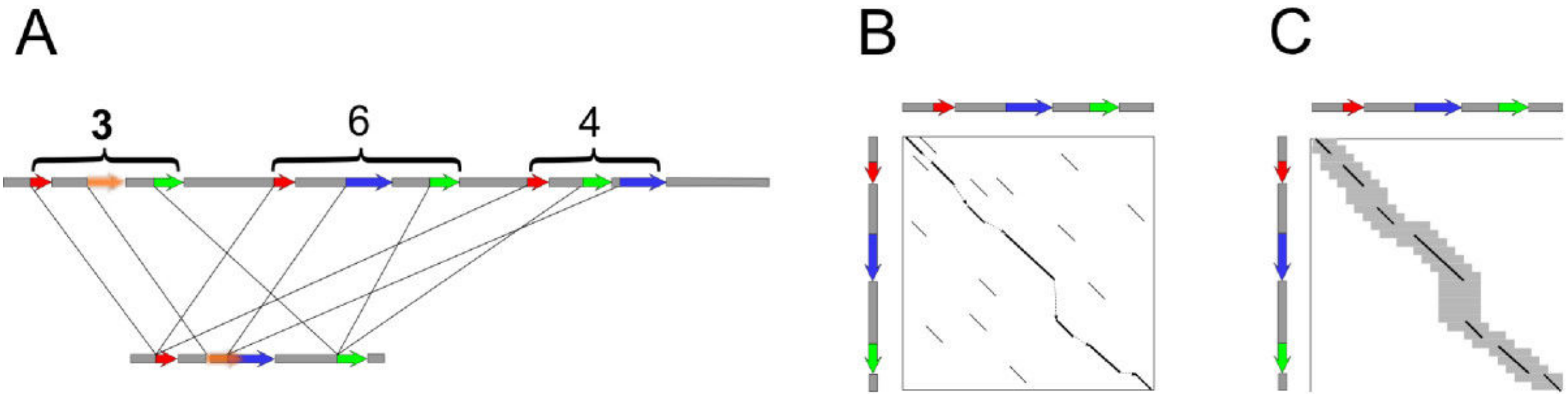
What about long read mapping?



- **BLASR** and **Daligner** designed for long error-prone (but random) reads (PacBio)
- Now there's alternative such as GMAP and minimap2 which are much faster

https://doi.org/10.1186/1471-2105-13-238

What about long read mapping?



- **BLASR**
- One of the first tools in long read alignments (meaning it's easier to understand)
- Combines multiple methods
- Starts by finding short exact matches using suffix or B-W
- Next locally identifies a linear chain of shorter exact matches
- Performs banded Smith-Waterman constrained by the shorter exact matches

https://doi.org/10.1186/1471-2105-13-238

Minimap2 – the most popular tool to use in long read alignment

1. Read **-I** [=*4G*] reference bases, extract (**-k**,**-w**)-minimizers and index them in a hash table.
2. Read **-K** [=*200M*] query bases. For each query sequence, do step 3 through 7:
3. For each (**-k**,**-w**)-minimizer on the query, check against the reference index. If a reference minimizer is not among the top **-f** [=*2e-4*] most frequent, collect its the occurrences in the reference, which are called *seeds*.
4. Sort seeds by position in the reference. Chain them with dynamic programming. Each chain represents a potential mapping. For read overlapping, report all chains and then go to step 8. For reference mapping, do step 5 through 7:
5. Let *P* be the set of primary mappings, which is an empty set initially. For each chain from the best to the worst according to their chaining scores: if on the query, the chain overlaps with a chain in *P* by **--mask-level** [=*0.5*] or higher fraction of the shorter chain, mark the chain as *secondary* to the chain in *P*; otherwise, add the chain to *P*.
6. Retain all primary mappings. Also retain up to **-N** [=*5*] top secondary mappings if their chaining scores are higher than **-p** [=*0.8*] of their corresponding primary mappings.
7. If alignment is requested, filter out an internal seed if it potentially leads to both a long insertion and a long deletion. Extend from the left-most seed. Perform global alignments between internal seeds. Split the chain if the accumulative score along the global alignment drops by **-z** [=*400*], disregarding long gaps. Extend from the right-most seed. Output chains and their alignments.
8. If there are more query sequences in the input, go to step 2 until no more queries are left.
9. If there are more reference sequences, reopen the query file from the start and go to step 1; otherwise stop.

1 Indexing

2-4 Collect and sort seeds

5-7 Reference mappings

Note: It is the backbone of many assemblers and applications

https://github.com/lh3/minimap2

What about even longer mappings (genome vs genome)

PLOS | COMPUTATIONAL BIOLOGY

RESEARCH ARTICLE

MUMmer4: A fast and versatile genome alignment system

Guillaume Marçais[1,2]*, Arthur L. Delcher[3], Adam M. Phillippy[4], Rachel Coston[3], Steven L. Salzberg[3,5], Aleksey Zimin[1,3]*

Aligner	Graphical User Interface	Multi-platform Windows/Linux	Multi-threaded	Callable from C++, scripting languages	Whole genome aln.	Short read aln.	Long read aln.	SAM format output	P-value output
MUMmer4			✓	✓	✓	✓	✓	✓	
MUMmer3					✓				
Blast	✓	✓	✓		✓				✓
Blat					✓				✓
Mauve	✓	✓			✓				
LASTZ					✓			✓	✓
bwa-mem			✓		-	✓	✓	✓	
Bowtie2			✓		-	✓	-	✓	
BLASR			✓		-	-	✓	✓	✓

What about even longer mappings (genome)

PLOS | COMPUTATIONAL BIOLOGY

RESEARCH ARTICLE

MUMmer4: A fast and versatile genome alignment system

Guillaume Marçais[1,2]*, Arthur L. Delcher[3], Adam M. Phillippy[4], Rachel Coston[3], Steven L. Salzberg[3,5], Aleksey Zimin[1,3]*

		Arabidopsis	Tardigrade	Human/Chimp
nucmer4	Wall time (min)	3.7	4.0	207
	CPU time (min)	22	26	2897
	Memory (GB)	4.6	4.9	66
Mauve	Wall time (min)	41	273	> 2 days
	CPU time (min)	38.6	268	> 2 days
	Memory (GB)	3.3	4.0	> 2 days
LASTZ default	Wall time (min)	1122	> 2 days	> 2 days
	CPU time (min)	1113	> 2 days	> 2 days
	Memory (GB)	1.3		
LASTZ match	Wall time (min)	66	77	> 2 days
	CPU time (min)	66	76	> 2 days
	Memory (GB)	0.6	0.4	

Mapping algorithm – a summary

Feature	Hash table index tools	BWT tools
Speed	Slower	Faster
Memory	Higher	Lower
Sensitivity	Higher	Lower

Build an index of your reference

Align your reads to your index

Choose an aligner!

Bowtie2, BWA-MEM, SMALT

Minimap2, GMAP, MUMMER4 (Pacbio or Nanopore)

As reads get longer, there seems to be a new generation of mappers arriving

Use the output to do subsequent analysis

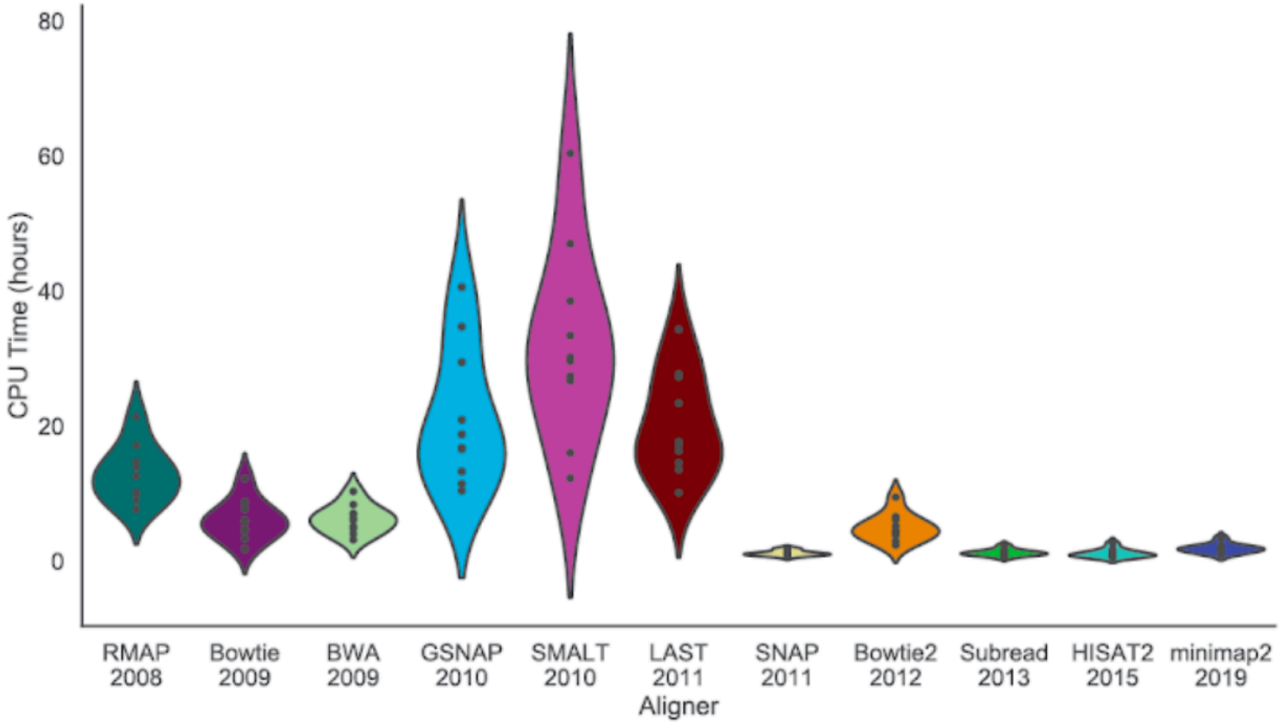
What's the output?

How to use this output?

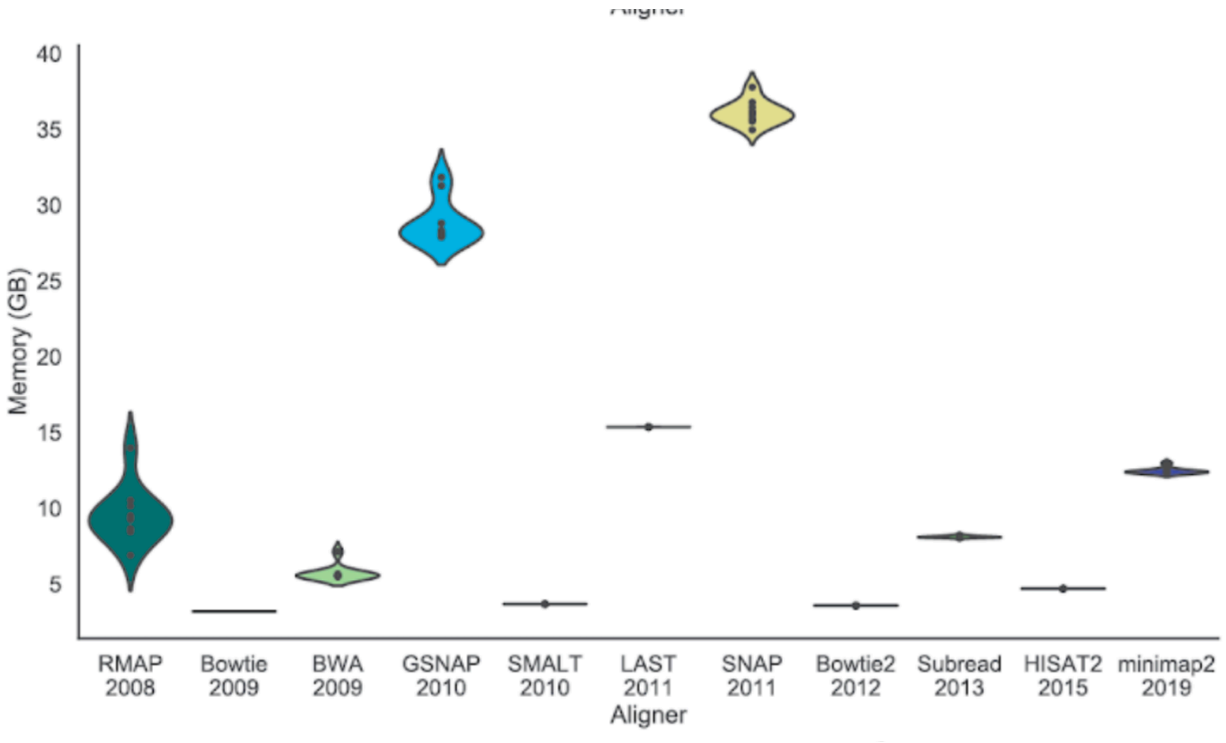
Mapping algorithm – a summary

- Improvement in both indexing and seed searching
- Allows more applications to be developed

CPU time



Memory (GB)



https://arxiv.org/ftp/arxiv/papers/2003/2003.00110.pdf

Note: **a preprint**

For your reference – speedups in various alignment algorithms including BLAST

- Improvement in both indexing and seed searching
- Allows more applications to be developed

Fast and sensitive protein alignment using DIAMOND

Benjamin Buchfink[1], Chao Xie[2,3] &
Daniel H Huson[1,2]

The alignment of sequencing reads against a protein reference database is a major computational bottleneck in metagenomics and data-intensive evolutionary projects. Although recent tools offer improved performance over the gold standard BLASTX, they exhibit only a modest speedup or low sensitivity.
We introduce DIAMOND, an open-source algorithm based on double indexing that is 20,000 times faster than BLASTX on short reads and has a similar degree of sensitivity.

Most sequence comparison programs, including BLASTX and RAPSearch2, use single consecutive seeds, which need to be short (length 3–6 amino acids) to ensure sensitivity. To increase speed without losing sensitivity, DIAMOND uses spaced seeds—that is, longer seeds in which only a subset of positions are used[9,10]. The number and exact layout of those positions are called the weight and shape of the spaced seed, respectively. To achieve high sensitivity, DIAMOND uses a set of four carefully chosen shapes[11] of length 15–24 and weight 12 by default. The most sensitive version of DIAMOND uses 16 shapes of weight 9. In addition, DIAMOND uses a reduced amino acid alphabet of size 11 to enhance sensitivity[12]. A simple exact match criterion determines which seeds are passed on to the extension phase, in which a Smith-Waterman alignment[13] is computed.

In a recent metagenomic study of 12 permafrost samples[14], a BLASTX comparison of 176 million high-quality DNA reads against the KEGG reference database[3] was reported to require 800,000 CPU hours at a supercomputing center[15]. When we used DIAMOND with its default settings, the analysis of all 246 million reads took 2.3 h on a single workstation, producing a total of 568.9 million alignments on 43 million reads.

https://www.nature.com/articles/nmeth.3176.pdf

Mapping process

Back to the beginning: FASTQ

```
@HISEQ:409:HA7CJADXX:1:1101:1202:2113 1:N:0:GCNAAT
AAAAAAGTTTCCATACAATTACAAGCATCACACTGTGGGCATGCACTTGGGAAAGAA
+
==?DBD@<AA<ADAFHGGE<ECHHCG+:1::?D;G4::?BBGCFHI<BCCC;FCGC9
```

Read ID
Sequence

Quality score

Quality Score	Error Probability
Q40	0.0001 (1 in 10,000)
Q30	0.001 (1 in 1,000)
Q20	0.01 (1 in 100)
Q10	0.1 (1 in 10)

Q-Score Bins	Example of Empirically Mapped Q-Scores*
N (no call)	N (no call)
2–9	6
10–19	15
20–24	22
25–29	27
30–34	33
35–39	37
≥ 40	40

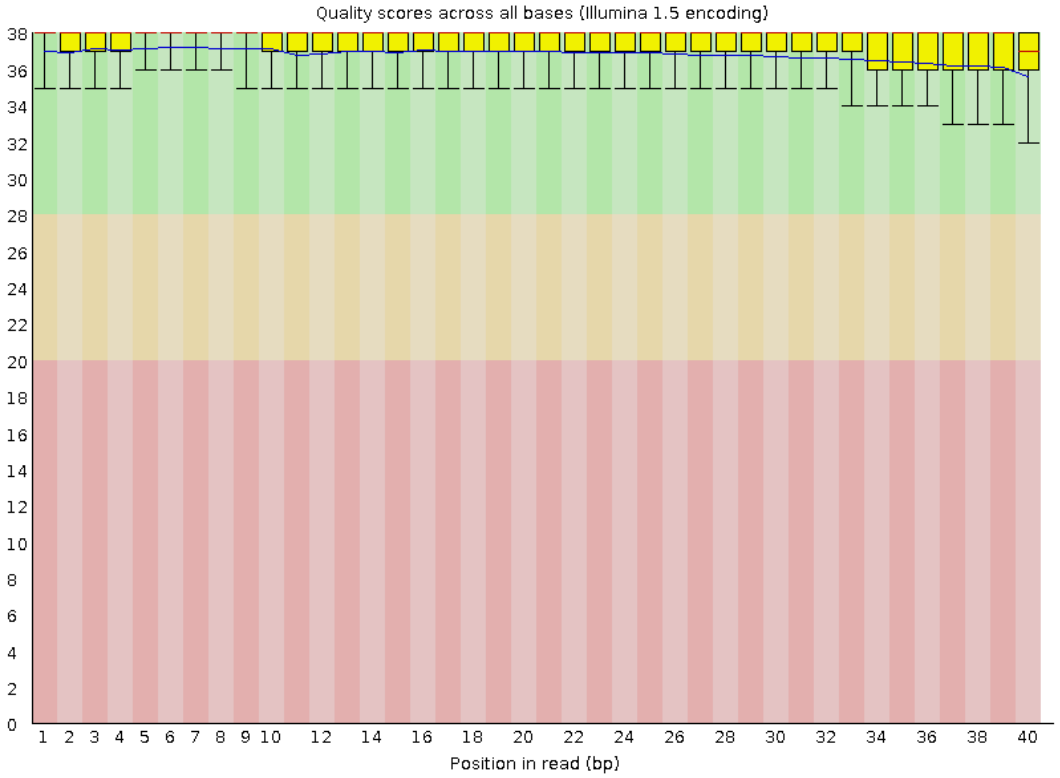
http://www.illumina.com/content/dam/illumina-marketing/documents/products/technotes/technote_understanding_quality_scores.pdf

QC first - always always the first step

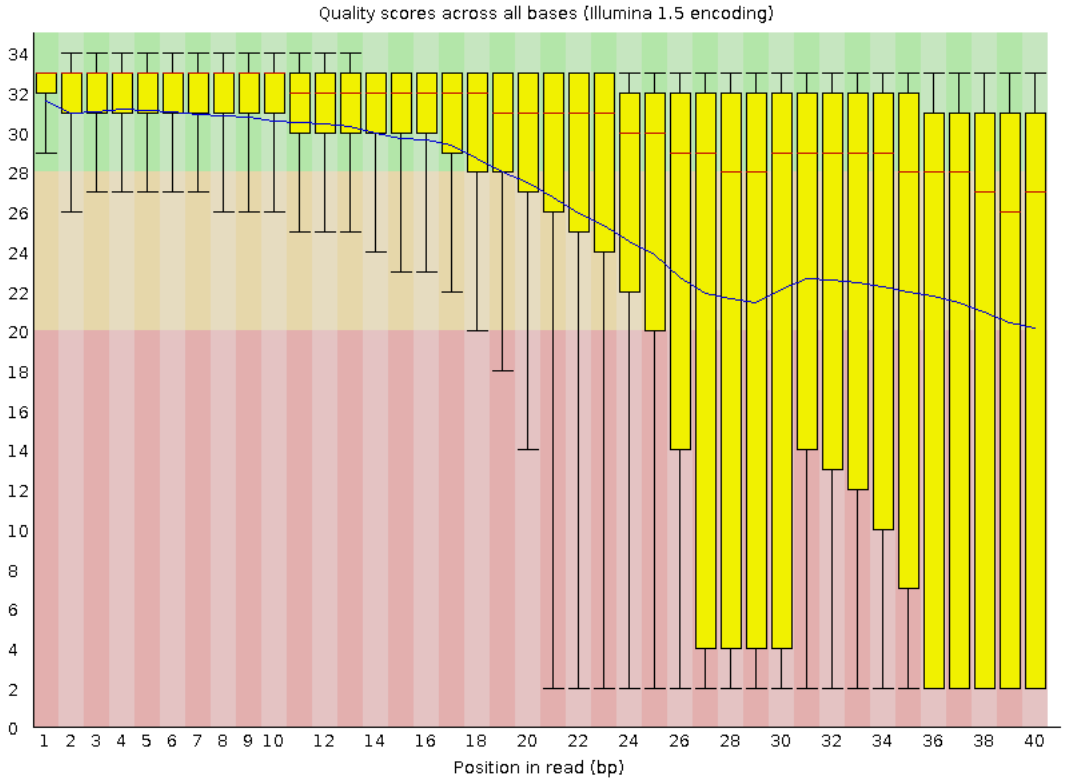
- Contamination! *
- Is it of good quality?
  - Read quality
  - Adaptor contamination
  - Insert size distribution
  - PCR duplicate rate

- Is it your species or someone else’s (sample swap)?

Sequence quality - FastQC



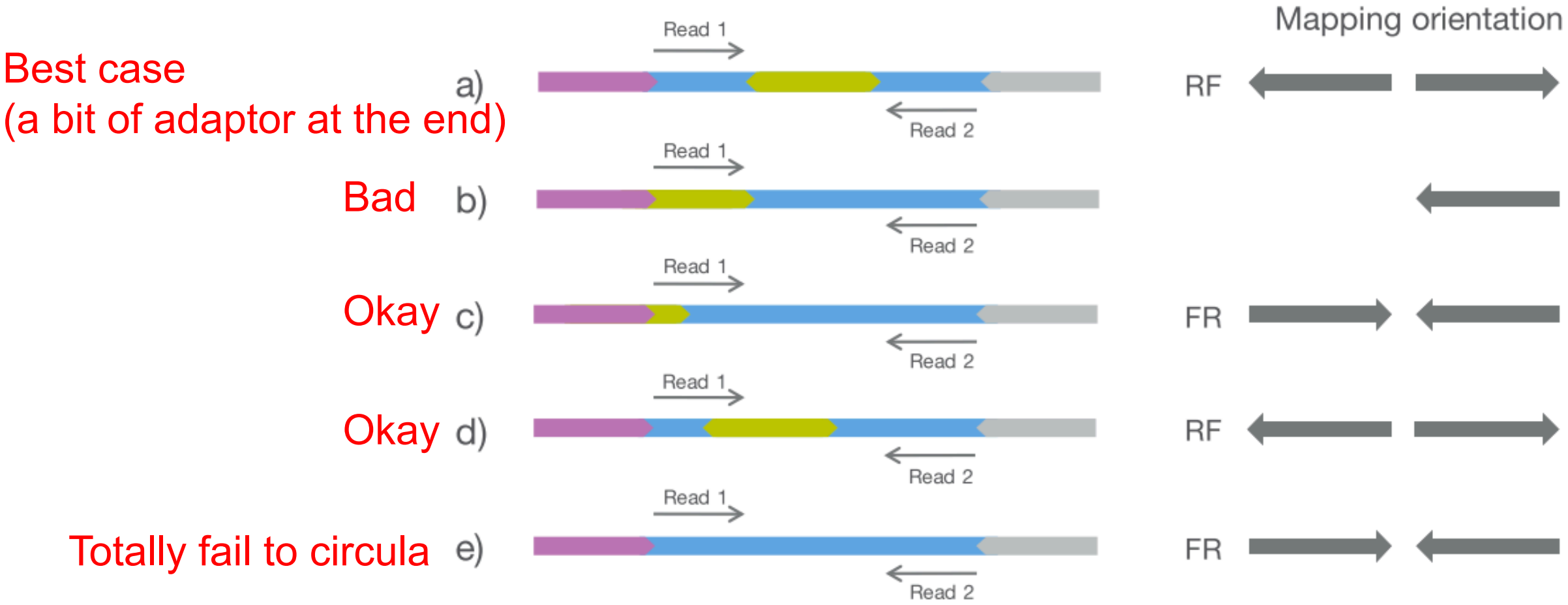
Good (unlikely)



Bad

http://www.bioinformatics.babraham.ac.uk/projects/fastqc/

Basically the adaptor sequence can appear everywhere (but in a logic way)

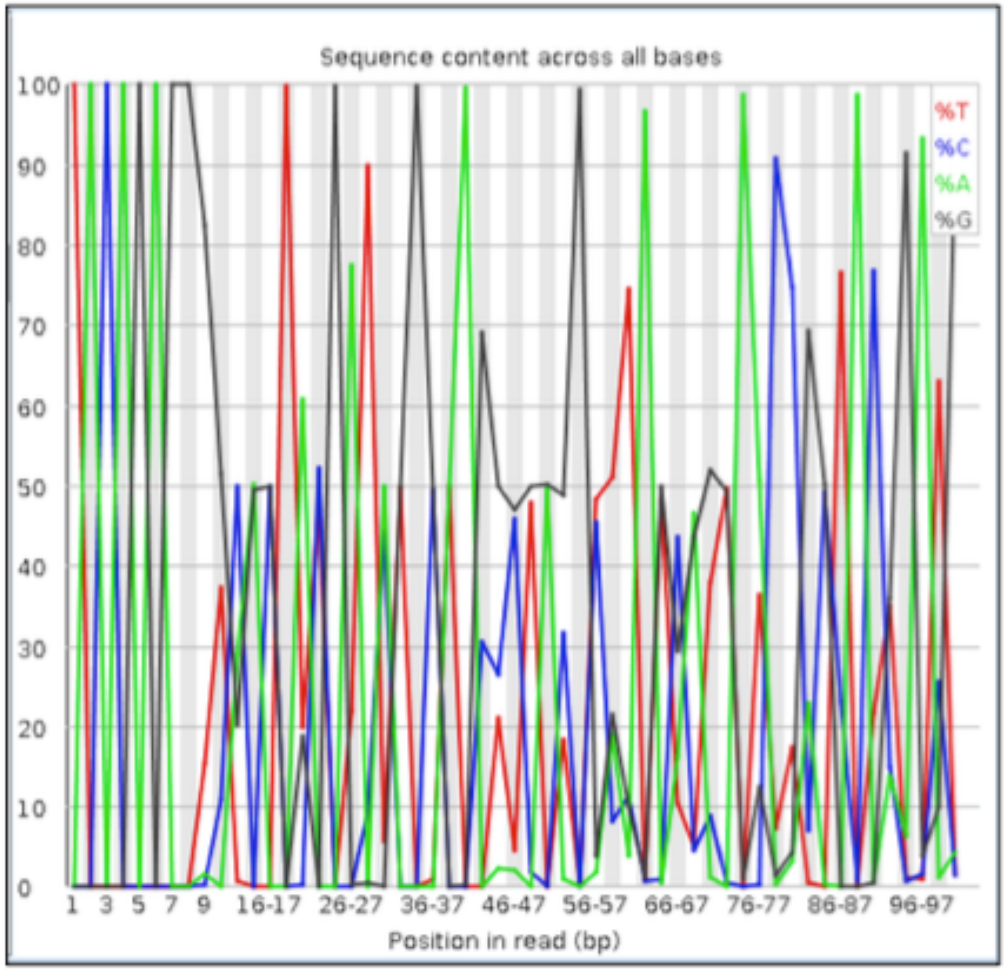


Composition of example library templates from a mate pair experiment. For each example (a–e), the position of the junction adapter sequence is shown in green and the mapping orientation (either FR or RF, 'forward-reverse' and 'reverse-forward', respectively) of the resulting read pairs is shown to the right. Sections of genomic DNA sequence are shown in blue and the TruSeq adapter sequences are shown in purple and grey.
Amplification/sequencing primer adapters are shown in grey and purple.

Illumina

FastQC will offer some insights in adaptor

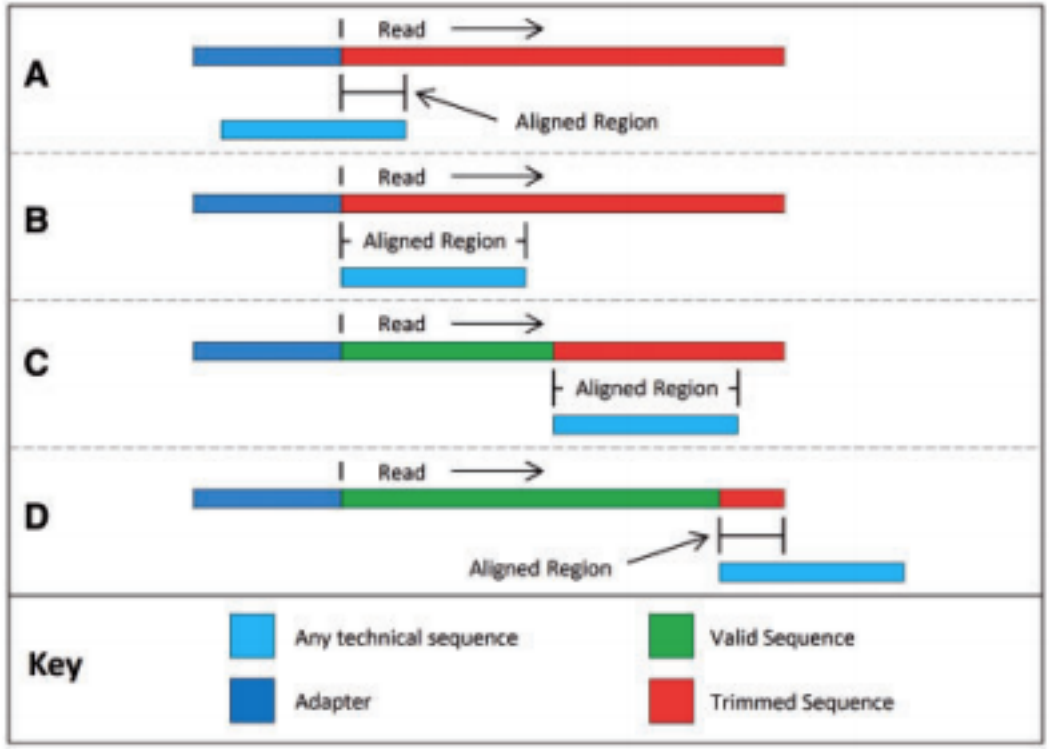
TACAGAGG overrepresented – what is it?

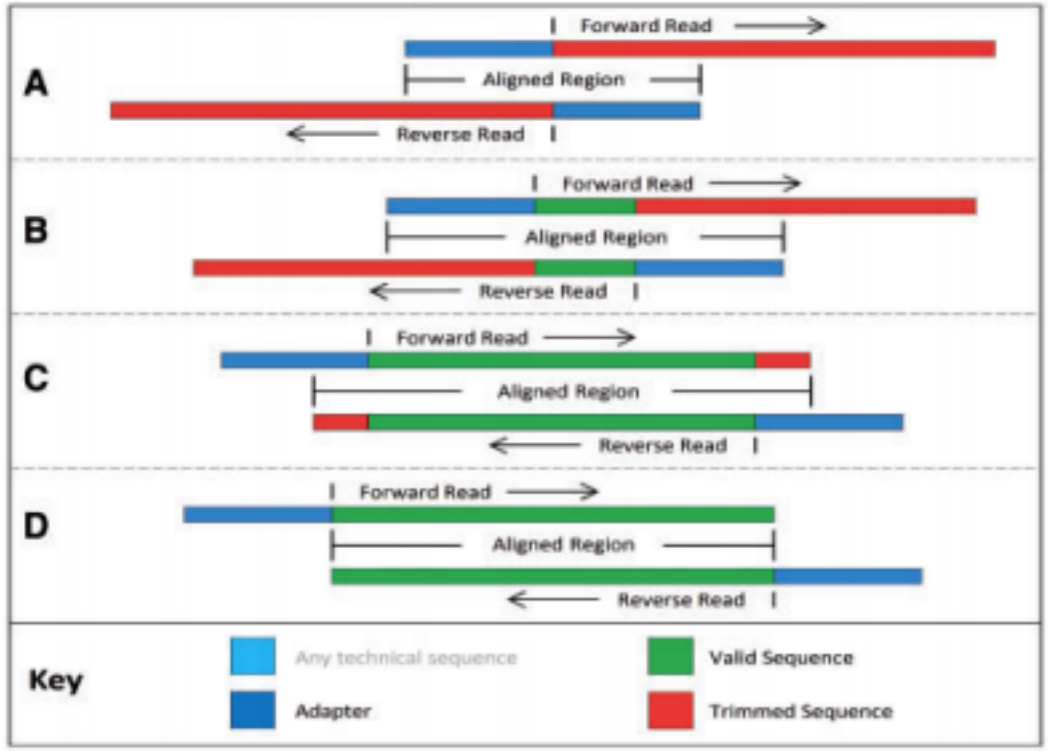


Overrepresented sequences

Sequence	Count	Percentage	Possible Source
AGCAGCATTGTACA...	3398	3.398	No Hit
TACAGTCCGACGAT...	1814	1.814	Illumina PCR Prime...
TCTACAGTCCGACG...	1570	1.57	RNA PCR Primer, In...
TATTGCACTTGTCCC...	1421	1.421	No Hit
TTCTACAGTCCGAC...	1181	1.181	RNA PCR Primer, In...
CTACAGTCCGACGA...	1168	1.168	Illumina PCR Prime...
CATTGCACTTGTCTC...	839	0.839	No Hit
ACAGTCCGACGATC...	835	0.835	RNA PCR Primer, In...
AGTTCTACAGTCCG...	648	0.648	Illumina PCR Prime...
AAAGTGCTGCGACA...	491	0.491	No Hit
TCGTATGCCGTCTT...	465	0.465	Illumina Single En...
CAGTCCGACGATCT...	436	0.436	Illumina PCR Prime...
TNNNNNNNNNNNN...	392	0.392	No Hit
TAGCTTATCAGACT...	388	0.388	No Hit
TATTGCACTCGTCC...	366	0.366	TruSeq Adapter, I...
ACCGGGCGGAAAC...	357	0.357	No Hit
ANNNNNNNNNNNN...	355	0.355	No Hit
GTTCTACAGTCCGA...	353	0.353	Illumina PCR Prime...
AAGTGCTGCGACAT...	341	0.341	No Hit

Trimmomatic for quality and adaptor trimming (many other tools also exist)

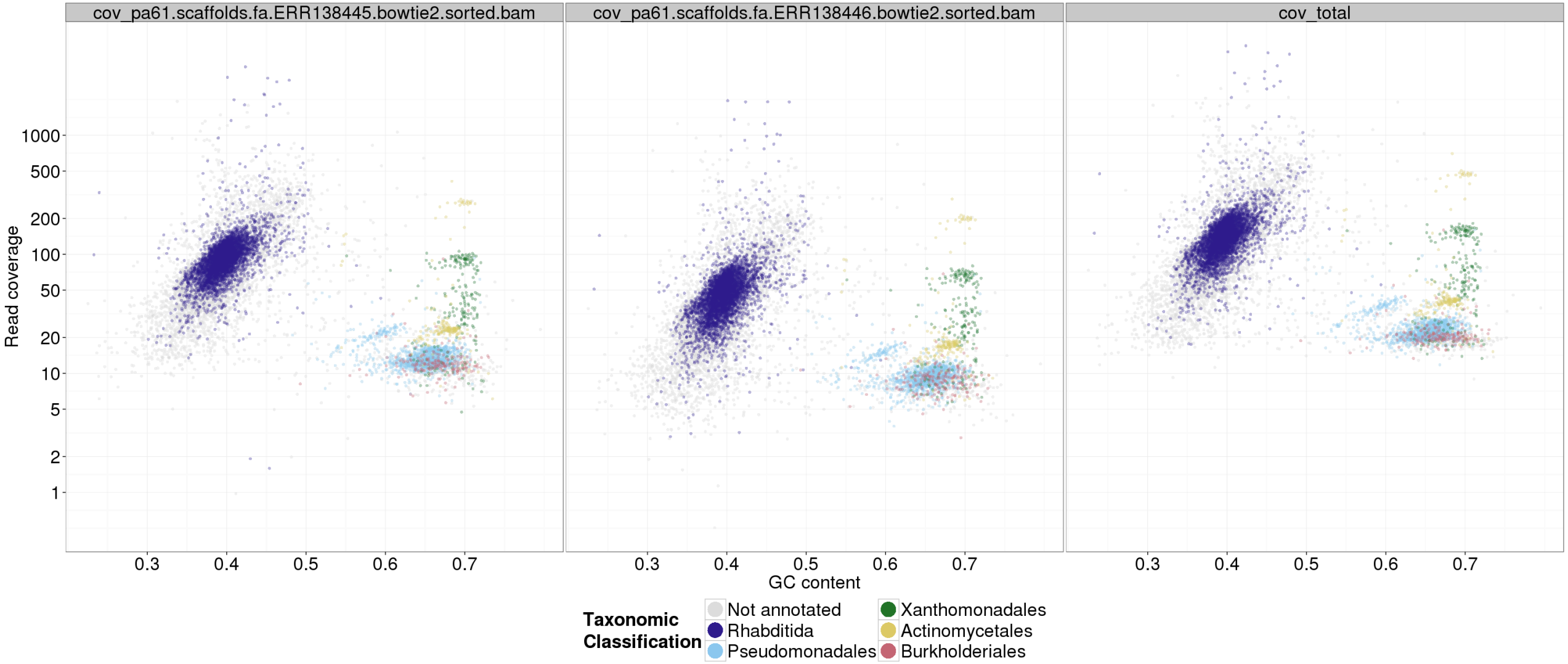




Trimmomatic: a flexible trimmer for Illumina sequence data - NCBI - NIH
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4103590/ ▾ 翻譯這個網頁
由 AM Bolger 著作 - 2014 - 被引用 5183 次 - 相關文章
2014年4月1日 - Motivation: Although many next-generation sequencing (NGS) read preprocessing tools already existed, we could not find any tool or combination of tools that met our requirements in terms of flexibility, correct handling of paired-end data and high performance. We have developed Trimmomatic as a more ...

Bolger *et al.*, (2014)

Check what your samples contain - Blobology



https://github.com/blaxterlab/blobology

Source of contamination

- Difficult to remove (gut from microorganisms)
- Fail to remove
- Not careful
- Bad company
- Sequencer carry over (from previous run)
- Sample (barcode) mix up

- Or simply bad day (not your fault)

Salter *et al. BMC Biology* 2014, **12**:87
http://www.biomedcentral.com/1741-7007/12/87

BMC Biology

RESEARCH ARTICLE Open Access

Reagent and laboratory contamination can critically impact sequence-based microbiome analyses

Source of contamination

Mukherjee *et al. Standards in Genomic Sciences* 2015, **10**:18
http://www.standardsingenomics.com/content/10/1/18

Standards in Genomic Sciences

COMMENTARY Open Access

Large-scale contamination of microbial isolate genomes by Illumina PhiX control

Supratim Mukherjee[1*], Marcel Huntemann[1], Natalia Ivanova[1], Nikos C Kyrpides[1,2] and Amrita Pati[1]

….In this study we screened over **18,000 publicly available microbial isolate genome sequences in the Integrated Microbial Genomes database and identified more than 1000 genomes that are contaminated with PhiX, a control frequently used during Illumina sequencing runs**. …….The presence of PhiX contamination in several publicly available isolate genomes can result in additional errors when such data are used in comparative genomics analyses. **Such contamination of public databases have far-reaching consequences in the form of erroneous data interpretation and analyses, and necessitates better measures to proofread raw sequences before releasing them to the broader scientific community.**

Sample storage matters (case of humans)

3 months storage resulted in less efficient DNA extraction

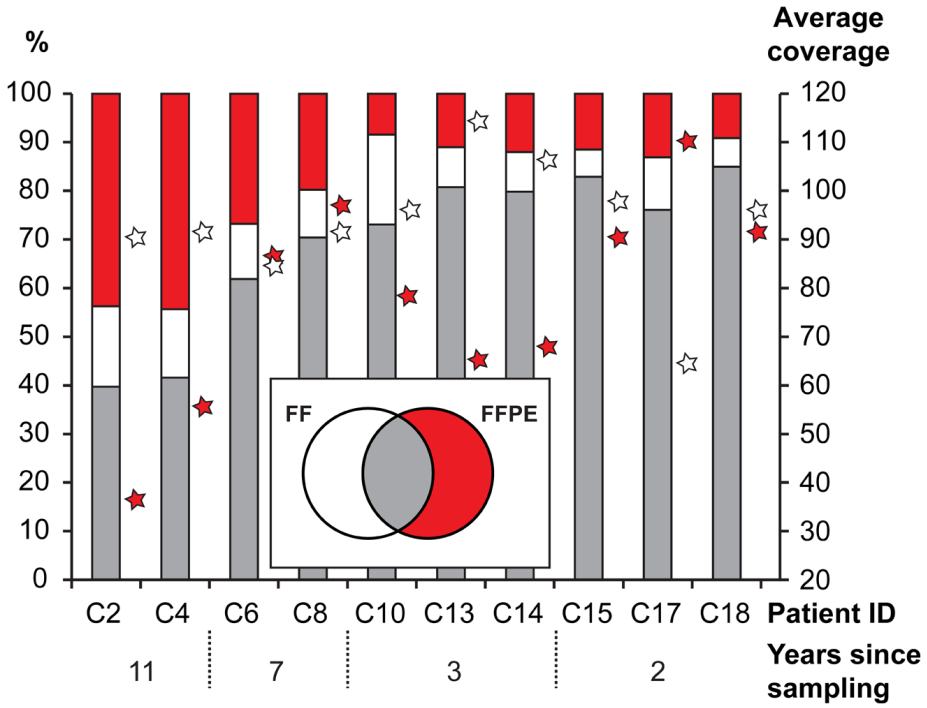
- High fragmentation: loss of material
- Decrease in library complexity
- High increase in PCR duplicates, 60-85% for FFPE vs. 30% for FF

C > U deamination is a common cause of artifacts

- U-tolerant polymerase didn't help
- Pattern, T <> C, A <> G transition

The fraction of mapped reads decreases with storage time

- Increase in partial mappings
- Increase in gapped mappings



Hedegaard et al. 2014

Mapping output format: SAM/BAM

Spec defined by maq/bwa/samtools author Heng Li

SAM: text version
tab-delimited
Exome (GBs) ; Whole genome (TBs)

BAM: binary/compressed version
indexed so it’s faster to look up using samtools
Exome (1-2GBs) ; Whole genome (GBs)

SAM file header

```
@HD     VN:1.4  SO:coordinate
@SQ     SN:PNOK.scaff0001.C     LN:7761079
@SQ     SN:PNOK.scaff0002.C     LN:4533150
@SQ     SN:PNOK.scaff0003.C     LN:3409659
@SQ     SN:PNOK.scaff0004.0     LN:3380754
@SQ     SN:PNOK.scaff0005.0     LN:2749859
@SQ     SN:PNOK.scaff0006.0     LN:2613677
@SQ     SN:PNOK.scaff0007.0     LN:1690816
@SQ     SN:PNOK.scaff0008       LN:1673160
@SQ     SN:PNOK.scaff0009.0     LN:1538597
@SQ     SN:PNOK.scaff0010       LN:1377172
@SQ     SN:PNOK.scaff0011       LN:633856
@SQ     SN:PNOK.scaff0012       LN:52253
@SQ     SN:PNOK.mito    LN:163443
@PG     ID:smalt        VN:0.7.4        CL:/h
```

Always start with @

Contains “background” information

@HD = Header

@SQ = Sequence dictionary

SAM file header

Very detailed in how one should specify the headers

Subsequent programs (like variant calling) will use these info

http://samtools.github.io/hts-specs/SAMv1.pdf

Tag	Description
@HD	The header line. The first line if present.
VN*	Format version. *Accepted format*: /^[0-9]+\.[0-9]+$/.
SO	Sorting order of alignments. *Valid values*: **unknown** (default), **unsorted**, **queryname** and **coordinate**. For coordinate sort, the major sort key is the RNAME field, with order defined by the order of @SQ lines in the header. The minor sort key is the POS field. For alignments with equal RNAME and POS, order is arbitrary. All alignments with '*' in RNAME field follow alignments with some other value but otherwise are in arbitrary order.
GO	Grouping of alignments, indicating that similar alignment records are grouped together but the file is not necessarily sorted overall. *Valid values*: **none** (default), **query** (alignments are grouped by QNAME), and **reference** (alignments are grouped by RNAME/POS).
@SQ	Reference sequence dictionary. The order of @SQ lines defines the alignment sorting order.
SN*	Reference sequence name. Each @SQ line must have a unique SN tag. The value of this field is used in the alignment records in RNAME and RNEXT fields. Regular expression: [!-)+-<>-~][!-~]*
LN*	Reference sequence length. *Range*: $[1,2^{31}-1]$
AS	Genome assembly identifier.
M5	MD5 checksum of the sequence in the uppercase, excluding spaces but including pads (as '*'s).
SP	Species.
UR	URI of the sequence. This value may start with one of the standard protocols, e.g http: or ftp:. If it does not start with one of these protocols, it is assumed to be a file-system path.
@RG	Read group. Unordered multiple @RG lines are allowed.
ID*	Read group identifier. Each @RG line must have a unique ID. The value of ID is used in the RG tags of alignment records. Must be unique among all read groups in header section. Read group IDs may be modified when merging SAM files in order to handle collisions.
CN	Name of sequencing center producing the read.
DS	Description.
DT	Date the run was produced (ISO8601 date or date/time).
FO	Flow order. The array of nucleotide bases that correspond to the nucleotides used for each flow of each read. Multi-base flows are encoded in IUPAC format, and non-nucleotide flows by various other characters. *Format*: /*|[ACMGRSVTWYHKDBN]+/
KS	The array of nucleotide bases that correspond to the key sequence of each read.
LB	Library.
PG	Programs used for processing the read group.
PI	Predicted median insert size.
PL	Platform/technology used to produce the reads. *Valid values*: CAPILLARY, LS454, ILLUMINA, SOLID, HELICOS, IONTORRENT, ONT, and PACBIO.
PM	Platform model. Free-form text providing further details of the platform/technology used.
PU	Platform unit (e.g. flowcell-barcode.lane for Illumina or slide for SOLiD). Unique identifier.
SM	Sample. Use pool name where a pool is being sequenced.
@PG	Program.
ID*	Program record identifier. Each @PG line must have a unique ID. The value of ID is used in the alignment PG tag and PP tags of other @PG lines. PG IDs may be modified when merging SAM files in order to handle collisions.
PN	Program name
CL	Command line

SAM file mapping

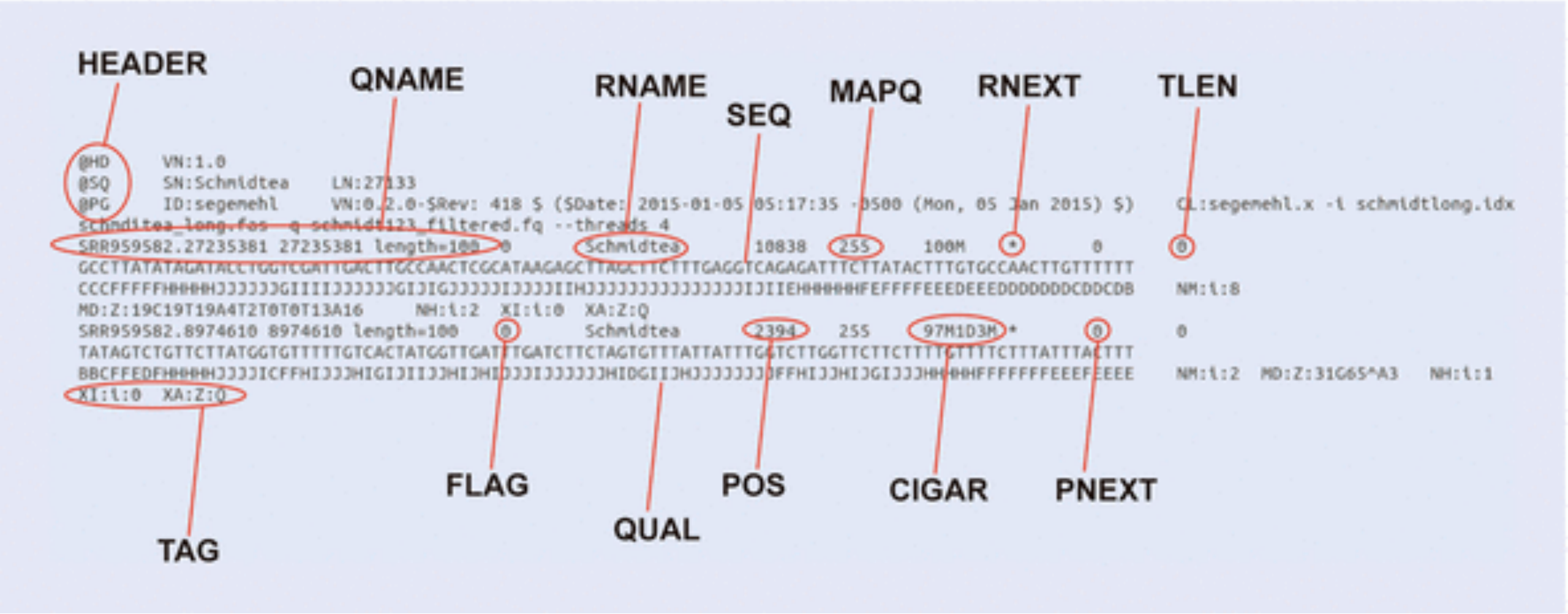
Read 1

Read 2

```
HWI-M01162:89:000000000-AC0DK:1:1103:14628:10475        163     PNOK.scaff0001.C        539     60
      7S49M1I34M1D179M        =       680     441     AACAATCCACATGGTATGTTTCTATTGTTACTACAAGATTATTG
ATCATTCTATTTTGCAACGATGGCGTGATCCTCGCGTTATACGATTTTAACTAAGGAGACCTGACGAGTATTTATACCAAAGAGGTCTACAGGAGAGG
GGAGTCAAATCCCCACCTCTCGTCTTCTTAGATCCTCTCATTACCTCGCTTCGCTGTGCTCAGCTTCGAACCTAACACCTAGGTGTCACATGATTAGG
ATTAGTAAGCAAATTACTTAATCAAATGGT  CCCCCGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
GGGGGGGGGGGEGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGDFGGGGGFFGFFFFFFFFFFFFGFFFFFFFFFFBFFFFFECEFFFFAEFFFFFCA<EFF<CC
FFFFFFFF  PG:Z:MarkDuplicates     NM:i:18 AS:i:206
HWI-M01162:89:000000000-AC0DK:1:1107:19538:18571        81      PNOK.scaff0001.C        570     29
      282M    =       571     282     TTACTGATCATTCCATTTGCAACGATGGCGTGATCCTCGCATCATACAATTACTAACTAA
GGAGACCTGACGAGTATTTATACCAAAGAGGTCTACAGGAGAGGGGAGTCAAATCCCCACCTCTCGTCTCTTTAGATCCTCTCATTACCTCGCTTCGC
TGCGCTCAGCTTCGAACCTAATAGTTAAGTGTCACGTGATTAGGATTAGTAAGCAAATTACTTAATCATATGGTCACTAATATGCTTTGTCATAGATA
TTACATAACAGGCACATGTTCGTATC      FFFFFFFFFFFFFFFFFGFGFGGFGGFGGGGGGGGGGGGGGFGGGGGGGGGGGGGGGGGGGGGGGGG
GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGFGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
GGGGGGGGFGGGGGGCCCCC      PG:Z:MarkDuplicates     NM:i:0  AS:i:282
```

Sorted by chromosome position

SAM file mapping



SAM file spec

Col	Field	Type	Regexp/Range	Brief description
1	QNAME	String	`[!-?A-~]{1,254}`	Query template NAME
2	FLAG	Int	$[0,2^{16}-1]$	bitwise FLAG
3	RNAME	String	`\*|[!-()+-<>-~][!-~]*`	Reference sequence NAME
4	POS	Int	$[0,2^{31}-1]$	1-based leftmost mapping POSition
5	MAPQ	Int	$[0,2^{8}-1]$	MAPping Quality
6	CIGAR	String	`\*|([0-9]+[MIDNSHPX=])+`	CIGAR string
7	RNEXT	String	`\*|=|[!-()+-<>-~][!-~]*`	Ref. name of the mate/next read
8	PNEXT	Int	$[0,2^{31}-1]$	Position of the mate/next read
9	TLEN	Int	$[-2^{31}+1,2^{31}-1]$	observed Template LENgth
10	SEQ	String	`\*|[A-Za-z=.]+`	segment SEQuence
11	QUAL	String	`[!-~]+`	ASCII of Phred-scaled base QUALity+33

http://samtools.github.io/hts-specs/SAMv1.pdf

SAM file mapping

Col	Field	
1	QNAME	HWI-M01162:89:000000000-AC0DK:1:2103:24027:15590
2	FLAG	163
3	RNAME	PNOK.scaff0001.C
4	POS	539
5	MAPQ	60
6	CIGAR	300M
7	RNEXT	=
8	PNEXT	680
9	TLEN	441
10	SEQ	CGATAACAAATAATCACATGGTATGTTTCTATTGTCACTACAAGATTACTGATCATTCCATTTGCAACGATGG CGTGATCCTCGCATCATACAATTACTAACTAAGGAGACCTGACGAGTATTTATACCAAAGAGGTCTACAGGA GAGGGGAGTCAAATCCCCACCTCTCGTCTCTTTAGATCCTCTCATTACCTCGCTTCGCTGCGCTCAGCTTC GAACCTAATAGTTAAGTGTCACGTGATTAGGATTAGTAAGCAAATTACTTAATCATATGGTCACTAATATGCT TTGTCATAGAT
11	QUAL	B>>FFFFFFFFFFFFEFFFFFFFFFFFFFFFFGC4FGFFFGGFFCGGGGGGGGGFGGGGGGGGGGGGGGGGGG GGDDGGGGGGGGGF9GGGFGGGGGFFGGGGGGGGGGGGGGGGGGGGGFGGGGGGGGGGGGGGGGGG GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGEGGGGGGGGGGGGGG GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG GGGGGGGGGGGGGGGGGGGGCCCCC

Storing everything means fastq can be deleted (if you are not going to try different aligners and have same pipeline)

SAM flags

hexadecimal	decimal	binary bit; 0=no, 1=yes	position of bit	description
0x1	1	"0000 0000 0001"	1	paired-end (or multiple-segment) sequencing technology
0x2	2	"0000 0000 0010"	2	each segment properly aligned according to the aligner
0x4	4	"0000 0000 0100"	3	segment unmapped
0x8	8	"0000 0000 1000"	4	next segment in the template unmapped
0x10	16	"0000 0001 0000"	5	SEQ is reverse complemented
0x20	32	"0000 0010 0000"	6	SEQ of the next segment in the template is reverse complemented
0x40	64	"0000 0100 0000"	7	the first segment in the template
0x80	128	"0000 1000 0000"	8	the last segment in the template
0x100	256	"0001 0000 0000"	9	secondary alignment
0x200	512	"0010 0000 0000"	10	not passing quality controls
0x400	1024	"0100 0000 0000"	11	PCR or optical duplicate
0x800	2048	"1000 0000 0000"	12	supplementary alignment

http://gatkforums.broadinstitute.org/gatk/discussion/7019/sam-flags-down-a-boat

CIGAR String a few examples

ATCGATCGATCGATCG Reference

ATGGACGATTCGTGAA Read mapping = 5M1D3M1I3M4S

```
203M1D4M1I48M1I13M
164M
232M
159M
162M1D4M1I43M
101M7S
227M
155M
105M
```

Soft clip usually the result of lower mapping quality

op	Description
M	Alignment match (can be a sequence match or mismatch
I	Insertion to the reference
D	Deletion from the reference
N	Skipped region from the reference
S	Soft clip on the read (clipped sequence present in <seq>)
H	Hard clip on the read (clipped sequence NOT present in <seq>)
P	Padding (silent deletion from the padded reference sequence)

CIGAR string of long reads

```
9368d6b3-3242-4583-a37c-8ecbfa44ccee    0       ref2.scaff0001  1       60      1574S76M1D12M6D17M1D10M1D3M1D6M4D2M1D6M1D16M2D20M1I10M1I5M1I5M4D4M1D3M1I12M1D42M2D7M1I
13M1I15M2I5M1I4M1I8M3D10M3D9M1D4M1I6M1D4M1D5M1I11M1D13M1D10M2D18M4I11M2I13M1D6M1D2M3D13M1D14M2D2M5D26M2D4M2D11M1I8M1I4M1I6M1D8M1D3M2D3M1I41M1I14M1D10M1D36M1D54M2D2M1D
7M1I6M5D12M2I2M1I12M2I2M1D16M1I1M2I5M1I19M1D9M1D2M1D7M1I15M1D9M2D12M1I7M2I2M1D3M1D8M1I13M1D5M1D6M1D21M3I4M1I8M1I15M3D1M2D18M3D10M1D5M1I9M1D21M1I22M4D2M1D3M2D14M1I1M1I
15M2I5M1D2M1D10M1D11M2D6M1I11M5I11M1I6M1D19M8I14M1I10M1I27M3I14M2D5M1D6M1I4M3D2M1I28M1D17M2I13M3D23M2D5M4D1M2D9M3I16M1D17M2D11M1I14M1D16M1D2M1D7M2D10M1I1M1I11M1I22M1I
54M1D9M1I9M2I9M1I3M1I14M2D7M1D4M1D7M5D1M1D14M1I15M1D3M1I9M2D13M1I4M1D5M2D10M2D1M2D13M3D16M2I4M1I5M1D3M1I17M2D6M5D46M4D14M5D4M1D21M3D45M1D17M2D5M2D1M1D2M4D30M1D29M3I9M
1D7M1D15M1I42M2I7M1D10M1I9M1D3M1D3M1D5M4D16M4I13M1D29M1D4M3D6M1I35M6D20M6I4M1I6M3D16M1D7M1I34M1D11M4D21M1D23M2D1M1D35M4D12M1D6M1I2M1I25M1D13M1I35M2D11M3D20M5D2M1D21M1
I12M2D3M2D21M3I31M1D37M3D6M1D13M1D35M1I17M1D5M3I16M1I71M1I1M1I2M1I4M1I5M1I19M4D35M2D32M1D28M1I7M2I8M1I6M1I14M1I16M1D58M2D12M1I6M1D11M1D17M3I28M2I2M1D18M2I6M2D3M1D9M1D
15M2D1M1D16M1I5M1I30M4I6M2I9M1I5M1D26M2D2M3D1M1D4M1I13M2D21M5D5M2D10M4D17M1D24M1D11M1D6M4D15M1I20M1D32M1D16M5D22M2I11M1I35M2D7M1D43M2D10M1D6M2I8M1D5M1D9M5D2M1D9M1I9M1
D35M3I1M1I9M1D9M3D5M3D5M3D19M1I23M2D17M1I4M1I8M1D13M2I11M3I1M1I20M1I1M1I22M2D36M3D19M2I19M2D9M2D6M2I1M1I9M1I30M1I10M1D3M1D23M1I31M1I20M2I14M2D1M1D15M1D22M2D3M1D9M1D10
M6D3M1D2M1D21M1I22M1I16M1I13M2I2M1I7M1D38M5D6M3D7M1D8M1D32M1I3M3D39M5D25M1D1M1D9M1I8M1D2M2D14M1D47M1D1M1D7M2I20M1D3M1D8M2D39M2D4M1D51M1D5M4D8M2I5M1D12M1D6M72S    *
```

```
606ec45a-9559-4dcc-8901-f39b24c67e21    2048    ref2.scaff0001  1       60      15175H9M2D7M1D5M1D2M1D25M1D22M1D8M1D30M1D9M2D14M2D12M1D29M1D9M1I4M2D1M1D2M5D12M1D7M1I7
M2D12M1I28M6D14M1D27M2D15M2D13M1I18M2D1M3D1M1D14M2D28M1D8M1I7M2D13M1D30M1D46M1I31M1I4M3D8M1D14M1I11M2I7M1I22M1D6M1D7M1D15M3D46M1D5M1I5M1D2M1D8M2D16M8D39M6D24M2D6M1I6M
1I12M1D15M11D13M1D20M1D11M1D3M1D2M1D4M1I23M1I22M1D7M2D12M1I5M1D51M1I11M4D7M1D12M1I2M1D4M1D9M1D43M1D50M1I12M1D1M1D22M1D11M1I4M3I3M1D13M1I7M1I2M1D20M1D13M1D7M1D7M5D30M1
D5M1D7M3D1M1D14M2I4M1I21M1I72M1I12M1I28M1I6M1D28M1D8M2D5M1I8M3D2M5D6M1D58M1I29M1D17M1D3M1D10M1D11M2D2M1D11M3D9M1D2M4D8M2D11M2I12M5D47M8D9M3D24M1I16M1I22M1D55M1D24M1I2
0M2D13M2D2M1D61M1I11M1D13M1D7M1I4M5D6M2D42M1I17M1I3M1D20M1I10M1I14M3D25M2I5M1D22M1D11M8D11M1D1M1D10M5D3M1D12M1D30M2I5M1D3M1D10M1I9M2I13M1D6M1I31M2I8M1D4M1D1M3D29M2D20
M1I6M3D7M5D13M1I24M1D12M3D1M3D22M1D9M1D25M1D13M1D1M1D4M1D28M1I40M1D5M2D9M3D13M1D5M1I19M2I17M2D16M1D2M1D7M2D11M1I35M1D7M1D29M1D5M1D2M1I22M1D5M2D19M1D5M3D19M1D5M6D13M3D
8M3D9M1D25M1D1M1D18M1D3M1D9M1D25M2D8M1D27M2D7M2D3M1D6M1D10M1D1M3D3M2D14M2D16M1I19M1D4M2I3M1D4M1D25M1D8M3D8M4D1M5D8M1I15M2D16M2D21M1D5M3D51M1I5M1I10M1I14M1D2M1I27M1D47
M1D35M2D17M3D1M2D4M1D5M2D18M4D6M3D2M3D7M4D10M1I2M1D19M2D21M1I8M3I16M1I12M1D6M2D23M1D12M1D3M3I32M2I8M2I11M4I2M1D7M1D33M2D17M2D2M2D9M1I12M5D40M2D35M1D11M1I11M3D1M2D41M1
D4M1D30M1D7M5D2M1D22M1D1M2D12M5D14M2D6M1I35M1I25M1D9M2D4M2D31M1D2M1D11M2D7M1I8M1D5M2D15M1D14M4D4M1I24M1I55M1D9M1D19M3D32M1D1M2D26M2D10M1D5M1D5M1D26M1I18M1I9M1D10M6D2M
3D64M2D12M1D8M1D12M1I12M3I24M1I20M1D16M1I18M2D19M1I31M2D4M1D11M2D15M1D8M4D11M1I13M1D22M1I28M1D7M1I46M1D8M2D26M1D1M2D5M2I9M1I2M1D40M2D20M4D15M1I12M1D5M2D13M1D10M2D5M1I
4M1I22M2D40M1I6M3D19M1D17M1I20M5D27M2D7M1D5M1D3M3D5M2I16M1D12M1D4M5D12M1I4M1D6M1D5M1D3M2D17M3I4M1I27M1I22M1D11M1D14M1I35M1D23M3D13M1D10M14D16M1I10M1D14M1I6M1D3M1D28M1
I7M1I11M1I7M2D12M4D27M1D7M1D1M3D5M1D9M3I24M2D7M3D21M1D8M1D30M1I14M7D9M2D39M1I27M3I4M1D15M1I2M2I25M1D8M2D13M1D29M4D3M6D43M3D1M3D36M1D2M1D16M1D10M1D5M1D8M1D5M1I20M2D6M1
D7M4D3M1D10M1I13M1D8M1D2M1I1M1I4M1D24M2D12M1D10M1D2M3D8M1I10M2D5M1D10M1D9M1I13M1I4M1I3M3D7M2I13M1I6M1I7M2D3M1D24M4I5M1I9M1D8M1I13M1I8M1D35M1D22M6D7M3D3M8D6M3D22M20148
```

Mapping quality

Probability that a read is mapped incorrectly

Useful for calling SNP later on

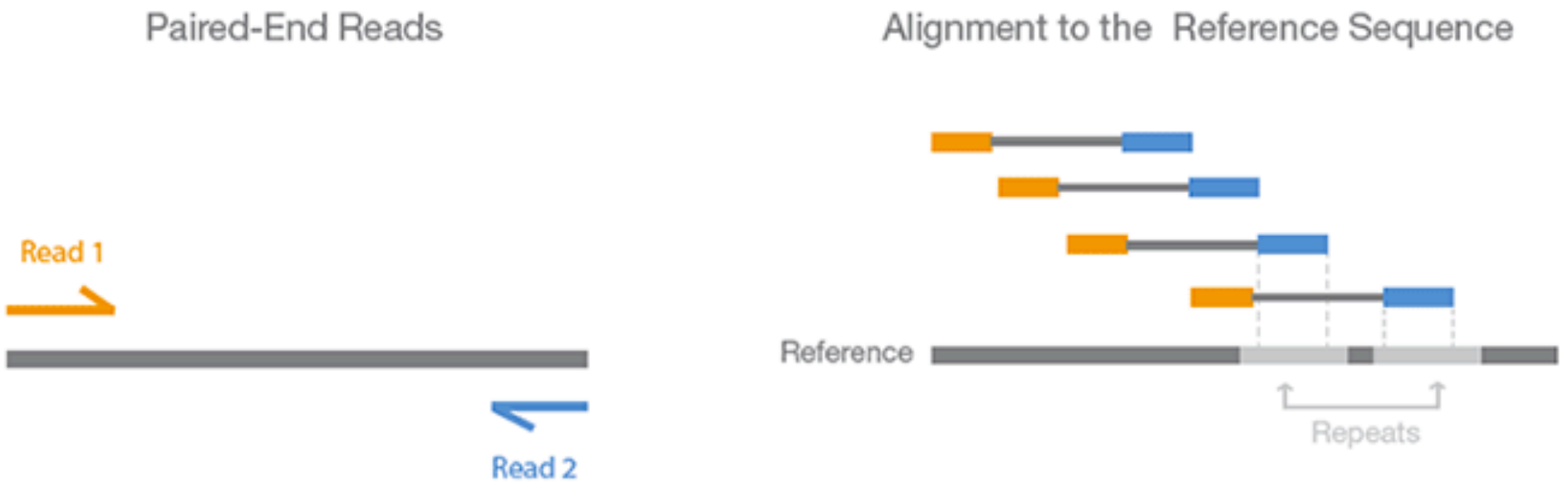
Function of

- Uniqueness
- Number of mismatches
- Number of indels
- Quality of bases in read

MQ30 = 1 in 1000 alignment is wrong
MQ40 = 1 in 10000 alignment is wrong

Post mapping QC: insert size in PE mapping?

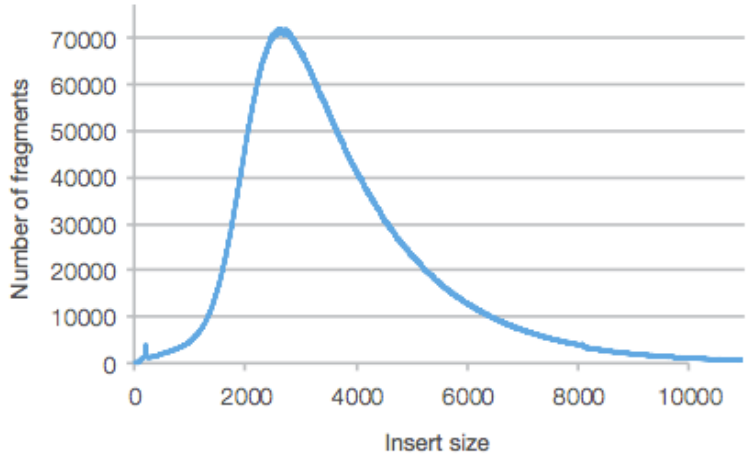
Figure 4. Paired-End Sequencing and Alignment

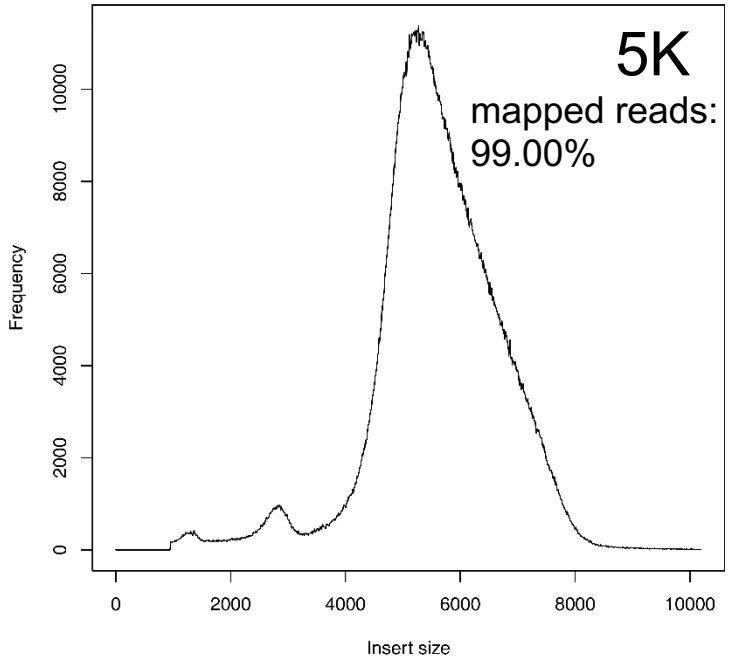


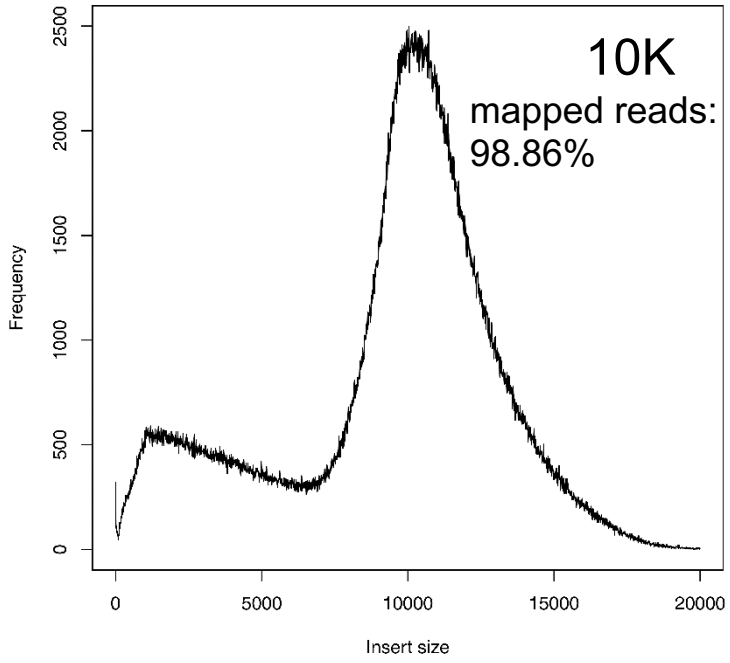
Paired-end sequencing enables both ends of the DNA fragment to be sequenced. Because the distance between each paired read is known, alignment algorithms can use this information to map the reads over repetitive regions more precisely. This results in much better alignment of the reads, especially across difficult-to-sequence, repetitive regions of the genome.

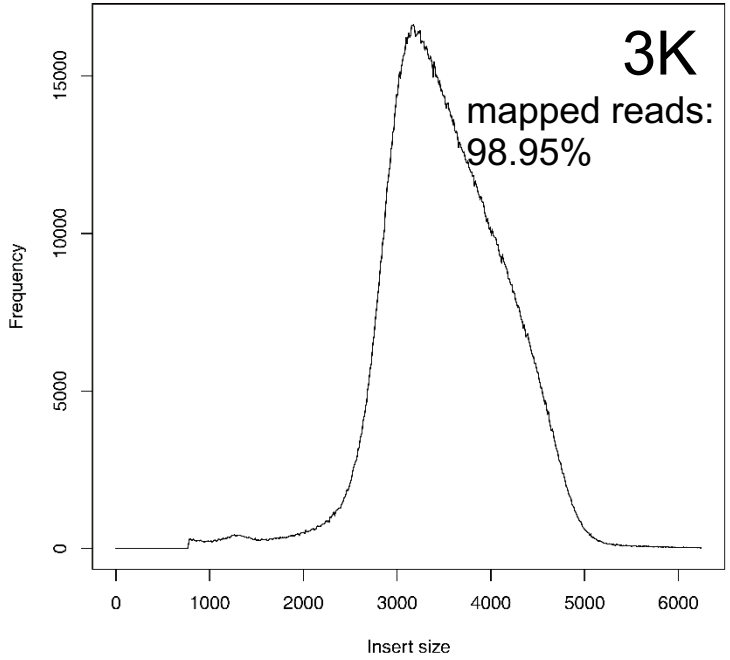
DNA fragment length should be longer than most repeat size in your genome
No point to boost up coverage if your fragment len < repeat lengtg

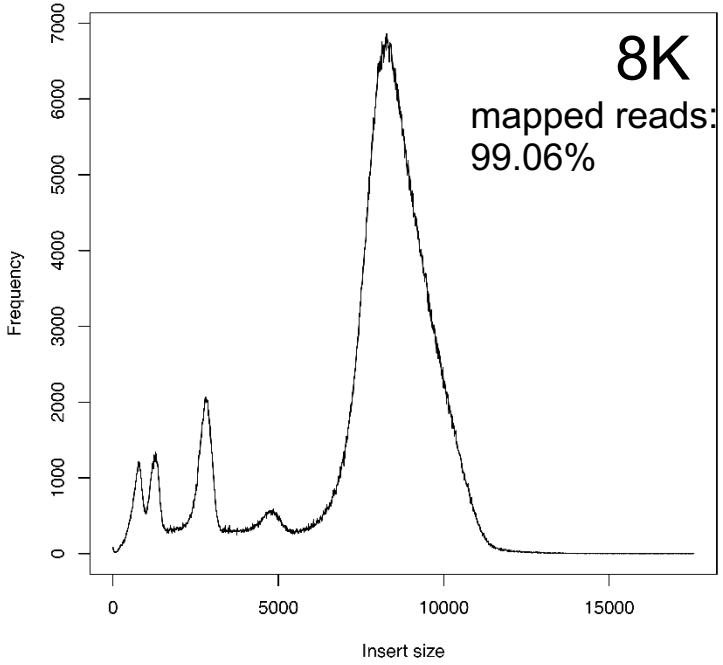
Insert size

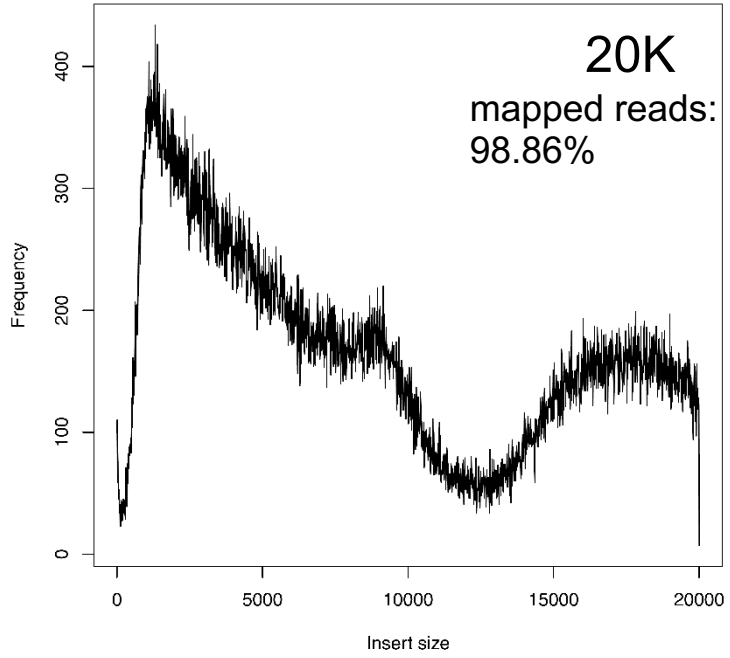

Number of fragments
70000
60000
50000
40000
30000
20000
10000
0
0
2000
4000
6000
8000
10000
Insert size


5K
mapped reads:
99.00%
Frequency
Insert size


10K
mapped reads:
98.86%
Frequency
Insert size


3K
mapped reads:
98.95%
Frequency
Insert size


8K
mapped reads:
99.06%
Frequency
Insert size


20K
mapped reads:
98.86%
Frequency
Insert size

Post mapping QC – how much coverage?

```
*************************************************
Stats for BAM file(s):
*************************************************

Total reads:        2963812
Mapped reads:       2926492      (98.7408%)
Forward strand:     1463708      (49.386%)
Reverse strand:     1462784      (49.3548%)
Failed QC:          0    (0%)
Duplicates:         8469 (0.285747%)
Paired-end reads:   2963812      (100%)
'Proper-pairs':     2808018      (94.7435%)
Both pairs mapped:  2901342      (97.8922%)
Read 1:             1481906
Read 2:             1481906
Singletons:         25150        (0.848569%)
Average insert size (absolute value): 808.327
Median insert size (absolute value): 466
```

2963812 reads
x 300 bp per read
/ 32000000bp genome
= **27.8X**

This number is overestimated because

1. ~1.3% not mapped
2. Trimmed reads (not all reads have now 300bp)

Barnett *et al.*, (2011) **bamtools**

1 million dollar question: how much coverage is better

In mapping:

- ~15X for SNP calling in bacteria
- ~30X for SNP calling in diploid (to delineate heterozygous bases)
- >50X for exome (because you need to be sure)
- No point with >100X in the Illumina world

PCR duplicates

PCR duplicates during sample prep
= the same fragment is sequenced again and again and again

Some worse than others (because starting material is not good)

< 5% is good

High duplication rate will lead to problems in downstream analysis

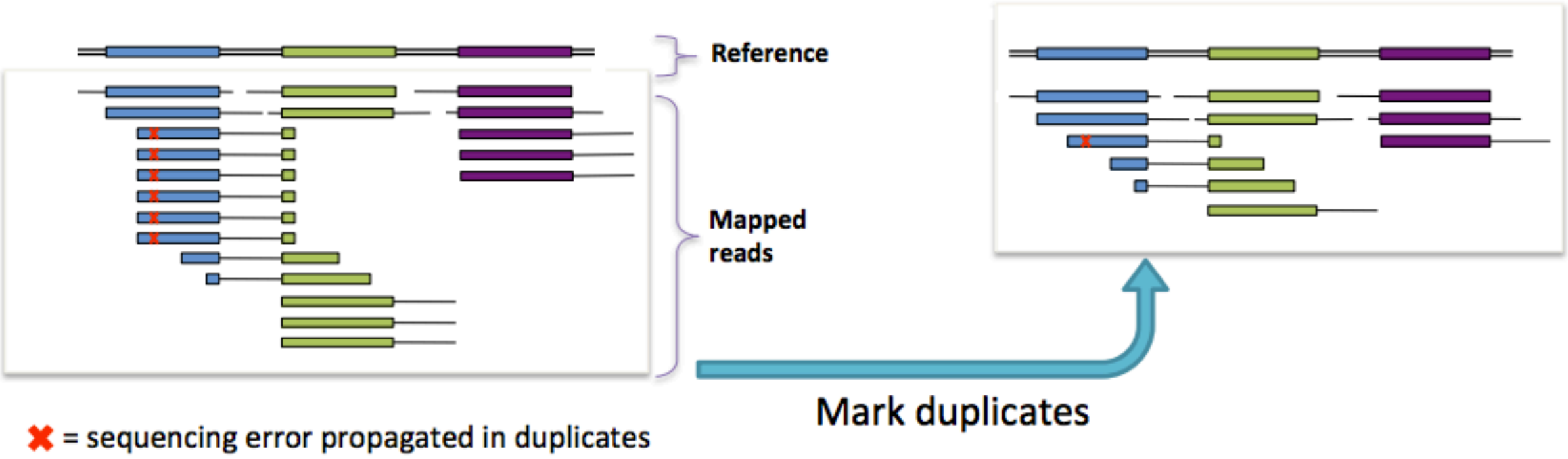
Example: 30X ; 1 out of ~30 fragment get duplicated 15 times
= skew allele frequency

= false SNP discovery

Can be detected (and removed) by read pairs map at the complete position. **We usually keep one copy only**

PCR duplicates

Can be detected (and removed) by read pairs map at the complete position.
We usually keep one copy only



Broad - GATK

De- duplication

Showing duplicate reads

Hiding duplicate reads

Broad - GATK

Case study: Check lane quality and assembly

	Total reads	Mapped reads (%)	Duplicates	Proper-pairs	Both pairs mapped	Median
BRC PE	217,190,726	95.80%	1.47%	53.97%	92.85%	968
Old PE	249,742,439	**4.40%**	1.51%	2.81%	3.08%	59
Old PE	1,167,521,211	98.21%	**11.81%**	68.97%	97.18%	465
Old PE	917,638,787	97.97%	**5.12%**	75.54%	96.99%	261
Company hmm	38,508,236	94.15%	**7.51%**	48.22%	90.53%	1681
Company hmm	76,992,221	95.09%	**10.75%**	48.57%	92.43%	1675
Company hmm	26,348,302	93.54%	**6.23%**	47.58%	89.29%	1681
Company hmm	398,746,361	98.42%	**79.36%**	57.28%	97.23%	1500
Company hmm	396,241,991	98.42%	**79.03%**	57.31%	97.24%	1500
Company hmm	39,879,176	92.45%	**29.40%**	40.66%	88.55%	4623
Company hmm	43,010,934	92.10%	**31.27%**	40.40%	87.99%	4623
Company hmm	316,963,201	97.71%	**84.14%**	57.79%	96.11%	410
Company hmm	71,118,483	96.00%	**70.88%**	42.97%	93.67%	283
Company hmm	61,803,780	94.60%	**73.18%**	45.36%	91.55%	285

Case study: Check lane quality and assembly

PE from one of my projects

Directory	TotalReads	MappedReads	MappedReads(%)	ForwardStrand	ForwardStrand(%)	ReverseStrand	ReverseStrand(%)	FailedQC	FailedQC(%)	Duplicates	Duplicates(%)
map.718-S1	3171334	2920504	92.09%	1461181	46.07%	1459323	46.02%	0	0%	28099	0.89%
map.KPN91	2963812	2926683	98.75%	1463794	49.39%	1462889	49.36%	0	0%	8464	0.29%
map.KPN92	38811800	37864479	97.56%	18931099	48.78%	18933380	48.78%	0	0%	614696	1.58%
map.NTU	151505774	140827017	92.95%	70410162	46.47%	70416855	46.48%	0	0%	72381797	47.77%

MP from a company in Japan (Nextera)

3kb.map	29432914	28598764	97.17%	14280601	48.52%	14318163	48.65%	0	0%	2369419	8.05%
5kb.map	8887196	8436683	94.93%	4208676	47.36%	4228007	47.57%	0	0%	1455786	16.38%

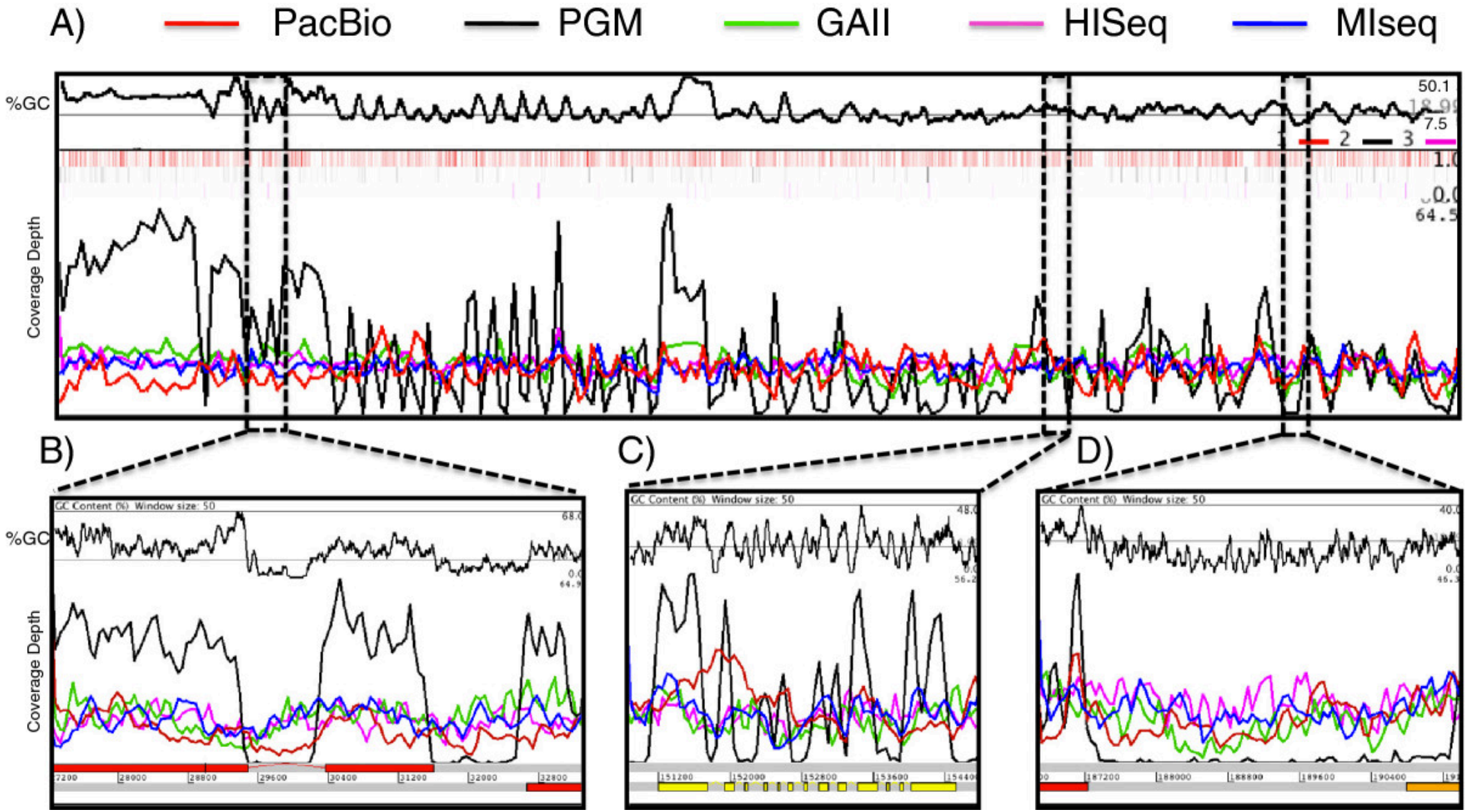
MP from another institute

Directory	TotalReads	MappedReads	MappedReads(%)	ForwardStrand	ForwardStrand(%)	ReverseStrand	ReverseStrand(%)	FailedQC	FailedQC(%)	Duplicates	Duplicates(%)
MP.2kb	87149392	61884157	71.01%	31033842	35.61%	30850315	35.40%	0	0%	6893810	7.91%
MP.4kb	92488172	60082343	64.96%	30124954	32.57%	29957389	32.39%	0	0%	6688542	7.23%
MP.6kb	79969510	50558184	63.22%	25273991	31.60%	25284193	31.62%	0	0%	3919754	4.90%
MP.9kb	63262972	44161175	69.81%	22132740	34.99%	22028435	34.82%	0	0%	6938278	10.97%

a project from BRC

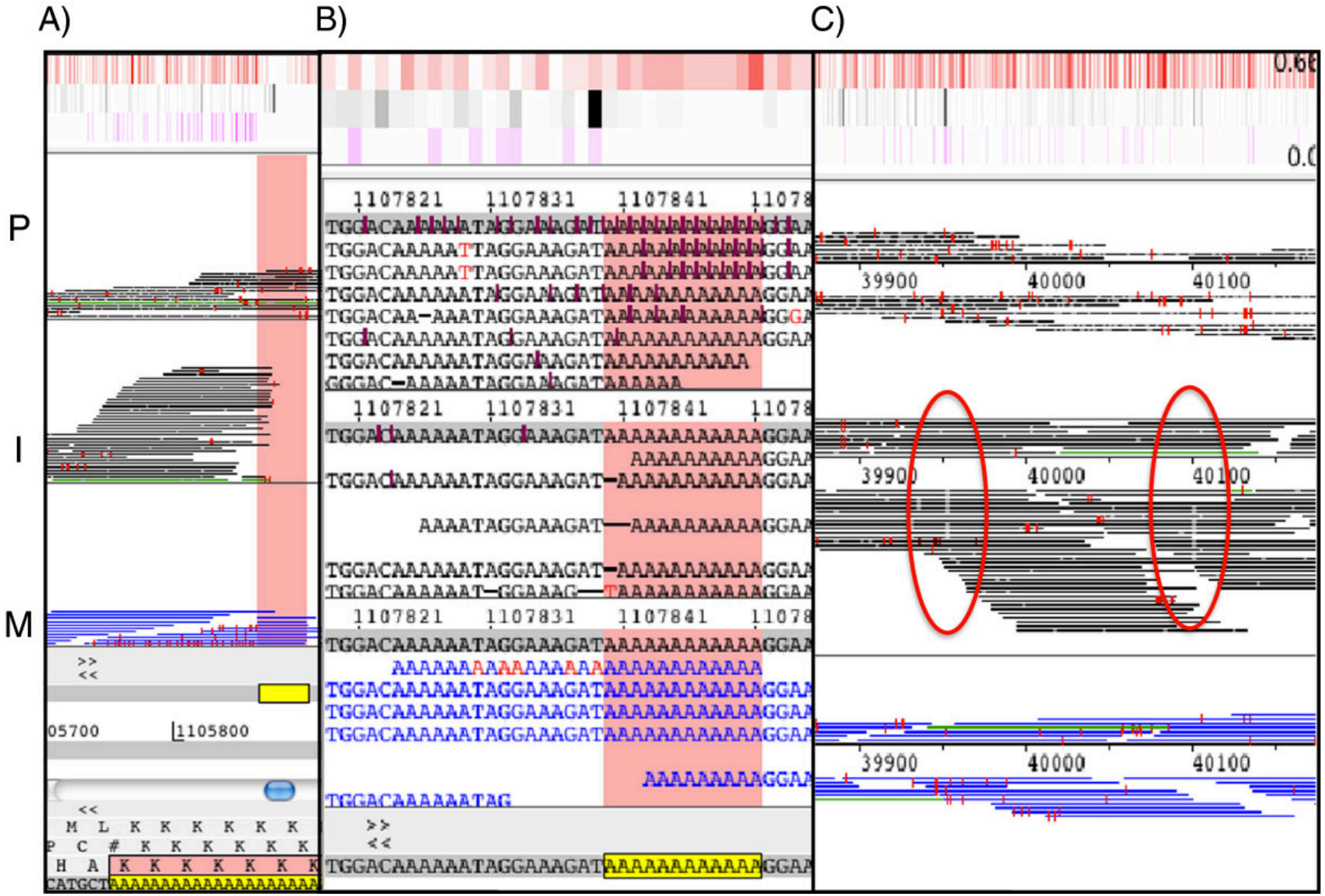
Directory	TotalReads	MappedReads	MappedReads(%)	ForwardStrand	ForwardStrand(%)	ReverseStrand	ReverseStrand(%)	FailedQC	FailedQC(%)	Duplicates	Duplicates(%)
MP10kb.map	4809196	4757184	98.92%	2380384	49.50%	2376800	49.42%	0	0%	31589	0.66%
MP15kb.map	4557418	4492023	98.57%	2247623	49.32%	2244400	49.25%	0	0%	101159	2.22%
MP4kb.map	5349212	5266083	98.45%	2633803	49.24%	2632280	49.21%	0	0%	26721	0.50%
MP6kb.map	5185824	5129611	98.92%	2566177	49.48%	2563434	49.43%	0	0%	30809	0.59%

Sequencing biases



Quail *et al* 2012

Platform specific biases



Quail *et al* 2012

Experiment biases

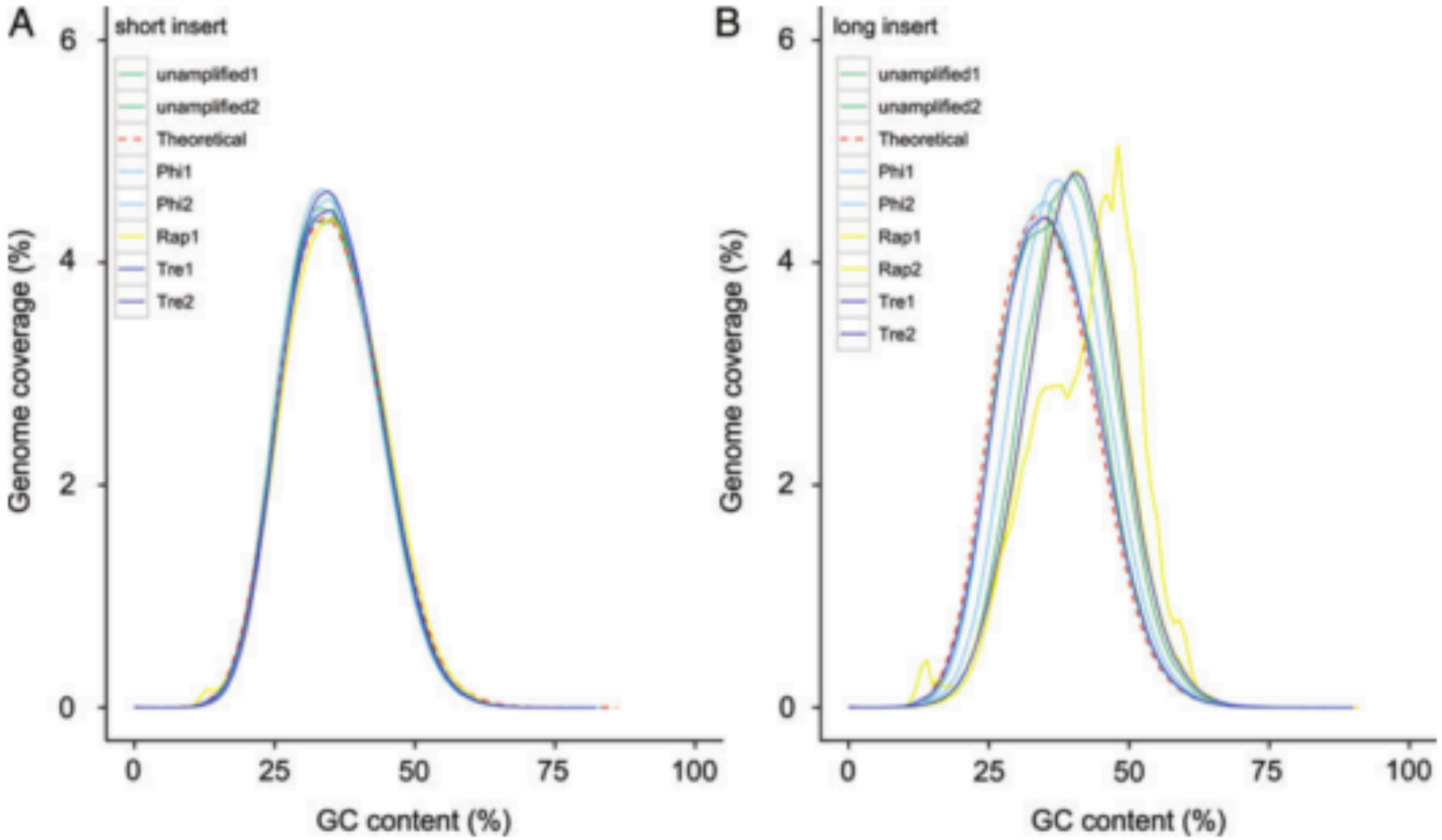


Figure 5. Distribution of GC content in sequenced reads of (A) short- and (B) long-insert libraries.

Tsai *et al* 2014

Mapping output: A summary

There is a lot you can do from the initial mapping output

- Post mapping QC
- Assembly QC

At this point you should decide whether

- it’s a good run and you can go ahead to the next stage
- you need additional run
- you need to abandon the whole run

Variant calling

Variant calling

You have just:

Mapped the reads to where they belong (supposedly)

Provided accurate mapping quality scores

Next:

Give the correct file (**BAM**) to variant callers

How to determine the above are correct?

SNP discovery

Heterozygous and homozygous SNP

```
      ATCGATGACTGACTGAATGGTTGAC
      ATCGATGACTGACTGAATGGTTGAC
      ATCCATGACTGACTGAATGGTTGAC
      ATCGATGACTGACTGAATGGTTGAC
      ATCGATAACTGACTGAATGGTTGAC
      ATCGATGACTGAGTGAATGGTTGAC
      ATCGATGACTGAGTGAATGGTTGAC
      ATCGATGACTGAGTGAATGGTTGAC
      ATCGATGACTGAGTGAATGGTTGAC
10X   ATCGATGACTGAGTGAATGGTTGAC

      …ATCGATGACTGACTGACTGGTTGAC…
                                  reference
```

INDELS (insertion deletions) and Structural variations

Indel examples

wild-type sequence

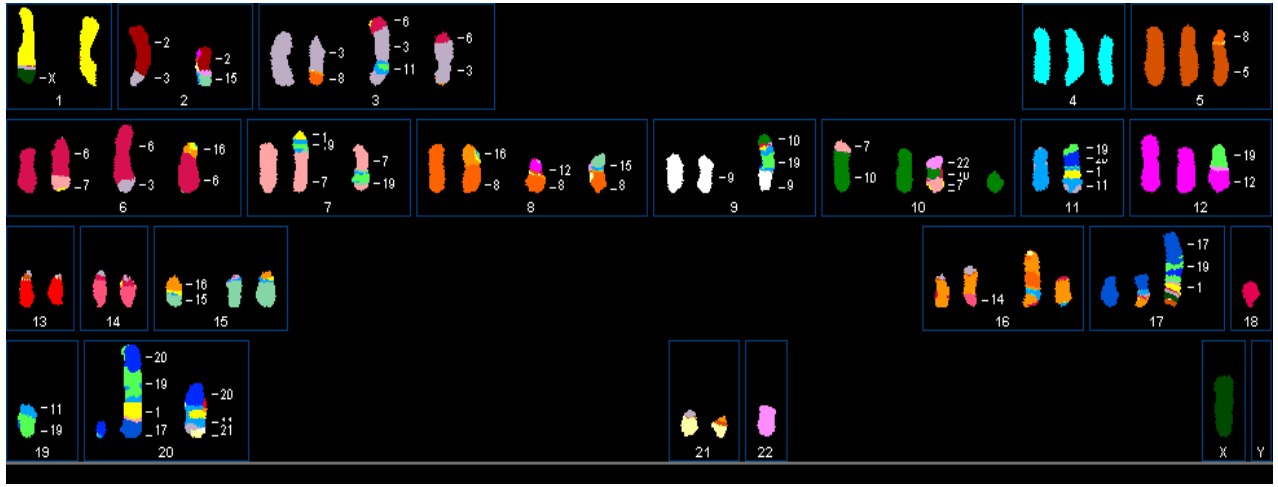
ATCTTCAGCCATAAAAGATGAAGTT

3 bp deletion

ATCTTCAGCCAAAGATGAAGTT

4 bp insertion (orange)

ATCTTCAGCCATATGTGAAAGATGAAGTT



SNP Discovery: Base Qualities

High quality

Low quality

Michael Strömberg

SNPs & Bayesian Statistics

of individuals

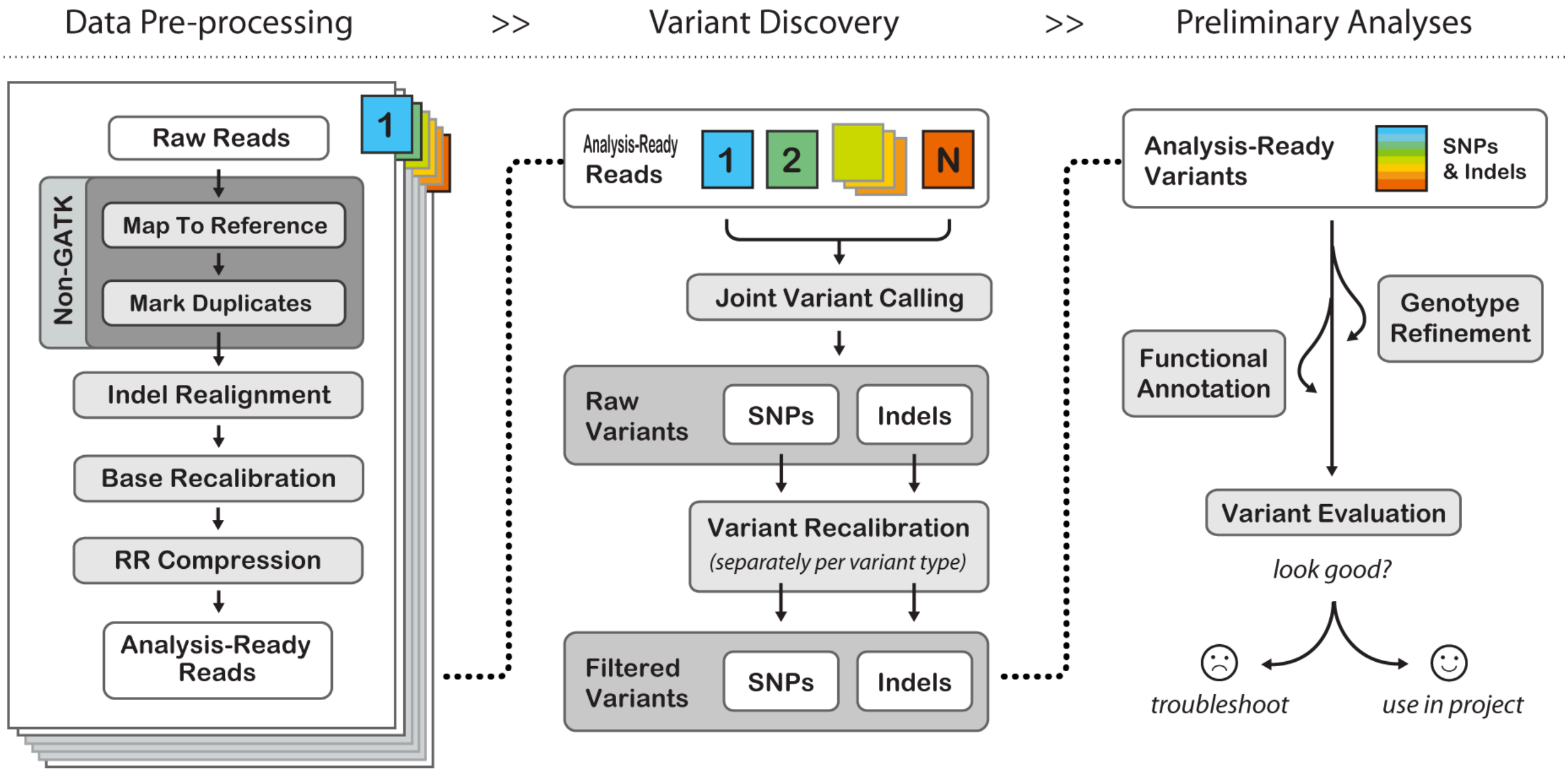
base quality

allele call in read

$$\Pr\left(G_1, G_2, \ldots, G_n \mid B\right) = \frac{\prod_{i=1}^{n}\left[\sum_{\forall T^k} \Pr\left(B_i \mid T_i^k\right)\Pr\left(T_i^k \mid G_i\right)\right]\Pr\left(G_1, G_2, \ldots, G_n\right)}{\sum_{\forall G^l}\left\{\prod_{i=1}^{n}\left[\sum_{\forall T^k} \Pr\left(B_i \mid T_i^k\right)\Pr\left(T_i^k \mid G_i^l\right)\right]\Pr\left(G_1^l, G_2^l, \ldots, G_n^l\right)\right\}}$$

Michael Strömberg

Strategies that improve variant calling



Local realignement

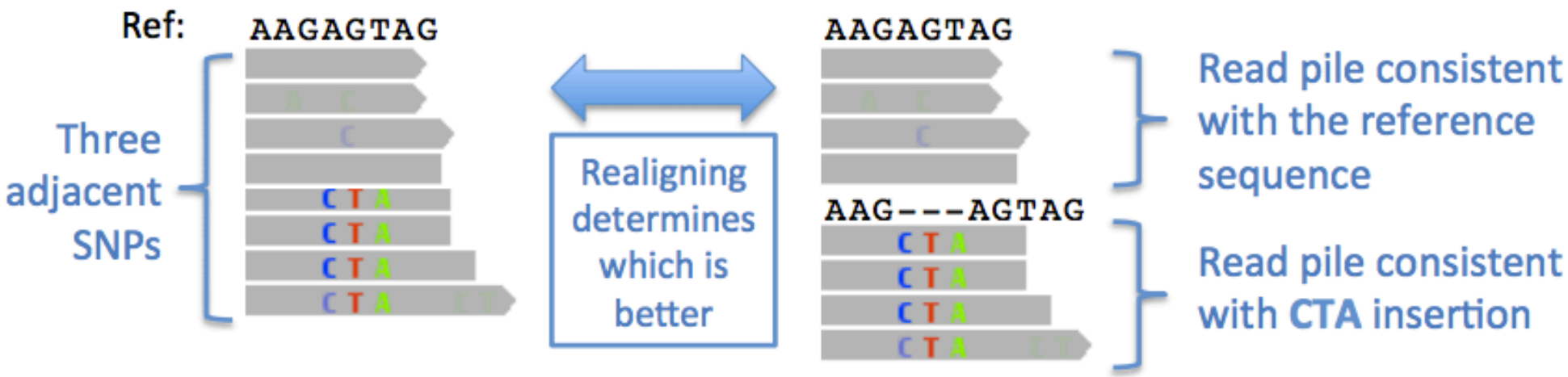
HiSeq data, raw BWA alignments

HiSeq data, after MSA

DePristo et al. *Nat Genet* 2011

Local realignement - principle

1. Find the best alternate consensus sequence that, together with the reference, best fits the reads in a pile (maximum of 1 indel)



2. The score for an alternate consensus is the total sum of the quality scores of mismatching bases

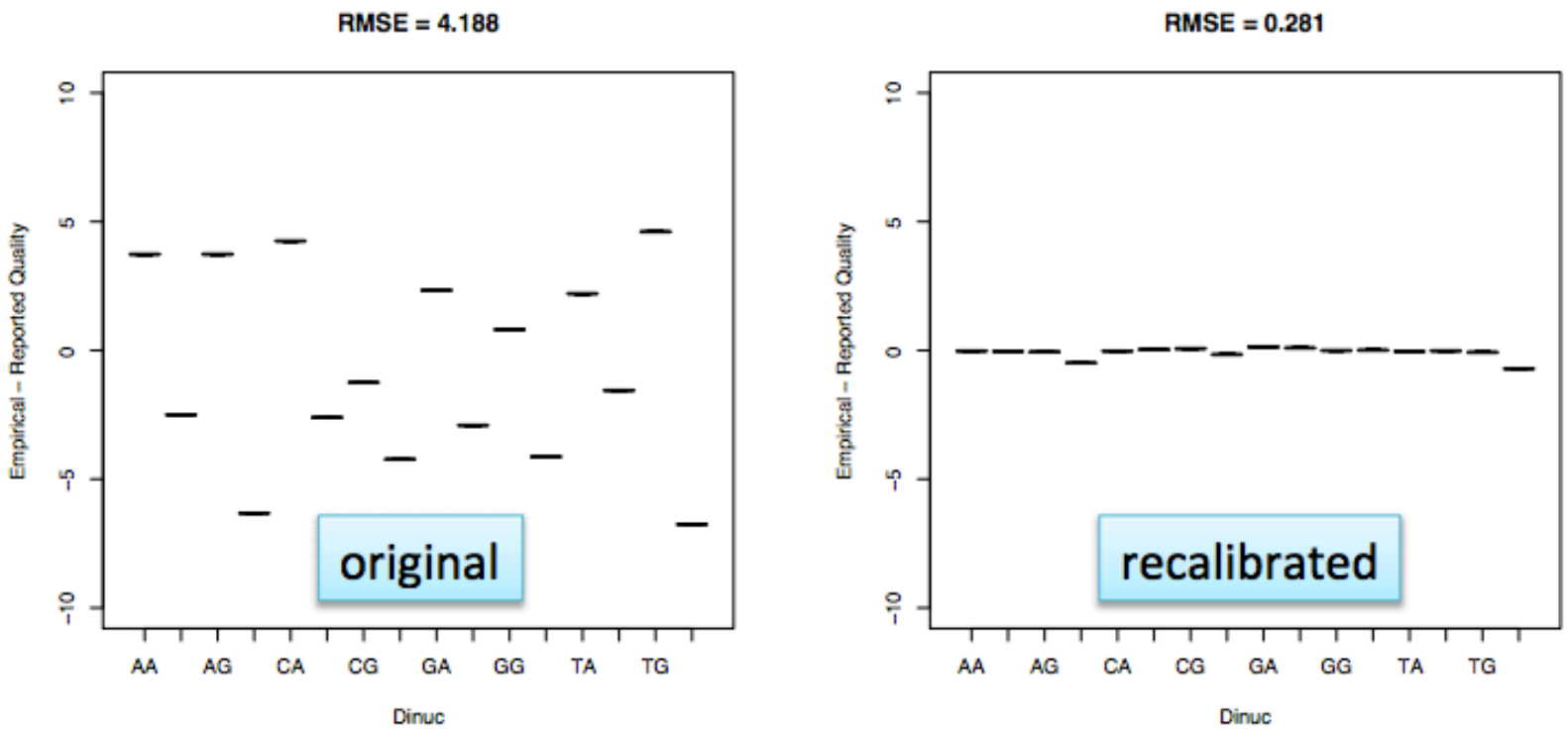
3. If the score of the best alternate consensus is sufficiently better than the original alignments (using a LOD score), then we accept the proposed realignment of the reads

GATK, Broad

Base quality recalibration

- Quality scores are critical for all downstream analysis
- Systematic biases are a major contributor to bad calls

Example of bias: qualities reported depending on nucleotide context

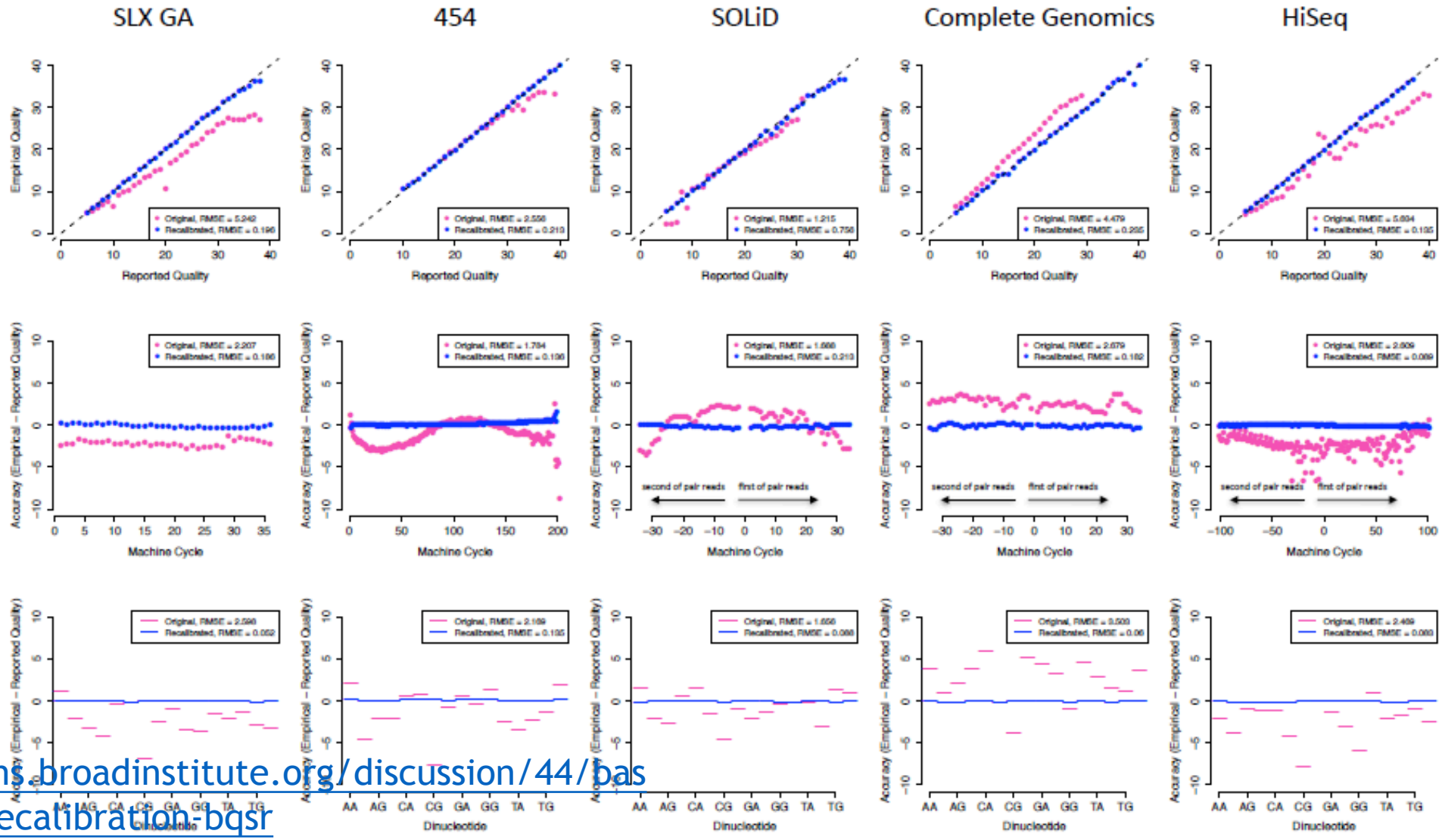


BQSR method identifies bias and applies correction

GATK, Broad

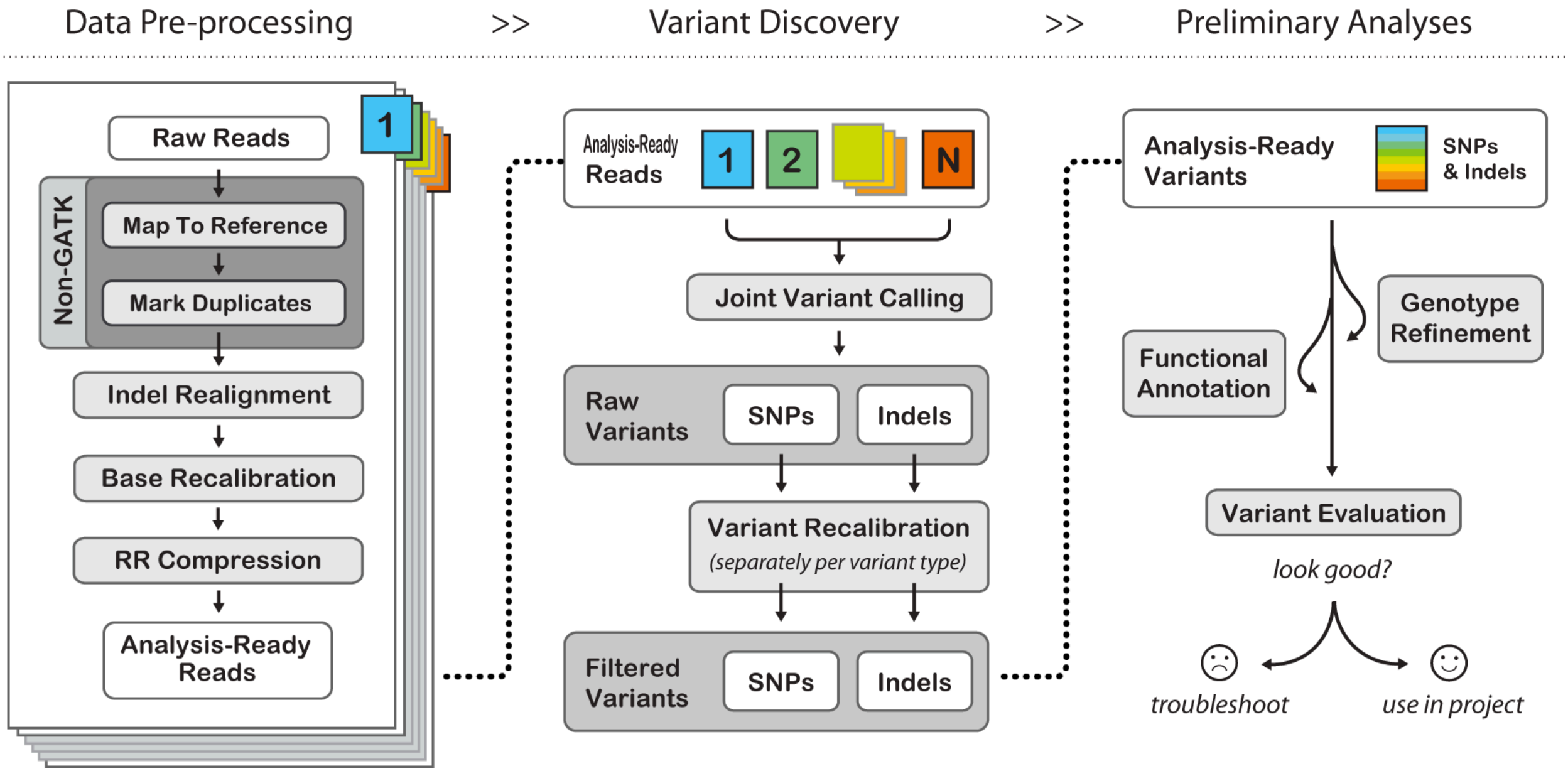
Highlighted as one of the major methodological advances of the 1000 Genomes Pilot Project!

Base Quality Score Recalibration provides a calibrated error model from which to make mutation calls



http://gatkforums.broadinstitute.org/discussion/44/base-quality-score-recalibration-bqsr

Improve beyond analysis-ready reads



Using haplotypes for base calling

- Suppose that only 2 haplotypes have been observed in a population:

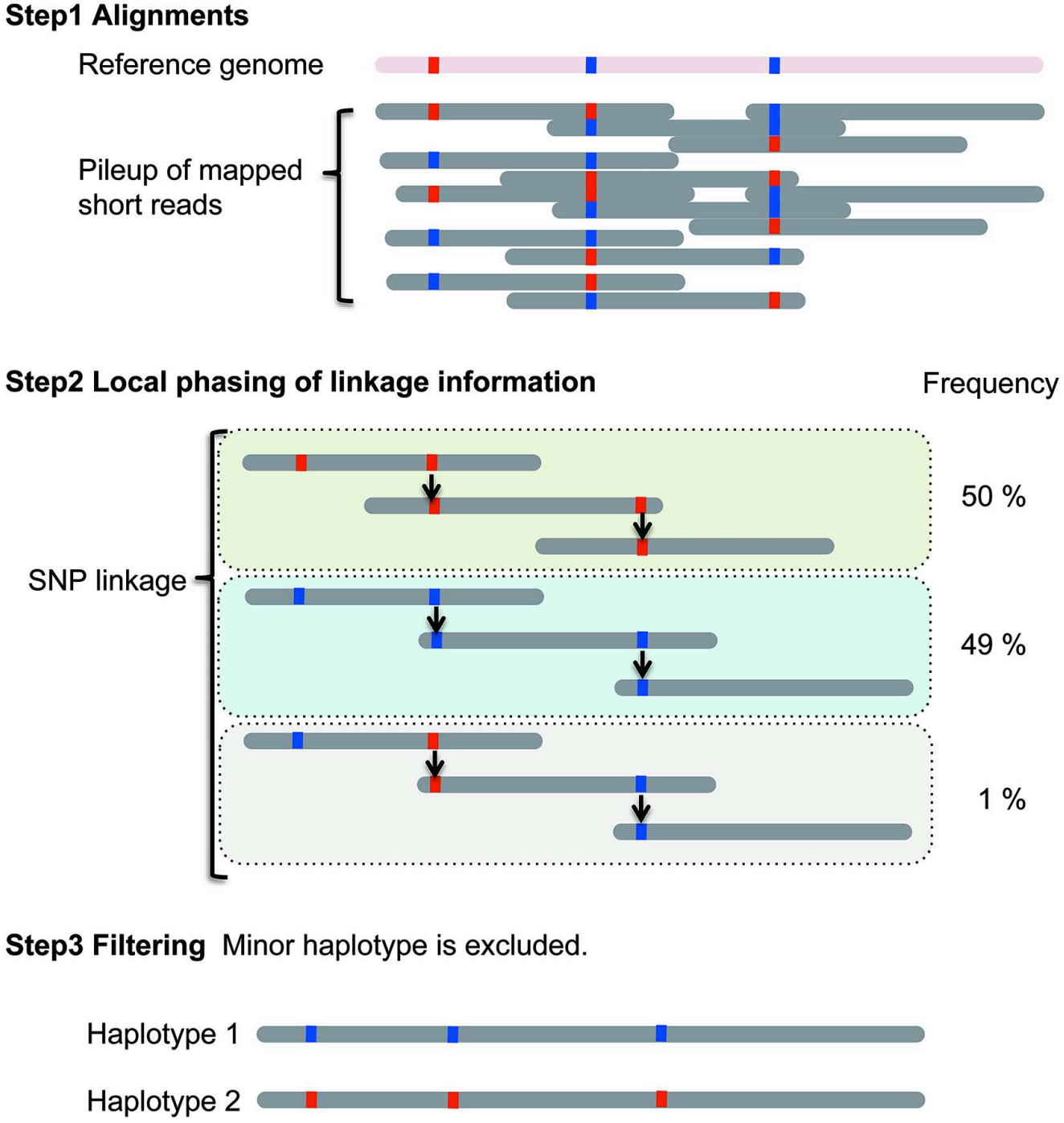
```
Chr1: ..........A....T.......G..........
Chr1: ..........C....G.......A..........
```

- And that you observe the following reads:

```
......A....N.......G..
  ..A....N.......G.....
 ...A....N.......G...
```

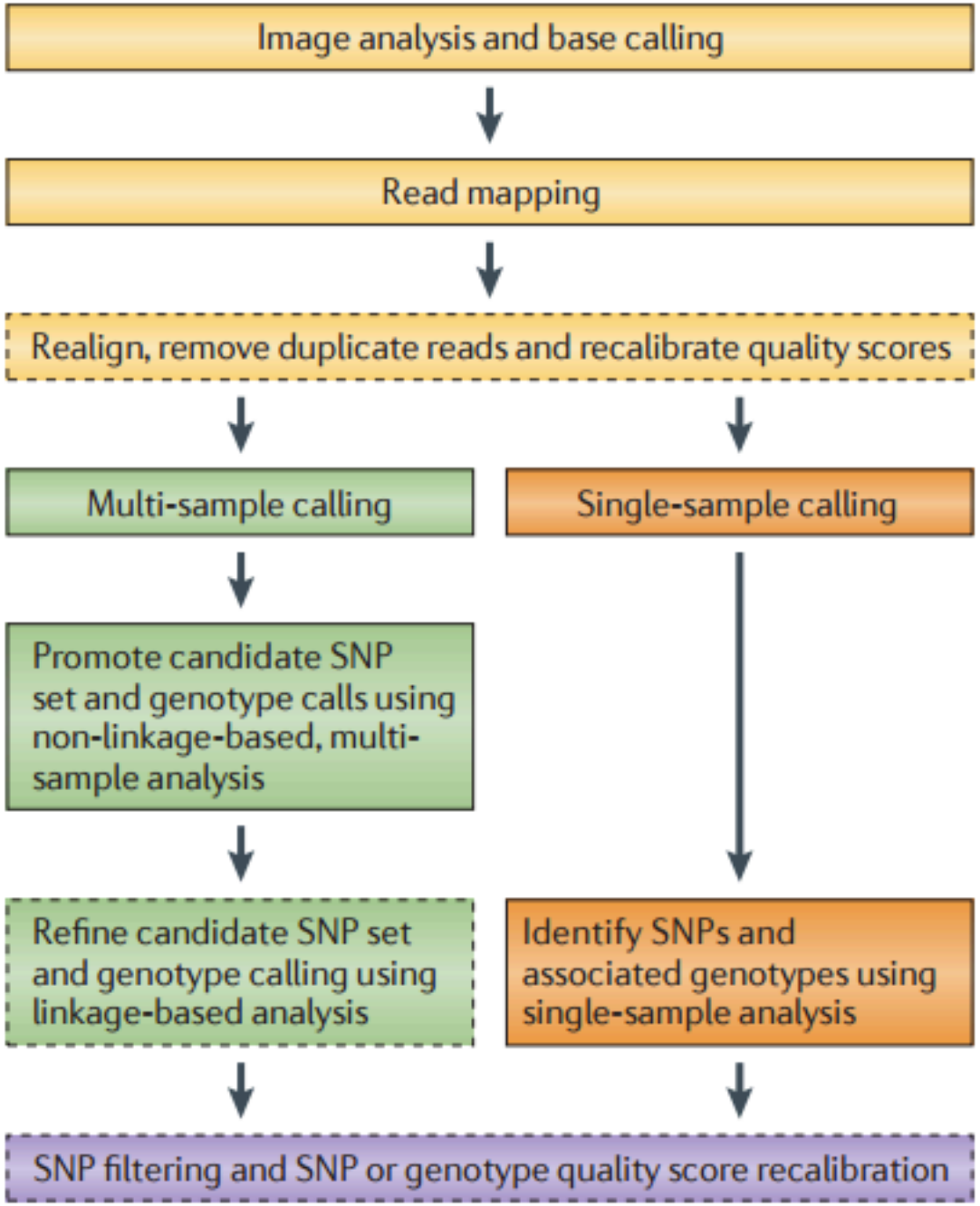
- Can you guess the value of `N` ?

Building haplotypes



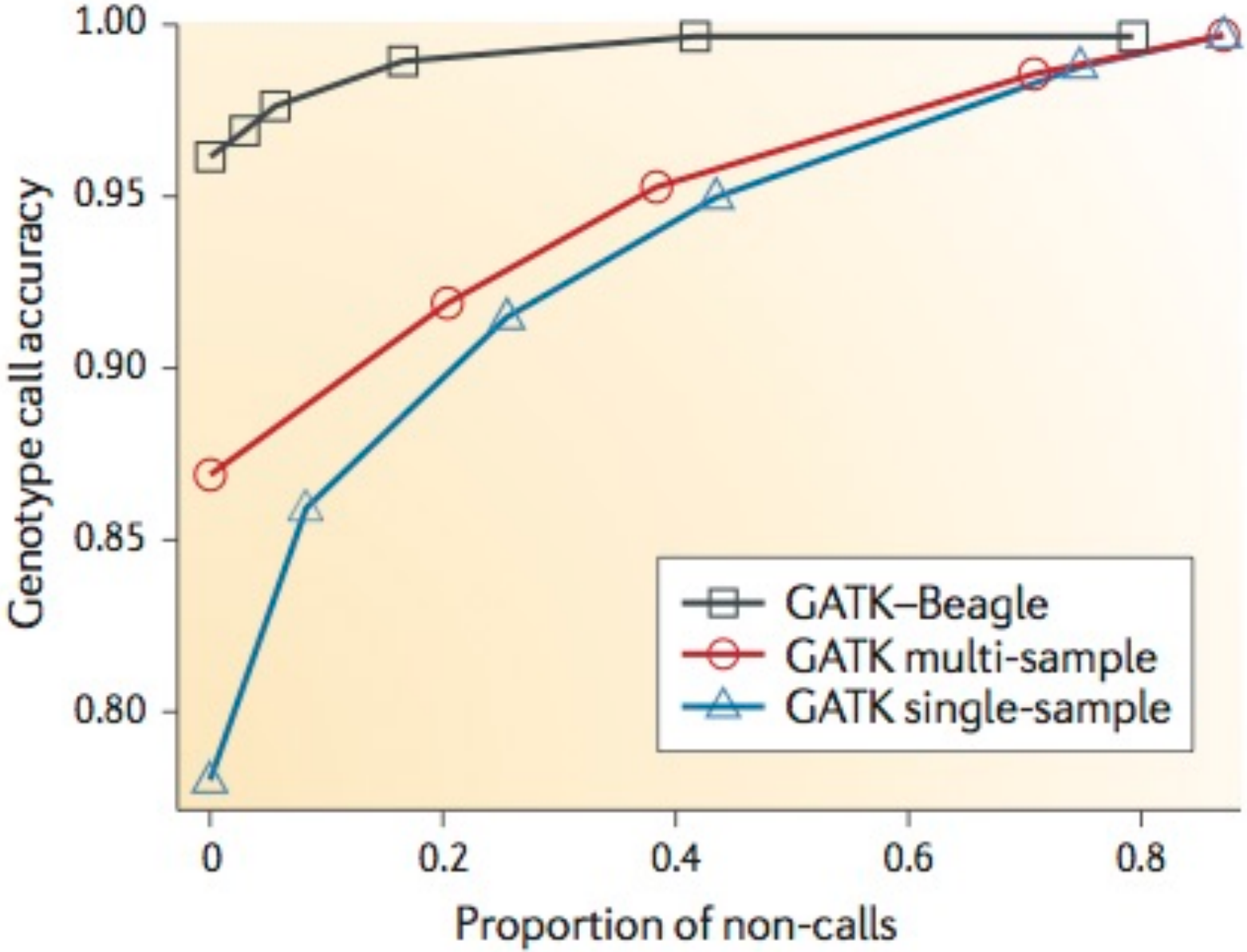
Yoshida et al (2015)

Use multiple samples

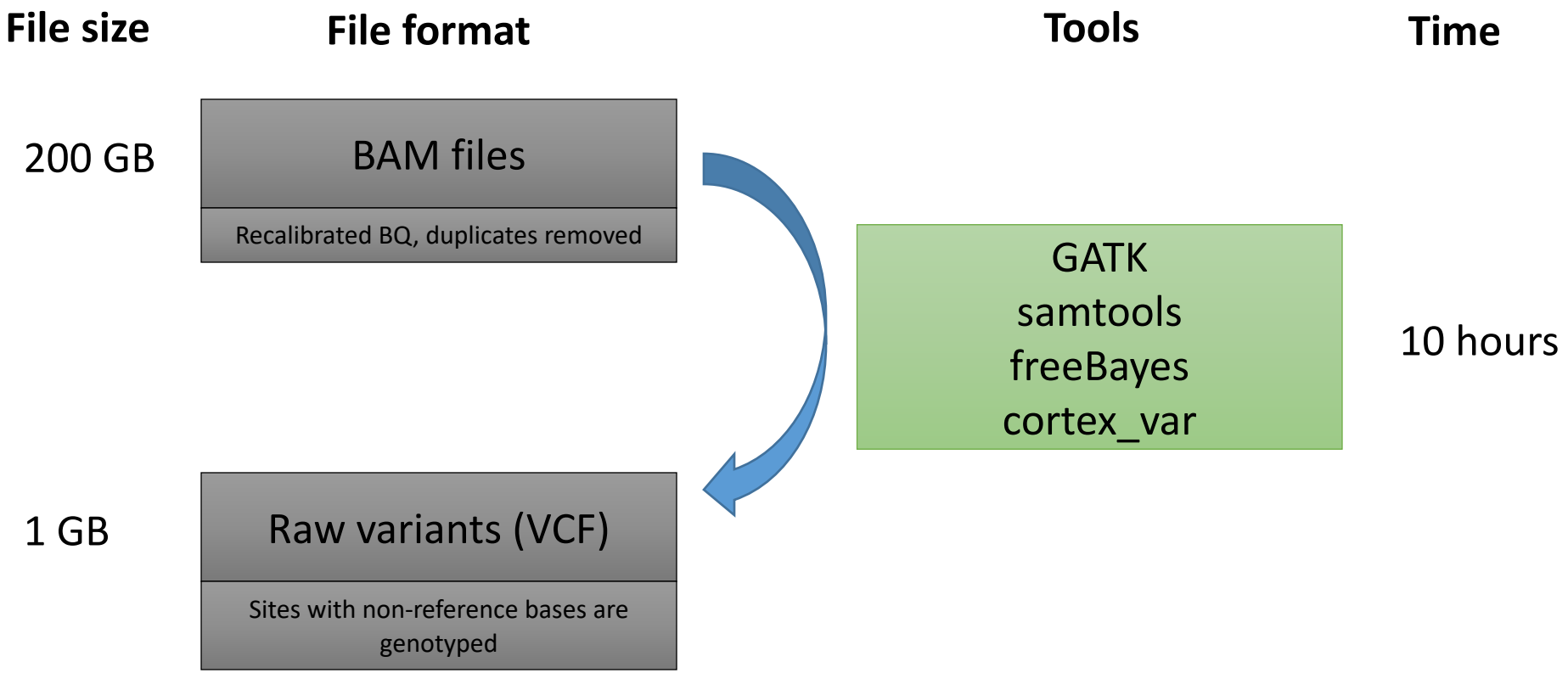


Nielsen et al (2012)

Haplotype imputation increase genotype accuracy



Nielsen et al (2012)



Adapted from Mark DePristo

VCF format

```
##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
```

Mandatory header line

Mandatory header line

Reference base

Alternative base

Quality score

Allele frequency, read depth, etc.

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

Variant filtering

Raw variant calls have a lot of false positives.

- How to filter?
- Which one do you look at first?

Manual filtering based on different parameters

- allele frequency, quality score, depth of coverge…
- Location (contig ends SNPs are usually inaccurate)
- Case by case

look at the strongest effect filter

Annotating variants

- Annotations using reference genomes

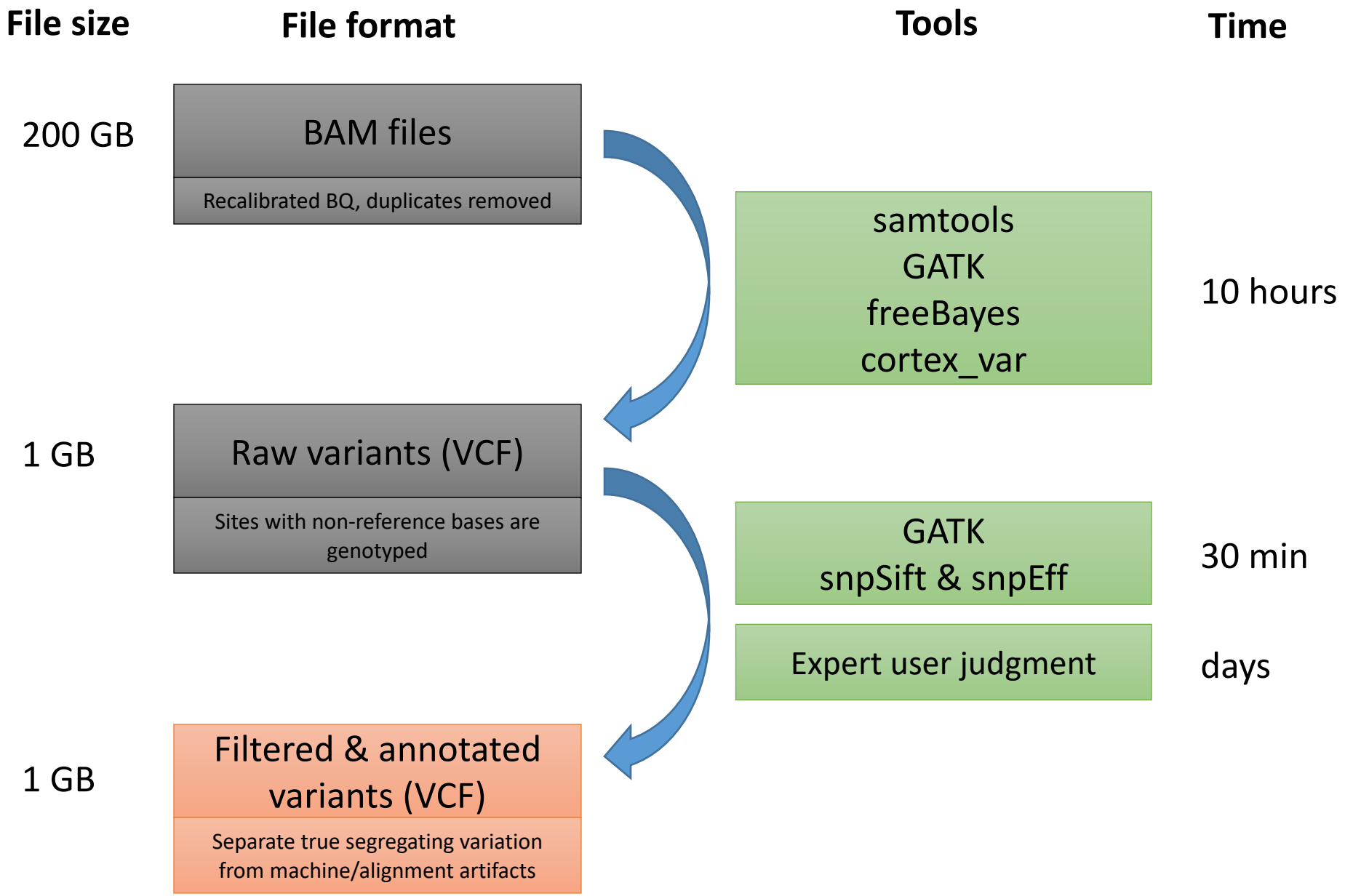
  Programs available: SNP-eff, annovar
- Calculate effects:
  - Coding (e.g. Syn, Non-Syn, Stop gained, Splice)
  - Non-coding (e.g. TFBS)

One of the mostly intensively research areas:

Linking variation to function

Unfortunately, only applicable to humans

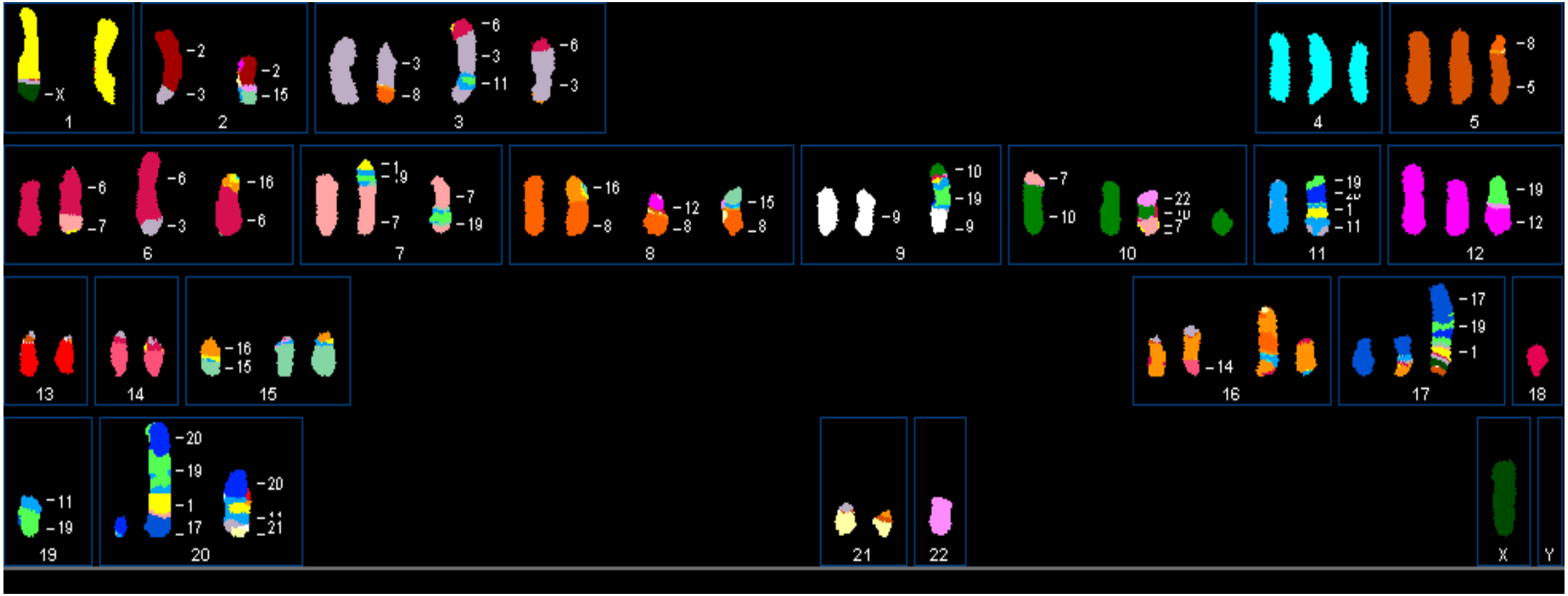
For a new species, you have to start from scratch



Adapted from Mark DePristo

Structural variation (short reads)

More difficult structural variants



Traditional structural variations
Can we see them in higher resolution?

Alkan et al. 2011

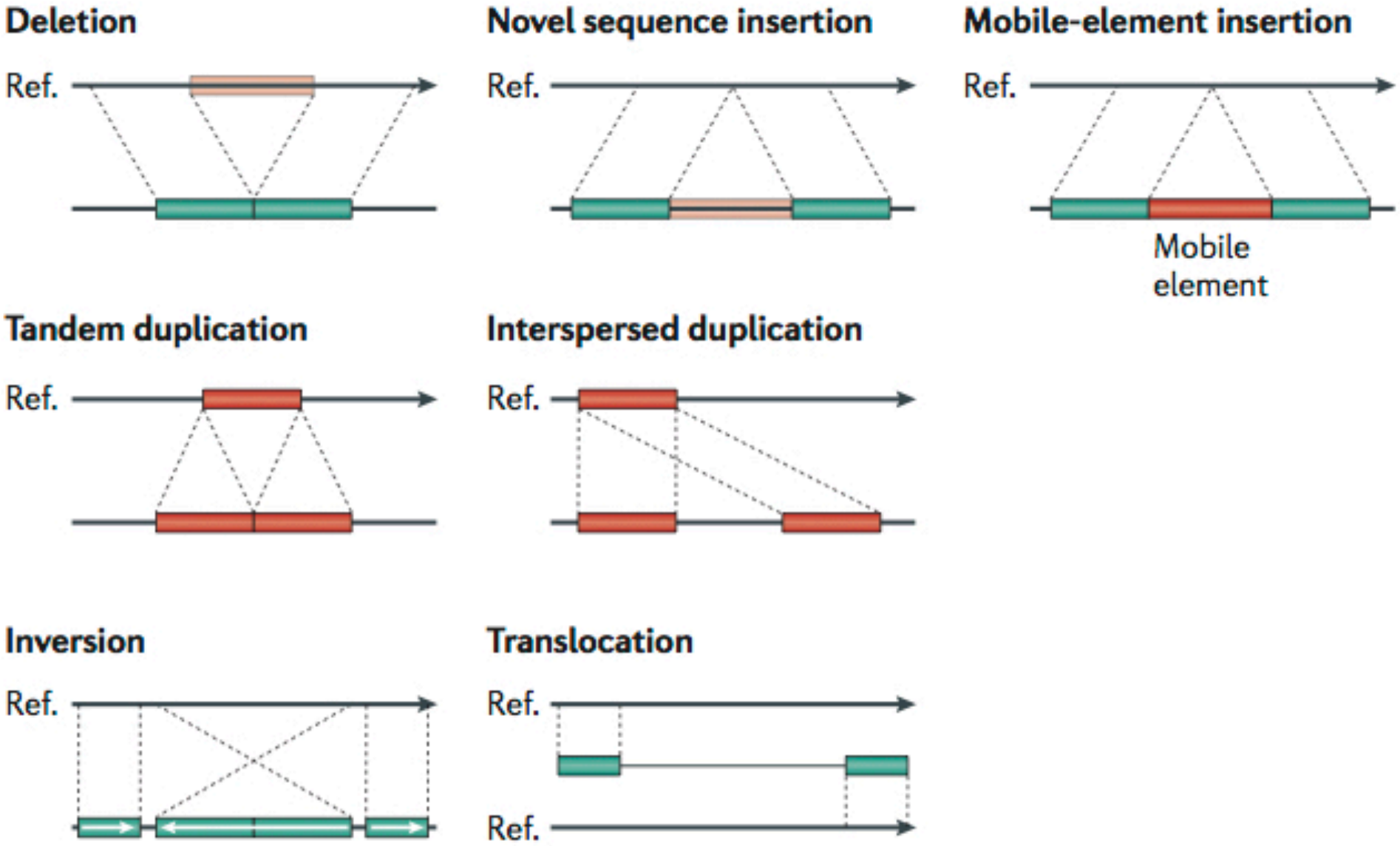
More difficult structural variants

Structural Variants (SVs): Genomic rearrangements that affect >50bp (or 100bp, or 1Kb) of sequence, including:

- deletions
- novel insertions
- inversions
- mobile-element transpositions
- duplications
- translocations

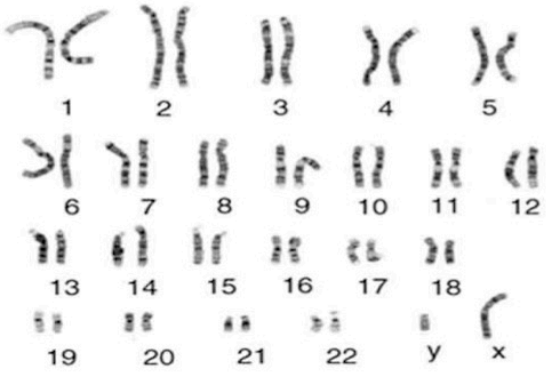
Alkan et al. 2011

SV classes



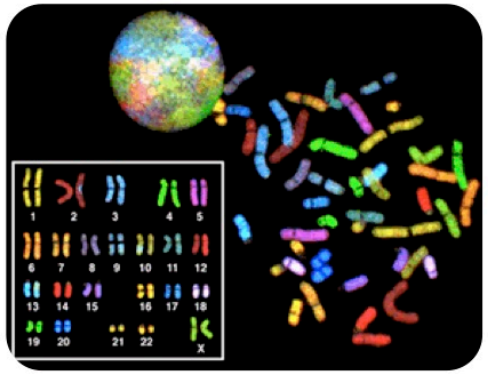
Alkan et al. 2011

Again, our understanding is driven by technology



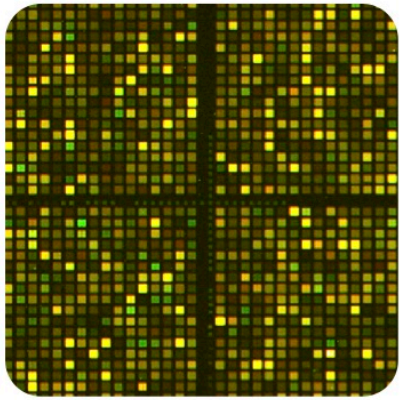
1940s - 1980s

Cytogenetics / Karyotyping



1990s

CGH / FISH /
SKY / COBRA



2000s

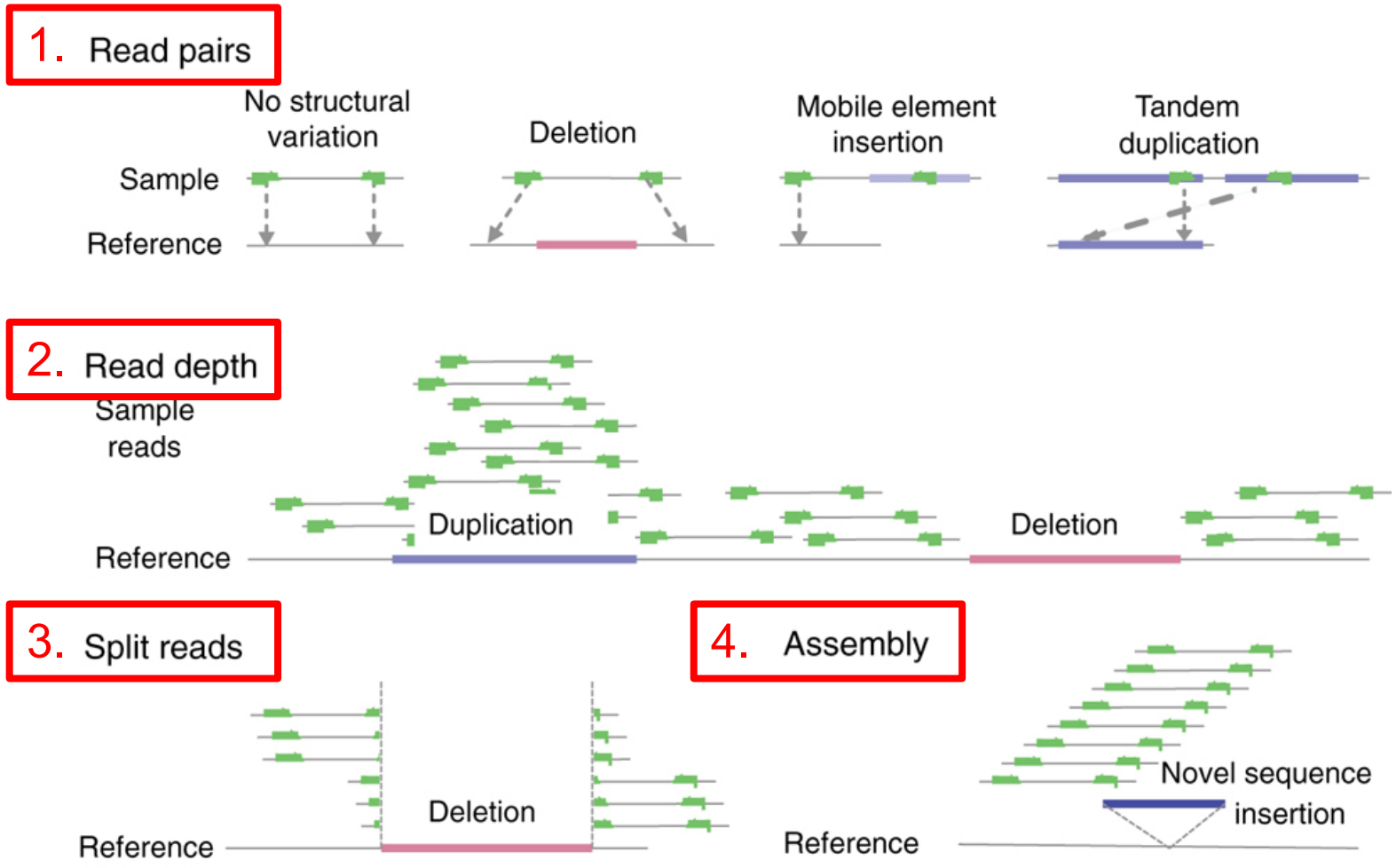
Genomic microarrays
BAC-aCGH / oligo-aCGH

Today

High throughput
DNA sequencing

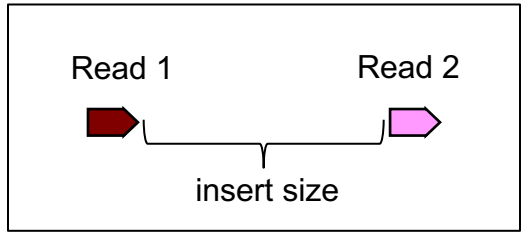
Aaron Quinlan

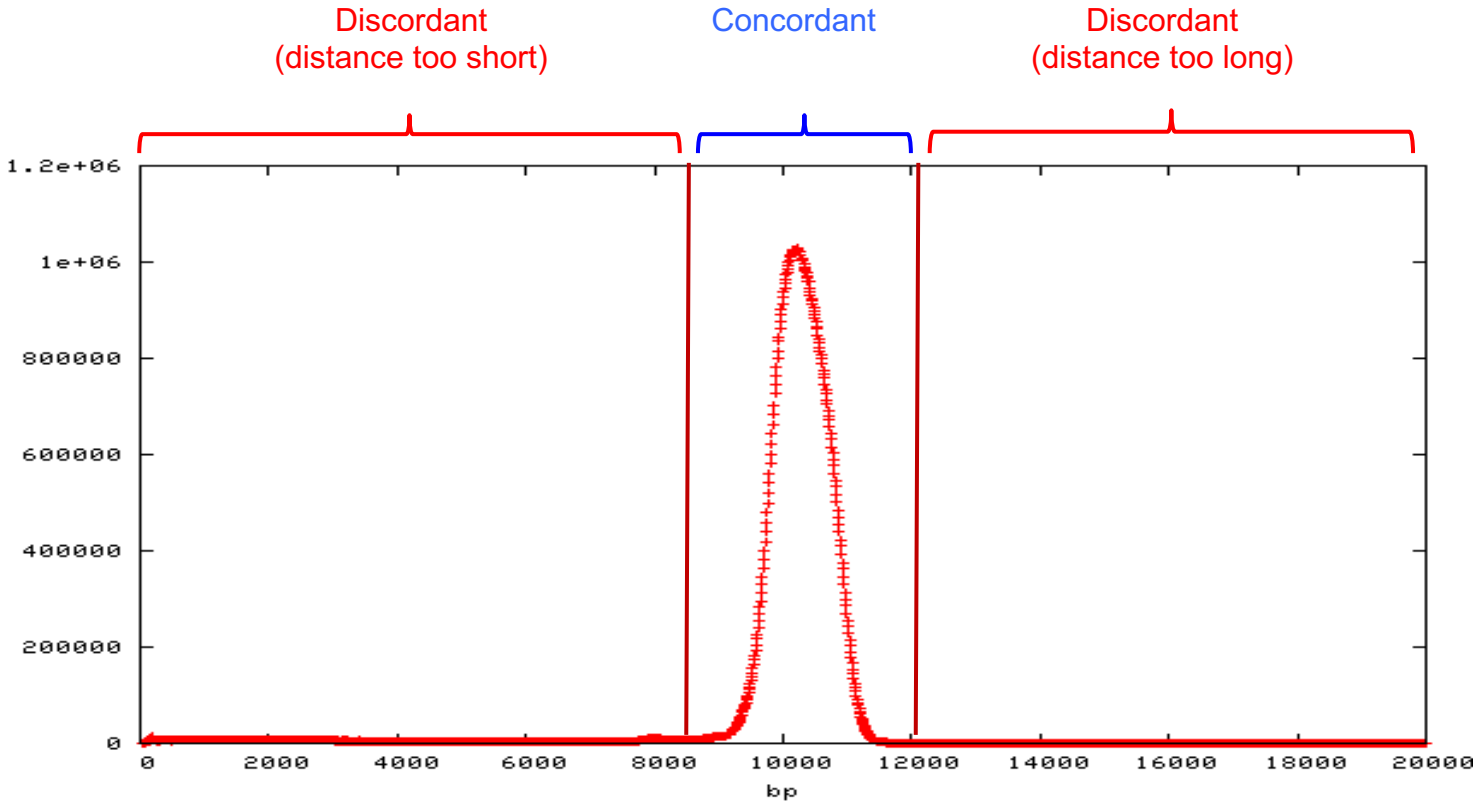
Strategies for calling SVs from NGS data



Baker *Nat Methods* 2012

Discordant read pairs

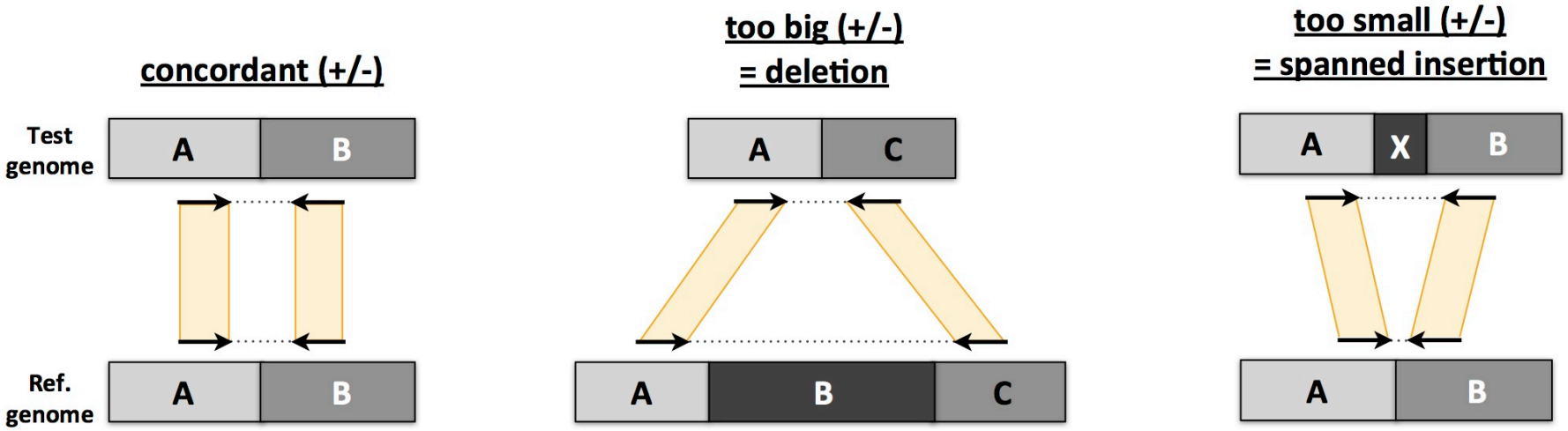


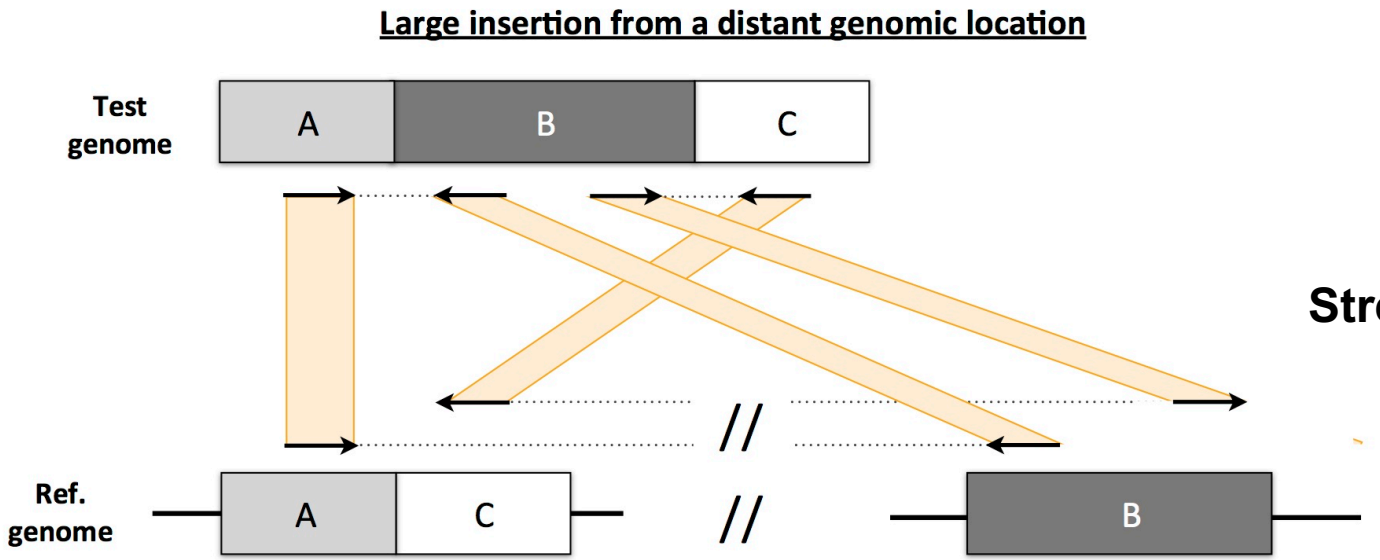


Genomic distance between mapped paired tags

Reads pairs are also Discordant when order or orientation isn't as expected.
Do they fall into particular region of the assembly?

Using discordant reads to detect SVs





Weaknesses

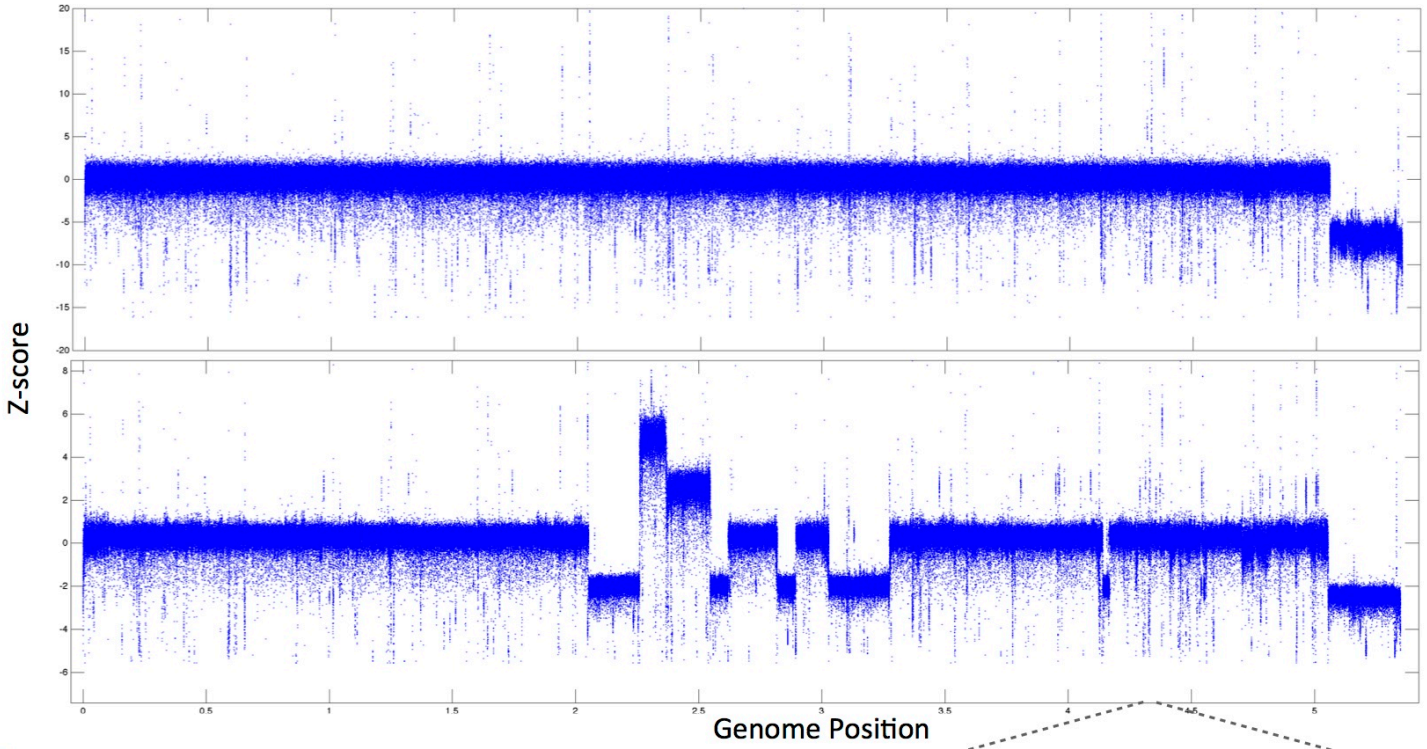
Difficult to interpret read-pairs in repetitive regions
Difficult to fully characterize highly rearranged regions
High rate of false positives

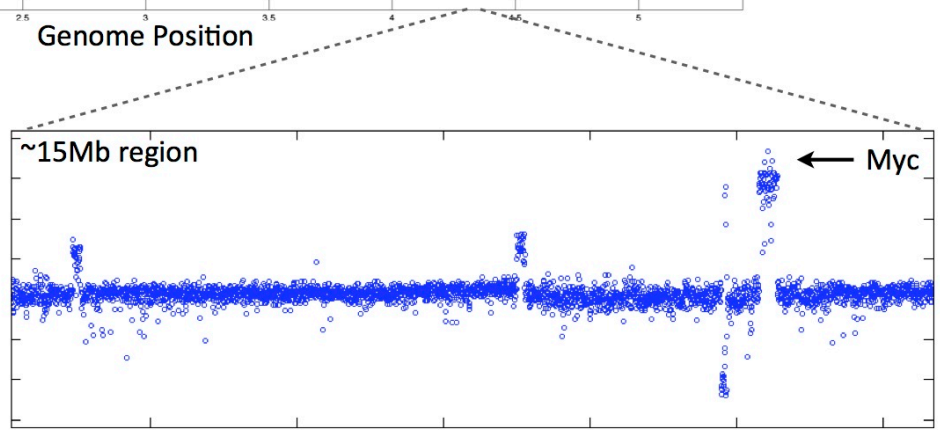
Strengths:

Most classes of variation can, in principle, be detected

Adapted from Aaron Quinlan

Read-depth





Strengths:

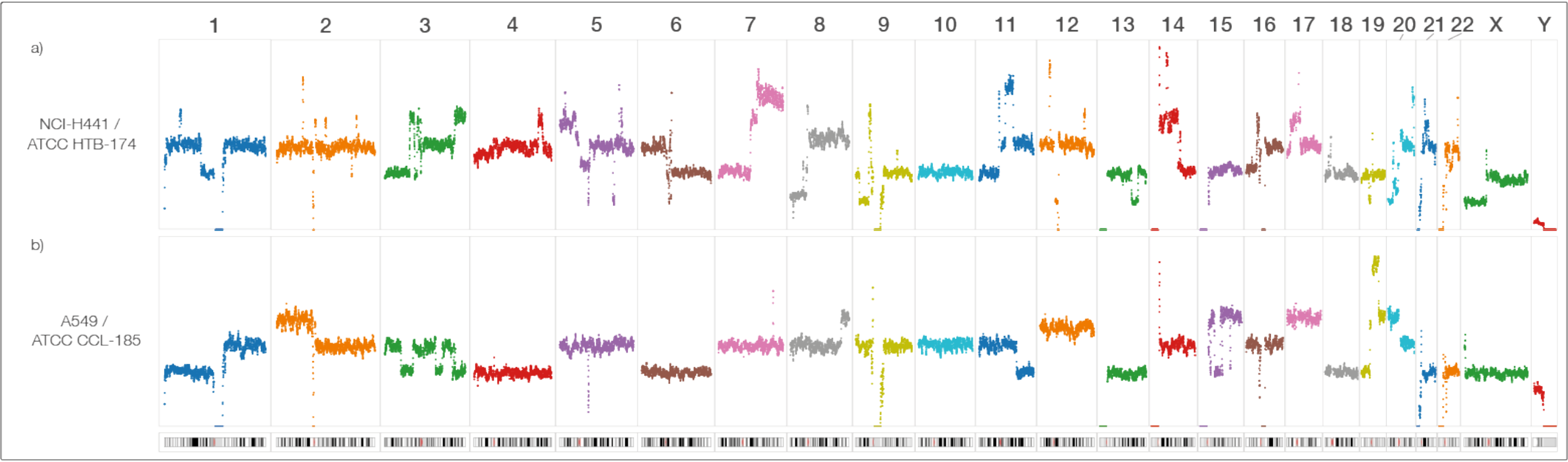
1) Fast and simple.
2) Easy to identify gene amplifications.
3) Relatively straightforward interpretation: is gene X amplified or deleted?

Weaknesses:

1) Limited resolution (5-10kb) = imprecise boundaries
2) Cannot detect balanced events or reveal variant architecture.

Aaron Quinlan

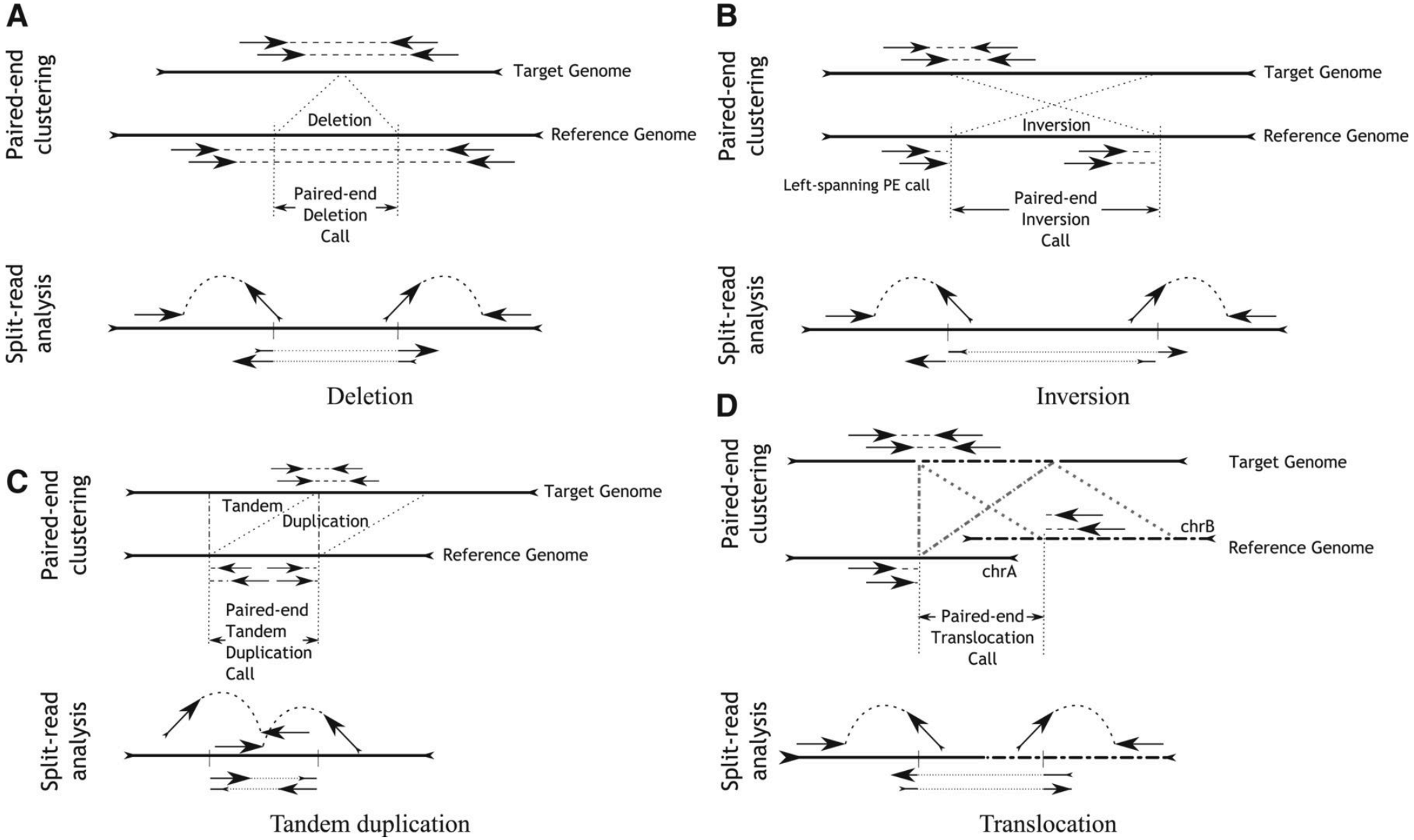
Read-depth can be used to call aneuploidies



Whole-genome sequencing of two lung cancer cell lines. Each has a different pattern of duplications, deletions and translocations a) cell line H441 b) cell line A549

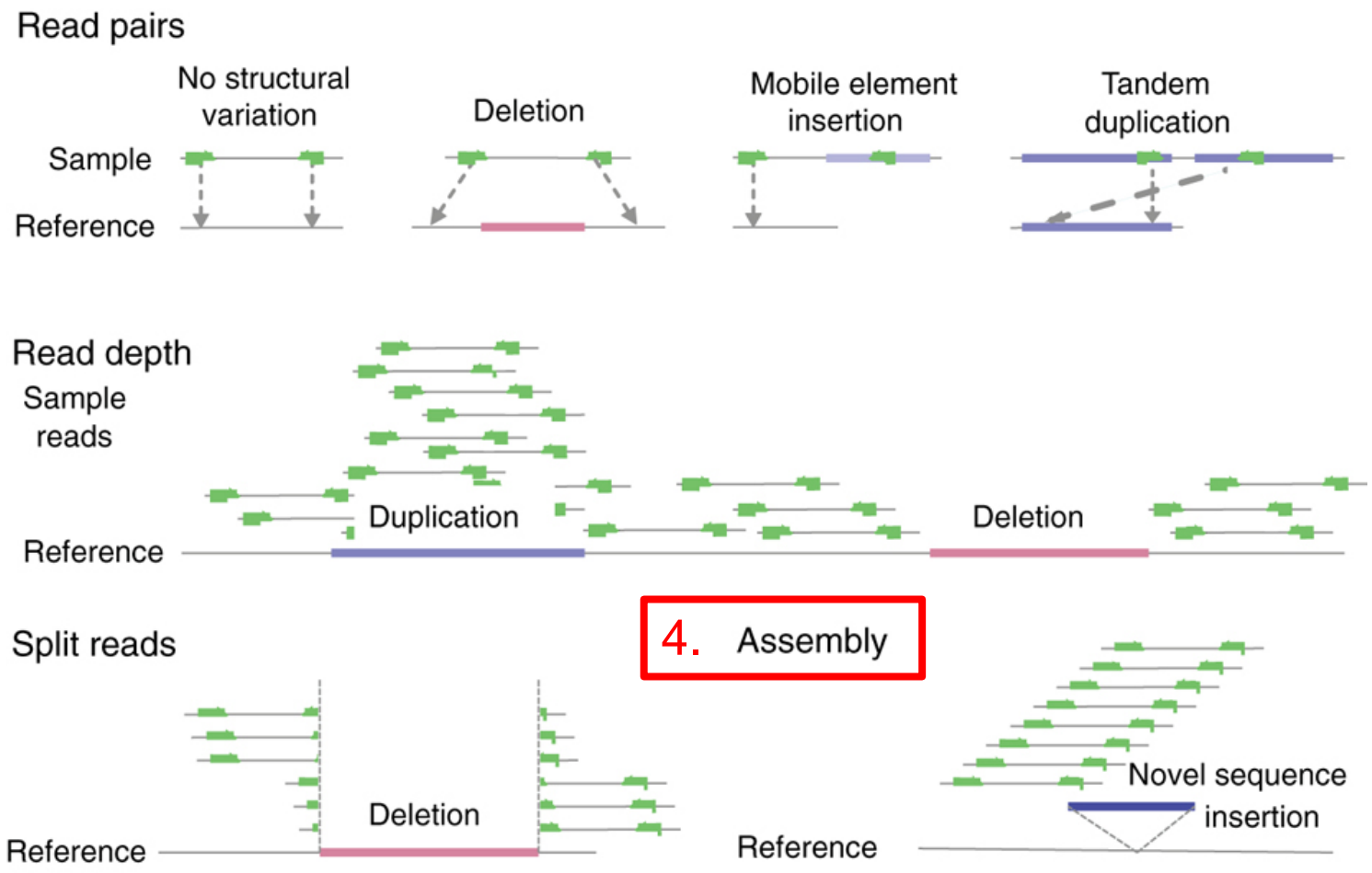
https://nanoporetech.com/

Split reads



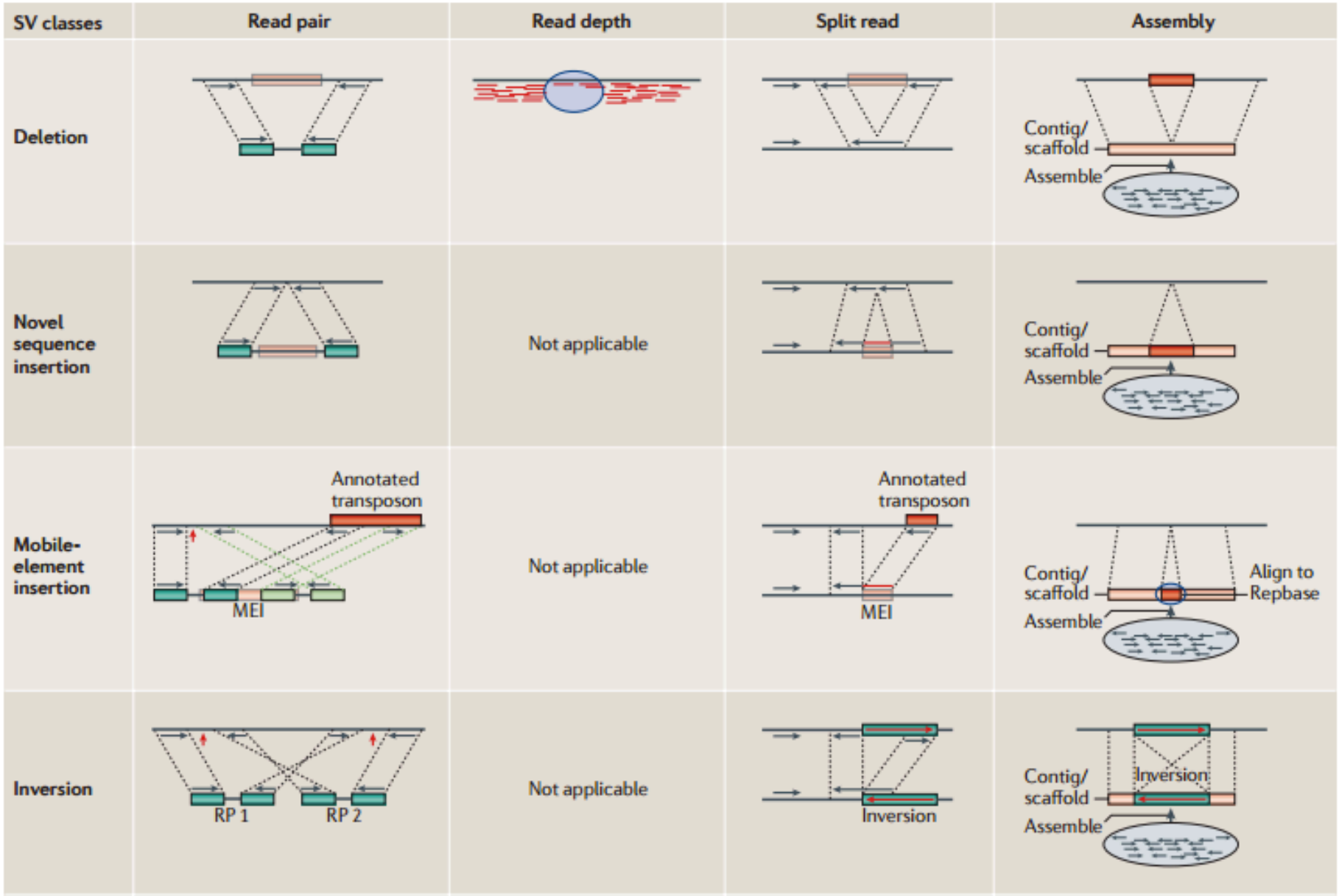
Rausch et al. *Bioinformatics* 2012

Strategies for calling SVs from NGS data



Baker *Nat Methods* 2012

De novo assembly for SVs

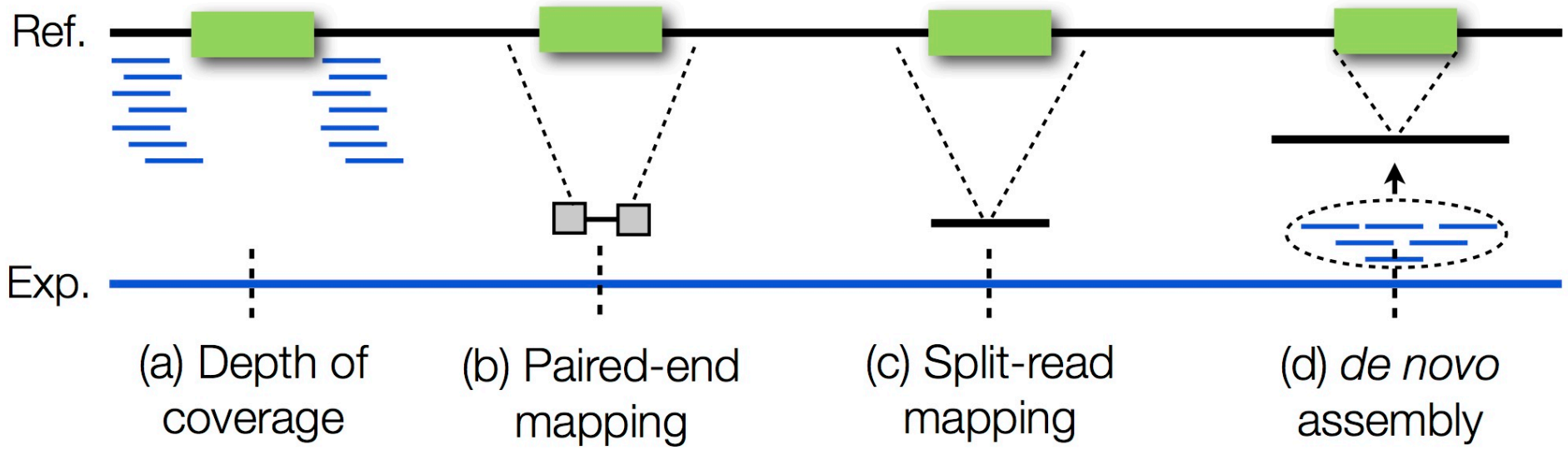


Alkan et al. 2011

Summary of strategies for calling SVs (short reads)

1. Align DNA sequences from sample to human reference genome

2. Look for evidence of structural differences

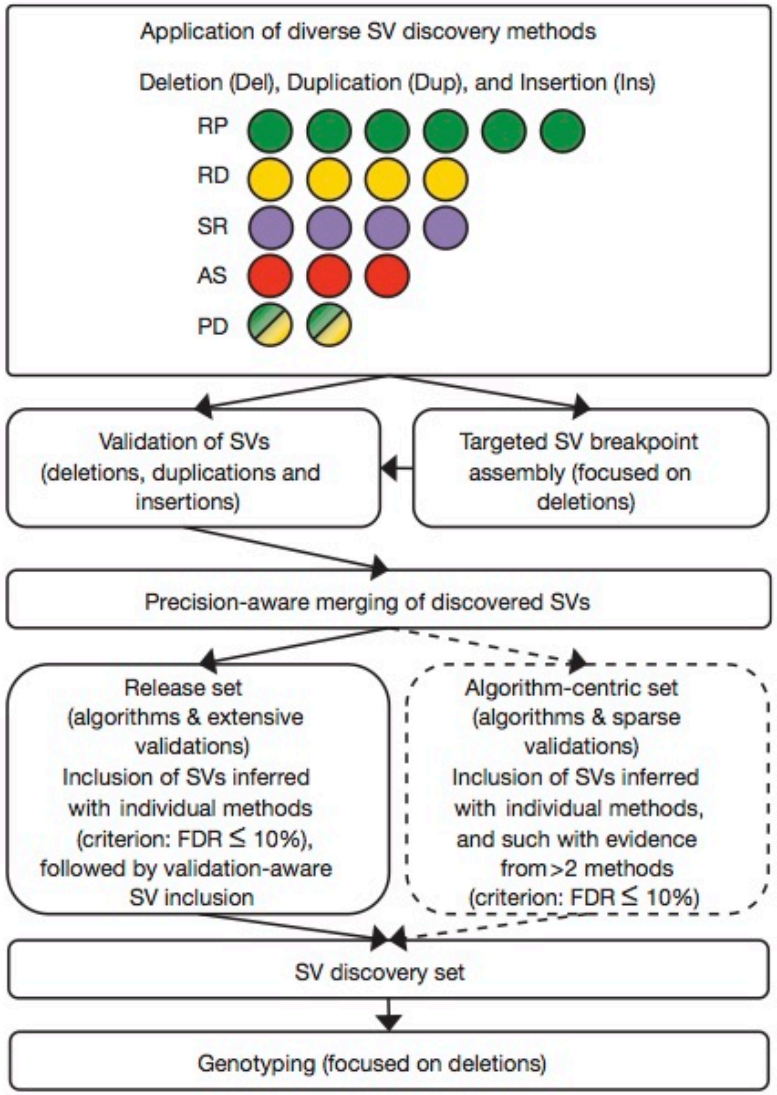


Low **Resolution** High

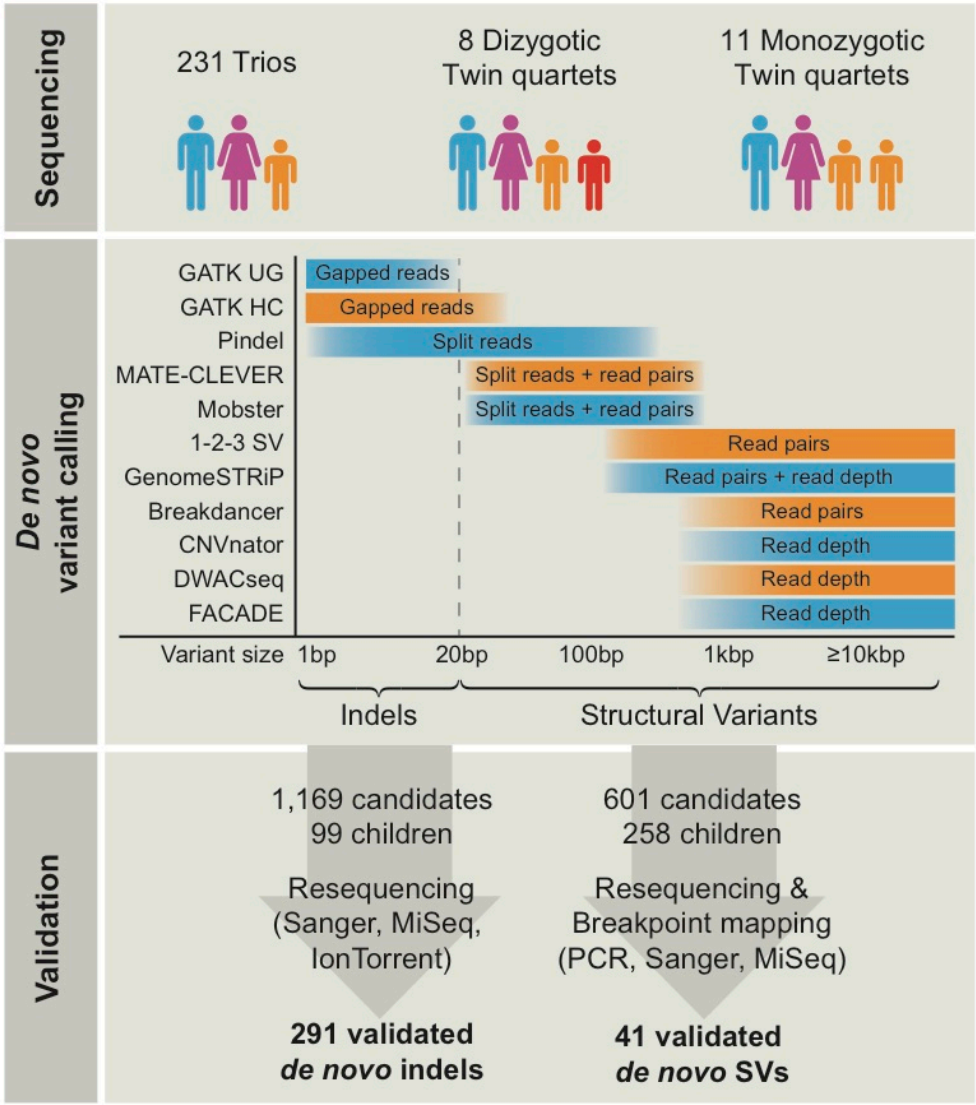
Low **Difficulty / Cost** High

Aaron Quinlan

Bottom line for short reads calling SVs : try many methods and validate

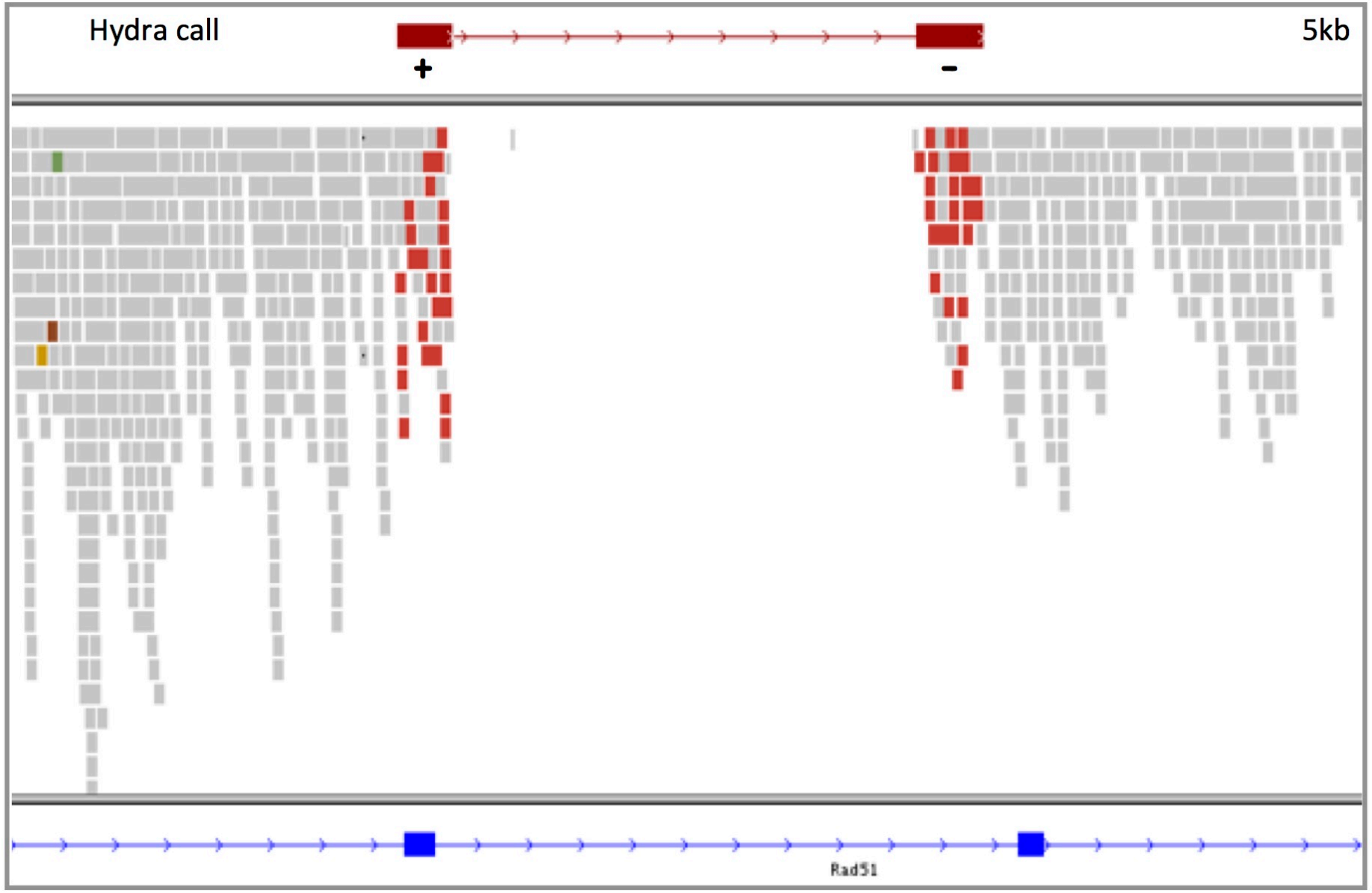


Mills et al. *Nature* 2011



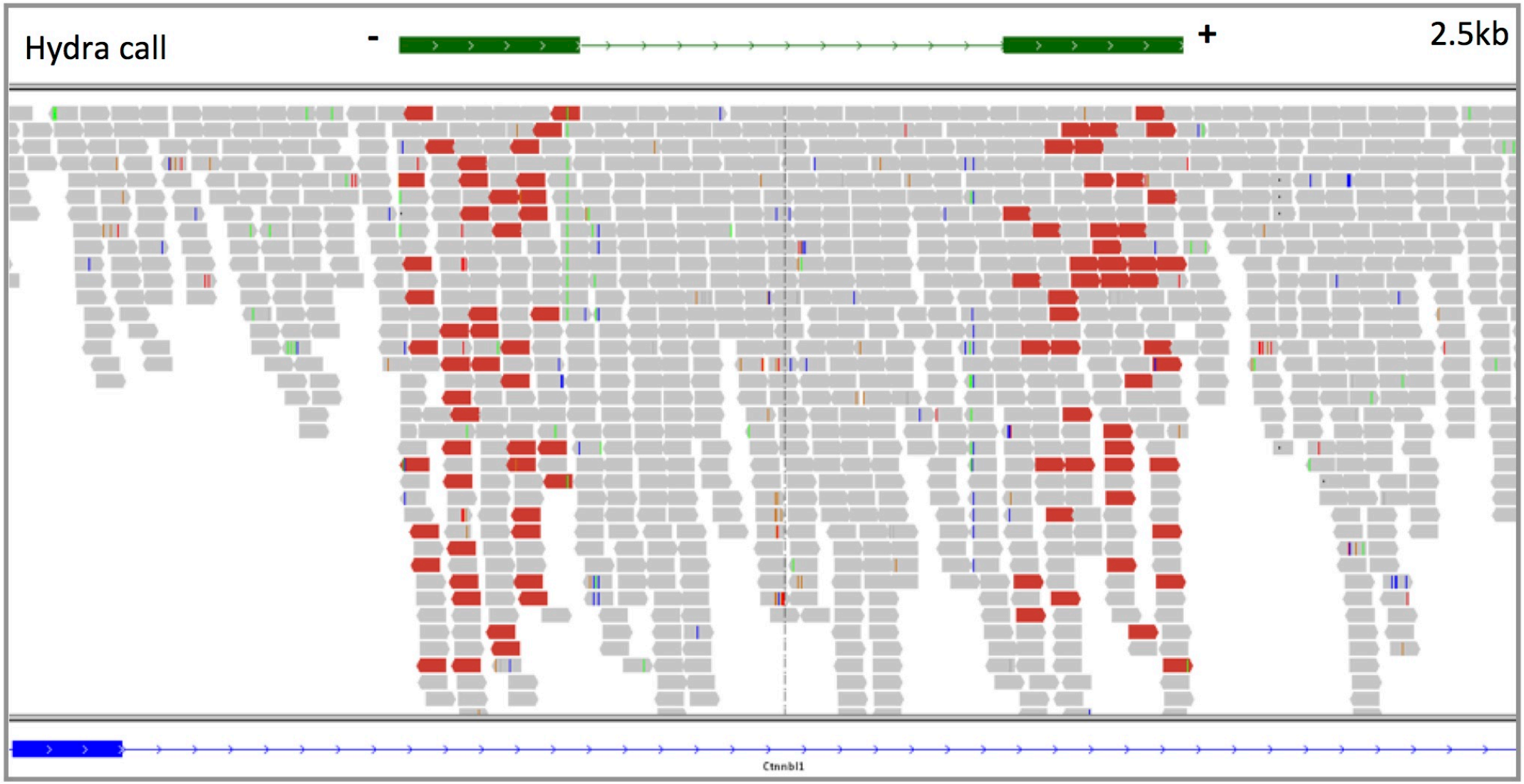
Kloosterman et al. 2015

Visual validation: a deletion



Aaron Quinlan

Visual validation: a duplication



Aaron Quinlan

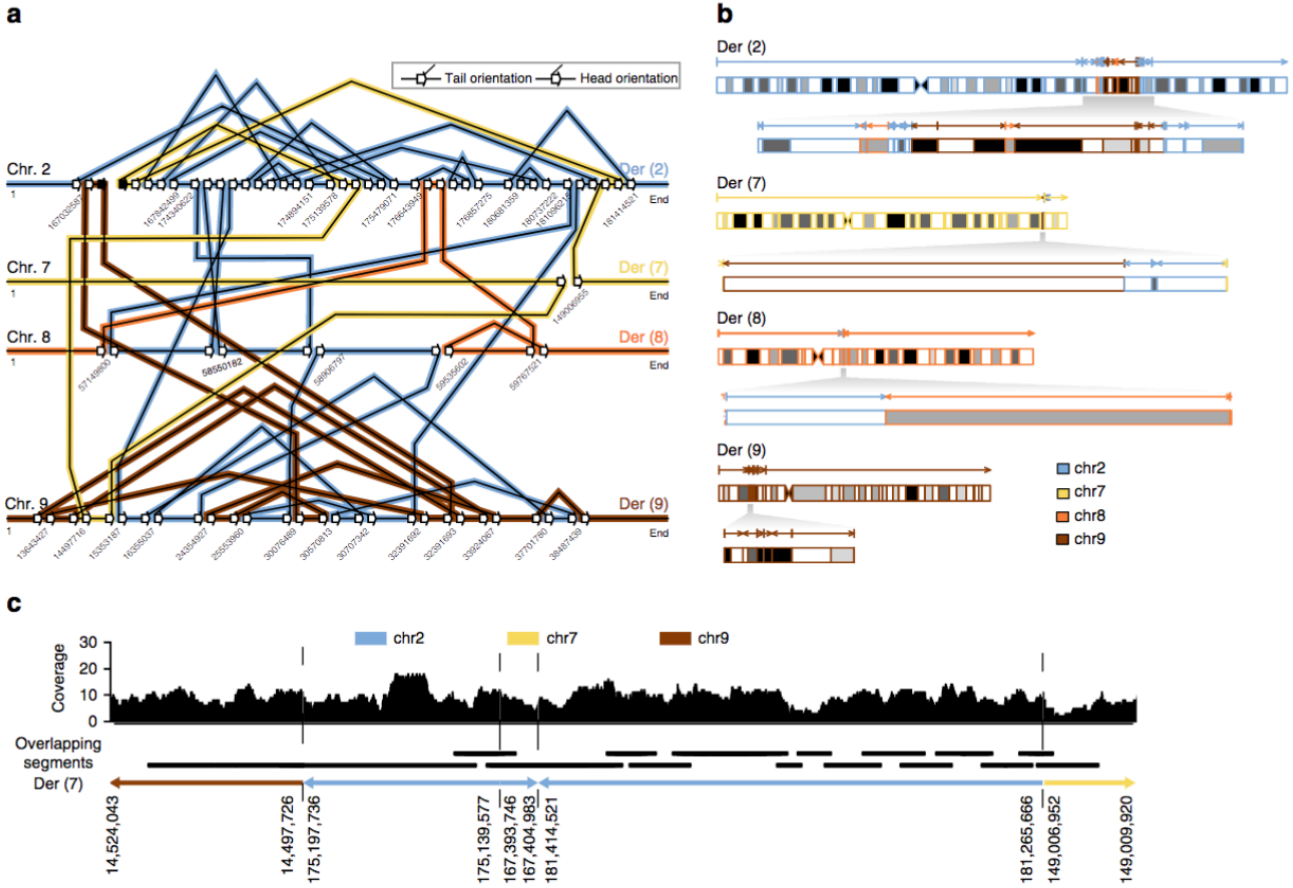
ARTICLE

DOI: 10.1038/s41467-017-01343-4 OPEN

Mapping and phasing of structural variation in patient genomes using nanopore sequencing

Mircea Cretu Stancu[1], Markus J. van Roosmalen[1], Ivo Renkens[1], Marleen M. Nieboer[1], Sjors Middelkamp[1], Joep de Ligt[1], Giulia Pregno[2], Daniela Giachino[2], Giorgia Mandrile[2], Jose Espejo Valle-Inclan[1], Jerome Korzelius[1], Ewart de Bruijn[1], Edwin Cuppen[3], Michael E. Talkowski[4,5,6], Tobias Marschall[7,8], Jeroen de Ridder[1] & Wigard P. Kloosterman[1]

- long reads are superior to short reads with regard to detection of de novo chromothripsis rearrangements.
- long reads also enable efficient phasing of genetic variations, which we leveraged to determine the parental origin of all de novo chromothripsis breakpoints and to resolve the structure of these complex rearrangements.



Structural variants: A summary

Actually it’s all the same methods

Reference assembly -> check depth -> detect duplication

New assembly -> check depth -> detect ploidy chromosomes / mis-assemblies

Current: SV should be called using long reads

Other experiments requiring mapping

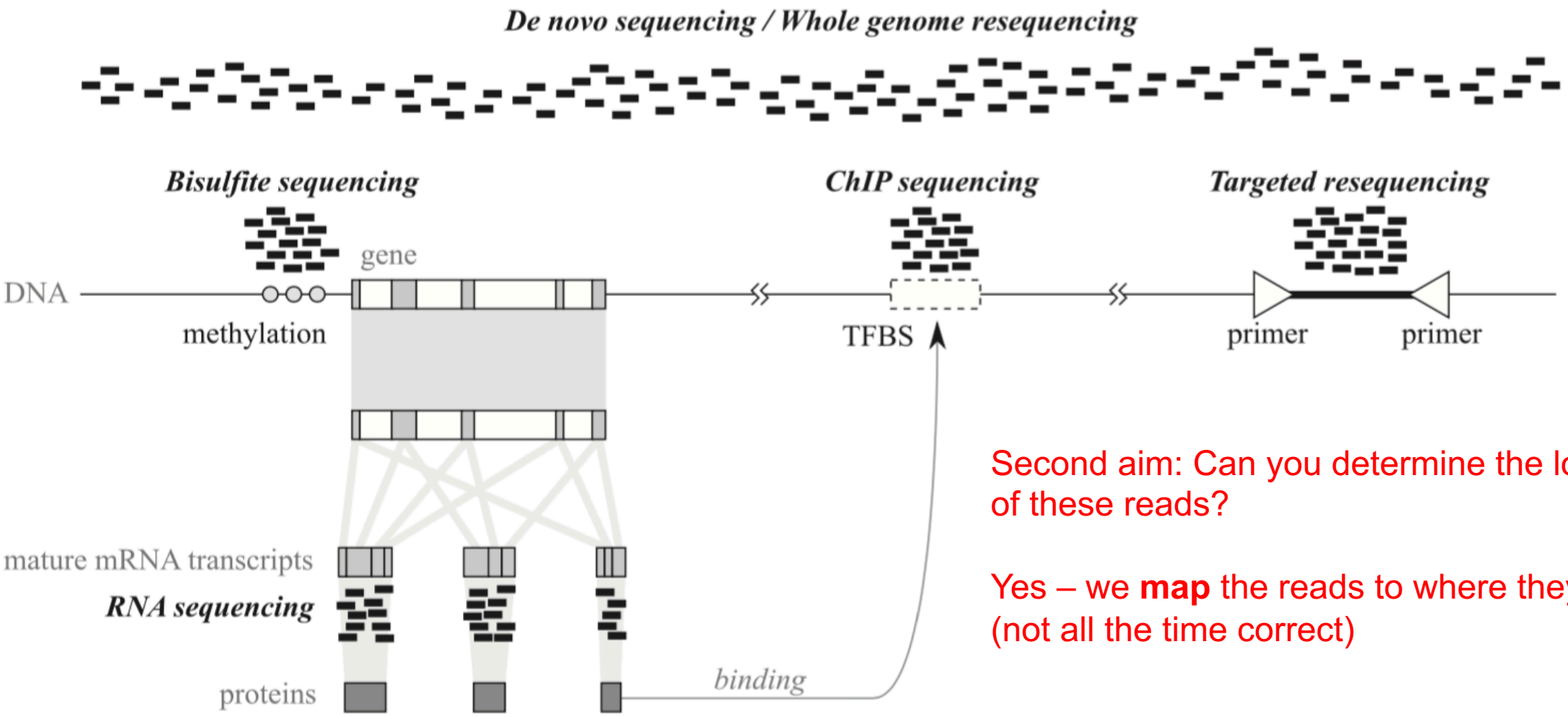


Figure 1.2 A schematic summary of high-throughput sequencing applications. Details are described in Section 1.3.

Second aim: Can you determine the location of these reads?

Yes – we **map** the reads to where they belong (not all the time correct)

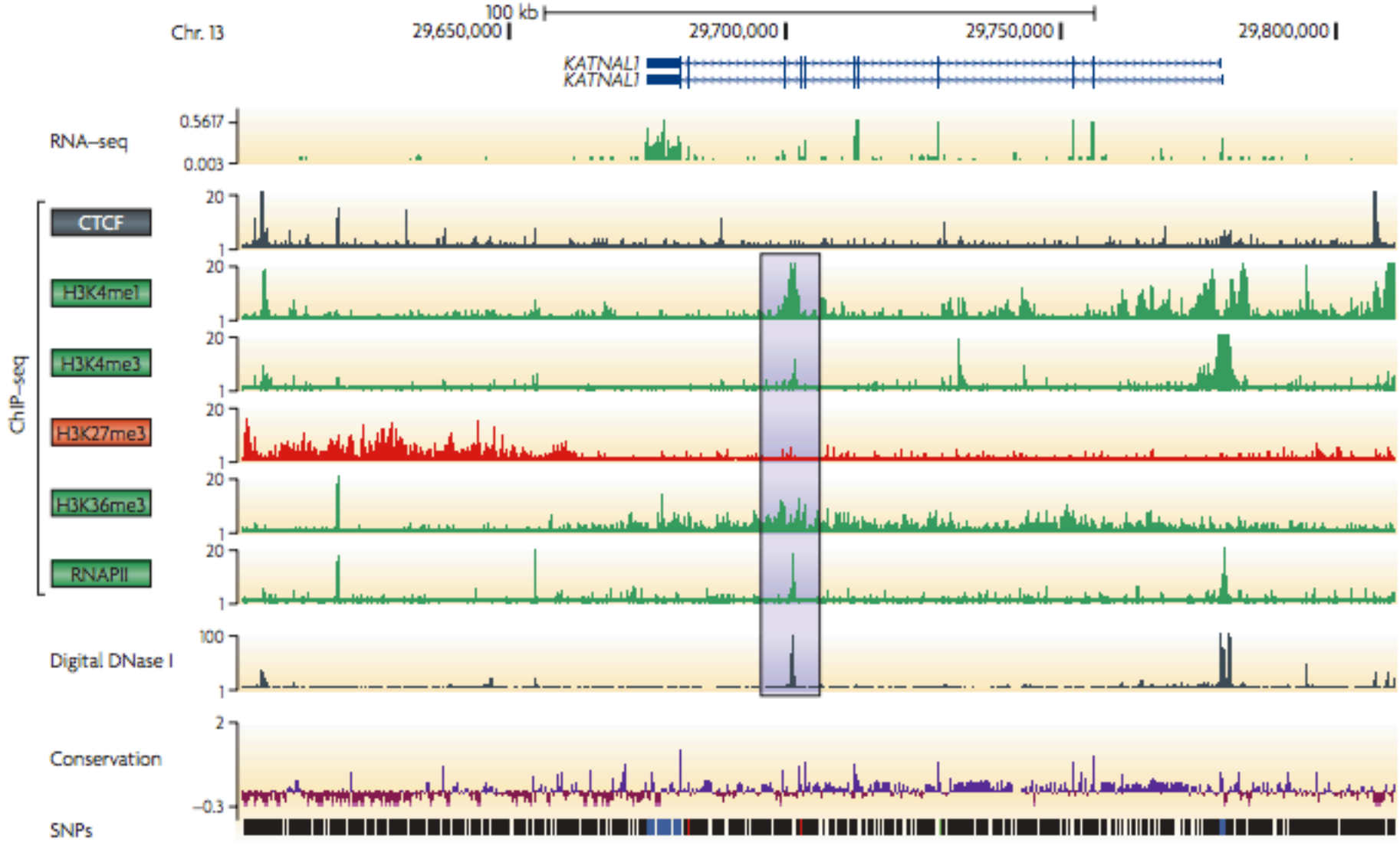
Makinen *et al*

Other experiments requiring mapping

*-seq methodologies

Identify peaks!

How is peak different to coverage?

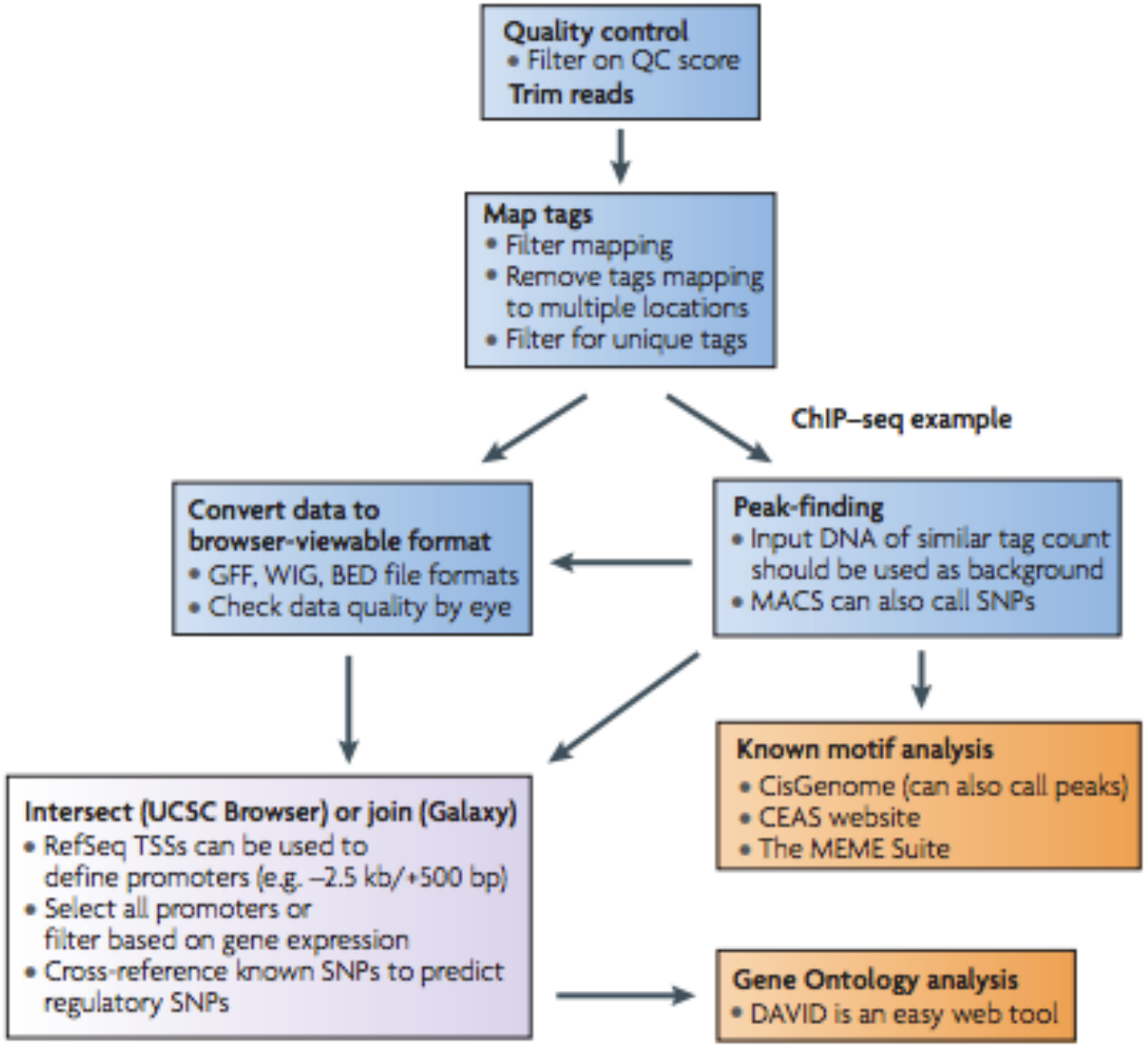


Hawkins et al 2010

Other experiments requiring mapping

Similar methods

Different analysis



Hawkins et al 2010

Validation and standardisation

Genome in a Bottle Consortium

The Genome in a Bottle Consortium is a public-private-academic consortium hosted by NIST to develop the technical infrastructure (reference standards, reference methods, and reference data) to enable translation of whole human genome sequencing to clinical practice.

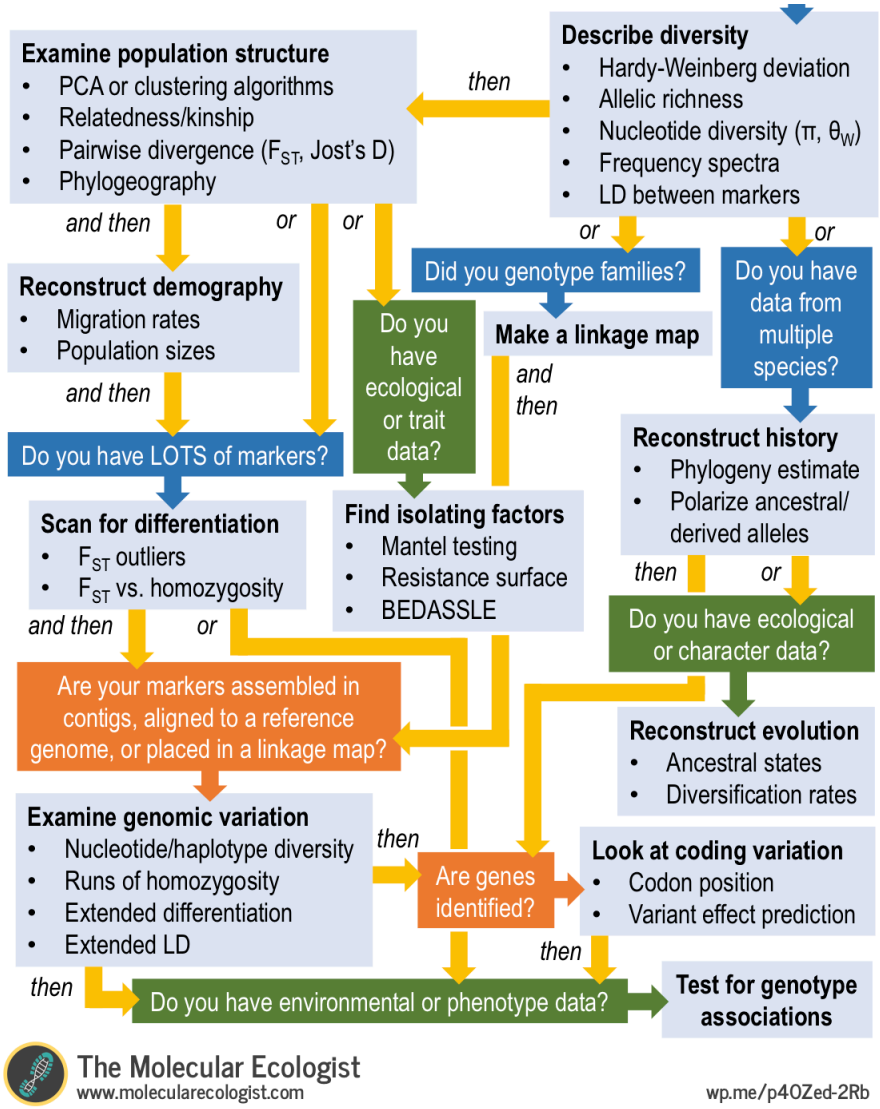
NA12878 cell line, sequenced many platforms, read lengths and sample preps ; A lot and lot of **Benchmarks**

https://sites.stanford.edu/abms/giab

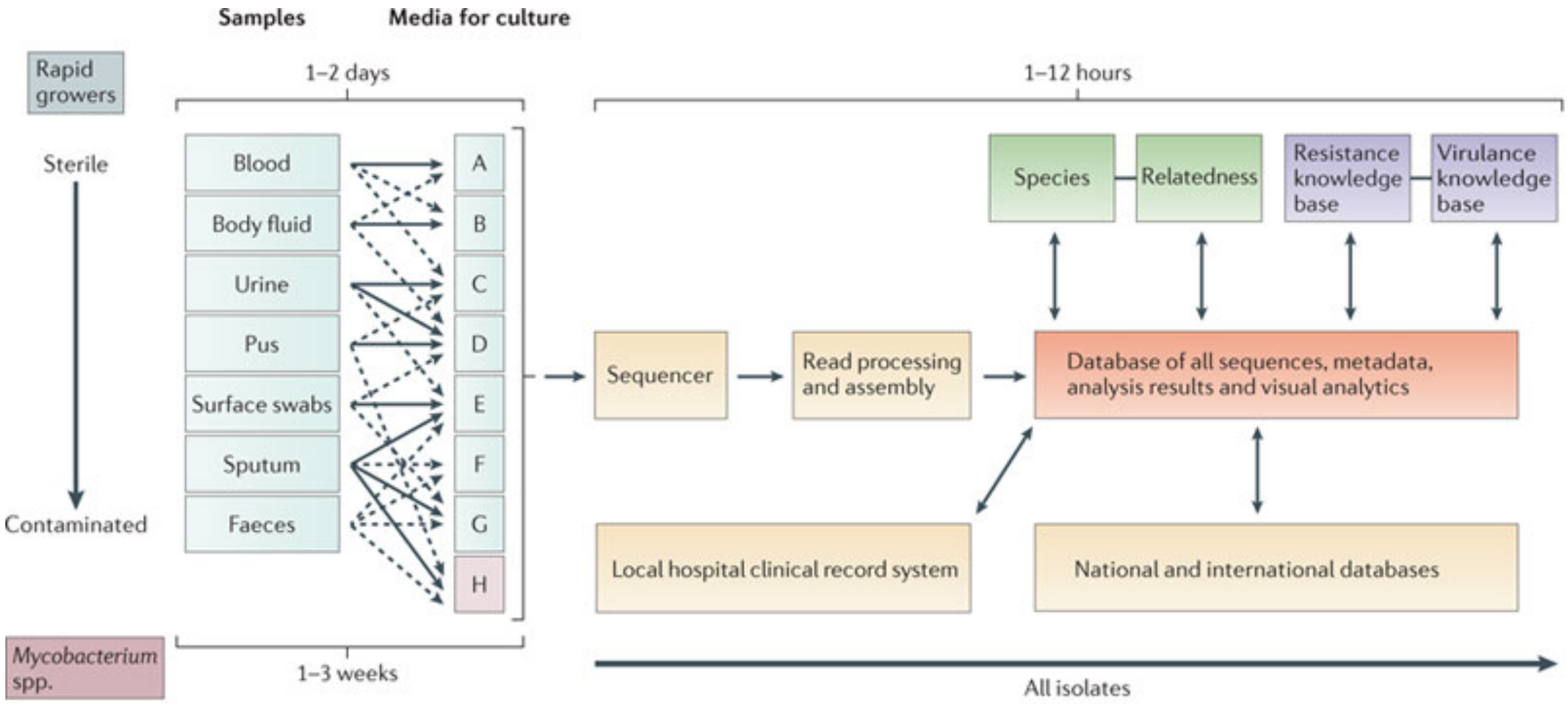
Again, only in humans…

Ultimately, mapping is to quickly identify relationship between individuals / species once reference is known

Tradeoffs between $$$, sample size, sensitivity, speed



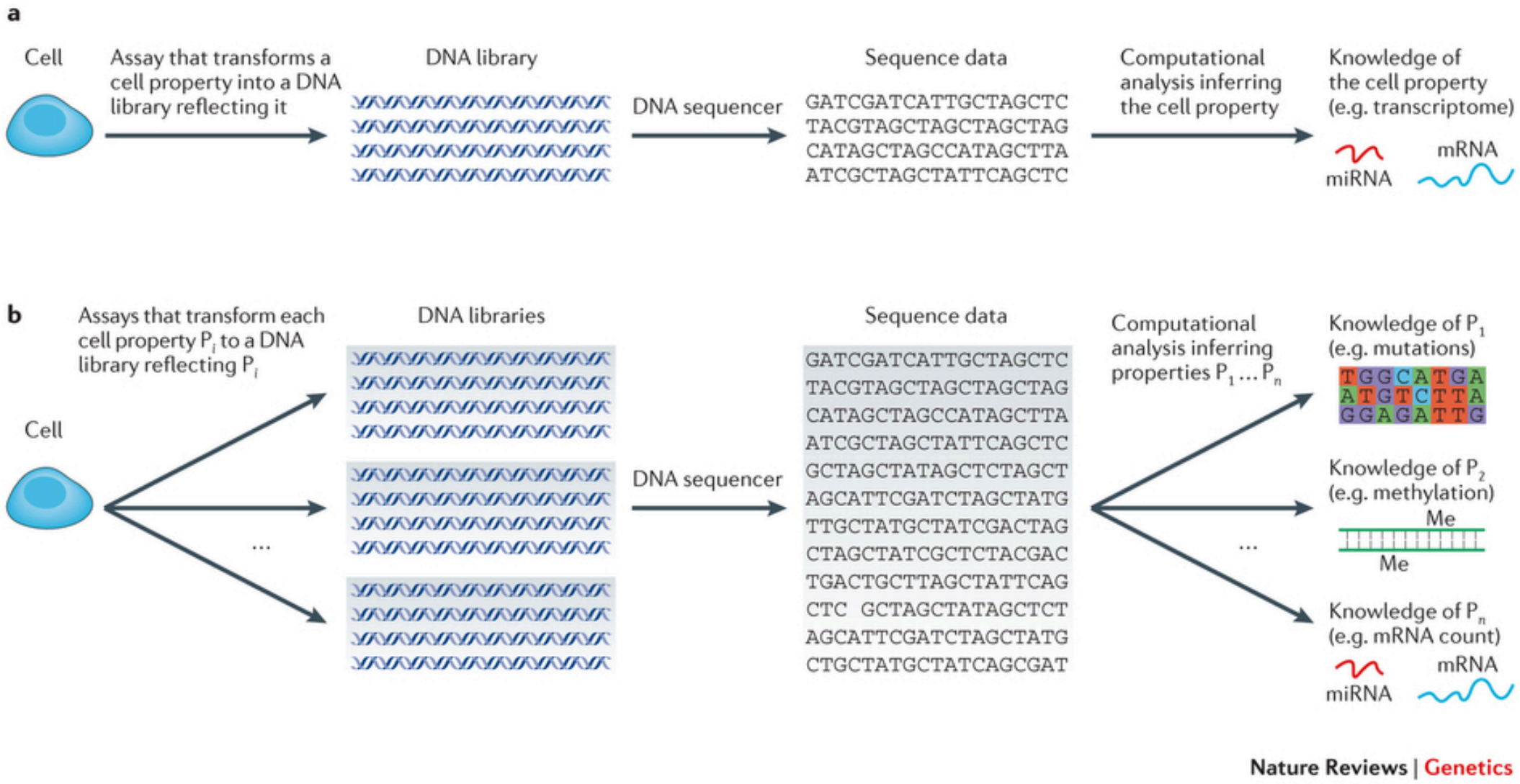
workflow of clinical labs / ecological samples



Nature Reviews | Genetics

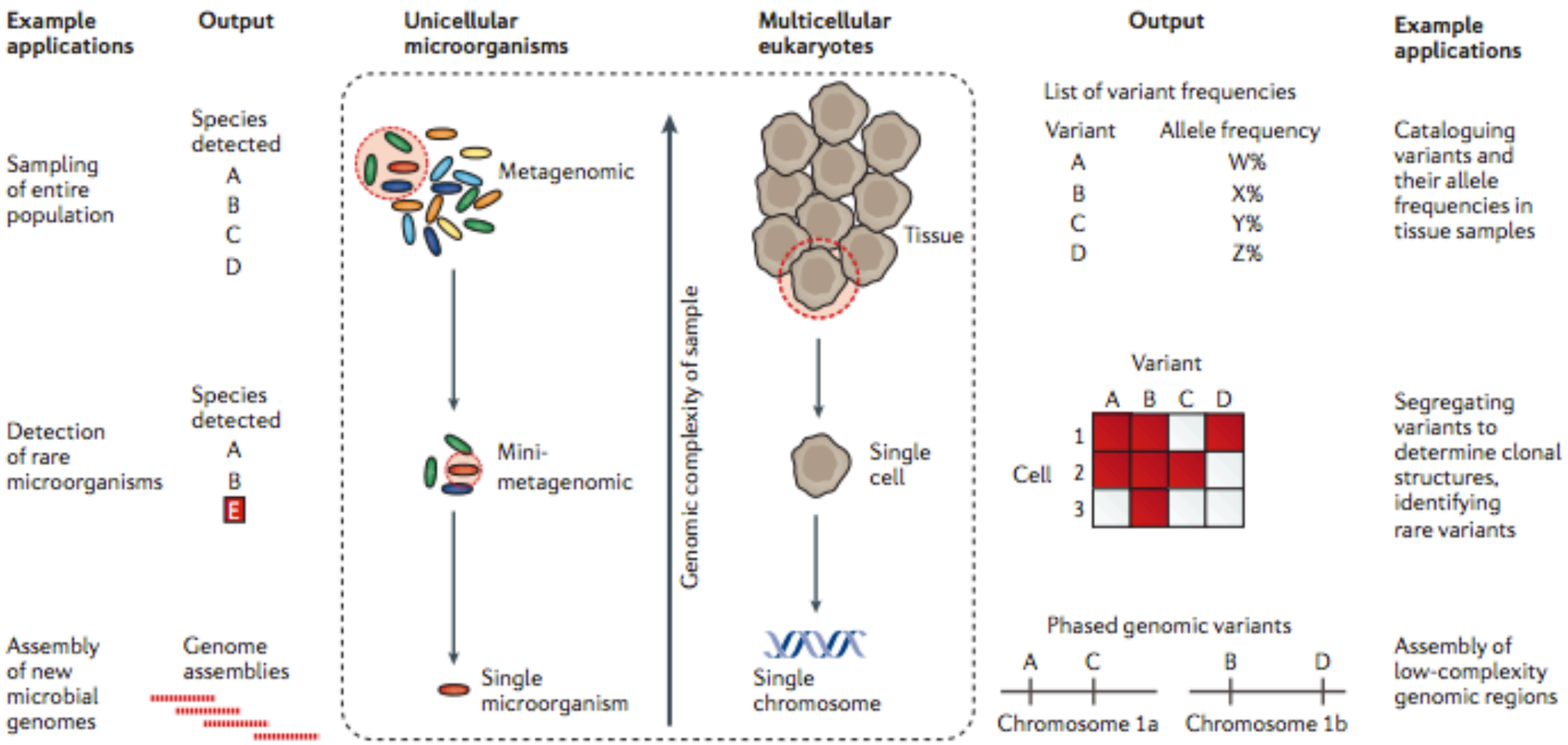
Didelot et al 2012

single cell genomics



doi:10.1038/nrg3542

single cell genomics



Gawad *et al* 2016

Example

Hoffmann *et al. Climate Change Responses* (2015) 2:1
DOI 10.1186/s40665-014-0009-x

Climate Change Responses

REVIEW Open Access

A framework for incorporating evolutionary genomics into biodiversity conservation and management

Ary Hoffmann[1*], Philippa Griffin[1], Shannon Dillon[2], Renee Catullo[3], Rahul Rane[1], Margaret Byrne[4], Rebecca Jordan[1], John Oakeshott[5], Andrew Weeks[1], Leo Joseph[6], Peter Lockhart[7], Justin Borevitz[3] and Carla Sgrò[8]

Opinion

CellPress

Genomics and the challenging translation into conservation practice

Aaron B.A. Shafer[1], Jochen B.W. Wolf[1], Paulo C. Alves[2], Linnea Bergström[1], Michael W. Bruford[3], Ioana Brännström[1], Guy Colling[4], Love Dalén[5], Luc De Meester[6], Robert Ekblom[1], Katie D. Fawcett[7], Simone Fior[8], Mehrdad Hajibabaei[9], Jason A. Hill[10], A. Rus Hoezel[11], Jacob Höglund[1], Evelyn L. Jensen[12], Johannes Krause[13], Torsten N. Kristensen[14], Michael Krützen[15], John K. McKay[16], Anita J. Norman[17], Rob Ogden[18], E. Martin Österling[19], N. Joop Ouborg[20], John Piccolo[19], Danijela Popović[21], Craig R. Primmer[22], Floyd A. Reed[23], Marie Roumet[8], Jordi Salmona[24], Tamara Schenekar[25], Michael K. Schwartz[26], Gernot Segelbacher[27], Helen Senn[18], Jens Thaulow[28], Mia Valtonen[29], Andrew Veale[12], Philippine Vergeer[30], Nagarjun Vijay[1], Carles Vilà[31], Matthias Weissensteiner[1], Lovisa Wennerström[10], Christopher W. Wheat[10], and Piotr Zieliński[32]

Table 1. Main areas traditionally addressed by conservation genetics [3], current status of genetic and genomic approaches, and the contribution that genomics can potentially make

Category	Status of conservation genetics	Possible contribution of conservation genomics	Required for transition from basic to applied[a]
Evolutionary genetics of natural populations			
Demographic inference – population history	Regularly used Moderate resolution	Improved accuracy and precision Finer-scale population structure Less limited by sample size	Clear understanding of limitations and biases User-friendly software
Adaptive genetic variation	Minimally used Based on population correlations [77] or candidate gene approaches	Improved detection of adaptive loci Management frameworks proposed [28] Methods still emerging Interpretations unclear	In-depth validation studies Genome annotation
Quantitative genetic variation	Limited resolution Often dependent on pedigrees or targeted gene approaches	Improved detection of quantitative trait loci Active application (e.g., genome-wide association studies)	Ecological studies Genome annotation
Taxonomic identification and general diagnostics	Regularly used Moderate resolution Restricted to single individuals	Assay species simultaneously [78] Improved hybridization detection Improved detection of pathogens	Defined pipelines (Box 3) Repeatability
Effects of small population size			
Inbreeding detection	Regularly used Limited resolution [34]	Improved estimates of inbreeding [34,62] Novel genomic metrics [79] Assess impact on specific genomic regions or adaptive loci	User-friendly bioinformatics Genome annotation Practitioner demand
Population viability	Minimally used [80]	Improved estimates of inbreeding metrics used in viability models [80]	Practitioner demand
Additional applications			
Genetic monitoring	Minimally used [11]	Improved sampling regimens [63] More powerful biodiversity surveys	Practitioner demand Compliance [11]
Population census	Regularly used	Higher-throughput screening	Practitioner demand
Maternity, paternity, and kinship analysis	Regularly used	Useful when microsatellite power is limited [81]	Practitioner demand

Genomics research and development

- SNP discovery
- SNP validation and selection
- Genome-wide genotyping

Marker assessment and selection

- Population screening
- Population genomic analysis
- SNP panel selection

Applied traceability tools

- Platform selection
- Method validation
- Standard operating procedures

The*FishPopTrace*target species

reproduced with permission from the Scandinavian Fishing Yearbook (c)

European hake (*Merluccius merluccius L.*)

Atlantic herring (*Clupea harengus L.*)

Atlantic cod (*Gadus morhua L.*)

Common sole (*Solea solea L.*)

TRENDS in Ecology & Evolution

Trends in Ecology & Evolution, February 2015, Vol. 30, No. 2

Reading materials

Introduction to Read-Based Alignment
http://evomicsorg.wpengine.netdna-cdn.com/wp-content/uploads/2015/06/Intro-Read-Alignment.pdf

Ben Langmead
http://www.langmead-lab.org/teaching-materials/

Additional references:
https://www.notion.so/References-papers-links-in-start-learning-genomics-b7e57b28e9194bb29a02f483e0b894ad

Written assignment

Construct a BWT of the following sequence:

ANNABANANA

Question:

1. What is the output of last column?
2. Write out how you searched the string ANNA
   (hint: follow wiki*)

To be handed in **2020.04.15**

https://en.wikipedia.org/wiki/Burrows%E2%80%93Wheeler_transform