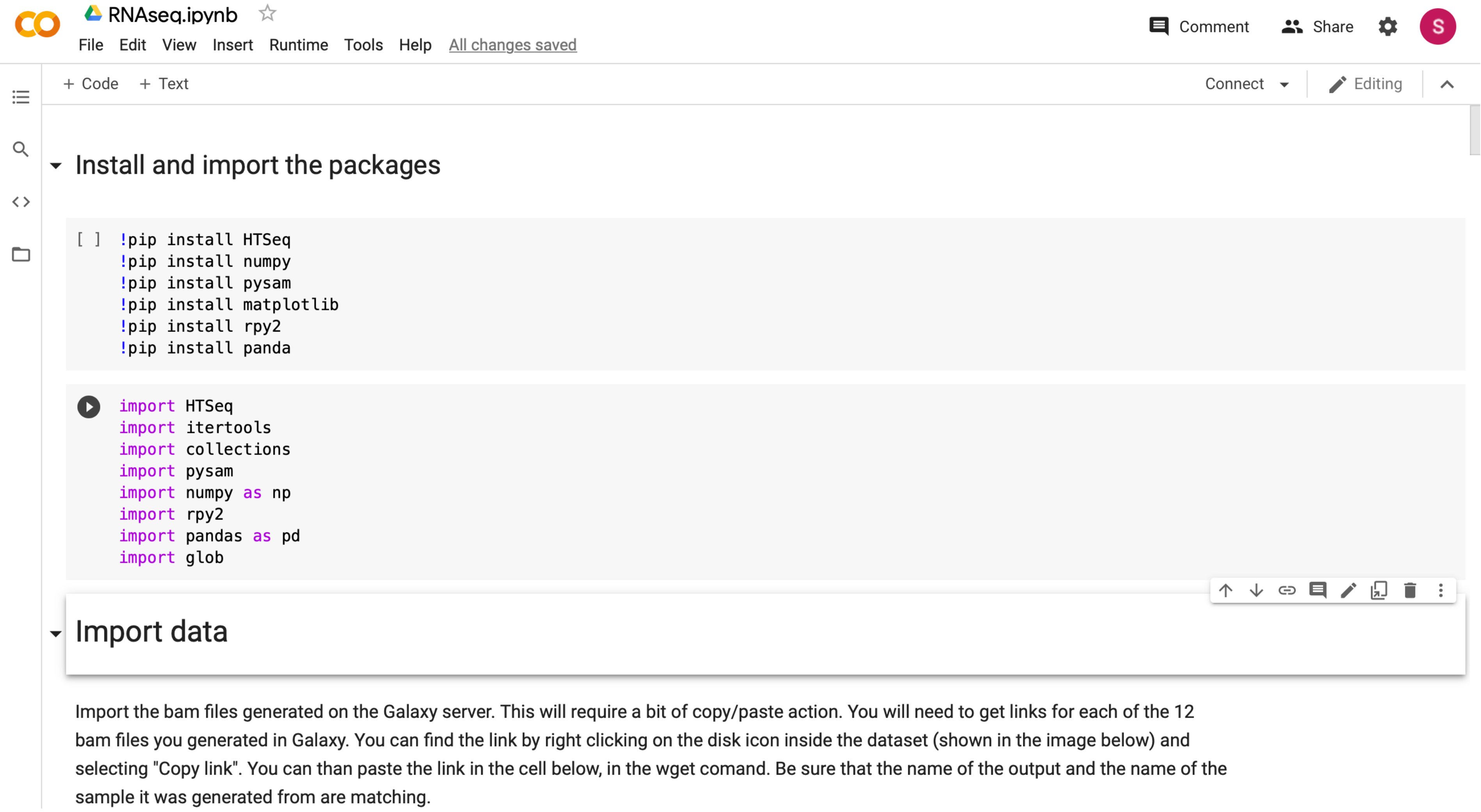
Jupyter Notebooks on UCloud



Install and import the packages

```
[ ] !pip install HTSeq
    !pip install numpy
    !pip install pysam
    !pip install matplotlib
    !pip install rpy2
    !pip install panda
```

```
import HTSeq
import itertools
import collections
import pysam
import numpy as np
import rpy2
import pandas as pd
import glob
```

Import data

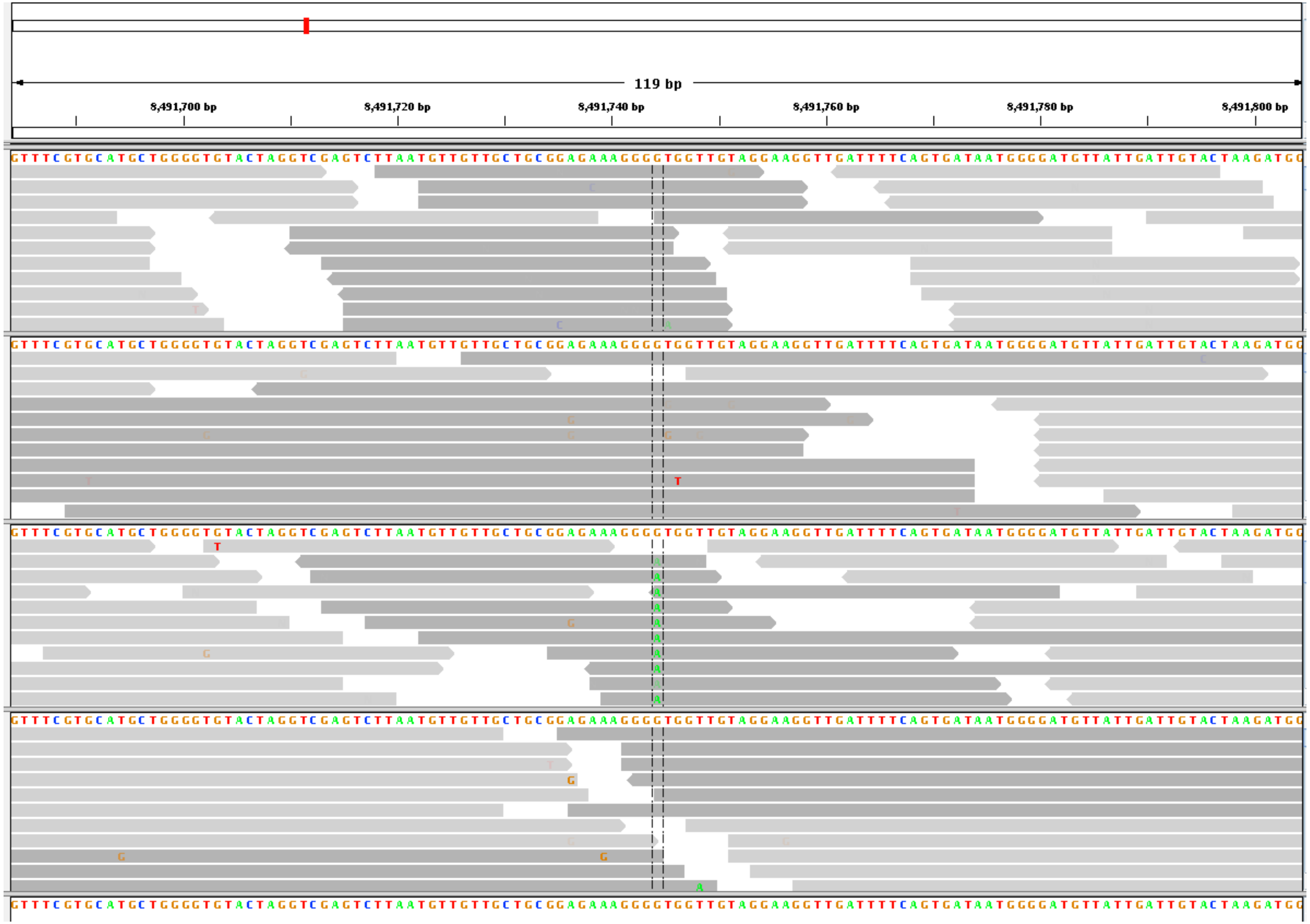
Import the bam files generated on the Galaxy server. This will require a bit of copy/paste action. You will need to get links for each of the 12 bam files you generated in Galaxy. You can find the link by right clicking on the disk icon inside the dataset (shown in the image below) and selecting "Copy link". You can than paste the link in the cell below, in the wget comand. Be sure that the name of the output and the name of the sample it was generated from are matching.

Overview

Lecture overview

- Mapping algorithms
- Mapping software
- Input format - FASTQ
- Output format – SAM/BAM/CRAM

Mapping reads example


119 bp
8,491,700 bp
8,491,720 bp
8,491,740 bp
8,491,760 bp
8,491,780 bp
8,491,800 bp
GTTTCGTGCATGCTGGGGTGTACTAGGTCGAGTCTTAATGTTGTTGCTGCGGAGAAAGGGGTGGTTGTAGGAAGGTTGATTTTCAGTGATAATGGGGATGTTATTGATTGTACTAAGATGC

Naïve mapping – much too slow

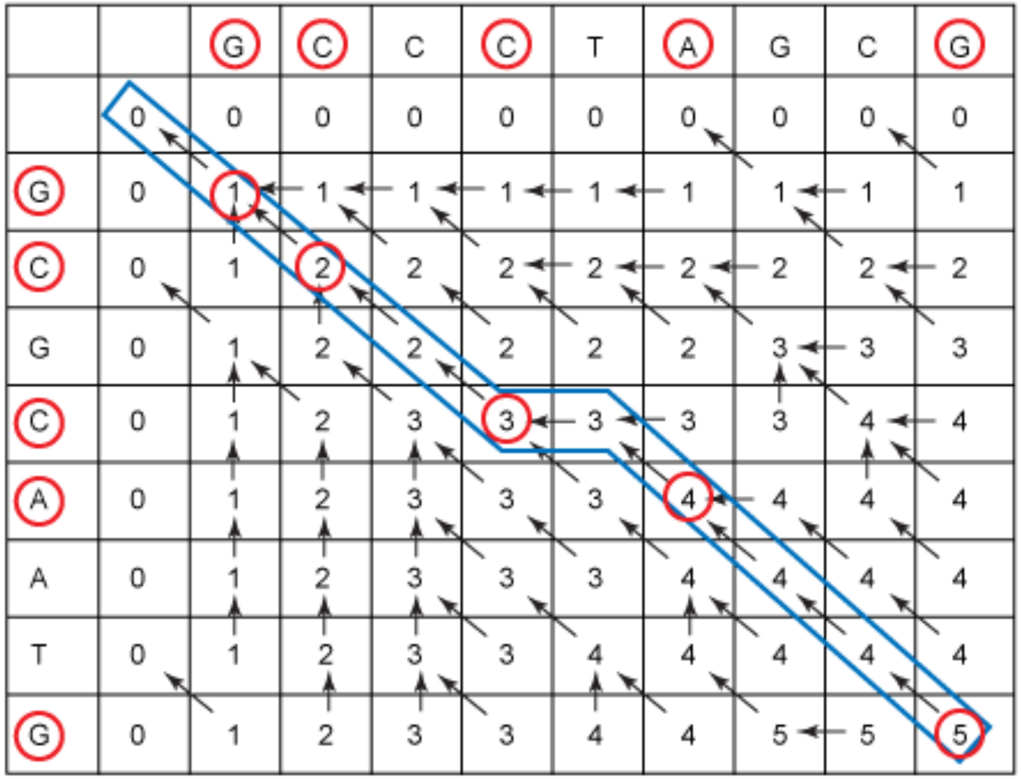
Try every position in the reference until match:

ACGTTACCGAATCGATCAAG
TCGA

m = query length
n = genome length

Time: *0*(mn)

Dynamic programming – still too slow



		G	C	C	C	T	A	G	C	G
	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1
C	0	1	2	2	2	2	2	2	2	2
G	0	1	2	2	2	2	2	3	3	3
C	0	1	2	3	3	3	3	3	4	4
A	0	1	2	3	3	3	4	4	4	4
A	0	1	2	3	3	3	4	4	4	4
T	0	1	2	3	3	4	4	4	4	4
G	0	1	2	3	3	4	4	5	5	5

Needleman-Wunsch algorithm is used in BLAST, but still runs in *0*(mn)

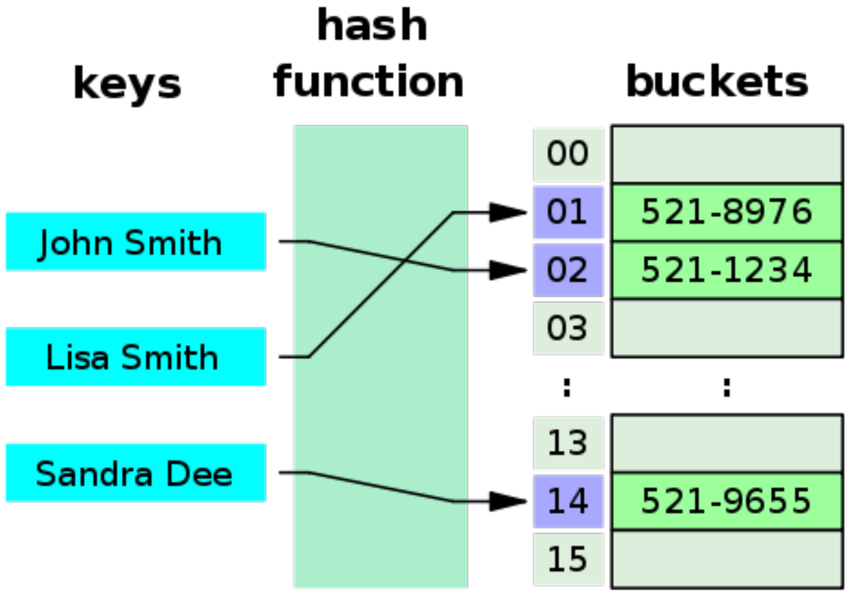
Mapping algorithms (http://www.ibm.com/developerworks/library/j-seqalign/)

Mapping by indexing

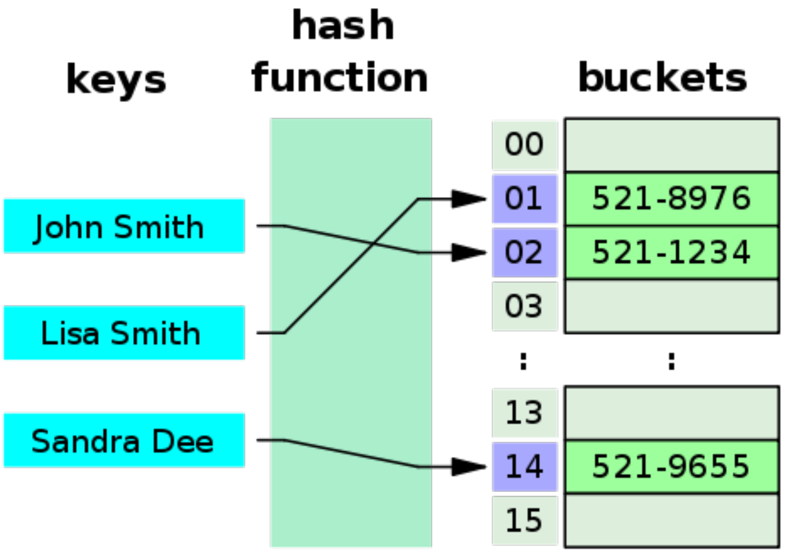
- Mapping millions of reads
  - Fast algorithms needed
  - Indexing speeds up searches
    - Hash tables
    - Suffix trees
    - Burrows-Wheeler transform and FM index

Seeds and hashes

- Hash tables
  - Represent a collection of key/value pairs that are organized based on the hash code of the key.



Seeds and hashes



keys

hash function

buckets

John Smith

Lisa Smith

Sandra Dee

00

01 521-8976

02 521-1234

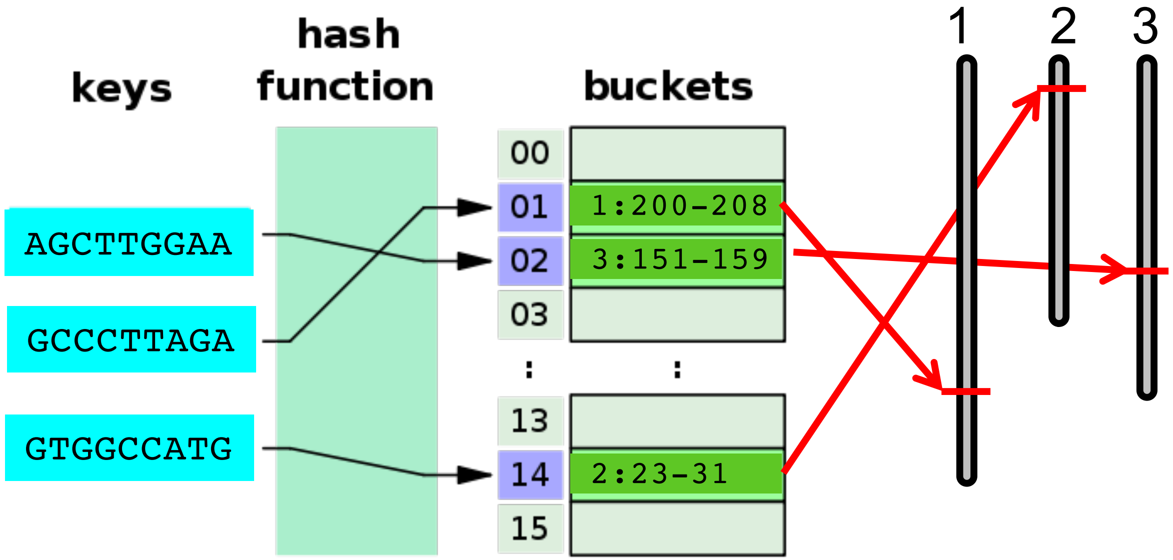
03

:

13

14 521-9655

15



keys

hash function

buckets

AGCTTGGAA

GCCCTTAGA

GTGGCCATG

00

01 1:200-208

02 3:151-159

03

:

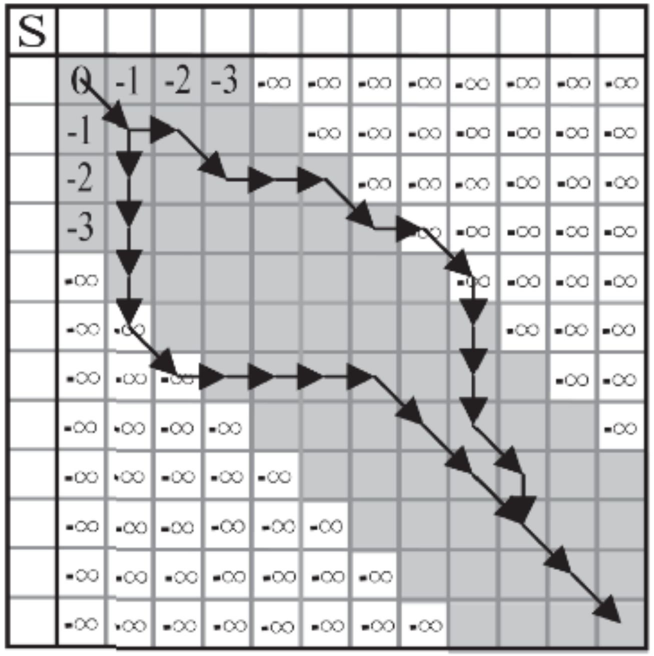
13

14 2:23-31

15

Seeds and hashes

Novoalign and ***Stampy***
use a combination of hash tables and dynamic programming to achieve sensitive and accurate alignments



S												
	0	-1	-2	-3	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞
	-1					-∞	-∞	-∞	-∞	-∞	-∞	-∞
	-2						-∞	-∞	-∞	-∞	-∞	-∞
	-3							-∞	-∞	-∞	-∞	-∞
	-∞								-∞	-∞	-∞	-∞
	-∞	-∞								-∞	-∞	-∞
	-∞	-∞	-∞								-∞	-∞
	-∞	-∞	-∞	-∞								-∞
	-∞	-∞	-∞	-∞	-∞							
	-∞	-∞	-∞	-∞	-∞	-∞						
	-∞	-∞	-∞	-∞	-∞	-∞	-∞					
	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞				

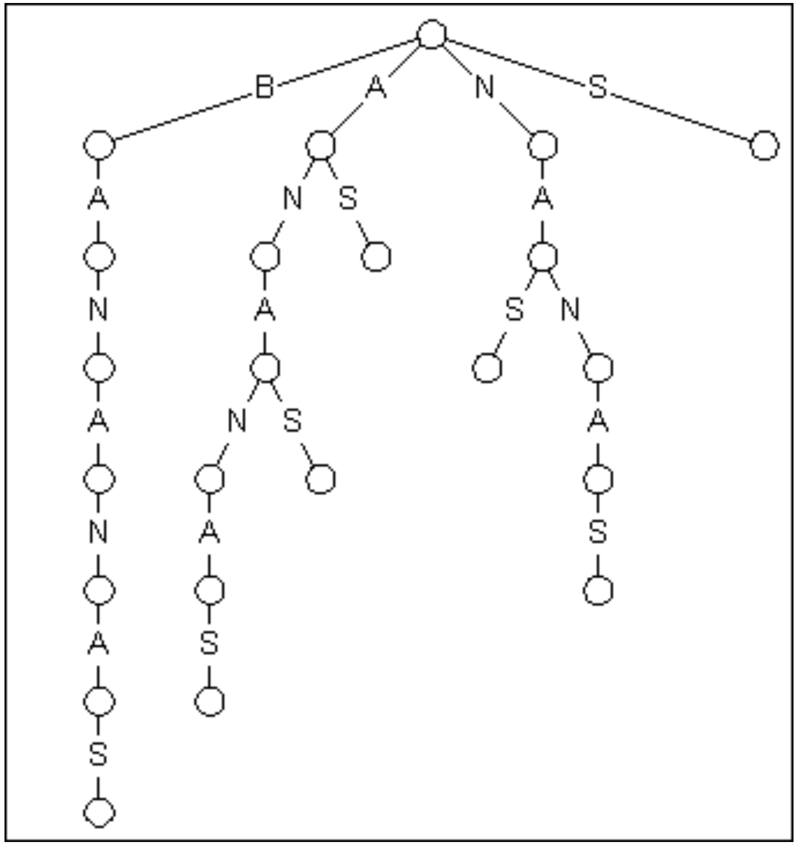
Banded affine gap alignment

Suffix trees

- Exact matches for any substring of L length with just L comparisons. Independent of reference length.

ananas

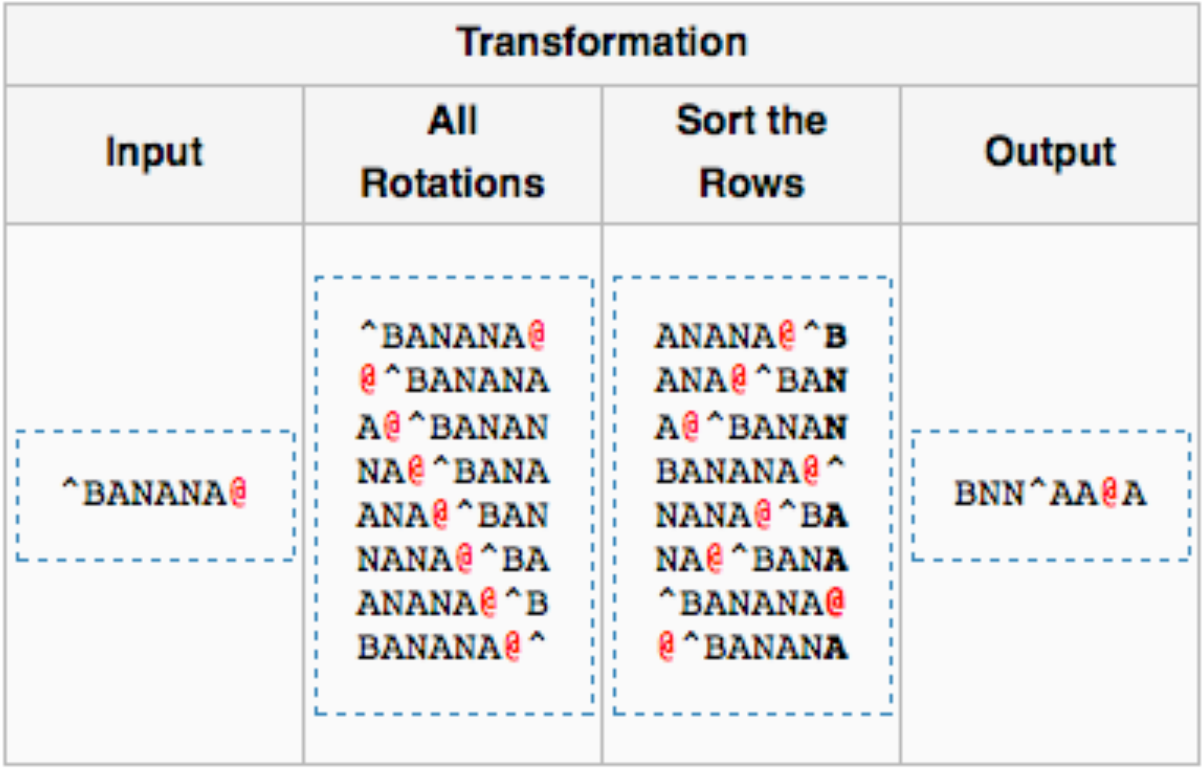
anna



Burrows-Wheeler transform & FM-index

- Suffix trees built from Burrows-Wheeler transformed data are much more efficient
- FM-index - a compressed suffix tree
- http://www.di.unipi.it/~ferragin/Libraries/fmindexV2/index.html

Burrows-Wheeler transform & FM-index



Transformation			
Input	All Rotations	Sort the Rows	Output
^BANANA@	^BANANA@ @^BANANA A@^BANAN NA@^BANA ANA@^BAN NANA@^BA ANANA@^B BANANA@^	ANANA@^B ANA@^BAN A@^BANAN BANANA@^ NANA@^BA NA@^BANA ^BANANA@ @^BANANA	BNN^AA@A

Making fast mapping possible

- Reduce global alignment problem to exact matching
- Use efficient data structures - hash tables, suffix trees, Burrows-Wheeler transform with FM-index
- Limit the use of dynamic programming to short local alignments defined by exact matching seeds

Lecture overview

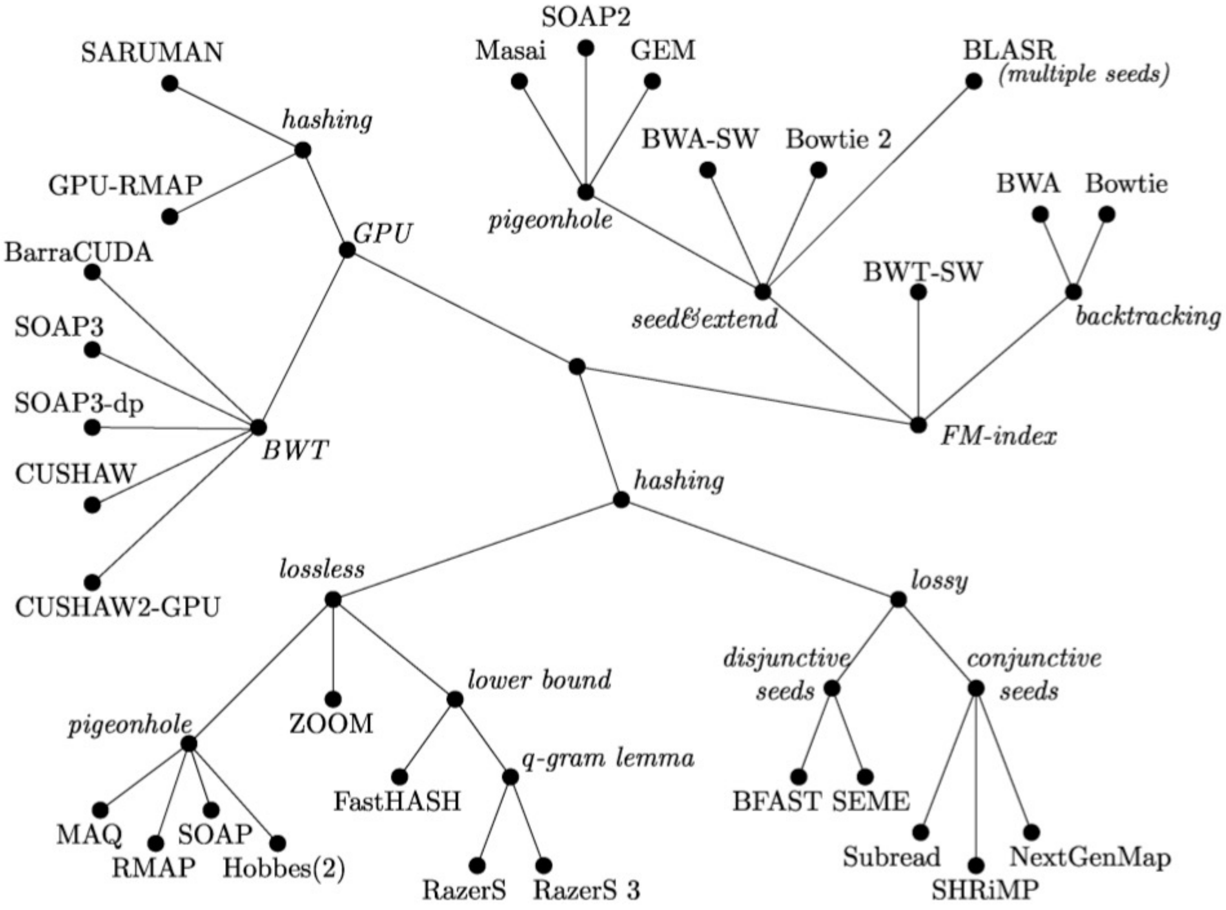
- Mapping algorithms
- Mapping software

Software overview

Bowtie2	FM-index does long reads and gapped alignments
BWA mem	FM-index, long reads, automatic local/global alignment
NextGenMap	Hashing ref, high speed and good sensitivity
Minimap2	Minimizers and hashing

Use BWA mem or Bowtie2 for short reads.
Minimap2 or NextGenMap (PMID: 23975764) might also be interesting to try.

Software overview



Mapping algorithms

PMID: 28502990

Long NGS reads

NGS reads are no longer necessarily short reads

Multiple seeds per read

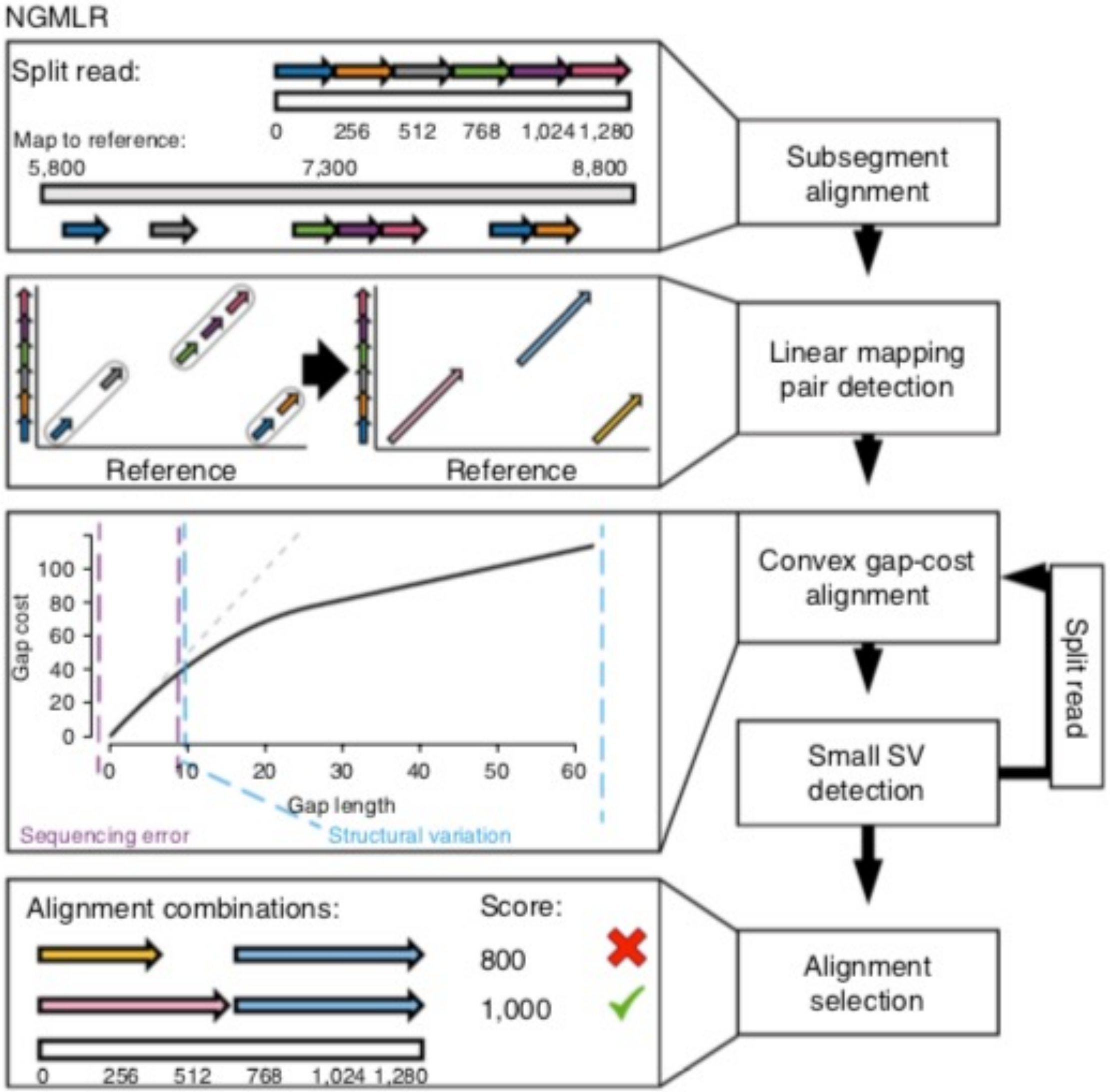
Bowtie2: FM-index + dynamic programming

bwa mem: FM-index + re-seeding + seed chaining
and chain filtering +
global vs. local alignment assessment

NGMLR: Hashing + split reads + alignment combinations

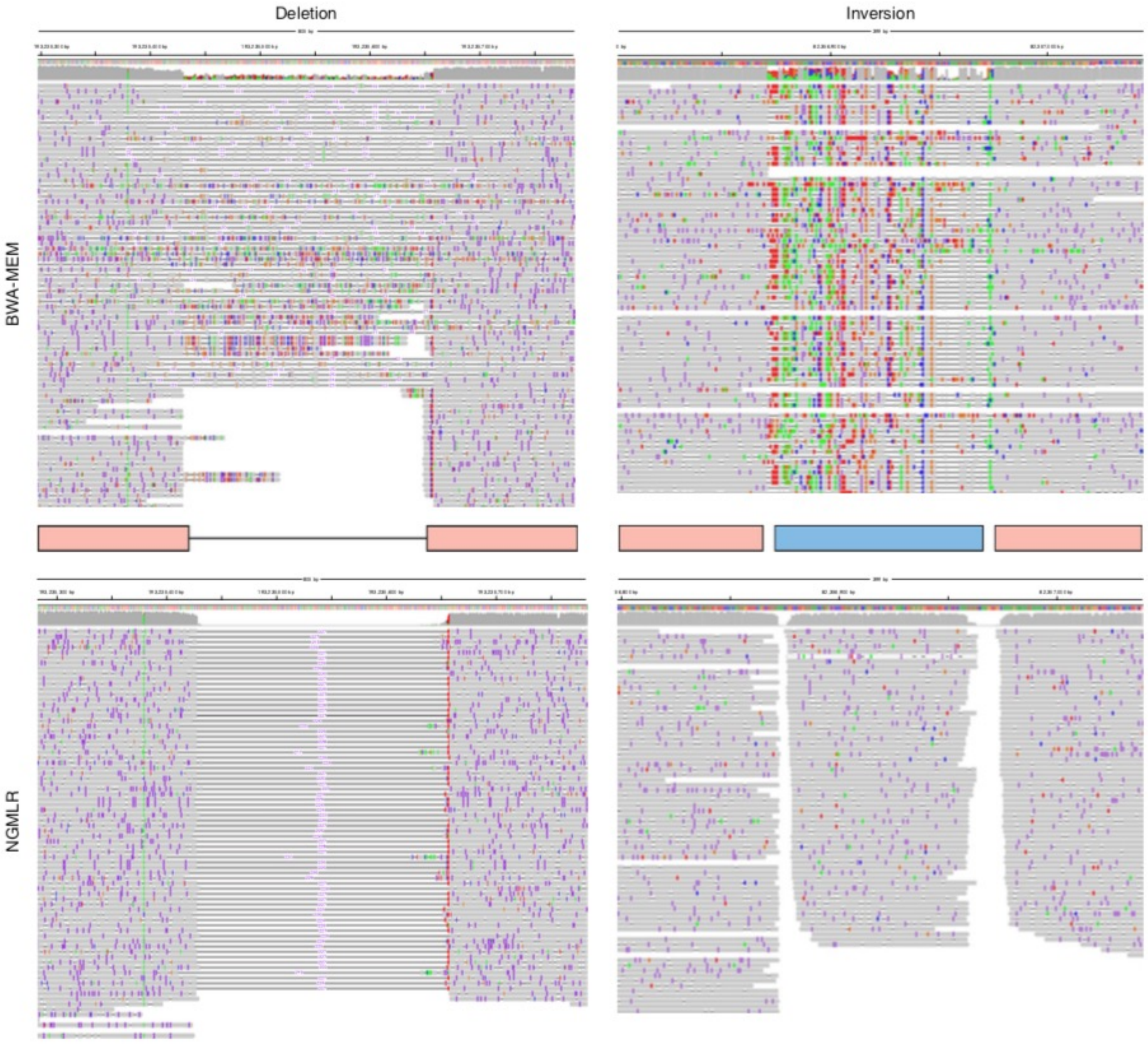
Minimap2: Minimizers and hashing. PacBios official choice
https://github.com/PacificBiosciences/pbmm2

NGMLR - NextGenMapLongReads



PMID: 29713083

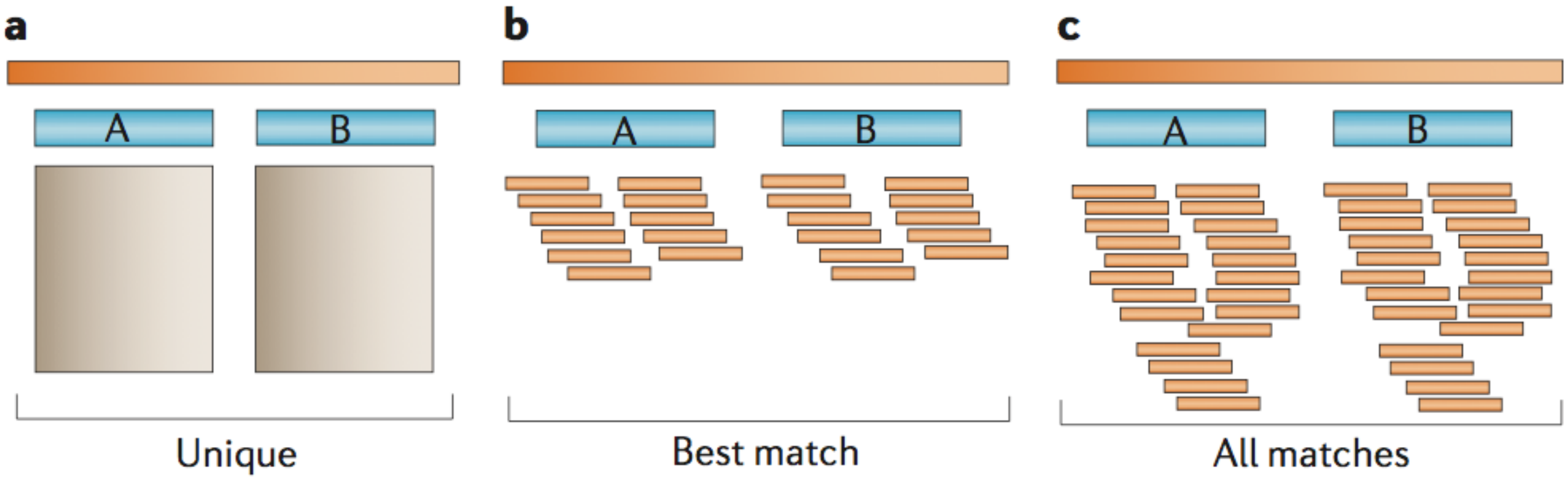
NGMLR



Mapping algorithms

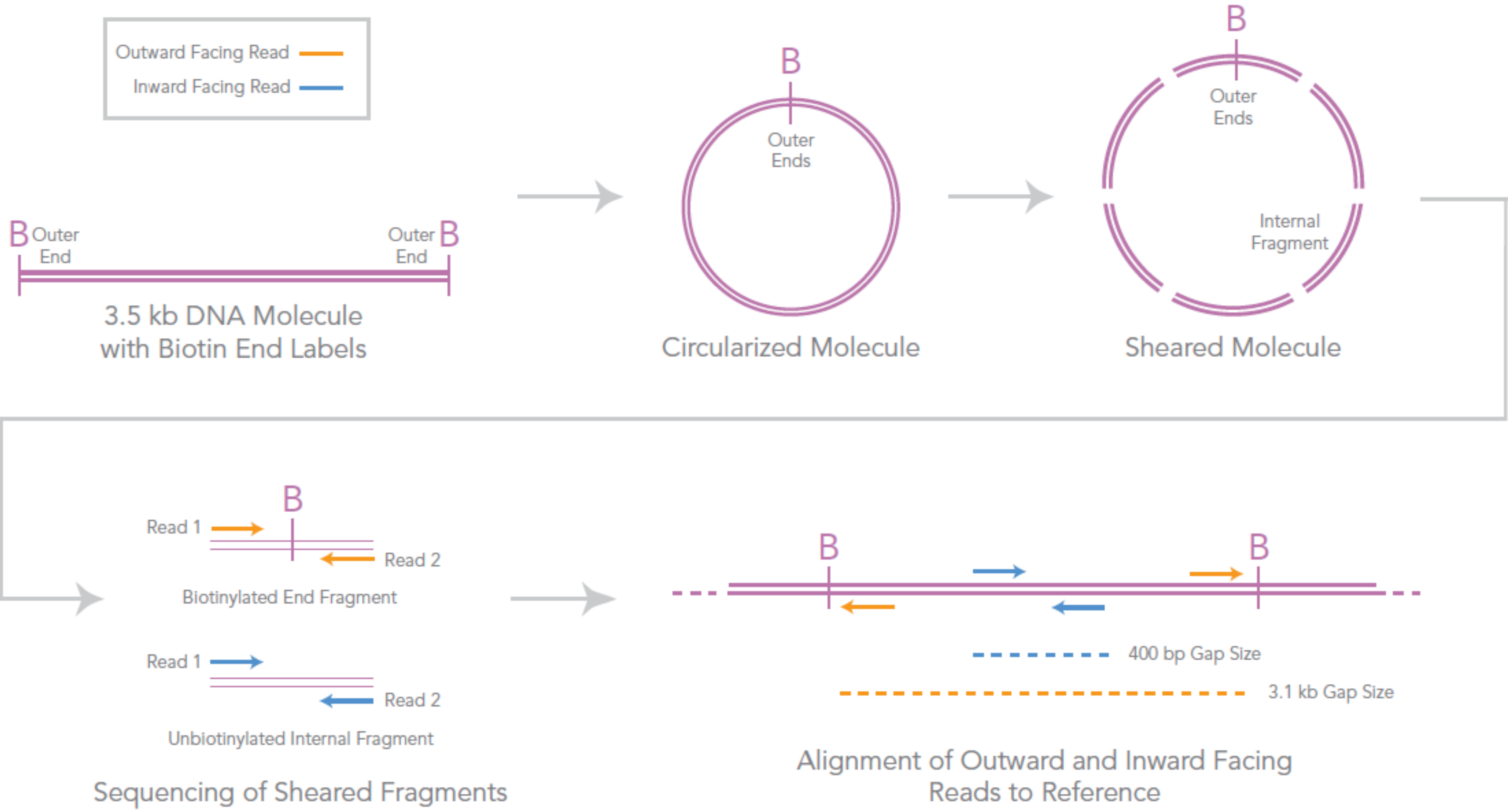
PMID: 29713083

Repeats



PMID: 22124482

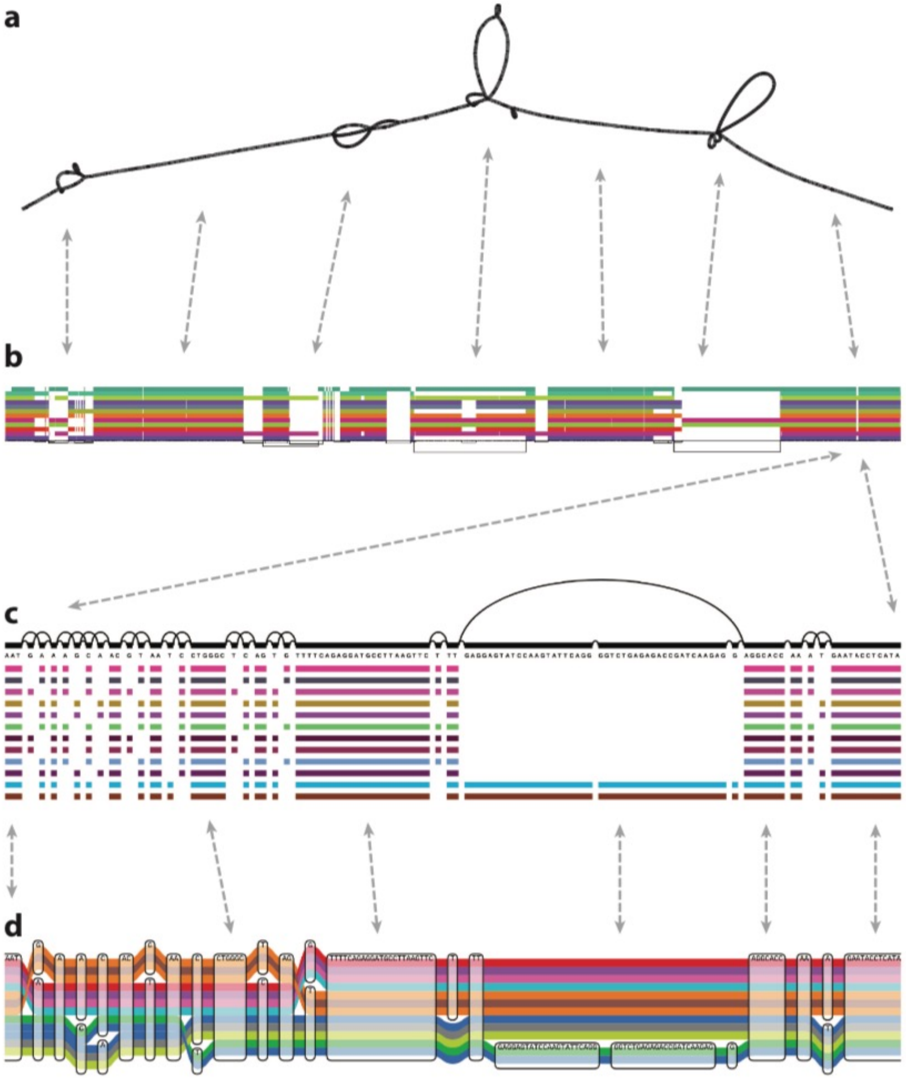
Library preparation and read orientation



Library preparation

Illumina Mate Pair Library v2 Sample Preparation Guide

Graph-based approaches



The practical haplotype graph

a platform for storing and using pangenomes for imputation

10.1101/2021.08.27.457652

https://www.annualreviews.org/doi/10.1146/annurev-genom-120219-080406

Lecture overview

- Mapping algorithms
- Mapping software
- Input format - FASTQ

Input format - FASTQ

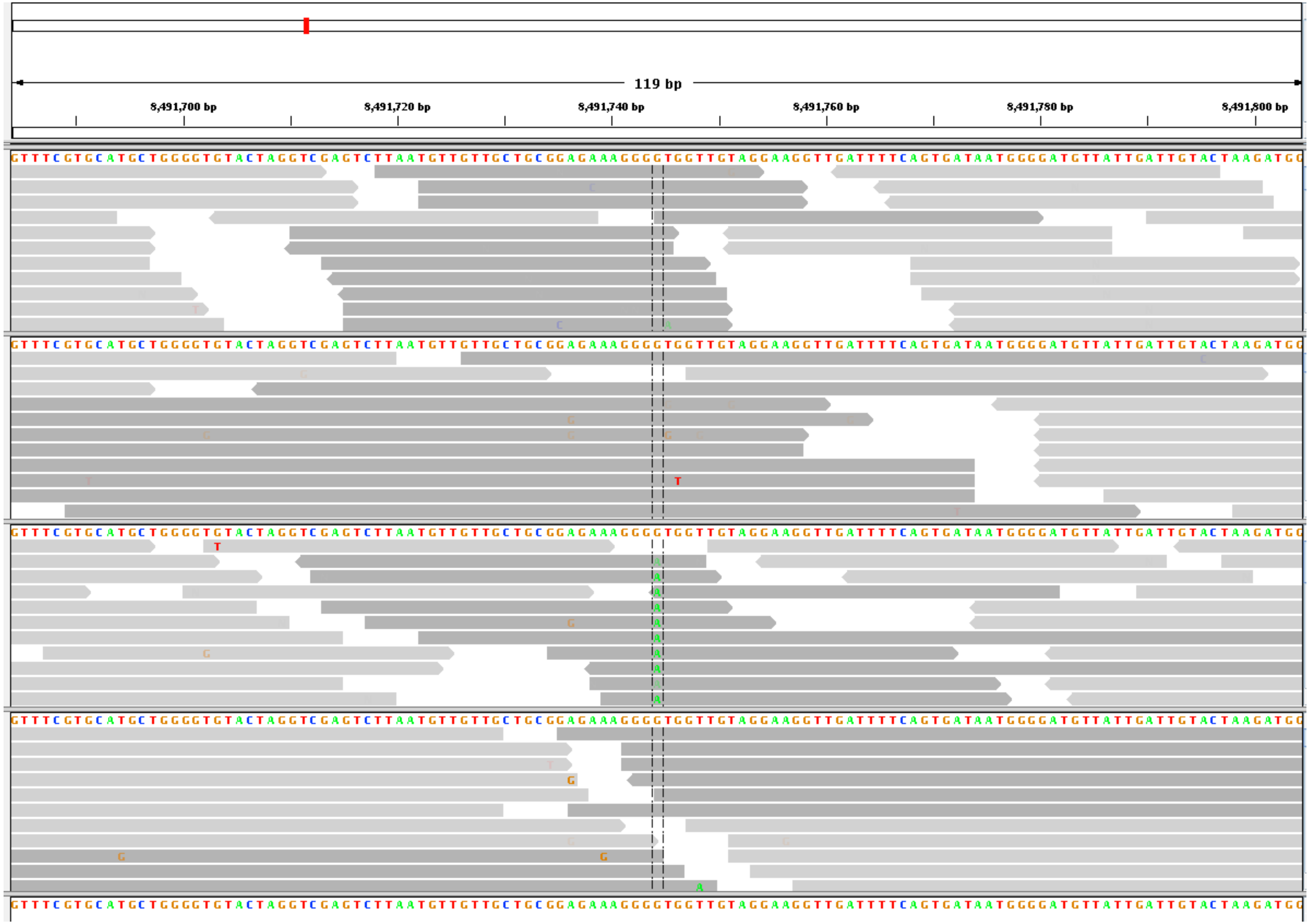
```
Read name:          @HWI-EAS133_0001:6:1:2:987#0/1
Read sequence:      TCACACCACTGACAAGTNTGACCGAATACAGACAAA
Read name:          +HWI-EAS133_0001:6:1:2:987#0/1
Base call quality:  aa`aaaaaab`_aaa`Bab``^aaaaaaaaaa_a`]
```

PMID: 20015970

Lecture overview

- Mapping algorithms
- Mapping software
- Input format - FASTQ
- Output format - SAM

Mapping reads example


119 bp
8,491,700 bp
8,491,720 bp
8,491,740 bp
8,491,760 bp
8,491,780 bp
8,491,800 bp
GTTTCGTGCATGCTGGGGTGTACTAGGTCGAGTCTTAATGTTGTTGCTGCGGAGAAAGGGGTGGTTGTAGGAAGGTTGATTTTCAGTGATAATGGGGATGTTATTGATTGTACTAAGATGG

Output format - SAM

Header section @HD:

@SQ Reference sequence dictionary

@RG Read group

@PG Program

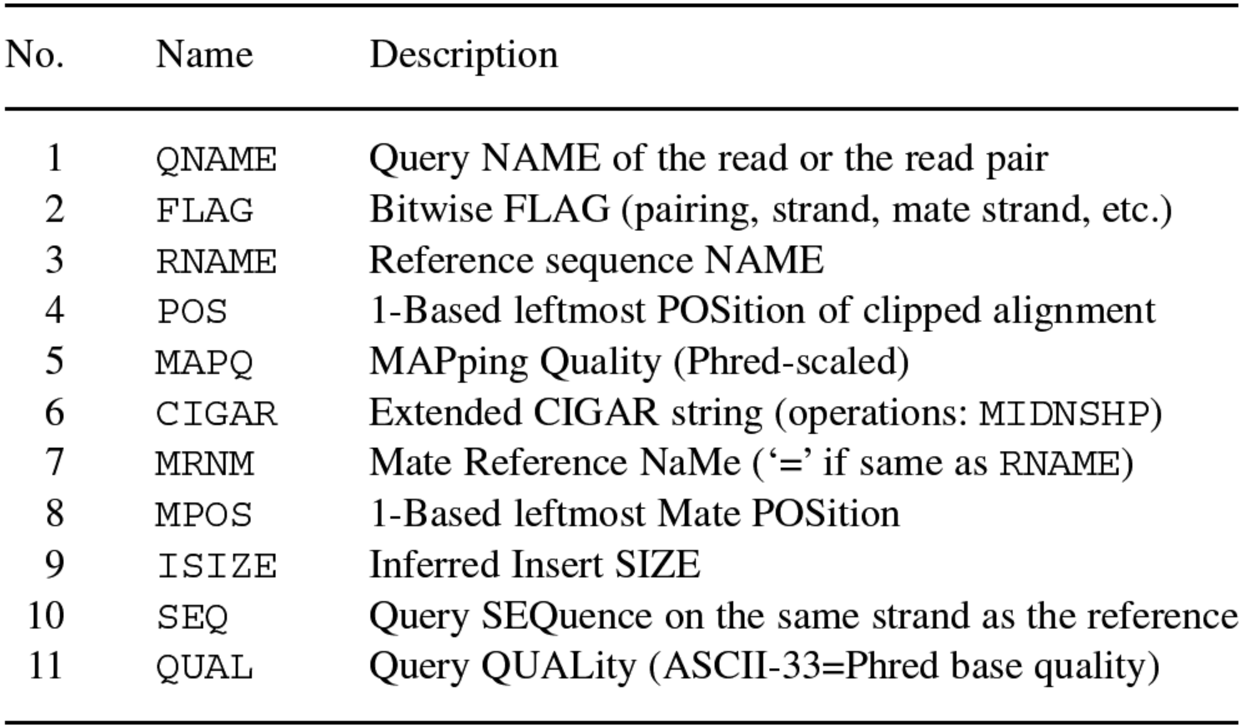
@CO One-line text comment

PMID: 19505943 and SAM format specification

Output format - SAM

Sequence Alignment/Map format

Table 1. Mandatory fields in the SAM format



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

PMID: 19505943

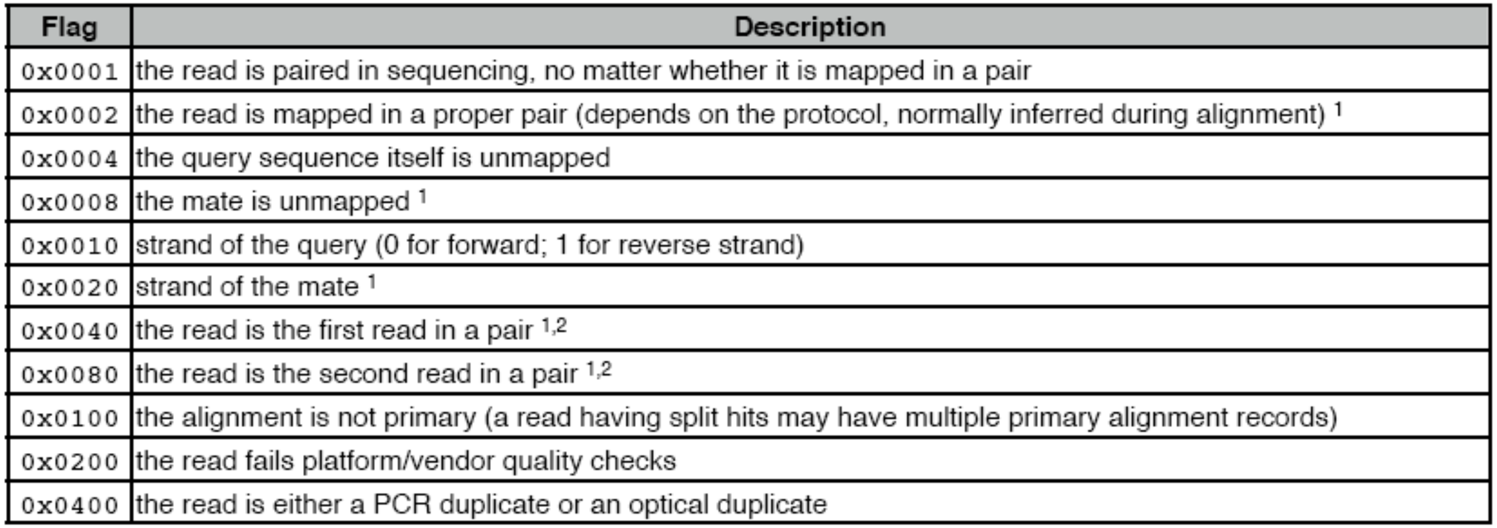
Output format - SAM

1	QNAME	HWI-ST476:149:D1BMPACXX:8:2205:8604:71436
2	FLAG	163
3	RNAME	chr3_1000001
4	POS	**4**
5	MAPQ	60
6	CIGAR	101M
7	MRNM	=
8	MPOS	**278**
9	ISIZE	**375**
10	SEQ	TTCCAATCTTCACAATTCATTTTTTCA[…]
11	QUAL	CCFFFFFHHHHHJJJIJJIJJIJJJJJJI[…]
12	OPT	XT:A:U NM:i:0 SM:i:37 AM:i:37 X0:i:1 X1:i:0 XM:i:0 XO:i:0 XG:i:0 MD:Z:101

How is the insert size (template length) calculated?

PMID: 19505943 and SAM format specification

Output format - SAM



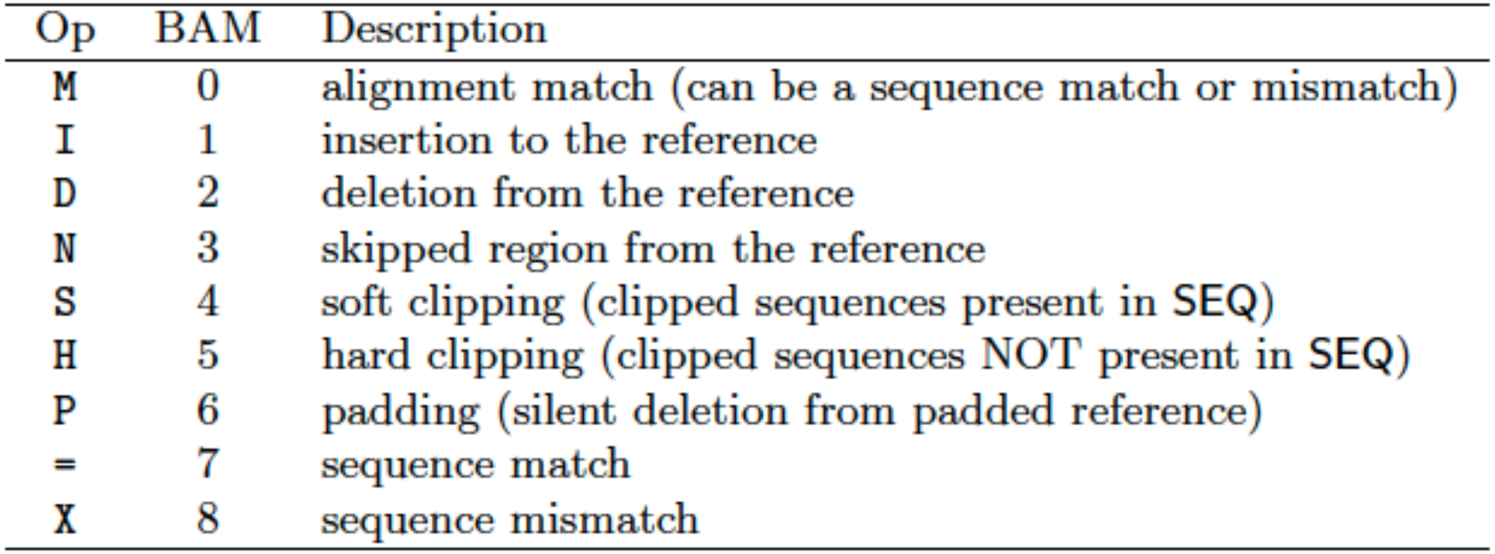
	Flag	Description
1	0x0001	the read is paired in sequencing, no matter whether it is mapped in a pair
2	0x0002	the read is mapped in a proper pair (depends on the protocol, normally inferred during alignment) [1]
4	0x0004	the query sequence itself is unmapped
8	0x0008	the mate is unmapped [1]
16	0x0010	strand of the query (0 for forward; 1 for reverse strand)
32	0x0020	strand of the mate [1]
64	0x0040	the read is the first read in a pair [1,2]
128	0x0080	the read is the second read in a pair [1,2]
256	0x0100	the alignment is not primary (a read having split hits may have multiple primary alignment records)
512	0x0200	the read fails platform/vendor quality checks
1024	0x0400	the read is either a PCR duplicate or an optical duplicate

According to FLAG 163 (=128+32+2+1), the read mapped is the

- second read in the pair (128)
- regarded as properly paired (1 + 2)
- its mate is mapped to the reverse strand (32)

PMID: 19505943 and SAM format specification

Output format - SAM

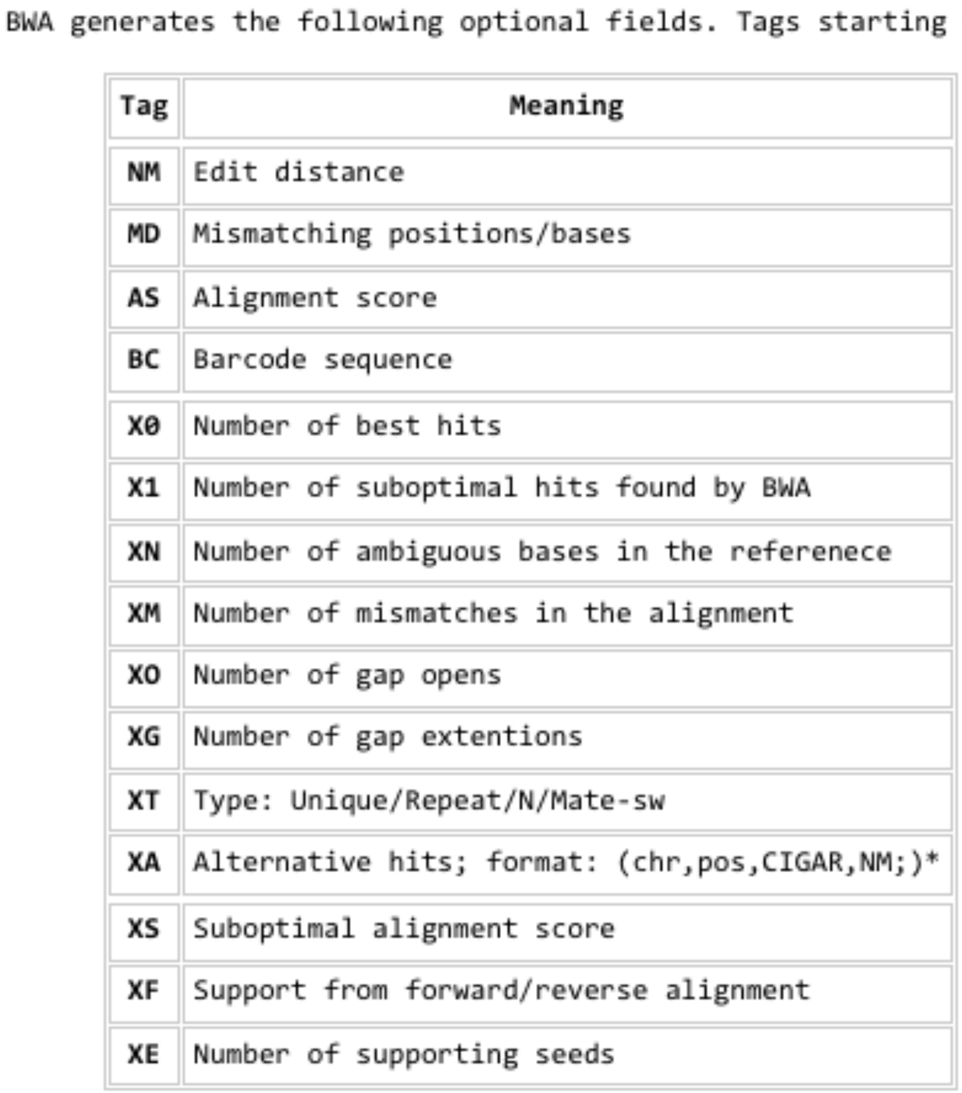


Op	BAM	Description
M	0	alignment match (can be a sequence match or mismatch)
I	1	insertion to the reference
D	2	deletion from the reference
N	3	skipped region from the reference
S	4	soft clipping (clipped sequences present in SEQ)
H	5	hard clipping (clipped sequences NOT present in SEQ)
P	6	padding (silent deletion from padded reference)
=	7	sequence match
X	8	sequence mismatch

The extended CIGAR string...

PMID: 19505943 and SAM format specification

Output format - SAM

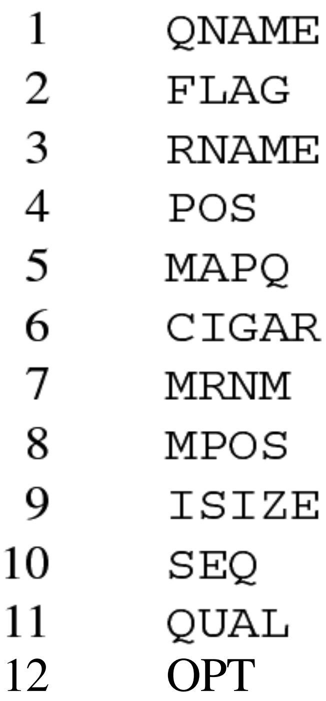


BWA generates the following optional fields. Tags starting with ‘X’ are specific to BWA.

Tag	Meaning
NM	Edit distance
MD	Mismatching positions/bases
AS	Alignment score
BC	Barcode sequence
X0	Number of best hits
X1	Number of suboptimal hits found by BWA
XN	Number of ambiguous bases in the referenece
XM	Number of mismatches in the alignment
XO	Number of gap opens
XG	Number of gap extentions
XT	Type: Unique/Repeat/N/Mate-sw
XA	Alternative hits; format: (chr,pos,CIGAR,NM;)*
XS	Suboptimal alignment score
XF	Support from forward/reverse alignment
XE	Number of supporting seeds

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

Output format - SAM



1	QNAME	HWI-ST476:149:D1BMPACXX:8:2205:8604:71436
2	FLAG	163
3	RNAME	chr3_1000001
4	POS	**4**
5	MAPQ	60
6	CIGAR	101M
7	MRNM	=
8	MPOS	**278**
9	ISIZE	**375**
10	SEQ	TTCCAATCTTCACAATTCATTTTTTCA[…]
11	QUAL	CCFFFFFHHHHJJJIJJIJJIJJJJJJI[…]
12	OPT	XT:A:U NM:i:0 SM:i:37 AM:i:37 X0:i:1 X1:i:0 XM:i:0 XO:i:0 XG:i:0 MD:Z:101

PMID: 19505943 and SAM format specification

Output format – SAM/BAM

BAM files are binary versions of the SAM files.

These take up less space and can be indexed for quick lookups and data extraction.

They are not human readable like the SAM files.

CRAM files are compressed binary alignment files, 30-60% smaller than corresponding BAMs.

SAM/BAM links:
http://www.htslib.org/doc/
http://samtools.github.io/hts-specs/

PMID: 19505943 and SAM format specification