

Introduction to NGS Analysis

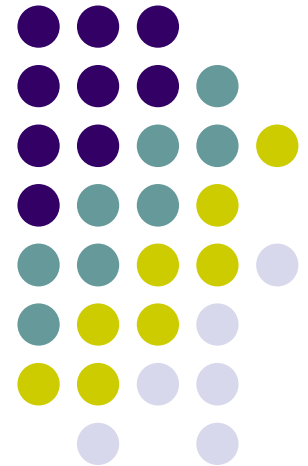
Anna Battenhouse

abattenhouse@utexas.edu

February, 2019

Associate Research Scientist
Center for Systems and Synthetic Biology (CSSB)
Ed Marcotte & Vishwanath Iyer labs

Center for Biomedical Research Support (CBRS)
Bioinformatics Consulting Group (BCG)
Biomedical Research Computing Facility (BRCF)
Genome Sequencing & Analysis Facility (GSAF)



NGS Workflow

core processes

fastq

upstream processes

experimental design

DNA/RNA isolation

library preparation

next-gen sequencing

delivery of raw reads

QC raw read sequences

yes

map reads to reference

alignment metrics & QC

basic analysis
(e.g. coverage, genes)

further analysis & significance determination
(e.g. FPKM, peak or variant calls)

confident calls

has reference?

reference assembly

fasta

BAM

bed, gff, vcf, etc.

no

assembly
(genome or transcriptome)

metrics & QC

downstream processes

differential analysis

annotation

motif analysis

custom analysis

Outline

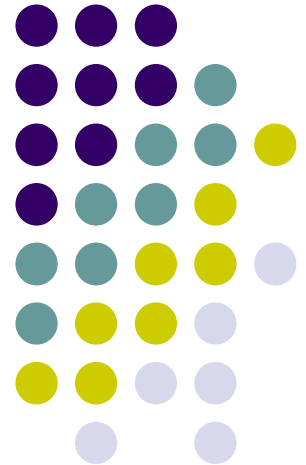


1. Overview of sequencing technologies
2. NGS concepts and terminology
3. The FASTQ format and raw data QC & preparation
4. Alignment to a reference

Part 1:

Overview of Sequencing Technologies

- High-throughput (“next gen”) sequencing
- Illumina short-read sequencing
- Long read sequencing



NGS Workflow

core processes

fastq

QC raw read sequences

yes

map reads to reference

alignment metrics & QC

basic analysis
(e.g. coverage, genes)

further analysis & significance determination
(e.g. FPKM, peak or variant calls)

confident calls

has reference?

fasta

BAM

bed, gff, vcf, etc.

no

assembly
(genome or transcriptome)

metrics & QC

downstream processes

differential analysis

annotation

motif analysis

custom analysis

upstream processes

experimental design

DNA/RNA isolation

library preparation

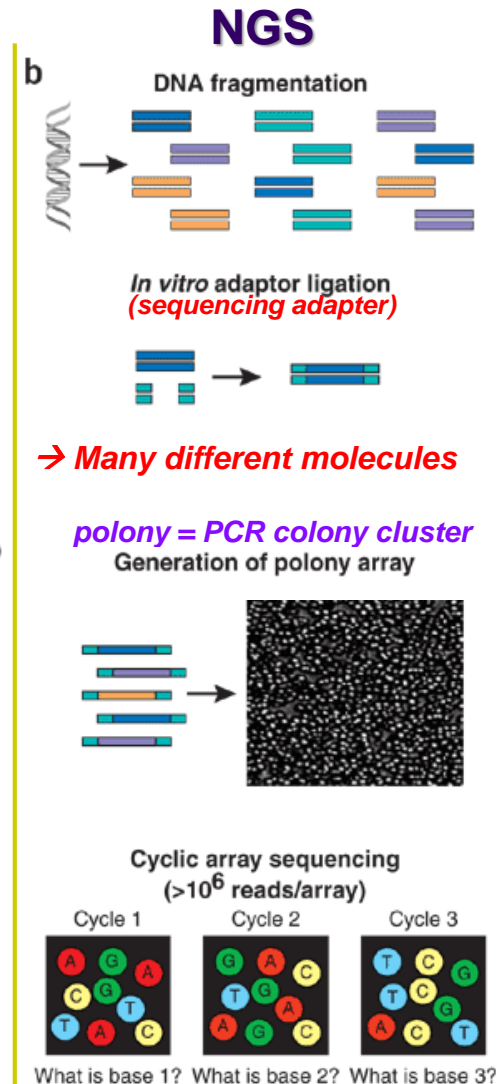
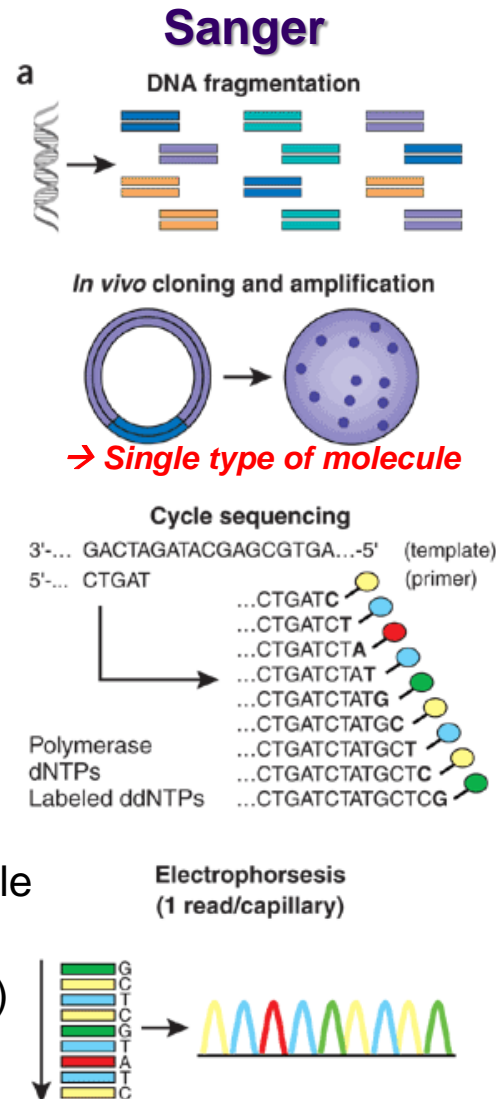
next-gen sequencing

delivery of raw reads

“Next Generation” sequencing



- Massively parallel
 - simultaneously sequence “library” of *millions* of different DNA fragments
- *PCR colony clusters* generated
 - individual template DNA fragments titrated onto a flowcell to achieve inter-fragment separation
 - PCR “bridge amplification” creates *clusters* of identical molecules
- *Sequencing by synthesis*
 - fluorescently-labeled dNTs added
 - incorporation generates signal
 - flowcell image captured after each cycle
 - images computationally converted to base calls (including a quality score)



Shendure et al, Nature Biotechnology. 2008.

<https://www.nature.com/articles/nbt1486>

“Next Generation” sequencing (2nd generation)

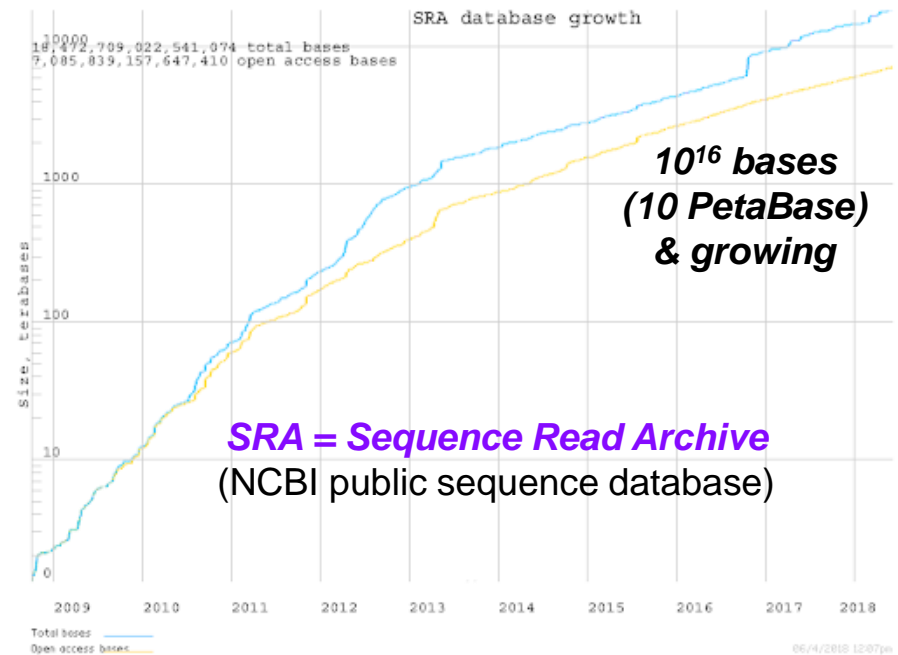
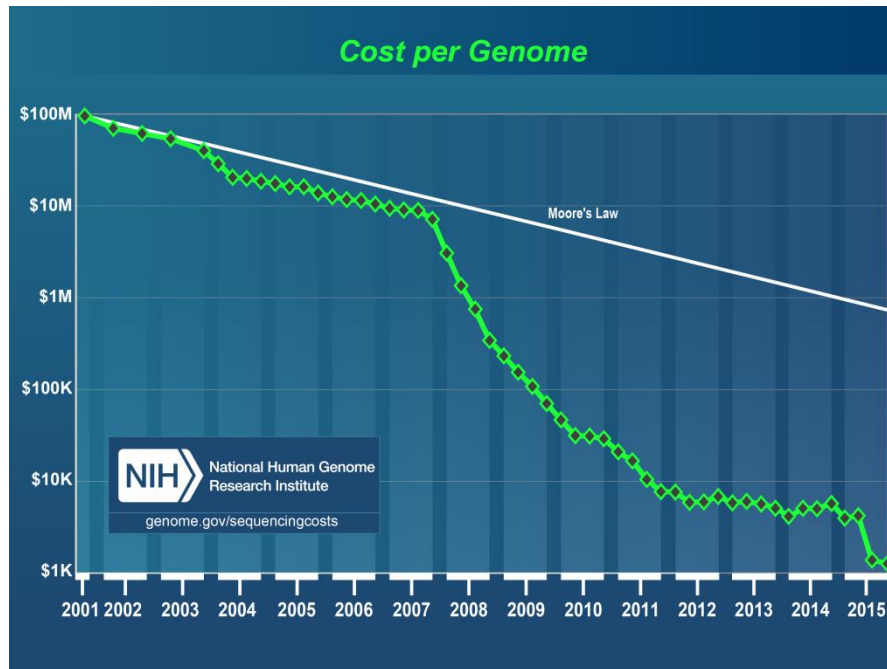


- Pro's:

- much faster!
- much lower cost!
- both deeper and wider coverage!

- Con's:

- data deluge!
- storage requirements!
- analysis lags!



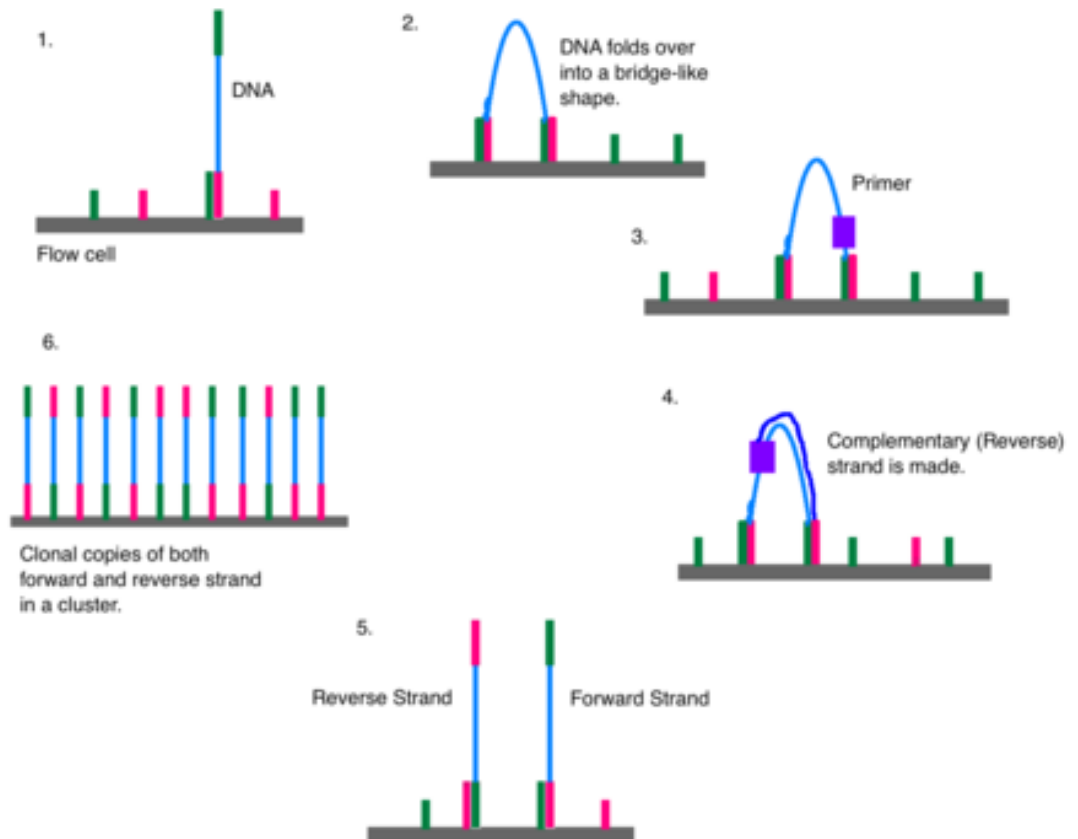
Illumina sequencing



1. Library preparation
2. **Cluster generation via bridge amplification**
3. Sequencing by synthesis
4. Image capture
5. Convert to base calls

Short Illumina video
(<https://tinyurl.com/hvnmwjb>)

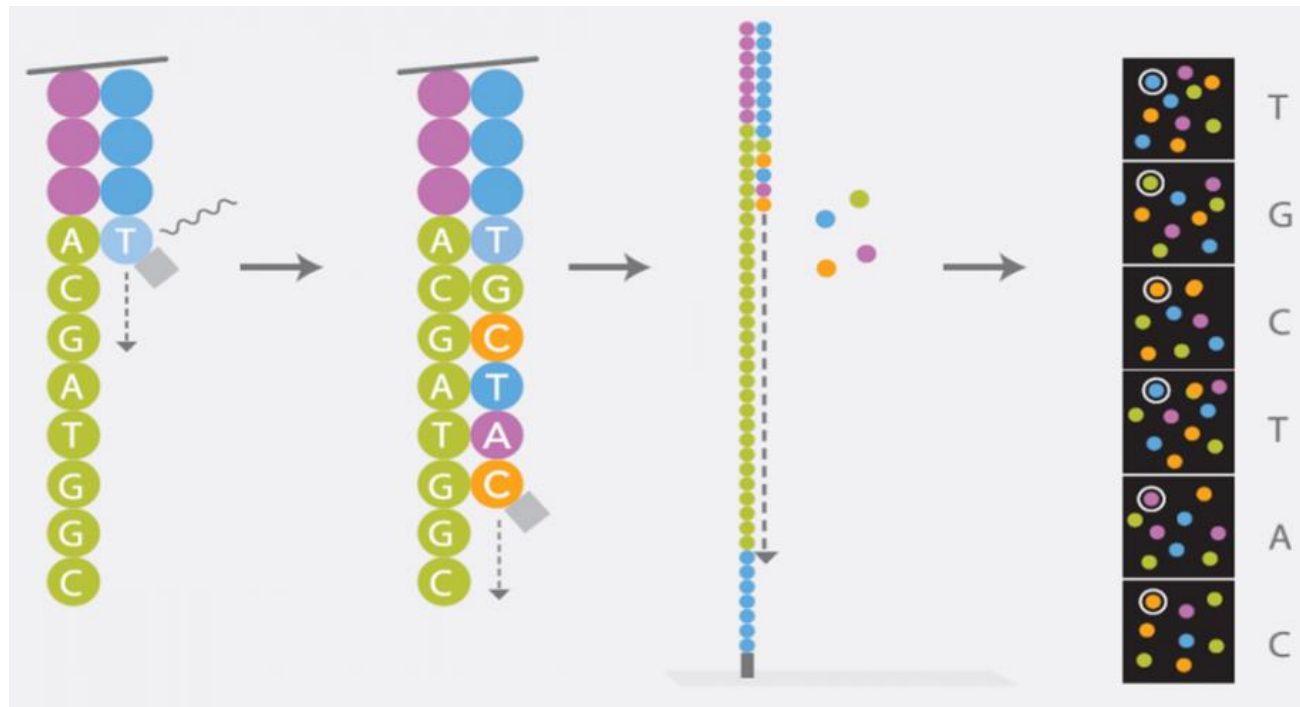
- Note
 - 2 PCR amplifications performed
 1. during **library preparation**
 2. during **cluster generation**
 - **amplification always introduces bias!**



Illumina sequencing



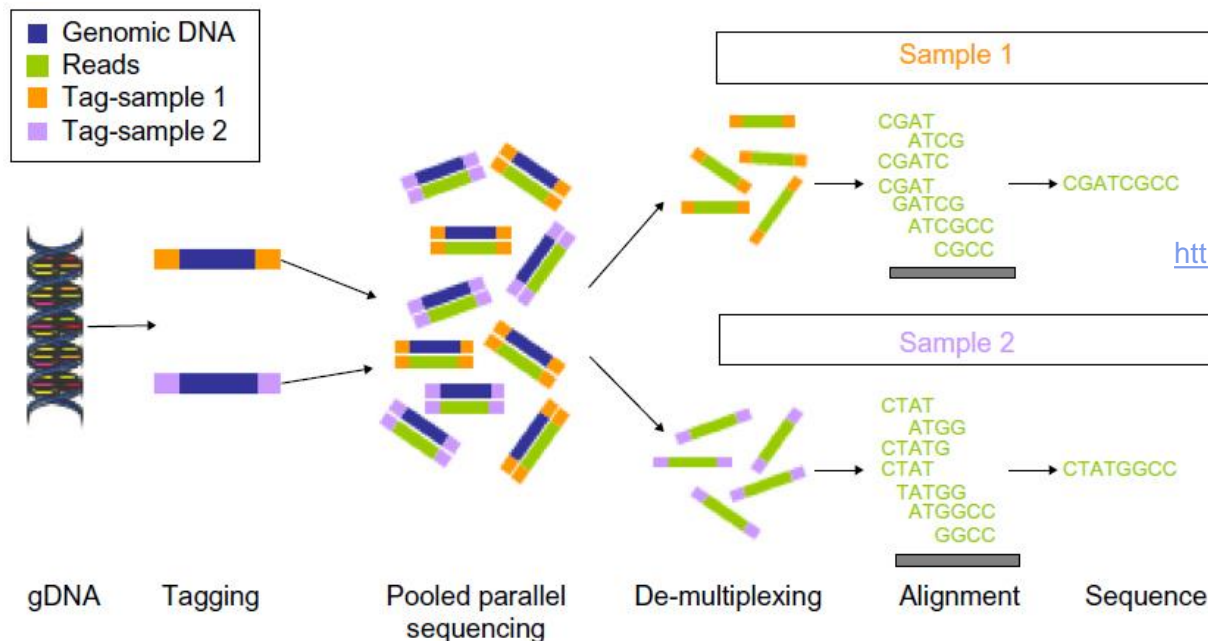
1. Library preparation
2. Cluster generation via bridge amplification
3. *Sequencing by synthesis*
4. *Image capture*
5. *Convert to base calls*



Multiplexing



- Illumina sequencers have one or more flowcell “lanes”, each of which can generate millions of reads
 - ~20M reads/lane for MiSeq, ~250M reads/lane for HiSeq
- When less than a full flowcell lane is needed, multiple samples with different **barcodes** (a.k.a. **indexes**) can be run on the same lane
 - 6-8 bp **library barcode** attached to DNA library fragments
 - data from sequencer must be **demultiplexed** to determine which reads belong to which library



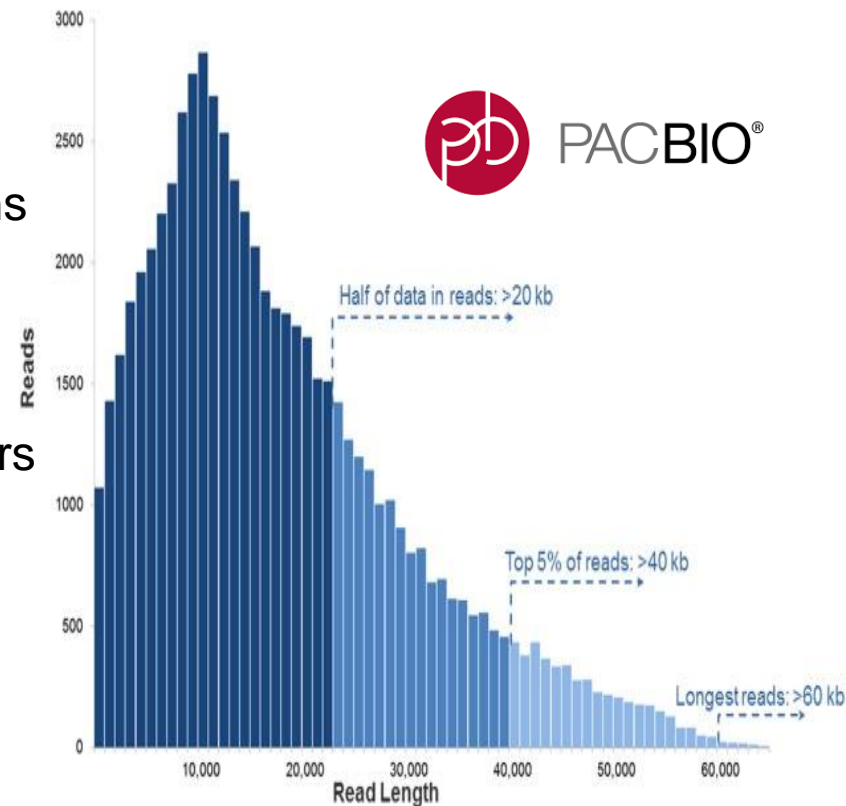
<https://doi.org/10.2147/BLCTT.S51503>

Long read sequencing

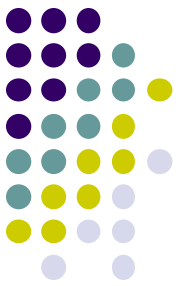


- Short read technology limitations
 - PCR amplification bias
 - short reads are difficult to assemble
 - e.g., too short to span a long repeat region
 - difficult to detect large structural variations like inversions

- Newer “*single molecule*” sequencing
 - sequences *single molecules*, not clusters
 - allows for *much* longer reads (multi-Kb!)
 - no signal wash-out due to lack of synchronization among cluster molecules
 - *but*: reads have high error rate
 - 10-15% vs <1% for Illumina
 - one amplification usually still required (during library prep)



Long read sequencing



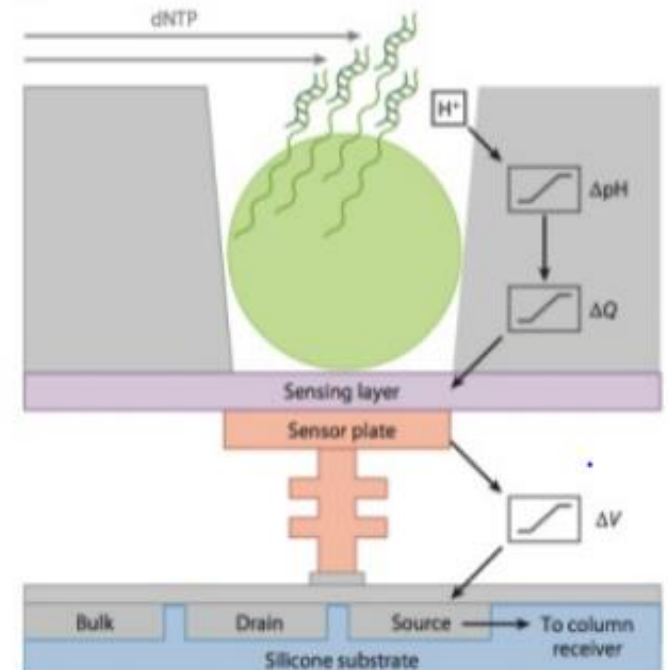
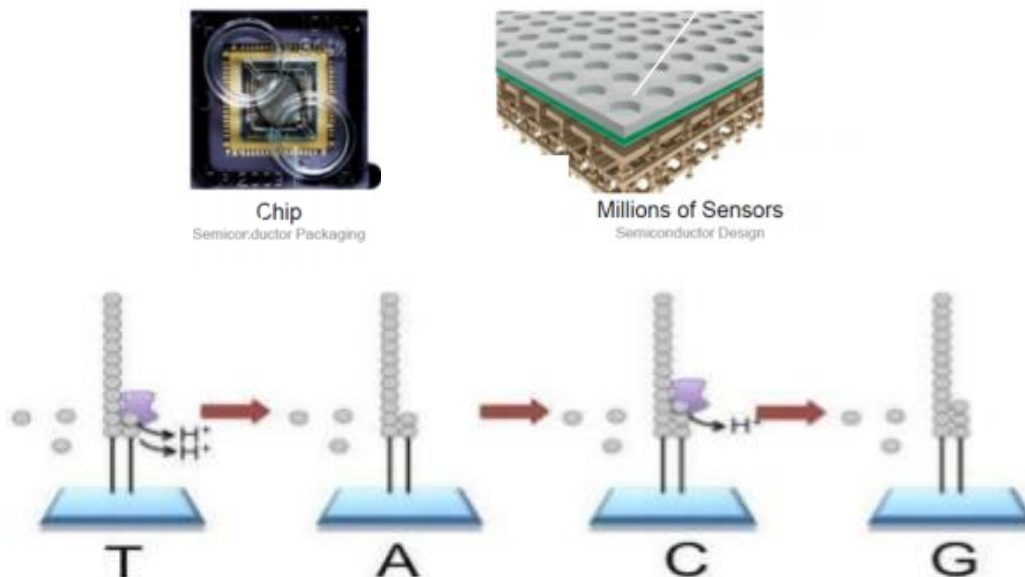
- Oxford Nanopore Ion Torrent system

- <https://nanoporetech.com/>
- DNA “spaghetti’s” through tiny protein pores
- Addition of different bases produces different pH changes
 - measured as different changes in electrical conductivity
- MinION is hand-held, starter kit costs ~\$1,000 – including reagents!

ion torrent



by *life* technologies™



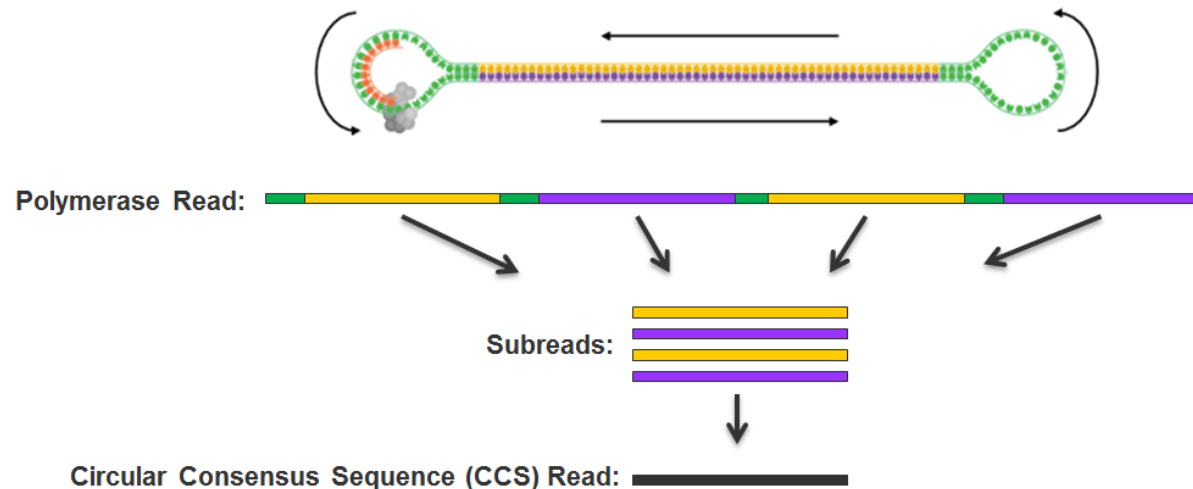
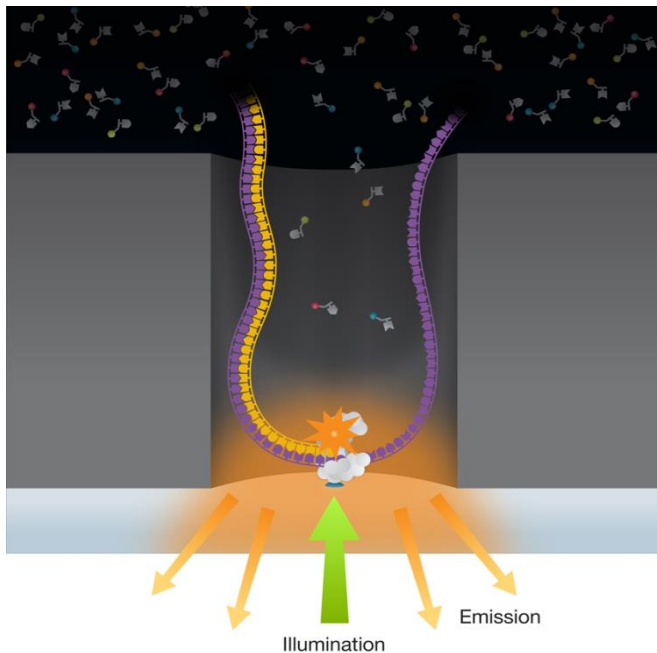
Long read sequencing



- PacBio SMRT system



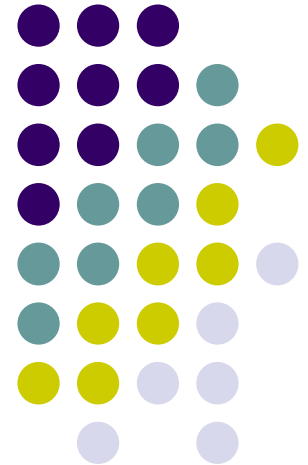
- <http://www.pacb.com/smrt-science/smrt-sequencing/>
- Sequencing by synthesis in **Zero-Mode Waveguide** (ZMW) wells
- DNA is circularized then repeatedly sequenced to achieve “consensus”



Part 2:

NGS Concepts & Terminology

- Experiment types & library complexity
- Sequencing terminology
- Sequence duplication issues



NGS Workflow

core processes

fastq

QC raw read sequences

yes

map reads to reference

alignment metrics & QC

basic analysis
(e.g. coverage, genes)

further analysis & significance determination
(e.g. FPKM, peak or variant calls)

confident calls

has reference?

reference assembly

fasta

BAM

bed, gff, vcf, etc.

no

assembly
(genome or transcriptome)

metrics & QC

downstream processes

differential analysis

annotation

motif analysis

custom analysis

upstream processes

experimental design

DNA/RNA isolation

library preparation

next-gen sequencing

delivery of raw reads



Library Complexity

Library complexity (diversity)

is a measure of the number of ***distinct molecular species*** in the library.

Many different molecules → *high complexity*

Few different molecules → *low complexity*

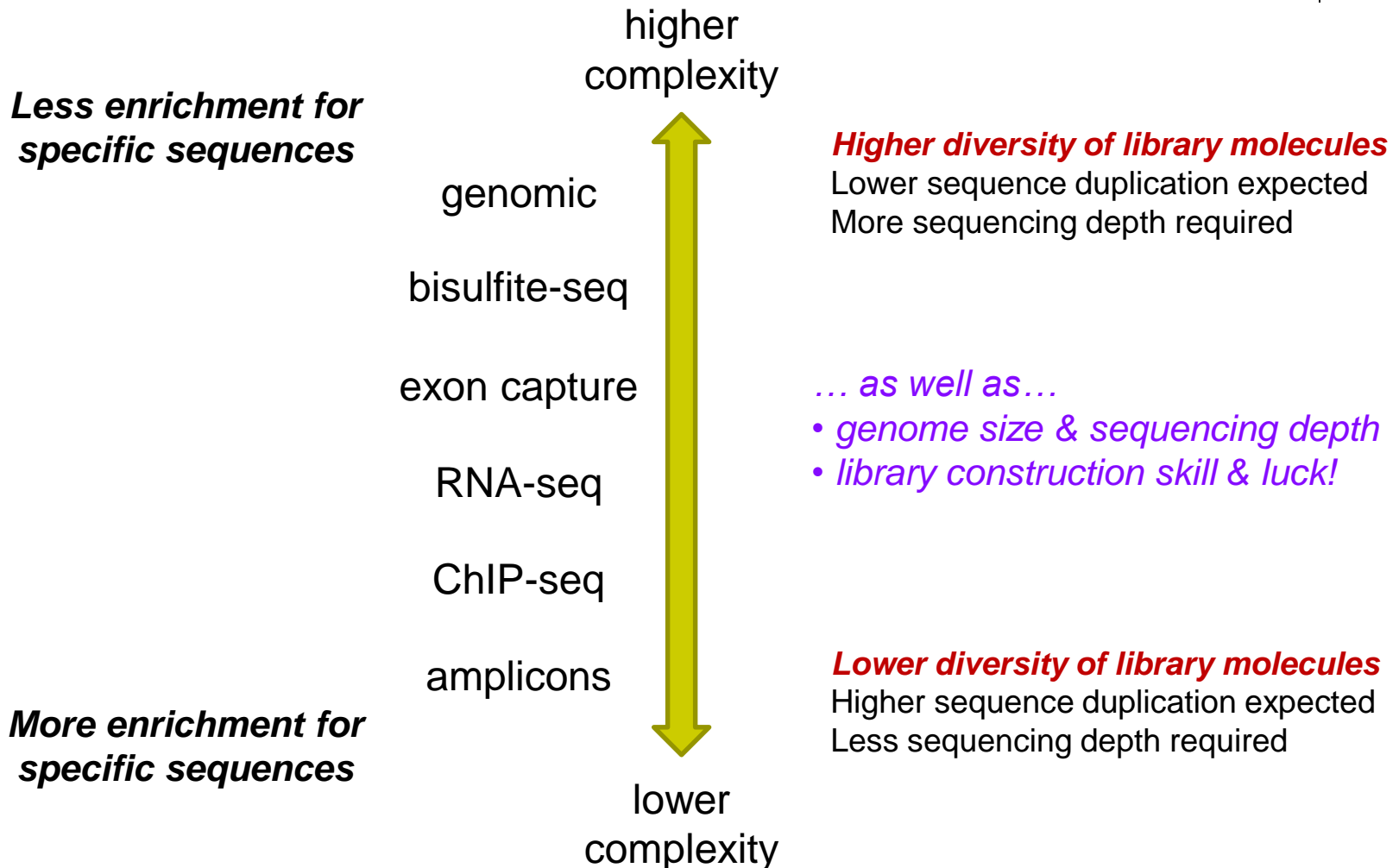
The number of different molecules in a library depends on *enrichment* performed during library construction.

Popular Experiment Types



- **Whole Genome sequencing (WGS)**
 - **library:** all genomic DNA
 - **complexity:** high (fragments must cover the entire genome)
 - **applications:** genome assembly, variant analysis
- **Exome sequencing (WXS)**
 - **library:** DNA from eukaryotic exonic regions (uses special kits)
 - **complexity:** high/med (only ~5% of eukaryotic genome is in exons)
 - **applications:** polymorphism/SNP detection; genotyping
- **RNA-seq**
 - **library:** extracted RNA converted to cDNA
 - **complexity:** med/high (only a subset of genes are expressed in any given tissue)
 - **applications:** differential gene expression
- **Amplicon panels (targeted sequencing)**
 - **library:** DNA from a set of PCR-amplified regions using custom primers
 - **complexity:** very low (only 1 to a few thousand different library molecules)
 - **applications:** genetic screening panels; metagenomics; mutagenesis

Library complexity is primarily a function of experiment type



Read types



single-end



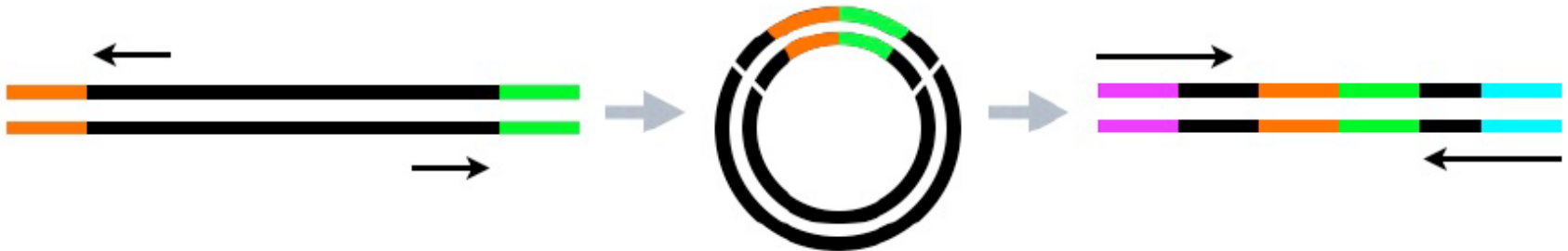
independent reads

paired-end



two inwardly oriented reads separated by ~200 nt

mate-paired

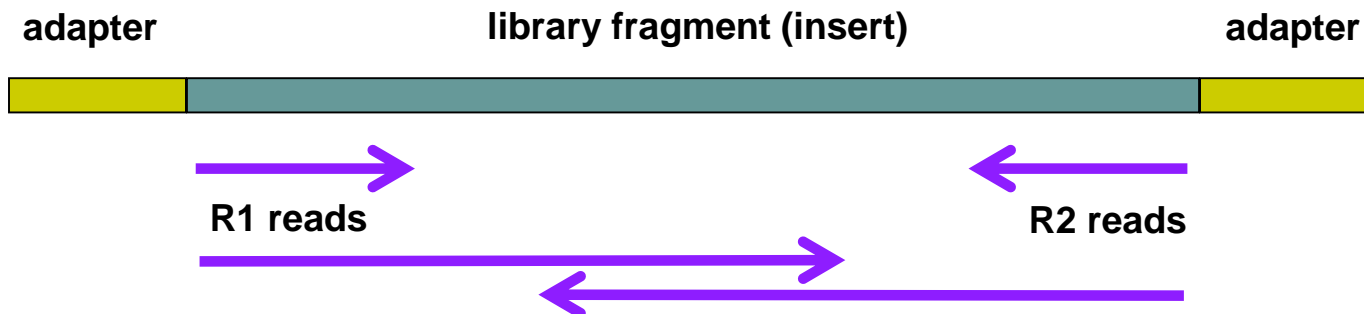


two outwardly oriented reads separated by ~3000 nt



Reads and Fragments

- With paired-end sequencing, keep in mind the distinction between
 - the library *fragment* from your library that was sequenced
 - also called *inserts*
 - the *sequence reads* (R1s & R2s) you receive
 - also called *tags*
 - an R1 and its associated R2 form a *read pair*
 - a readout of part (or all) of the fragment molecule



Single end vs Paired end



- **single end** (SE) reads are less expensive
- **paired end** (PE) reads can be mapped more reliably
 - especially against lower complexity genomic regions
 - an unmapped read can be “rescued” if its mate maps well
 - they provide more bases around a locus
 - e.g. for analysis of polymorphisms
 - actual fragment sizes can be easily determined
 - from the alignment records for each dual-mapping “proper pair”
 - also help distinguish the true complexity of a library
 - by clarifying which *fragments* are duplicates (vs *read* duplicates)
 - **but** PE reads are more expensive – and larger
 - more storage space and processing time required
- General guidelines
 - use PE for high location accuracy and/or base-level sensitivity
 - use SE for lower-complexity, higher duplication experiments

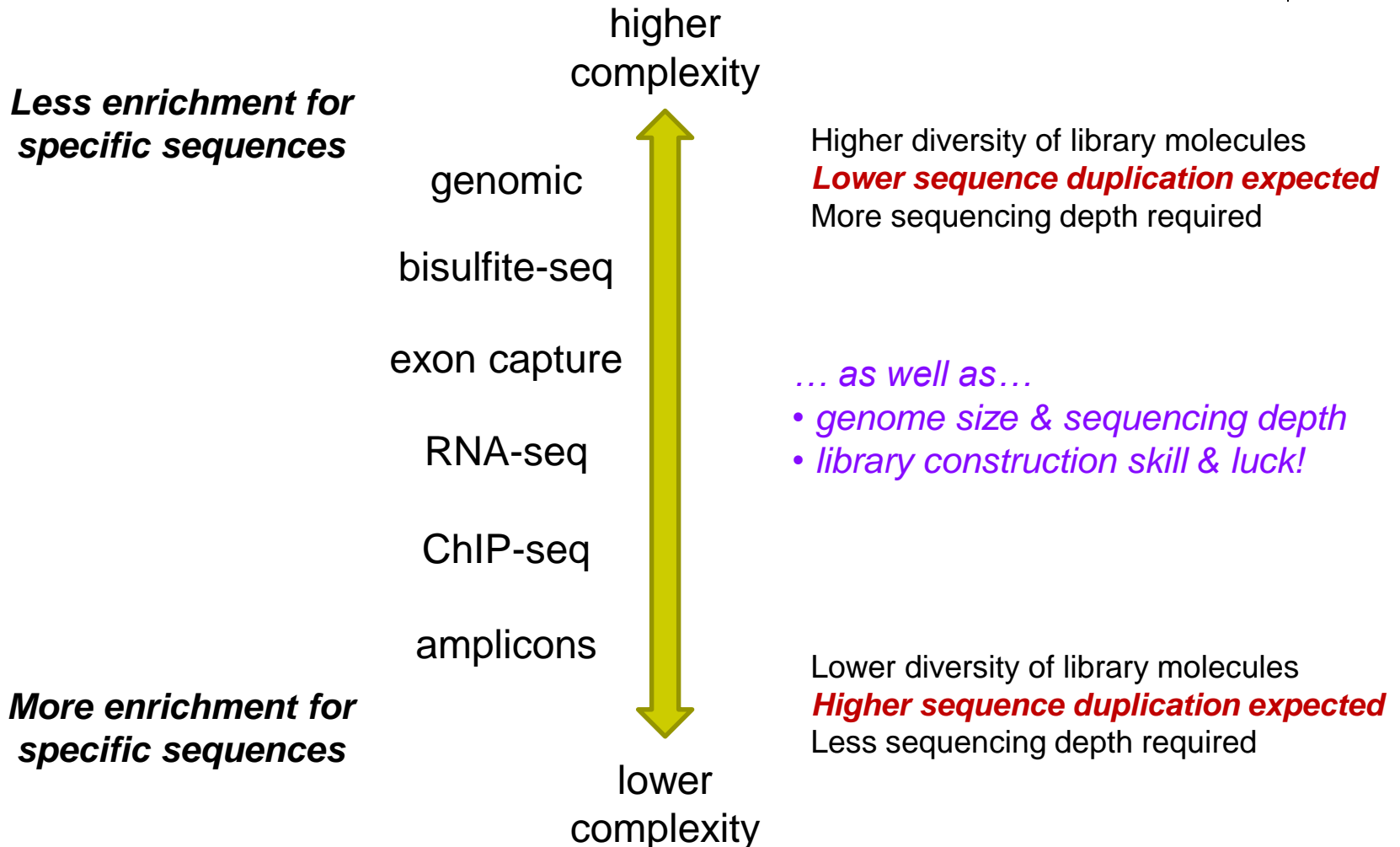


Sequence Duplication



- The set of sequences you receive can contain exact duplicates
- Duplication can arise from:
 1. sequencing of species enriched in your library (**biological – good!**)
 - each read comes from a different DNA molecule (cluster)
 2. sequencing of artifacts (**technical – bad!**)
 - differentially amplified PCR species (PCR duplicates)
 - recall that 2 PCR amplifications are performed w/Illumina sequencing
- ***cannot tell which using “standard” sequencing methods!***
- Standard best practice is to “mark duplicates” during initial processing
 - then decide what to do with them later...
 - e.g. retain (use all), remove (use only non-duplicates), dose (use some)
- Different experiment types have different *expected* duplication

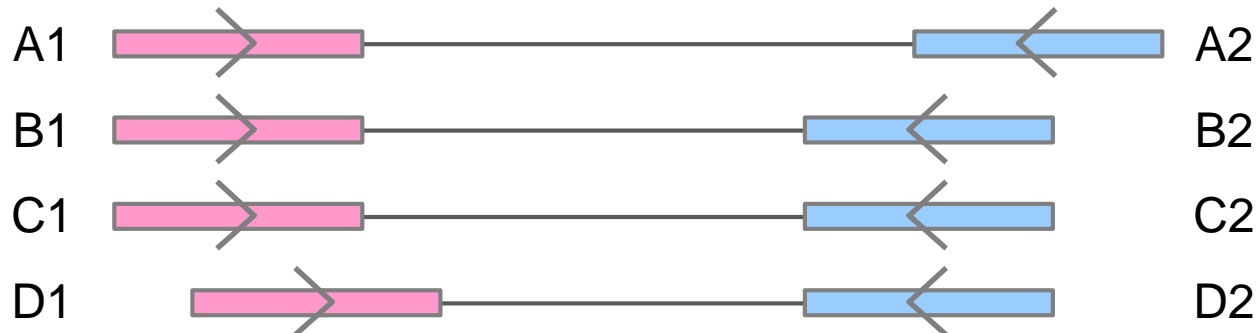
Expected sequence duplication is primarily a function of experiment type





Read vs Fragment duplication

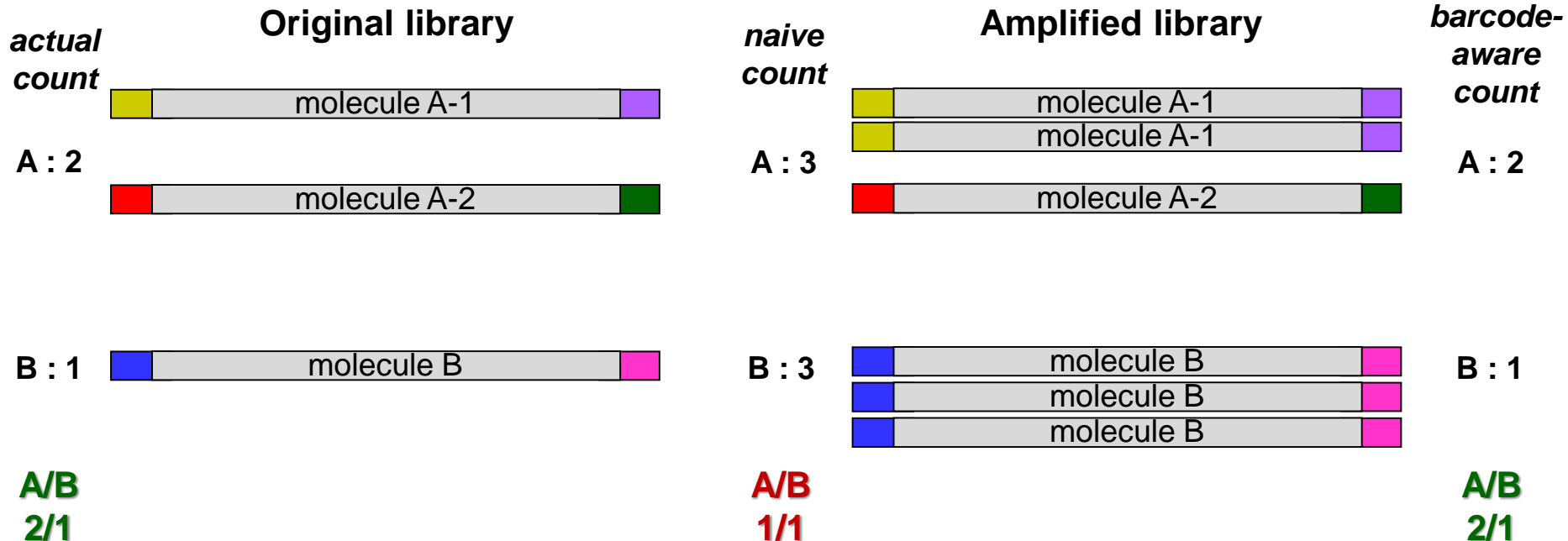
- Consider the 4 fragments below
 - 4 R1 reads (pink), 4 R2 reads (blue)
- Duplication when only 1 end considered
 - A1, B1, C1 have identical sequences, D1 different
 - 2 unique + 2 duplicates = 50% duplication rate
 - B2, C2, D2 have identical sequences, A2 different
 - 2 unique + 2 duplicates = 50% duplication rate
- Duplication when both ends considered
 - fragments B and C are duplicates (same external sequences)
 - 3 unique + 1 duplicate = 25% duplication rate



Molecular Barcoding



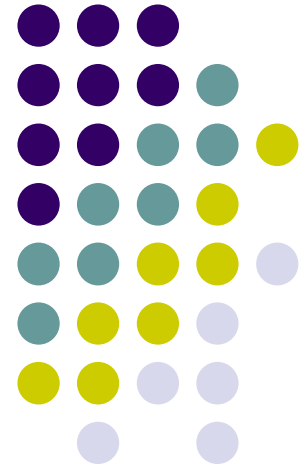
- Resolves ambiguity between biological and technical (PCR amplification) duplicates
 - adds secondary, **internal** barcodes to **pre-PCR** molecules
 - combination of barcodes + insert sequence can provide accurate quantification
 - but requires specialized pre- and post-processing



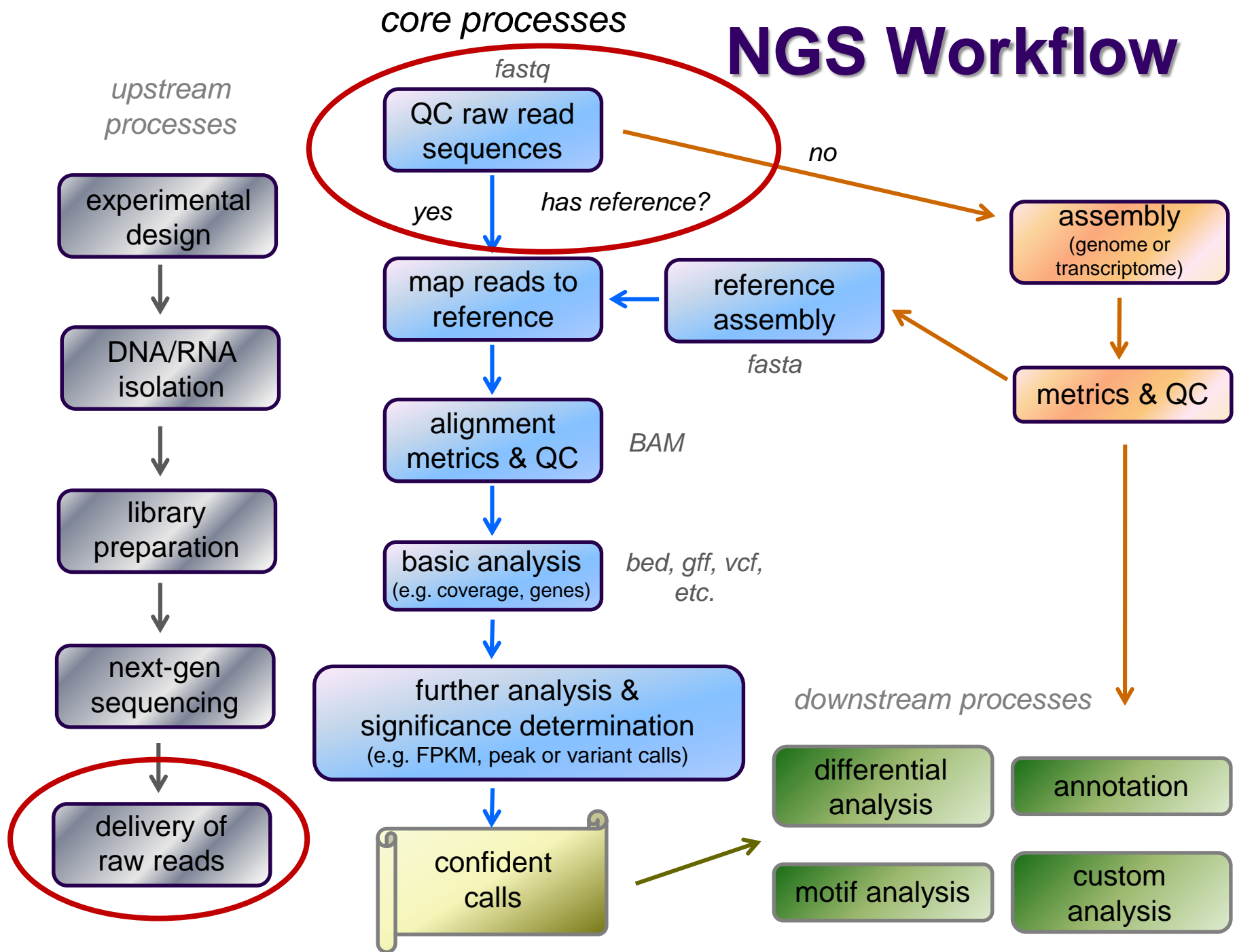
Part 3:

The FASTQ format, Data QC & preparation

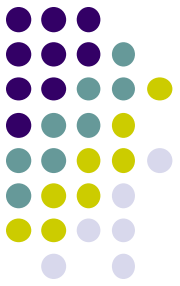
- FASTA and FASTQ formats
- QC of raw sequences with **FastQC** tool
- Dealing with adapters



NGS Workflow



FASTQ format



- Text format for storing sequence and quality data
 - http://en.wikipedia.org/wiki/FASTQ_format
- 4 lines per sequence:
 1. **@read name**
 2. **called base sequence (ACGTN)**
always 5' to 3'; *usually* excludes 5' adapter
 3. **+optional read name**
 4. **base quality scores encoded as text characters**
- FASTQ representation of a single, 50 base R1 sequence

```
@HWI-ST1097:97:D0WW0ACXX:8:1101:2007:2085 1:N:0:ACTTGA  
ATTCTCCAAGATTTGGCAAATGATGAGTACAATTATATGCCCAATTTACA  
+  
?@@?DD;?;FF?HHBB+:ABECGHDHDCF4?FGIGACFDFH;FHEIIIB9?
```



FASTQ read names

- Illumina Fastq read names encode information about the source cluster
 - unique identifier (“fragment name”) begins with @, then:
 - sequencing machine name
 - lane number
 - flowcell coordinates
 - a space separates the name from extra read information:
 - end number (1 for R1, 2 for R2)
 - two quality fields (N = **not** QC failed)
 - library barcode sequence
- R1, R2 reads ***have the same fragment name***
 - this is how the reads are linked to model the original fragment molecule

@HWI-ST1097:97:D0WW0ACXX:8:1101:2007:2085 1:N:0:ACTTGA

@HWI-ST1097:97:D0WW0ACXX:8:1101:2007:2085 2:N:0:ACTTGA



FASTQ quality scores

- Base qualities expressed as **Phred** scores
 - log scaled, **higher = better**
 - $20 = 1/10^2 = 1/100$ errors, $30 = 1/10^3 = 1/1000$ errors

$$\text{Probability of Error} = 10^{-Q/10}$$

- Integer Phred score converted to Ascii character (add 33)

<http://www.asciitable.com/>

Quality character	!"#\$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJ
ASCII Value	33 43 53 63 73
Base Quality (Q)	0 10 20 30 40

?@@?DD ; ? ; FF?HHBB+ : ABECGHDHDCF4?FGIGACFDFH ; FHEIIB9?

Raw sequence quality control



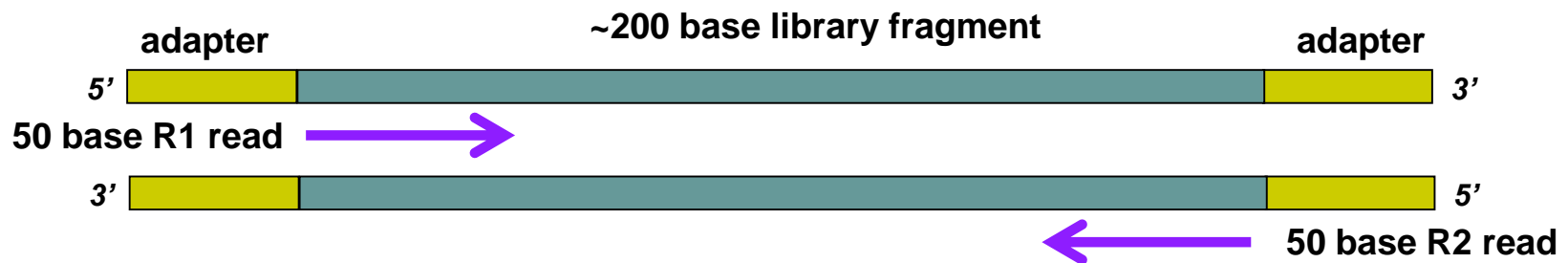
- Critical step! Garbage in = Garbage out
 - general sequence quality
 - base quality distributions
 - sequence duplication rate
 - trim 3' adapter sequences?
 - important for RNAseq
 - trim 3' bases with poor quality?
 - important for *de novo* assembly
 - other contaminants?
 - biological – rRNA in RNAseq
 - technical – samples sequenced w/other barcodes



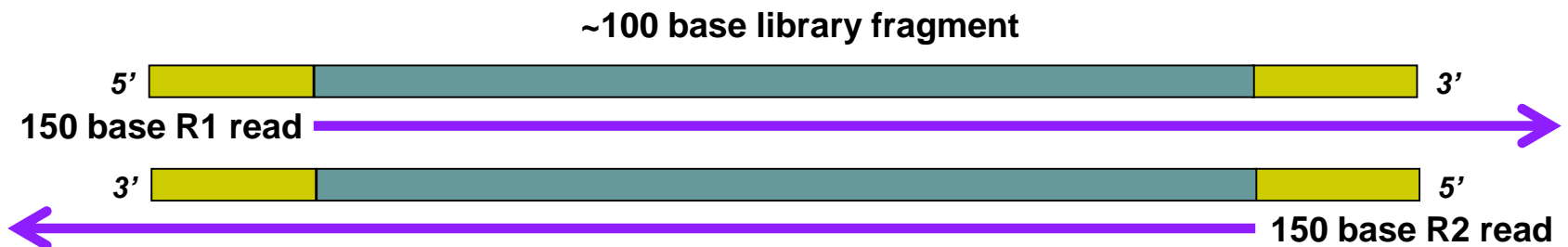


3' Adapter contamination

A. reads short compared to fragment size (no contamination)



B. Reads long compared to library fragment (3' adapter contamination)



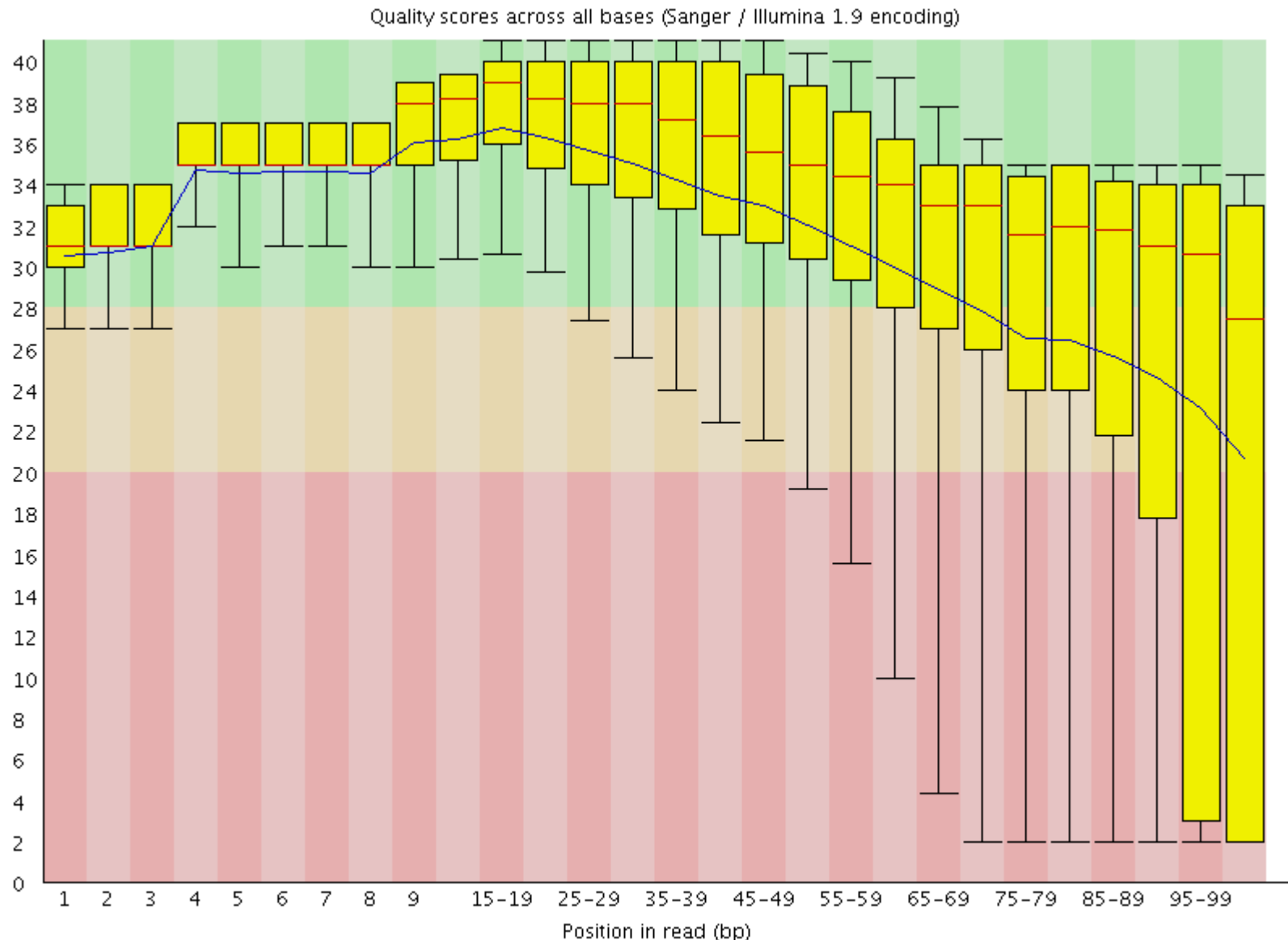
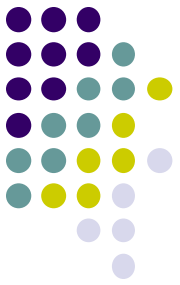
The presence of the 3' adapter sequence in the read can cause problems during alignment, because it does not match the genome.

FastQC



- Quality Assurance tool for FASTQ sequences
<http://www.bioinformatics.babraham.ac.uk>
- Most useful reports:
 1. *Per-base sequence quality Report*
 - Should I trim low quality bases?
 2. *Sequence duplication levels Report*
 - How complex is my sequence library?
 3. *Overrepresented sequences Report*
 - Do I need to remove adapter sequences?

1. FastQC Per-base sequence quality report



2. FastQC Sequence duplication report Yeast ChIP-seq

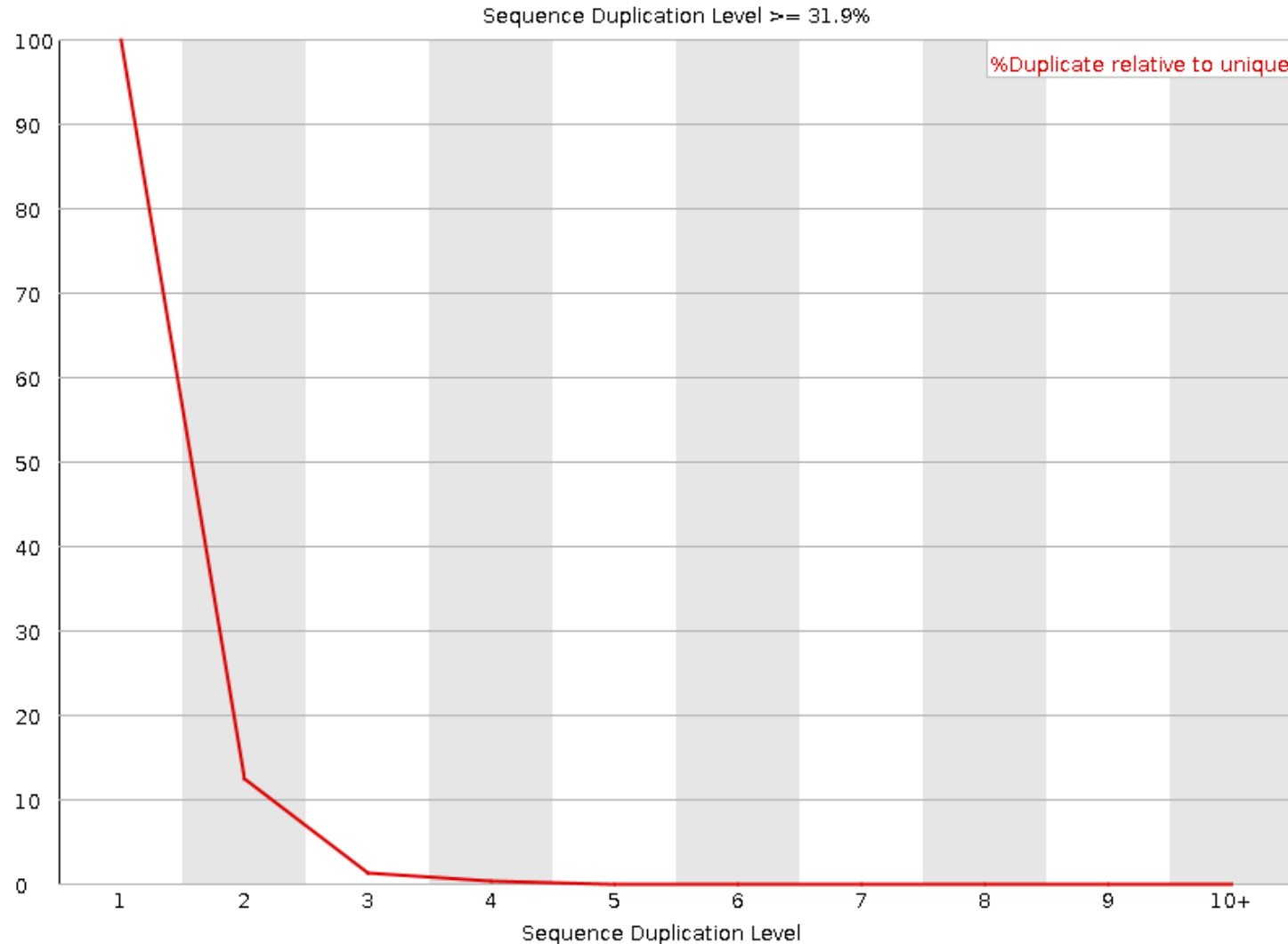


For every 100 unique sequences there are:

~12 sequences w/2 copies

~1-2 with 3 copies

Ok – Some duplication expected due to IP enrichment



2. Sequence duplication report

Yeast ChIP-exo

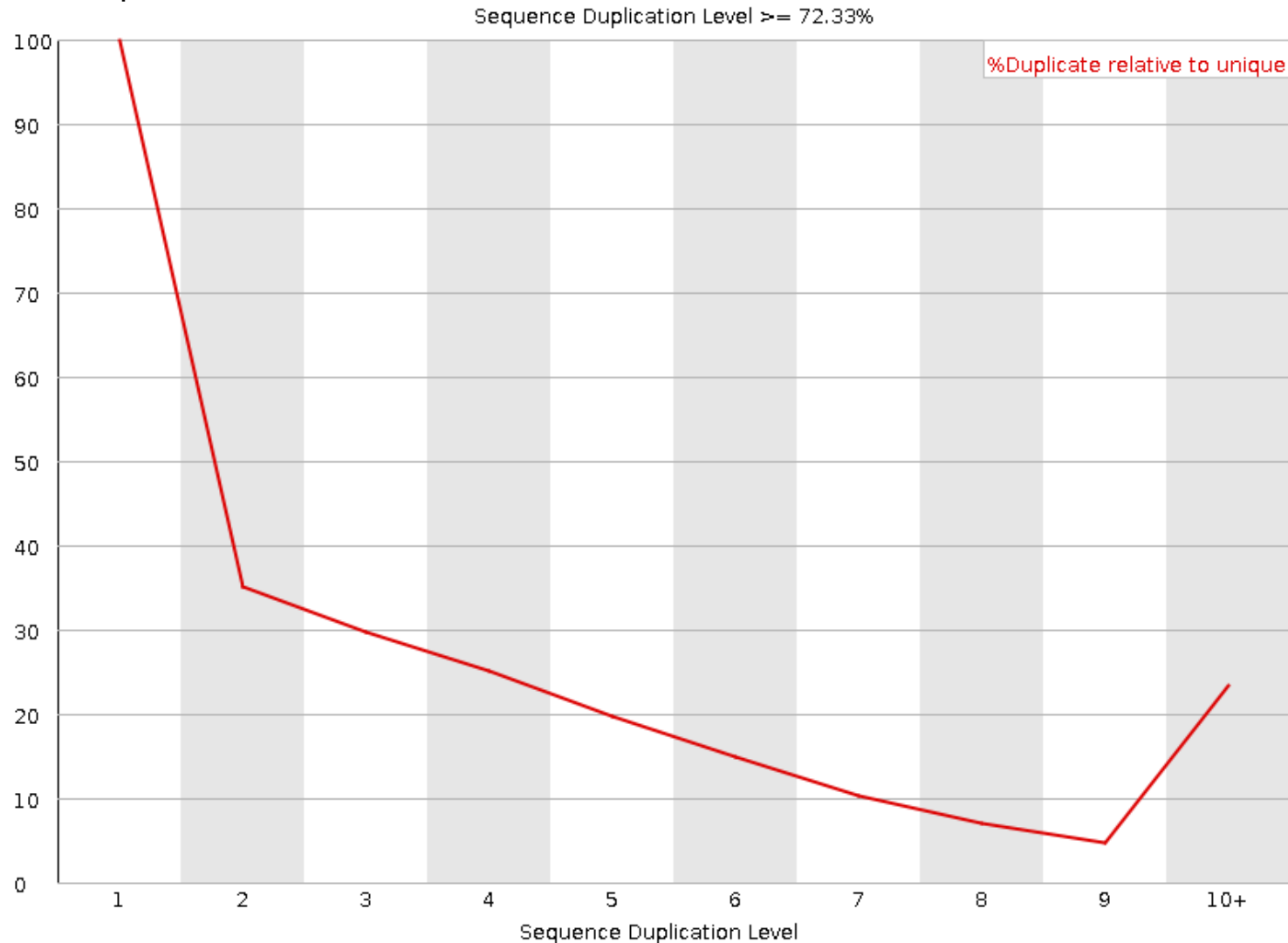


For every 100 unique sequences there are:

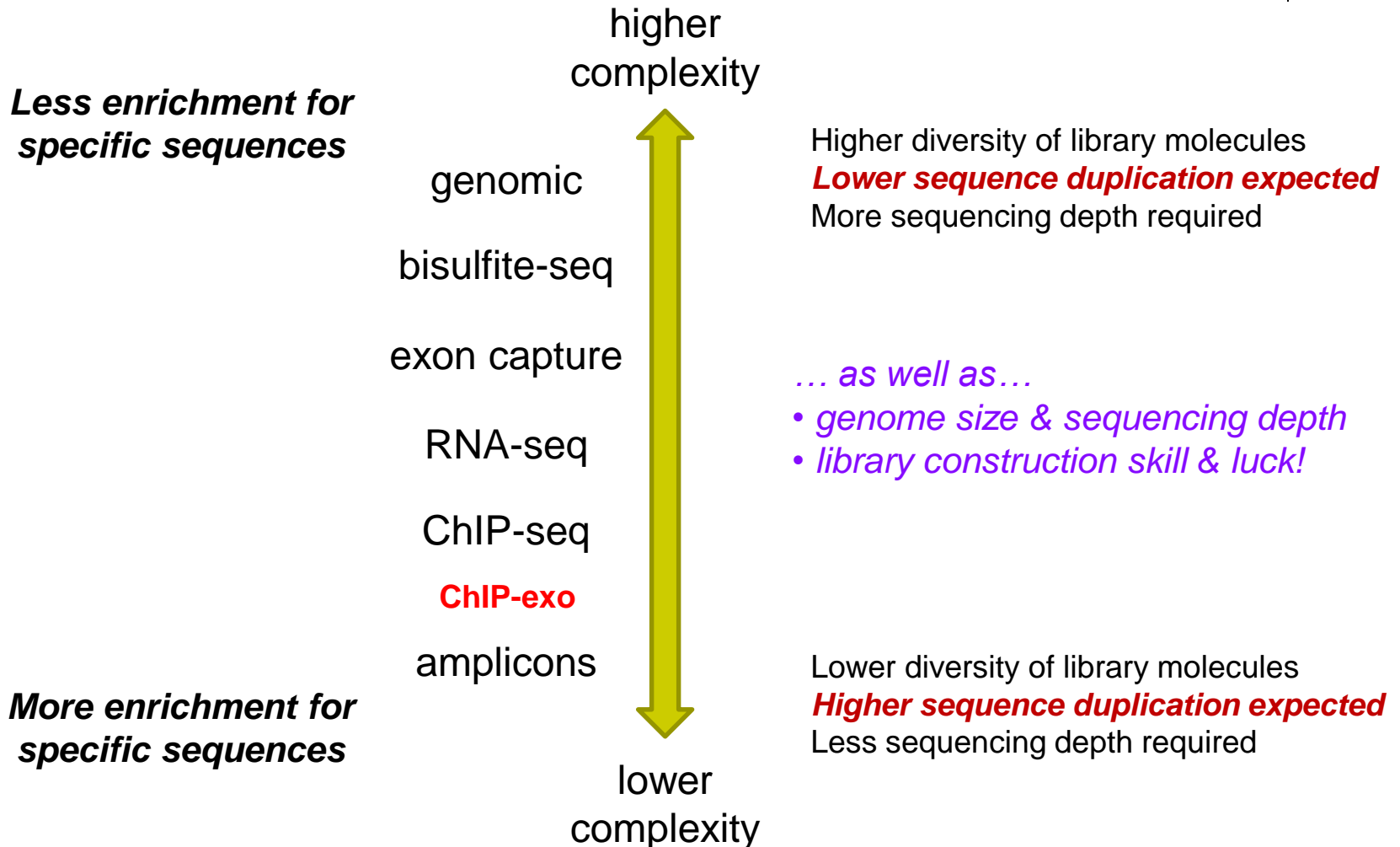
~35 sequences w/2 copies

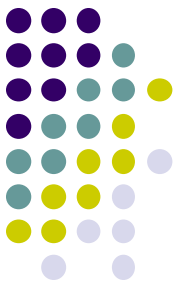
~22 with 10+ copies

Success! Protocol expected to have high duplication



Expected sequence duplication is primarily a function of experiment type

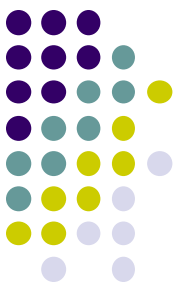




3. FastQC Overrepresented sequences report

- **FastQC** knows Illumina adapter sequences
- Here ~9-10% of sequences contain adapters
 - calls for adapter removal or trimming

Sequence	Count	Percentage	Possible Source
AGATCGGAAGAGCACACGTCTGAACTCCAGTCACCTCAGAATCTCGTATG	60030	5.01369306977828	TruSeq Adapter, Index 1 (97% over 37bp)
GATCGGAAGAGCACACGTCTGAACTCCAGTCACCTCAGAATCTCGTATGC	42955	3.5875926338884896	TruSeq Adapter, Index 1 (97% over 37bp)
CACACGTCTGAACTCCAGTCACCTCAGAATCTCGTATGCCGTCTTCTGCT	3574	0.29849973398946483	RNA PCR Primer, Index 40 (100% over 41bp)
CAGATCGGAAGAGCACACGTCTGAACTCCAGTCACCTCAGAATCTCGTAT	2519	0.2103863542024236	TruSeq Adapter, Index 1 (97% over 37bp)
GAGATCGGAAGAGCACACGTCTGAACTCCAGTCACCTCAGAATCTCGTAT	1251	0.10448325887543942	TruSeq Adapter, Index 1 (97% over 37bp)



3. Overrepresented sequences

- Here nearly 1/3 of sequences some type of non-adapter contamination
 - BLAST** the sequence to identify it

Sequence	Count	Percentage	Possible Source
GAAGGTCACGGCGAGACGAGCCGTTTATCATTACGATAGGTGTCAAGTGG	5632816	32.03026785752871	No Hit
TATTCTGGTGTCTAGGCGTAGAGGAACAACACCAATCCATCCCGAACTT	494014	2.8091456822607364	No Hit
TCAAACGAGGAAAGGCTTACGGTGGATACCTAGGCACCCAGAGACGAGGA	446641	2.539765344040083	No Hit
TAAACGACTCTCGGCAACGGATATCTCGGCTCTCGCATCGATGAAGAAC	179252	1.0192929387357474	No Hit
GAAGGTCACGGCGAGACGAGCCGTTTATCATTACGATAGGGGTCAAGTGG	171681	0.9762414422996221	No Hit
AACGACTCTCGGCAACGGATATCTCGGCTCTCGCATCGATGAAGAACGTA	143415	0.8155105483274229	No Hit
AGAACATGAAACCGTAAGCTCCCAAGCAGTGGGAGGAGCCCTGGGCTCTG	111584	0.6345077504066322	No Hit
AAAACGACTCTCGGCAACGGATATCTCGGCTCTCGCATCGATGAAGAACG	111255	0.6326369351474214	No Hit
ATTACGATAGGTGTCAAGTGGAAGTGCAGTGATGTATGCAGCTGAGGCAT	73682	0.41898300890326096	No Hit
GAAGGTCACGGCGAGACGAGCCGTTTATCATTACGATAGGTGTCAAGGGG	71661	0.4074908580252516	No Hit
GGATGCGATCATACCAGCACTAATGCACCGGATCCCATCAGAACTCCGCA	69548	0.3954755612388914	No Hit
ATATTCTGGTGTCTAGGCGTAGAGGAACAACACCAATCCATCCCGAACT	54017	0.30716057099328803	No Hit

Dealing with 3' adapters



- Three main options:

1. **Hard trim** all sequences by specific amount

- e.g. trim 100 base reads to 50 bases
- *Pro*: fast & easy to perform; trims low-quality 3' bases
- *Con*: removes information (bases) you might want

2. **Remove adapters** specifically

- e.g. using specific tools
- *Pro*: removes adapter contamination without losing sequenced bases
- *Con*: requires knowledge of insert fragment structure & adapters

3. Perform a **local alignment** (vs **global**)

- e.g. **bowtie2 --local** or **bwa mem**
- *Pro*: mitigates adapter contamination while retaining full query sequence
- *Con*: limited aligner support (but always needed for RNA-seq alignment)



FASTQ trimming

- Tools:
 - **cutadapt** – <https://code.google.com/p/cutadapt/>
 - **trimmomatic** – <http://www.usadellab.org/cms/?page=trimmomatic>
 - FASTX-Toolkit – http://hannonlab.cshl.edu/fastx_toolkit/
- Features:
 - hard-trim specific number of bases
 - trimming of low quality bases
 - specific trimming of adapters
 - support for trimming paired end read sets (except FASTX)
 - **cutadapt** has protocol for separating reads based on internal barcode

Local vs. Global alignment



- **Global** alignment
 - requires query sequence to map **fully** (end-to-end) to reference
- **Local** alignment
 - allows a **subset** of the query sequence to map to reference
 - “untemplated” adapter sequences will be “soft clipped” (ignored)

global (*end-to-end*)
alignment of query

local (*subsequence*)
alignment of query

CACAAGTACAATTATACAC

CTAG**CTTATCGCCCTGAA**GGACT

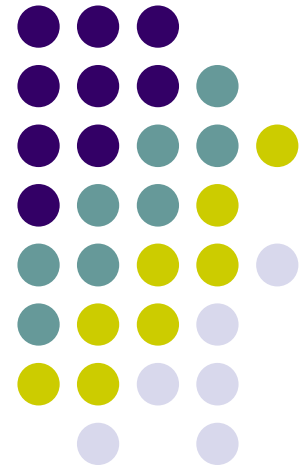
TACATA**CACAAGTACAATTATACAC**AGACATTAGTT**CTTATCGCCCTGAA**AATTCTCC

reference sequence

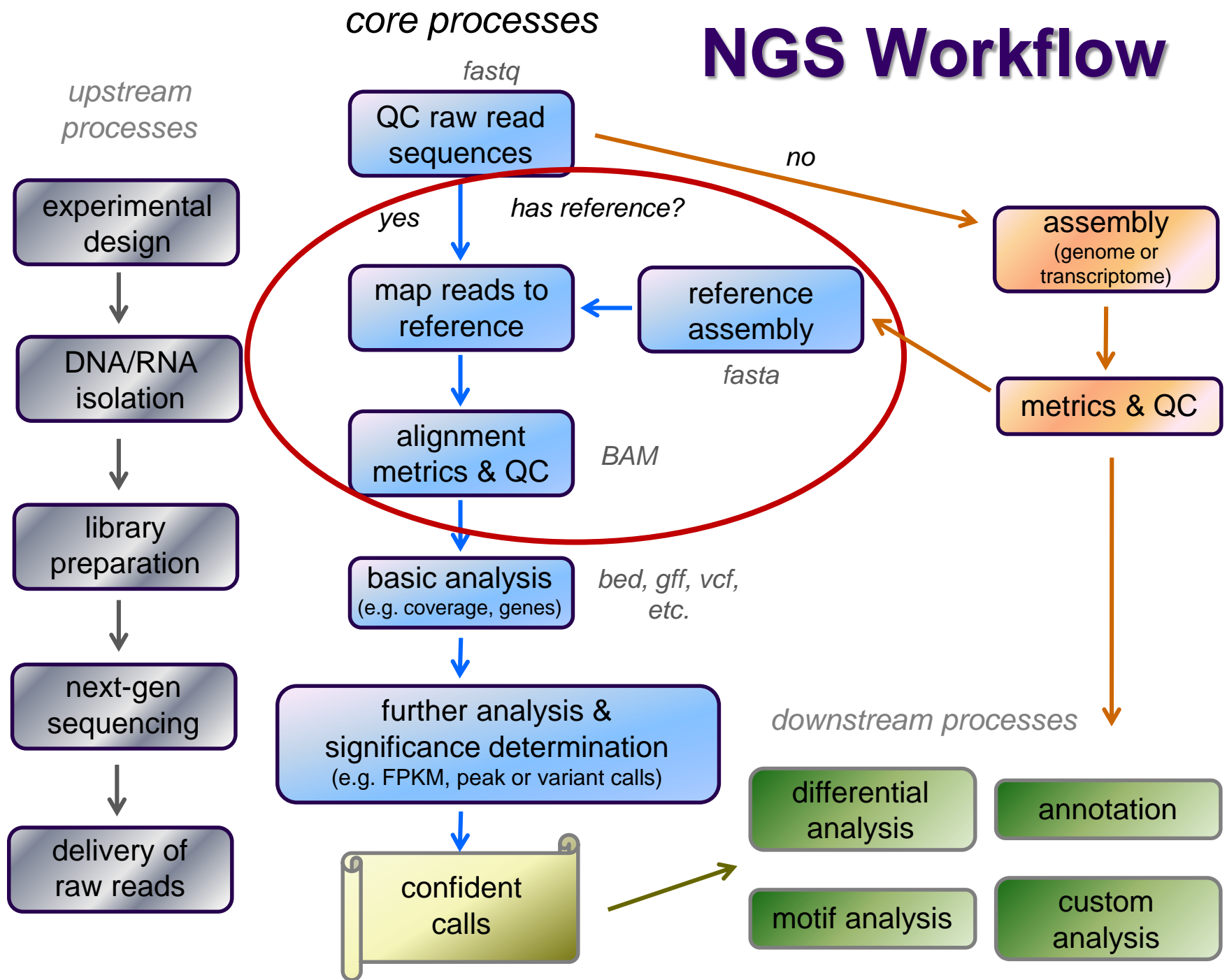
Part 4:

Alignment to a reference assembly

- Alignment overview & concepts
- Preparing a reference genome
- Alignment workflow steps



NGS Workflow





Short Read Aligners

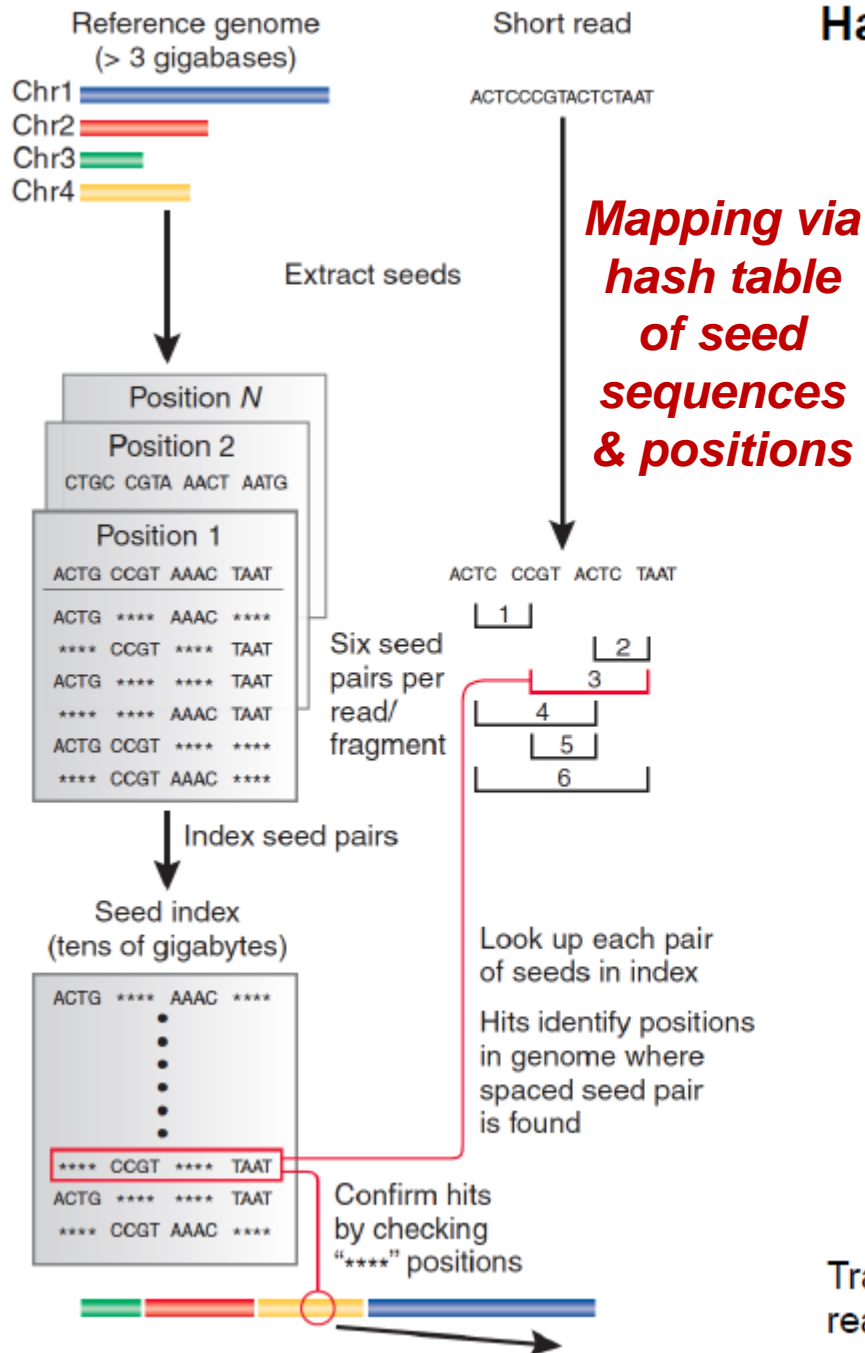
- Short read mappers determine placement of *query sequences* (your reads) against a known *reference*
 - **BLAST**:
 - one query sequence (or a few)
 - many matches for each
 - short read aligners
 - many millions of query sequences
 - want only one “best” mapping (or a few)
- Many aligners available! Two of the most popular
 - **bwa** (Burrows Wheeler Aligner) by Heng Li
<http://bio-bwa.sourceforge.net/>
 - **bowtie2** – part of the Johns Hopkins Tuxedo suite of tools
<http://bowtie-bio.sourceforge.net/bowtie2/manual.shtml>
 - Given similar input parameters, they produce similar alignments
 - and both run relatively quickly

Mapping vs Alignment

- **Mapping** determines one or more **positions** (a.k.a. **seeds** or **hits**) where a read shares a short sequence with the reference
- **Alignment** starts with the seed and determines how read bases are best **matched**, base-by-base, around the seed
- Mapping quality and alignment scores are both reported
 - High mapping quality \neq High alignment score
 - **mapping quality** describes **positioning**
 - reflects the probability that the read is *incorrectly* mapped to the reported location
 - is a Phred score: $P(\text{incorrectly mapped}) = 10^{-\text{mappingQuality}/10}$
 - reflects the **complexity** or information content of the sequence (**mappability**)
 - **alignment score** describes **fit**
 - reflects the correspondence between the read and the reference sequence

	Read 1	Read 2	or	
• Maps to one location <i>high mapping quality</i>	GCGTAGTCTGCC	ATCGGGAGATCC		• Maps to 2 locations <i>low mapping quality</i>
• Has 2 mismatches <i>low alignment score</i>				• Matches perfectly <i>high alignment score</i>
	TAGCCTAGTGTGCCGC	TAATCGGGAGATCCGC		TTATCGGGAGATCCGC
		reference sequence		

a Spaced seeds



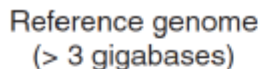
Hash table enables lookup of exact matches.

Subsequence	Reference Positions
ATAGCTAATCCAAA	2341, 2617264
ATAGCTAATCCAAT	
ATAGCTAATCCAAC	134, 13311, 732661,
ATAGCTATCCAAAG	
ATAGCTAATCCATA	
ATAGCTAATCCATT	3452
ATAGCTAATCCATC	
ATAGCTATCCAATG	234456673

Table is sorted and complete so you can jump immediately to matches.
(But this can take a lot of memory.)

May include N bases, skip positions, etc.

Trapnell, C. & Salzberg, S. L. How to map billions of short reads onto genomes. *Nature Biotech.* **27**, 455–457 (2009).

b

Short read

Chr1
Chr2
Chr3
Chr4

Concatenate into
single string

Burrows-Wheeler transform and indexing

Bowtie index
(~2 gigabytes)

Look up
'suffixes'
of read

Hits identify positions in genome where read is found

Convert each
hit back to
genome location

Burrows-Wheeler transform compresses sequence.

Input

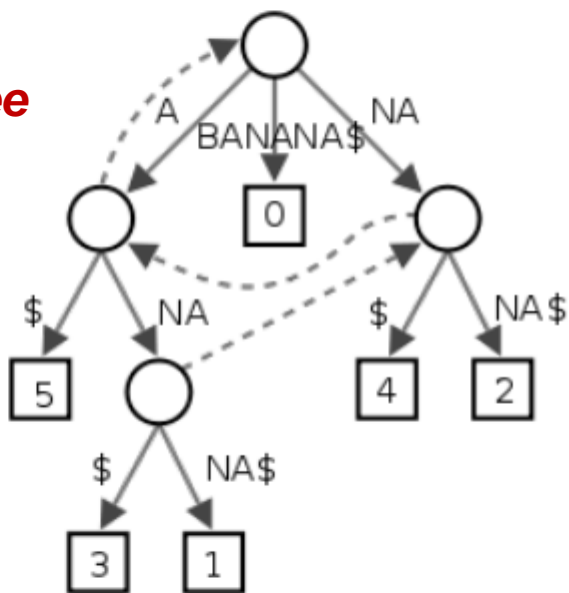
SIX.MIXED.PIXIES.SIFT.SIXTY.PIXIE.DUST.BOXES

Output

TEXYDST.E.IXIXIXSSMPPS.B..E.S.EUSFXDIIIOIIIT

Suffix tree enables fast lookup of subsequences.

Mapping via suffix array tree

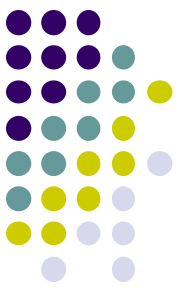


http://en.wikipedia.org/wiki/Suffix_tree

Exact matches at all positions below a node.

Trapnell, C. & Salzberg, S. L. How to map billions of short reads onto genomes. *Nature Biotech.* **27**, 455–457 (2009).

Alignment via dynamic programming



- Dynamic programming algorithm
(Smith-Waterman | Needleman-Wunsch)

	G	A	A	T	T	C	A	G	T	T	A
G	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1
A	0	1	1	2	2	2	2	2	2	2	3
T	0	1	2	2	3	3	3	3	3	3	3
C	0	1	2	2	3	3	4	4	4	4	4
G	0	1	2	2	3	3	4	4	5	5	5
A	0	1	2	3	3	3	4	5	5	5	6

	G	A	A	T	T	C	A	G	T	T	A
G	0										
G		1									
A		1	1								
T				2	2						
C					3						
G						4	4				
A							5	5	5		
										6	

G _ A A T T C A G T T A
 | | | | | | | | | |
 G G _ A _ T C _ G _ _ A

- Alignment score = Σ**

- match reward
- base mismatch penalty
- gap open penalty
- gap extension penalty
- rewards and penalties may be adjusted for quality scores of bases involved

Reference sequence

ATTTGCGATCGGATGAAGACGAA

|||||

ATTTGCGATCGGATGTTGACTTT

ATTTGCGATCGGATGAAGACG..AA

|||||XX||Xi||

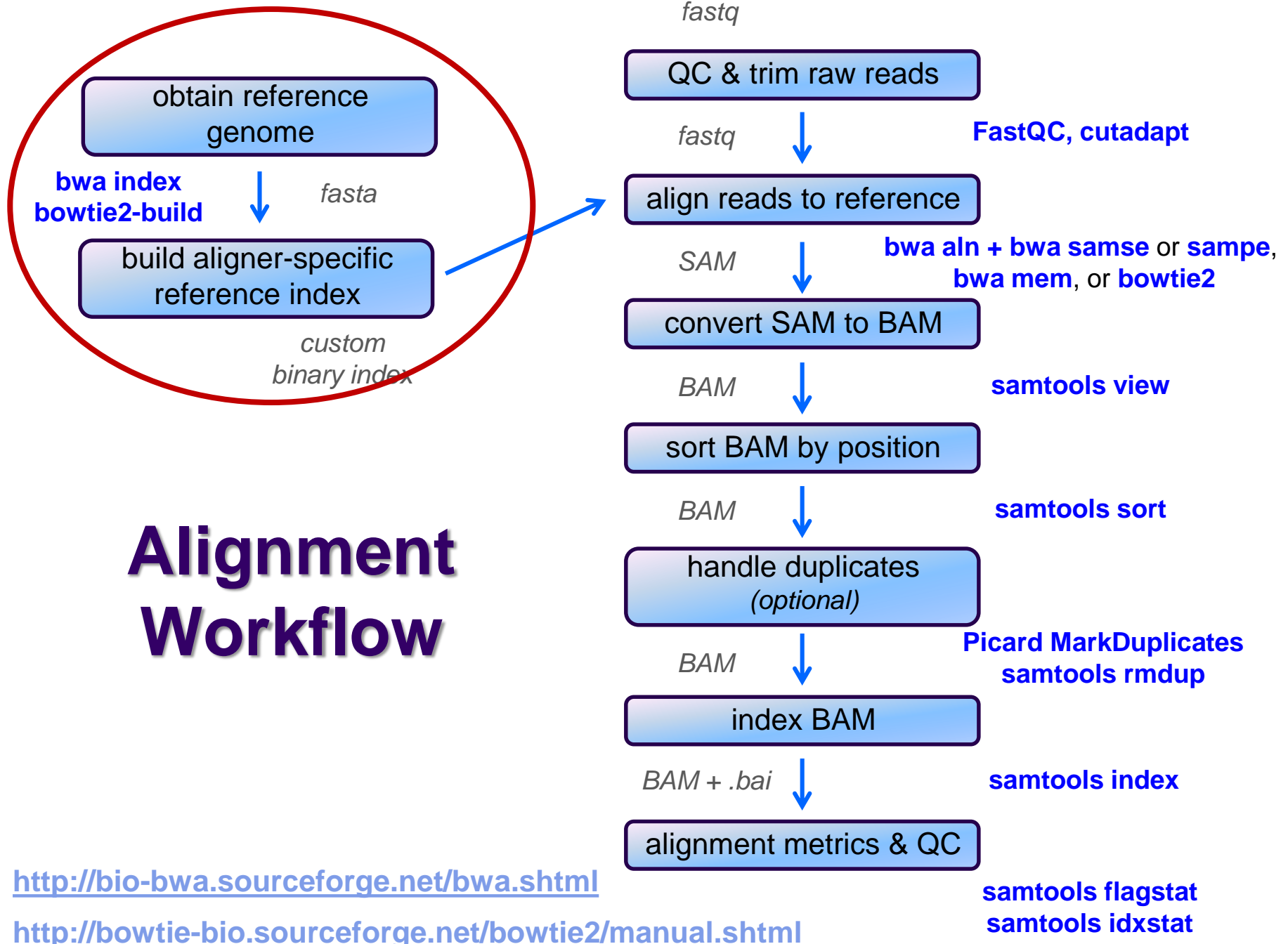
ATTTGCGATCGGATGTTGACTTTAA



Paired End mapping

- Having paired-end reads improves mapping
 - mapping one read with high confidence anchors the pair
 - even when its mate read by itself maps several places equally
- Three possible outcomes of mapping an R1/R2 pair
 1. only one of a pair might map (*singleton/orphan*)
 2. both reads can map within the most likely distance range and with correct orientation (*proper pair*)
 3. both reads can map but with an unexpected insert size or orientation, or to different contigs (*discordant pair*)
- Insert size is reported in the alignment record
 - for both proper and discordant pairs

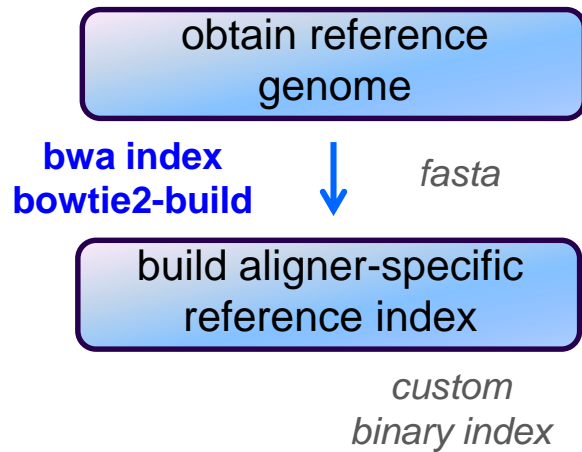
Alignment Workflow



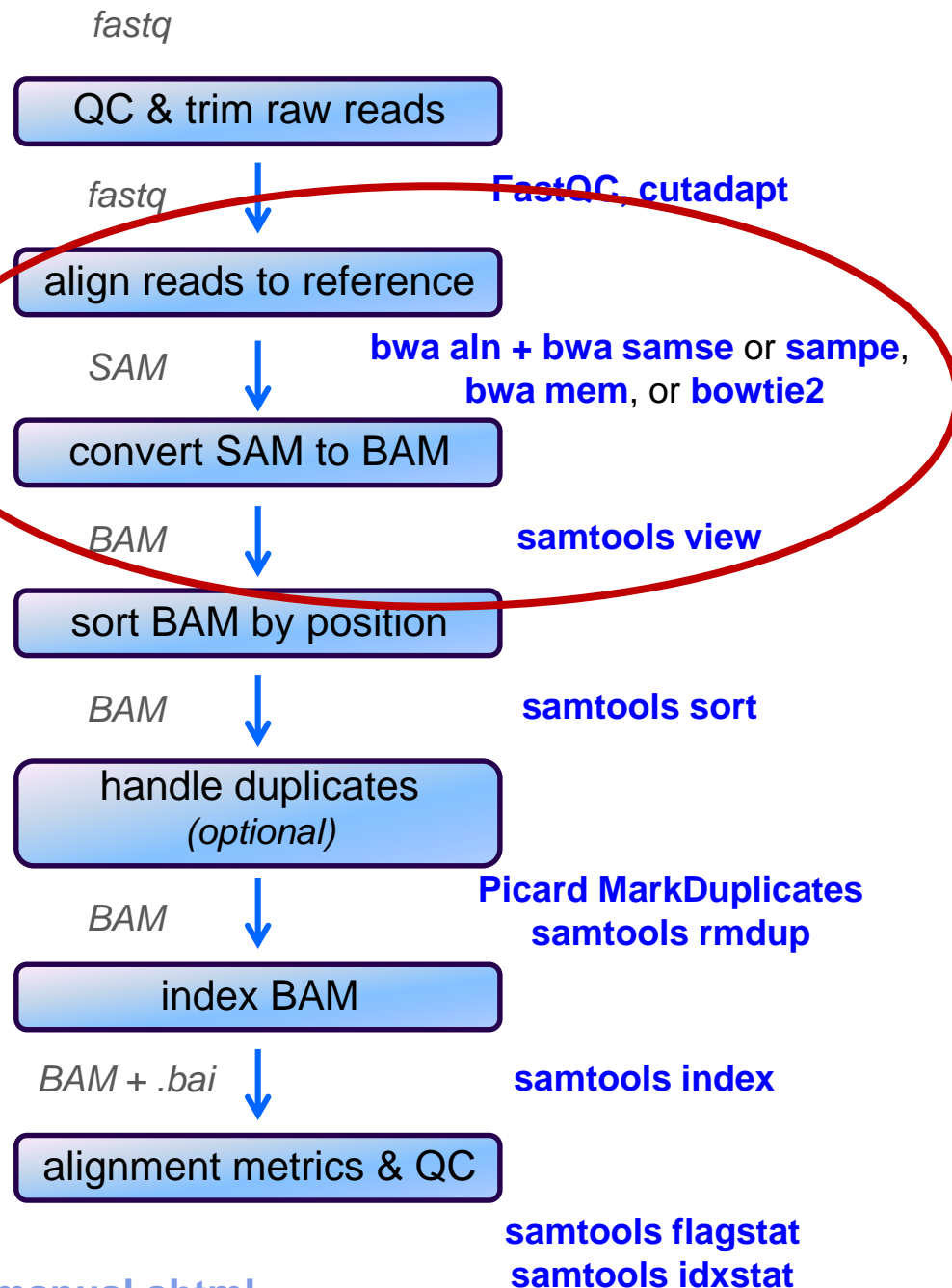


Obtaining/building a reference

- What is a reference?
 - *any set of named DNA sequences*
 - e.g. names are chromosome names
 - technically referred to as **contigs**
 - assembled genomes
 - Ensembl, UCSC, for eukaryotes
 - FASTA files (**.fa**, **.fasta**), + annotations (genome feature files, **.gff**)
 - NCBI RefSeq or GenBank for prokaryotes/microbes
 - any set of sequences of interest, e.g:
 - transcriptome (set of transcribed gene sequences)
 - rRNA genes (e.g. for filtering)
- Building a reference index
 - may take several hours to build
 - but you build each index once, use for multiple alignments



Alignment Workflow



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

<http://bowtie-bio.sourceforge.net/bowtie2/manual.shtml>

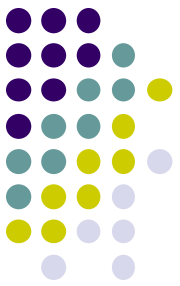


SAM / BAM file format

- Aligners take FASTQ as input, output alignments in **S**equence **A**lignment **M**ap (SAM) format
 - plain-text file format that describes how reads align to a reference
 - <http://samtools.github.io/hts-specs/SAMv1.pdf> (the Bible)
- SAM and BAM are two forms of the same data
 - **BAM** – **B**inary **A**lignment **M**ap
 - **same data** in a custom compressed (**gzip**'d) format
 - **much** smaller than SAM files
 - when indexed, support fast random access (SAM files do not)
- SAM file consists of
 - a **header** (includes reference sequence names and lengths)
 - **alignment records**, one for each sequence read
 - alignments for R1 and R2 reads have *separate records*
 - records have 11 fixed fields + extensible-format **key:type:value** tuples

SAM file format

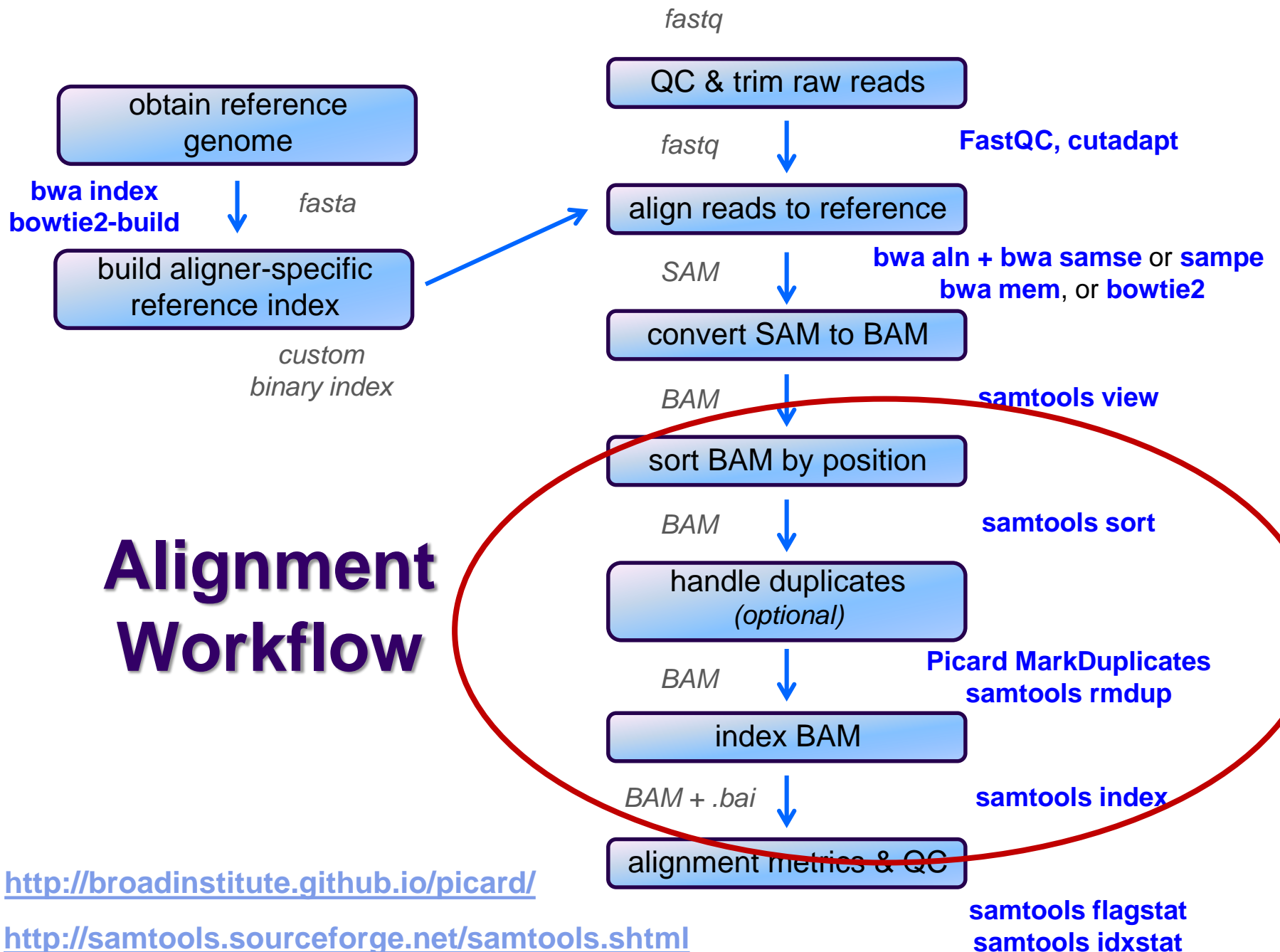
Fixed fields (tab-separated)



Col	Field	Type	Regexp/Range	Brief description
1	QNAME	String	[!-?A-~]{1,255}	Query template NAME <i>read name from fastq</i>
2	FLAG	Int	[0,2 ¹⁶ -1]	<u>bitwise FLAGs</u>
3	RNAME	String	* [!-()+-<>-~] [!-~]*	Reference sequence NAME <i>contig + start</i>
4	POS	Int	[0,2 ²⁹ -1]	<u>1-based leftmost mapping POSition</u> <i>= locus</i>
5	MAPQ	Int	[0,2 ⁸ -1]	MAPping Quality
6	CIGAR	String	* ([0-9]+[MIDNSHPX=])+	<u>CIGAR string</u> <i>use this to find end coordinate</i>
7	RNEXT	String	* = [!-()+-<>-~] [!-~]*	Ref. name of the mate/next segment
8	PNEXT	Int	[0,2 ²⁹ -1]	Position of the mate/next segment
9	TLEN	Int	[-2 ²⁹ +1,2 ²⁹ -1]	observed Template LENgth <i>insert size, if paired</i>
10	SEQ	String	* [A-Za-z=.]+	segment SEQuence
11	QUAL	String	[!-~]+	ASCII of Phred-scaled base QUALity+33

SRR030257.264529 99 NC_012967 1521 29 34M2S = 1564 79 *positive for plus strand reads*
 CTGGCCATTATCTCGGTGGTAGGACATGGCATGCCC
 AAAAAA;AA;AAAAAA??A%.;?&'3735',()0*,
 XT:A:M NM:i:3 SM:i:29 AM:i:29 XM:i:3 XO:i:0 XG:i:0 MD:Z:23T0G4T4

SRR030257.2669090 147 NC_012967 1521 60 36M = 1458 -99 *negative for minus strand reads*
 CTGGCCATTATCTCGGTGGTAGGTGATGGTATGCGC
 <<9:<<AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
 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:36





Sorting / indexing BAM files

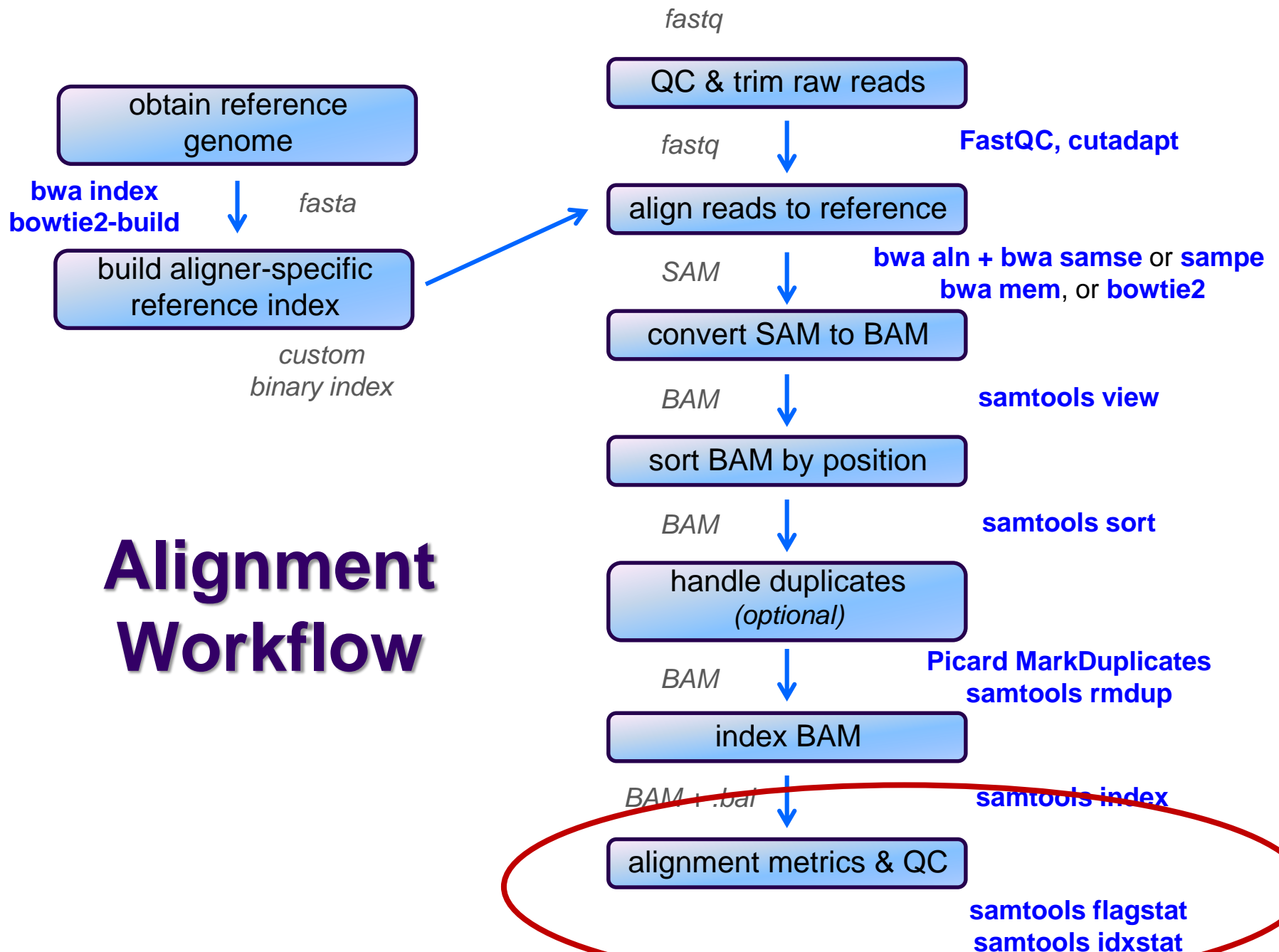
- SAM created by aligner contains read records in *name order*
 - same order as read names in the input FASTQ file
 - R1, R2 have adjacent SAM records
 - SAM → BAM conversion does not change the name-sorted order
- Sorting BAM puts records in *position (locus) order*
 - by contig name then start position (leftmost)
 - ***sorting is very compute, I/O and memory intensive!***
 - can take hours for large BAM
- Indexing a locus-sorted BAM allows fast random access
 - creates a small, binary alignment index file (.bai)
 - quite fast



Handling Duplicates

- Optional step, but very important for many protocols
- Definition of *alignment duplicates*:
 - single-end reads or singleton/discordant PE alignment reads
 - alignments have the same **start** positions
 - properly paired reads
 - pairs have same **external** coordinates (5' + 3' coordinates of the **insert**)
- Two choices for handling:
 - **samtools rmdup** – **removes** duplicates entirely
 - faster, but data is lost
 - **Picard MarkDuplicates** – **flags** duplicates only (0x400 bam flag)
 - slower, but all alignments are retained
 - both tools are quirky in their own ways

Alignment Workflow



Alignment metrics



- **samtools flagstat**

- simple statistics based on alignment record flag values
 - total sequences (R1+R2), total mapped
 - number properly paired
 - number of duplicates (0 if duplicates were not marked)

```
161490318 + 0 in total (QC-passed reads + QC-failed reads)
0 + 0 secondary
0 + 0 supplementary
31602827 + 0 duplicates
158093331 + 0 mapped (97.90% : N/A)
161490318 + 0 paired in sequencing
80745159 + 0 read1
80745159 + 0 read2
153721151 + 0 properly paired (95.19% : N/A)
156184878 + 0 with itself and mate mapped
1908453 + 0 singletons (1.18% : N/A)
1061095 + 0 with mate mapped to a different chr
606632 + 0 with mate mapped to a different chr (mapQ>=5)
```



Alignment wrap up

- Many tools involved
 - choose one or two and learn their options well
- Many steps are involved in the full alignment workflow
 - important to go through manually a few times for learning
 - but gets tedious quickly!
 - best practice
 - automate series of complex steps by wrapping into a ***pipeline script***
 - e.g. **bash** or **python** script
 - the Bioinformatics team has a set of pipeline scripts available at TACC
 - in shard project directory **/work/projects/BiolTeam/common/script/**
 - **align_bowtie2_illumina.sh**, **align_bwa_illumina.sh**, **trim_adapters.sh**, etc.

Other NGS Resources at UT



- CCBB Summer School courses
 - 4 half-day sessions in May
 - Intro to NGS, RNAseq, several others
 - lots of hands-on, including w/TACC
- Genome Sequencing & Analysis Facility (GSAF)
 - Jessica Podnar, Director, gsaf@utgsaf.org
- Bioinformatics consultants
 - Dennis Wylie, Dhivya Arasappan, Benni Goetz, Anna
- Biomedical Research Support Facility (BRCF)
 - provides local compute and managed storage resources
 - <https://wikis.utexas.edu/display/RCTFUsers>
- BiolTeam wiki – <https://wikis.utexas.edu/display/bioiteam/>