

Introduction to NGS Analysis

Anna Battenhouse

abattenhouse@utexas.edu

March 8, 2022

Associate Research Scientist

Center for Biomedical Research Support (CBRS)

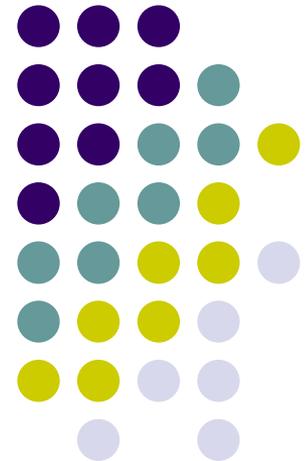
Bioinformatics Consulting Group (BCG)

Biomedical Research Computing Facility (BRCF)

Genome Sequencing & Analysis Facility (GSAF)

Center for Systems and Synthetic Biology (CSSB)

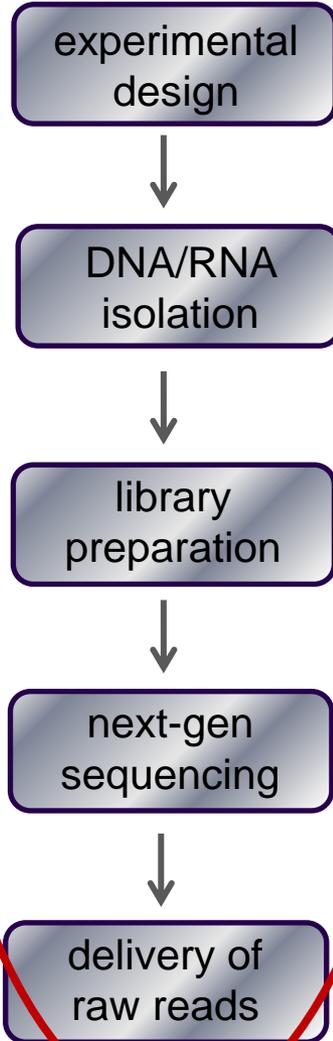
Ed Marcotte & Vishwanath Iyer labs



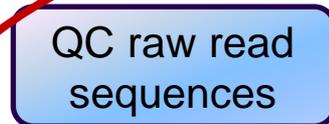
NGS Workflow

core processes

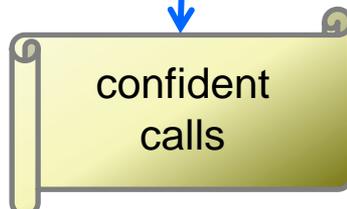
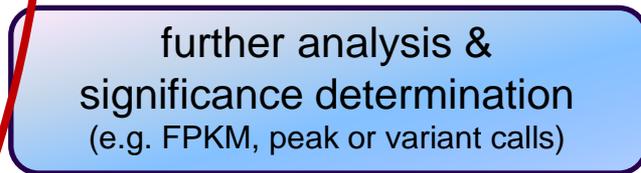
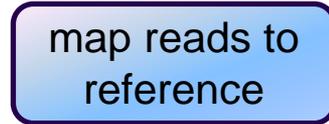
upstream processes



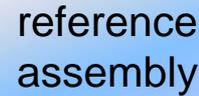
fastq



yes



has reference?



fasta

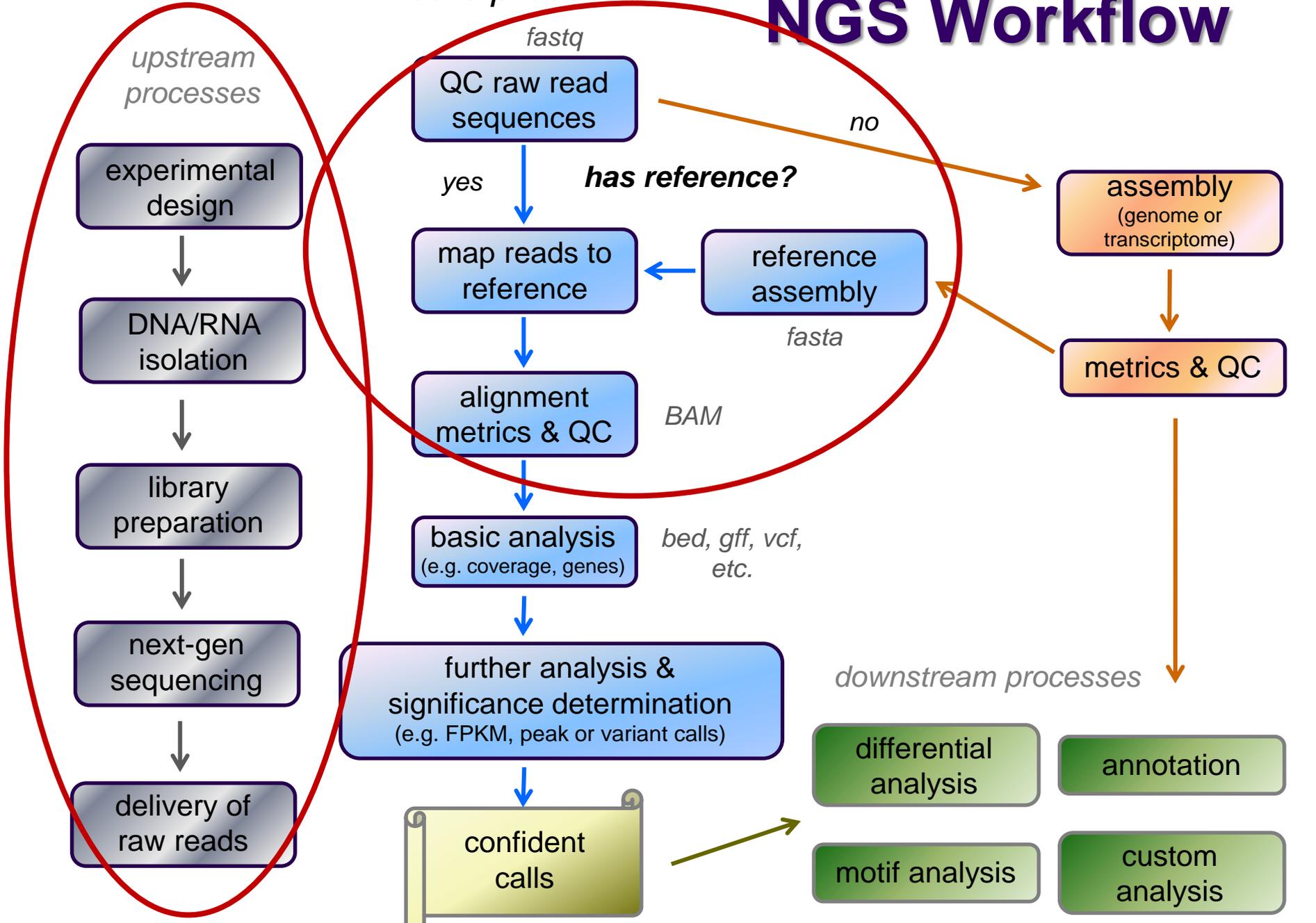
BAM

bed, gff, vcf, etc.

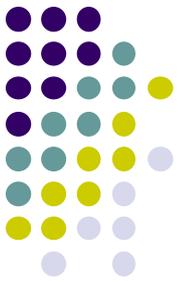
no



downstream processes



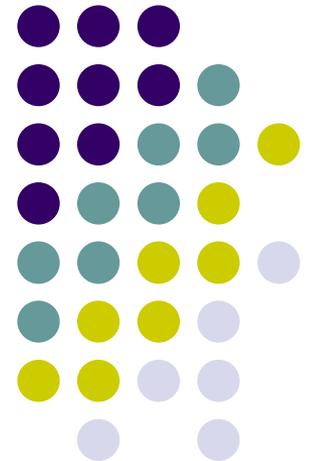
Outline



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

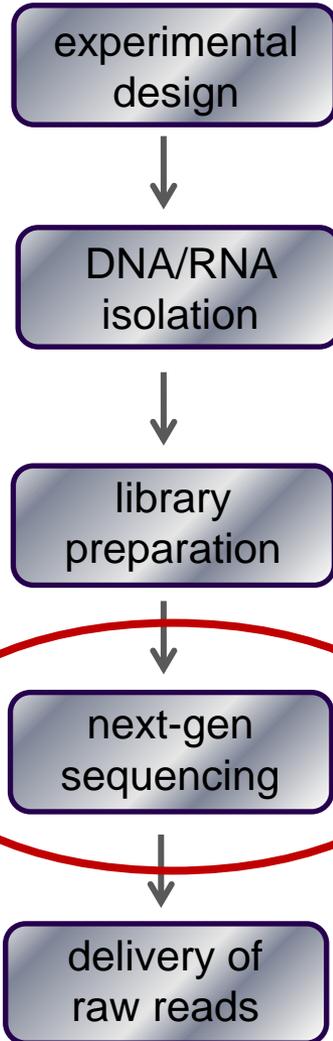
- High-throughput (“next gen”) sequencing
- Illumina short-read sequencing
- Long read sequencing



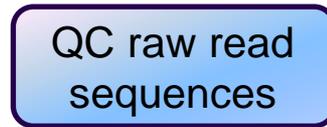
NGS Workflow

core processes

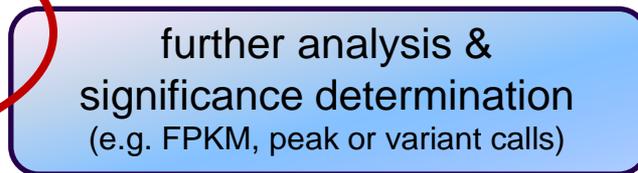
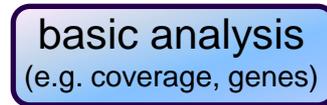
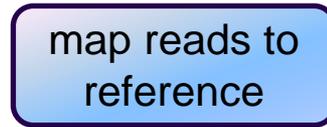
upstream processes



fastq



yes



has reference?

reference assembly

fasta

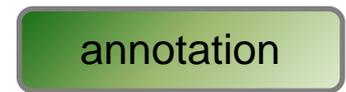
BAM

bed, gff, vcf, etc.

no



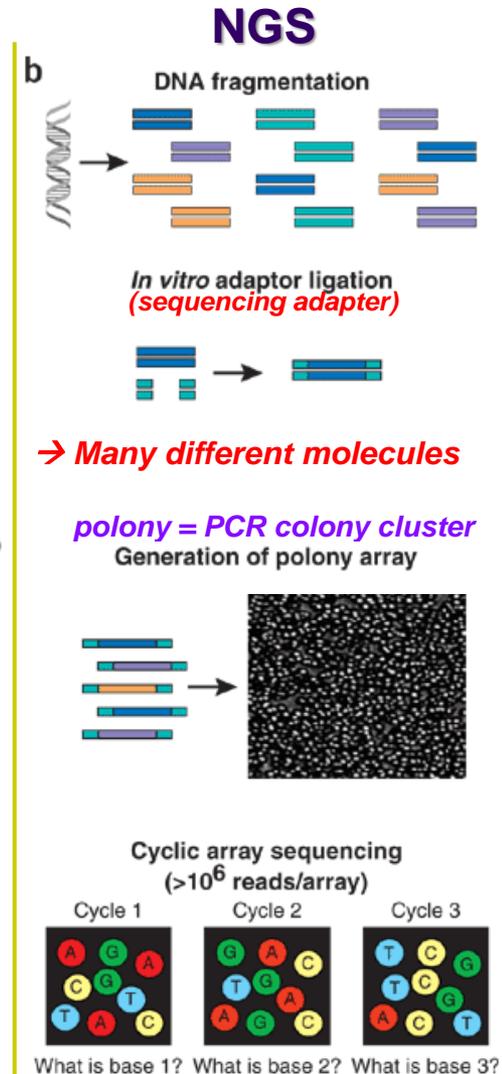
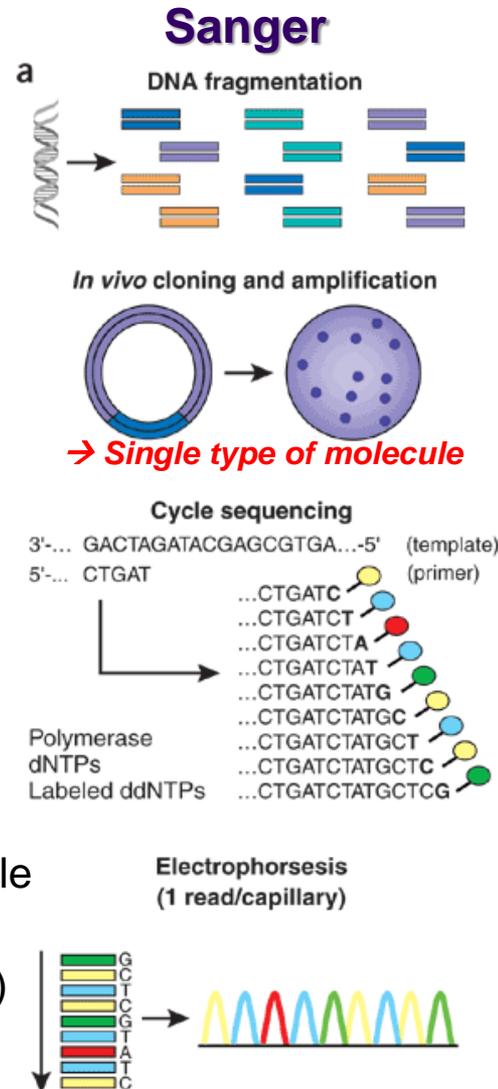
downstream processes



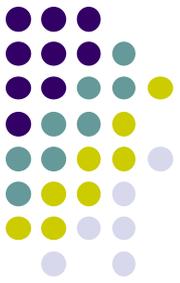
“Next Generation” sequencing



- 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)
 - results in 30 – 300 base “reads”
 - much shorter than Sanger sequencing



“Next Generation” sequencing (2nd generation)

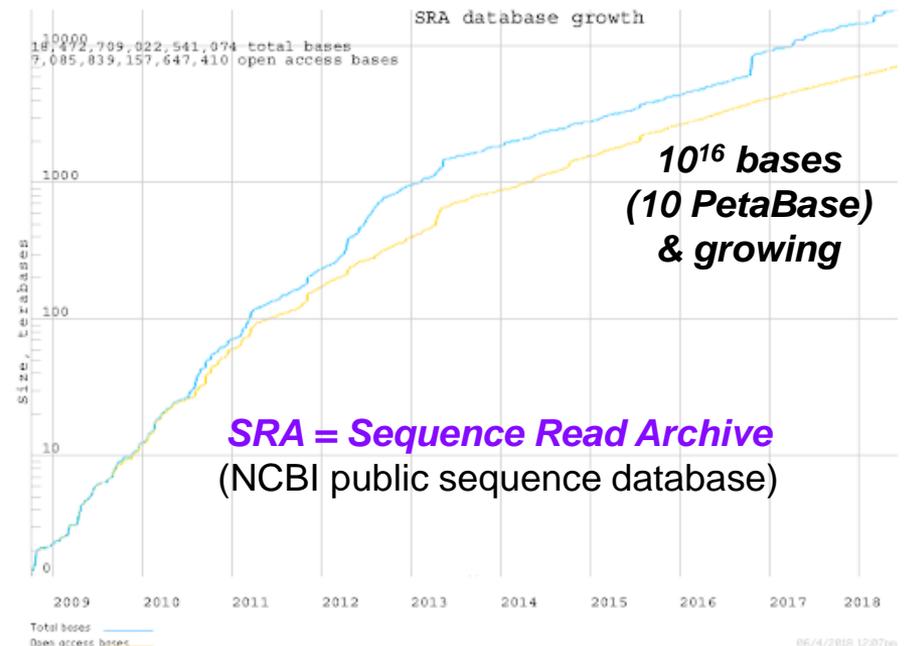
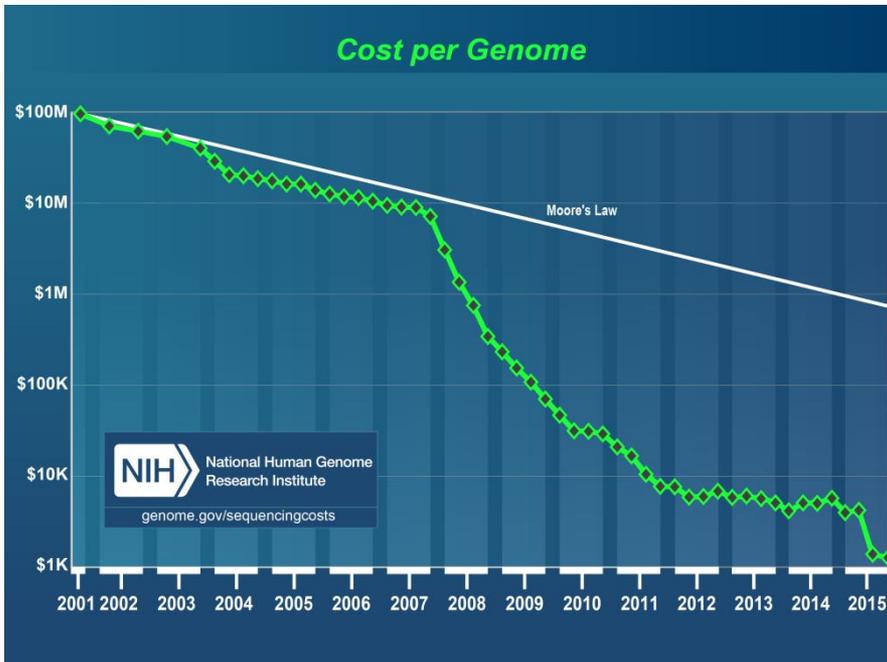


- Pro's:

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

- Con's:

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



Illumina sequencing



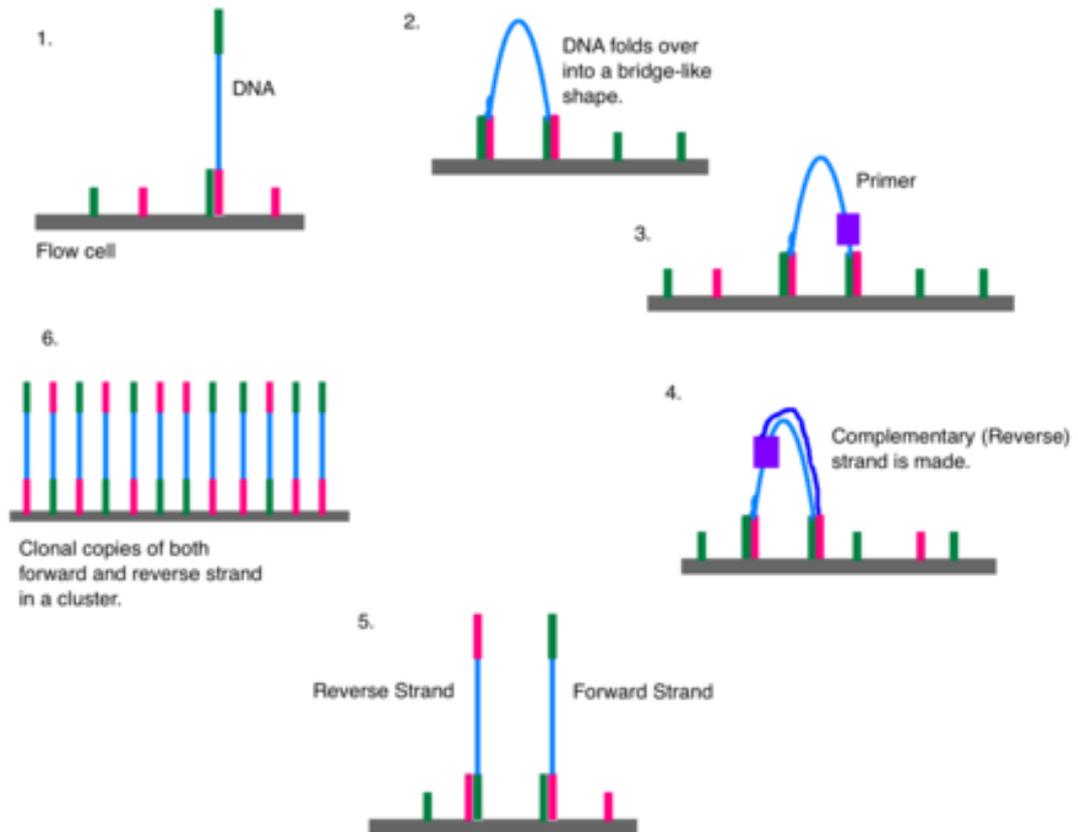
1. Library preparation
2. **Cluster generation via bridge amplification**
3. Sequencing by synthesis
4. Image capture
5. Convert to base calls

[Short Illumina video](https://tinyurl.com/hvnmwjb)

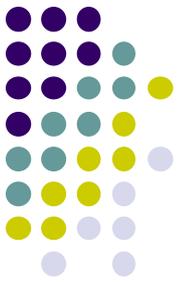
(<https://tinyurl.com/hvnmwjb>)

- Note

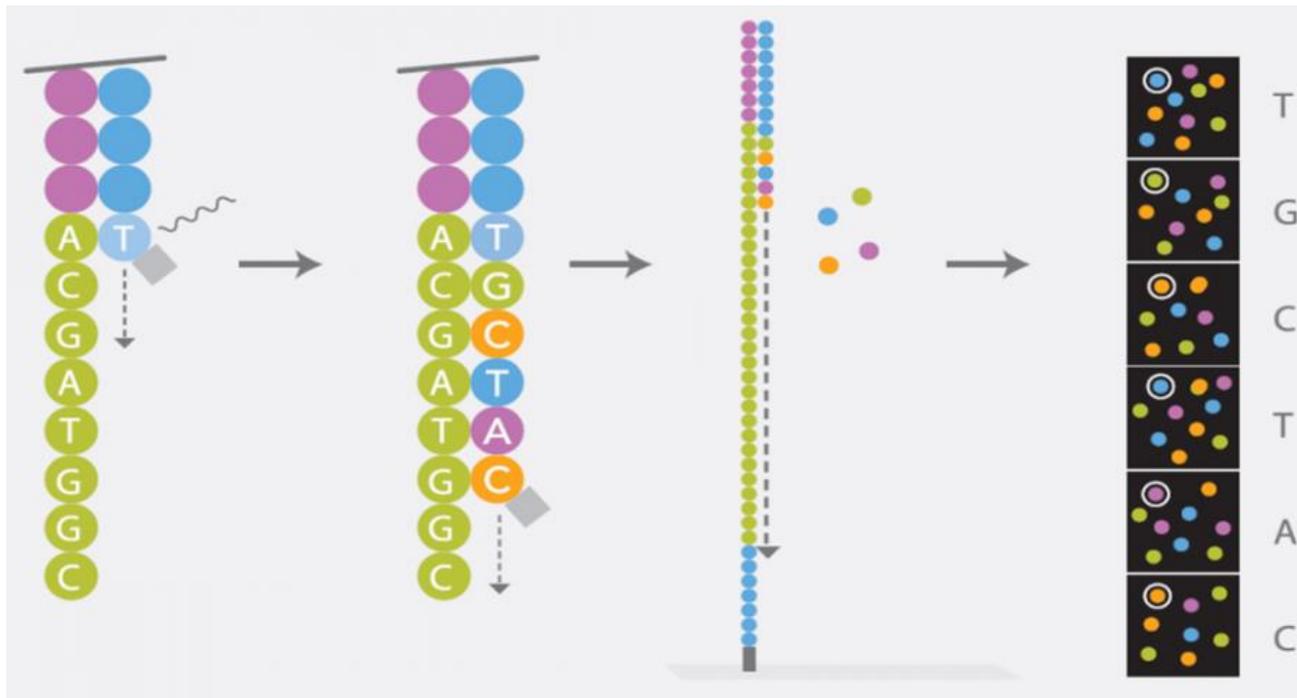
- 2 PCR amplifications performed
 1. during **library preparation**
 2. during **cluster generation**
- **amplification always introduces bias!**



Illumina sequencing



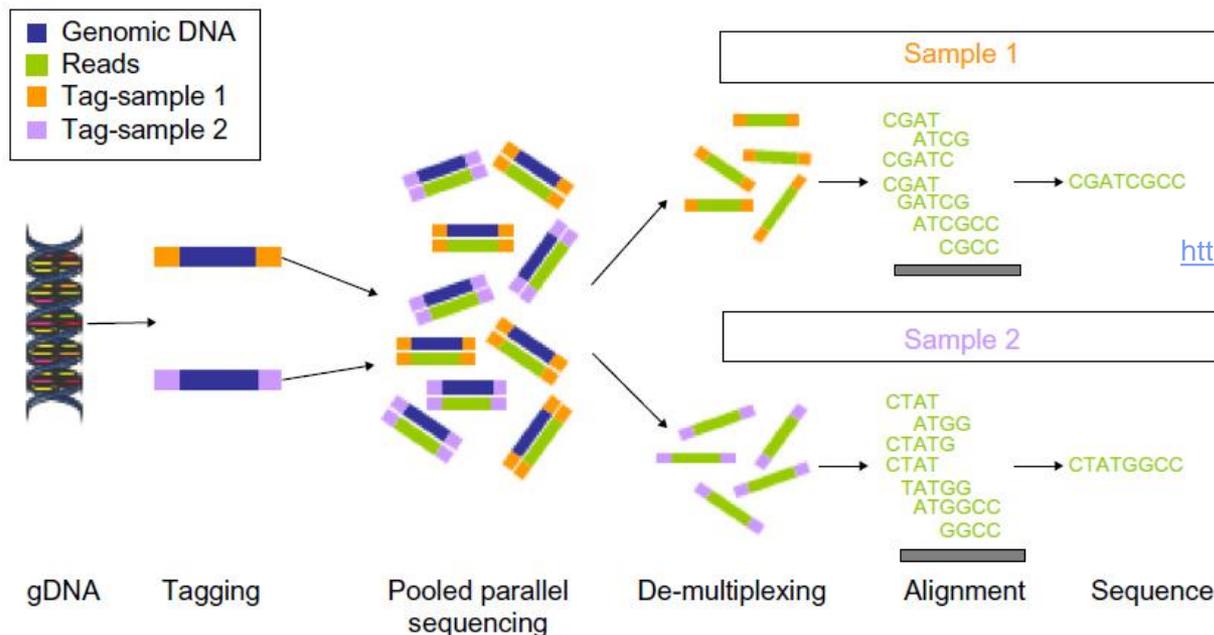
1. Library preparation
2. Cluster generation via bridge amplification
3. *Sequencing by synthesis*
4. *Image capture*
5. *Convert to base calls*



Multiplexing



- Illumina sequencers have one or more flowcell “lanes”, each of which can generate millions of reads
 - ~20**M** reads/lane for MiSeq, ~10**G** reads/lane for NovaSeq
- 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



Long read sequencing

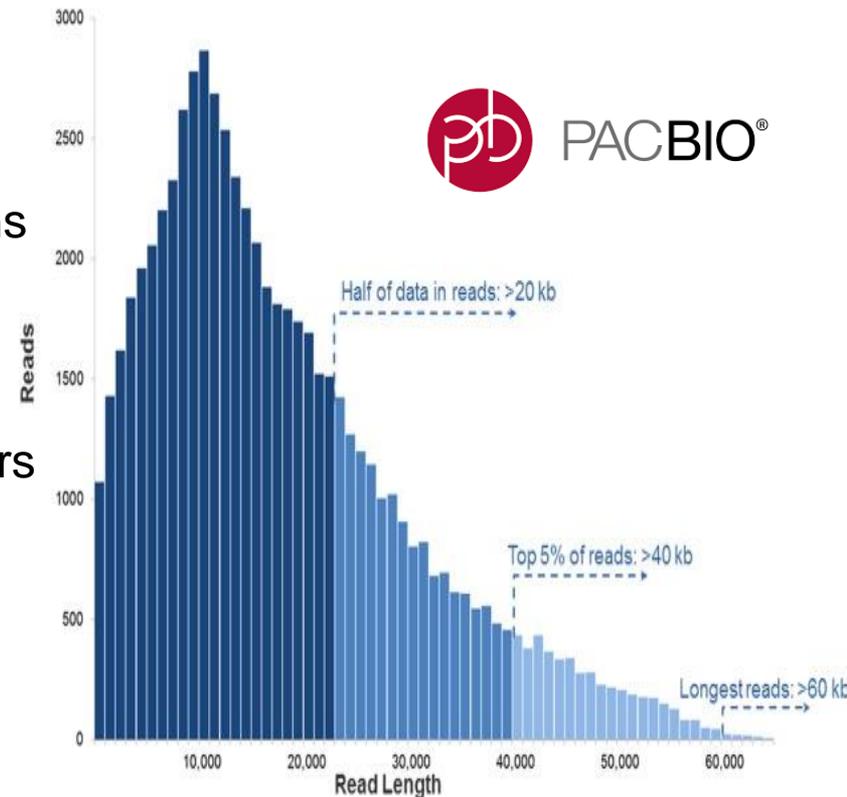


- Short read technology limitations
 - 30 – 300 base reads (150 typical)
 - PCR amplification bias
 - short reads are difficult to assemble
 - e.g., too short to span a long repeat region
 - difficult to detect large structural variations like inversions

- 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:

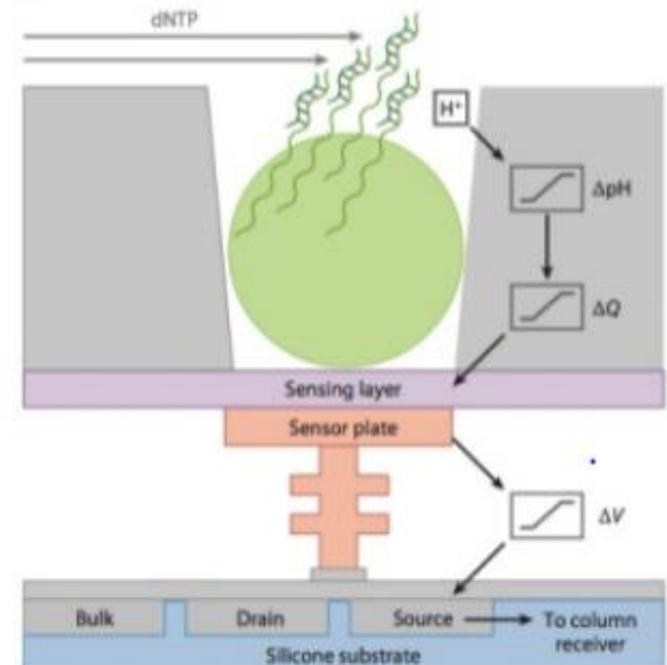
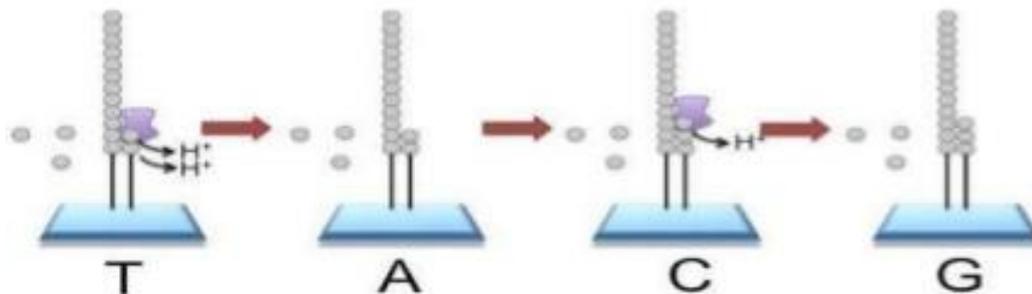
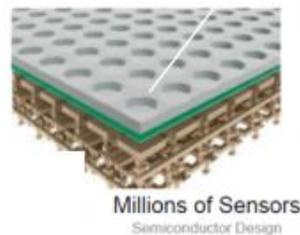
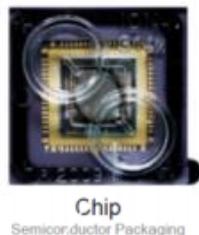
- weaker signal leads to high error rate
 - 10% vs <1% for Illumina
 - and fewer reads are generated (~100 K)



Long read sequencing



- Oxford Nanopore ION technology systems (e.g. MinION)
 - <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!
 - inexpensive, but high error rates (~10%)



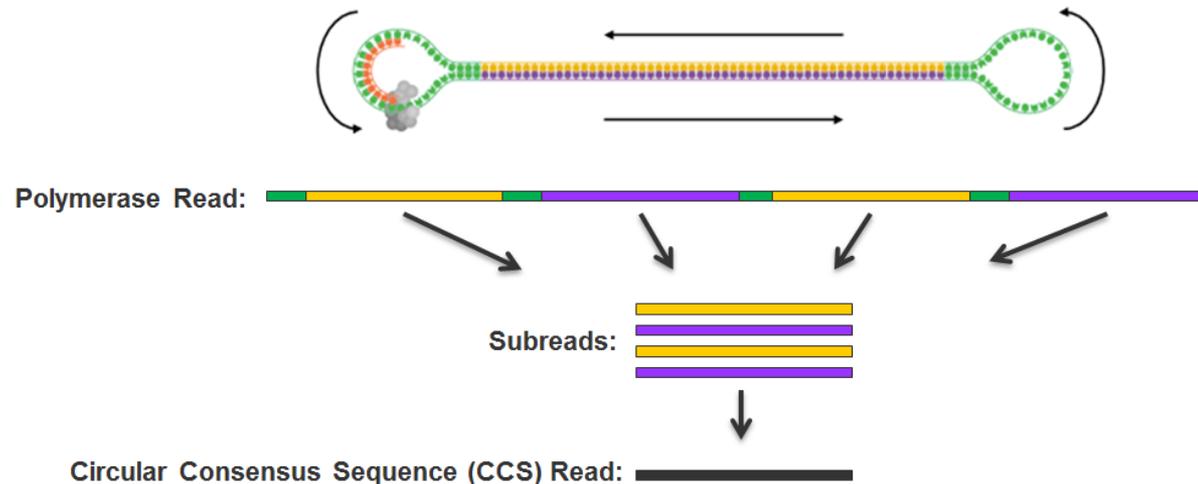
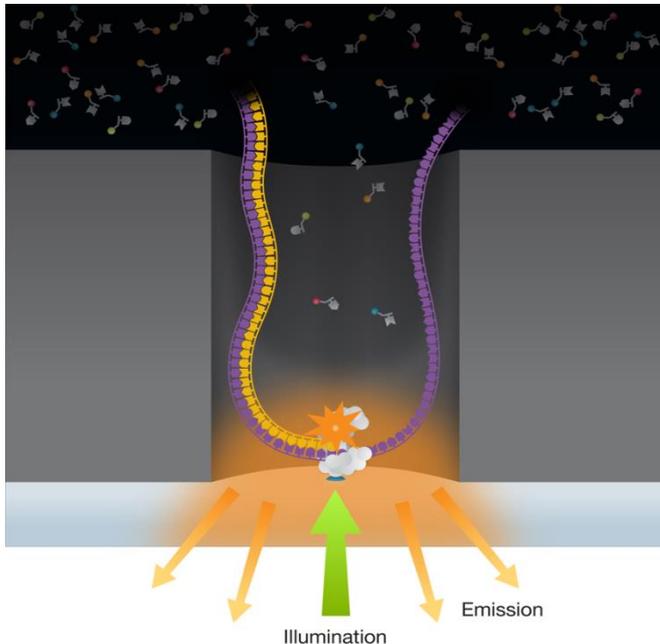
Long read sequencing



- PacBio SMRT system

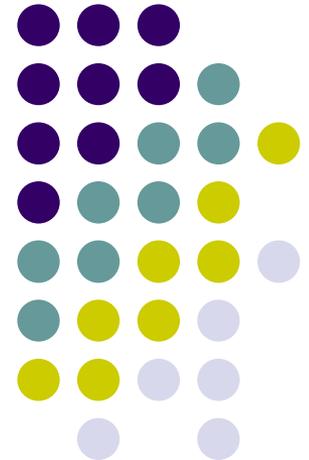


- <http://www.pacb.com/smrt-science/smrt-sequencing/>
- Sequencing by synthesis in **Zero-Mode Waveguide** (ZMW) wells
- DNA is circularized then repeatedly sequenced to achieve “consensus”
 - reduces error rate (~1-2%), but equipment *quite* expensive



Part 2: NGS Concepts & Terminology

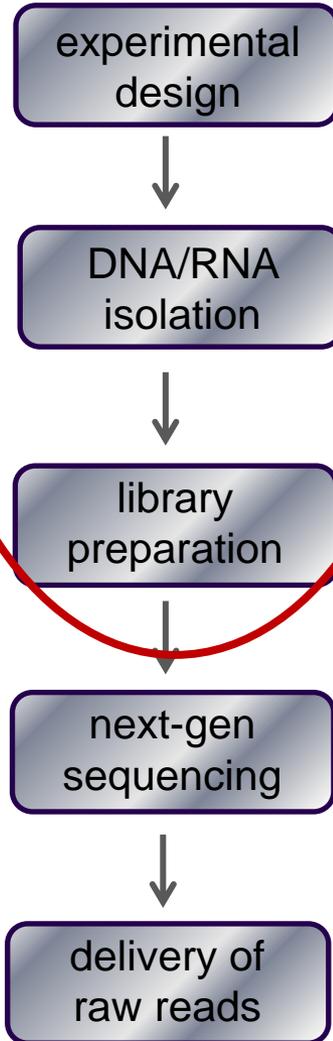
- Experiment types & library complexity
- Sequencing terminology
- Sequence duplication issues



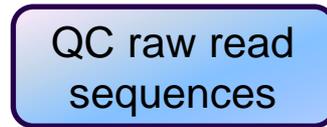
NGS Workflow

core processes

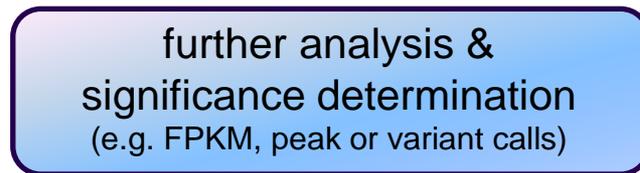
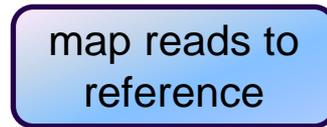
upstream processes



fastq



yes



has reference?

fasta

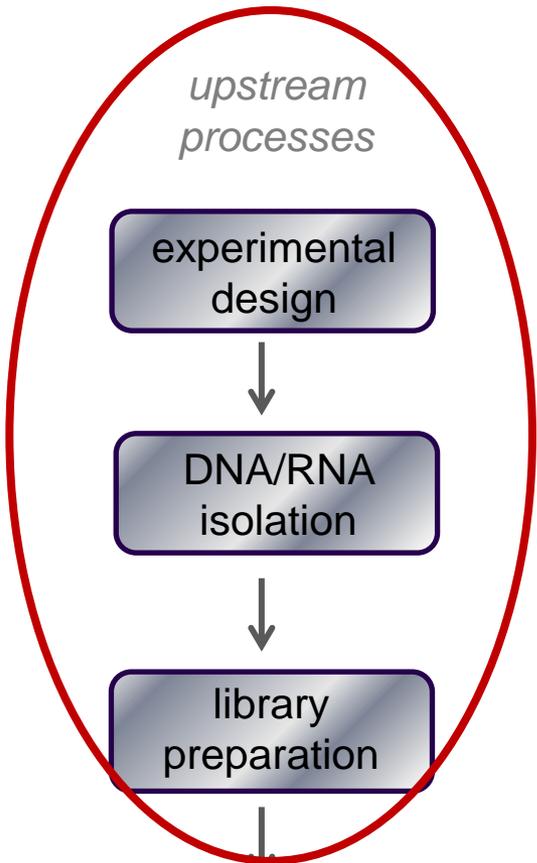
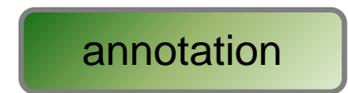
BAM

bed, gff, vcf, etc.

no



downstream processes





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.

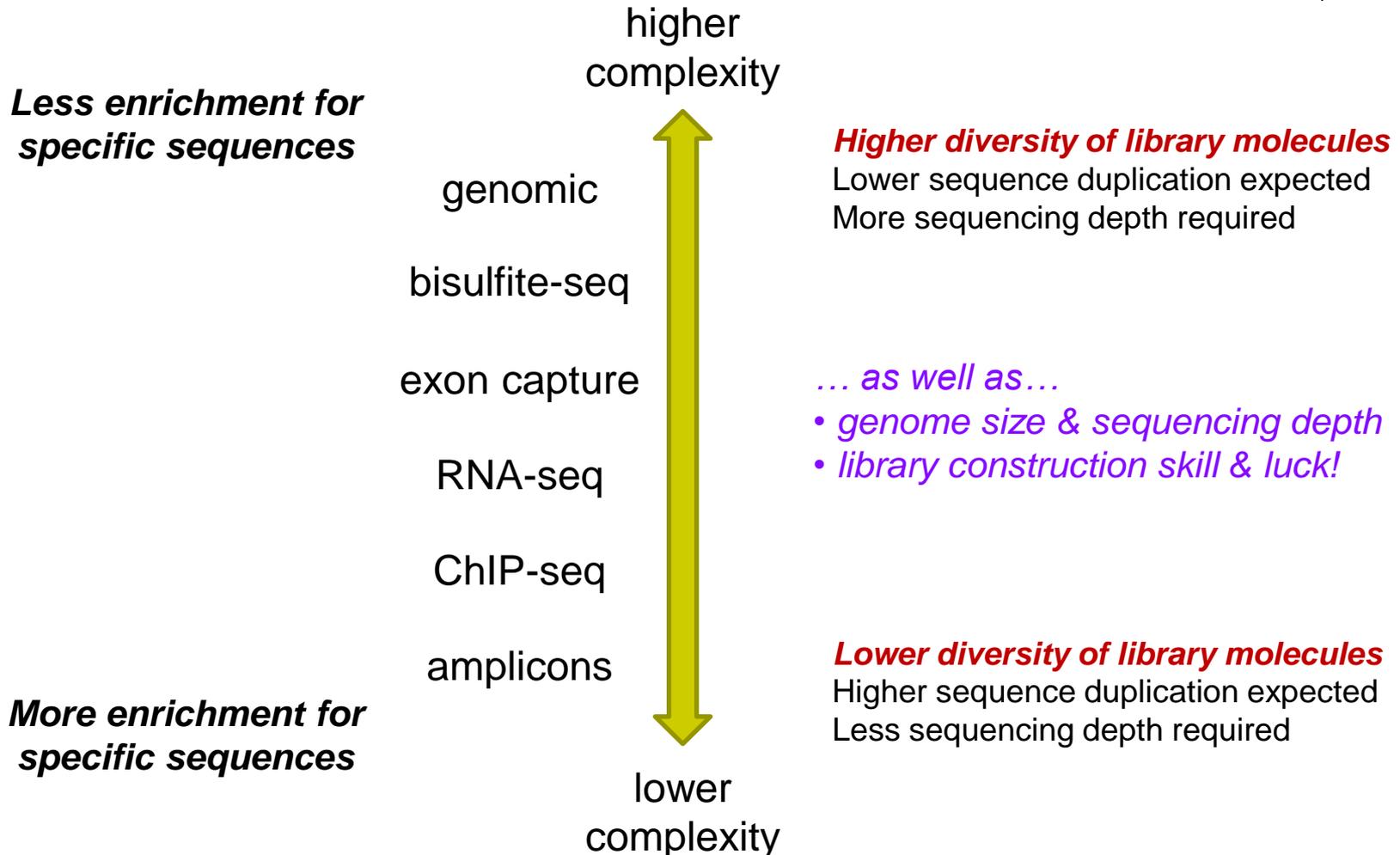
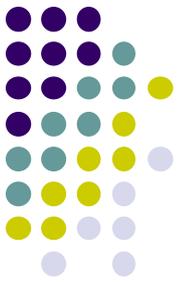
Popular Experiment Types



- **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

Type	Library construction	Applications	Complexity
Whole genome (WGS)	<ul style="list-style-type: none"> extract genomic DNA & fragment 	<ul style="list-style-type: none"> Genome assembly Variant detection, genotyping 	high
Bisulfite sequencing	<ul style="list-style-type: none"> bisulfite treatment converts C → U but not 5meC 	<ul style="list-style-type: none"> Methylation profiling (CpG) 	high
RAD-seq, ddRAD	<ul style="list-style-type: none"> restriction-enzyme digest DNA & fragment 	<ul style="list-style-type: none"> Variant detection (SNPs) Population genetics, QTL mapping 	high
Exome (WXS)	<ul style="list-style-type: none"> capture DNA from exons only (manufacturer kits) 	<ul style="list-style-type: none"> Variant detection, genotyping 	high-medium
ATAC-seq	<ul style="list-style-type: none"> high-activity transposase cuts DNA & ligates adapters 	<ul style="list-style-type: none"> Profile nucleosome-free regions (“open chromatin”) 	medium-high
RNA-seq, Tag-seq	<ul style="list-style-type: none"> extract RNA & fragment convert to cDNA 	<ul style="list-style-type: none"> Differential gene or isoform expression Transcriptome assembly 	medium, medium-low for Tag-seq
Transposon seq (Tn-seq)	<ul style="list-style-type: none"> create library of transposon-mutated genomic DNA amplify mutants via Tn-PCR 	<ul style="list-style-type: none"> Characterize genotype/phenotype relationships w/high sensitivity 	medium
ChIP-seq	<ul style="list-style-type: none"> cross-link proteins to DNA pull-down proteins of interest w/ specific antibody, reverse cross-links 	<ul style="list-style-type: none"> Genome-wide binding profiles of transcription factors, epigenetic marks & other proteins 	medium (but variable)
GRO-seq	<ul style="list-style-type: none"> isolate actively-transcribed RNA 	<ul style="list-style-type: none"> Characterize transcriptional dynamics 	medium-low
RIP-seq	<ul style="list-style-type: none"> like ChIP-seq, but with RNA 	<ul style="list-style-type: none"> Characterize protein-bound RNAs 	low-medium
miRNA-seq	<ul style="list-style-type: none"> isolate 15-25bp RNA band 	<ul style="list-style-type: none"> miRNA profiling 	low
Amplicons	<ul style="list-style-type: none"> amplify 1-1000+ genes/regions 	<ul style="list-style-type: none"> genotyping, metagenomics, mutagenesis 	low

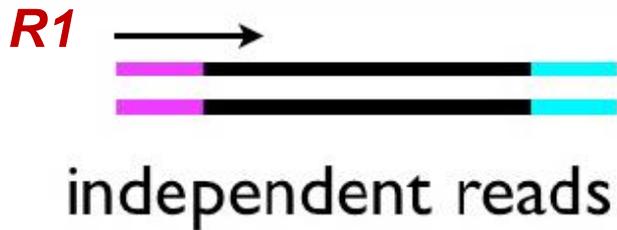
Library complexity is primarily a function of experiment type



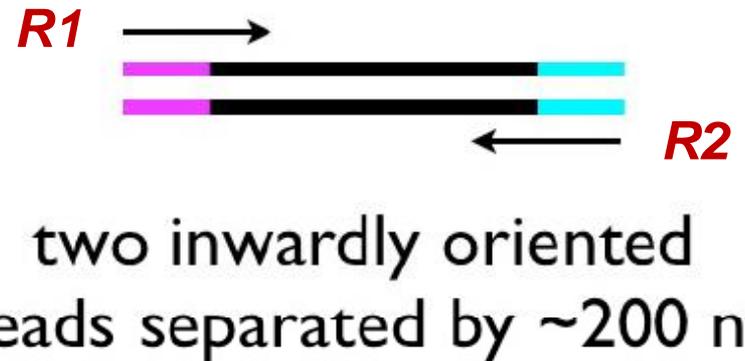
Read types



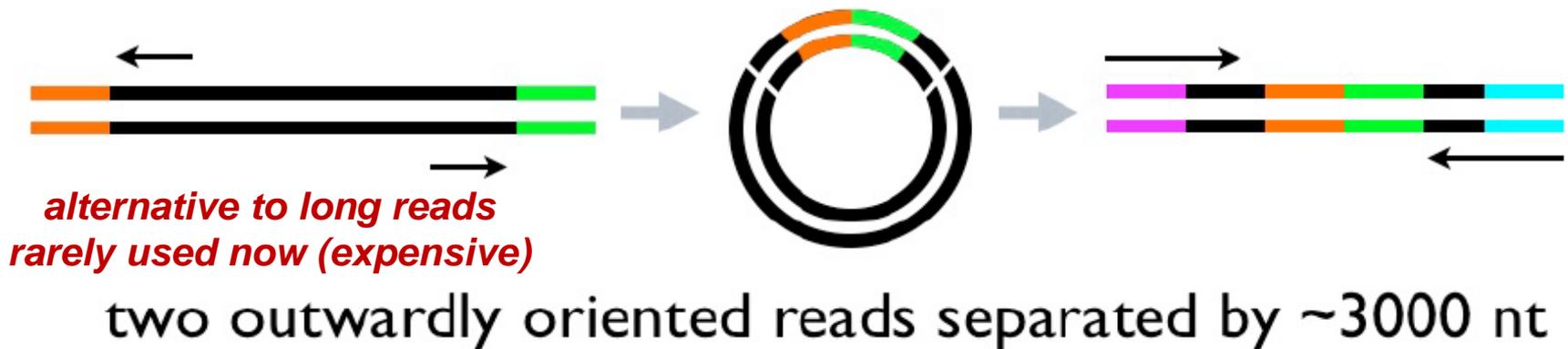
single-end



paired-end



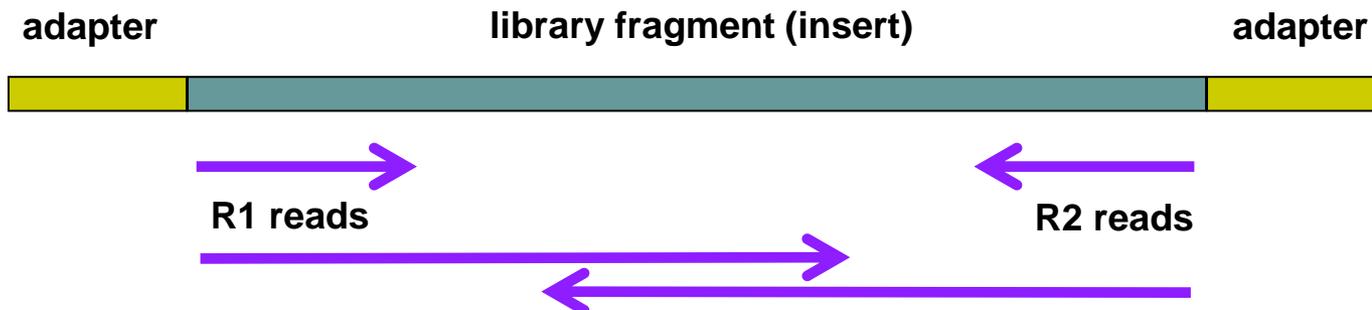
mate-paired



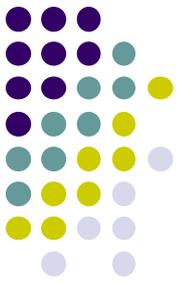


Reads and Fragments

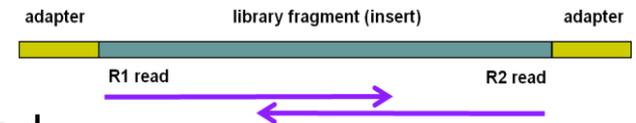
- With paired-end sequencing, keep in mind the distinction between
 - the library **fragment** from your library that was sequenced
 - also called **inserts**
 - the **sequence reads** (R1s & R2s) you receive
 - also called **tags**
- An R1 and its associated R2 form a **read pair**
 - a readout of part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



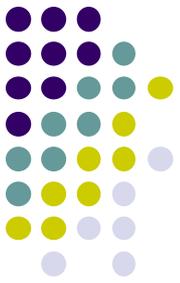
Single end vs Paired end



- **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 experiment types

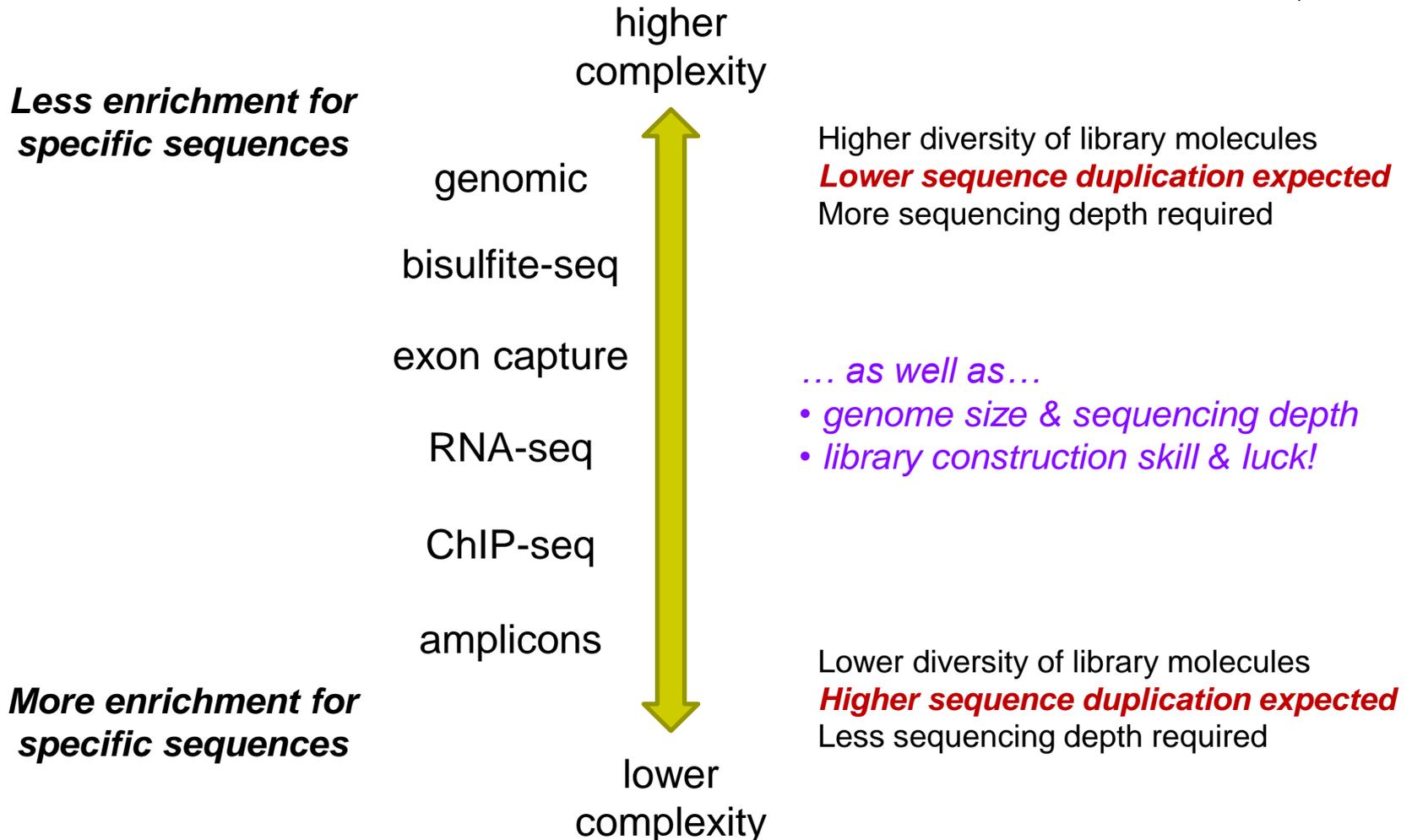
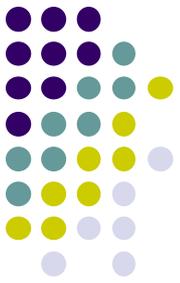


Sequence Duplication



- 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
 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

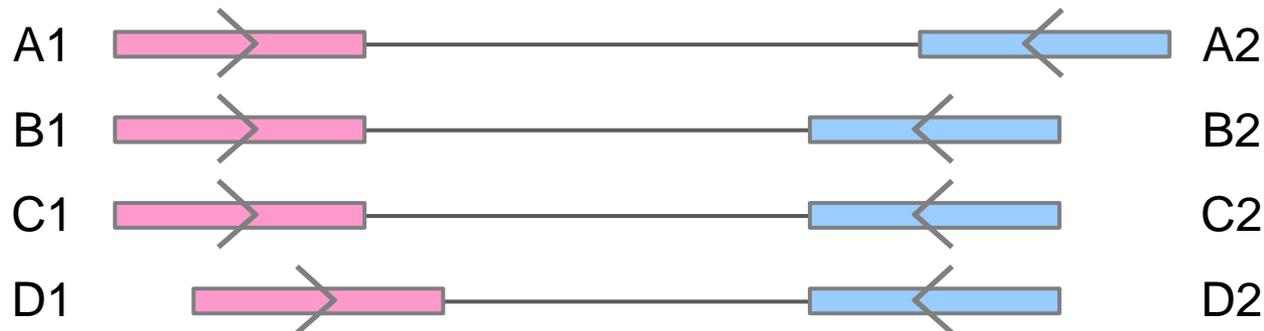
Expected sequence duplication is primarily a function of experiment type



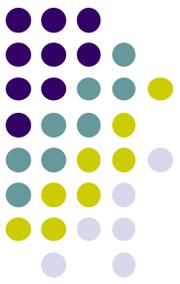


Read vs Fragment duplication

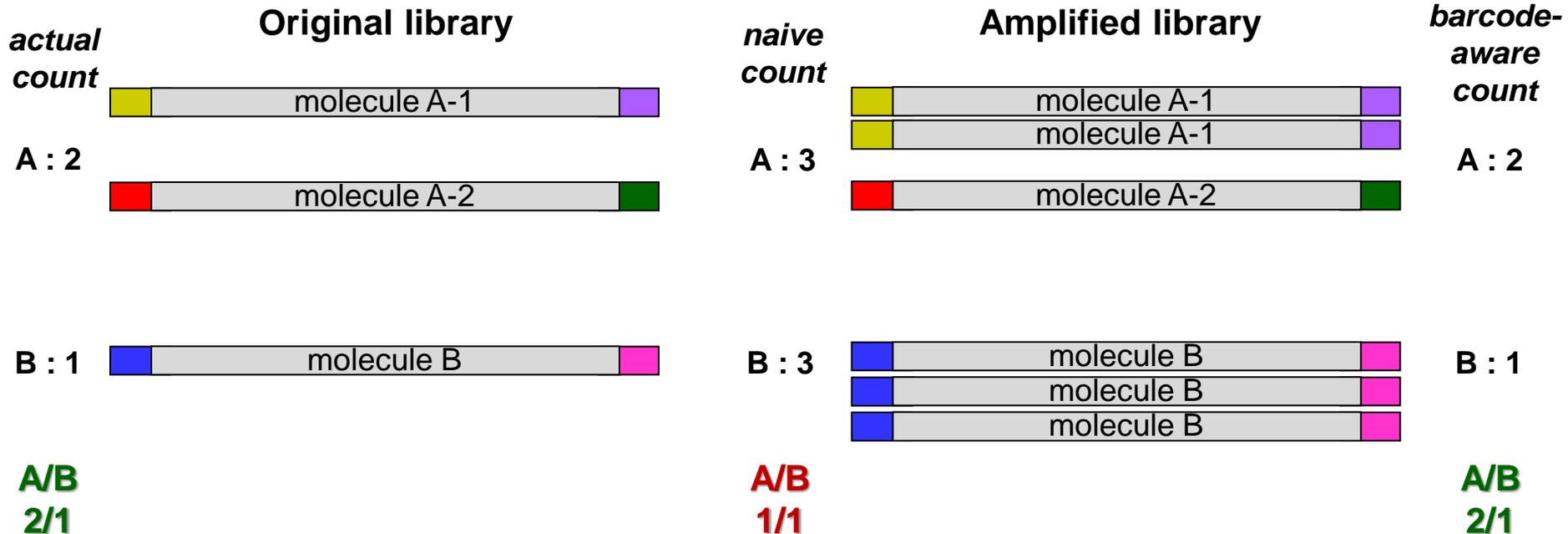
- Consider the 4 “aligned” fragments below
 - 4 R1 reads (pink), 4 R2 reads (blue)
- Duplication when only 1 end considered
 - A1, B1, C1 have identical sequences, D1 different
 - 2 unique + 2 duplicates = 50% duplication rate
 - B2, C2, D2 have identical sequences, A2 different
 - 2 unique + 2 duplicates = 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



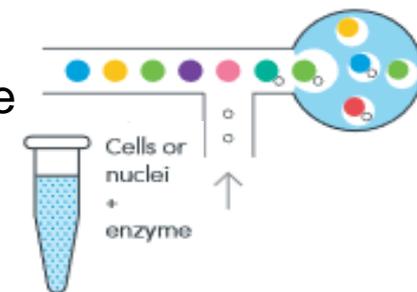
- Resolves ambiguity between biological and technical (PCR amplification) duplicates
 - adds secondary, **internal** barcodes to **pre-PCR** molecules
 - a.k.a **UMIs** (**U**nique **M**olecular **I**ndexes)
 - combination of barcodes + insert sequence provides accurate quantification
 - but requires specialized pre- and post-processing
 - e.g. UMI de-duplication for 3' Tag-seq, scRNA-seq



Single Cell sequencing



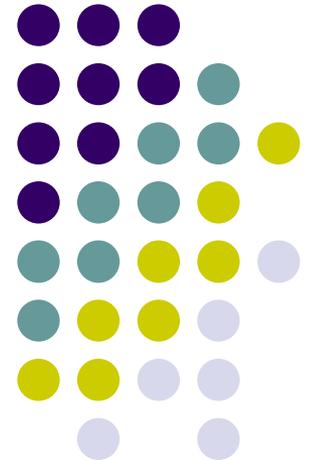
- Standard sequencing library starts with millions of cells
 - will be in different states unless synchronized
 - a heterogeneous “ensemble” with (possibly) high cell-to-cell variability
- **Single cell sequencing** technologies aim to capture this variability
 - examples:
 - cells in different layers/regions of somatic tissue
 - cells in different areas of a tumor
 - essentially a very sophisticated library preparation technique
- Typical protocol (RNA-seq)
 1. isolate a few thousand cells (varying methods)
 2. the single-cell platform partitions each cell into an emulsion droplet
 - e.g. 10x Genomics (<https://www.10xgenomics.com/solutions/single-cell/>)
 3. a different barcode is added to the RNA in each cell
 4. resulting library submitted for standard Illumina short-read sequencing
 5. custom downstream analysis links results to their cell (barcode) of origin



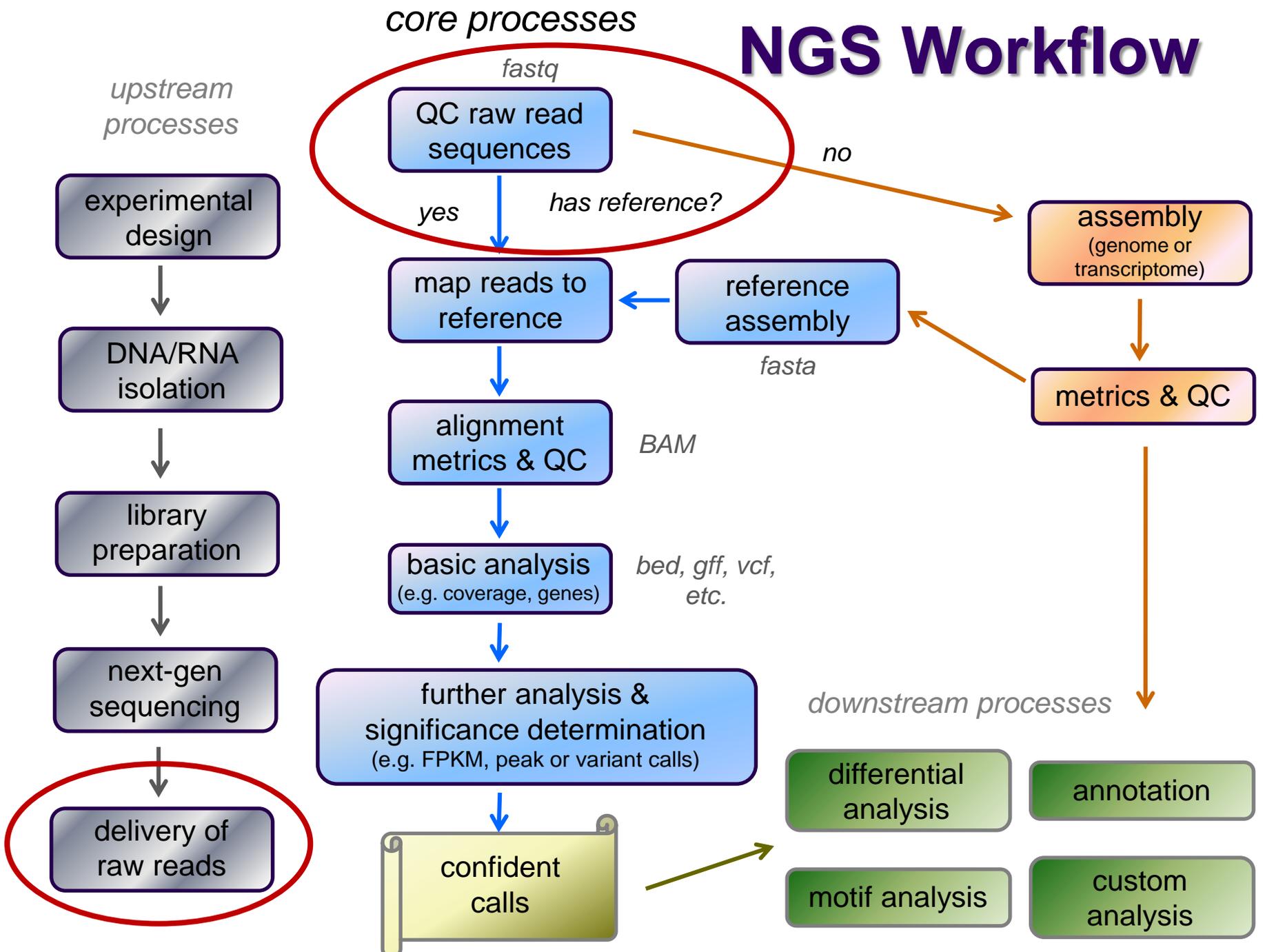
Part 3:

The FASTQ format, Data QC & preparation

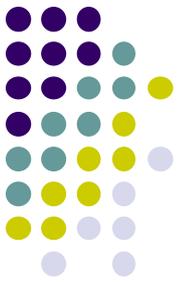
- The FASTQ format
- QC of raw sequences with **FastQC** tool
- Dealing with adapters



NGS Workflow

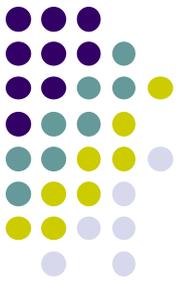


FASTQ format



- Text format for storing sequence and quality data
 - http://en.wikipedia.org/wiki/FASTQ_format
- 4 lines per sequence:
 1. **@read name** (plus extra information after a space)
 - *R1 and R2 reads have the same read name*
 2. **called base sequence (ACGTN)**
always 5' to 3'; usually excludes 5' adapter
 3. **+optional read information**
 4. **base quality scores encoded as text characters**
- FASTQ representation of a single, 50 base R2 sequence

```
@HWI-ST1097:97:D0WW0ACXX:4:1101:2007:2085 2:N:0:ACTTGA  
ATTCTCCAAGATTTGGCAAATGATGAGTACAATTATATGCCCAATTTACA  
+  
?@@?DD;?;FF?HHBB+:ABECGHDHDCF4?FGIGACFDHFH;FHEIIB9?
```



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

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

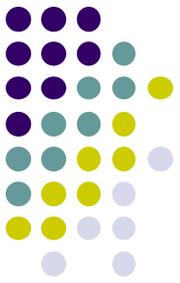
- Integer Phred score converted to Ascii character (add 33)

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

Quality character	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?	@	A	B	C	D	E	F	G	H	I	J
ASCII Value	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	
Base Quality (Q)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	

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

Raw sequence quality control



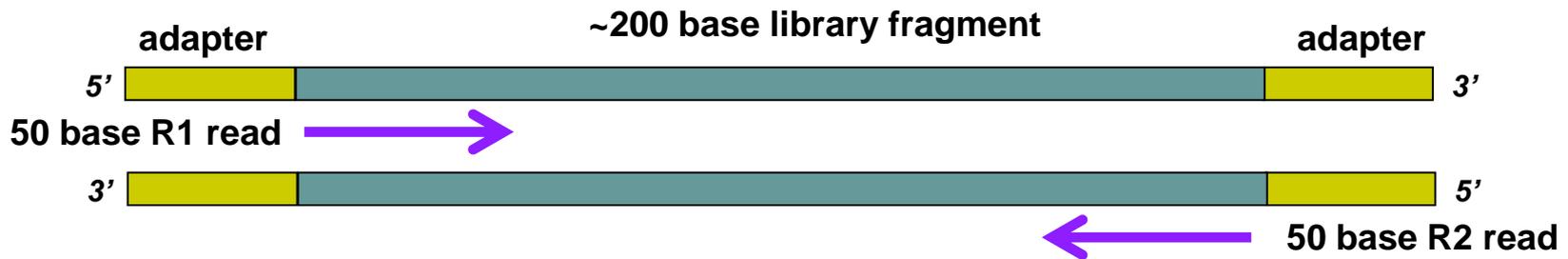
- Critical step! Garbage in = Garbage out
 - general sequence quality
 - base quality distributions
 - sequence duplication rate
 - trim 3' bases with poor quality?
 - important for *de novo* assembly
 - trim 3' adapter sequences?
 - important for RNAseq
 - other contaminants?
 - biological – rRNA in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





3' Adapter contamination

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

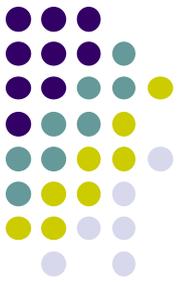


B. Reads long compared to library fragment (3' adapter contamination)



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

FastQC

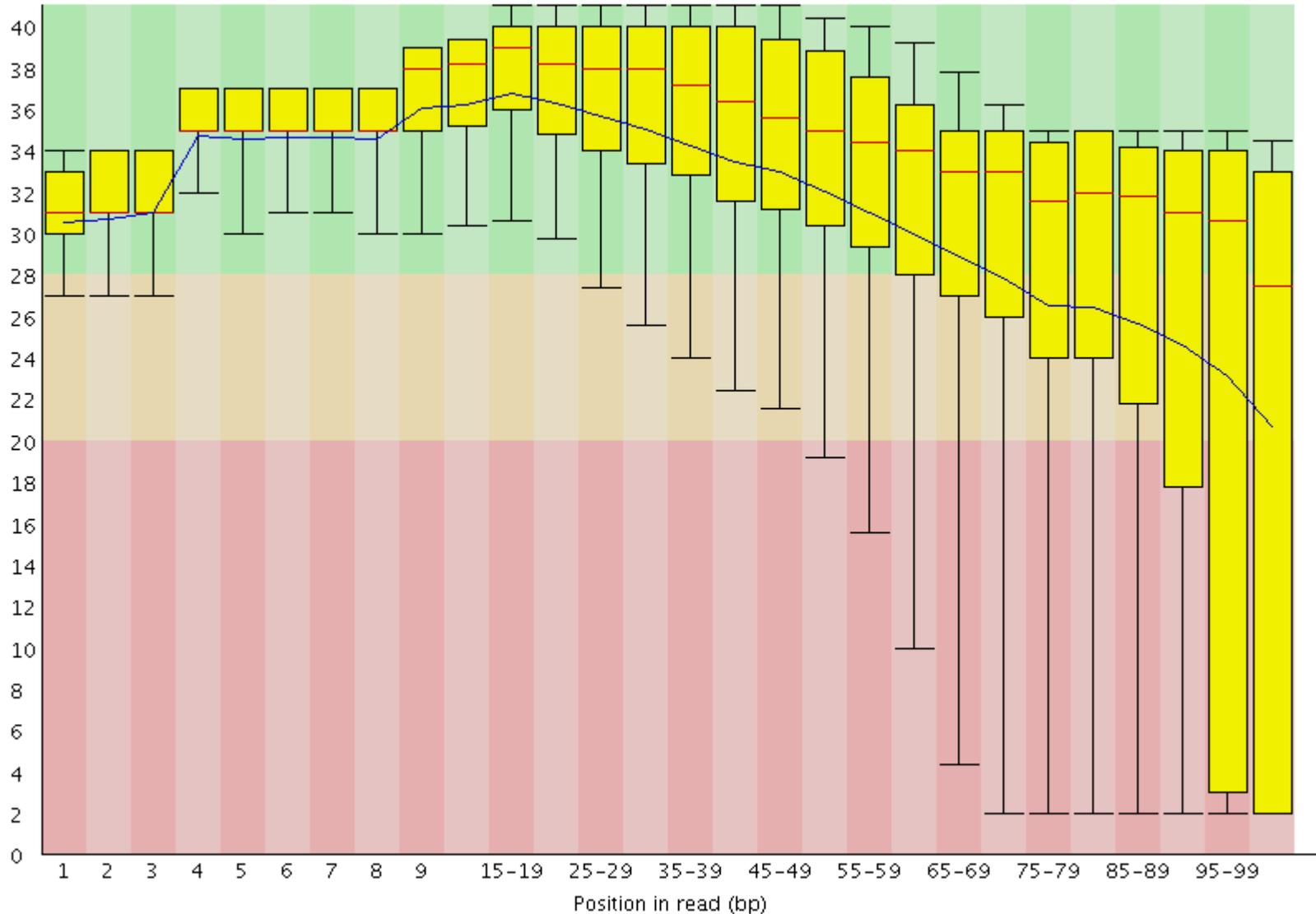


- Quality Assurance tool for FASTQ sequences
<http://www.bioinformatics.babraham.ac.uk>
- Most useful reports:
 1. *Per-base sequence quality Report*
 - Should I trim low quality bases?
 2. *Sequence duplication levels Report*
 - How complex is my sequence library?
 3. *Overrepresented sequences Report*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

1. FastQC Per-base sequence quality report



Quality scores across all bases (Sanger / Illumina 1.9 encoding)



2. FastQC Sequence duplication report

Yeast ChIP-seq

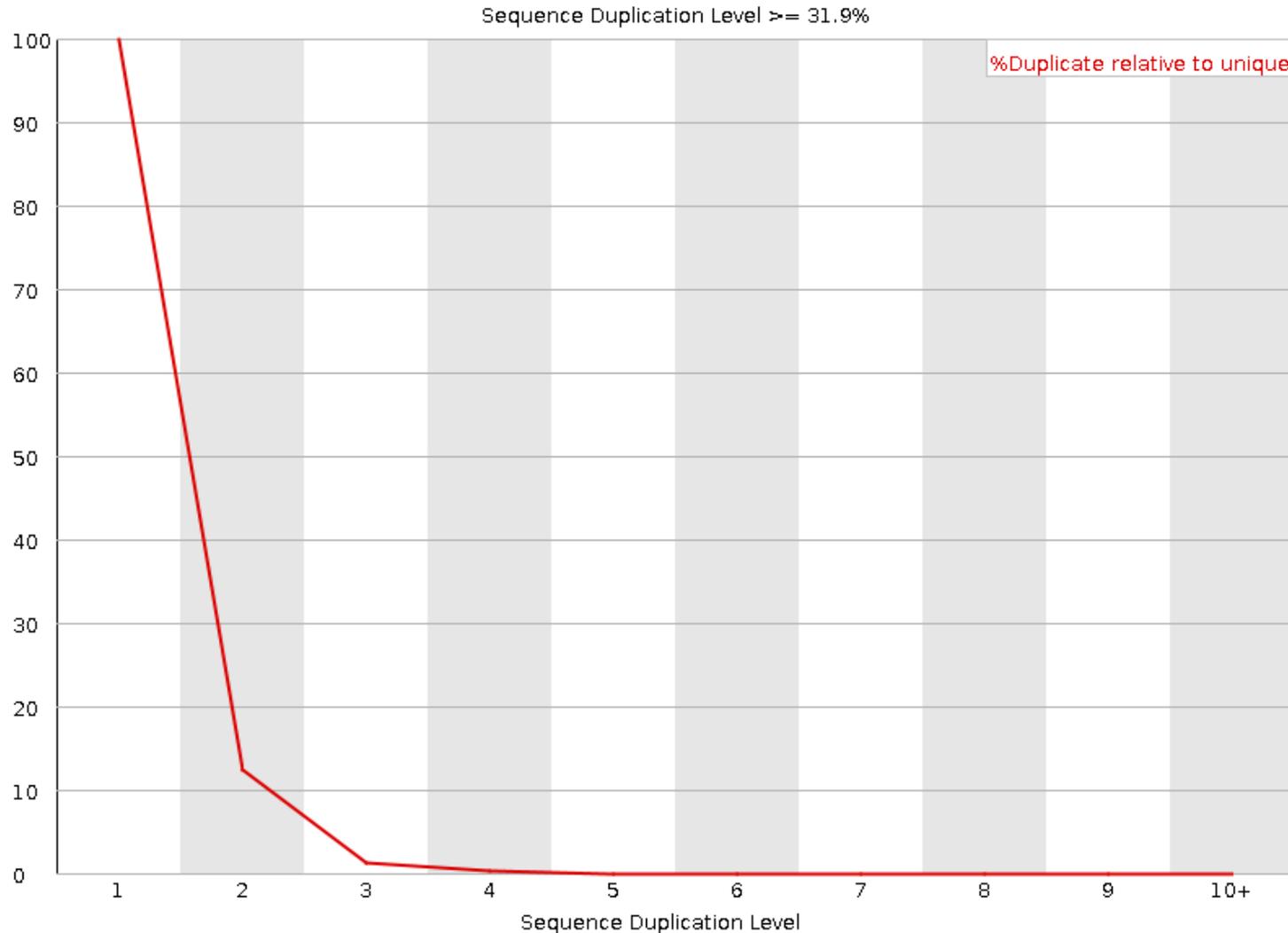


For every 100 unique sequences there are:

~12 sequences w/2 copies

~1-2 with 3 copies

Ok – Some duplication expected due to IP enrichment

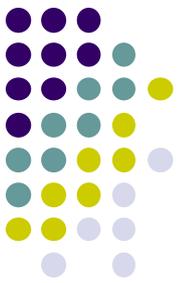




3. FastQC Overrepresented sequences report

- **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)

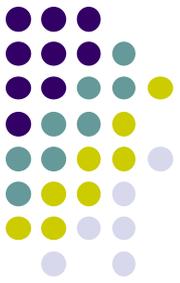


3. Overrepresented sequences

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

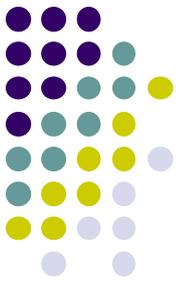
Sequence	Count	Percentage	Possible Source
GAAGGTCACGGCGAGACGAGCCGTTTATCATTACGATAGGTGTCAAGTGG	5632816	32.03026785752871	No Hit
TATTCTGGTGTCTTAGGCGTAGAGGAACAACACCAATCCATCCCGAACTT	494014	2.8091456822607364	No Hit
TCAAACGAGGAAAGGCTTACGGTGGATACCTAGGCACCCAGAGACGAGGA	446641	2.539765344040083	No Hit
TAAAACGACTCTCGGCAACGGATATCTCGGCTCTCGCATCGATGAAGAAC	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
ATTACGATAGGTGTCAAGTGGAAAGTGCAGTGATGTATGCAGCTGAGGCAT	73682	0.41898300890326096	No Hit
GAAGGTCACGGCGAGACGAGCCGTTTATCATTACGATAGGTGTCAAGGGG	71661	0.4074908580252516	No Hit
GGATGCGATCATACCAGCACTAATGCACCGGATCCCATCAGAACTCCGCA	69548	0.3954755612388914	No Hit
ATATTCTGGTGTCTTAGGCGTAGAGGAACAACACCAATCCATCCCGAACT	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 (*always needed for RNA-seq alignment*)
 - *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

FASTQ trimming and adapter removal



- 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

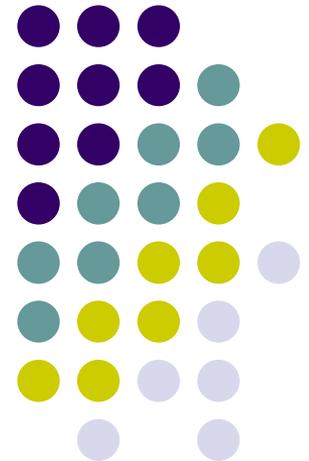
CTAGCTTATCGCCCTGAAGGACT

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

reference sequence

Part 4: Alignment to a reference assembly

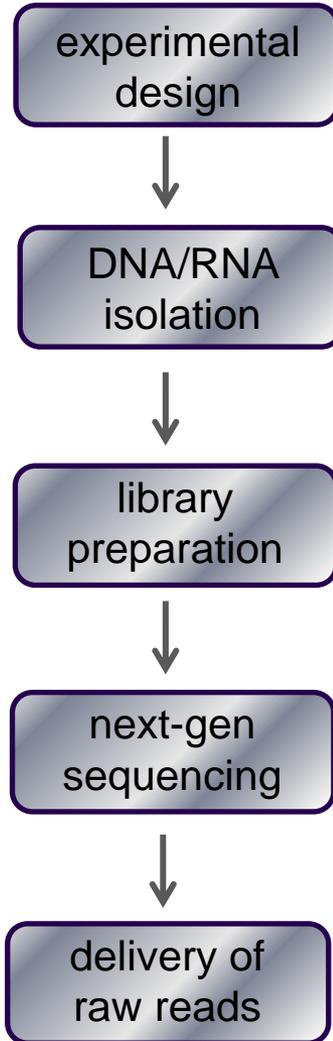
- Alignment overview & concepts
- Preparing a reference genome
- Alignment workflow steps



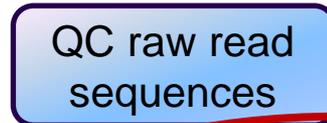
NGS Workflow

core processes

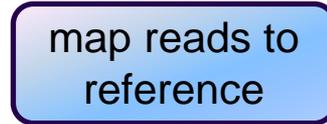
upstream processes



fastq



yes



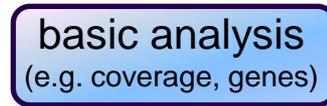
has reference?



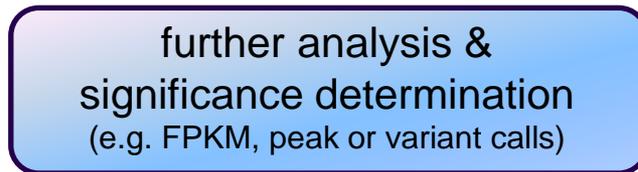
fasta



BAM



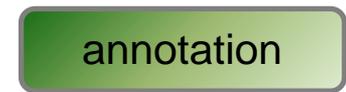
bed, gff, vcf, etc.



no



downstream processes



Short Read Aligners



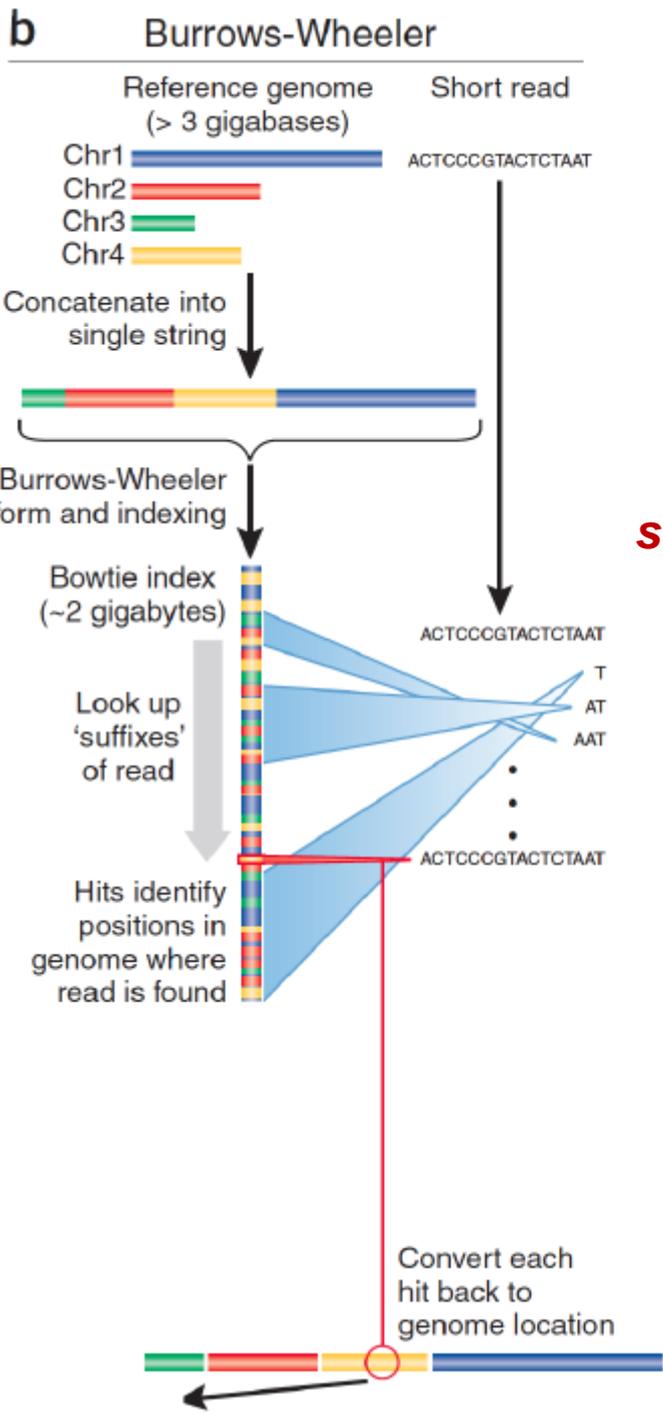
- Short read mappers determine placement of *query sequences* (your reads) against a known *reference*
 - **BLAST**:
 - one query sequence (or a few)
 - many matches for each
 - short read aligners
 - many millions of query sequences
 - want only one “best” mapping (or a few)
- Many aligners available! 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: $P(\text{incorrectly mapped}) = 10^{-\text{mappingQuality}/10}$
 - **alignment score** describes **fit**
 - reflects the correspondence between the read and the reference sequence

<ul style="list-style-type: none"> • Maps to one location <i>high mapping quality</i> • Has 2 mismatches <i>low alignment score</i> 	Read 1	Read 2	<ul style="list-style-type: none"> • Maps to 2 locations <i>low mapping quality</i> • Matches perfectly <i>high alignment score</i>
	GCGTAGTCTGCC TAGCCTAGTGTGCCGC	ATCGGGAGATCC TAATCGGGAGATCCGC	
	<i>reference sequence</i>		

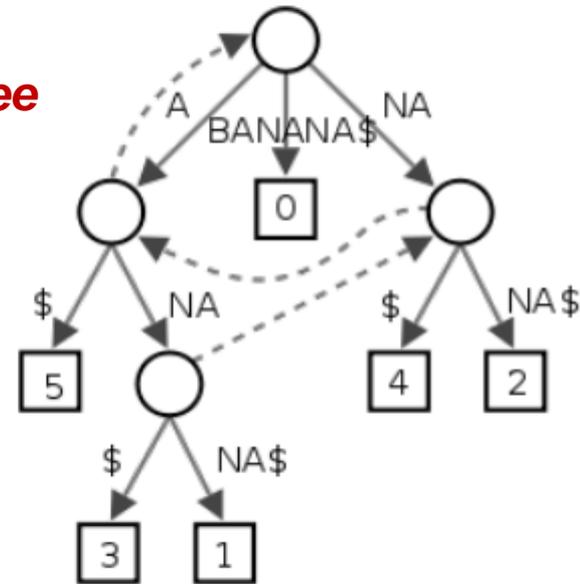


Burrows-Wheeler transform compresses sequence.

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

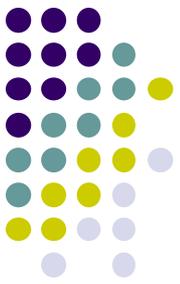


http://en.wikipedia.org/wiki/Suffix_tree

Exact matches at all positions below a node.

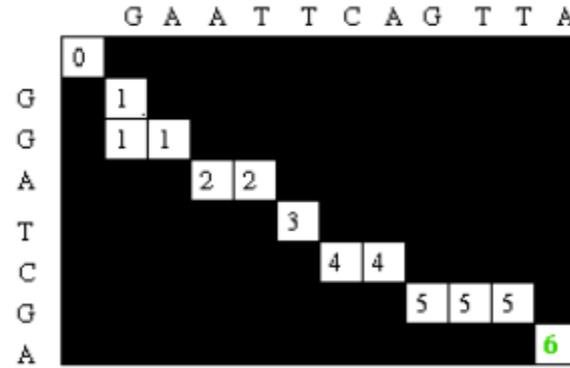
Trapnell, C. & Salzberg, S. L. How to map billions of short reads onto genomes. *Nature Biotech.* **27**, 455–457 (2009).

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
G	0	1	1	1	1	1	1	2	2	2	2
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 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

|||||

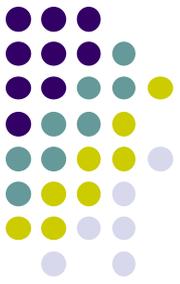
ATTTGCGATCGGATGTTGACTTT

ATTTGCGATCGGATGAAGACG..AA

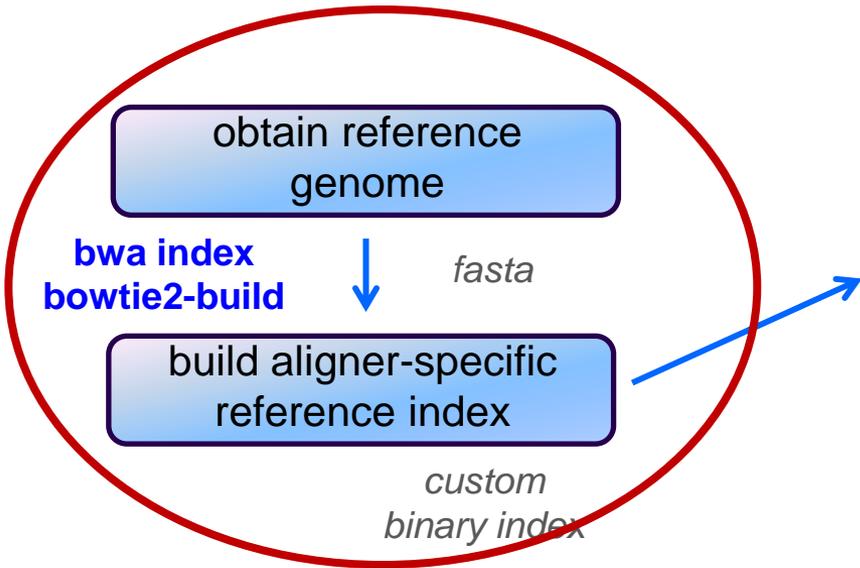
|||||XX|||Xi|||

ATTTGCGATCGGATGTTGACTTTAA

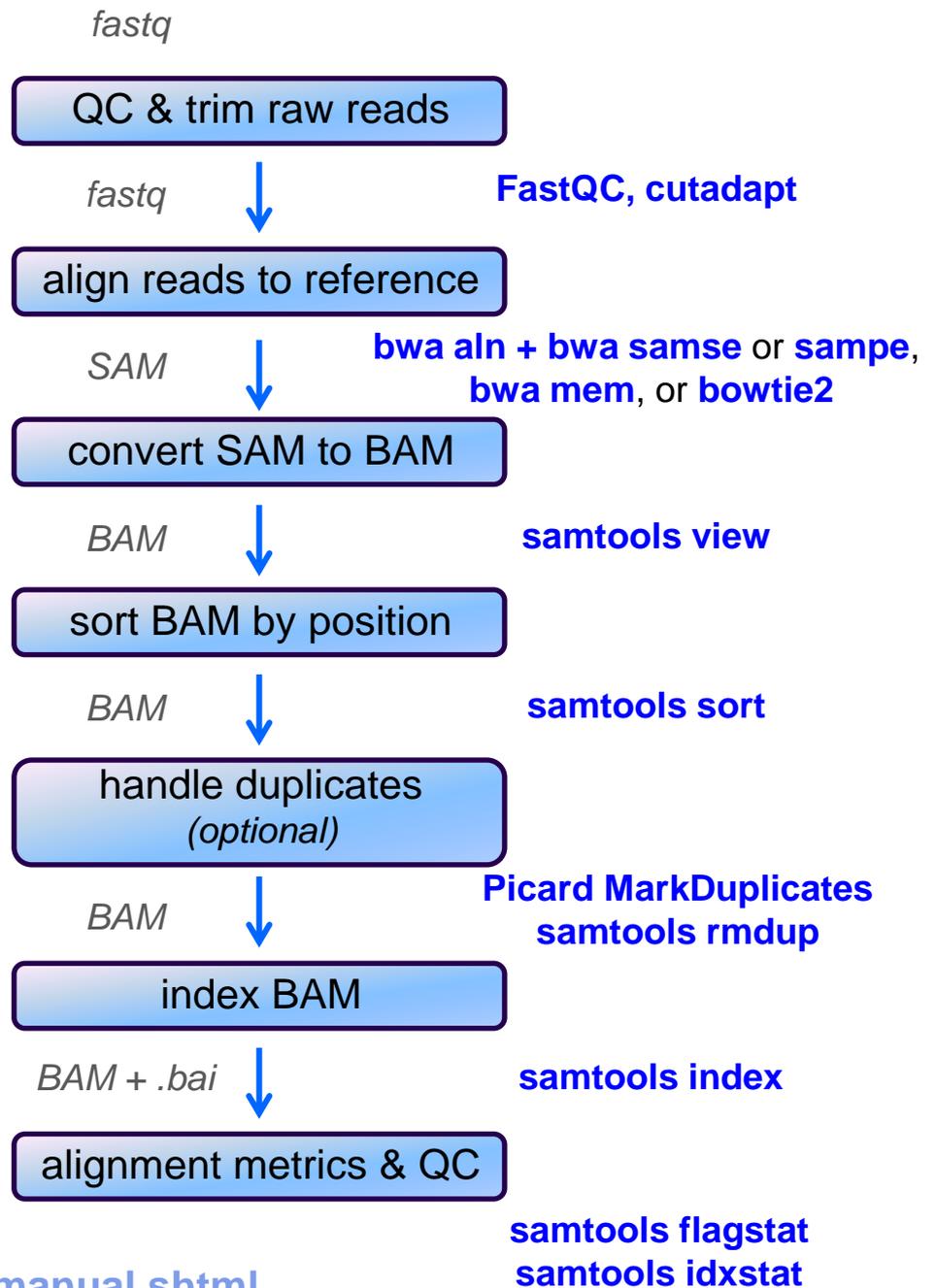
Paired End mapping



- 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 fragment size is reported in the alignment record
 - for both proper and discordant pairs



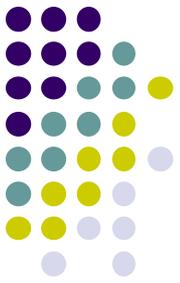
Alignment Workflow



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

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

Obtaining/building a reference



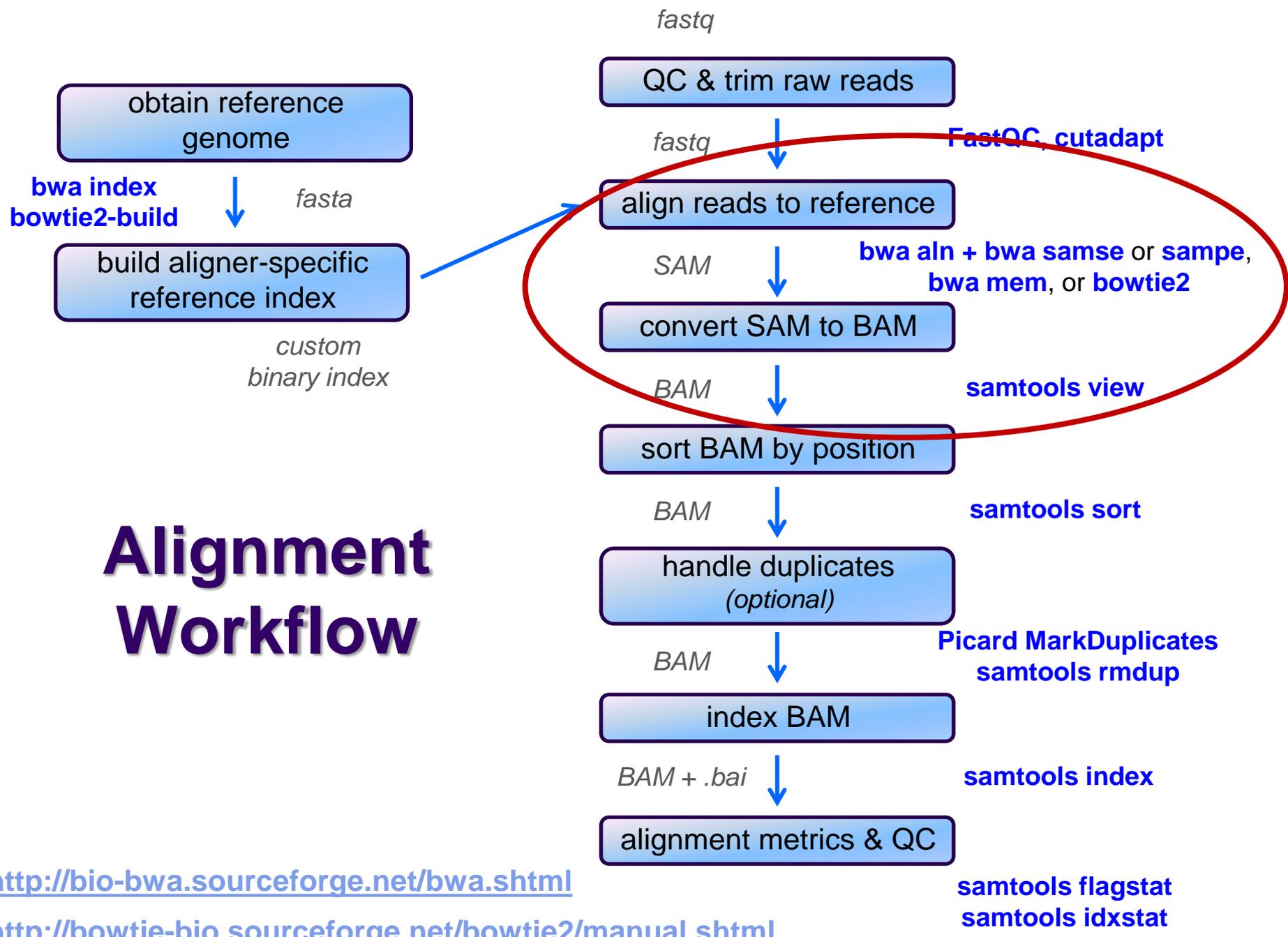
- What is a reference?
 - assembled genomes or transcriptomes
 - Ensembl, UCSC, for eukaryotes
 - NCBI RefSeq or GenBank for prokaryotes/microbes (*prefer RefSeq*)
 - *any set of named DNA sequences*
 - names can be chromosomes, genes, etc. (technically referred to as **contigs**)
- Building a reference index (aligner-specific)
 - may take several hours to build
 - but you build each index once, use for multiple alignments
 - requires FASTA files (**.fa**, **.fasta**) containing DNA sequences
 - annotations (genome feature files, **.gff**) may also be used to build the index, but will definitely be needed for downstream analysis

```
>chrM Mitochondrial Chromosome
GATCACAGGTCTATCACCCCTATTAACCACTCACGGGAGCTCTCCATGCAT
TTGGTATTTTCGTCTGGGGGGTGTGCACGCGATAGCATTGCGAGACGCTG
GAGCCGGAGCACCCCTATGTGCGAGTATCTGTCTTTGATTCTCCTCATT
...
```

sequence name line

- **always** starts with >
- followed by a **name** and other (optional) descriptive information
- one or more line(s) of **sequence characters**
- **never** starts with >

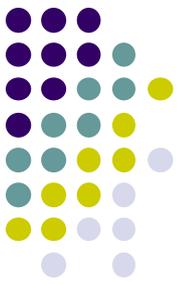
Alignment Workflow



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

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

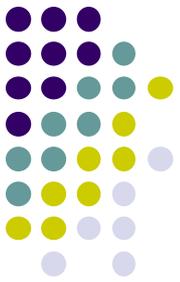
SAM / BAM file format



- 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)
 - and now <https://github.com/samtools/hts-specs/blob/master/SAMtags.pdf>
- 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 sorted + 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)



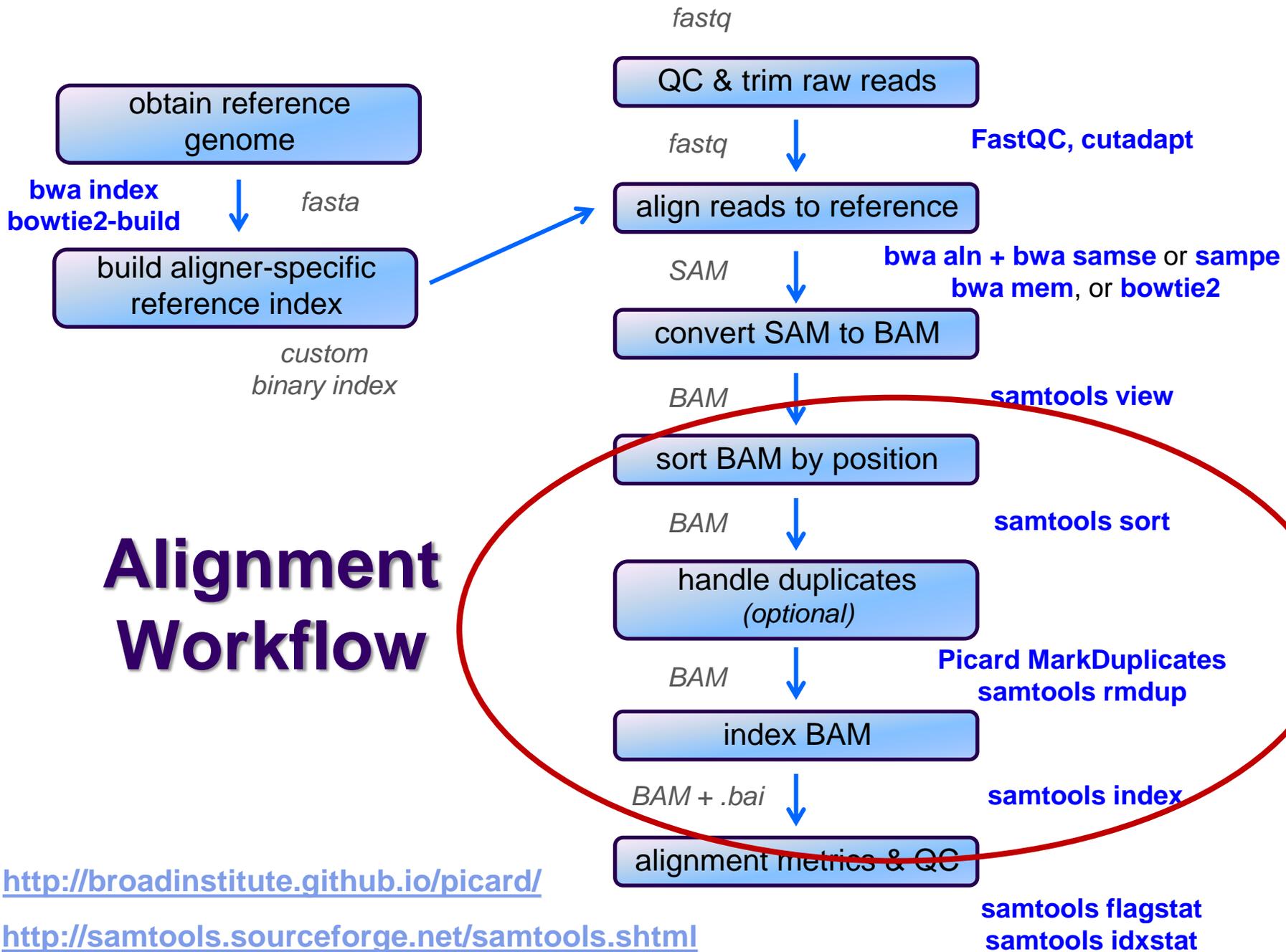
Col	Field	Type	Regexp/Range	Brief description
1	QNAME	String	[!-?A-~]{1,255}	Query template NAME <i>read name from fastq</i>
2	FLAG	Int	[0,2 ¹⁶ -1]	bitwise <u>FLAGs</u>
3	RNAME	String	* [!-()+-<>-~] [!-~]*	Reference sequence NAME <i>contig + start</i>
4	POS	Int	[0,2 ²⁹ -1]	<u>1-based leftmost mapping 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

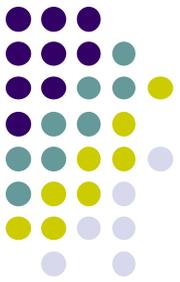
SRR030257.264529
99
NC_012967
1521
29
34M2S
 = 1564 79
positive
for plus
strand
reads

CTGGCCATTATCTCGGTGGTAGGACATGGCATGCC
 AAAAAA;AA;AAAAA?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

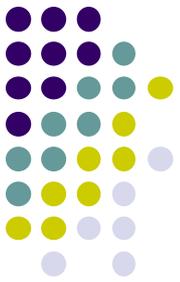
CTGGCCATTATCTCGGTGGTAGGTGATGGIATGCGC
 <<9:<<AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
 XT:A:U NM:i:0 SM:i:37 AM:i:37 X0:i:1 X1:i:0 XM:i:0 XO:i:0 XG:i:0 MD:Z:36





Sorting / indexing BAM files

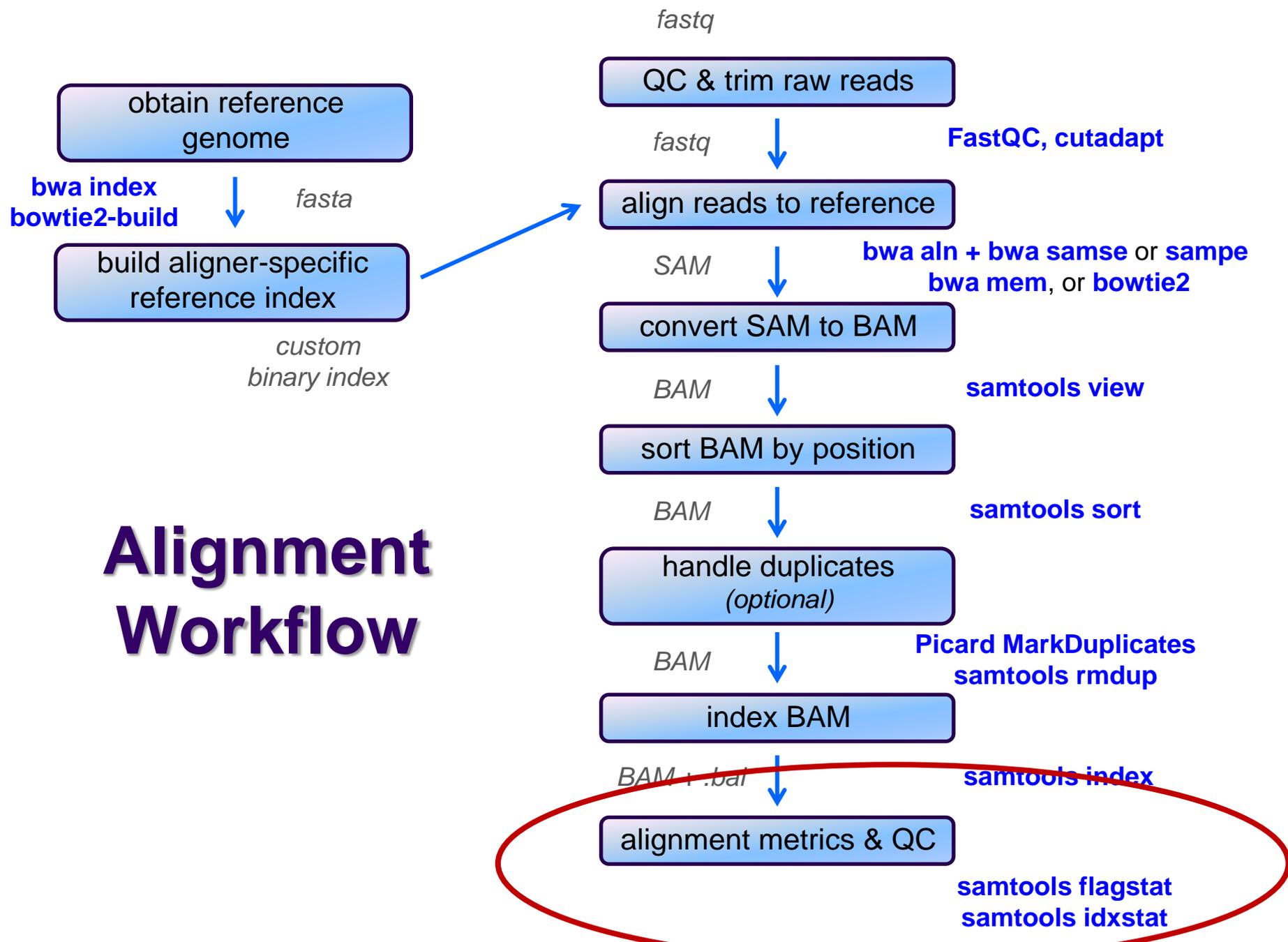
- SAM created by aligner contains read records in *name order*
 - same order as read names in the input FASTQ file
 - R1, R2 have adjacent SAM records
 - SAM → BAM conversion does not change the name-sorted order
- Sorting BAM puts records in *position (locus) order*
 - by contig name then start position (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 and BRCF “pods”
 - **[align_bowtie2_illumina.sh](#)**, **[align_bwa_illumina.sh](#)**, **[trim_adapters.sh](#)**, etc.
 - TACC: shared project directory **[/work/projects/BioITeam/common/script/](#)**
 - BRCF pods: read-only mount **[/mnt/bioi/](#)**

Other NGS Resources at UT



- CBRS training courses
 - Intro to NGS, RNAseq, many others
<https://research.utexas.edu/cbrs/cores/cbb/educational-resources/>
 - “Summer School” (4 or 5 half-day sessions in June)
 - lots of hands-on, including w/TACC
 - Short courses (3-4 hour workshops)
- Genome Sequencing & Analysis Facility (GSAF)
 - Jessica Podnar, Director, gsaf@utgsaf.org
- Bioinformatics consultants
 - Dennis Wylie, Dhivya Arasappan, Benni Goetz, Anna
 - BioTeam wiki – <https://wikis.utexas.edu/display/bioiteam/>
- Biomedical Research Support Facility (BRCF)
 - provides local compute and managed storage resources
<https://wikis.utexas.edu/display/RCTFUsers>