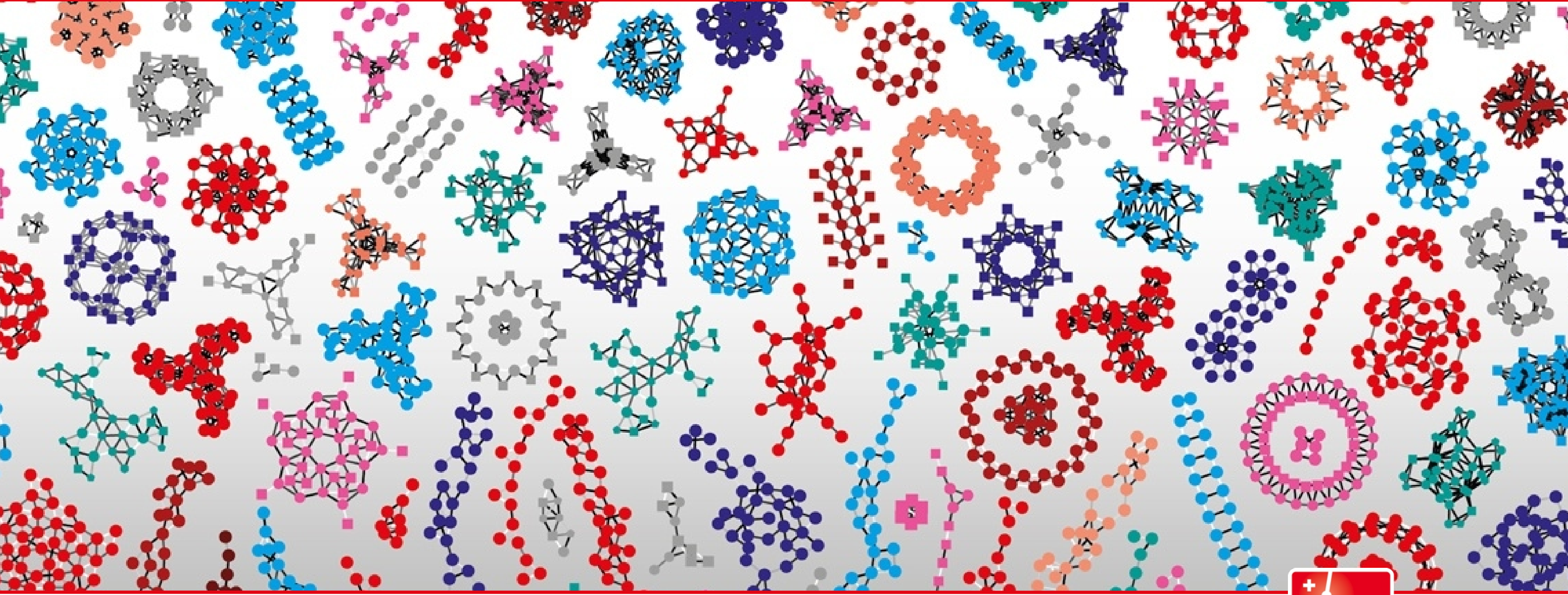
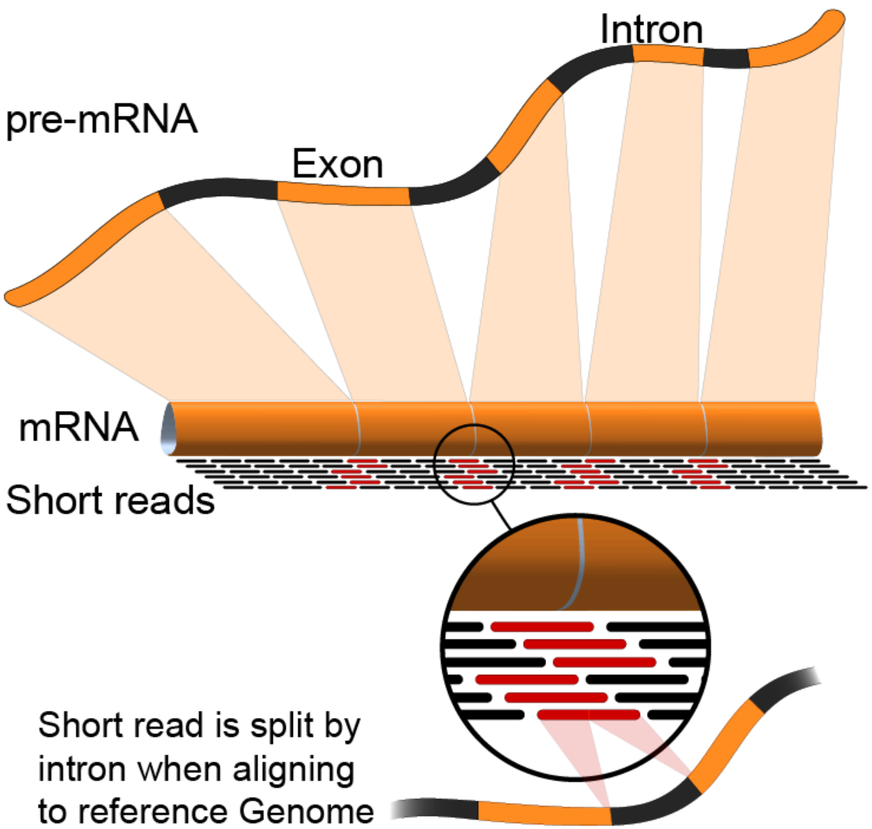


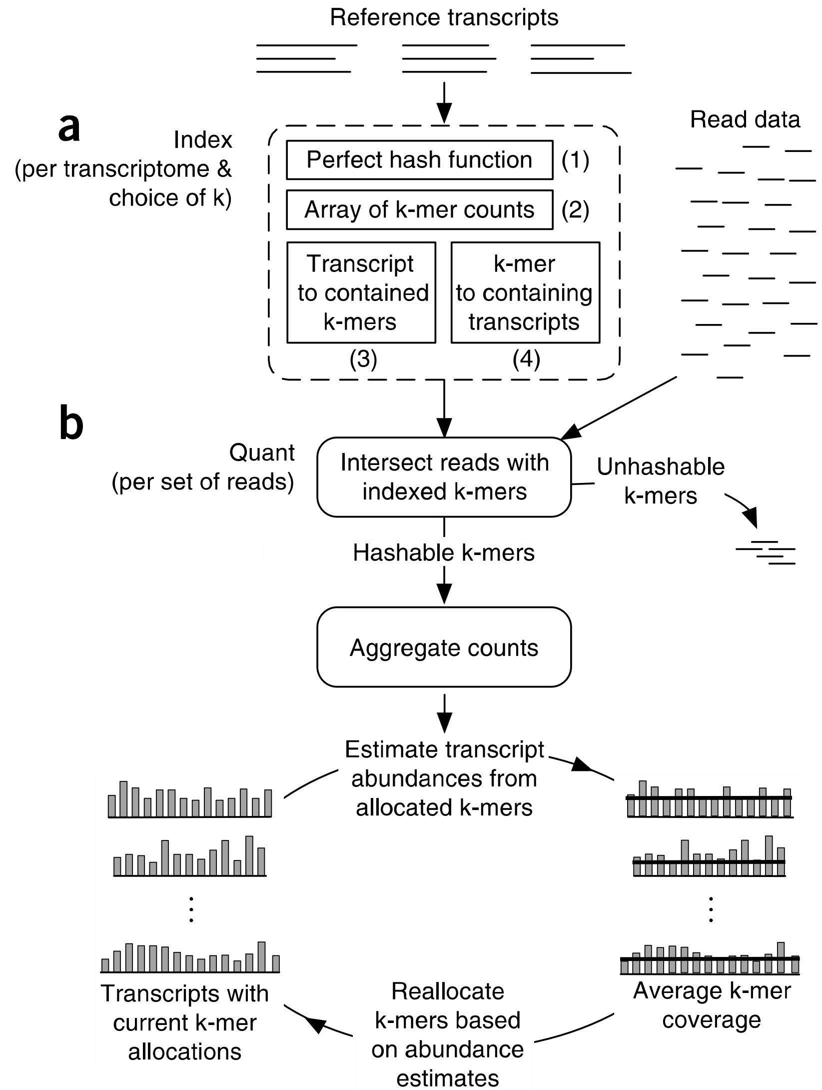
Swiss Institute of Bioinformatics

Introduction to RNA-Seq – Mapping & Aligning

Wandrille Duchemin

Alignment vs. Pseudoalignment

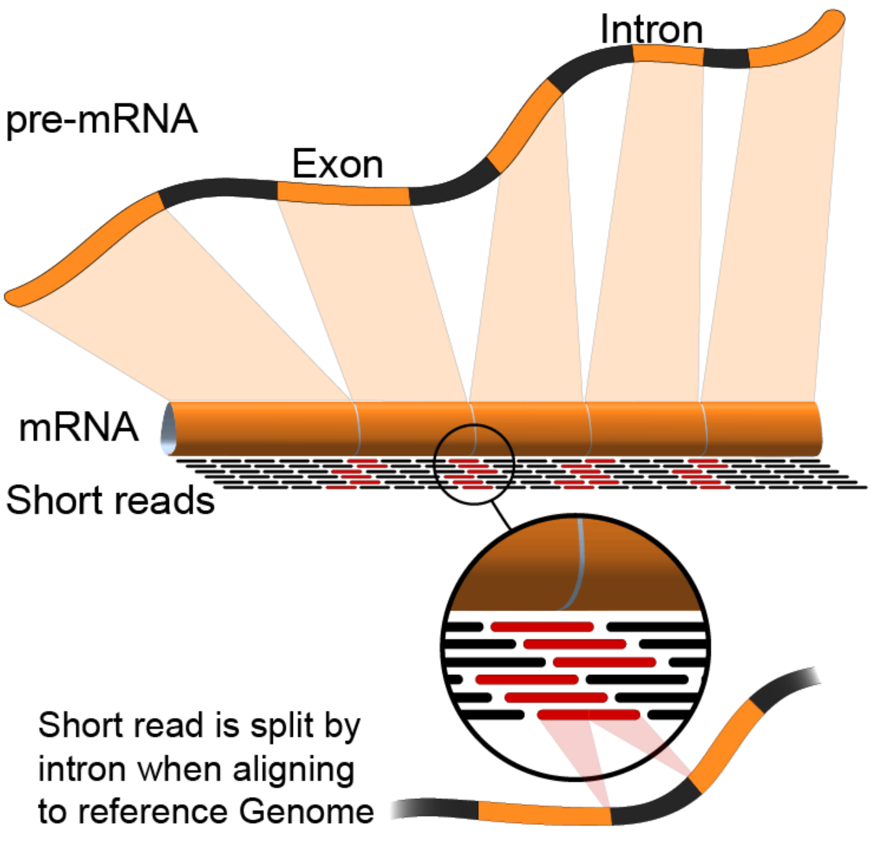


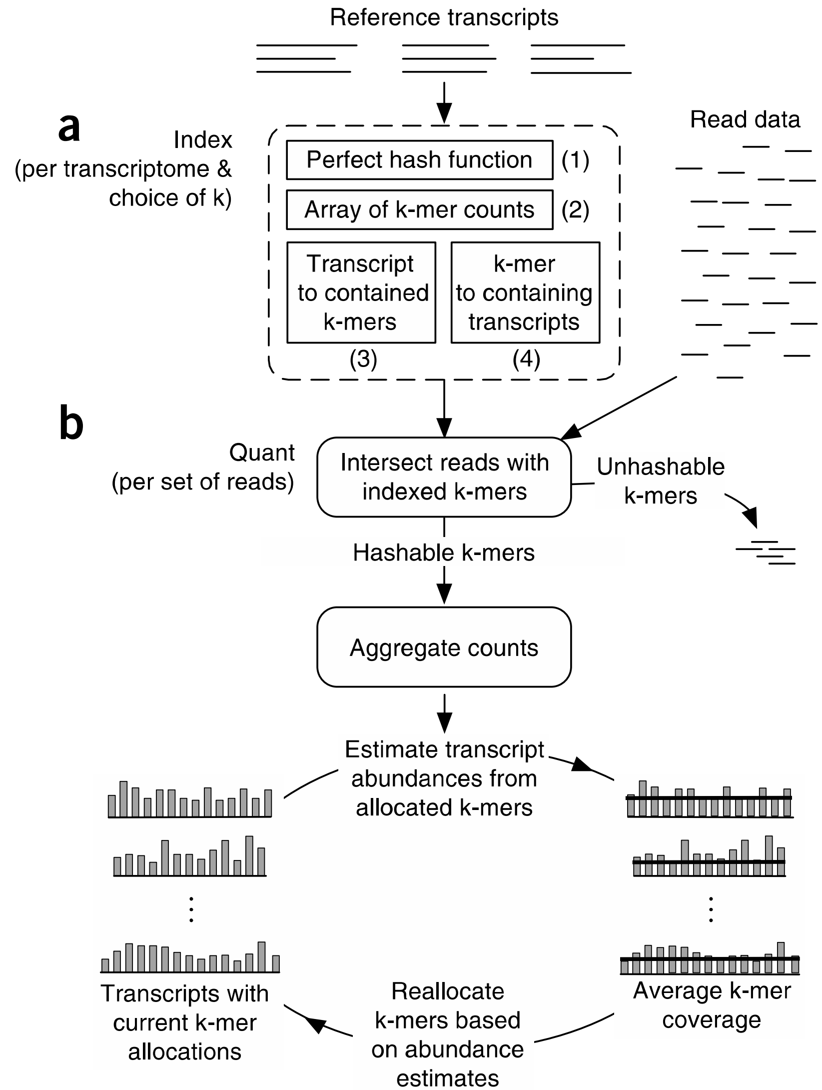


https://en.wikipedia.org/wiki/RNA-Seq

sailfish (Patro et al. 2014), see also “Kallisto” (Bray et al.2016)

Alignment vs. Pseudoalignment





resource intensive!

https://en.wikipedia.org/wiki/RNA-Seq

sailfish (Patro et al. 2014), see also “Kallisto” (Bray et al.2016)

Alignment vs. Pseudoalignment

alignment

↓

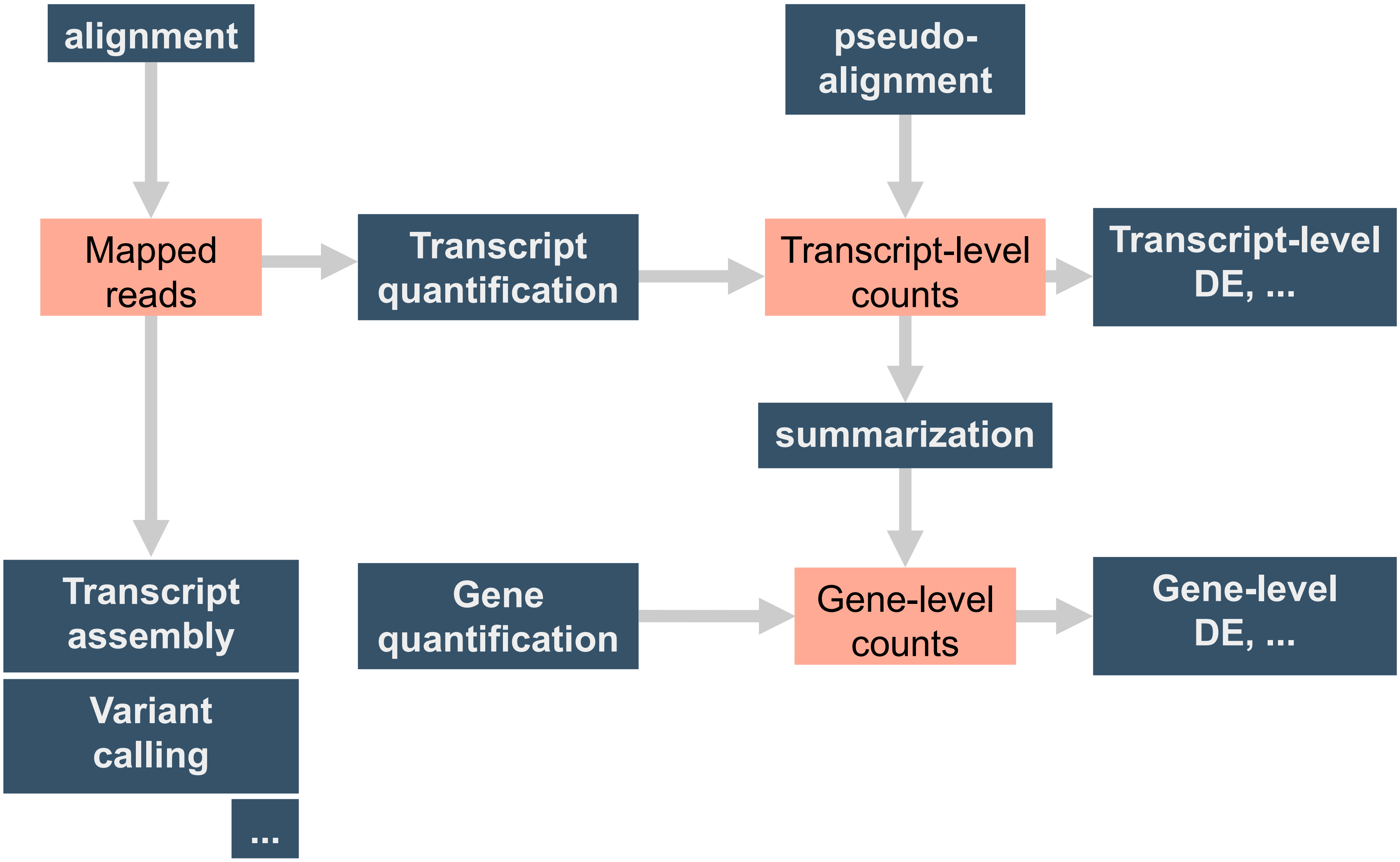
Mapped reads

pseudo-alignment

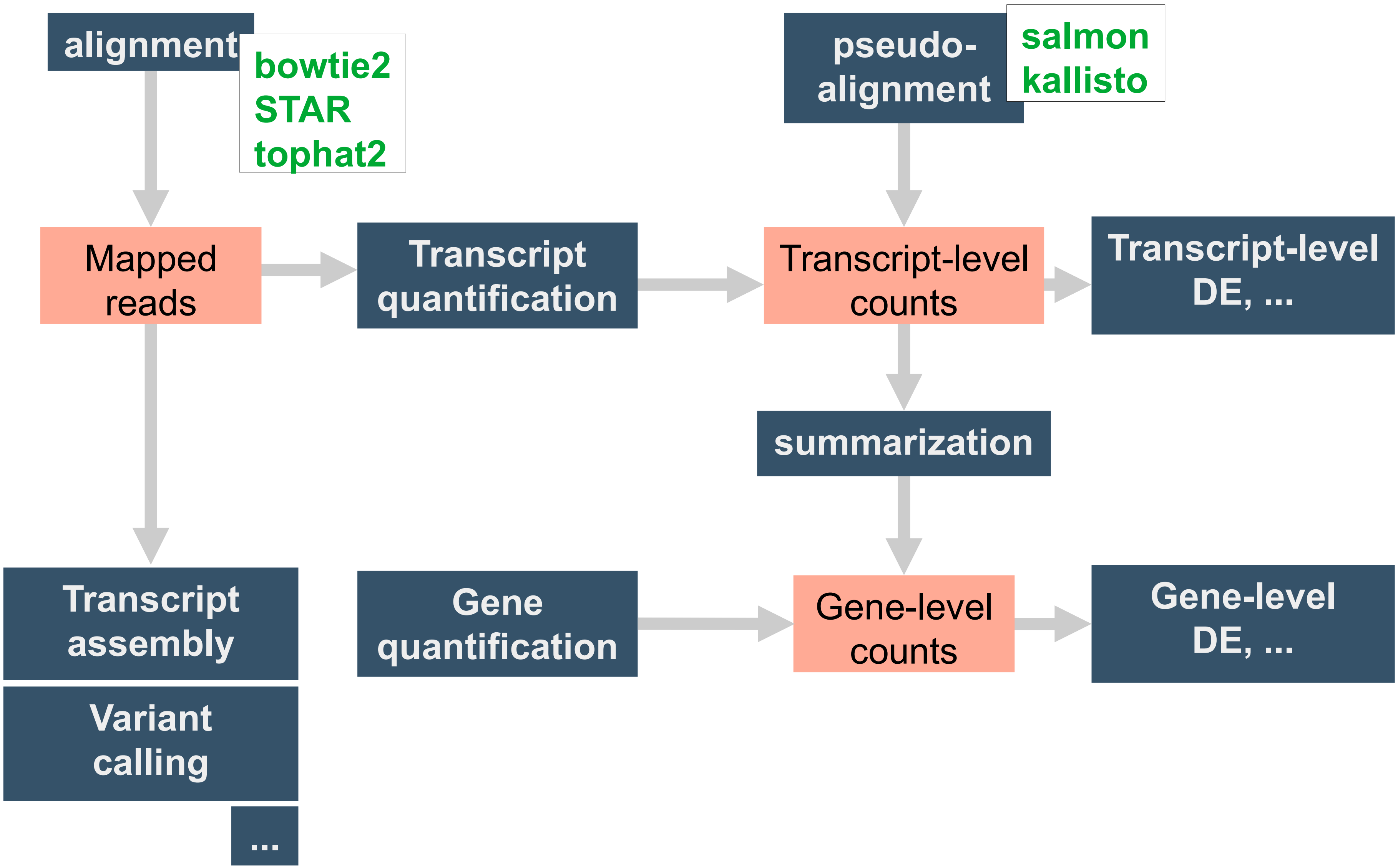
↓

Transcript-level counts

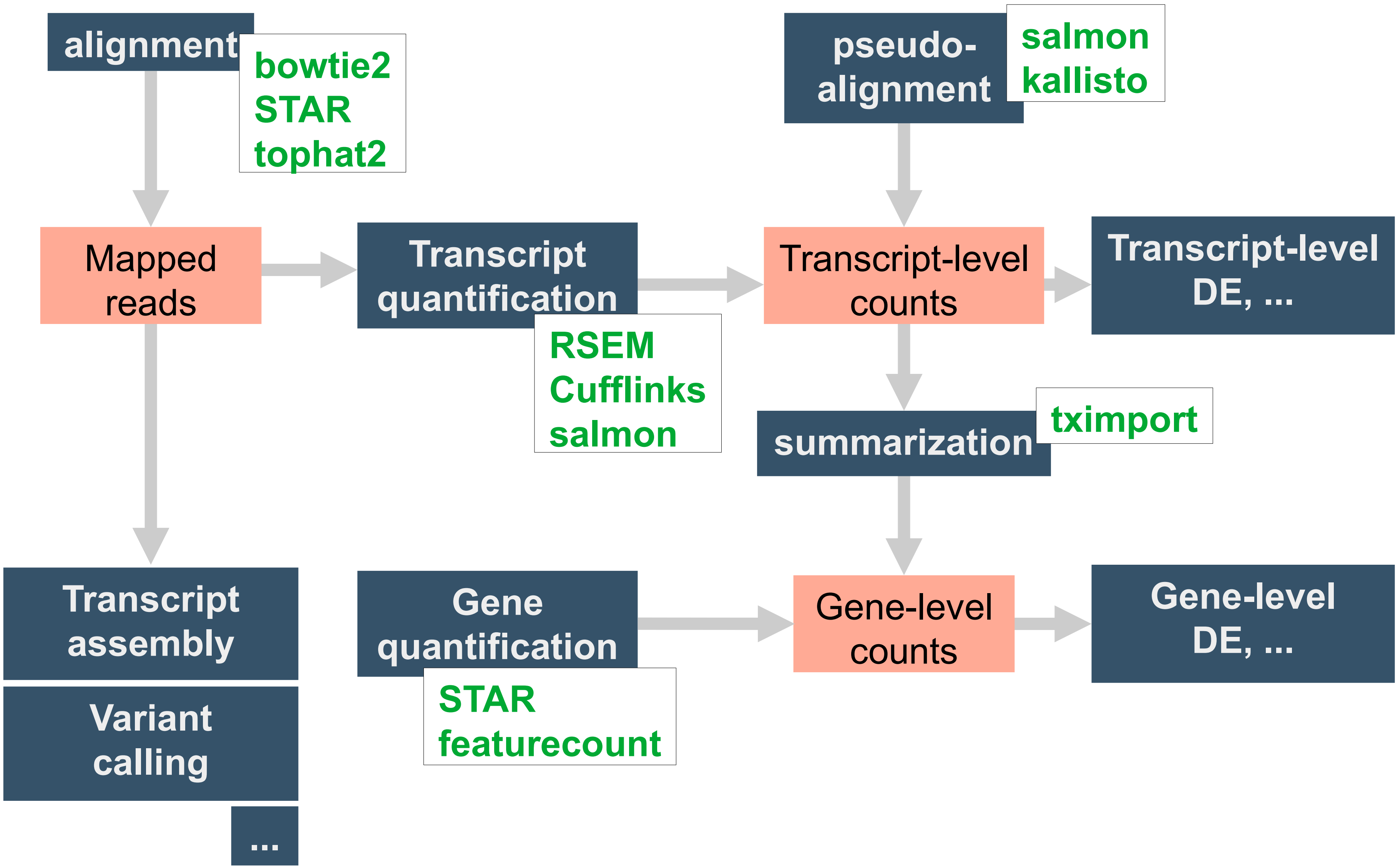
Alignment vs. Pseudoalignment



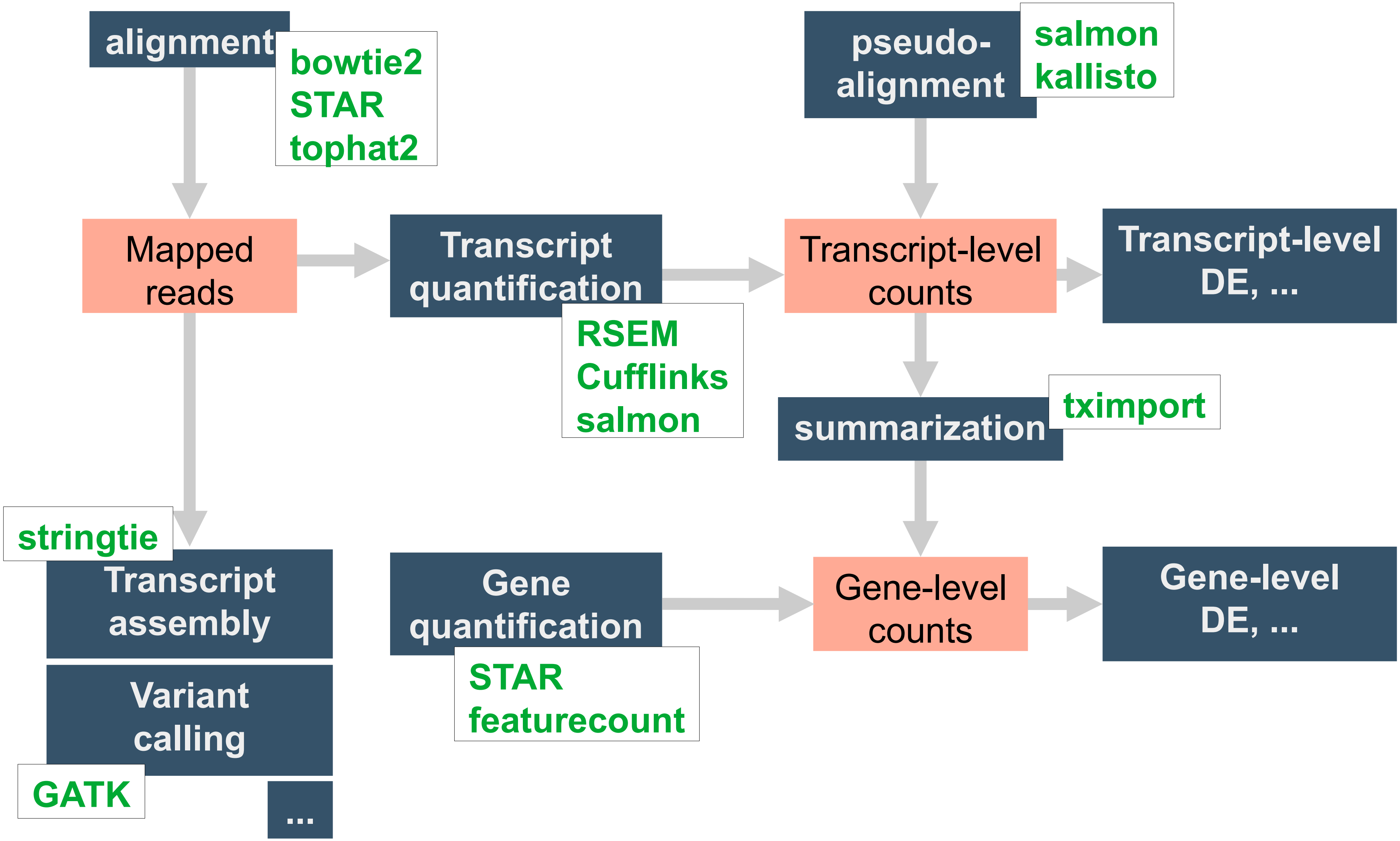
Alignment vs. Pseudoalignment



Alignment vs. Pseudoalignment

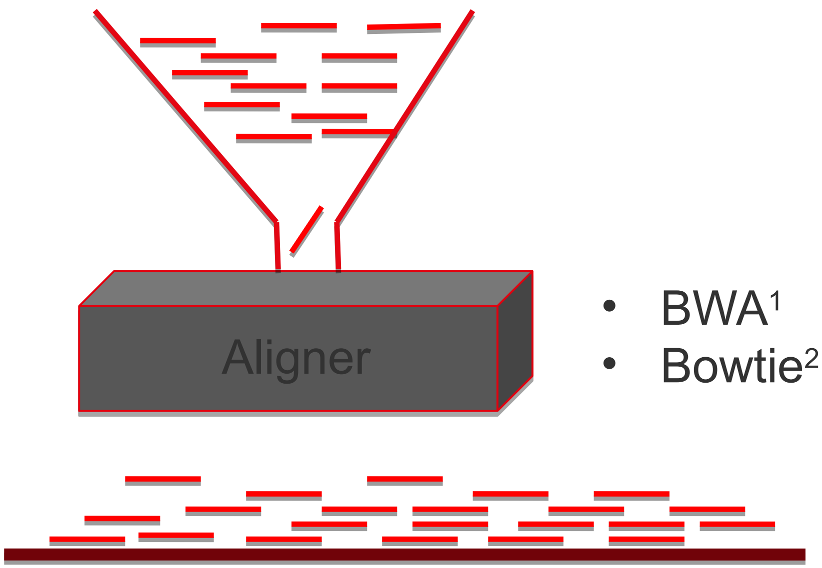


Alignment vs. Pseudoalignment



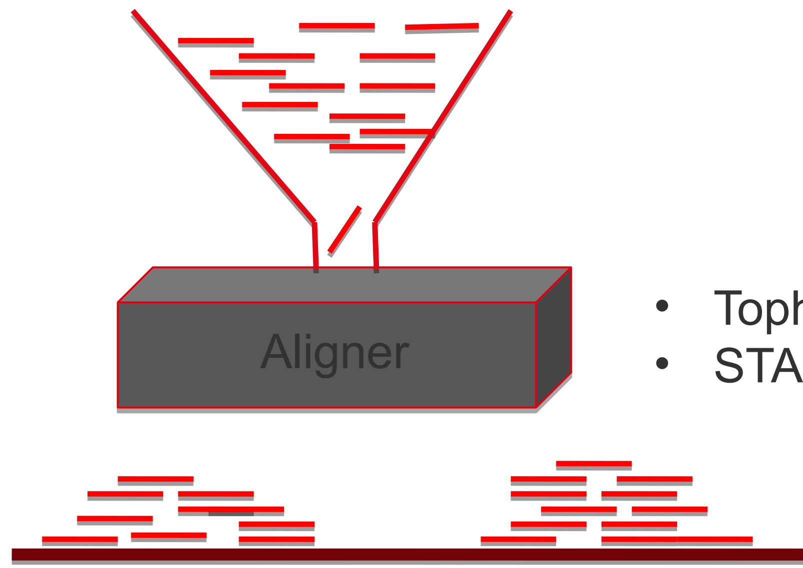
“Aligning” & “Mapping” Sequencing Reads

Whole genome re-sequencing



- BWA[1]
- Bowtie[2]

Transcriptome sequencing (RNA-seq)



- Tophat[3]
- STAR[4]

1. Li and Durbin 2009
2. Langemead et al. 2009
3. Trapenell et al. 2009; Kim et al. 2013
4. Dobin et al. 2013

Alignment using STAR

- **Phase 1 – Mapping using "Maximum Mappable Prefix**

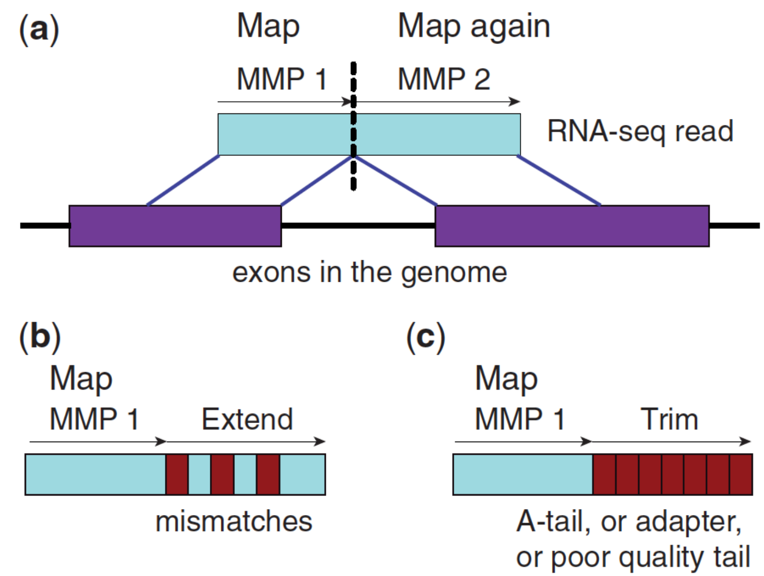
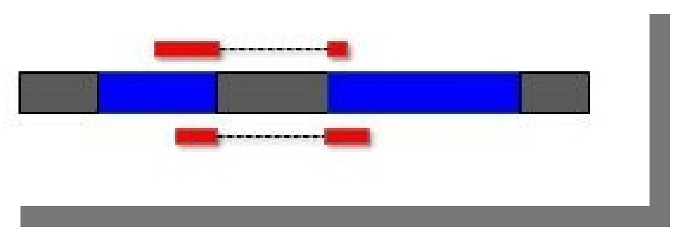


Fig. 1. Schematic representation of the Maximum Mappable Prefix search in the STAR algorithm for detecting (**a**) splice junctions, (**b**) mismatches and (**c**) tails

- **Phase 2 – "Stitching"**



Dobin *et al* 2013

Benchmarking the Aligners (simulated dataset)

	Correctly mapped 200 bases	>=150 bases correctly mapped	Unmapped	True positive junctions		False positive junctions	
				Number	Sensitivity	Number	FDR
Aligner	1	2	4	5	6	7	8
STAR	81.3%	95.0%	4.82%	148,487	92.7%	409	0.3%
TopHat2	82.6%	83.7%	6.70%	135,006	84.3%	1,228	0.9%

- Star : x20 faster than Tophat2
- Tophat2 : x6 more memory efficient (can be run on recent laptops)

Dobin & Gingeras 2013

Benchmarking the Aligners (simulated dataset)

	Correctly mapped 200 bases	>=150 bases correctly mapped	Unmapped	True positive junctions		False positive junctions	
				Number	Sensitivity	Number	FDR
Aligner	1	2	4	5	6	7	8
STAR	81.3%	95.0%	4.82%	148,487	92.7%	409	0.3%
TopHat2	82.6%	83.7%	6.70%	135,006	84.3%	1,228	0.9%

- Star : x20 faster than Tophat2
- Tophat2 : x6 more memory efficient (can be run on recent laptops)

Dobin & Gingeras 2013

Essentially, if you have access to a cluster you should be using STAR

Reference Index Preparation

Different for each software!

- **Need a suitable reference genome**
  - sequence
  - annotation

https://www.ensembl.org/info/data/ftp/index.html

https://hgdownload.soe.ucsc.edu/downloads.html

Genome Annotation Files

- **text file describing genomic features**
  - Gene, CDS, exon, intron, miRNA, etc
  - Chromosome, start, end, strand, attributes, etc

- **most common format: gtf / gff3**

http://www.ensembl.org/info/website/upload/gff.html

Example GTF

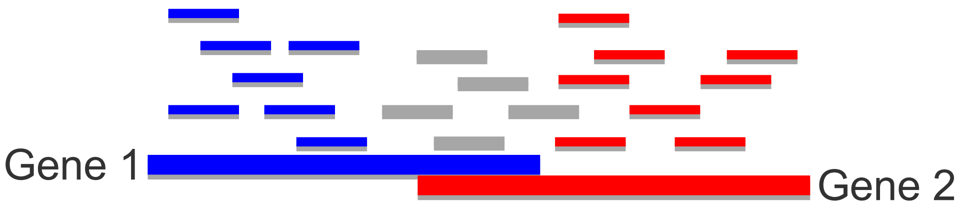
```
MT   RefSeq   gene          2751 3707 .   +   .   gene_id "ENSMUSG00000064341"; gene_vers
MT   RefSeq   transcript    2751 3707 .   +   .   gene_id "ENSMUSG00000064341"; gene_vers
MT   RefSeq   exon          2751 3707 .   +   .   gene_id "ENSMUSG00000064341"; gene_vers
MT   RefSeq   CDS           2751 3704 .   +   0   gene_id "ENSMUSG00000064341"; gene_vers
MT   RefSeq   start_codon   2751 2753 .   +   0   gene_id "ENSMUSG00000064341"; gene_vers
MT   RefSeq   stop_codon    3705 3707 .   +   0   gene_id "ENSMUSG00000064341"; gene_vers
```

After mapping: counting

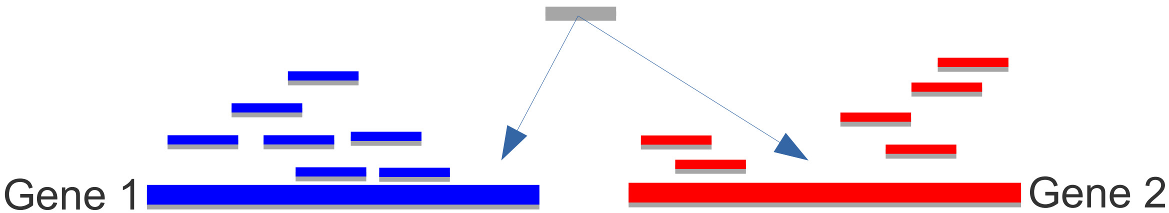
- **Pseudo-aligner : Transcript-level expression quantification**
- **Aligner : subsequently estimate expression from mapped reads**

Counting – Fundamental problems

Overlapping genes:

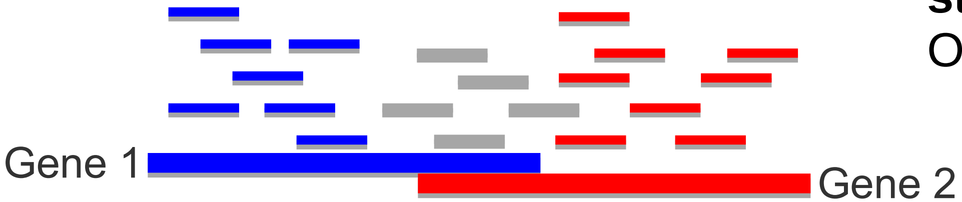


Multi-mapping reads:



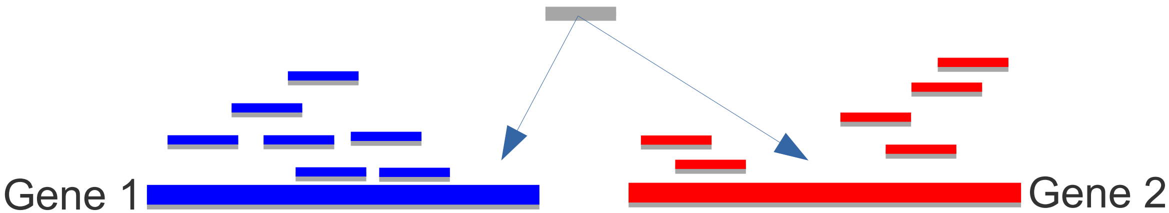
Counting – Fundamental problems

Overlapping genes:



stranded sequencing
OR
- discard reads?
- count in both?

Multi-mapping reads:



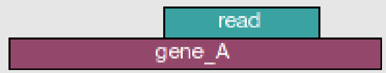
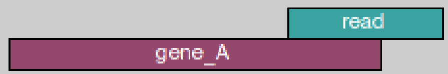
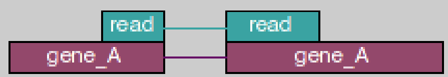
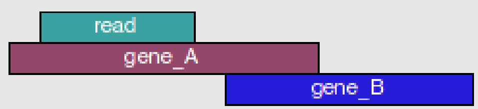
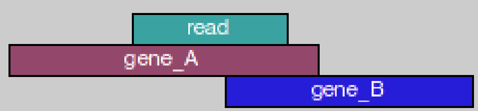
- discard reads?
- count in both?

Counting – gene-level counters

- **HTSeq**
- **FeatureCount**
- **STAR** (--quantMode GeneCounts)

Counting – gene-level counters

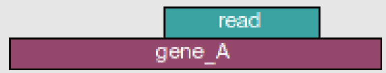
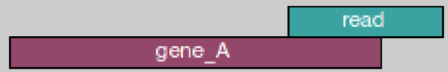
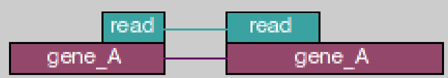
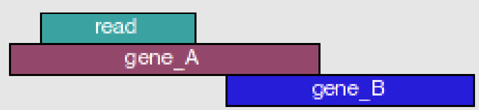
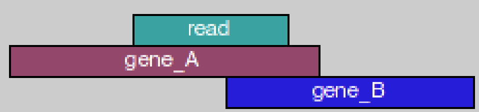
- **HTSeq**

	union	intersection _strict	intersection _nonempty
	gene_A	gene_A	gene_A
	gene_A	no_feature	gene_A
	gene_A	no_feature	gene_A
	gene_A	gene_A	gene_A
	gene_A	gene_A	gene_A
	ambiguous	gene_A	gene_A
	ambiguous	ambiguous	ambiguous

Counting – gene-level counters

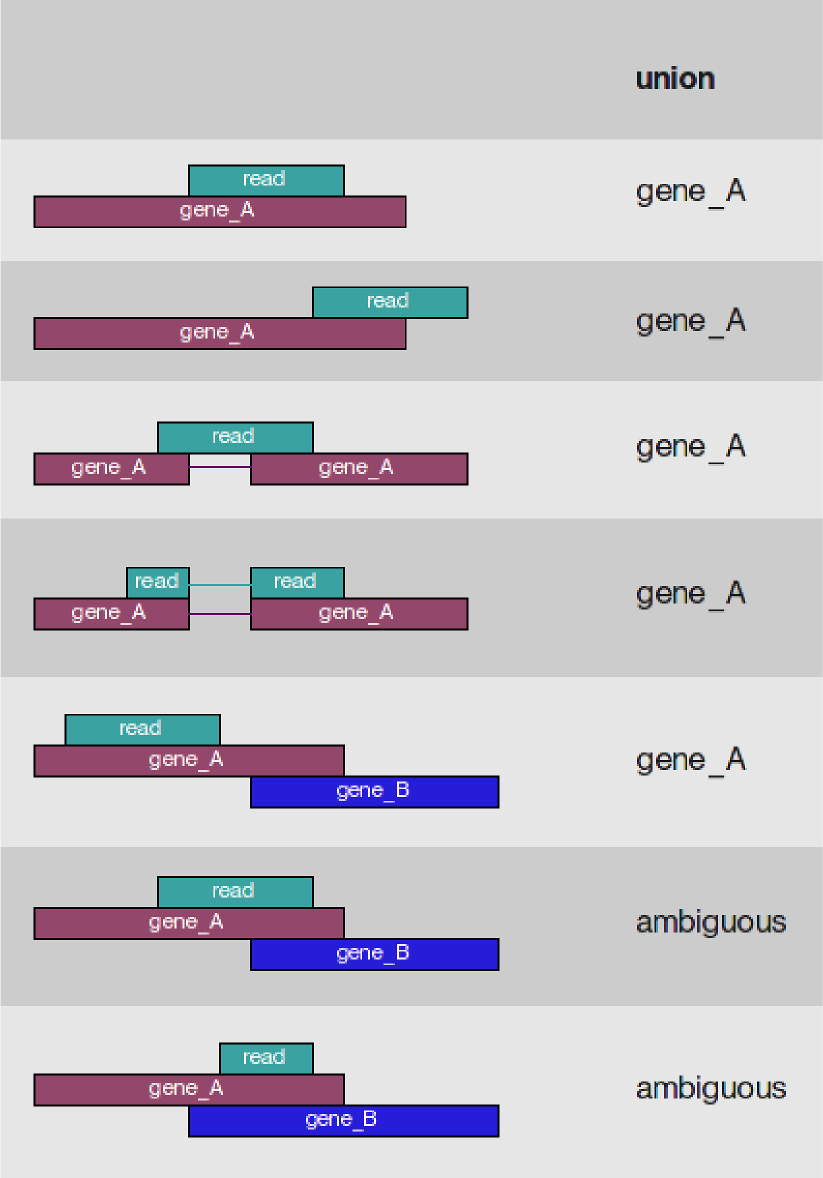
- **HTSeq**
- **FeatureCount**

+ options for fractional count

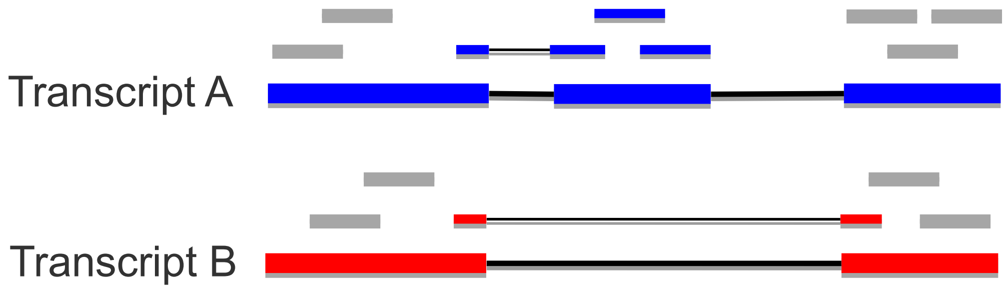
	union	intersection _strict	intersection _nonempty
	gene_A	gene_A	gene_A
	gene_A	no_feature	gene_A
	gene_A	no_feature	gene_A
	gene_A	gene_A	gene_A
	gene_A	gene_A	gene_A
	ambiguous	gene_A	gene_A
	ambiguous	ambiguous	ambiguous

Counting – gene-level counters

- **HTSeq**
- **FeatureCount**
- **STAR**
  (--quantMode GeneCounts)



Counting – transcript-level



→ **RSEM, Cufflink, Salmon, StringTie ...**

Practical

Go to the website and do the mapping practical

REFERENCES

Li H & Durbin R (2009) "Fast and accurate short read alignment with Burrows-Wheeler transform" Bioinformatics 25(14):1754-60

Langmead *et al* (2009) "Ultrafast and memory-efficient alignment of short DNA sequences to the human genome" Genome Biology 10(3):R25.

Trapnell *et al* (2009) "TopHat: discovering splice junctions with RNA-Seq" Bioinformatics 25(9):1105-11.

Kim *et al* (2013) "TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions" Genome Biology 14(4):R36.

Dobin *et al* (2013) "STAR: ultrafast universal RNA-seq aligner" Bioinformatics 29(1):15-21.

Patro *et al* (2014) "Sailfish enables alignment-free isoform quantification from RNA-seq reads using lightweight algorithms" Nature Biotechnology 32(5):462-4.

Patro *et al* (2017) "Salmon provides fast and bias-aware quantification of transcript expression" Nature Methods 14(4):417-419.

Bray *et al* (2016) "Near-optimal probabilistic RNA-seq quantification (Kallisto)" Nature Biotechnology 34(5):525-7.

Swiss Institute of Bioinformatics

Contributors:

Wandrille Duchemin
Geoffrey Fucile
Walid Gharib
Pablo Escobar Lopez

Annex: Pseudoalignment

transcript-level quantification →

pseudo alignment generally better than alignment

https://www.nature.com/articles/s41598-017-01617-3

https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-021-04198-1

kallisto & salmon : very close,
maybe small advantage for salmon

See: https://github.com/mikelove/salmon_kallisto_diffs + publications above

Annex: GTF (GFF2) Annotation Format

- **http://www.ensembl.org/info/website/upload/gff.html**
- **Tab-delimited, empty columns denoted with "."**
- **Column order:**
  - **seqname** – chromosome, scaffold, etc
  - **source** – origin of the annotation, db/project
  - **feature** – gene, transcript, exon, etc
  - **start** – feature start coordinate (1-based)
  - **end** – feature end coordinate (1-based)
  - **score** – floating point, *eg* quality score
  - **strand** – + (forward) or – (reverse)
  - **frame** – reading frame, 0, 1, or 2
  - **attribute** – semicolon-delimited feature descriptions

GTF vs GFF3

Columns	GTF2	GFF3
1. reference sequence name	same	same
2. annotation source	same	same
3. feature type	~~CDS, start_codon, end_codon are required.~~ feature requirements depend on software.	can be anything
4. start coordinate	same	same
5. end coordinate	same	same
6. score	not used	optional
7. strand	same	same
8. frame	same	same
9. attributes	• tag/value delimited by a space • each attribute must end with a semi-colon • must begin with gene_id and transcript_id attributes • Text values must be in quotes • ex. gene_id "gene01"; transcript_id "transcript01"; created_by "Damian";	• tag/value delimited by '=' • each attribute delimited by semi-colon • there are a list of pre-defined attributes here • must have a unique ID attribute • ex. ID=geneA;Parent=geneAP;Name=geneA

http://blog.nextgenetics.net/?e=27

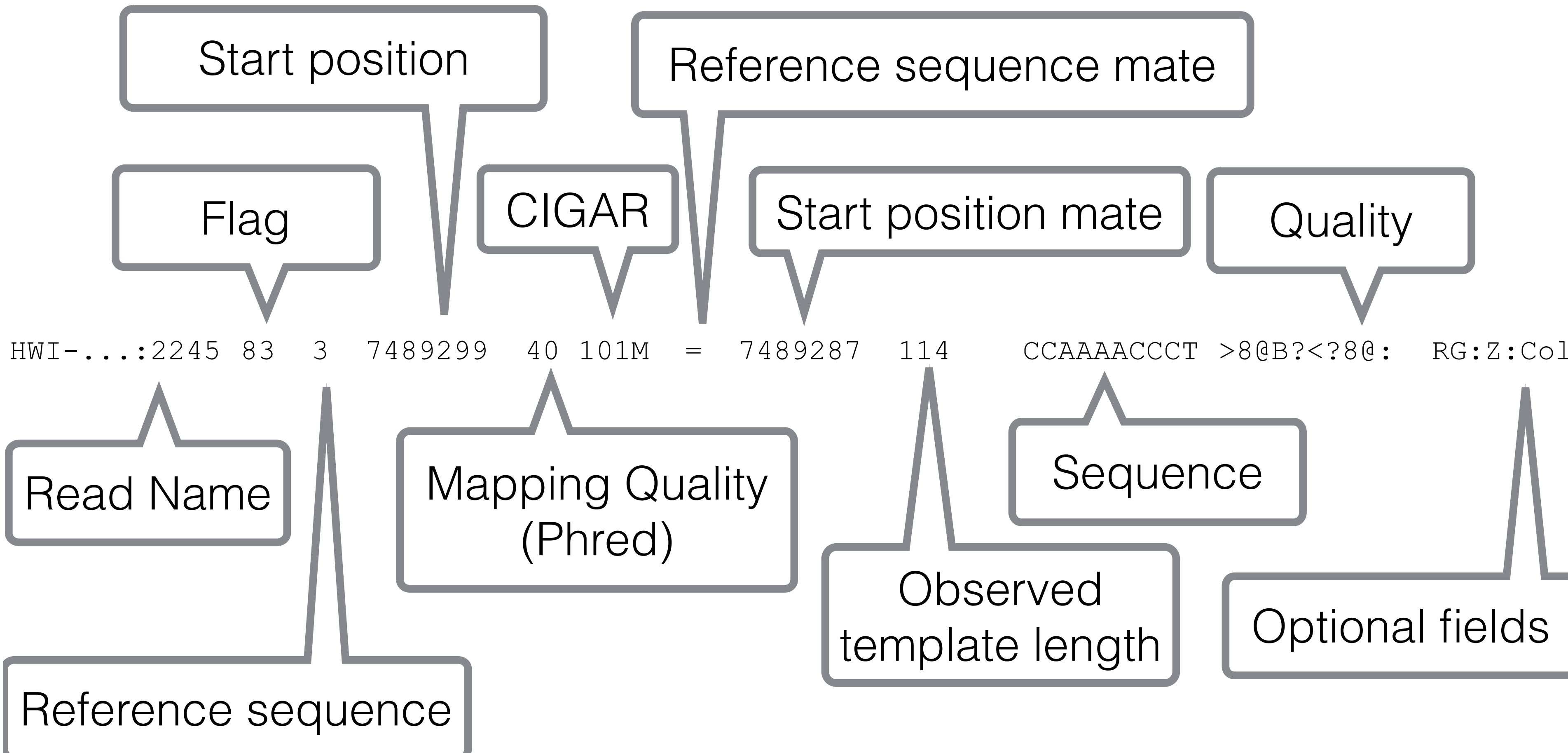
Annex : SAM Alignment Format

```
@SQ     SN:1 LN:30427671
@SQ     SN:2 LN:19698289
@SQ     SN:3 LN:23459830
@SQ     SN:4 LN:18585056
@SQ     SN:5 LN:26975502
@SQ     SN:M LN:366924
@SQ     SN:C LN:154478
@RG     ID:Col0_R1 PL:Illumina LB:1342 SM:Col0_R1
```

@SQ Reference Sequence: SN name, LN length
@RG Read Group: e.g. grouping samples

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

SAM alignments



SAM Alignment Format: Flags

Bit		Description
1	0x1	template having multiple segments in sequencing
2	0x2	each segment properly aligned according to the aligner
4	0x4	segment unmapped
8	0x8	next segment in the template unmapped
16	0x10	SEQ being reverse complemented
32	0x20	SEQ of the next segment in the template being reverse complemented
64	0x40	the first segment in the template
128	0x80	the last segment in the template
256	0x100	secondary alignment
512	0x200	not passing quality controls
1024	0x400	PCR or optical duplicate
2048	0x800	supplementary alignment

Example, flag 83 = 64+16+2+1 means it's first read (0x40) of pair-end reads (0x1) and it's mapped on minus strand (0x10) and both reads mapped (0x2).

https://broadinstitute.github.io/picard/explain-flags.html

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

SAM format: CIGAR string

- **Summary of alignment to the reference**
- ***eg*, 101M, 1S92M, 15M87N70M90N16M**

Code	Description	
M	Alignment match	Base-level match + mismatch
I	Insertion	
D	Deletion	
N	Skipped	*eg* intron
S	Soft clipping	Kept in SAM
H	Hard clipping	Not in SAM

SAM format: optional fields

- **Used by some aligners to encode additional information for downstream analyses**
- **Can cause incompatibilities among workflows**

Code	Description
RG	Read Group e.g. sample or lane
MD	String for mismatching positions
NH	Number of reported alignments that contains the query in the current record
HI	Query hit index, indicating the alignment record is the i-th one stored in SAM

BAM format

- **Binary SAM format**
- **Lossless compression of SAM format**
- **~4-fold smaller file size**
- **Genome viewers and many downstream applications require the BAM file to be sorted and have an index (typically .bai extension)**

Annex : Assessing read coverage for biases

- **The RSeQC package includes a function for evaluating "gene body coverage"**
- **This can be used to assess 5' or 3' bias, which might happen if your RNA is degraded or otherwise biased**
- **Requirements:**
  - Genome annotations in the 12-column BED format
  - Index (.bai) for sorted BAM file, which can be generated using the SAMtools package

```
samtools index sample1_sorted.bam

geneBody_coverage.py -r /data/GRCm38/Mus_musculus.GRCm38.89.bed12 \
                     -i sample1_sorted.bam \
                     -f pdf \
                     -o output_prefix
```

Annex - CRAM format

- **Binary SAM format, significantly improved over BAM lossless compression**
- **Compatible with BAM files**
- **Both lossless and lossy compression possible**
- **https://samtools.github.io/hts-specs/CRAMv3.pdf**

Annex - Other relevant formats: BED

- **Tab-delimited text file used to describe intervals**

- **Minimally:**
  - Sequence ID
  - Start
  - End

- **Optional:**
  - Name
  - Score
  - Strand

- **For large files, use binary index format bigBED**

- **BEDtools (http://code.google.com/p/bedtools)**

Annex - Other relevant formats: VCF (Variant Call Format)

- **Tab-delimited text to describe SNPs, structural variants, indels etc**
- **Contains:**
  - Chromosome
  - Position
  - Reference allele, alternative allele(s)
  - Various statistical metrics
- **BCF: indexed binary format**
- **https://samtools.github.io/hts-specs/VCFv4.2.pdf**