Next Generation Sequencing Technology Fundamentals & Applications

2013-06-18 Boot Camp

Joe Fass | Lead Data Analyst

UC Davis Genome Center Bioinformatics Core

General Comments

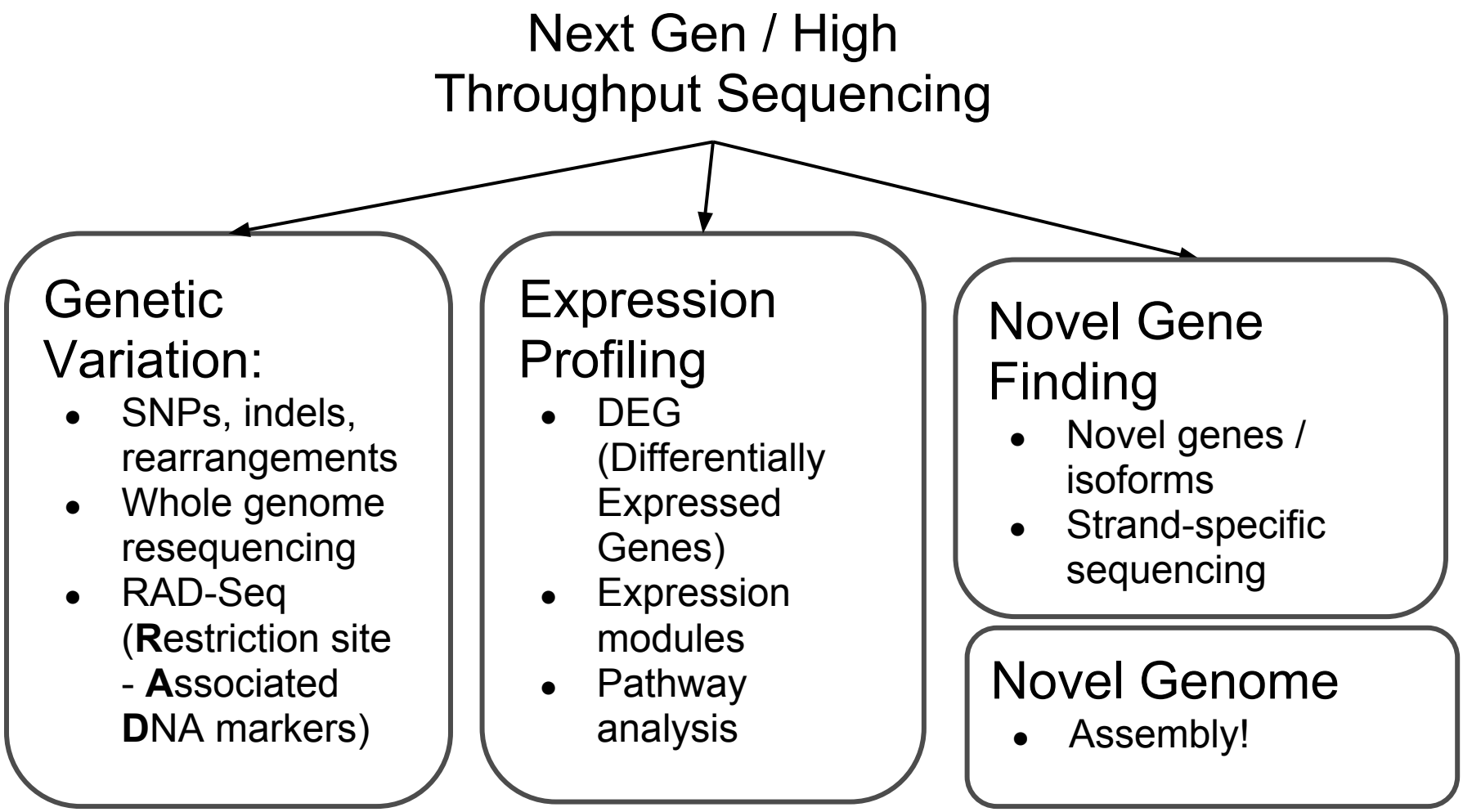
- Format / schedule
- Breaks / lunch
- Feedback form!
- Slides (see training site)

General Outline

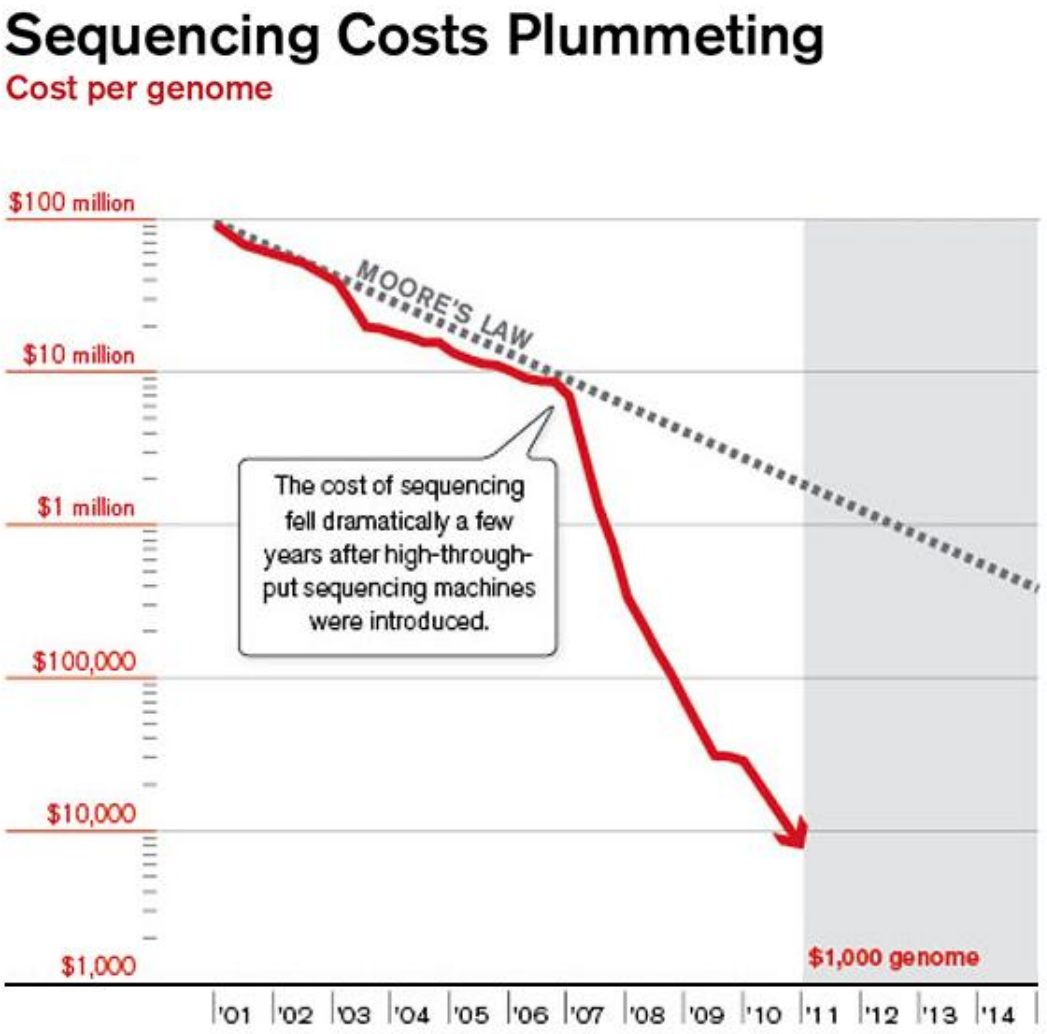
- General (sequencing) comments
- Sequencing technologies
- Errors, and fixing them (grooming / QC)
- Applications
  - Alignment
    - for Variant Discovery
    - for RNA-Seq
  - Assembly
- IT, Cost considerations
- Q&A

General comments on sequencing

Why sequence? A sampling ...



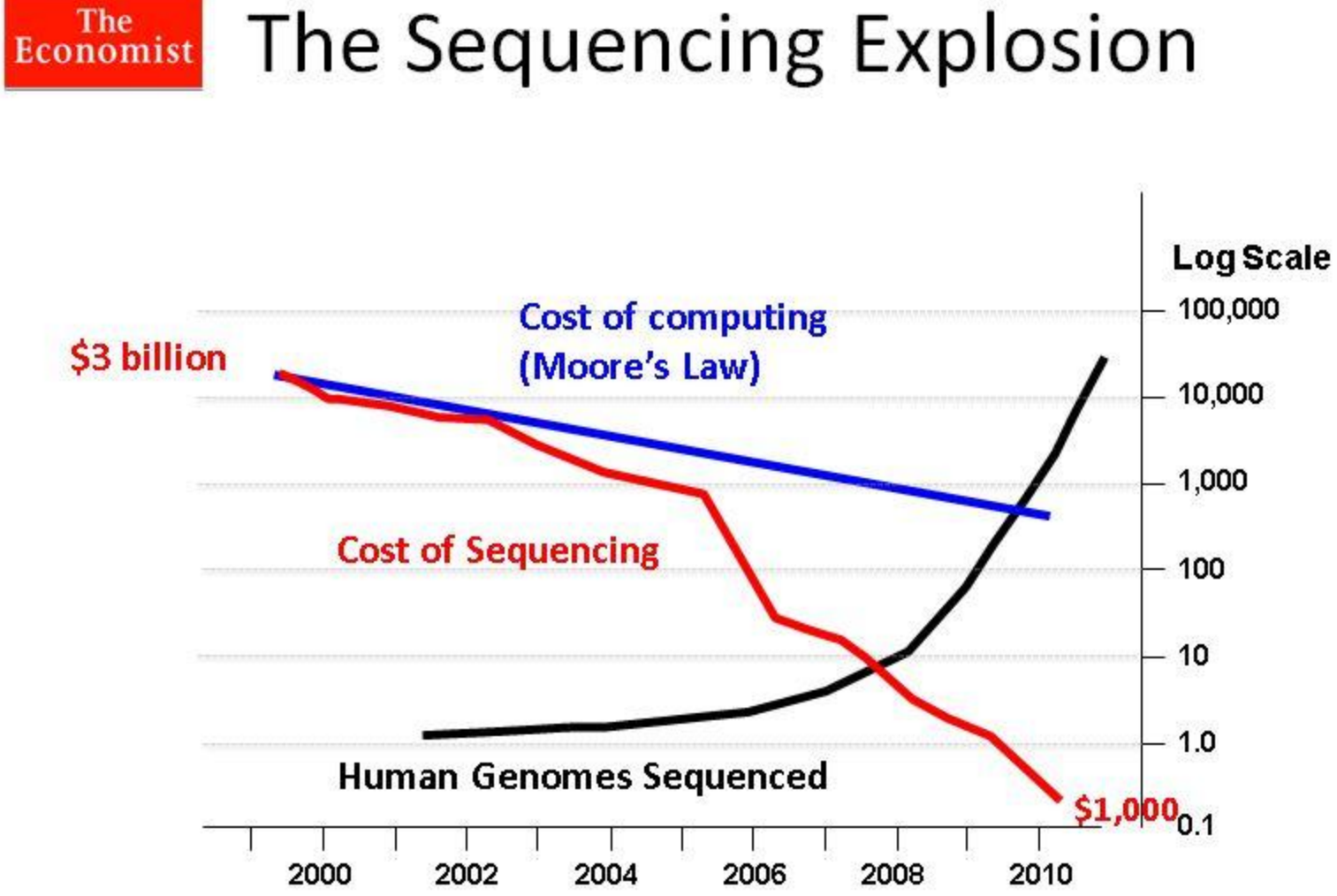
Sequencing Explosion



http://www.technologyreview.com/graphiti/427720/bases-to-bytes/

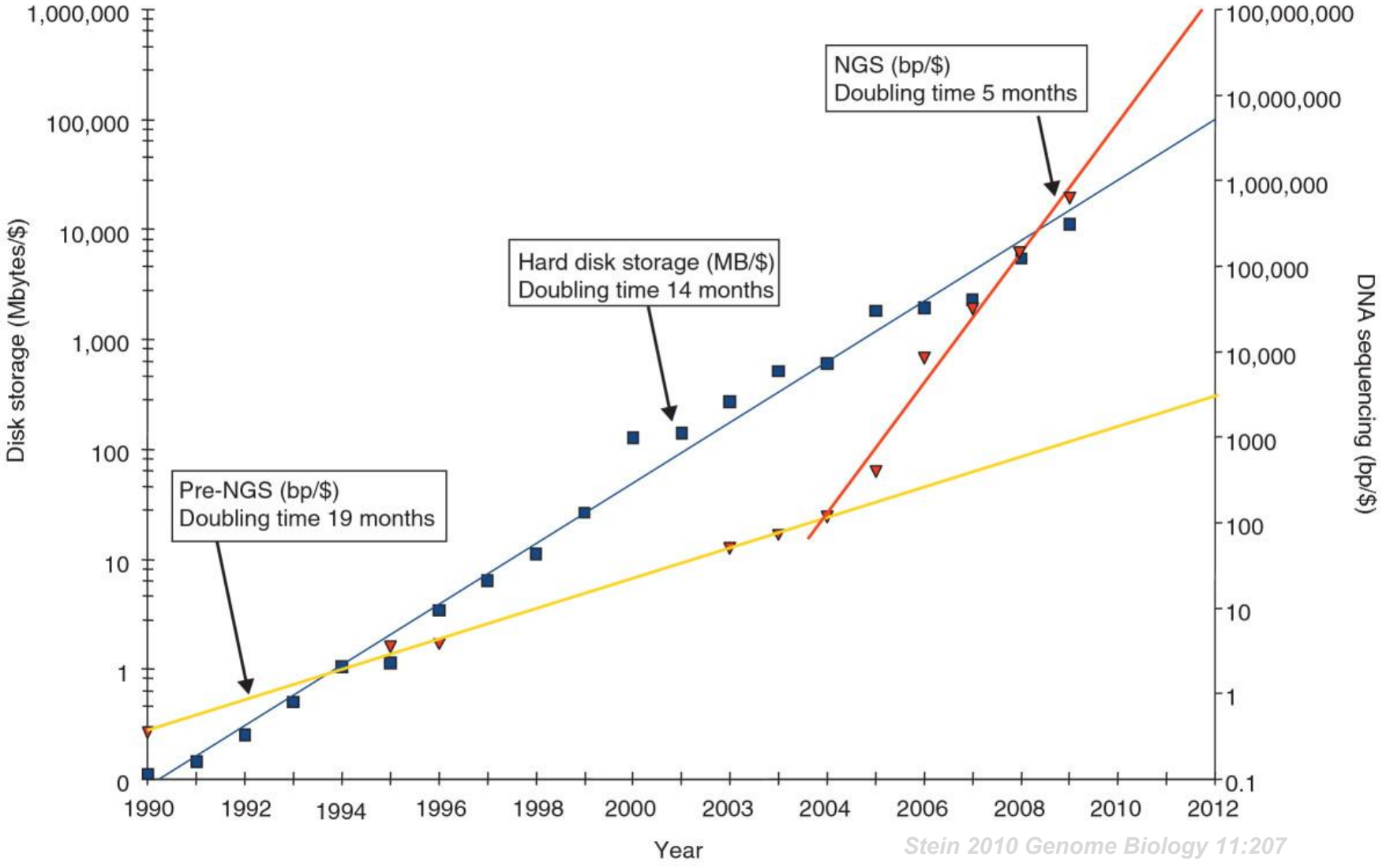
Sequencing Explosion

Adapted from The Economist



http://chrissemsarian.wordpress.com/2012/06/

Sequencing Explosion



Stein 2010 Genome Biology 11:207

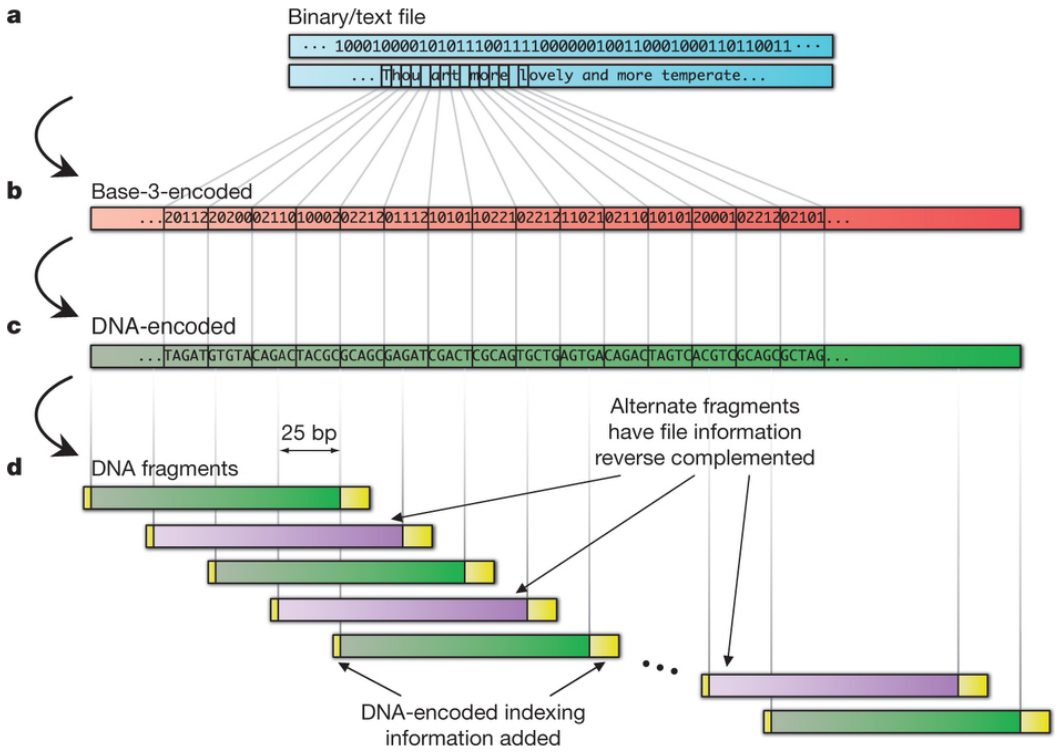
Sequencing Explosion

Storing data in DNA, as opposed to hard drives ...

Towards practical, high-capacity, low-maintenance information storage in synthesized DNA

Nick Goldman, Paul Bertone, Siyuan Chen, Christophe Dessimoz, Emily M. LeProust, Botond Sipos, & Ewan Birney

Nature (2013) doi:10.1038/nature11875

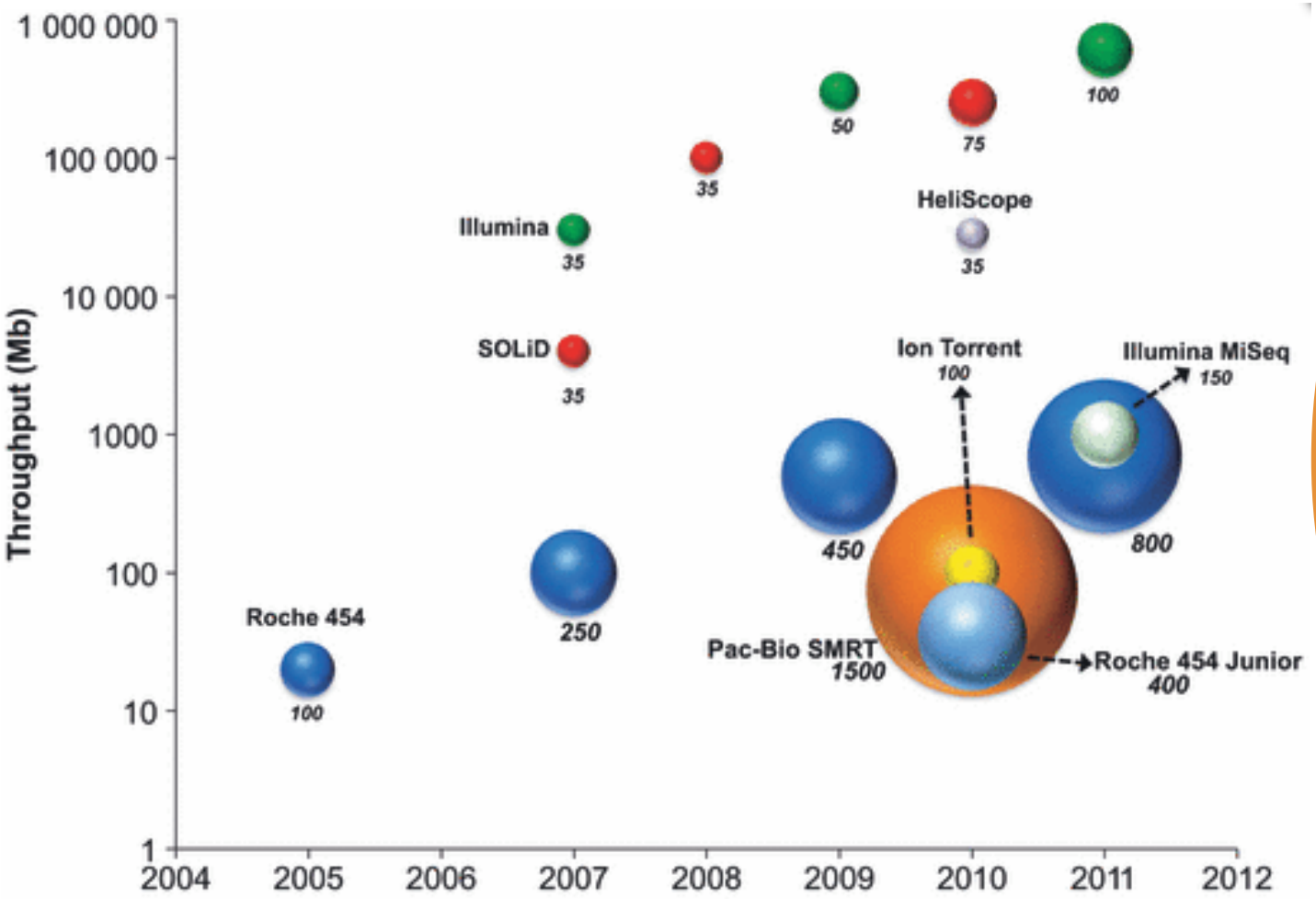


Sequencing Explosion

adapted from Mardis 2011 Nature 470:198

Sequencing Explosion

... Oxford Nanopore?



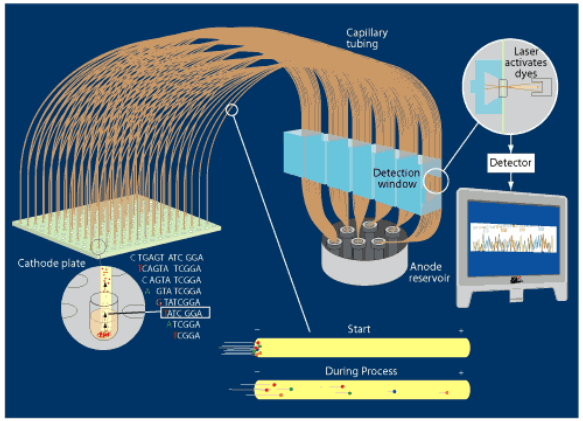
PacBio RS II

2013

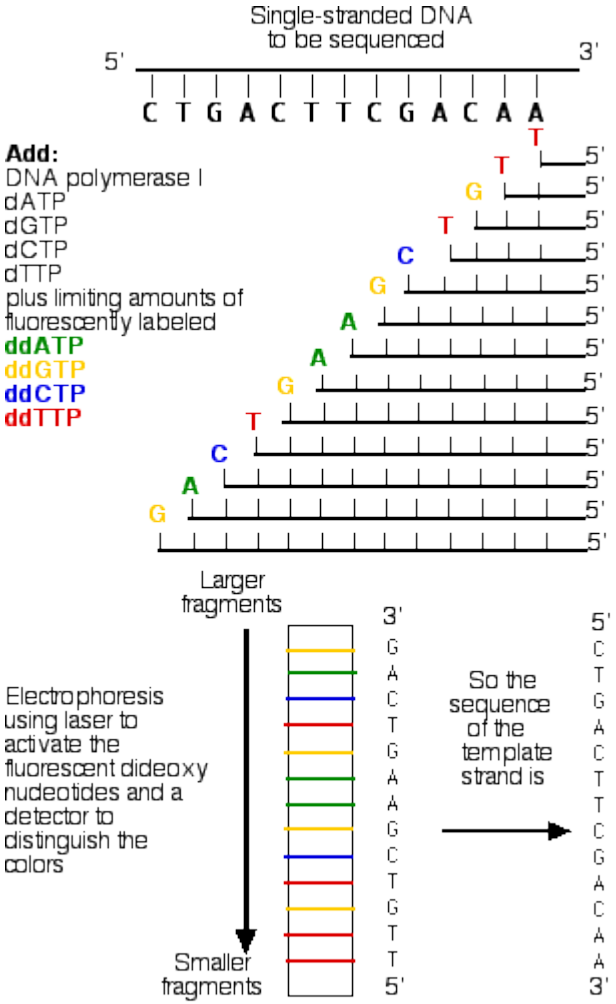
adapted from Shokralla 2012 Molecular Ecology 21:1794

Sanger Sequencing

- ddNTP's (with fluorescent labels) incorporated (along with unlabeled dNTP's) in amplification step, resulting in some molecules terminated *at every position*
- Gel / capillary electrophoresis orders molecules by length
- Fluorescent label (color) indicates terminal base identity at each position
- Read colors, in order, to derive sequence



http://www.jgi.doe.gov/sequencing/education/how/how_10.html



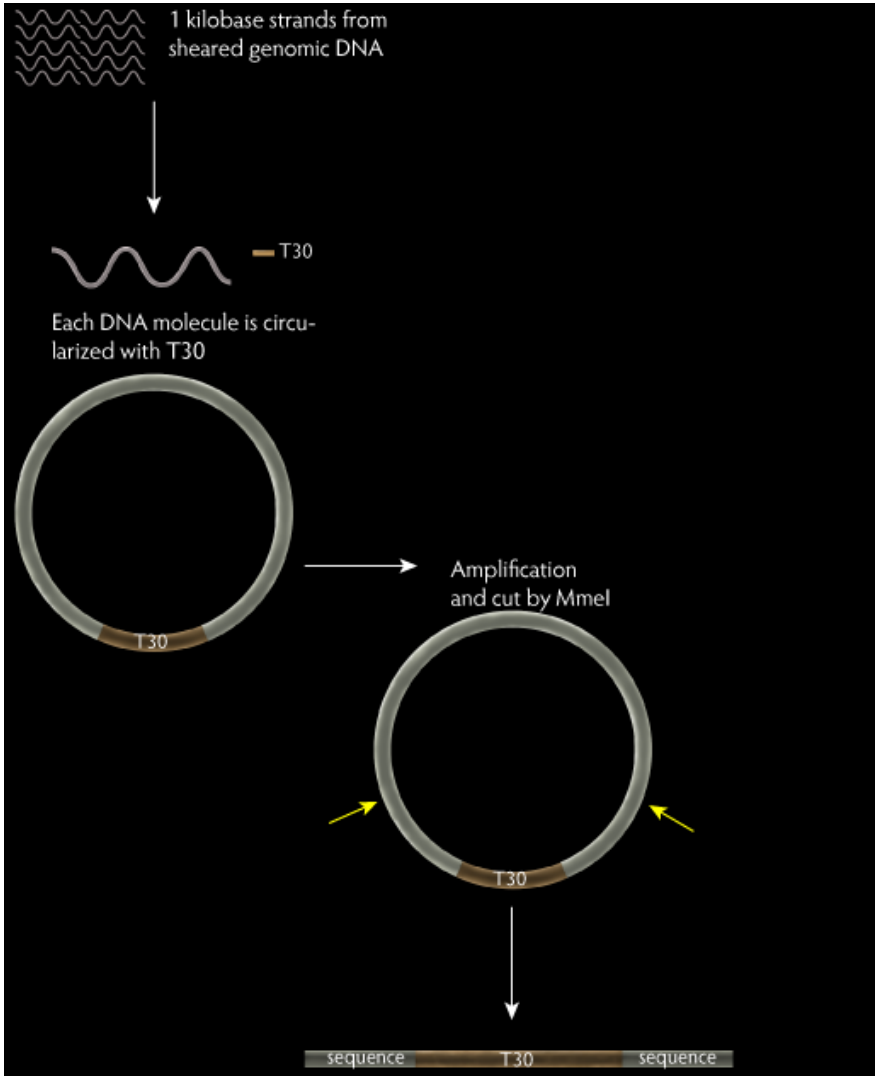
http://users.rcn.com/jkimball.ma.ultranet/BiologyPages/D/DNAsequencing.html

Polony Sequencing

Developed by **George Church** (helped initiate *Human Genome Project*, initiated *Personal Genome Project*) and used for the PGP.

Polonator sequencer / software is *open hardware* / *open source*!

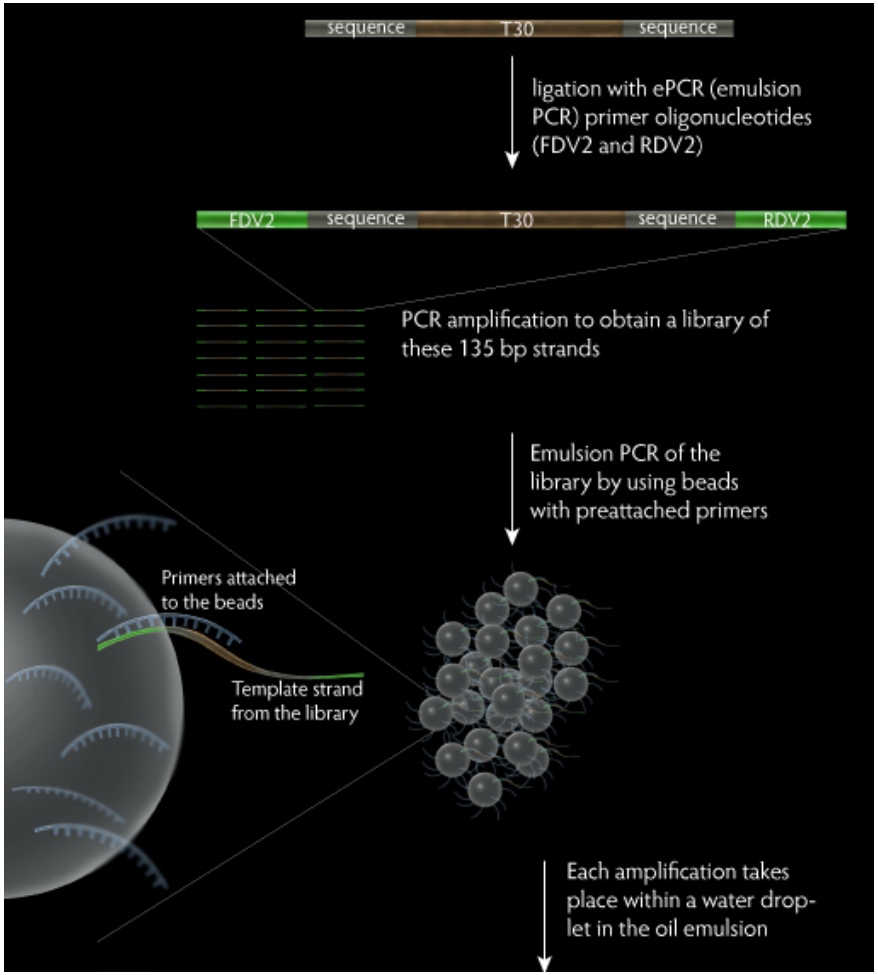
Paired read library generated from sheared DNA, circularized around known sequence, cut by restriction enzyme.



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

Polony Sequencing

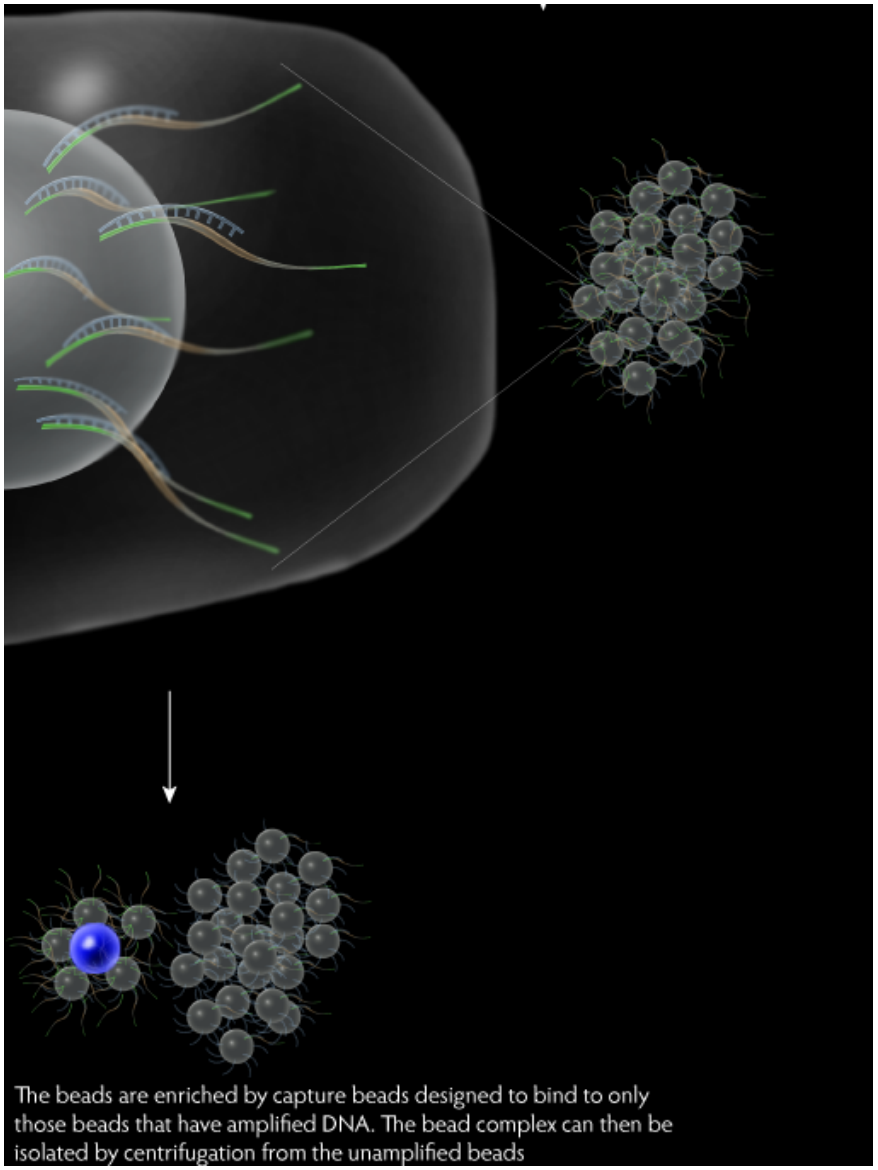
Emulsion PCR to amplify library fragments on beads. Goal is to have identical molecules on any particular bead.



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

Polony Sequencing

Capture beads bind only those "sequencing" beads that are occupied by amplified DNA.



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

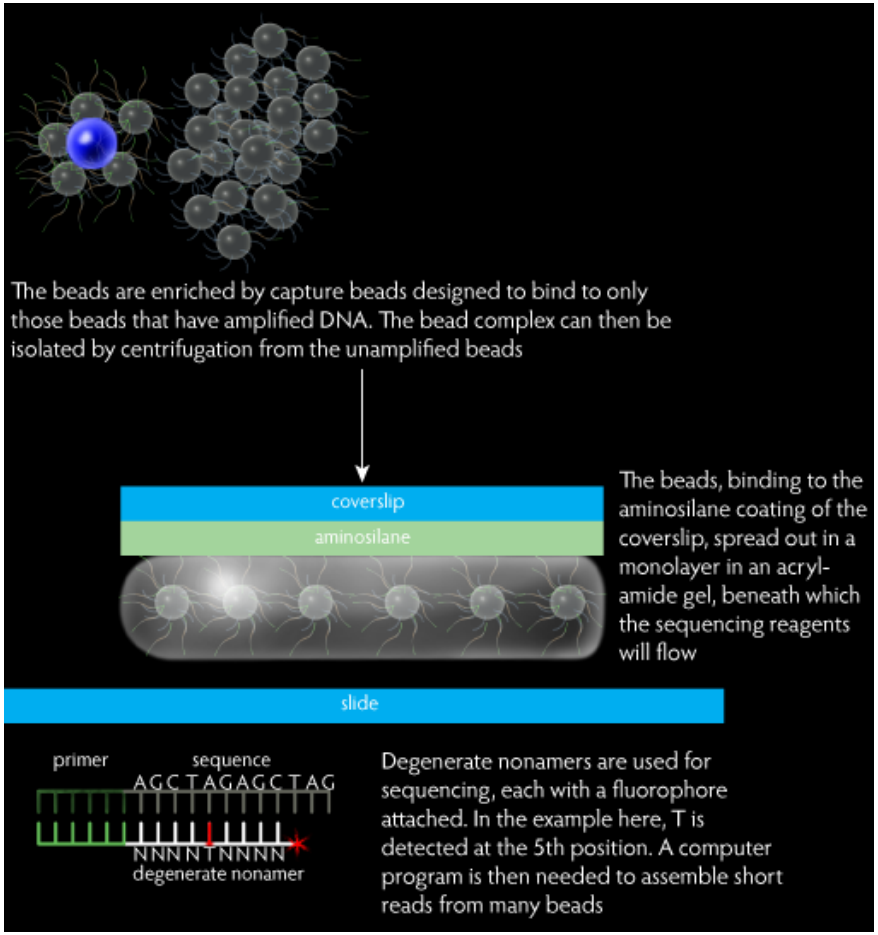
Polony Sequencing

Beads then adhered to coverslip, which becomes the top of a *flow cell*, so that fluorescently-labeled nonamers can be ligated at sequentially shifted spacing with respect to the primer sequence.

Paired reads of 13 bp each (each is a 7 bp read, then a gap of 4 or 5 bp, then a 6 bp read):

`ACGTTGANNNNTGCCAT` *(forward)*
`TTAGCATNNNNAAGTAA` *(reverse)*

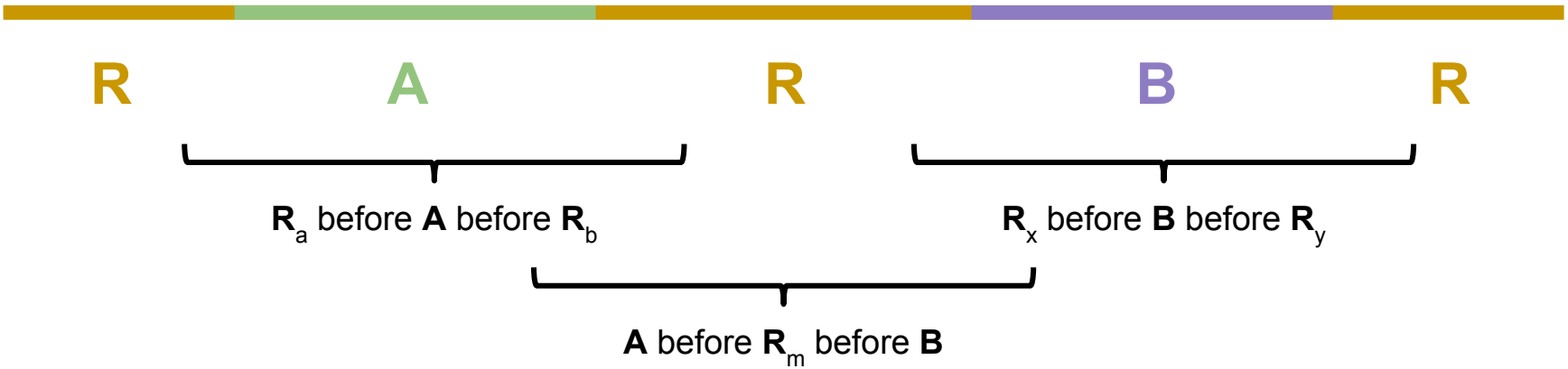
Requires very specific software for read assembly / alignment.

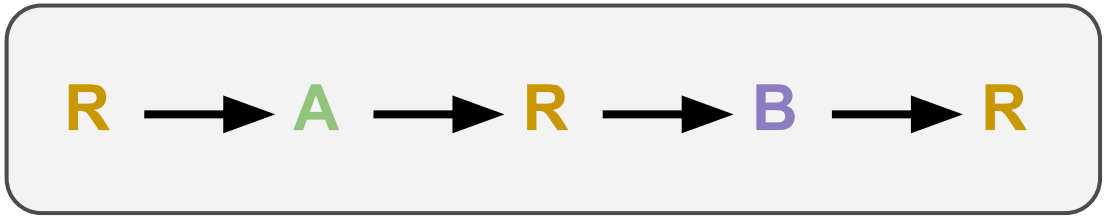


http://en.wikipedia.org/wiki/Polony_sequencing

General Strategies - Long Reads

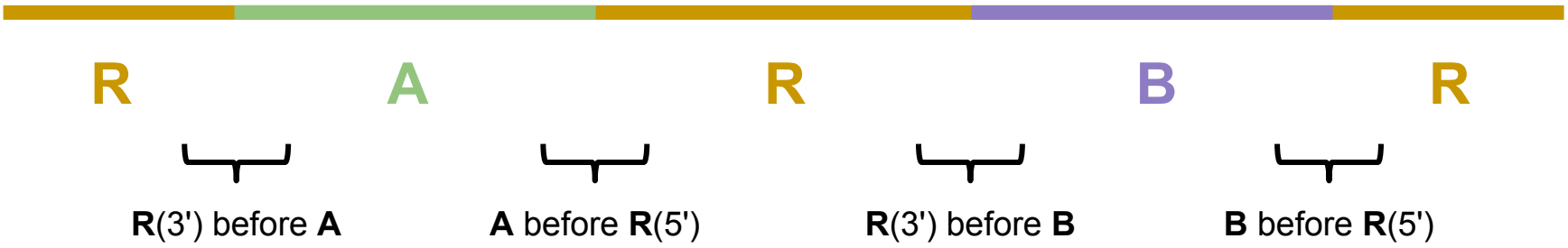
Long reads contain *long range, continuous* information:

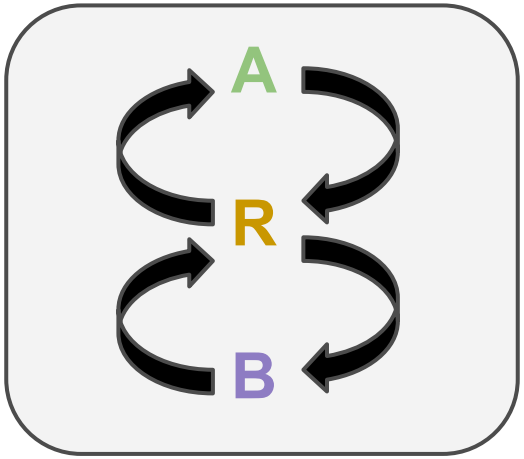




General Strategies - Short Reads

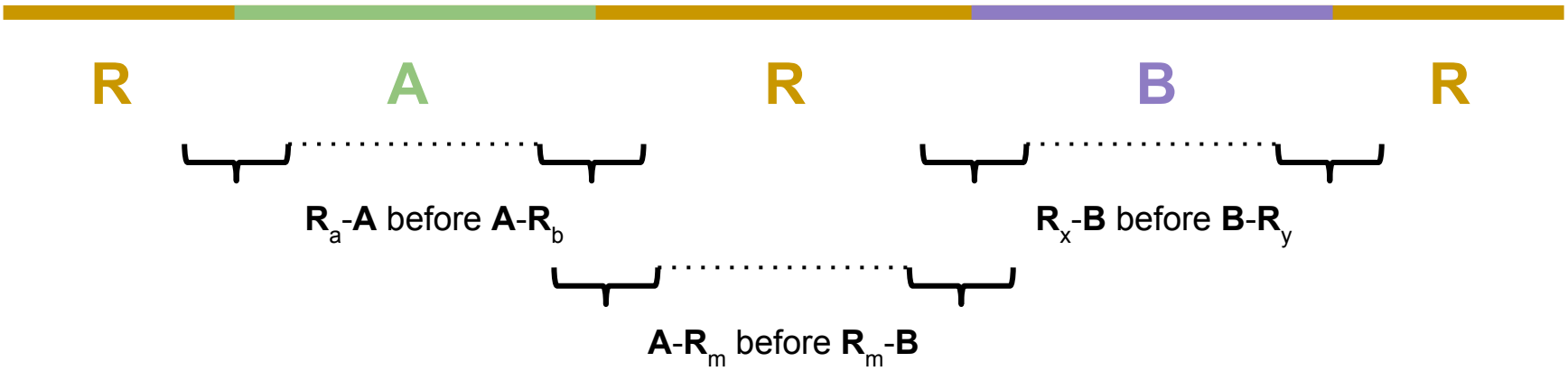
Shorter reads lack *long range* information:

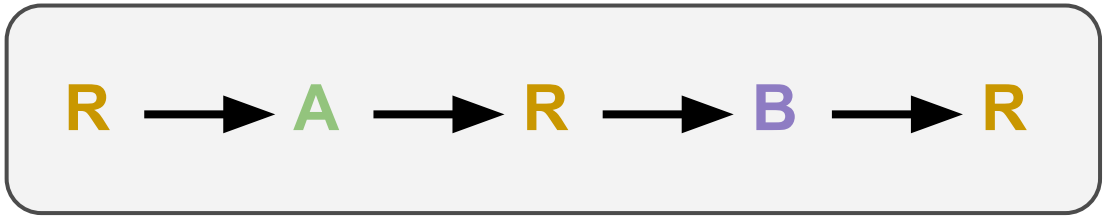




General Strategies - Paired Reads

Paired short reads re-capture *long range* information:





Sequencing Technologies

Current Sequencing Technologies

- (Roche) 454
- Illumina
- *SOLiD*
- PacBio
- Ion Torrent
- *Complete Genomics?*
- *(Illumina) Moleculo*
- *Oxford Nanopore*

Current Sequencing Technologies

Roche **454** GS FLX Titanium

Illumina MiSeq

Illumina HiSeq 2000 / 2500

Current Sequencing Technologies

PacBio RS

Ion Torrent
(Life Technologies)
Ion Proton

Ion Torrent
(Life Technologies)
Ion PGM

Current Sequencing Technologies

Oxford Nanopore
MinION

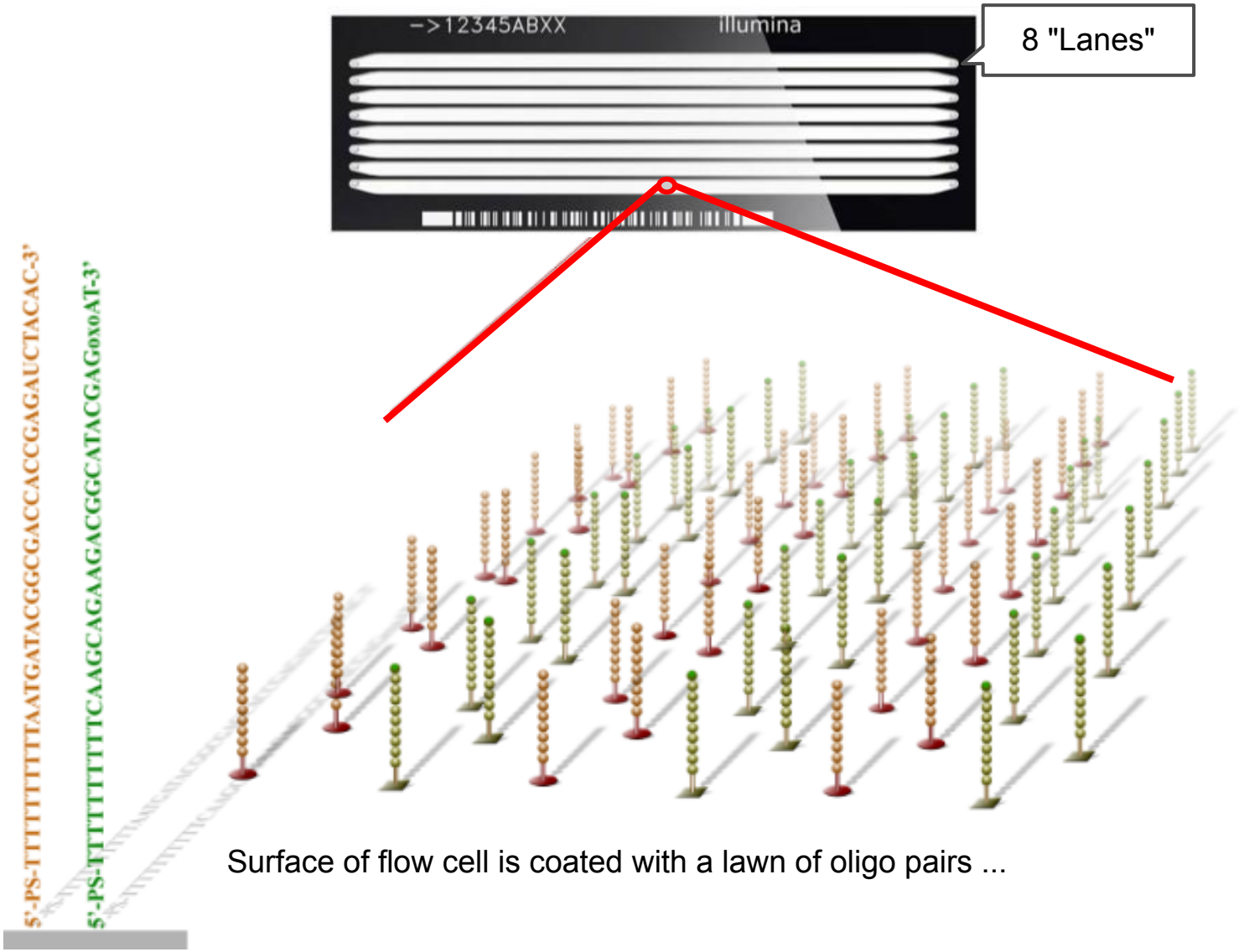
Oxford Nanopore
GridION

Illumina

Illumina MiSeq

Illumina HiSeq 2000 / 2500

Illumina

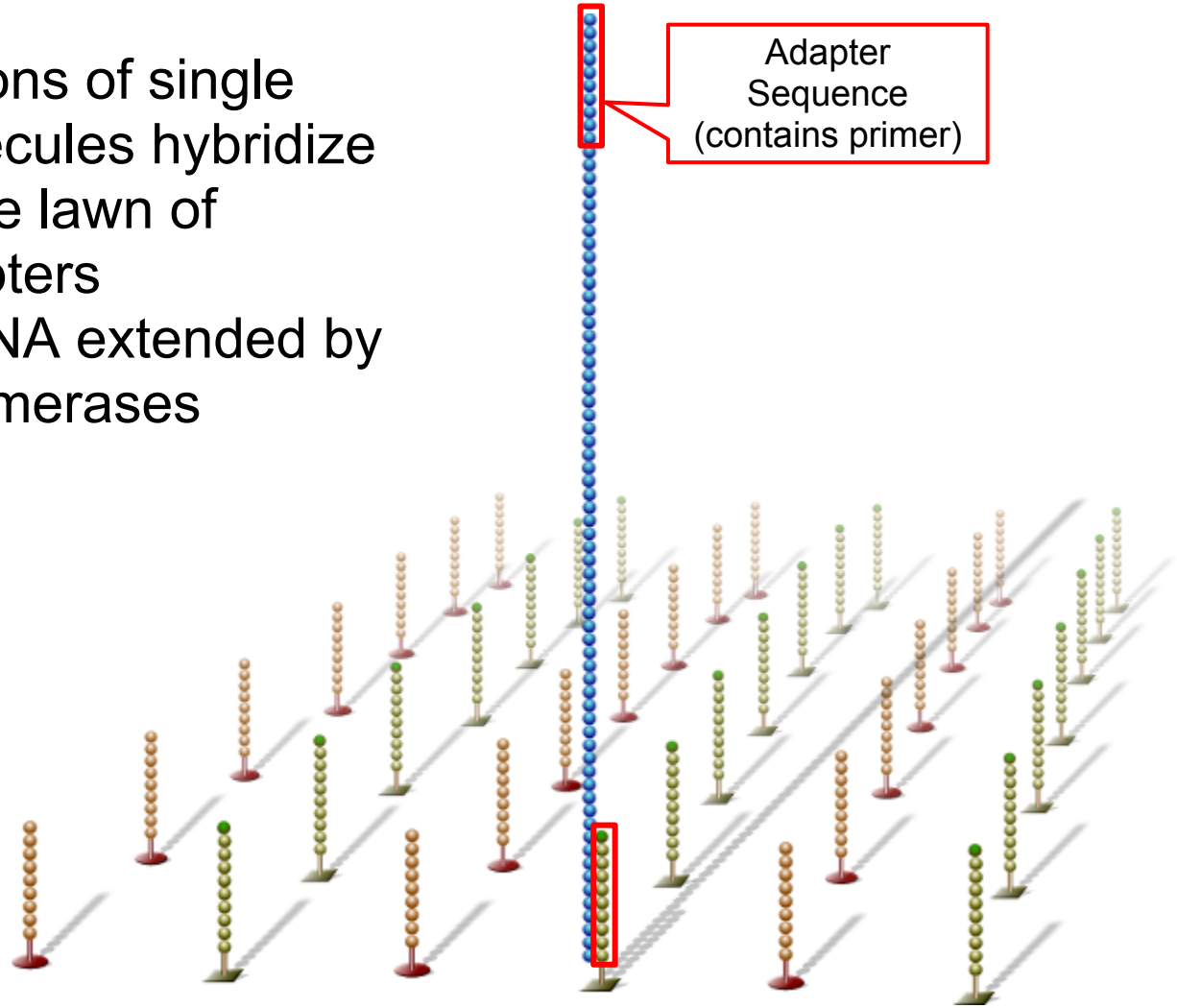


Surface of flow cell is coated with a lawn of oligo pairs ...

Illumina

Cluster Generation:
Hybridize Fragments & Extend

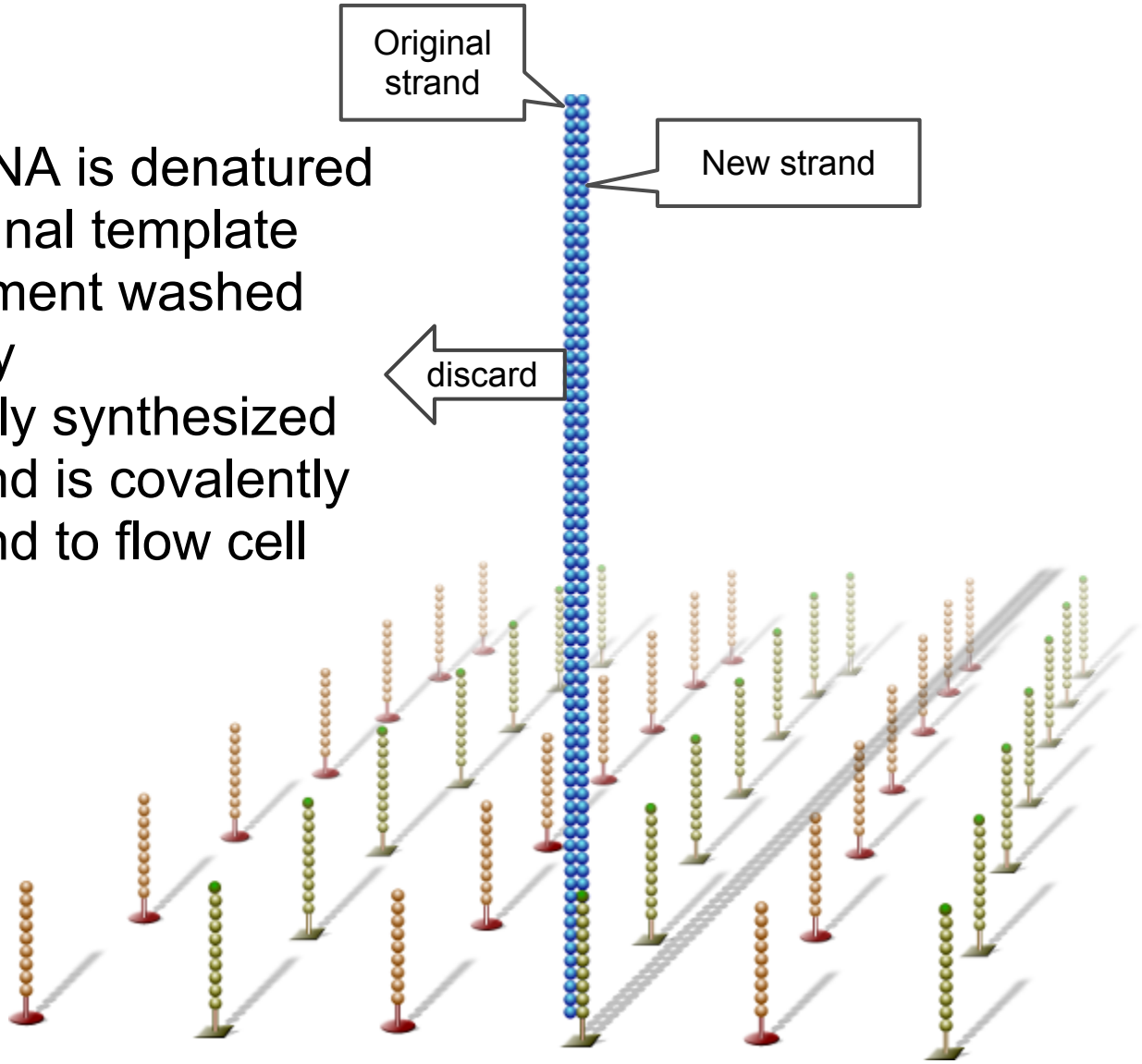
- Millions of single molecules hybridize to the lawn of adapters
- dsDNA extended by polymerases



Illumina

Cluster Generation:
Denature Double-stranded DNA

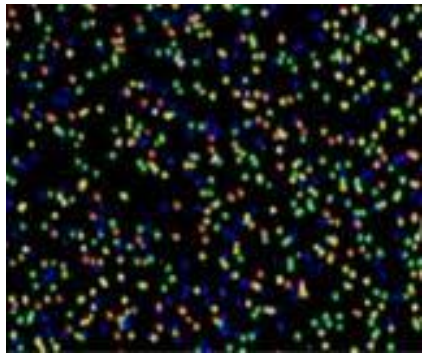
- dsDNA is denatured
- Original template fragment washed away
- Newly synthesized strand is covalently bound to flow cell



Illumina

Cluster Generation:
Covalently-Bound, Randomly Dispersed Single Molecules

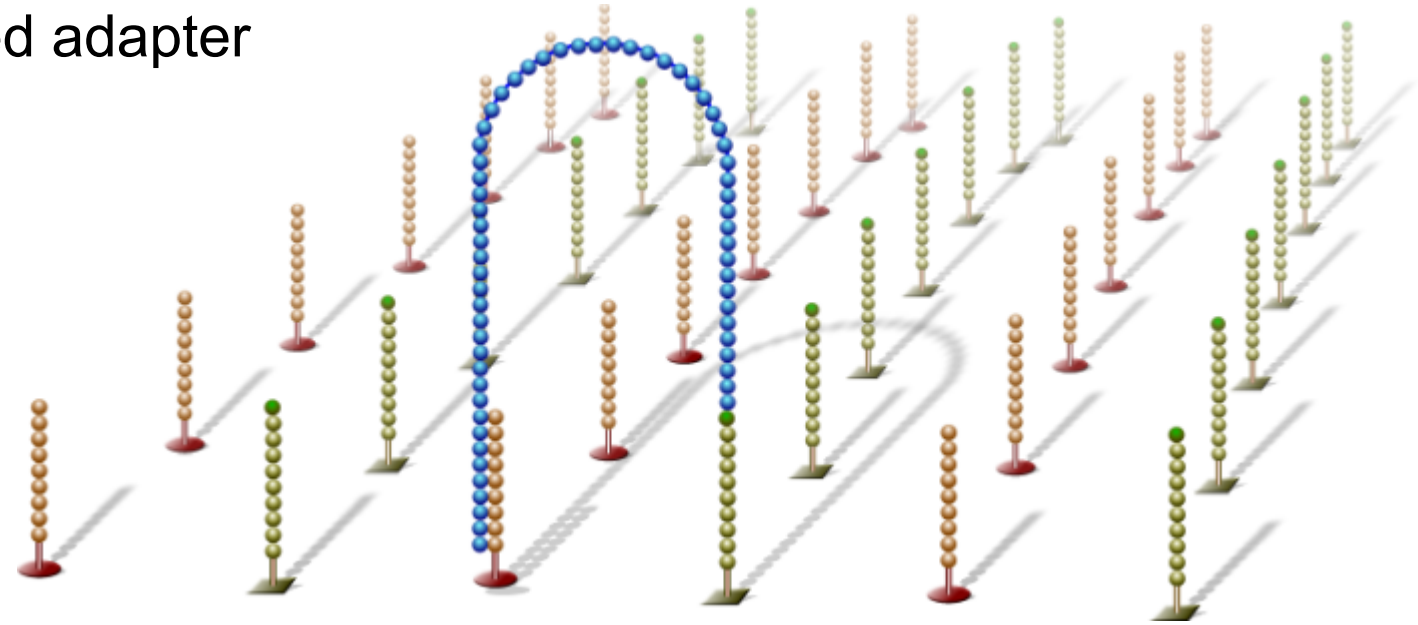
- Resulting covalently-bound DNA fragments are bound to the flow cell surface in a random pattern



Illumina

Cluster Generation:
Bridge Amplification

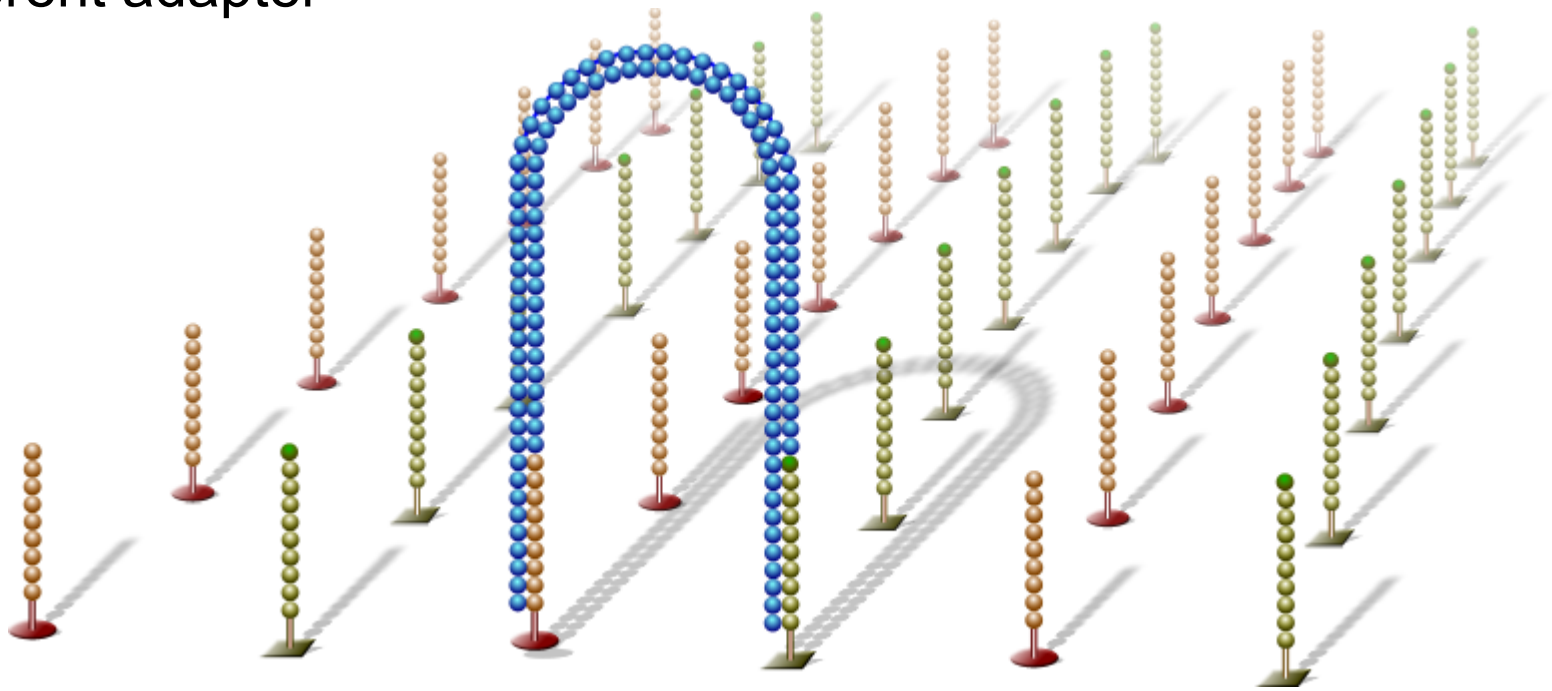
- Single-strand flops over to hybridize to adjacent adapter, forming a bridge
- dsDNA synthesized from primer in hybridized adapter



Illumina

Cluster Generation:
Bridge Amplification

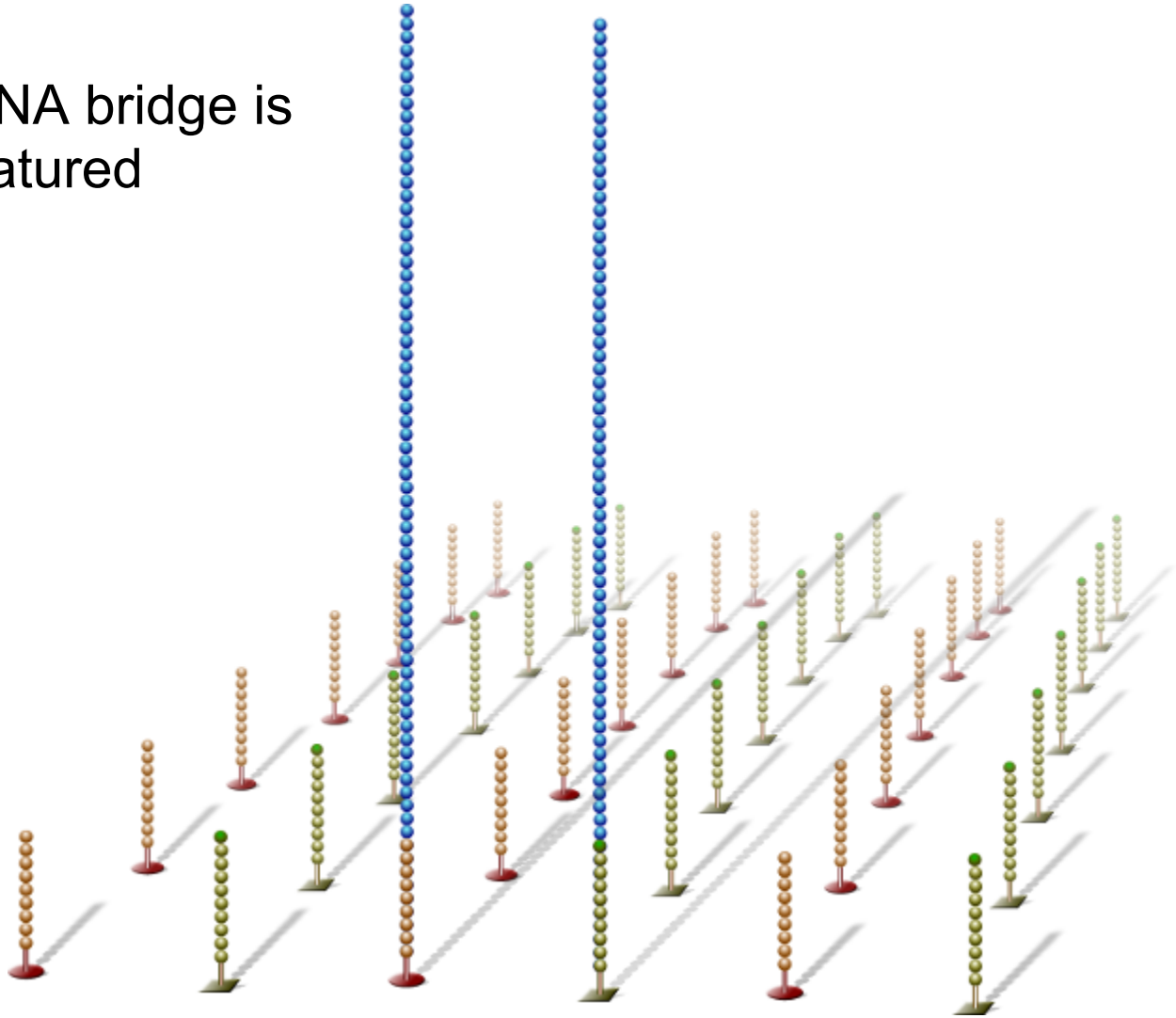
- dsDNA bridge now formed
- each strand covalently bound to different adapter



Illumina

Cluster Generation:
Bridge Amplification

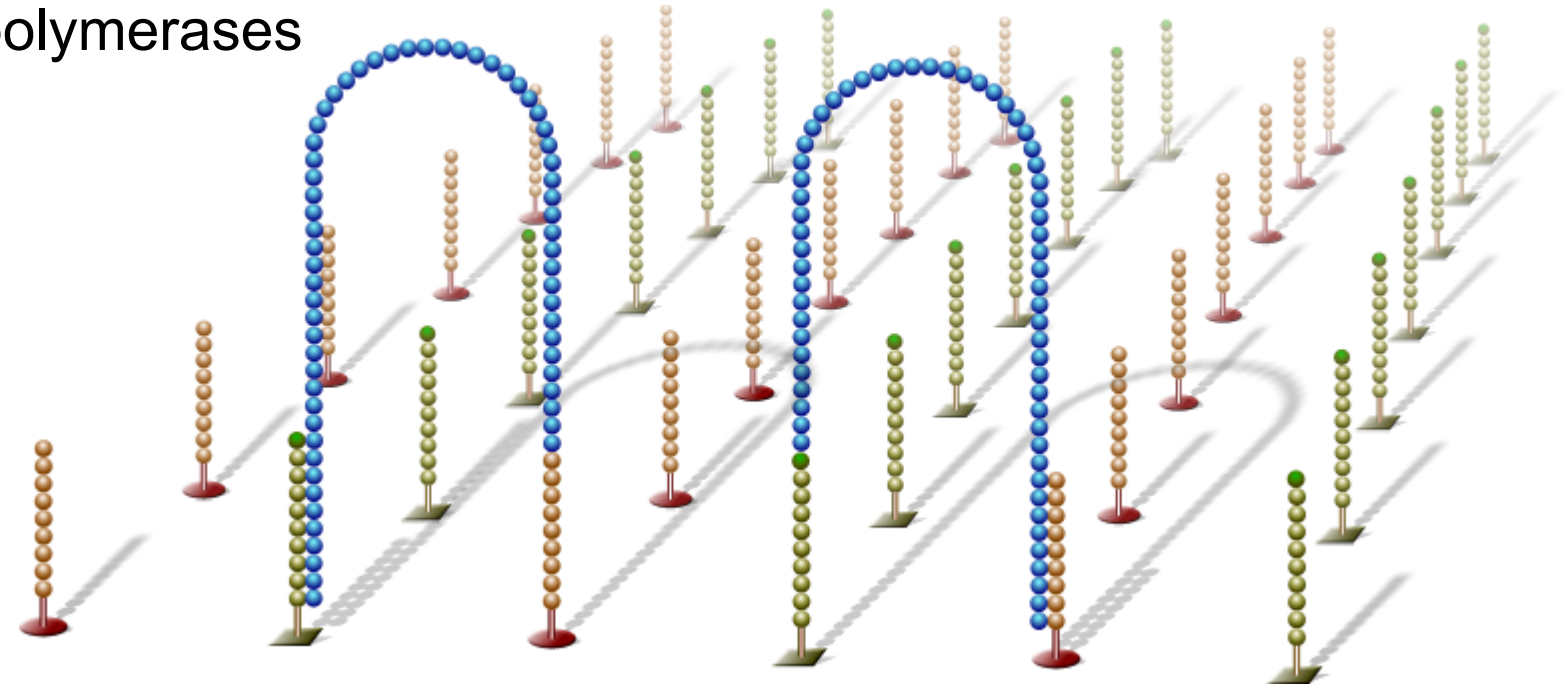
- dsDNA bridge is denatured



Illumina

Cluster Generation:
Bridge Amplification

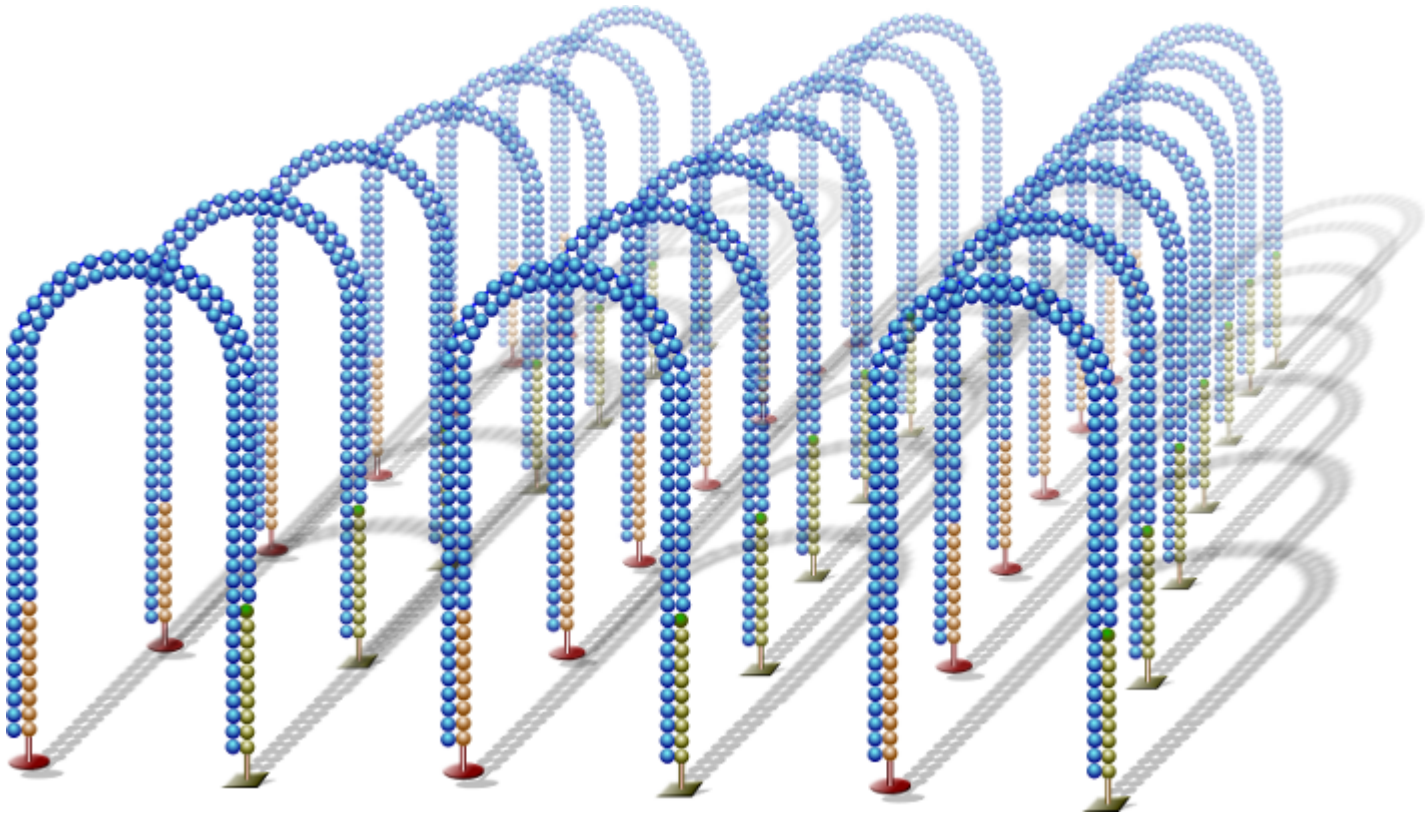
- Single strands flop over to hybridize to adjacent adapters, forming bridges
- dsDNA synthesized by polymerases



Illumina

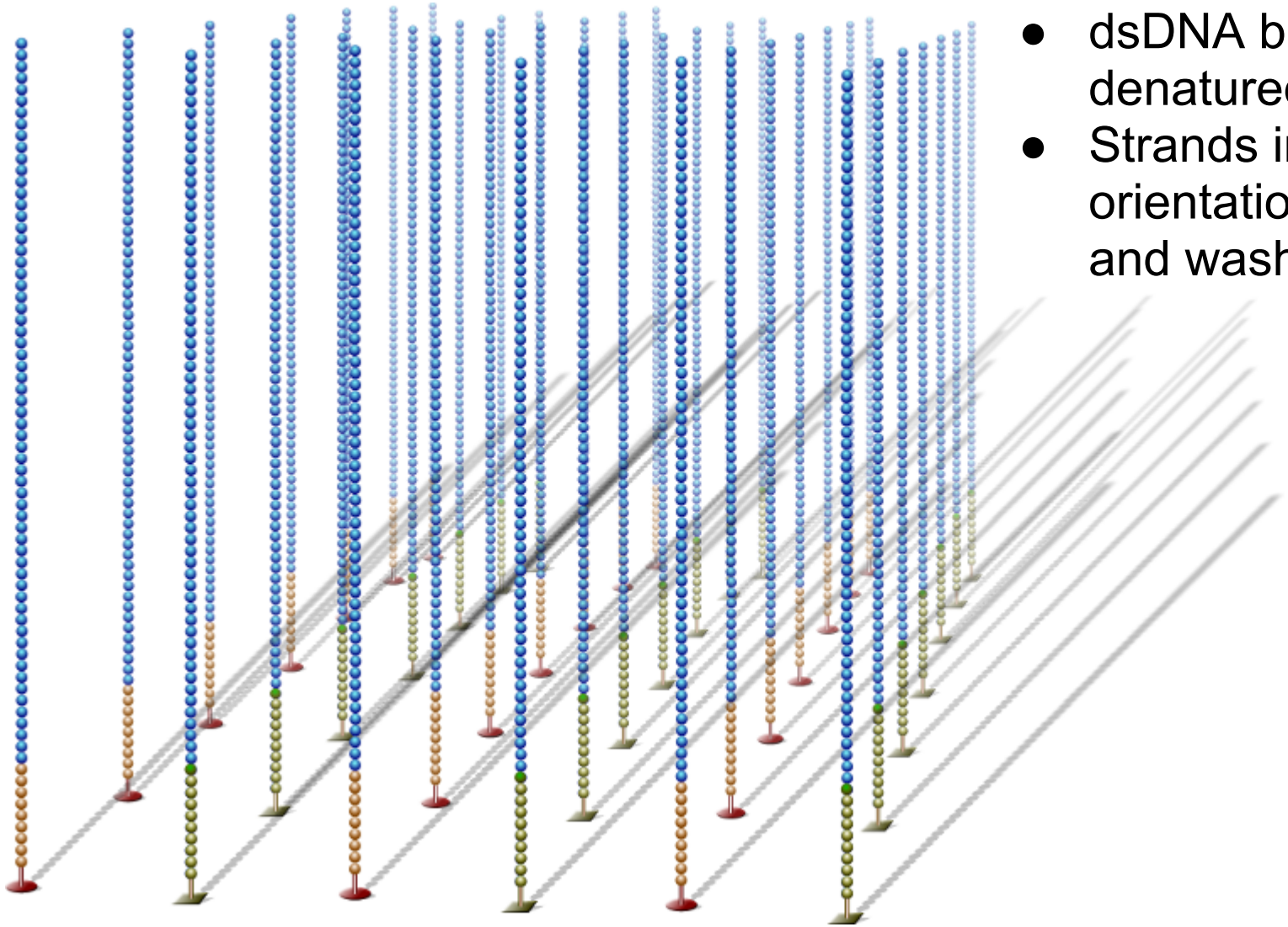
Cluster Generation:
Bridge Amplification

- Bridge amplification cycles repeated many times



Illumina

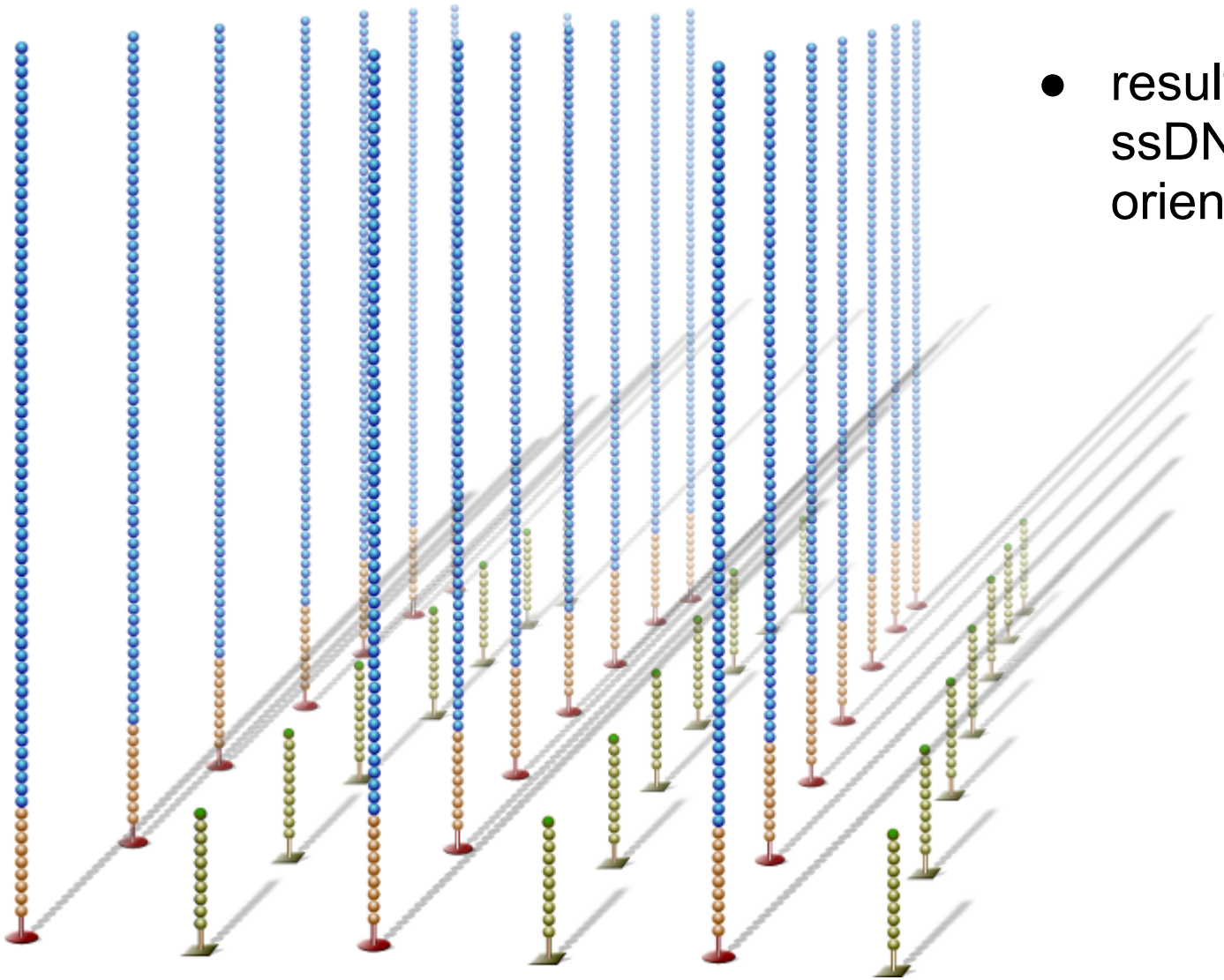
Cluster Generation



- dsDNA bridges denatured
- Strands in one of the orientations cleaved and washed away

Illumina

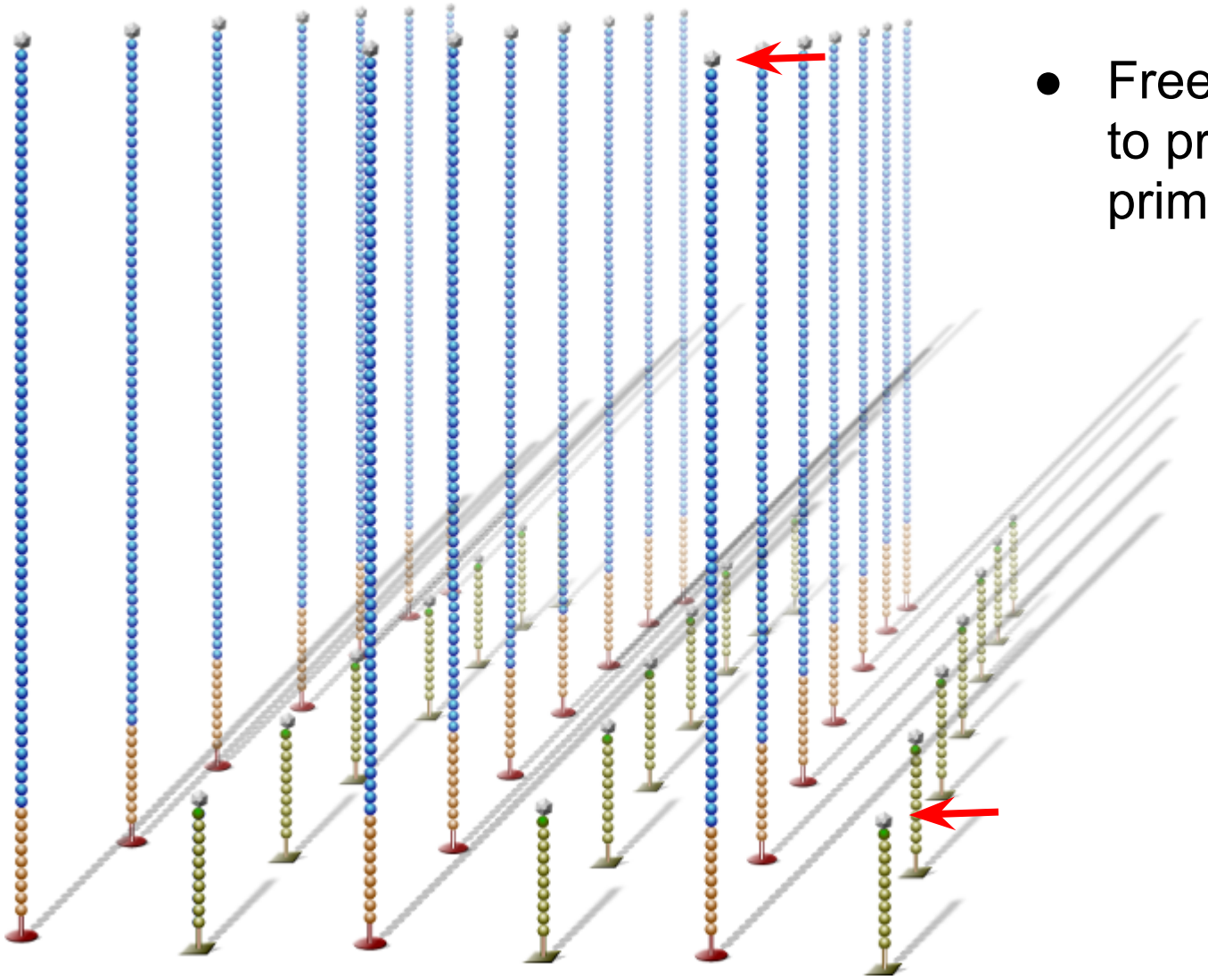
Cluster Generation



- resulting cluster has ssDNA in only one orientation

Illumina

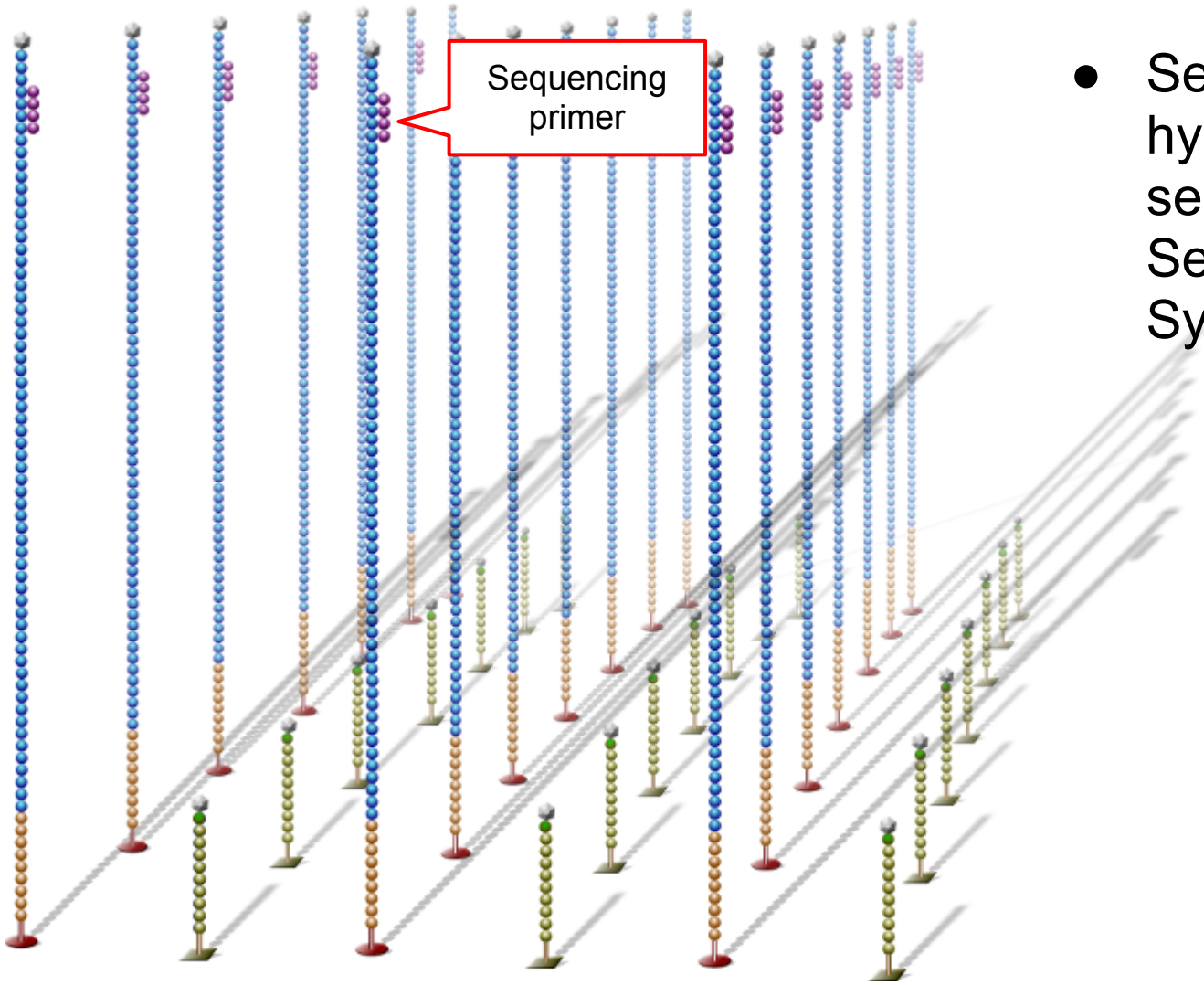
Cluster Generation



- Free 3'-ends blocked to prevent unwanted priming

Illumina

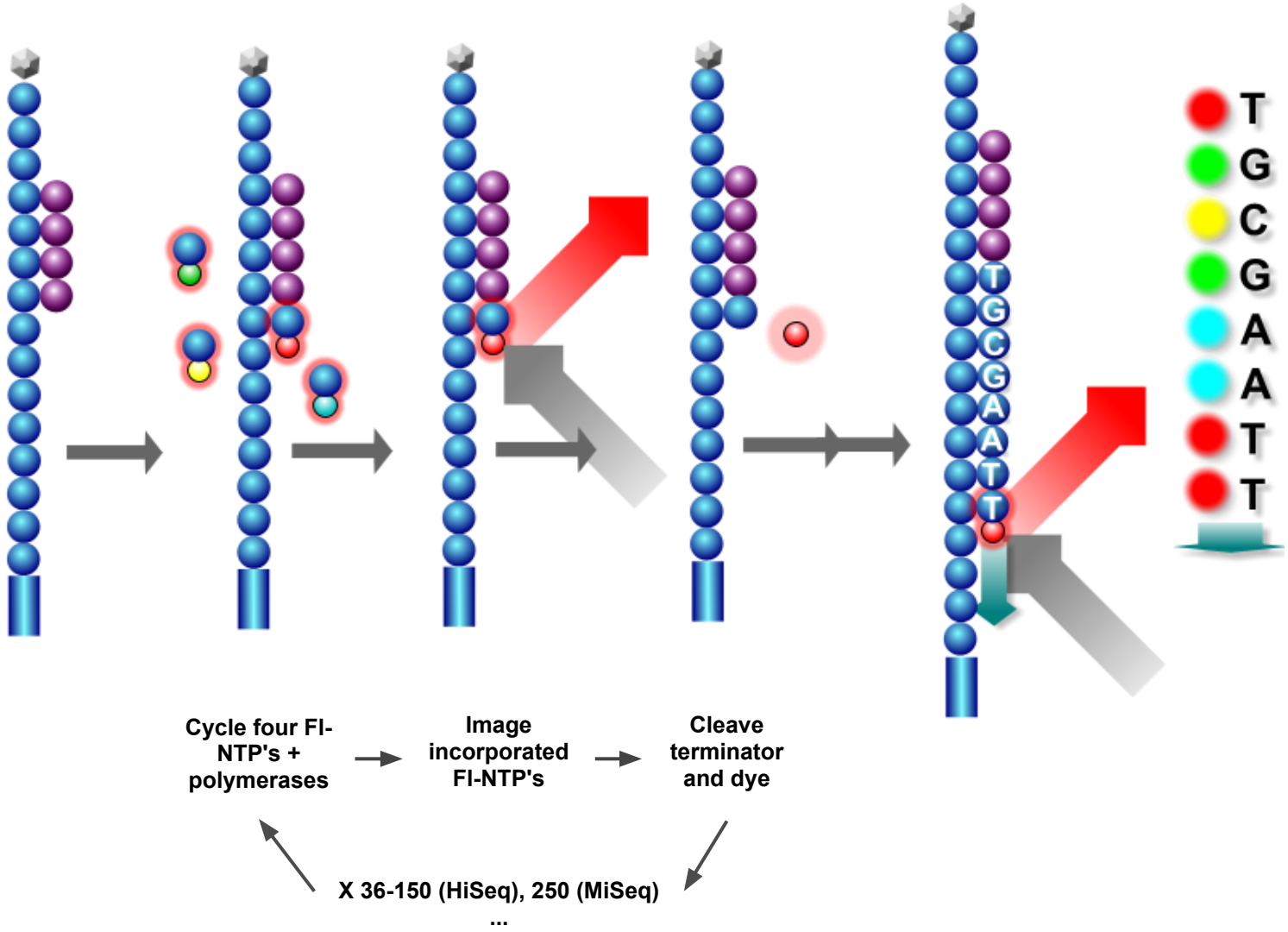
Sequencing By Synthesis



- Sequencing primer is hybridized to adapter sequence, starting Sequencing By Synthesis

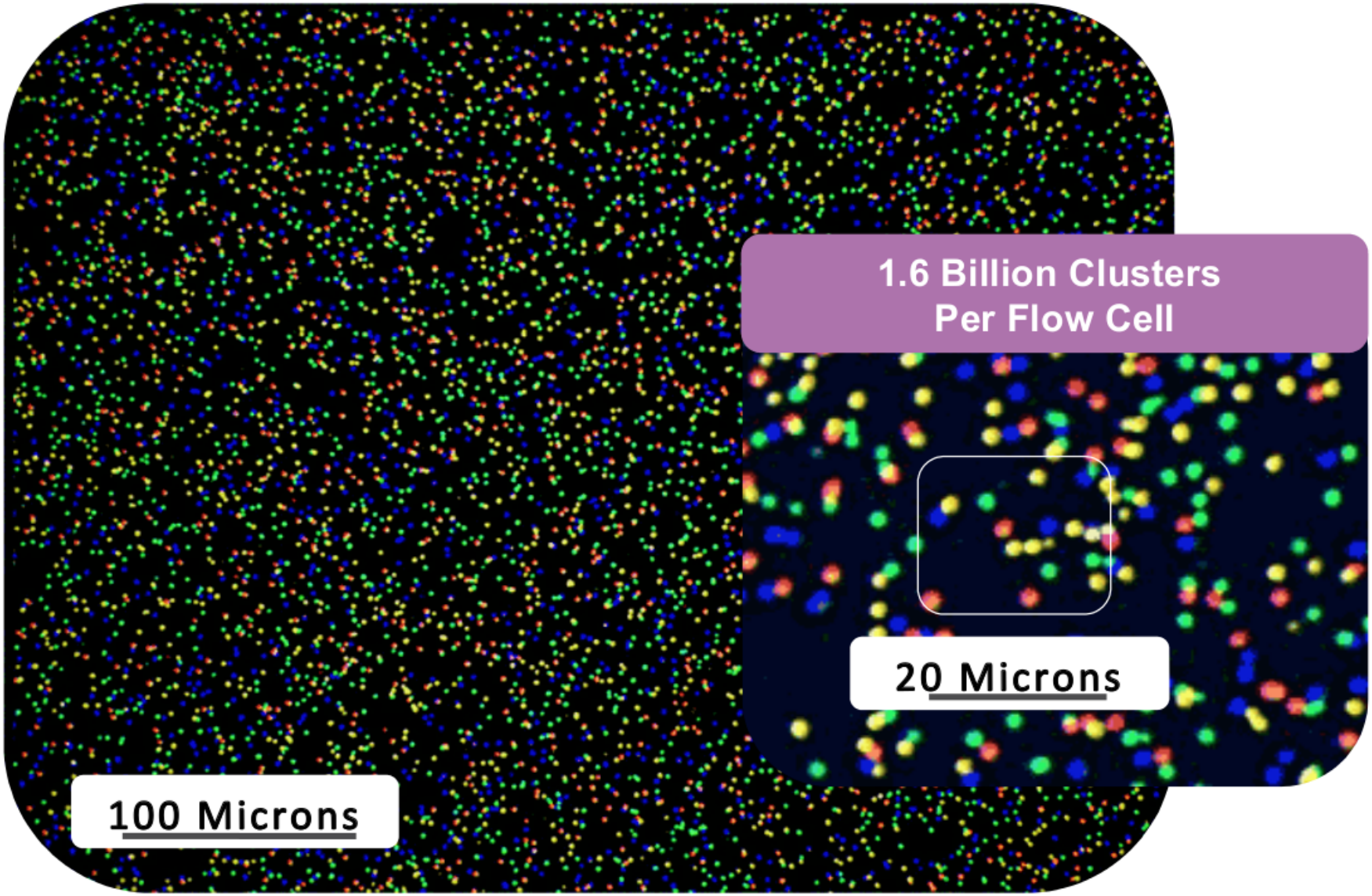
Illumina

Sequencing By Synthesis



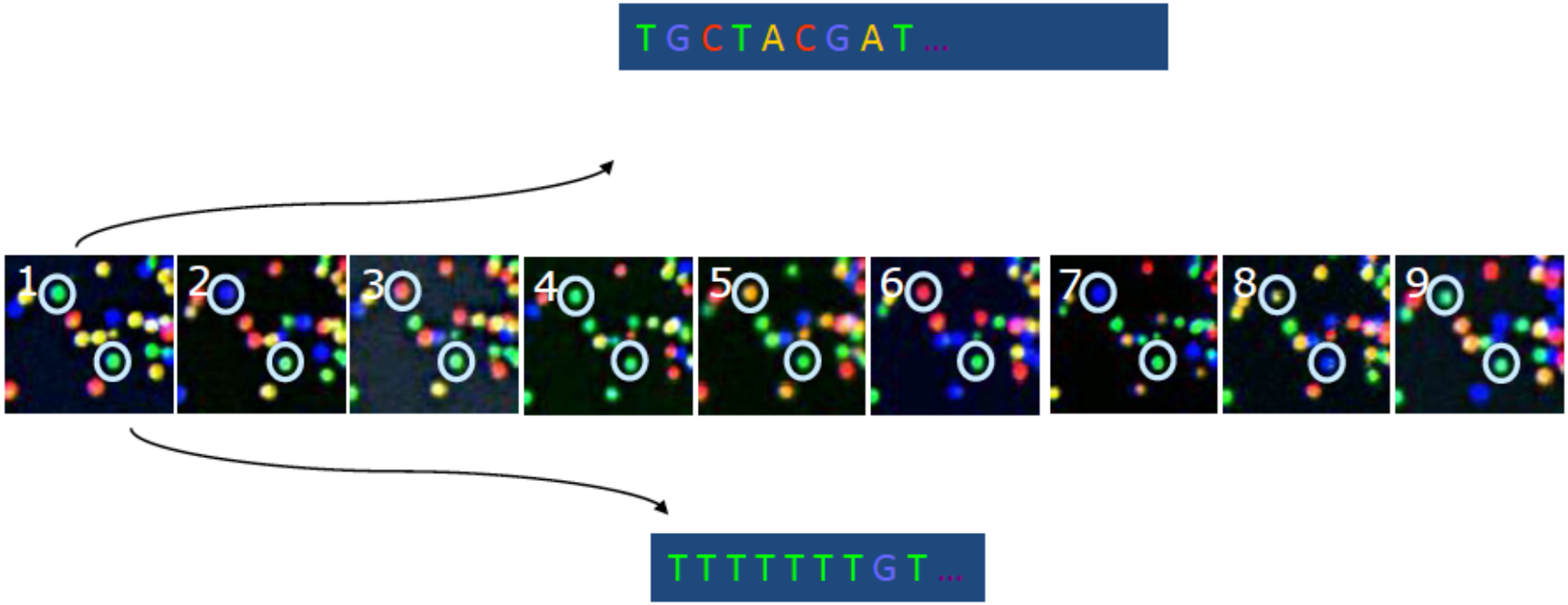
Illumina

Sequencing By Synthesis



Illumina

Sequencing By Synthesis

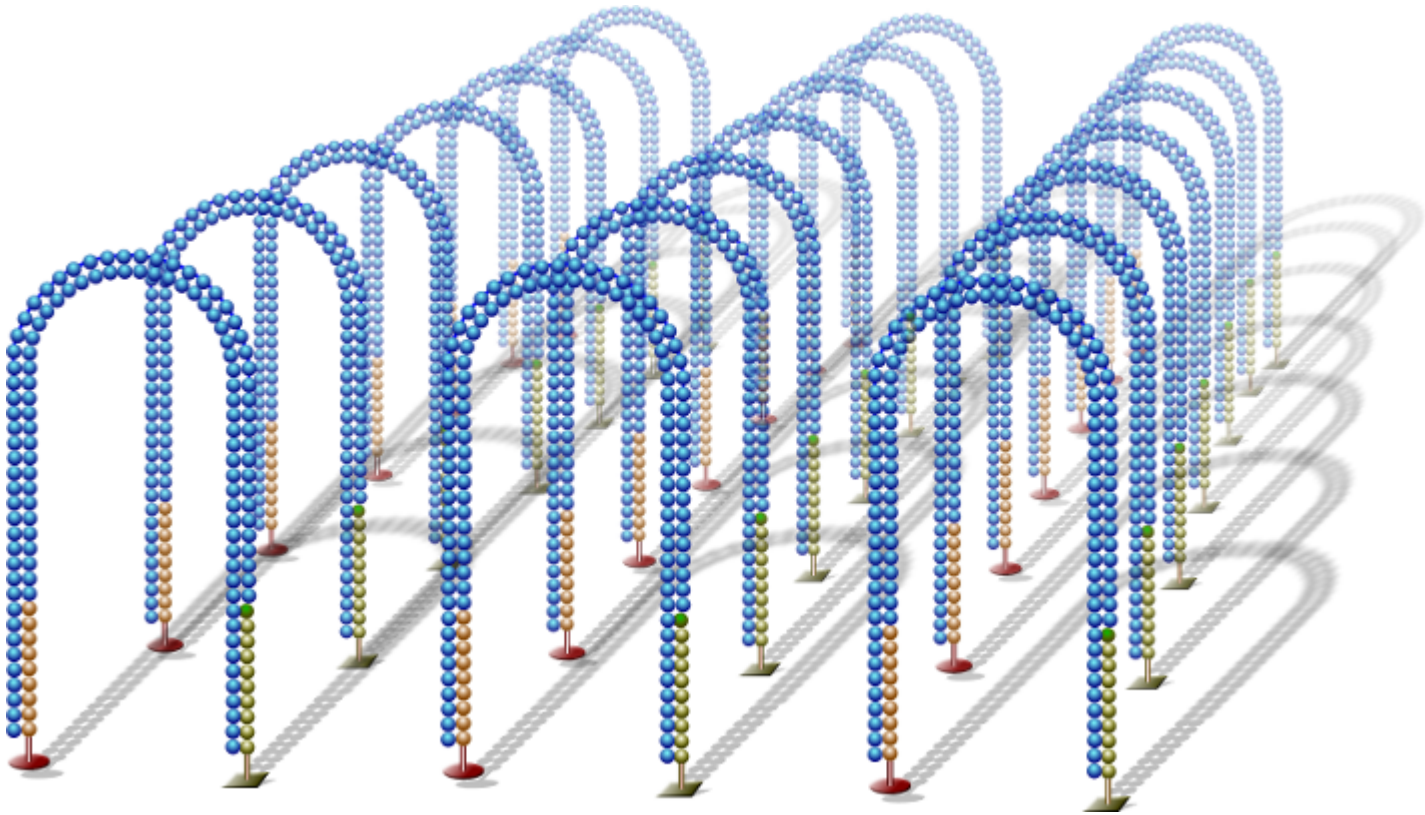


The identity of each base of a cluster is read off from sequential images.

Illumina

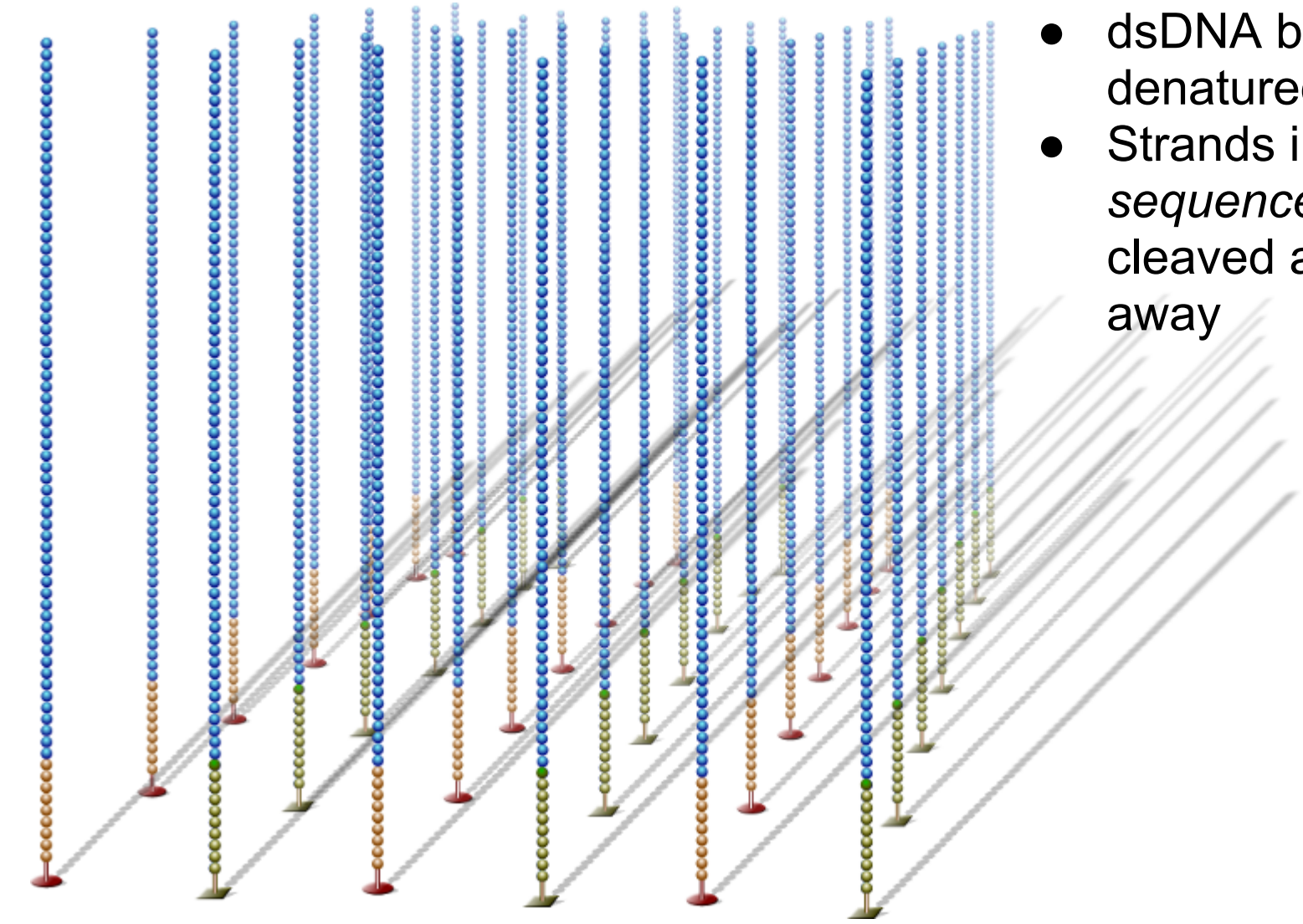
Paired-end sequencing

- Bridge amplification to generate strands with opposite orientation



Illumina

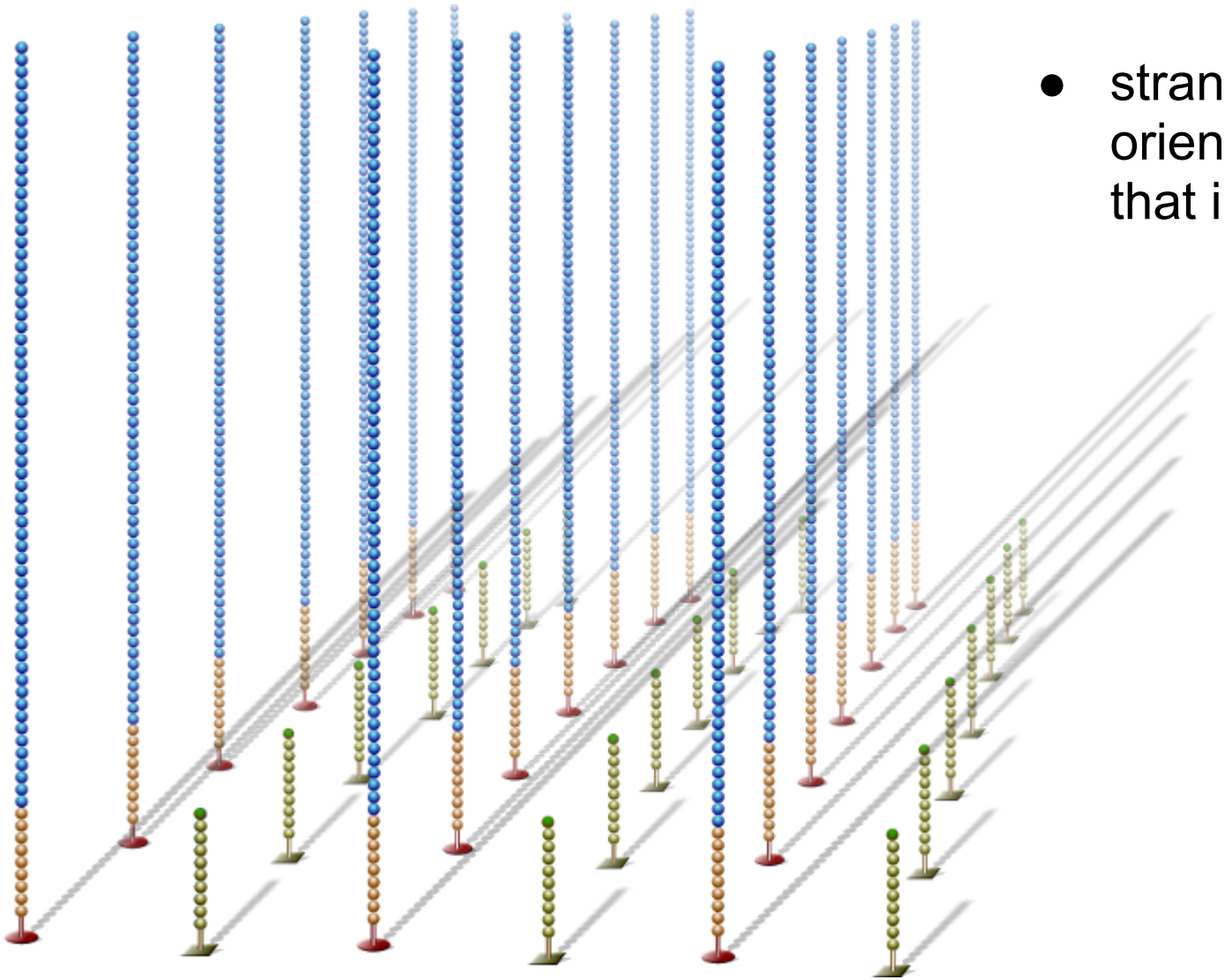
Paired-end sequencing



- dsDNA bridges denatured
- Strands in *already sequenced* orientation cleaved and washed away

Illumina

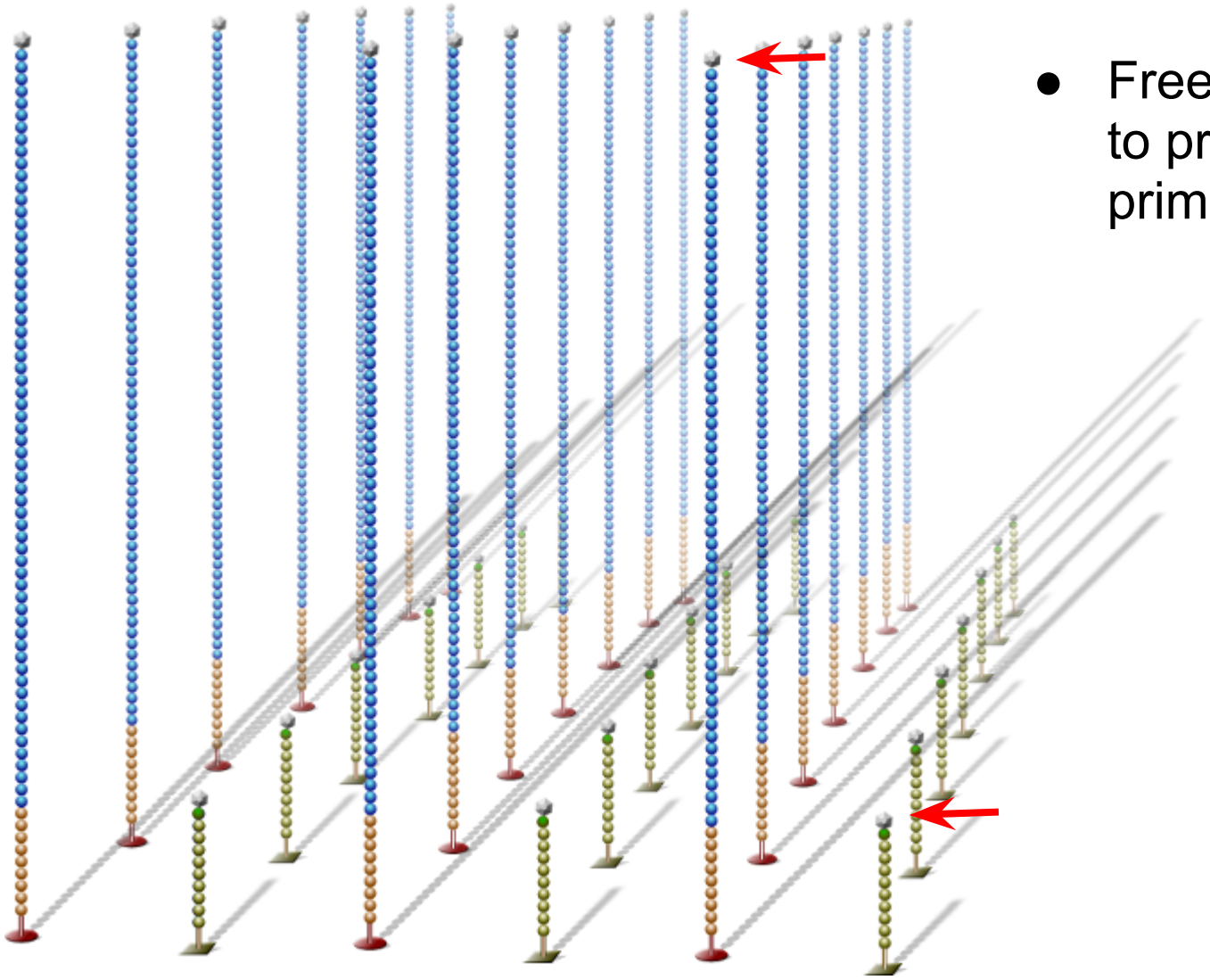
Paired-end sequencing



- strands with uniform orientation, *opposite* that in first read

Illumina

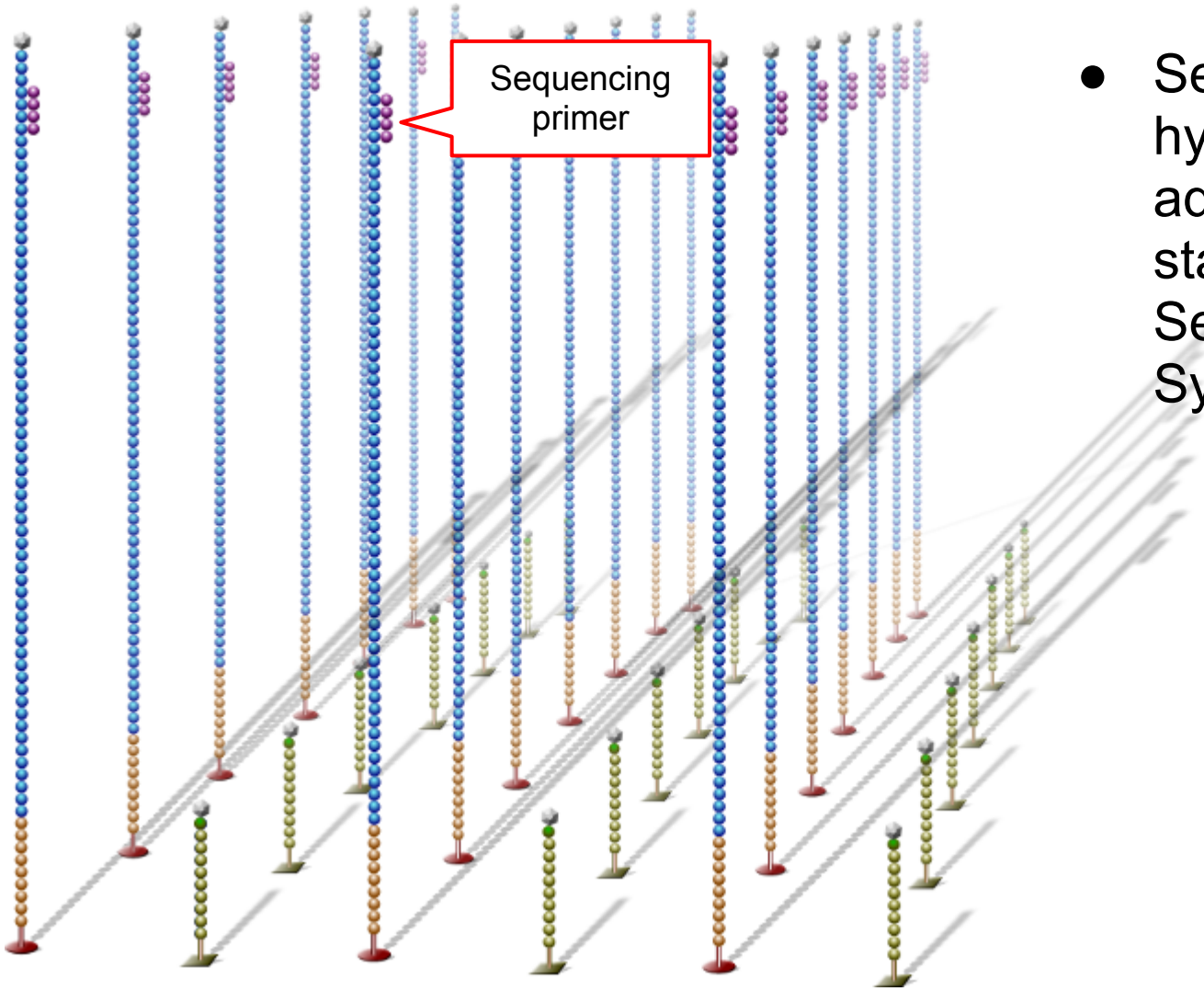
Paired-end sequencing



- Free 3'-ends blocked to prevent unwanted priming

Illumina

Paired-end sequencing



- Sequencing primer is hybridized to other adapter sequence, starting second read's Sequencing By Synthesis

Illumina

HiSeq 2000 stats:

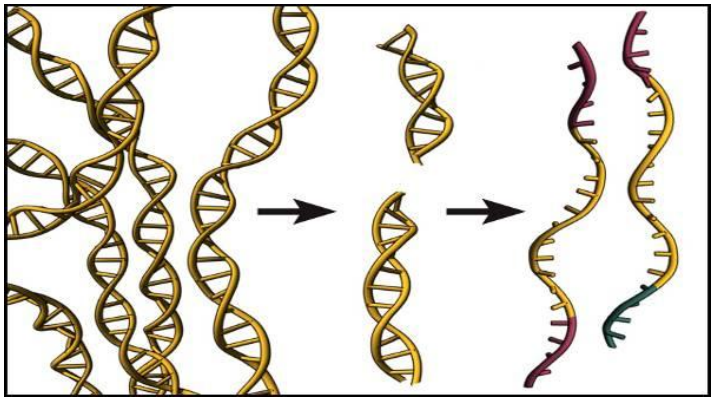
- Dual surface imaging
- Fast scanning and imaging
- Two flow cells (in sequence)
- Initially: 200 Gbp per run
- Currently: 600 Gbp per run
- Run time 7-8 days (100bp PE)
- 25 Gbp / day
- 2 billion paired-end reads
- < $5k per human genome
- < $100 per transcriptome

454

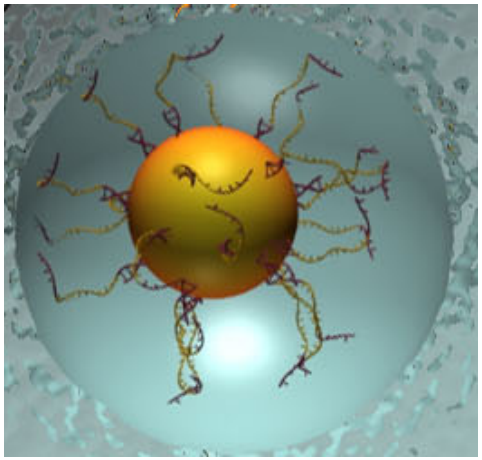
Roche **454** GS FLX Titanium

see http://en.wikipedia.org/wiki/Jonathan_M._Rothberg

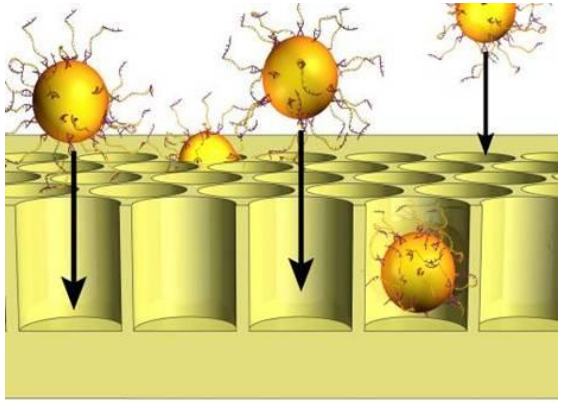
454



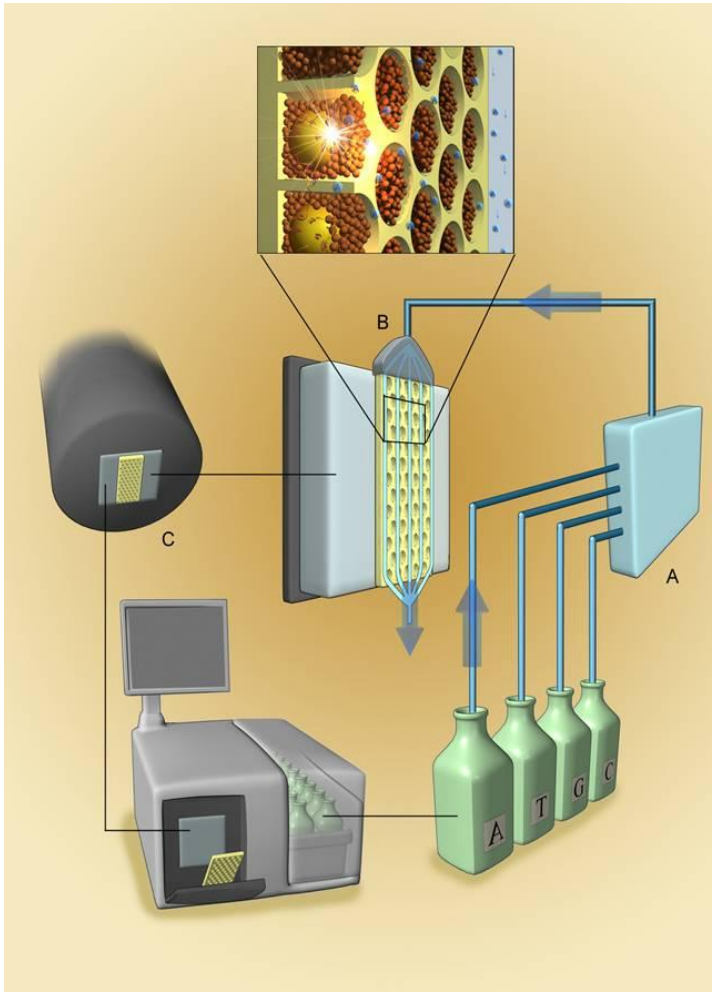
1) Adapter-ligated ssDNA library



2) Clonal amplification on 28 micron beads ... emulsion PCR

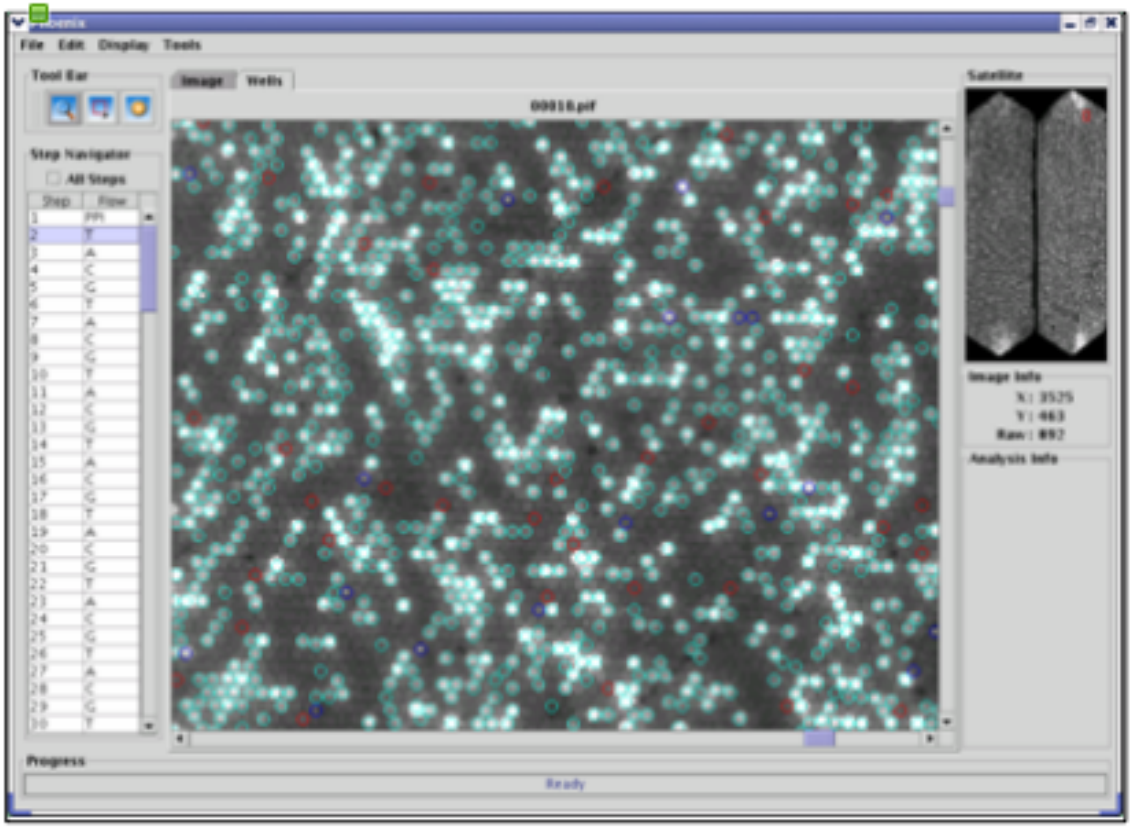


3) Beads deposited on PicoTiterPlate wells



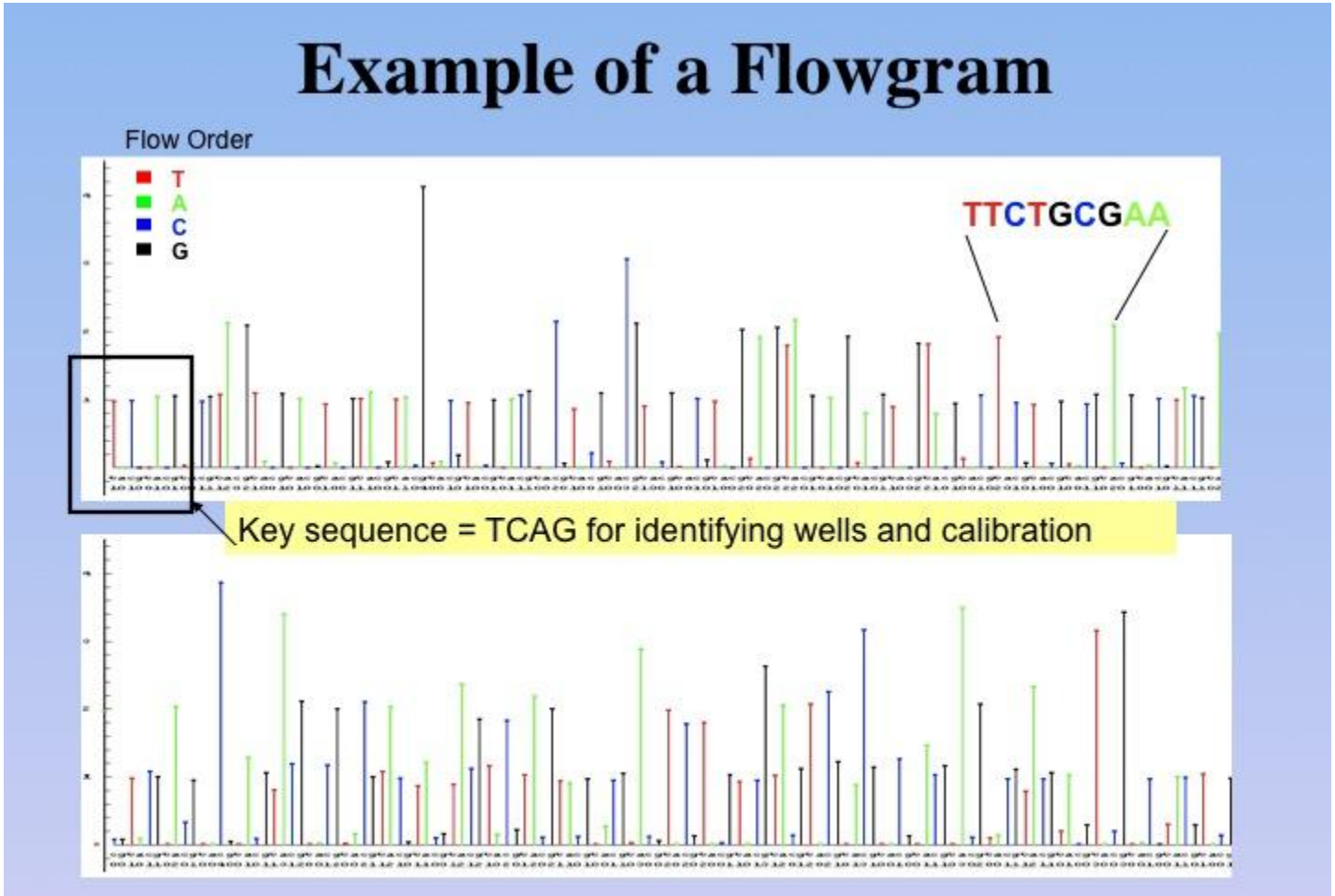
4) Sequencing by synthesis

454



Wells imaged to track fluorescence over time (coordinated with flow of different tagged nucleotides). Nucleotide flow over time is called "flowspace."

454

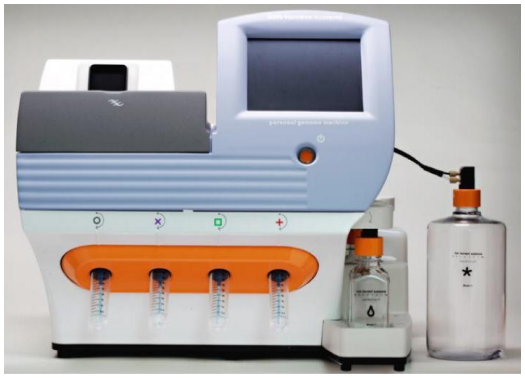


Nucleotides are not "terminated," so *homopolymer runs* add bases all at the same time. Number of bases is inferred from fluorescence signal *amplitude*.

Ion Torrent

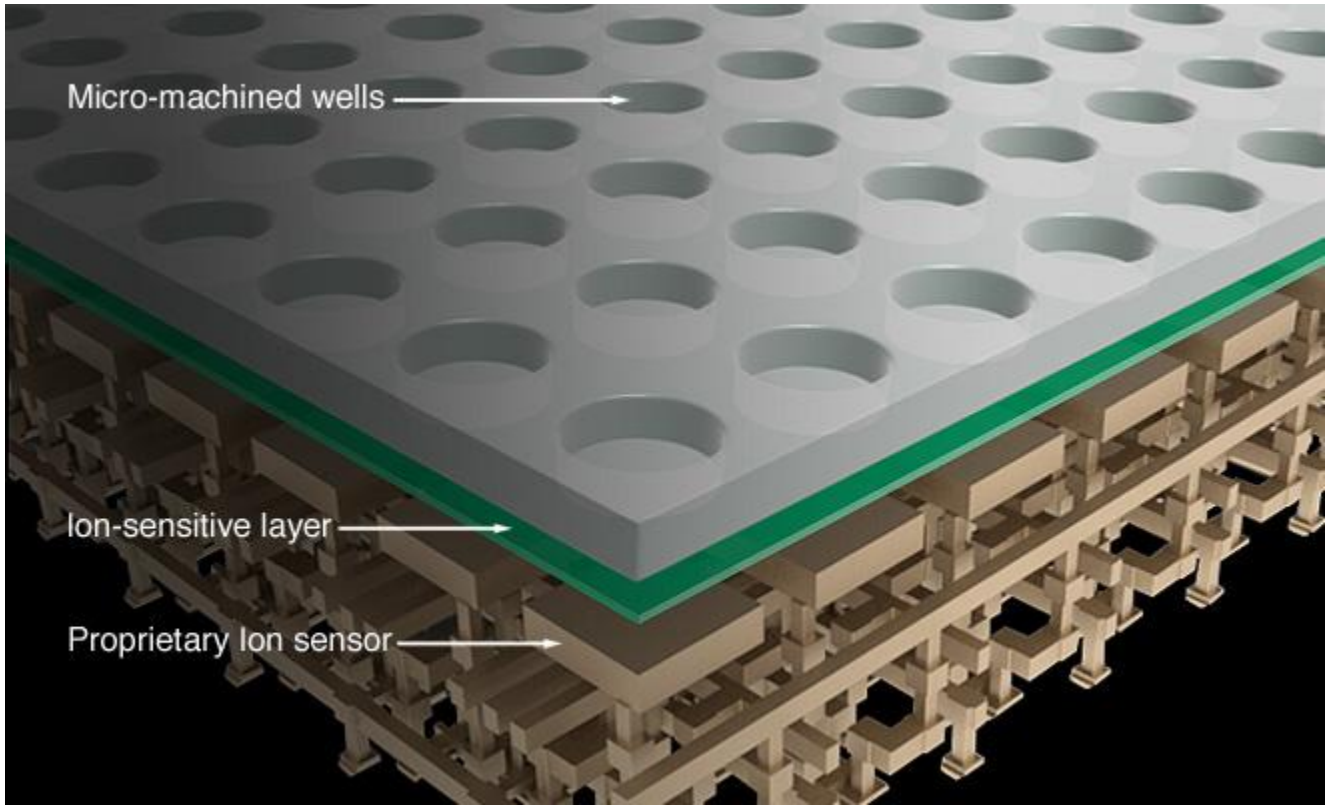
Ion Torrent
(Life Technologies)
Ion Proton

Ion Torrent
(Life Technologies)
Ion PGM



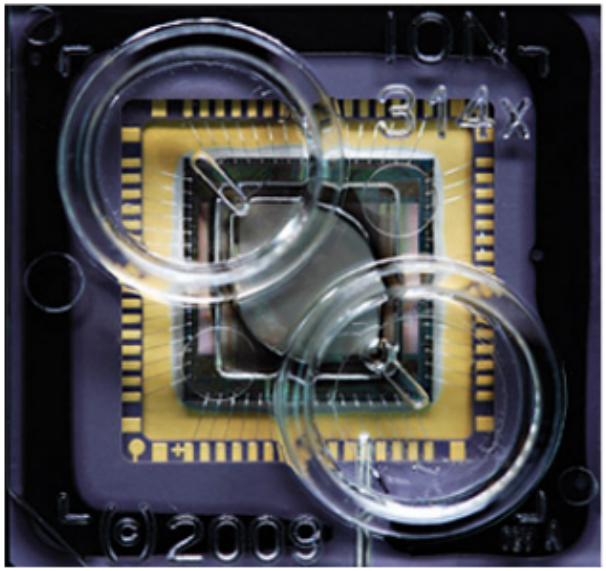
see http://en.wikipedia.org/wiki/Jonathan_M._Rothberg

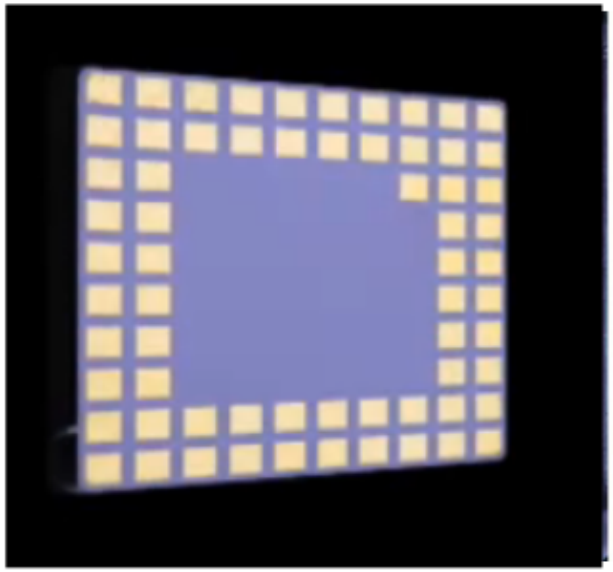
Ion Torrent

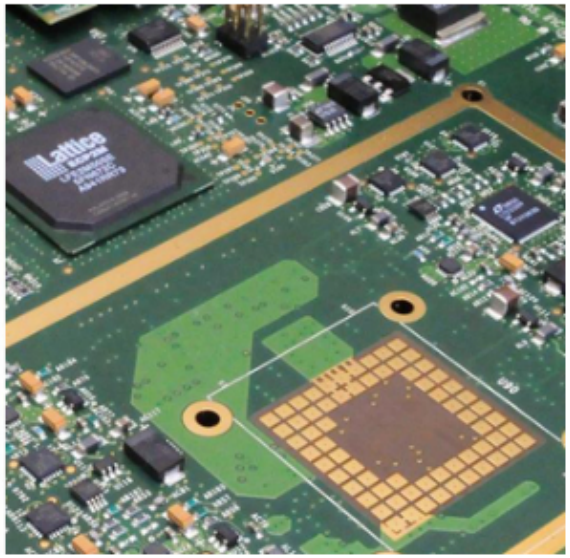


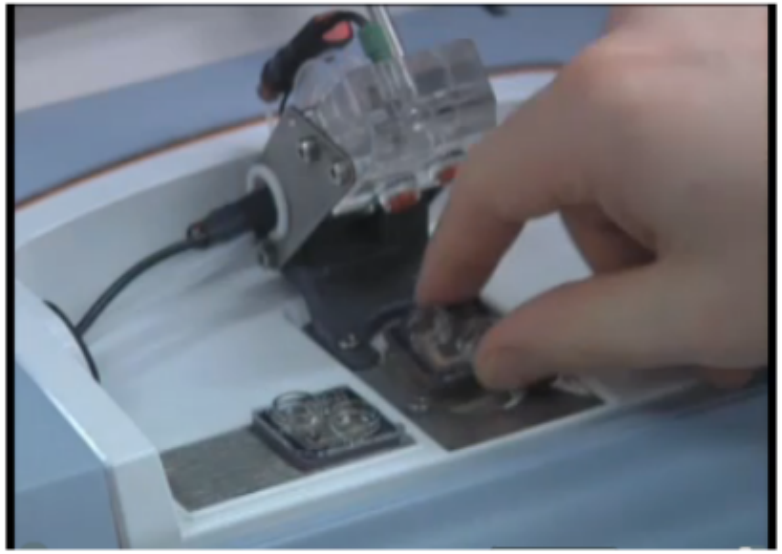
Ion Torrent uses a high-density array of micro-machined wells to perform this simple chemical reaction in a massively parallel way. Each well holds a different DNA template. Beneath the wells is an ion-sensitive layer and beneath that a proprietary Ion sensor.

Ion Torrent

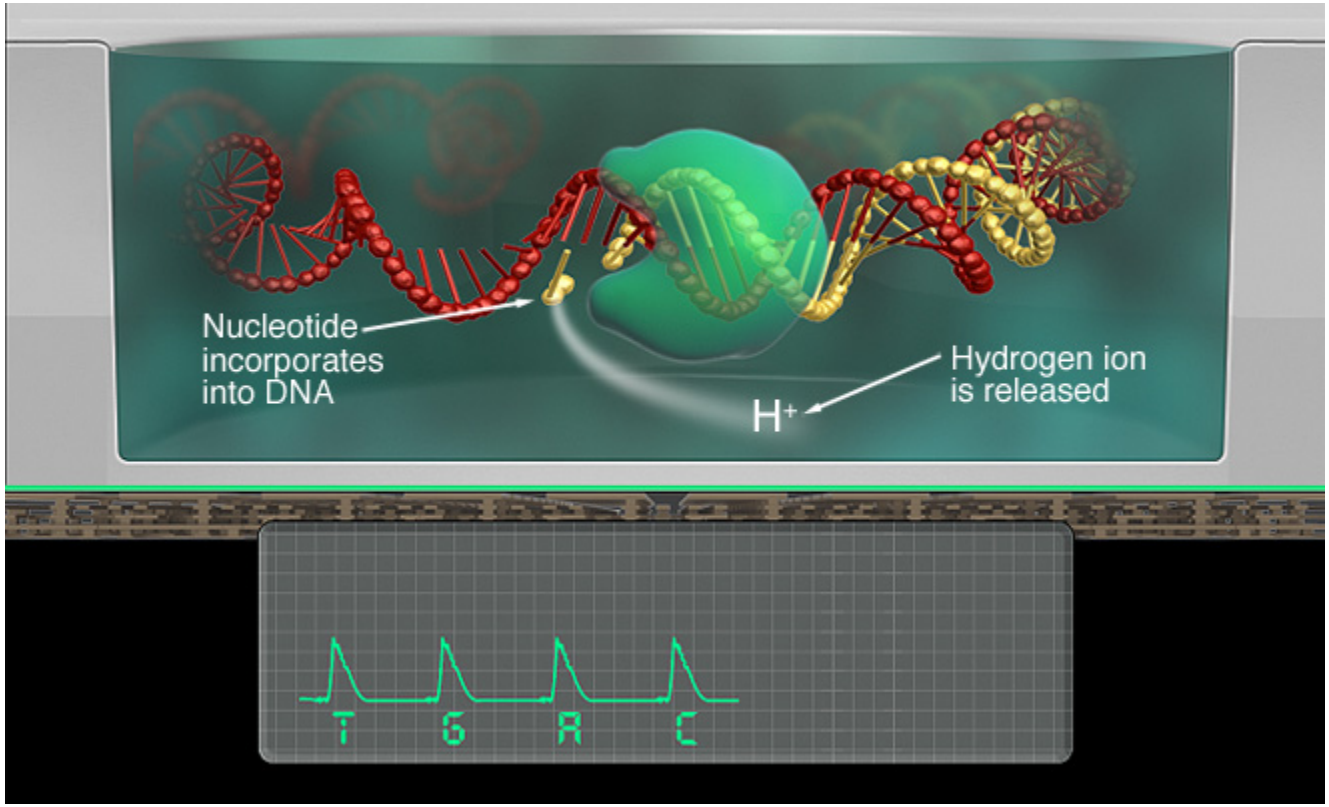








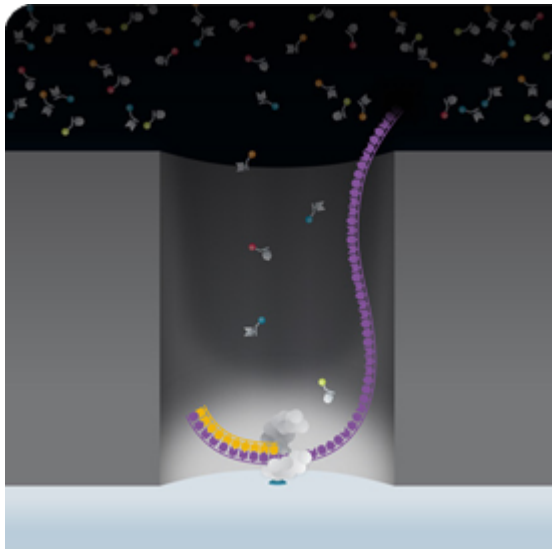
Ion Torrent

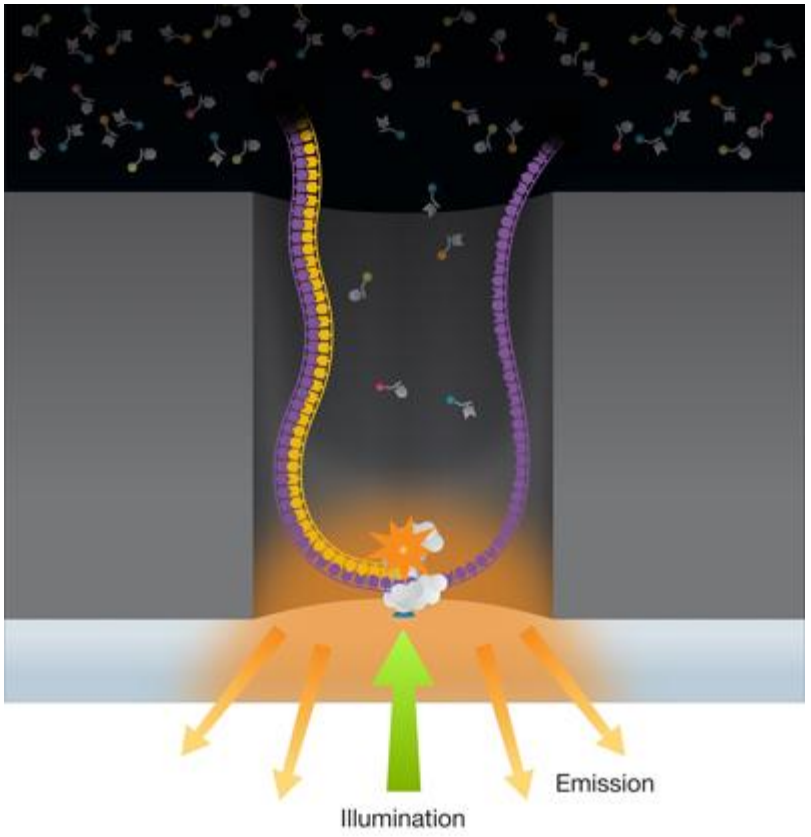


Nucleotide addition results in H+-ion release. Current / pH signal is detected by solid-state sensor ("world's smallest solid-state pH meter"). Nucleotide being "flowed" at the time determines base-call. Like 454, homopolymer run length is inferred from signal amplitude.

PacBio

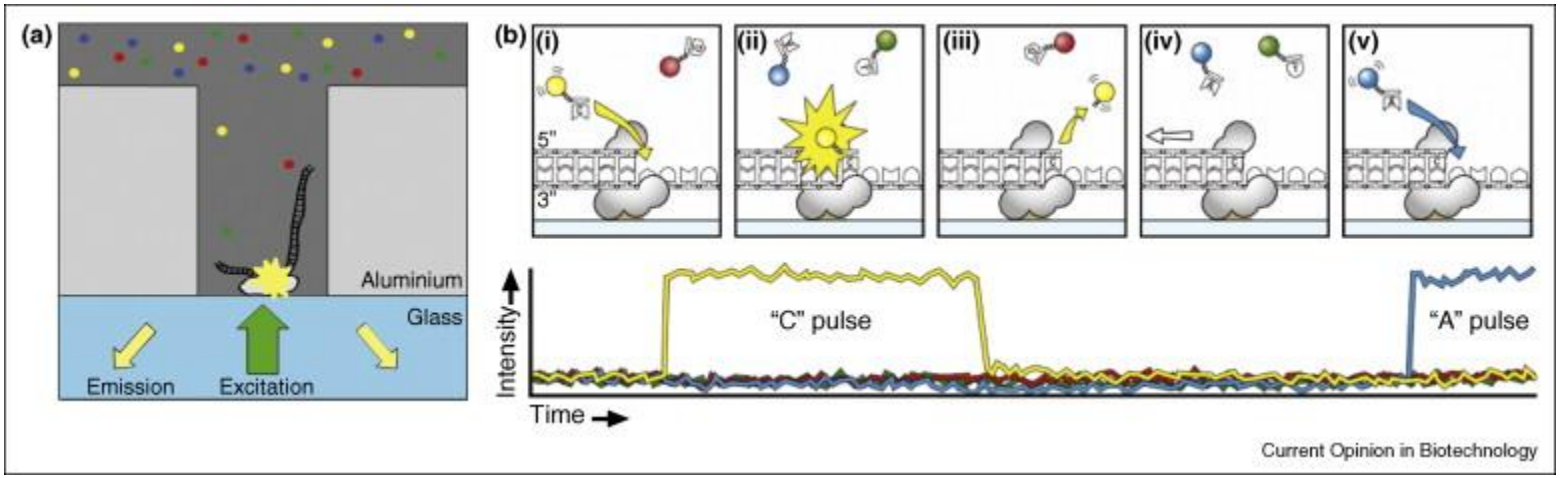
PacBio RS (Real-time Sequencer)





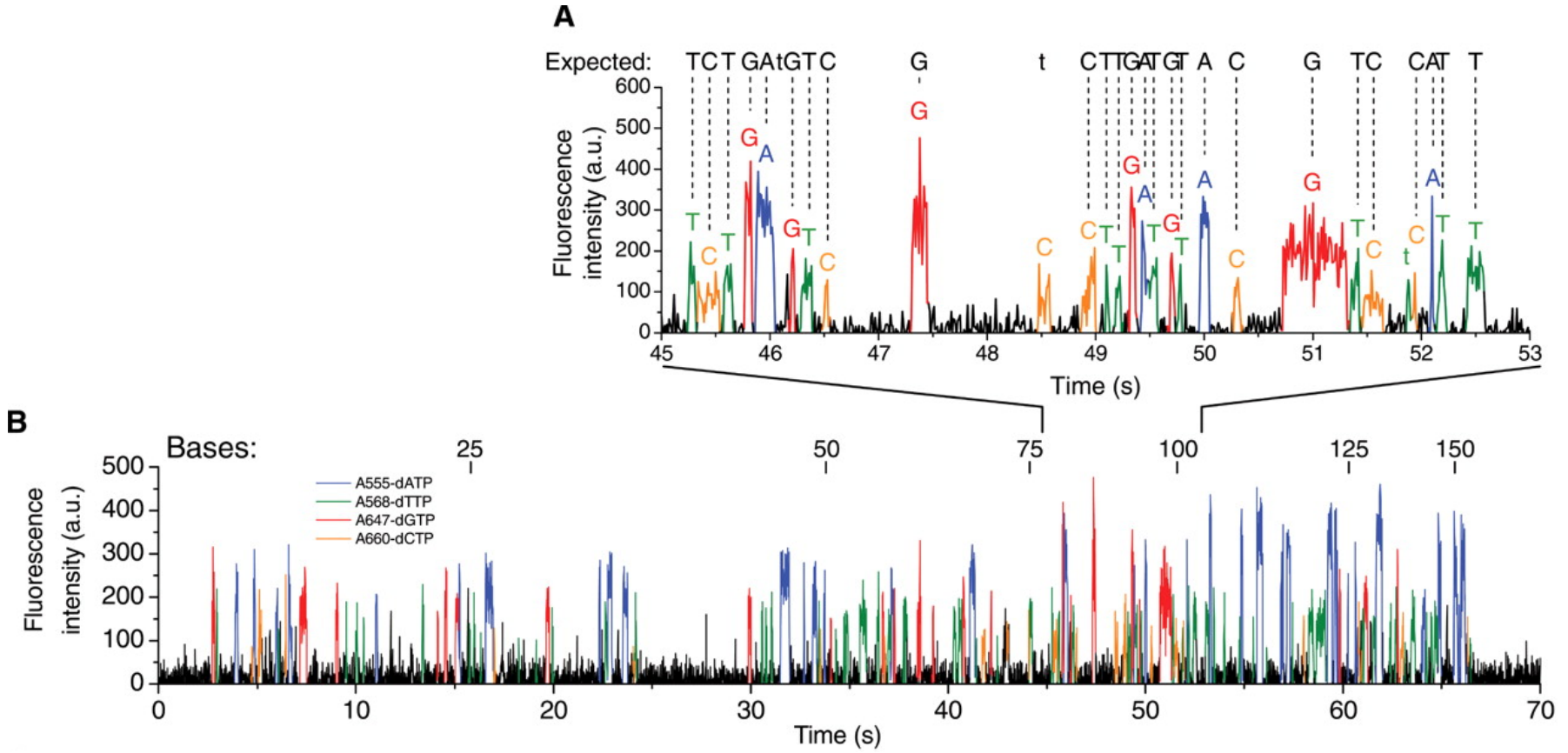
Polymerase / DNA complex adhered to bottom of imaging well (**Z**ero **M**ode **W**aveguide) ... evanescent wave illuminates tiny volume around polymerase.

PacBio RS (Real-time Sequencer)



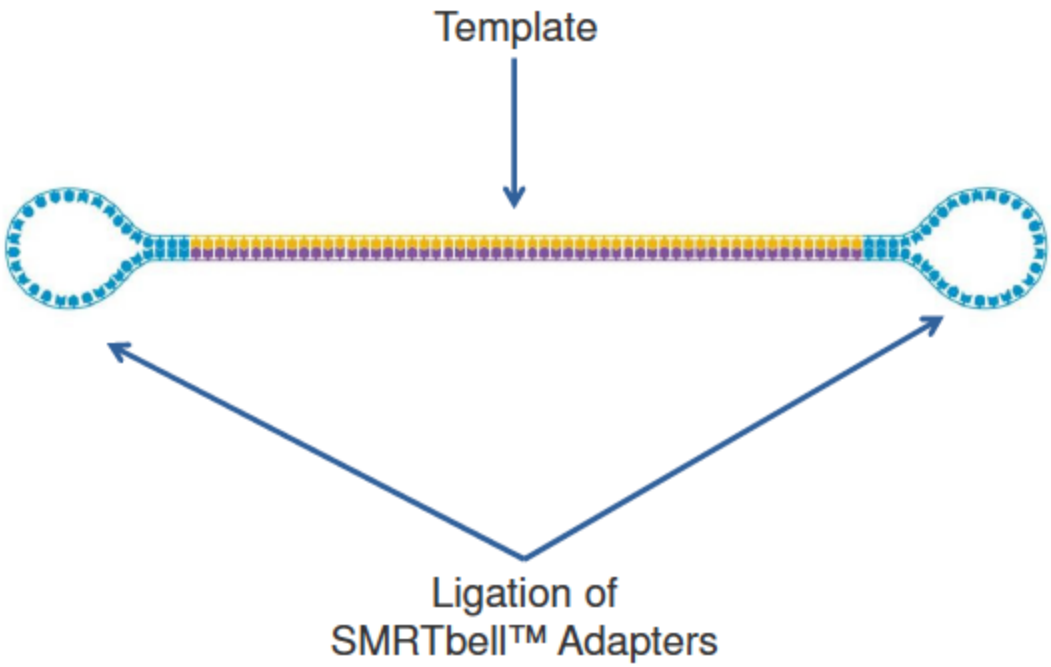
Fluorescently-tagged nucleotides are only seen (for an *appreciable* amount of time) when associated with polymerase. Persistent time in the excitation volume can be recognized as a "pulse."

PacBio RS (Real-time Sequencer)



Science, 02 January 2009/10.1126/science.1162986

PacBio SMRTbell Construct



PacBio SMRTbell Construct

PacBio SMRTbell Construct

PacBio SMRTbell Construct

Standard Sequencing for Continuous Long Reads (CLR)

Large Insert Sizes (>2kb)

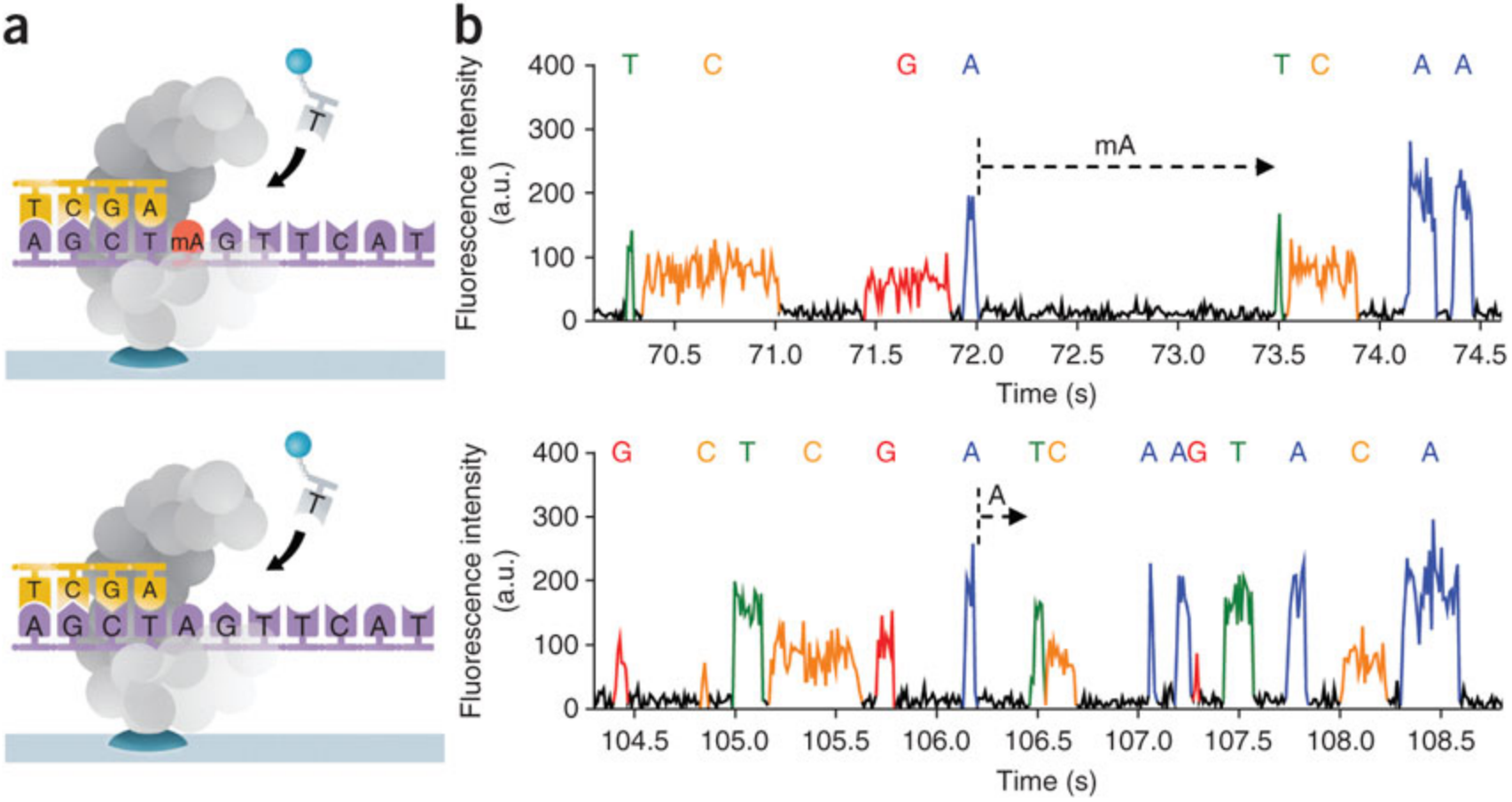
Generates one pass on each molecule sequenced

Circular Consensus Sequencing (CCS)

Small Insert Sizes (250 bp – 2 kb)

Generates multiple passes on each molecule sequenced

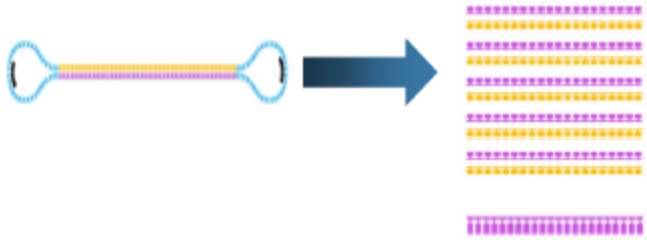
PacBio detection of modified bases



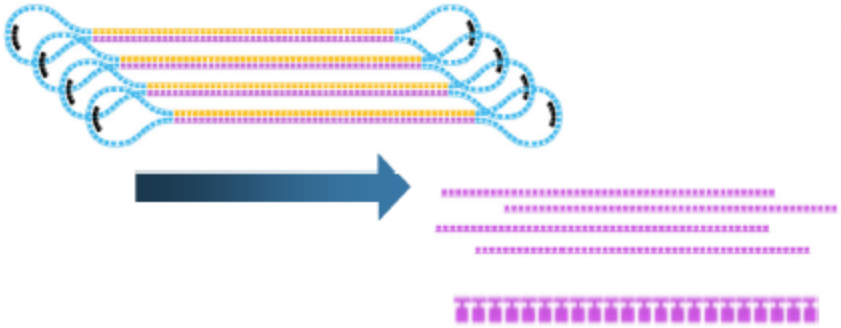
Movie → trace → pulse timing can reveal nucleotide modification, e.g. N6-methyladenosine

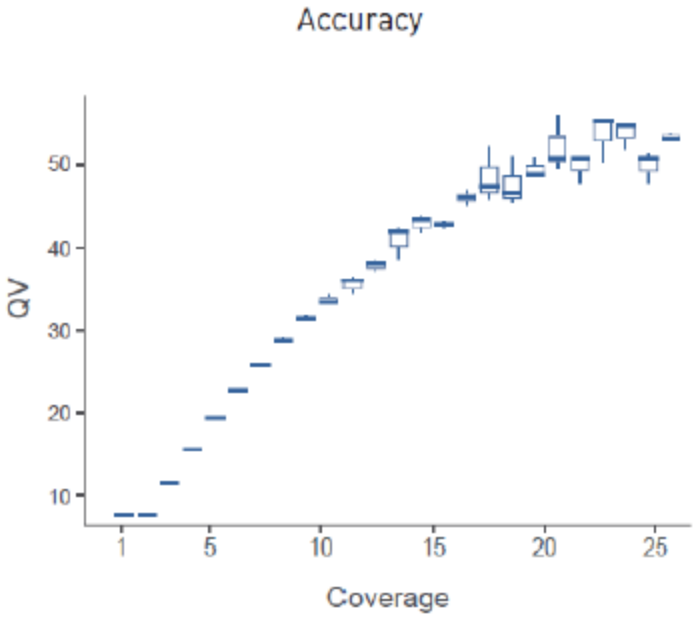
PacBio accuracy

Single-Molecule CCS:



Multi-Molecule Consensus:





Accuracy boost with more coverage

Oxford Nanopore

Oxford Nanopore
MinION

Oxford Nanopore
GridION

Oxford Nanopore

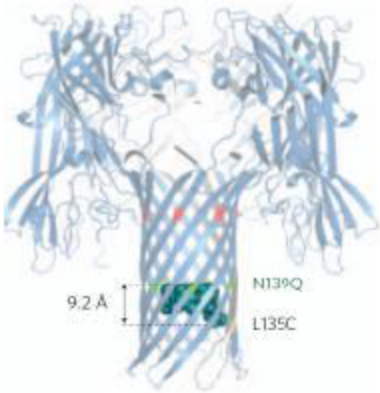
nature nanotechnology

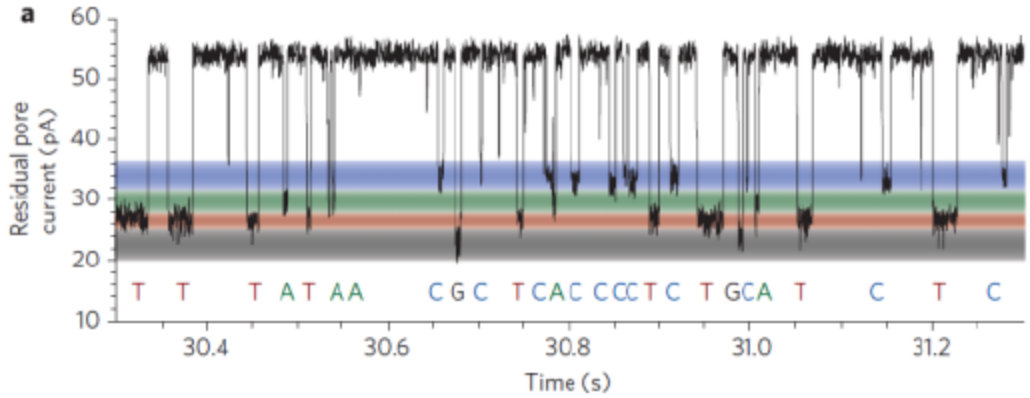
ARTICLES

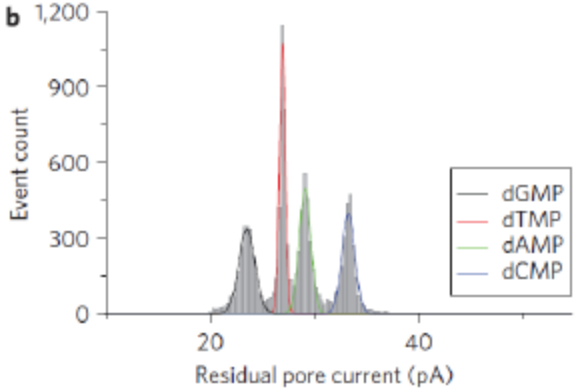
PUBLISHED ONLINE: XX XX 2009 | DOI: 10.1038/NNANO.2009.12

Continuous base identification for single-molecule nanopore DNA sequencing

James Clarke[1], Hai-Chen Wu[2], Lakmal Jayasinghe[1,2], Alpesh Patel[1], Stuart Reid[1] and Hagan Bayley[2]*

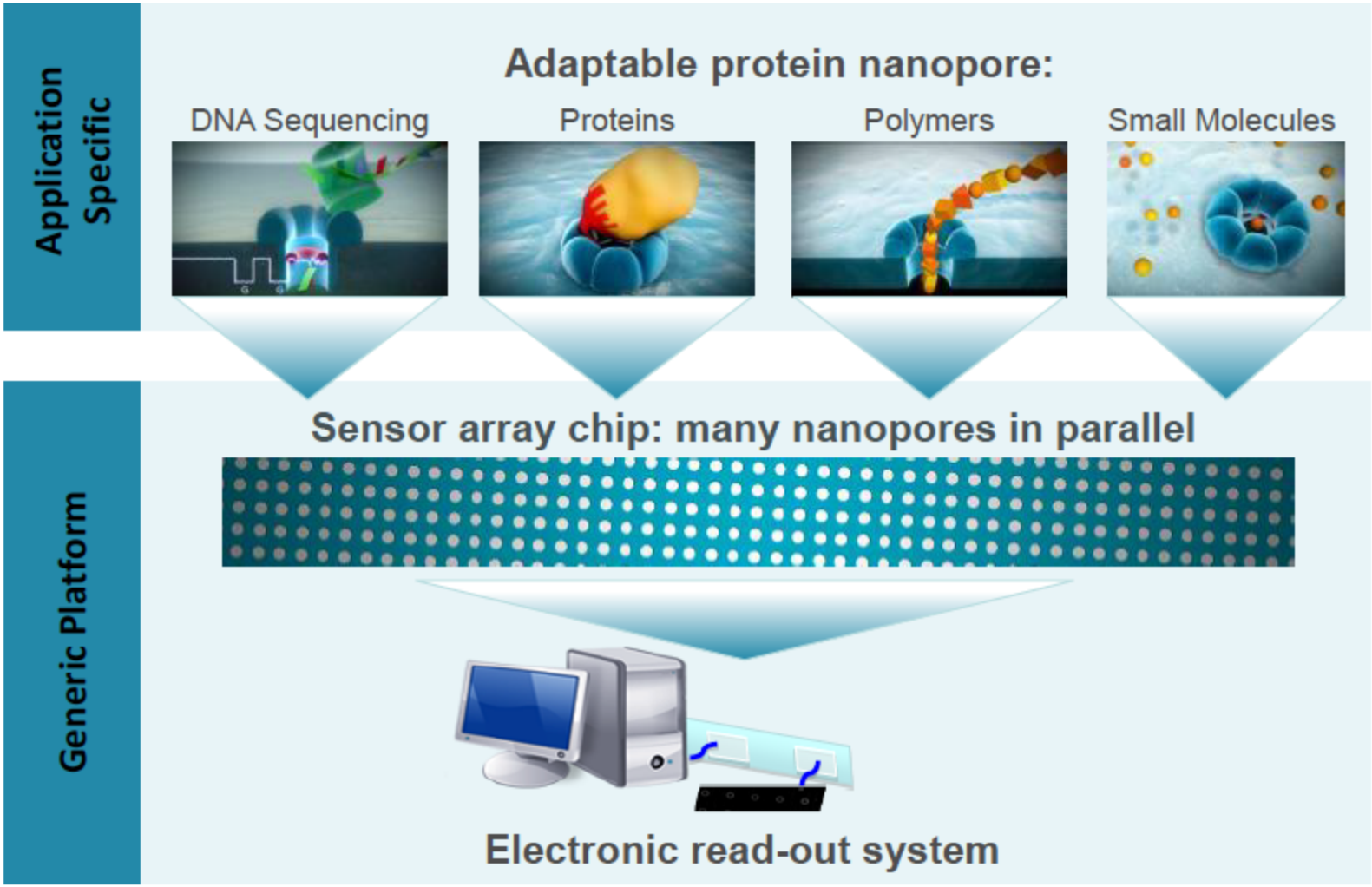






Oxford Nanopore

Oxford Nanopore



Tech Comparison




Feature	HiSeq2000	MiSeq	PacBio RS	Roche/454 FLX+
Number of reads	187 m/lane	15-18 m/lane	~40 K reads/SMRT cell	900-1500k/PTP
Read length	2 x 100 bp	2 x 250 bp	~ 3-10kb (120 min movie)	600-800 bp
Yield per run (PF data)	~37.5 Gb	~8.5 Gb	~Up to 0.2 Gb	~0.9 Gb
Pricing per run	$2,040	$1,179	$250	$6,800
Pricing per Gb	$54	$138	$1,250	$7,555
In Development	2 x 150 bp	2x300 bp	0.5G, 1Gb, 2Gb	???

Ryan Kim, ~Dec. 2012

Tech Comparison

- Non-technology considerations
  - error modes related to application
  - single-molecule preferred?
    - novel isoforms ... software evolving
    - haplotype determination (phasing)
    - base modification
- local expertise (!)
  - library prep
  - secondary analysis
- Availability / turnaround time ಠ_ಠ

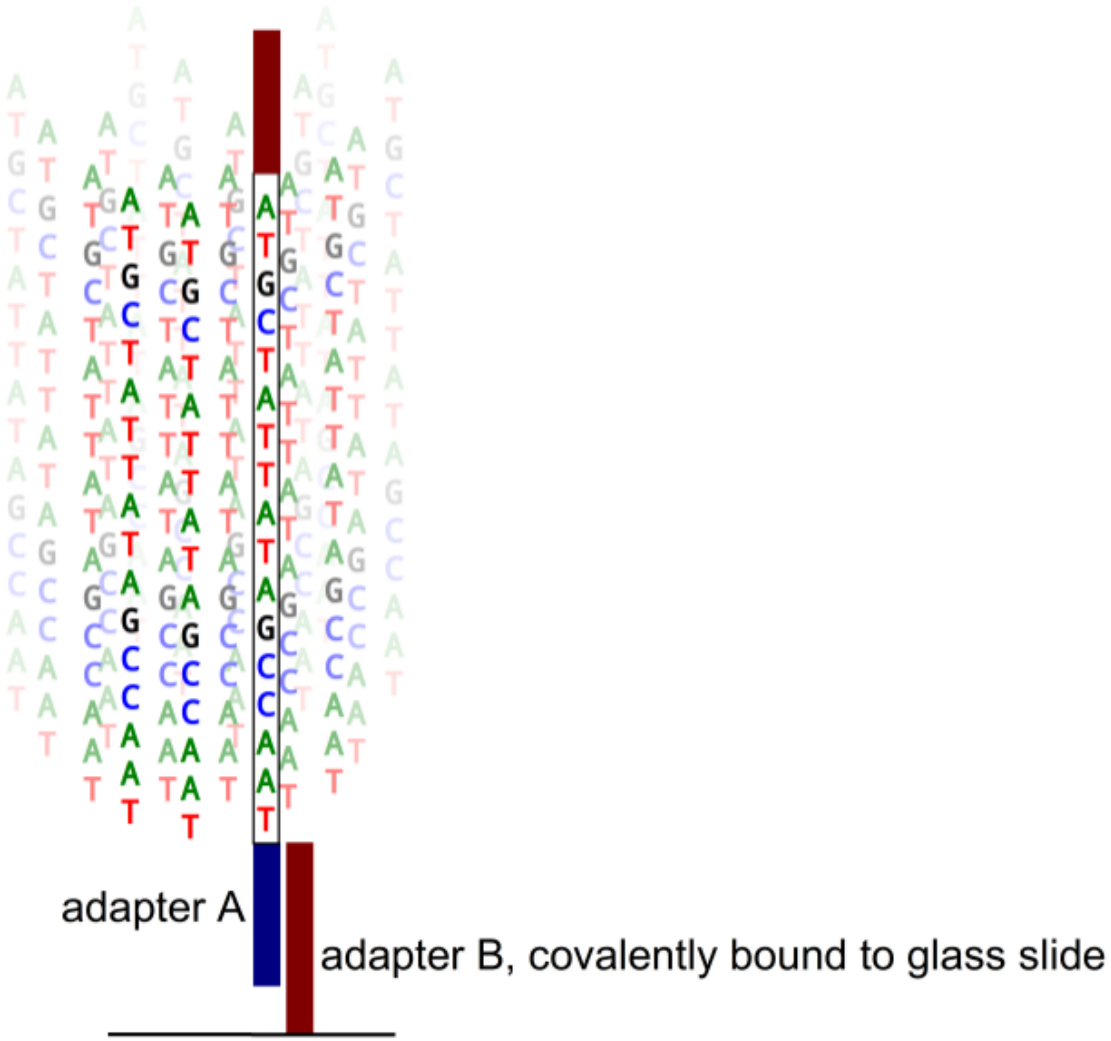
Errors, QC, Grooming

Error modes

Each technology has unique error modes, depending on the physico-chemical processes involved in *the whole sequencing life cycle* (not just base-calling step).

Illumina

3'-end noise



Illumina

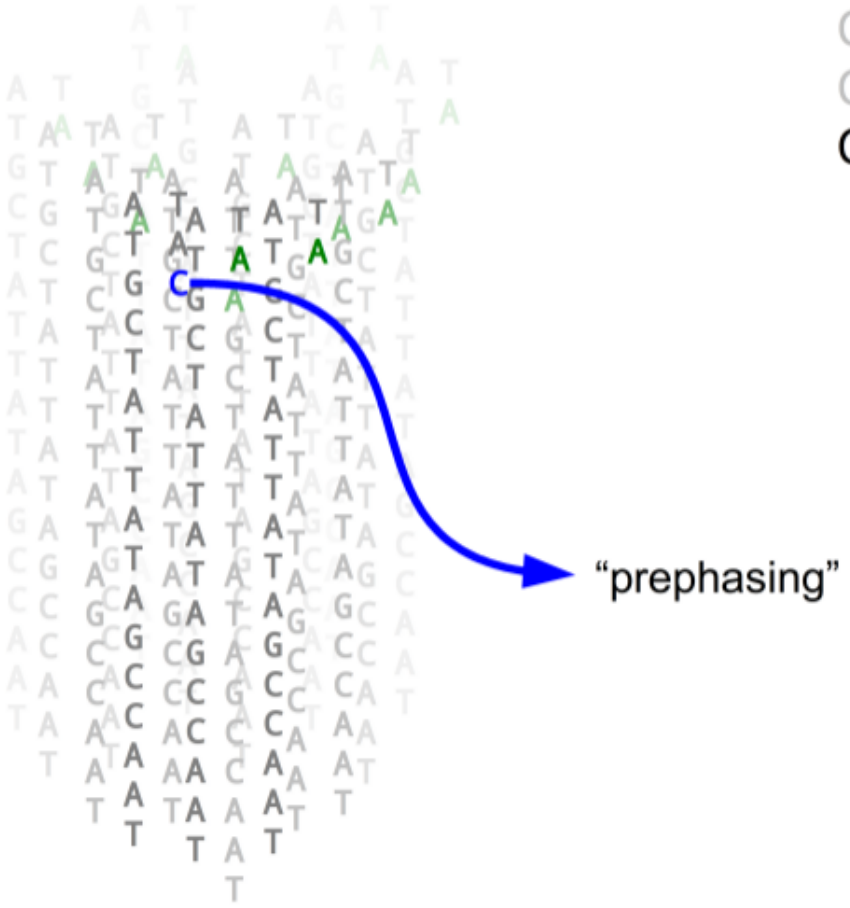
3'-end noise

Cluster generation

Cycle 1 *read as*: T

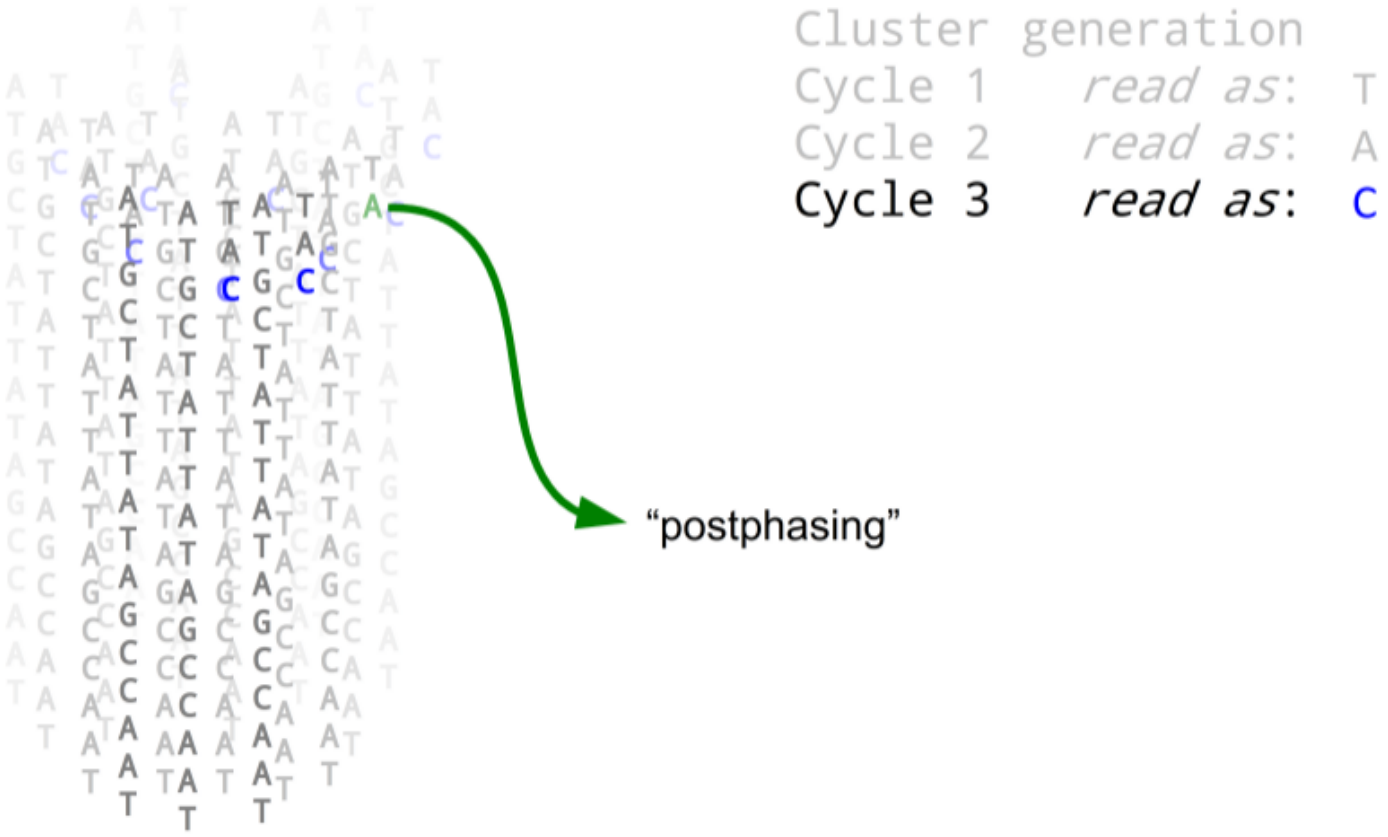
Illumina

3'-end noise



Illumina

3'-end noise



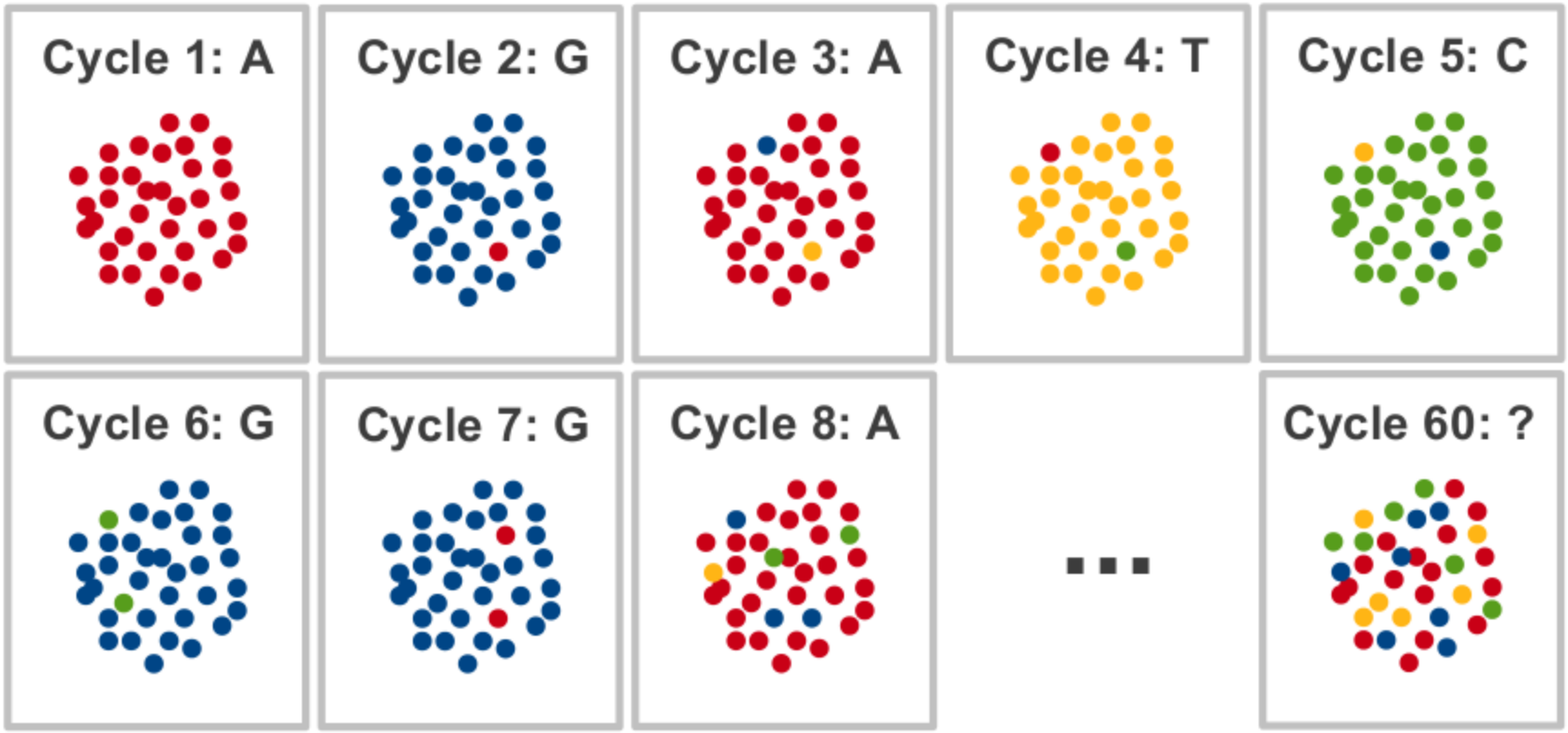
Illumina

3'-end noise

```
Cluster generation
Cycle 1   read as:  T
Cycle 2   read as:  A
Cycle 3   read as:  C
Cycle 4   read as:  G
Cycle 5   read as:  A
Cycle 6   read as:  T
Cycle 7   read as:  A
Cycle 8   read as:  A
Cycle 9   read as:  T
Cycle 10  read as:  A
Cycle 11  read as:  ?
Cycle 12  read as:  ?
Cycle 13  read as:  ?
Cycle 14  read as:  ?
Cycle 15  read as:  ?
Cycle 16  read as:  ?
```

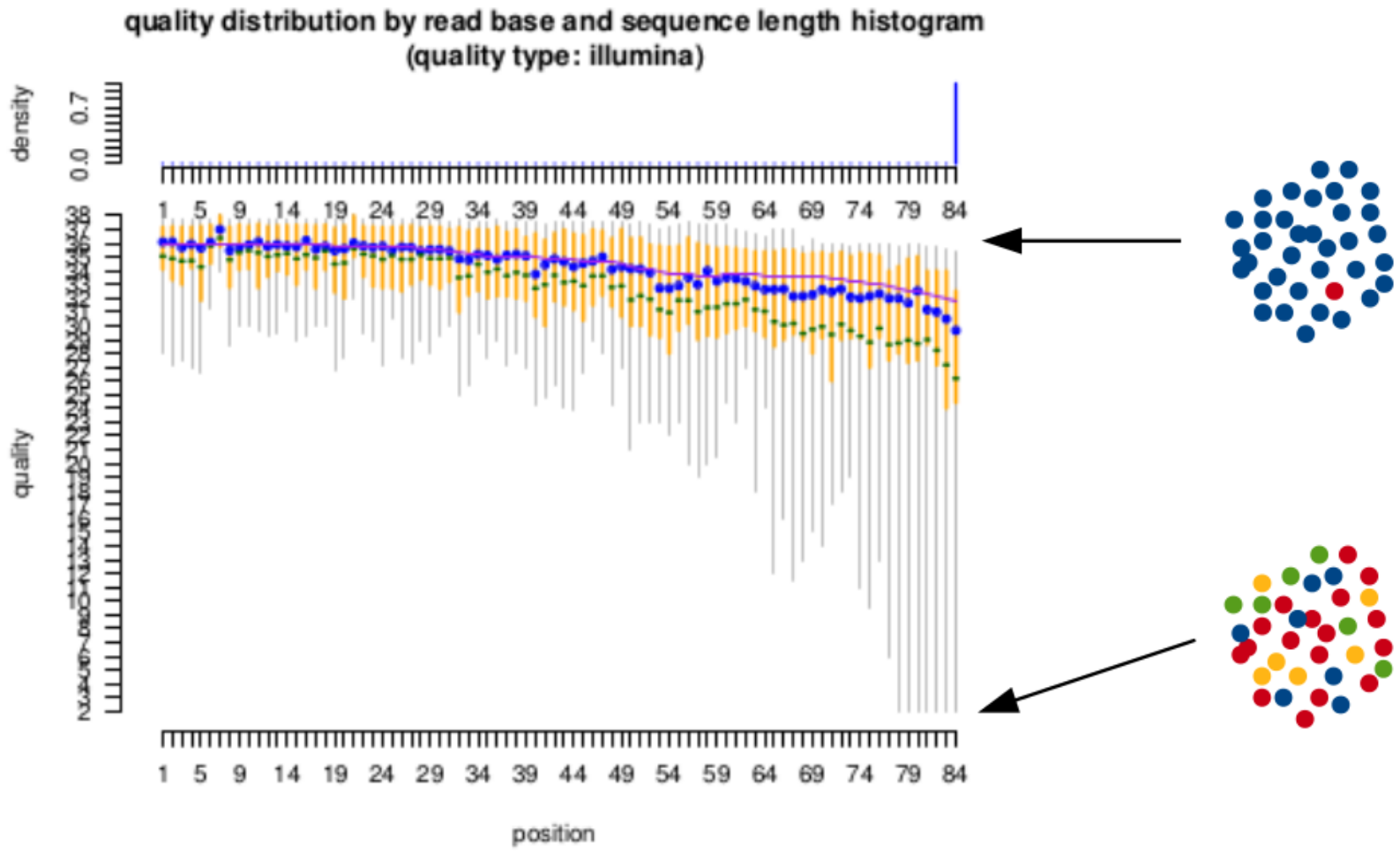
Illumina

3'-end noise

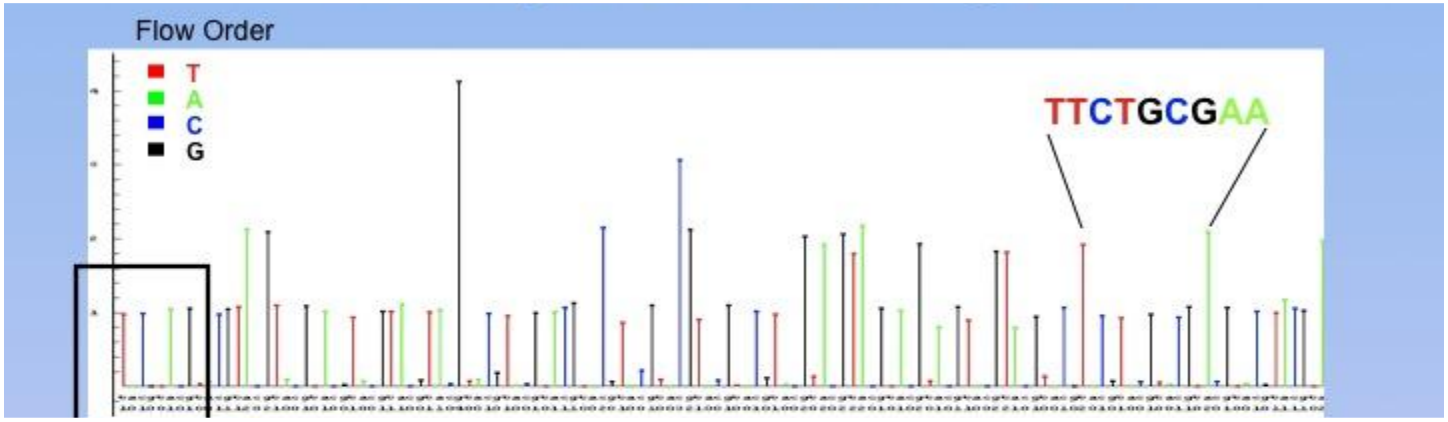


Illumina

3'-end noise



454, Ion Torrent *homopolymer runs*

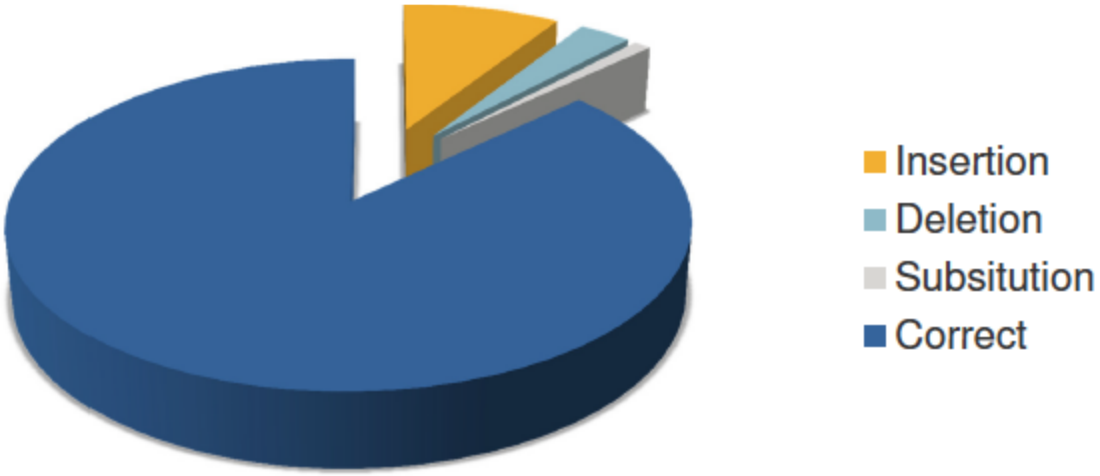


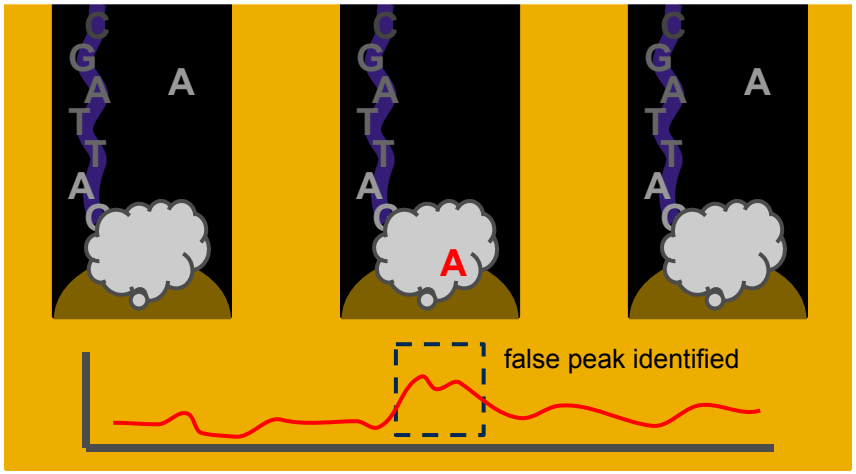
Nucleotides are not "terminated," so *homopolymer runs* add bases all at the same time.

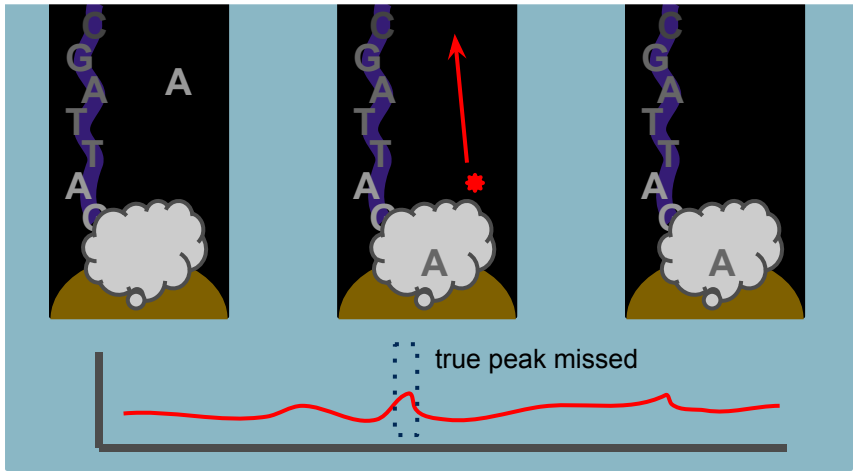
An absolute level of uncertainty / noise in the fluorescence signal will have a greater proportional effect on longer homopolymer runs.

I.e. 4 A's vs 5 A's is a 25% difference
... 8 A's vs 9 A's is a 12% difference

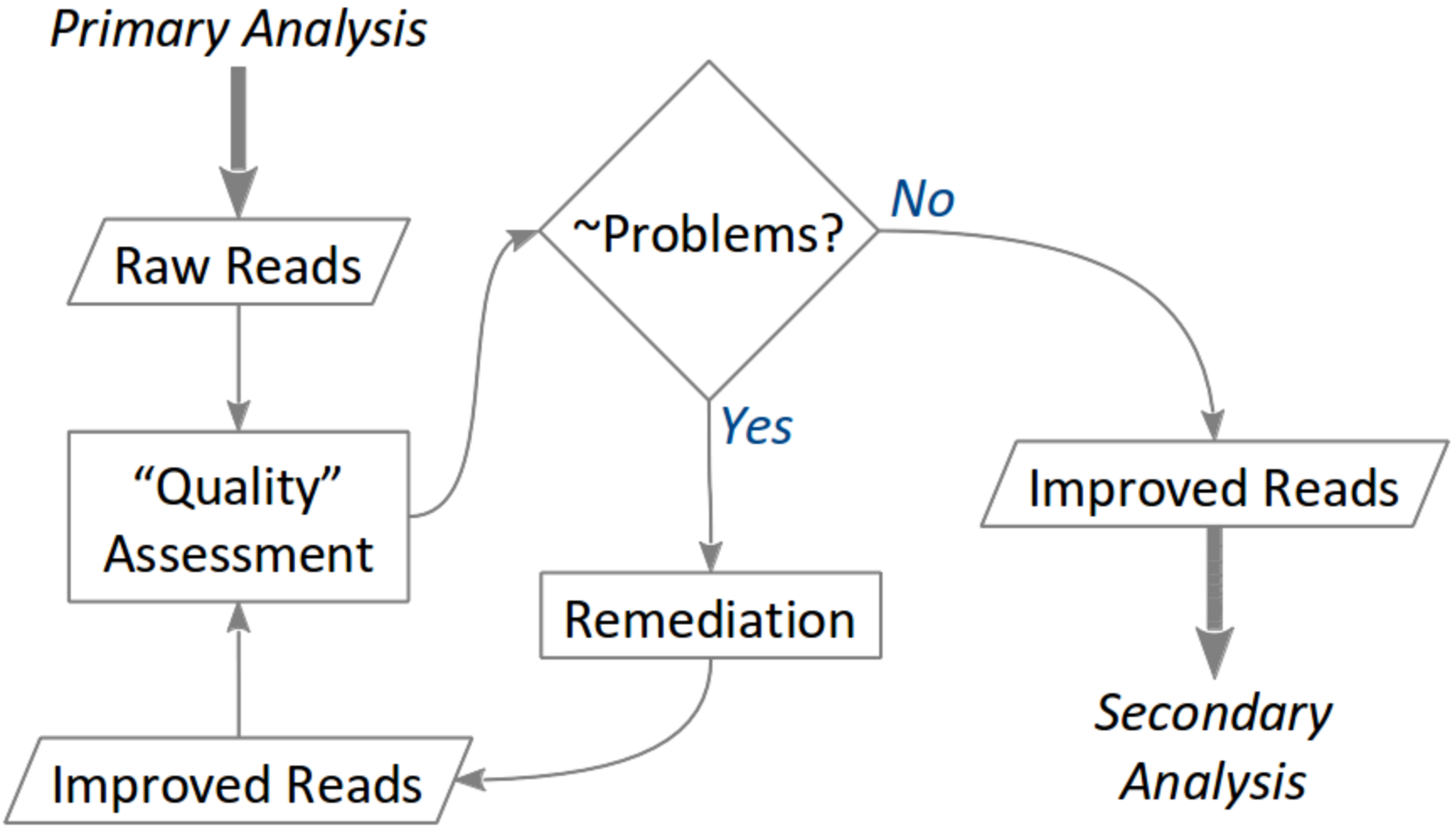
PacBio errors







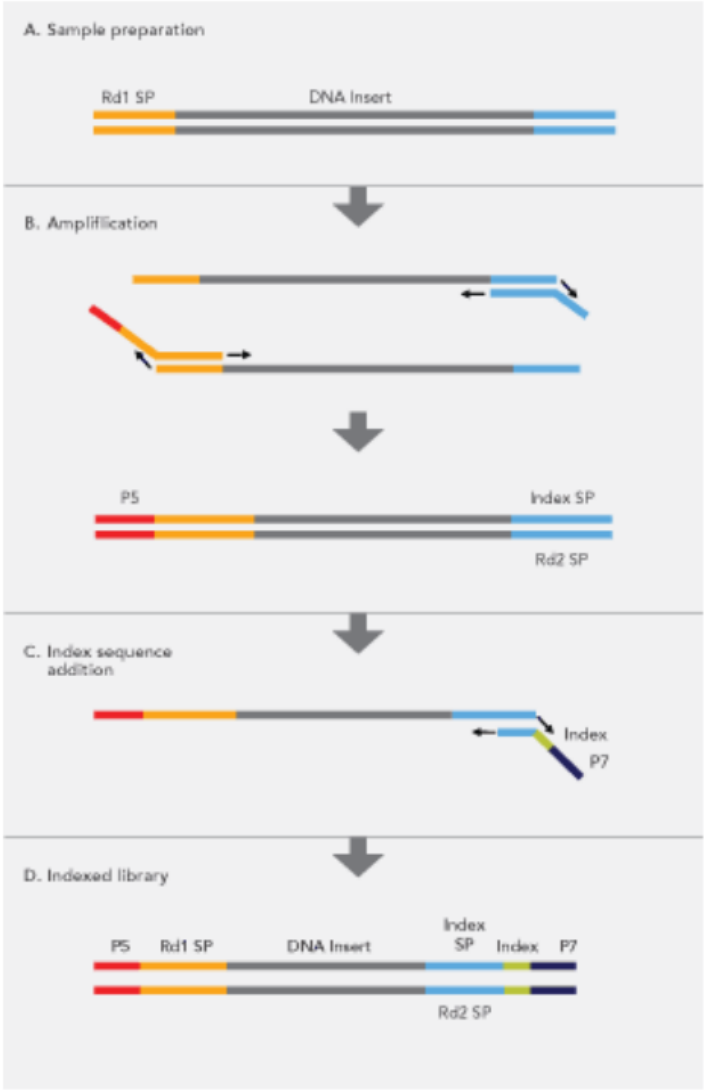
Data "grooming"



Read problems

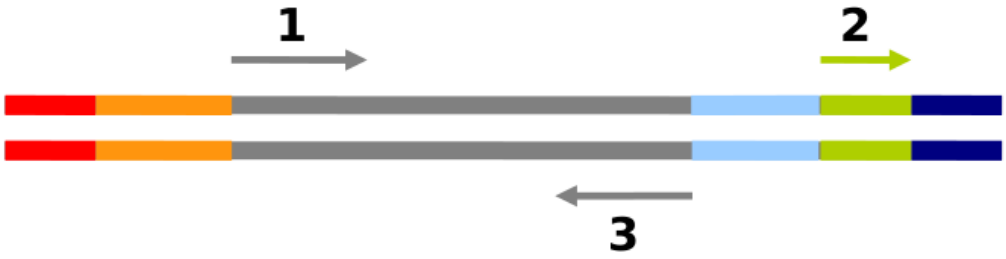
- Contaminating sequence *in reads*
  - adapters
- Poor quality sequence
  - substitution, indel errors
- Sample contamination
- Chimerism in library

Adapter contamination



illumina.com

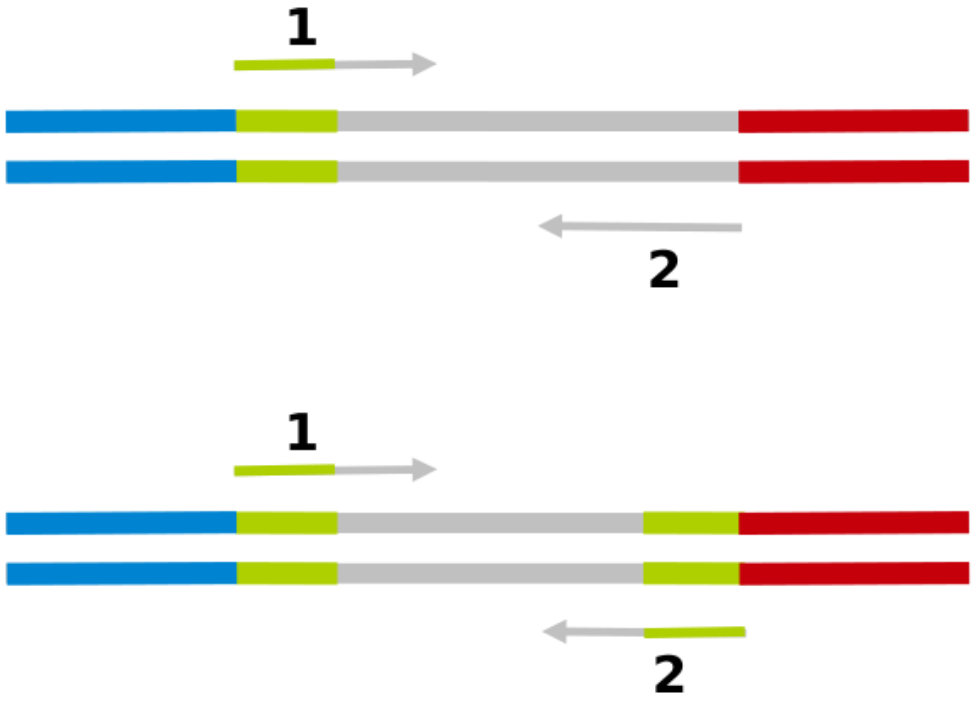
Reads 1 and 3 are "forward" and "reverse" reads from your DNA-of-interest, and they are on opposite strands.



Read (2) is the "barcode," which identifies particular reads as belonging to a particular sample.

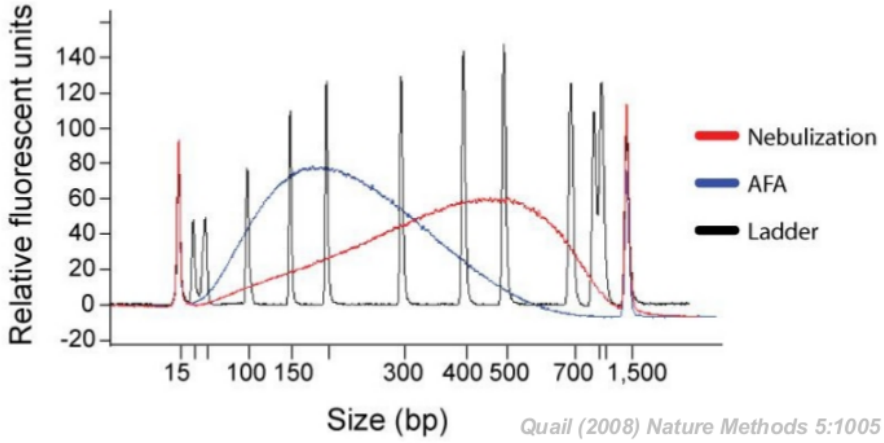
Adapter contamination

Older "in-line" or "homebrew" adapters can be added to one *or* both ends of DNA library fragments. Tools like *Sabre* (Nik Joshi) can recognize these, separate reads into different files, and remove barcode bases.



Adapter contamination

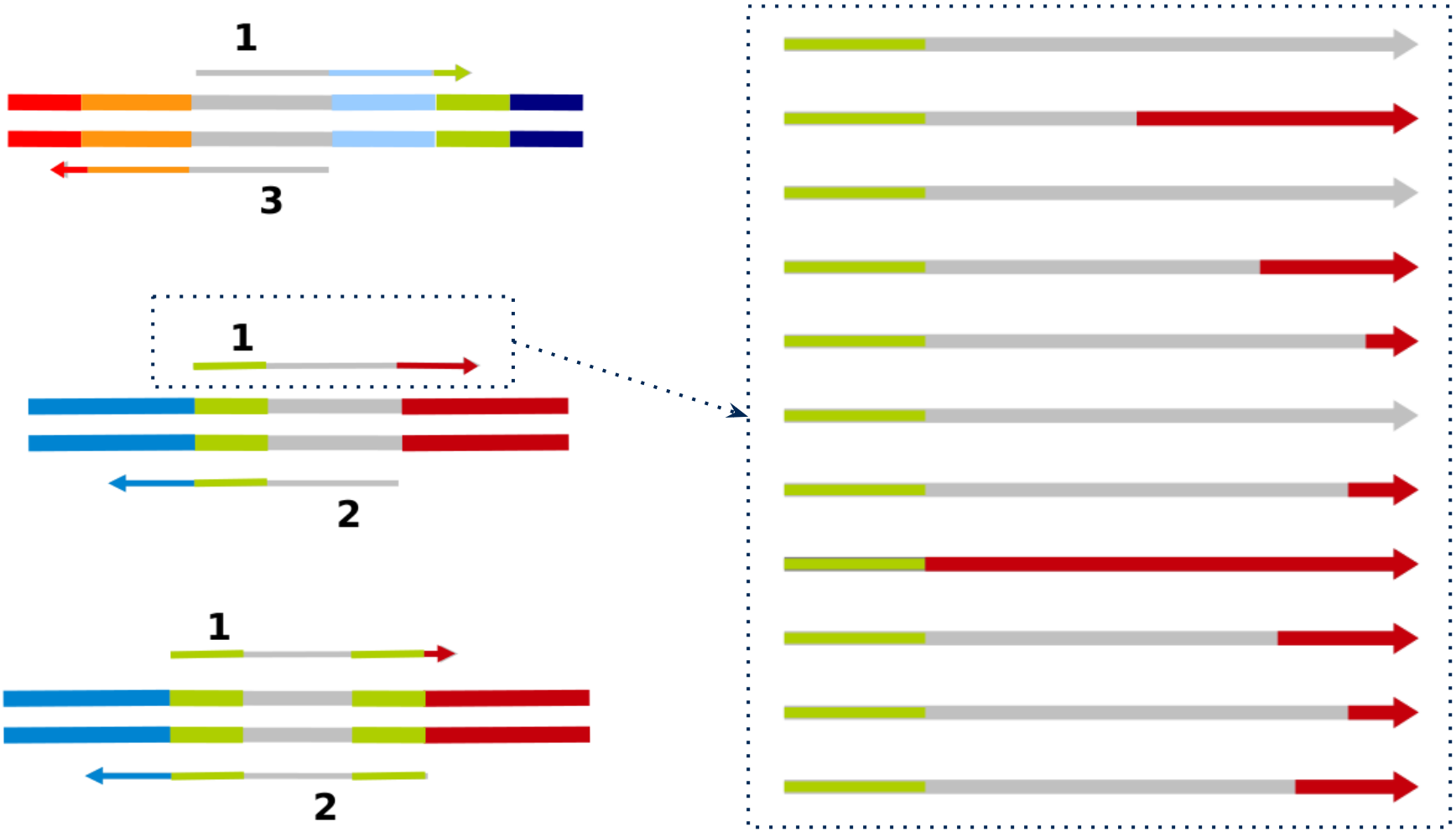
The problem is heterogeneous fragment sizes, resulting from any of the current library preparation techniques. All libraries will contain DNA fragments of variable size.



Quail (2008) Nature Methods 5:1005

Adapter contamination

Contamination is the result of the sequencer *reading through* a short read, into adapter sequence that *didn't come from your sample*!



Adapter contamination

Where can you find out adapter sequences?

- Google "github ucdavis-bioinformatics" → Scythe → "*_adapters.fa"
- Check Seqanswers.com
- Contact Illumina, PacBio, etc. for "tech notes" specifying the library prep primer / adapter sequences (not always that clear to work out).
- *Find them in your data*.

Base quality / FASTQ format

- Sanger-standard: ascii (phred + 33)
- Solexa: ~ascii (phred + 64)
- Illumina pipelines starting with v 1.3: ascii (phred + 64)
- Illumina pipelines starting with v 1.8: ascii (phred + 33)

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

$$\text{(probability of error)} = 10^{-(\text{phred score})/10}$$

Phred quality score	Probability that the base is called wrong	Accuracy of the base call
10	1 in 10	90%
20	1 in 100	99%
30	1 in 1,000	99.9%
40	1 in 10,000	99.99%
50	1 in 100,000	99.999%

Base quality / FASTQ format

ASCII Table

Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
32	20	040		Space	64	40	100	@	@	96	60	140	`	`
33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
39	27	047	'	'	71	47	107	G	G					
40	28	050	(	(	72	48	110	H	H	104				h
41	29	051	)	)	73	49	111	I	I					
42	2A	052	*	*	74	4A	112	J	J		6A	152	j	j
43	2B	053	+	+	75	4B	113	K	K		6B	153	k	k
44	2C	054	,	,	76	4C	114	L	L		6C	154	l	l
45	2D	055	-	-	77	4D	115	M	M		6D	155	m	m
46	2E	056	.	.	78	4E	116	N	N		6E	156	n	n
47	2F	057	/	/	79	4F	117	O	O		6F	157	o	o
48	30	060	0	0	80	50	120	P	P		70	160	p	p
49	31	061	1	1	81	51	121	Q	Q		71	161	q	q
50	32	062	2	2	82	52	122	R	R		72	162	r	r
51	33	063	3	3	83	53	123	S	S		73	163	s	s
52	34	064	4	4	84	54	124	T	T		74	164	t	t
53	35	065	5	5	85	55	125	U	U		75	165	u	u
54	36	066	6	6	86	56	126	V	V			166	v	v
55	37	067	7	7	87	57	127	W	W			167	w	w
56	38	070	8	8	88	58	130	X	X	120	78		x	x
57	39	071	9	9	89	59	131	Y	Y	121	79		y	y
58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172		z
59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	
60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

FASTQ

```
@SOLEXA2_0414:5:6:9490:4420#0/2
TGCAACTATGAGTCACGGCCACACCAGACCTTCCCATTGT
+SOLEXA2_0414:5:6:9490:4420#0/2
^SShfff^c\Y^a\^TJPZVb[`a_f^_ff\W\JS\JZJS
@SOLEXA2_0414:5:6:10399:4414#0/2
ACTAGCTAGTAGAACCCCCTACTCCACCATCCACTTCTTC
+SOLEXA2_0414:5:6:10399:4414#0/2
hhhhhhhShhaJ_]_fdJfSaba`b[ff^fZKS_R]SOIS
@SOLEXA2_0414:5:6:10756:4411#0/2
CAGTACTGTTGTAAGGTCTGGTTTGTCCTCGGCCACGGCG
+SOLEXA2_0414:5:6:10756:4411#0/2
ge]hga`Ve[\fcaURS\K[VaFTIVTV]TEMSRIVFZLL
```

Repeated blocks, four lines each:

1. … header, starting with "@"
2. … sequence
3. … header, starting with "+" (often left blank)
4. … base qualities (same length as sequence)

decimal value of character *related to* the quality / confidence of the basecall

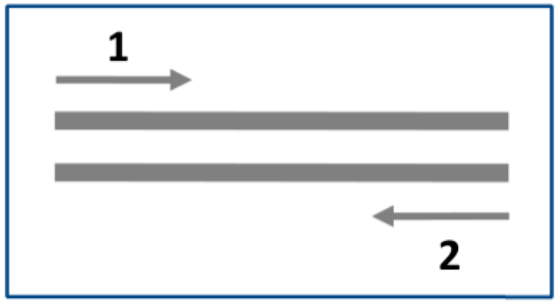
FASTQ - Pop Quiz!

1. What does a quality character of ";" mean?

2. In Sanger (standard) FASTQ, which ASCII character would I use to indicate that I'm absolutely sure that I'm wrong about a particular base?

3. If a particular 40 bp read from a run analyzed with Illumina Pipeline 1.6 (phred + 64) had consistent quality characters of "J", how many errors should you expect in the read?

FASTQ - Base order / read orientation



An "F/R" pair, or "innies"

1

```
@SOLEXA2_0414:3:1:19459:1418#0/1
NTGCGATCTCATGGACAAACCAGACCTTACAACTGTTACTCTGAATCTCCCGCAGTGTCCAAGGACAAACCGGACCTTACAACTG
+SOLEXA2_0414:3:1:19459:1418#0/1
BGGIFMRPOO__________P__T_YYYYRYYYYM[[[[[__Y_____________________W__V___WPWWPWXXVXTPVVV
@SOLEXA2_0414:3:1:19476:1420#0/1
NAGCAATTGTTTTTCTACATATTATTTGACATACTTCTATCTTCCCATGTTCTTACTATAGTTTGTGTATTGCCAAGTCTGTTGT
+SOLEXA2_0414:3:1:19476:1420#0/1
BGGIEKQIIH_QQQQQQ________YQQ[YYYYYY__________BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
```

2

```
@SOLEXA2_0414:3:1:19459:1418#0/2
NTNCGATGGAGATTCAGAGTAACAGTTGTAAGGTCCGGTTTGTCCTTGGACACTGCGGGAGATTCAGAGTAACAGTTGTAAGGTC
+SOLEXA2_0414:3:1:19459:1418#0/2
BJBHKPPPPPWTWWTYYYYYYW[MYYYYYY_____BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
@SOLEXA2_0414:3:1:19476:1420#0/2
NANCAATGCCACAAAATAAGTGCACTGTGCAATATCAAGTGTTTCAGCAAGTTTTTGGAATCTCAAGCACATTTTGACACCTTAT
+SOLEXA2_0414:3:1:19476:1420#0/2
BGBHGKKKJOYY[[YSSSSQYYY[YYYYUYYYYYV[WY[[_______________b______b__Y[YY[Y[[[[___Z_____^
```

Back to contamination / quality issues

```
@SOLEXA2_0414:5:2:10629:1818#0/1
AAGAACTTAACAGTTGTCAGGTCTGGTTTGTCCGAGTACTGAGATCGGAC
+SOLEXA2_0414:5:2:10629:1818#0/1
JJWJ\X_R\Zfffffffcccffcffk^\aa^fffcdfafffecfc]ffccb
@SOLEXA2_0414:5:2:17187:1812#0/1
CCTAGCTGCGCCGGACAGAACAGATCACAGCTCTCCAGGAATATTCGGCA
+SOLEXA2_0414:5:2:17187:1812#0/1
ccccScaJcXaBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
@SOLEXA2_0414:5:2:6078:1864#0/1
AAGAACTCTCTGCCTAGGACAAACCCGACAGTACTGAGATCGGAAGAGCG
+SOLEXA2_0414:5:2:6078:1864#0/1
ZSVPPZ\_\`ffffdKba^^ddddbIa\^OYNR_Wc^a^[RaaYYdOdbd
@SOLEXA2_0414:5:2:10727:1860#0/1
TGCCACTCCCCAGGCCACTGCGGGAGATTCCGCGTAACAGTTGTCAGATC
+SOLEXA2_0414:5:2:10727:1860#0/1
\W_V_W\`aaccc^ff]cdfffaa_eWeedUaIa^\a^MaW^a\JKSITV
@SOLEXA2_0414:5:2:12771:1881#0/1
AAGTACTCTAGGGACAACCCAGACCTTACACCTGTAGTACTGAGATCGGC
+SOLEXA2_0414:5:2:12771:1881#0/1
WGIOITTTNQ_____[YVYYYYYYaWWWWaKTQN[WWcWWbVcYc[OGZc
@SOLEXA2_0414:5:2:17271:1887#0/1
CCGAACCCAGTTGTAAGGTCTGGTTTGTCCAGTACTGAGATCGGAAGCGA
+SOLEXA2_0414:5:2:17271:1887#0/1
cXZRZXNXS]cccc_cccccccUcPUSS[]^O]]ccccccc_cbacbca
```

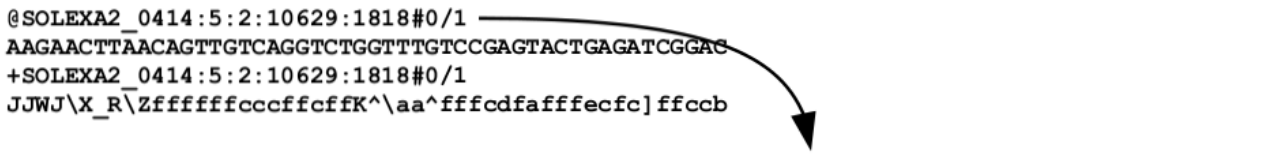
```
@SOLEXA2_0414:5:2:10629:1818#0/2
AAGAANNCNGACAANCCAGACCTNCCNACTGNGAAGTACTGAGATCGNAA
+SOLEXA2_0414:5:2:10629:1818#0/2
BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
@SOLEXA2_0414:5:2:17187:1812#0/2
CCTAGNNTAGCATCATGGGTTTAGACGATCCACTAGCCACAGCCTCCAGC
+SOLEXA2_0414:5:2:17187:1812#0/2
PVVS^BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
@SOLEXA2_0414:5:2:6078:1864#0/2
AAGAANNGNCTGGTNTNTCCTAGNCANAGAGNCCTGAGATCGGAAGANCG
+SOLEXA2_0414:5:2:6078:1864#0/2
BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
@SOLEXA2_0414:5:2:10727:1860#0/2
TGCAANNTNCTCTGNAACTCCCGNCGNGTCCNGGACAAACCAGACCTNAC
+SOLEXA2_0414:5:2:10727:1860#0/2
_^^ZaBBWBQXETXBKG[P[]_ZBKVBWJT\B\VXaa^^caLW`aKFBUV
@SOLEXA2_0414:5:2:12771:1881#0/2
CAGTANNACCGATGTACGGTCTGNTTNGTCCNTAGAGTACTGAGAACGGA
+SOLEXA2_0414:5:2:12771:1881#0/2
^MUXXBB^ZZVGSSWYJVZZZSTBJRB]VZYB_]^_`V``eeceKJ\_\U
@SOLEXA2_0414:5:2:17271:1887#0/2
CAGAANNGGCCCAACCCGACCTTACAACTGAGTACTGAGATCGGAAGCGC
+SOLEXA2_0414:5:2:17271:1887#0/2
YYSNXBBNZVZNWW[fcacf^\\SZTGRZNU\J_ZeeLed^dcaBBBBBB
```

Back to contamination / quality issues

```
@SOLEXA2_0414:5:2:10629:1818#0/1
AAGAACTTAACAGTTGTCAGGTCTGGTTTGTCCGAGTACTGAGATCGGAC
+SOLEXA2_0414:5:2:10629:1818#0/1
JJWJ\X_R\Zfffffcccffcffk^\aa^fffcdfafffecfc]ffccb
@SOLEXA2_0414:5:2:17187:1812#0/1
CCTAGCTGCGCCGGACAGAACAGATCACAGCTCTCCAGGAATATTCGGCA
+SOLEXA2_0414:5:2:17187:1812#0/1
ccccScaJcXaBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
@SOLEXA2_0414:5:2:6078:1864#0/1
AAGAACTCTCTGCCTAGGACAAACCCGACAGTACTGAGATCGGAAGAGCG
+SOLEXA2_0414:5:2:6078:1864#0/1
ZSVPPZ\_\`ffffdKba^^ddddbIa\^OYNR_Wc^a^[RaaYYdOdbd
@SOLEXA2_0414:5:2:10727:1860#0/1
TGCCACTCCCCAGGCCACTGCGGGAGATTCCGCGTAACAGTTGTCAGATC
+SOLEXA2_0414:5:2:10727:1860#0/1
\W_V_W\`aaccc^ff]cdfffaa_eWeedUaIa^\a^MaW^a\JKSITV
@SOLEXA2_0414:5:2:12771:1881#0/1
AAGTACTCTAGGGACAACCCAGACCTTACACCTGTAGTACTGAGATCGGC
+SOLEXA2_0414:5:2:12771:1881#0/1
WGIOITTTNQ_____[YVYYYYYYaWWWWaKTQN[WWcWWbVcYc[OGZc
@SOLEXA2_0414:5:2:17271:1887#0/1
CCGAACCCAGTTGTAAGGTCTGGTTTGTCCAGTACTGAGATCGGAAGCGA
+SOLEXA2_0414:5:2:17271:1887#0/1
cXZRZXNXS]cccc_cccccccUcPUSS[]^O]]ccccccc_cbacbca
```

```
@SOLEXA2_0414:5:2:10629:1818#0/2
AAGAANNCNGACAANCCAGACCTNCCNACTGNGAAGTACTGAGATCGNAA
+SOLEXA2_0414:5:2:10629:1818#0/2
BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
@SOLEXA2_0414:5:2:17187:1812#0/2
CCTAGNNTAGCATCATGGGTTTAGACGATCCACTAGCCACAGCCTCCAGC
+SOLEXA2_0414:5:2:17187:1812#0/2
PVVS^BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
@SOLEXA2_0414:5:2:6078:1864#0/2
AAGAANNGNCTGGTNTNTCCTAGNCANAGAGNCCTGAGATCGGAAGANCG
+SOLEXA2_0414:5:2:6078:1864#0/2
BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
@SOLEXA2_0414:5:2:10727:1860#0/2
TGCAANNTNCTCTGNAACTCCCGNCGNGTCCNGGACAAACCAGACCTNAC
+SOLEXA2_0414:5:2:10727:1860#0/2
_^^ZaBBWBQXETXBKG[P[]_ZBKVBWJT\B\VXaa^^caLW`aKFBUV
@SOLEXA2_0414:5:2:12771:1881#0/2
CAGTANNACCGATGTACGGTCTGNTTNGTCCNTAGAGTACTGAGAACGGA
+SOLEXA2_0414:5:2:12771:1881#0/2
^MUXXBB^ZZVGSSWYJVZZZSTBJRB]VZYB_]^_`V``eeceKJ\_\U
@SOLEXA2_0414:5:2:17271:1887#0/2
CAGAANNGGCCCAACCCGACCTTACAACTGAGTACTGAGATCGGAAGCGC
+SOLEXA2_0414:5:2:17271:1887#0/2
YYSNXBBNZVZNWW[fcacf^\\SZTGRZNU\J_ZeeLed^dcaBBBBBB
```

(side note - "file" issues)



Do your FASTQ files begin and end with the same IDs? Incomplete downloads, accidental sorting, different trimming, etc. can get your forward and reverse read files *out of sync* with each other.

File Formats

- FASTA
- FASTQ
- SFF (**S**tandard **F**lowgram **F**ormat)
- HDF5

File Formats

- Illumina
  - FASTQ, BAM
- 454, Ion Torrent(?)
  - SFF → FASTQ, FASTA + QUAL
  - sff_extract script, Roche's GS-Tools
- PacBio
  - HDF5 ("*.bas.h5") → FASTQ, FASTA + QUAL
  - Secondary_Analysis_Results ("filtered_subreads.fastq")

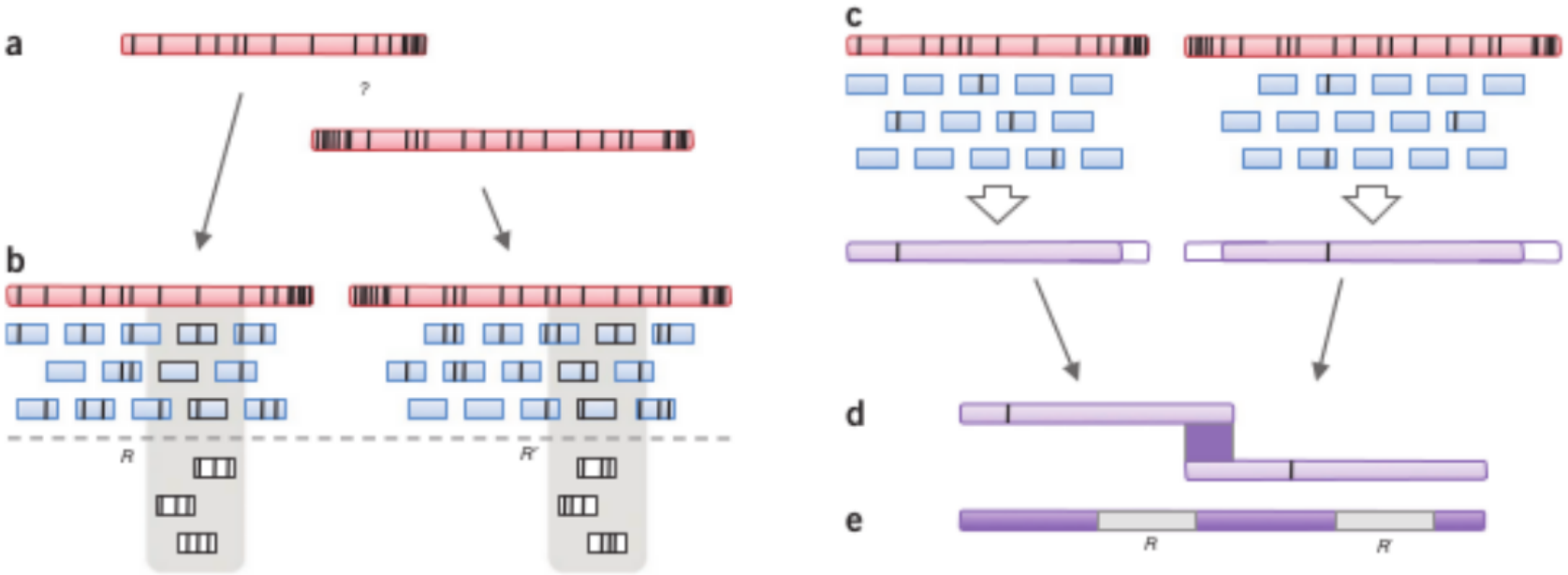
Error Correction

- Illumina
  - k-mer spectrum - based
    - Quake
    - CORAL
- PacBio
  - PBcR (Koren 2012 *Nature Biotech*)
  - ... in SMRT-Analysis tools

Error Correction (PBcR)

Koren (2012) "Hybrid error correction and *de novo* assembly of single-molecule sequencing reads" *Nature Biotechnology* 30:693

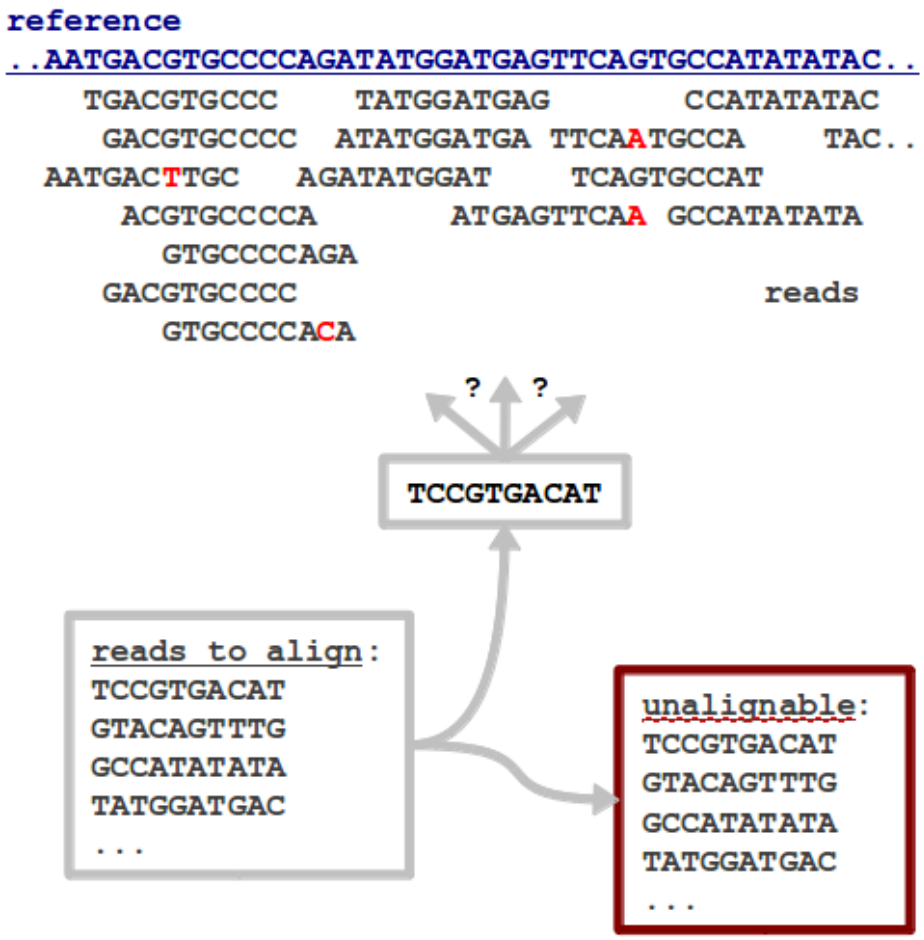
Implemented as part of CA (wgs-assembler.sourceforge.net/), can use either 454 or Illumina reads to correct PB reads.



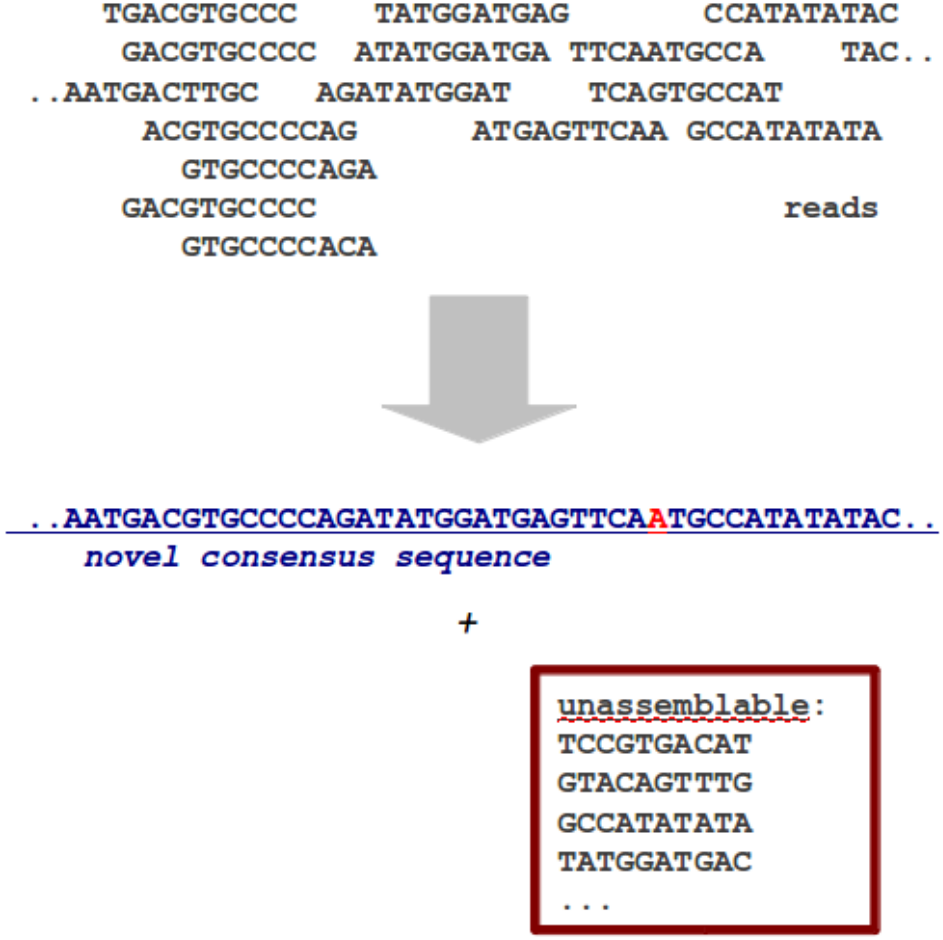
Applications

From reads to *molecules*

Alignment

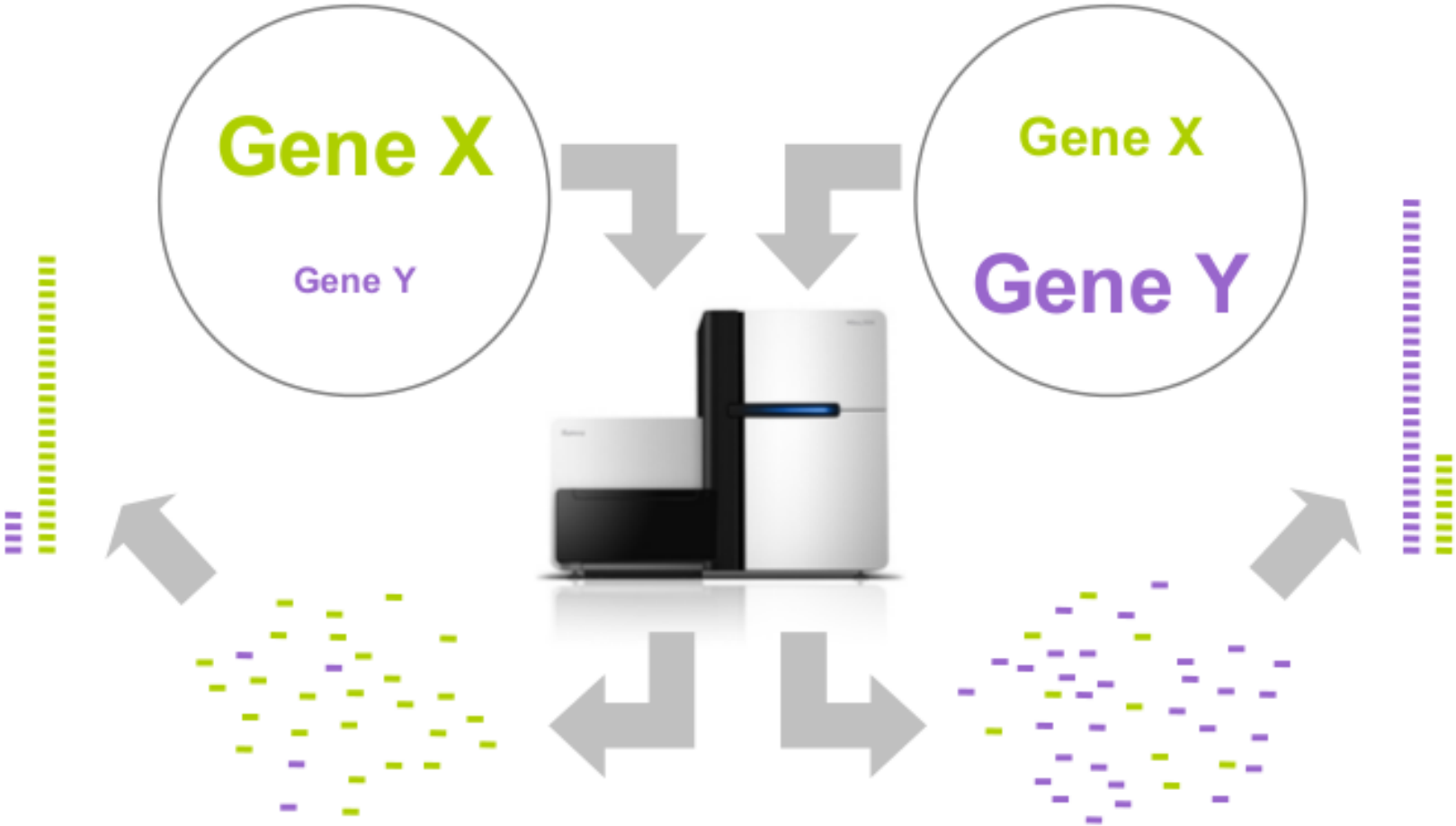


Assembly



Alignment

Why align?



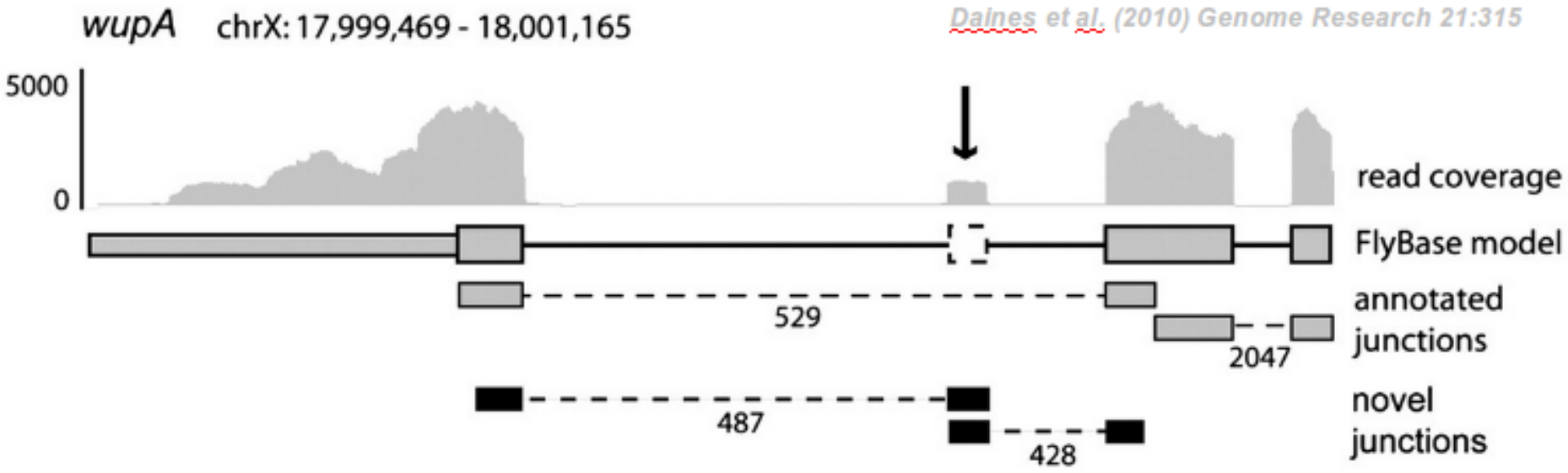
Why align?

Individual A → ← **Individual B**

```
ATGATAGCATCGTCGGGTGTCTGCTCAATAATAGTGCCGTATCATGCTGGTGTTATAATCGCCGCATGACATGATCAATGG
                                CAATAAAAGTGCCGTATCATGCTGGTGTTACAATCGCCGCA
                                          CGTATCATGCTGGTGTTACAATCGCCGCATGACATGATCAATGG
                             TGTCTGCTCAATAAAAGTGCCGTATCATGCTGGTGTTACAATC
                  ATCGTCGGGTGTCTGCTCAATAAAAGTGCCGTATCATG--GGTGTTATAA
                                CTCAATAAGAGTGCCGTATCATG--GGTGTTATAATCGCCGCA
                                                      GTTATAATCGCCGCATGACATGATCAATGG
```

To measure variation.

Why align?



To discover transcribed sequence.

Short Read Aligners: choices ...

Li, H and Homer, N (2010) Briefings in Bioinformatics 11:473
"A survey of sequence alignment algorithms for next-generation sequencing"

- **Gapped alignment** yields indels, and fewer FP SNP's!
- **Paired-end alignment** improves sensitivity
- Use of **base quality** could improve alignment, if qualities trusted

A survey of sequence alignment algorithms for next-generation sequencing

Table 1:
Popular short-read alignment software

Program	Algorithm	SOLiD	Long[a]	Gapped	PE[b]	Q[c]
Bfast	hashing ref.	Yes	No	Yes	Yes	No
Bowtie	FM-index	Yes	No	No	Yes	Yes
BWA	FM-index	Yes[d]	Yes[e]	Yes	Yes	No
MAQ	hashing reads	Yes	No	Yes[f]	Yes	Yes
Mosaik	hashing ref.	Yes	Yes	Yes	Yes	No
Novoalign[g]	hashing ref.	No	No	Yes	Yes	Yes

[a]Work well for Sanger and 454 reads, allowing gaps and clipping. [b]Paired end mapping. [c]Make use of base quality in alignment. [d]BWA trims the primer base and the first color for a color read. [e]Long-read alignment implemented in the BWA-SW module. [f]MAQ only does gapped alignment for Illumina paired-end reads. [g]Free executable for non-profit projects only.

fastest, at ~7 Gbp (vs human) per CPU day
... HiSeq 2500 generates 50-100 Gbp per day!*

Fall '12 - Apr '13: ... *now 150-180 Gbp / day!**

* *http://www.illumina.com/systems/hiseq_2500_1500/performance_specifications.ilmn*

Burrows-Wheeler Aligners

Burrows-Wheeler Transform used in bzip2 file compression tool; FM-index (Ferragina & Manzini) allow efficient finding of substring matches within compressed text – algorithm is *sub-linear* with respect to time and storage space required for a certain set of input data (reference 'ome, essentially).

Reduced memory footprint, faster execution.

BWA

BWA is fast, and can do gapped alignments. When run without seeding, it will find all hits within a given edit distance. Long read aligner is also fast, and can perform well for 454, Ion Torrent, Sanger, and PacBio reads. BWA is actively developed and has a strong user / developer community.

bio-bwa.sourceforge.net

Short reads – under 200 bp

Li H. and Durbin R. (2009) “Fast and accurate short read alignment with Burrows-Wheeler Transform.” Bioinformatics, 25:1754-60. [PMID: 19451168]

Long reads – over 200 bp … chimeric alignments built-in

Li H. and Durbin R. (2010) “Fast and accurate long read alignment with Burrows-Wheeler Transform.” Bioinformatics, 26:589-95. [PMID: 20080505]

… don't forget to join the mailing groups!

Bowtie

Bowtie (now Bowtie 2) is probably faster than BWA for some types of alignment, but it may not find the best alignments (see discussions on sensitivity, accuracy on SeqAnswers.com).

Bowtie is part of a suite of tools (Bowtie, Tophat, Cufflinks, Crossbow, Myrna) that address various alignment, RNAseq and genomic sequencing applications. Many of these tools are are already incorporated into the main Galaxy (Penn State / Emory).

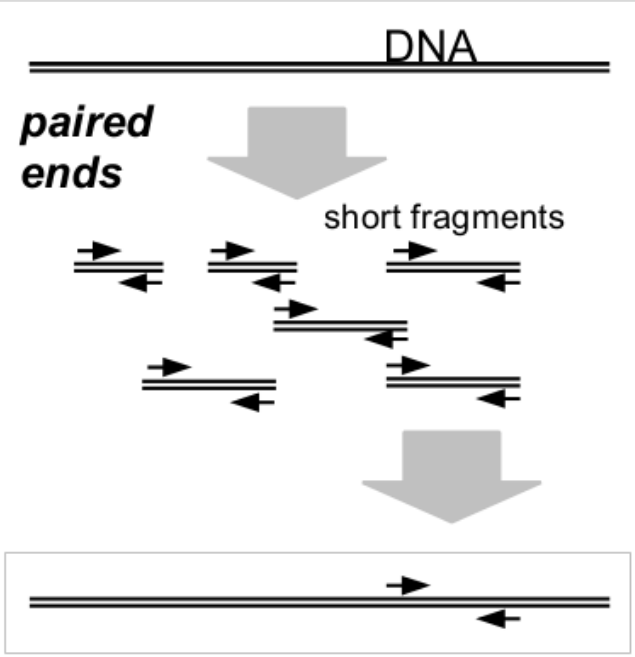
http://bowtie-bio.sourceforge.net

Langmead B., Trapnell C., Pop M., and Salzberg S.L. (2009) “Ultrafast and memory-efficient alignment of short DNA sequences to the human genome” Genome Biology 10:R25 [PMID: 19261174]

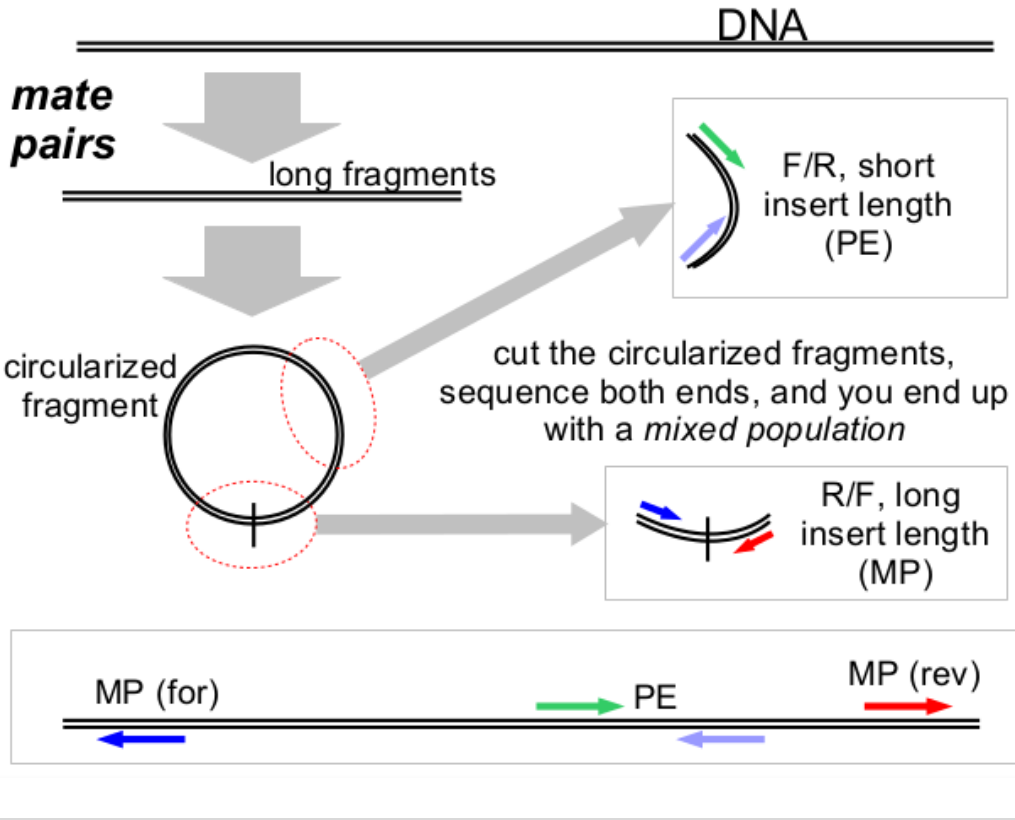
… don't forget to join the mailing groups!

Alignment concepts / parameters

Paired-End reads

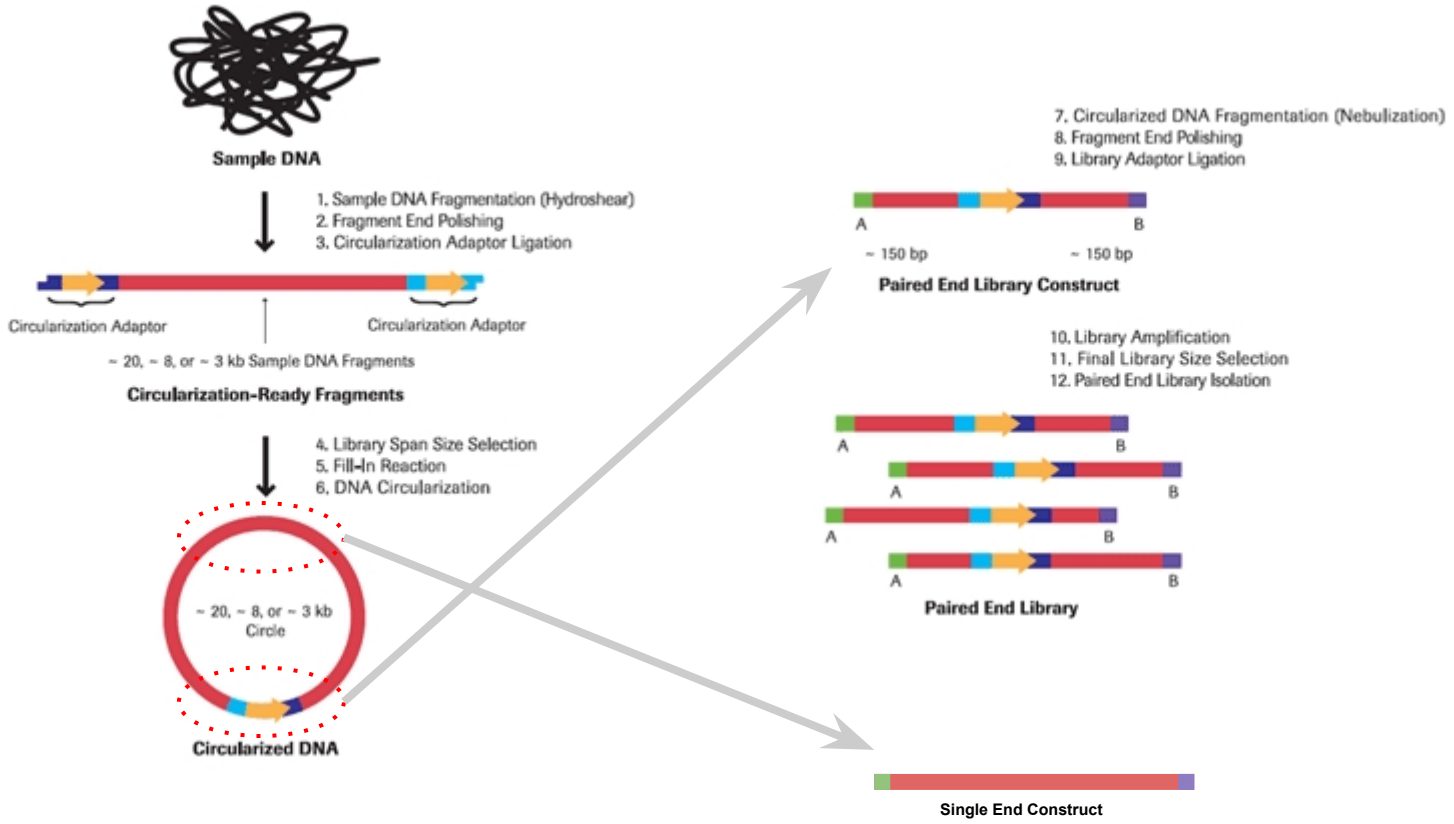


Mate-Paired reads



Alignment concepts / parameters

454 "Paired-End" reads



Alignment concepts / parameters

Edit Distance:

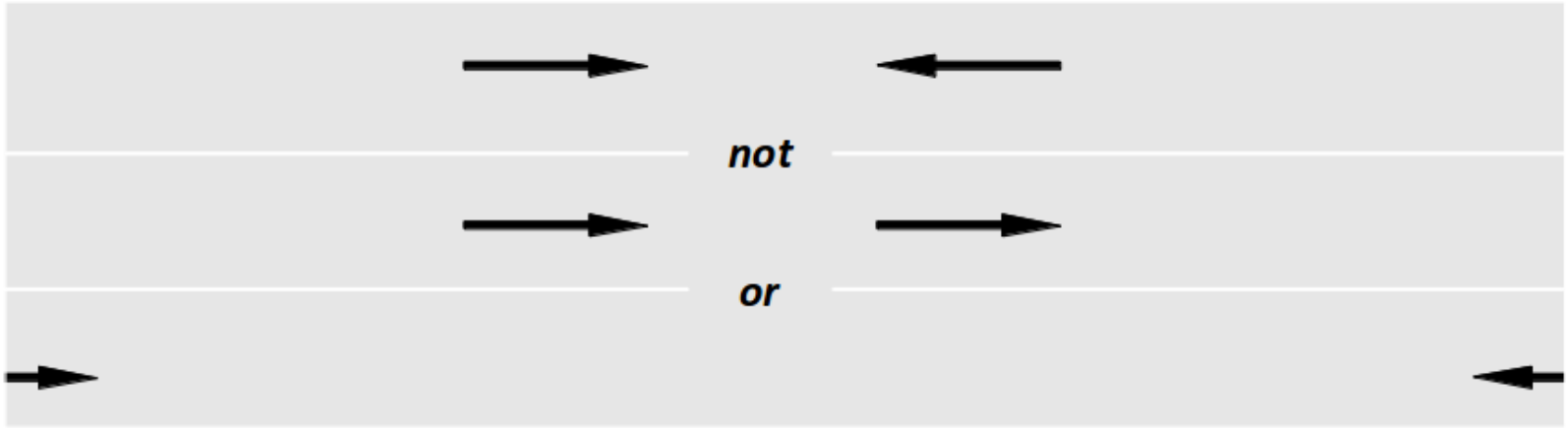
```
ATCGACCGCGCTAA-TATTAGTC . . .
   CGACGGCGCTAACTATTA
```

edit distance = 2

Mapping Quality:

prob. of incorrect position = $10^{-MQ/10}$ ... *(BWA)*

Proper Pairs:



Alignment concepts / parameters

Insert Size:

Insert Size (ISIZE)

Think: "inserted" between adapter sequences.

Inner Distance:

Inner Distance

Think: what's the length of the "inner" part of my DNA fragment, not accessed directly by reads?

Alignment concepts / parameters

Multimappers:

Reads that align *equally well* to more than one reference location.

Generally, multimappers are discounted in variant detection, and often discounted (ignored) in counting applications (RNA-Seq).

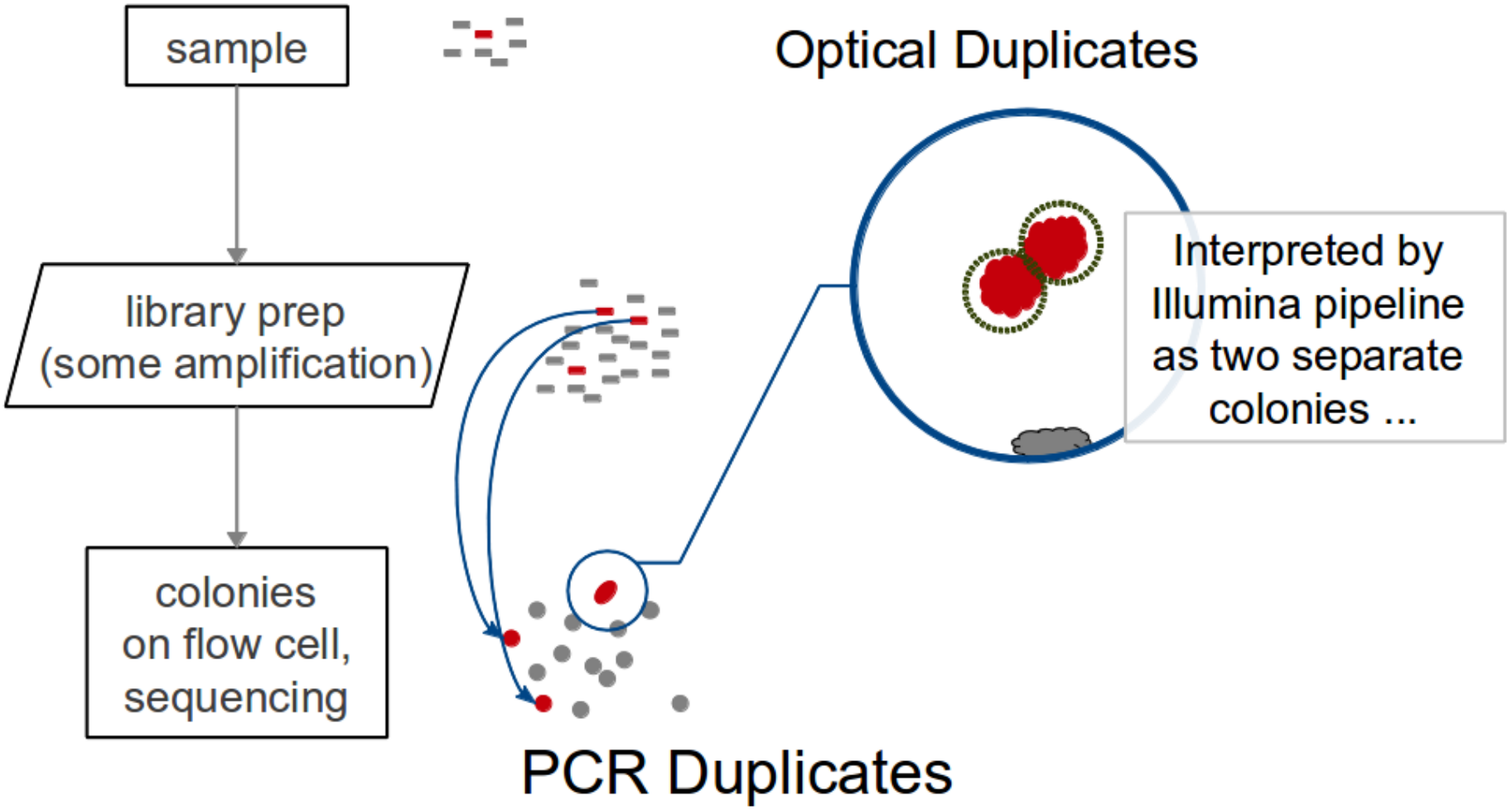
Note: *multimapper "rescue"* in some algorithms.

Duplicates:

Reads or read pairs arising from the same library fragment, either during library preparation (PCR duplicates) or colony formation (optical duplicates).

Generally, duplicates are discounted in variant detection , and ignored in counting applications (RNA-Seq).

Alignment concepts / parameters



File Format: SAM / BAM

http://samtools.sourceforge.net/

Li H.*, Handsaker B.*, Wysoker A., Fennell T., Ruan J., Homer N., Marth G., Abecasis G., Durbin R. and 1000 Genome Project Data Processing Subgroup (2009) The **Sequence alignment/map** (SAM) format and SAMtools. Bioinformatics, 25, 2078-9. [PMID: 19505943]

- SAM specification (currently v1.4)
- `samtools` man page
- FAQ
- *mailing list!*

File Format: SAM / BAM

```
@SQ     SN:chr18        LN:78077248
@SQ     SN:chr20        LN:63025520
@SQ     SN:chrY LN:59373566
@SQ     SN:chr19        LN:59128983
@SQ     SN:chr22        LN:51304566
@SQ     SN:chr21        LN:48129895
SOLEXA1:7:1:2:1304#0    4       *       0       0       *       *       0       0       ACAGTTGTAAGG
TCTGGTTTGTCCTTGTTGGTTGG     BCBCBCCCBACCBBCCCBCCCBB?7@9+8>0@;;5
SOLEXA1:7:1:2:1626#0    16      chr14   74750889        0       28M     *       0       0       CGCC
TCCTGGGTTCAAGCGATTCTCCAG    >??A?BB?ABA;@ABBAABBAABABBBB    XT:A:R  NM:i:0  XO:i:84 XM:i:0  XO:i:0
XG:i:0  MD:Z:28
SOLEXA1:7:1:3:83#0      16      chr11   118359388       37      51M     *       0       0       GCGC
CCTCTGGAGGACCAGCTGGAAAATTGGTGTTGTCGTCGTTGCAAATT     787844/>88786=4=7<=?6=7=86<@AB@@@@;?=>@=@A=BBBBA
ABA     XT:A:U  NM:i:0  XO:i:1  X1:i:0  XM:i:0  XO:i:0  XG:i:0  MD:Z:51
SOLEXA1:7:1:3:881#0     16      chr11   118358946       37      28M     *       0       0       GAAA
GGACAAACCAGACCTTACAACTGT    =CCA:ACCC@=ACCAAB@2>ACCC A5@1    XT:A:U  NM:i:0  XO:i:1  X1:i:0  XM:i:0
XO:i:0  XG:i:0  MD:Z:28
SOLEXA1:7:1:3:1858#0    0       chr11   118359143       37      61M     *       0       0       TACT
CTGAATCTCCCGCAGTGTCCAATACTGTACTTTTTTACATAGTCATTGCTTAATGAA   BBBCBCCBBCBBBBBCBBCBCBBBBBBB@BB@BBBBCBBB
BBBBAB=??ABAAABA<A?99   XT:A:U  NM:i:0  XO:i:1  X1:i:0  XM:i:0  XO:i:0  XG:i:0  MD:Z:61
```

one tab-delimited line per alignment

aligned to ...

CIGAR strings *and* ELAND-like MD-tags! More alignment formats means *more fun* ...

fields for bases and base qualities, so SAM / BAM can also be used to store unaligned reads

...

File Format: SAM / BAM

Col	Field	Type	Regexp/Range	Brief description
1	QNAME	String	`[!-?A-~]{1,255}`	Query template NAME
2	FLAG	Int	`[0,2^16-1]`	bitwise FLAG
3	RNAME	String	`\*|[!-()+-<>-~][!-~]*`	Reference sequence NAME
4	POS	Int	`[0,2^29-1]`	1-based leftmost mapping POSition
5	MAPQ	Int	`[0,2^8-1]`	MAPping Quality
6	CIGAR	String	`\*|([0-9]+[MIDNSHPX=])+`	CIGAR string
7	RNEXT	String	`\*|=|[!-()+-<>-~][!-~]*`	Ref. name of the mate/next 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
10	SEQ	String	`\*|[A-Za-z=.]+`	segment SEQuence
11	QUAL	String	`[!-~]+`	ASCII of Phred-scaled base QUALity+33

SAM Format Specification v1.4-r985

7,8 - formerly MRNM, MPOS (mate reference name, mate position)

9 - formerly ISIZE ("insert" size)

@SQ	name	LN:		
@...				
readName1	flag	referenceName	...	
readName2	...			
...				

header

alignments:
(one line per *alignment*)

```
@SQ     SN:chr18        LN:78077248
@SQ     SN:chr20        LN:63025520
@SQ     SN:chrY LN:59373566
@SQ     SN:chr19        LN:59128983
@SQ     SN:chr22        LN:51304566
@SQ     SN:chr21        LN:48129895
SOLEXA1:7:1:2:1304#0    4       *       0       0       *       *       0       0       ACAGTTGTAAGG
TCTGGTTTGTCCTTGTTGGTTGG     BCBCBCCCBACCBBCCCBCCCBB?7@9+8>0@;;5
SOLEXA1:7:1:2:1626#0    16      chr14   74750889        0       28M     *       0       0       CGCC
TCCTGGGTTCAAGCGATTCTCCAG    >??A?BB?ABA;@ABBAABBAABABBBB    XT:A:R  NM:i:0  XO:i:84 XM:i:0  XO:i:0
XG:i:0  MD:Z:28
SOLEXA1:7:1:3:83#0      16      chr11   118359388       37      51M     *       0       0       GCGC
CCTCTGGAGGACCAGCTGGAAAATTGGTGTTGTCGTCGTTGCAAATT     787844/>88786=4=7<=?6=7=86<@AB@@@@;?=>@=@A=BBBBA
ABA     XT:A:U  NM:i:0  XO:i:1  X1:i:0  XM:i:0  XO:i:0  XG:i:0  MD:Z:51
SOLEXA1:7:1:3:881#0     16      chr11   118358946       37      28M     *       0       0       GAAA
GGACAAACCAGACCTTACAACTGT    =CCA:ACCC@=ACCAAB@2>ACCCA5@1    XT:A:U  NM:i:0  XO:i:1  X1:i:0  XM:i:0
XO:i:0  XG:i:0  MD:Z:28
SOLEXA1:7:1:3:1858#0    0       chr11   118359143       37      61M     *       0       0       TACT
CTGAATCTCCCGCAGTGTCCAATACTGTACTTTTTTACATAGTCATTGCTTAATGAA   BBBCBCCBBCBBBBBCBBCBCBBBBBBB@BB@BBBBCBBB
BBBBAB=??ABAAABA<A?99   XT:A:U  NM:i:0  XO:i:1  X1:i:0  XM:i:0  XO:i:0  XG:i:0  MD:Z:61
```

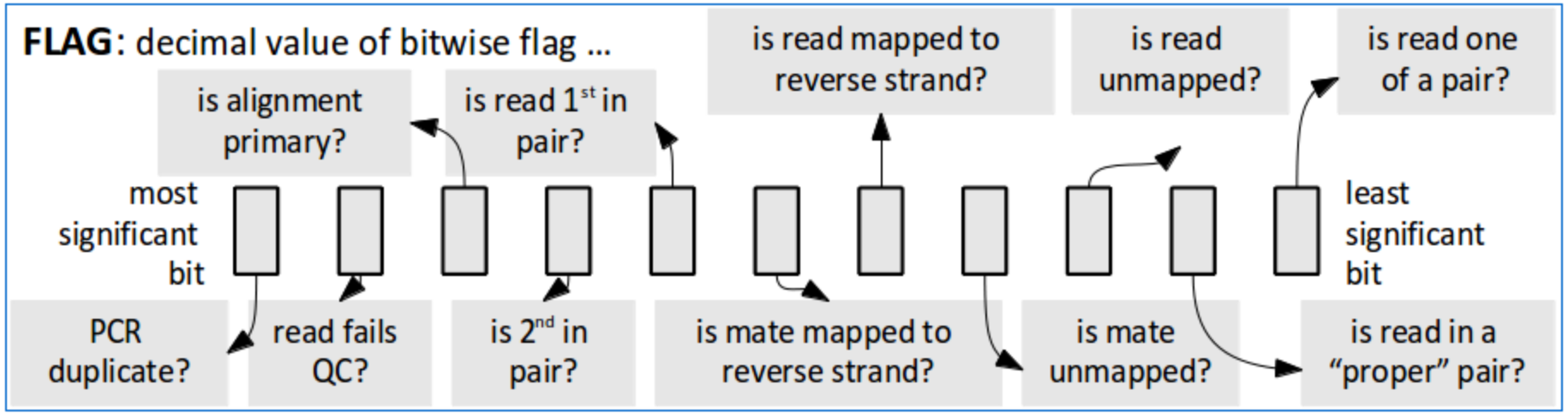
```
@SQ     SN:chr22        LN:51304566
@SQ     SN:chr21        LN:48129895
SOLEXA1:7:1:2:1304#0    4       *       0       0       *       *       0       0       ACAGTTGTAAGG
TCTGGTTTGTCCTTGTTGGTTGG     BCBCBCCCBACCBBCCCBCCCBB?7@9+8>0@;;5
SOLEXA1:7:1:2:1626#0    16      chr14   74750889        0       28M     *       0       0       CGCC
TCCTGGGTTCAAGCGATTCTCCAG    >??A?BB?ABA;@ABBAABBAABABBBB    XT:A:R  NM:i:0  XO:i:84 XM:i:0  XO:i:0
XG:i:0  MD:Z:28
SOLEXA1:7:1:3:83#0      16      chr11   118359388       37      51M     *       0       0       GCGC
CCTCTGGAGGACCAGCTGGAAAATTGGTGTTGTCGTCGTTGCAAATT     787844/>88786=4=7<=?6=7=86<@AB@@@@;?=>@=@A=BBBBA
ABA     XT:A:U  NM:i:0  XO:i:1  X1:i:0  XM:i:0  XO:i:0  XG:i:0  MD:Z:51
```

```
SOLEXA1:7:1:2:1626#0
16
chr14
74750889
0
28M
*
0
0
CGCCTCCTGGGTTCAAGCGATTCTCCAG
>??A?BB?ABA;@ABBAABBAABABBBB
XT:A:R  NM:i:0  XO:i:84 XM:i:0  XO:i:0  XG:i:0  MD:Z:28
```

QNAME: Query name

Note that, for Illumina paired-end reads, with format:
ILLUMINA-runID:lane:tile:X:Y#0/1 *(for forward reads)*
ILLUMINA-runID:lane:tile:X:Y#0/2 *(for reverse reads)*
… the "/1" or "/2" are stripped. The QNAME thus becomes *non-unique*, and the only way to figure out if this read is one of a pair is from the next field: FLAG

```
SOLEXA1:7:1:2:1626#0
16
chr14
74750889
0
28M
*
0
0
CGCCTCCTGGGTTCAAGCGATTCTCCAG
>??A?BB?ABA;@ABBAABBAABABBBB
XT:A:R  NM:i:0  X0:i:84 XM:i:0  XO:i:0  XG:i:0  MD:Z:28
```




```
SOLEXA1:7:1:2:1626#0
16
chr14
74750889
0
28M
*
0
0
CGCCTCCTGGGTTCAAGCGATTCTCCAG
>??A?BB?ABA;@ABBAABBAABABBBB
XT:A:R  NM:i:0  X0:i:84 XM:i:0  XO:i:0  XG:i:0  MD:Z:28
```

16 (decimal) = 0000010000 (binary)

so ... (from right to left): read is *not* paired, read is *not* in a pair – proper or otherwise (so bit is off), read is *not* unmapped, mate doesn't exist (so bit is off), read *is* mapped to reverse strand

FLAG: still confused?

picard.sourceforge.net/explain-flags.html

```
SOLEXA1:7:1:2:1626#0
16
chr14
74750889
0
28M
*
0
0
CGCCTCCTGGGTTCAAGCGATTCTCCAG
>??A?BB?ABA;@ABBAABBAABABBBB
XT:A:R  NM:i:0  X0:i:84 XM:i:0  XO:i:0  XG:i:0  MD:Z:28
```

FLAGs in the wild …

Single End reads:

- 0 and 16, or 4
- 20 (??!)

Paired End reads:

- 83 / 163
- 99 / 147
- 77 / 141
- 65 / 129
- 81 / 161

```
65 :
        read paired
        first in pair
129 :
        read paired
        second in pair

ILLUMINA-433A7D_0001:7:1:1078:6092#0    129     scaffold_1
ILLUMINA-433A7D_0001:7:1:1078:17864#0   83      scaffold_25
```

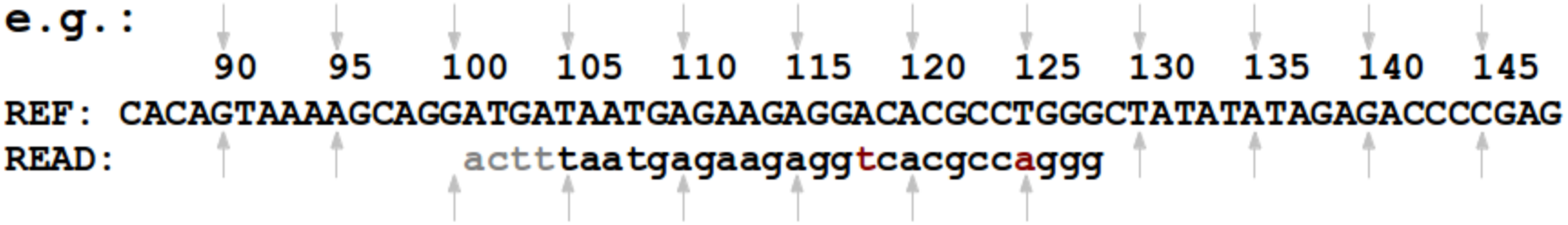
RNAME: reference sequence name

* Watch out! For both QNAME and RNAME, bwa will truncate fastq header text at first whitespace … so:
>chromosome 1
becomes
>chromosome (!)

```
SOLEXA1:7:1:2:1626#0
16
chr14
74750889
0
28M
*
0
0
CGCCTCCTGGGTTCAAGCGATTCTCCAG
>??A?BB?ABA;@ABBAABBAABABBBB
XT:A:R  NM:i:0  X0:i:84 XM:i:0  XO:i:0  XG:i:0  MD:Z:28
```

POS: 1-based leftmost position of (post-clipping) aligned sequence

e.g.:



For the read above, mapped to reverse strand, with last 4 bases clipped for quality, and two mismatches … value of POS field would be 105.

```
SOLEXA1:7:1:2:1626#0
16
chr14
74750889
0
28M
*
0
0
CGCCTCCTGGGTTCAAGCGATTCTCCAG
>??A?BB?ABA;@ABBAABBAABABBBB
XT:A:R  NM:i:0  X0:i:84 XM:i:0  XO:i:0  XG:i:0  MD:Z:28
```

MAPQ : mapping quality (Phred-scaled)

Mapping quality is a function of the edit distance (mismatches, indels), and the uniqueness of the alignment. Multiple equivalent best alignments yield a mapping quality of zero; alignments with an edit distance close to the best alignment lower the mapping quality.
Note that currently the read's base qualities *do not* affect the mapping quality.

```
SOLEXA1:7:1:2:1626#0
16
chr14
74750889
0
28M
*
0
0
CGCCTCCTGGGTTCAAGCGATTCTCCAG
>??A?BB?ABA;@ABBAABBAABABBBB
XT:A:R  NM:i:0  X0:i:84 XM:i:0  XO:i:0  XG:i:0  MD:Z:28
```

CIGAR : extended CIGAR string (Compact Idiosyncratic Gapped Alignment Report)
Format: [0-9][MIDNSHP][0-9][MIDNSHP]...
M = match *or* mismatch (?!), I/D = insertion / deletion, N = skipped bases on reference, S/H = soft / hard clip (soft means nt's still appear in sequence field), P = padding
e.g.: ”1S81M” means that the first (5'-most) nt is not part of the alignment, but the following 81 nt's are either matches or mis-matches.

```
SOLEXA1:7:1:2:1626#0
16
chr14
74750889
0
28M
*
0
0
CGCCTCCTGGGTTCAAGCGATTCTCCAG
>??A?BB?ABA;@ABBAABBAABABBBB
XT:A:R  NM:i:0  X0:i:84 XM:i:0  XO:i:0  XG:i:0  MD:Z:28
```

MRNM : reference sequence *to which the mate of this read is mapped*

"=" means the mate is mapped to the same reference sequence as the current read.

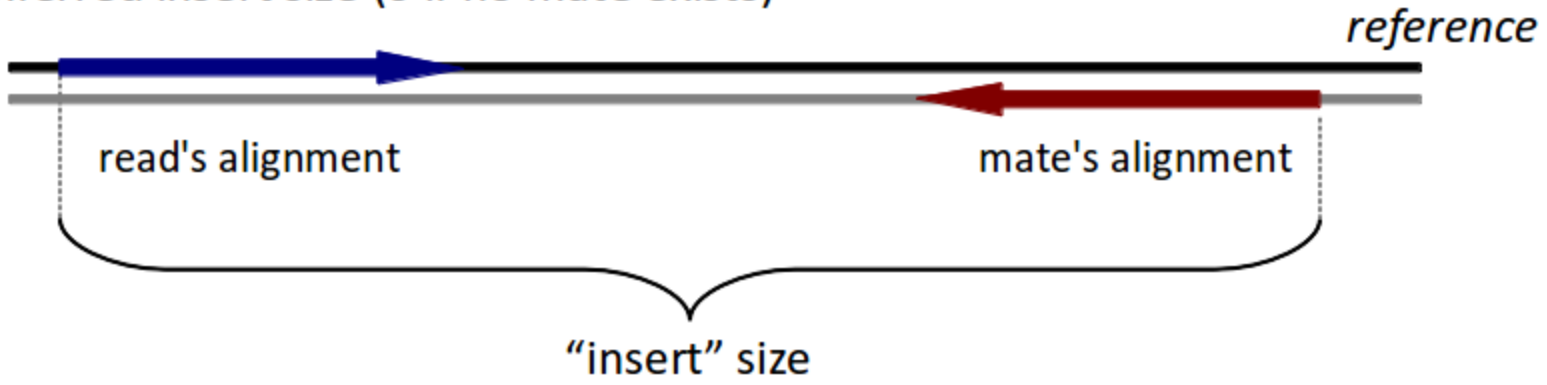
"*" means that the read is unpaired (has no mate)

```
SOLEXA1:7:1:2:1626#0
16
chr14
74750889
0
28M
*
0
0
CGCCTCCTGGGTTCAAGCGATTCTCCAG
>??A?BB?ABA;@ABBAABBAABABBBB
XT:A:R  NM:i:0  X0:i:84 XM:i:0  XO:i:0  XG:i:0  MD:Z:28
```

MPOS : 1-based, left-most position of 1st (post-clipping) nt of mate on the reference to which it's aligned (0 if no mate exists).

```
SOLEXA1:7:1:2:1626#0
16
chr14
74750889
0
28M
*
0
0
CGCCTCCTGGGTTCAAGCGATTCTCCAG
>??A?BB?ABA;@ABBAABBAABABBBB
XT:A:R  NM:i:0  X0:i:84 XM:i:0  XO:i:0  XG:i:0  MD:Z:28
```

ISIZE : inferred insert size (0 if no mate exists)




```
SOLEXA1:7:1:2:1626#0
16
chr14
74750889
0
28M
*
0
0
CGCCTCCTGGGTTCAAGCGATTCTCCAG
>??A?BB?ABA;@ABBAABBAABABBBB
XT:A:R  NM:i:0  X0:i:84 XM:i:0  XO:i:0  XG:i:0  MD:Z:28
```

SEQ and **QUAL**: read's nt's and base qualities, *as mapped to reference (forward) strand!* … (not including indels)

Thus, a read with nt's: "AAACATAGTTTGAGAAGAC", mapped to the reverse strand, would appear in the SEQ field as:
GTCTTCTCAAACTATGTTT
(with base qualities reversed as well)

```
SOLEXA1:7:1:2:1626#0
16
chr14
74750889
0
28M
*
0
0
CGCCTCCTGGGTTCAAGCGATTCTCCAG
>??A?BB?ABA;@ABBAABBAABABBBB
XT:A:R  NM:i:0  X0:i:84 XM:i:0  XO:i:0  XG:i:0  MD:Z:28
```

OPT : various pre-defined and user-defined tags in the format TAG:VTYPE:VALUE …

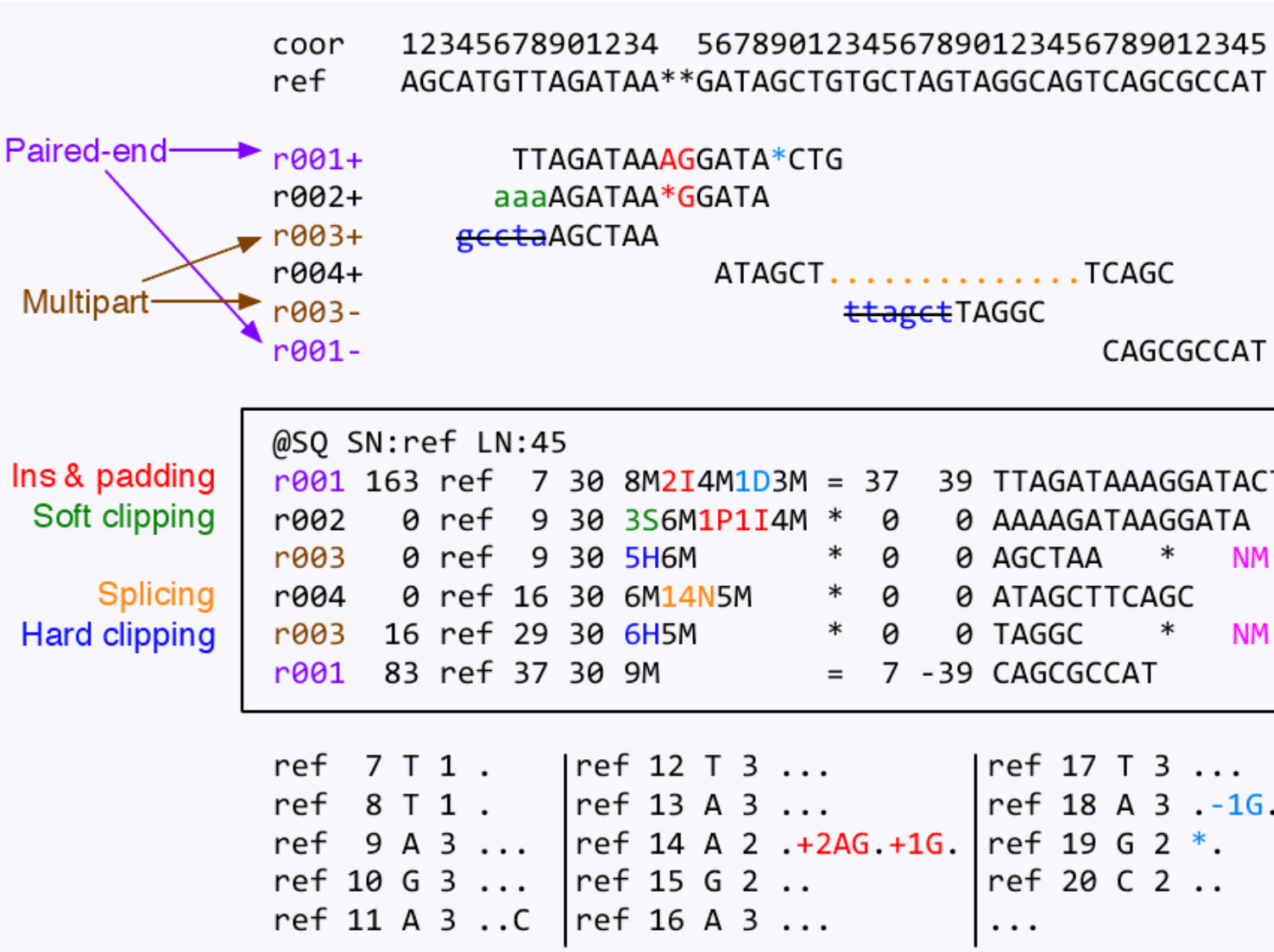
VTYPE is [A(printable character); i(signed integer); f(floating point); z(printable string); H(hex string)]

e.g.: RG:z:454 (read group 454 … commonly used for Roche 454 reads), NM:i:3 (edit distance of 3), MD:Z:22C0A22A34 (mismatching positions, in ELAND-esque format), X0:i:0 (number of zero mismatch alignments)

X[?] tags are user-defined, but many in use already

```
SOLEXA1:7:1:2:1626#0
16
chr14
74750889
0
28M
*
0
0
CGCCTCCTGGGTTCAAGCGATTCTCCAG
>??A?BB?ABA;@ABBAABBAABABBBB
XT:A:R  NM:i:0  X0:i:84 XM:i:0  XO:i:0  XG:i:0  MD:Z:28
```

File Format: SAM / BAM



google "Heng Li slides" - Challenges and Solutions in the Analysis of Next Generation Sequencing Data (2010)

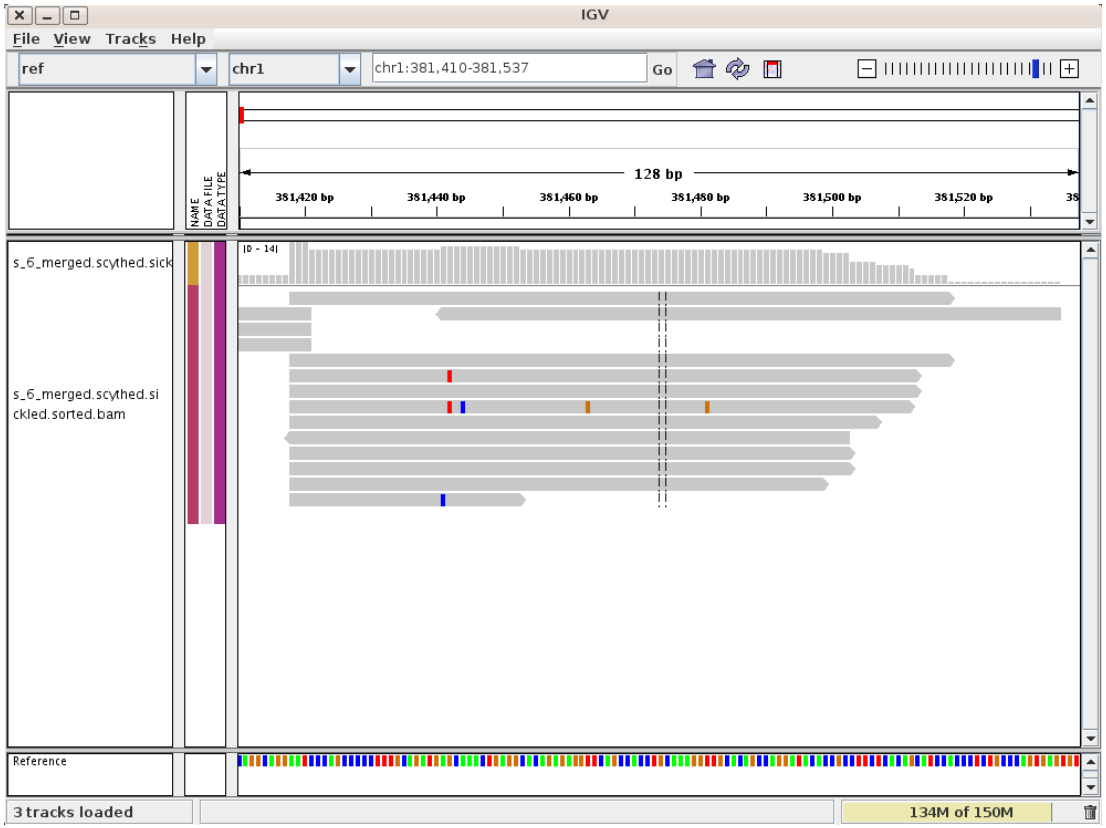
File Format: SAM / BAM

BAMs are compressed SAMs (so, binary, not human-readable text … don't look directly at them!). They can be indexed to allow rapid extraction of information, so alignment viewers do not need to uncompress the whole BAM file in order to look at information for a particular read or coordinate range, somewhere in the file.

Indexing your BAM file, myCoolBamFile.bam, will create an index file, myCoolBamFile.bam.bai, which is needed (in addition to the BAM file) by viewers and other downstream tools. An occasional downstream tool will require an index called myCoolBamFile.bai (notice that the ".bai" replaces the ".bam", instead of being appended after it).

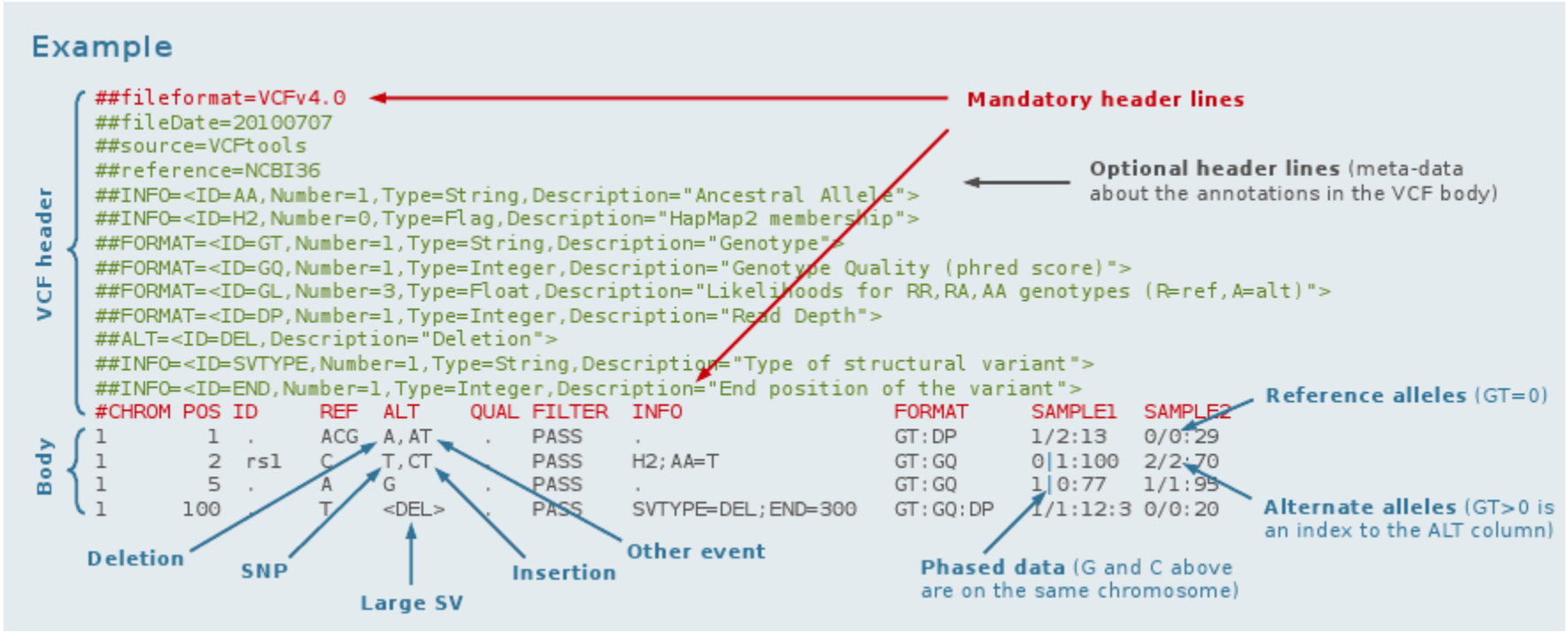
Alignment Viewers

- IGV (Integrated Genomics Viewer)
  - www.broadinstitute.org/igv/
- BAMview, tview (in SAMtools), IGB, GenomeView, SAMscope ...
- UCSC Genome Browser, GBrowse



Variant Calling

One main application of read alignment. A.k.a. "resequencing", SNP / indel discovery. VCF (variant call format) is now the standard format for variant reporting.



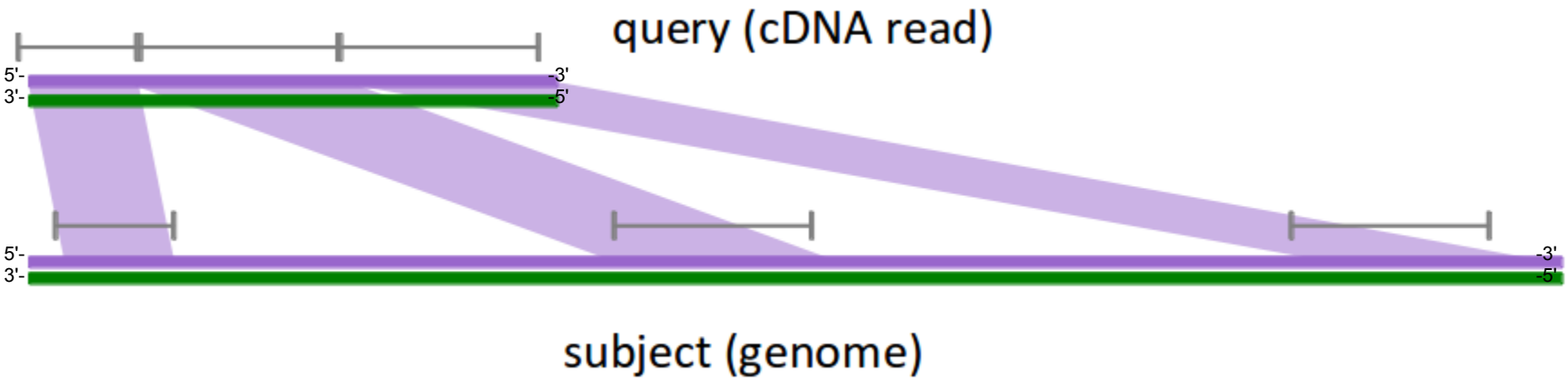
http://vcftools.sourceforge.net/specs.html ... VCF poster

Variant Effect Prediction

- snpEff
- Variant Effect Predictor (EMBL)
- SIFT

back to Alignment ...

Alignment, with splicing



Splicing-aware aligners find alignments that satisfy ***continuity***, ***order***, and ***orientation*** constraints.

(also, intron motifs)

Splicing-aware aligners

- BLAT
- GMAP / GSNAP
- TopHat (Bowtie)
- ...

GMAP / GSNAP

GMAP and GSNAP are spliced aligners developed at Genentech, but now free and open source. GSNAP is comparable to TopHat, while GMAP can align Sanger, 454, and PacBio reads at tolerable speeds. Could plausibly be used to identify alternative splicing isoforms based on single long reads.

http://research-pub.gene.com/gmap/

GMAP:

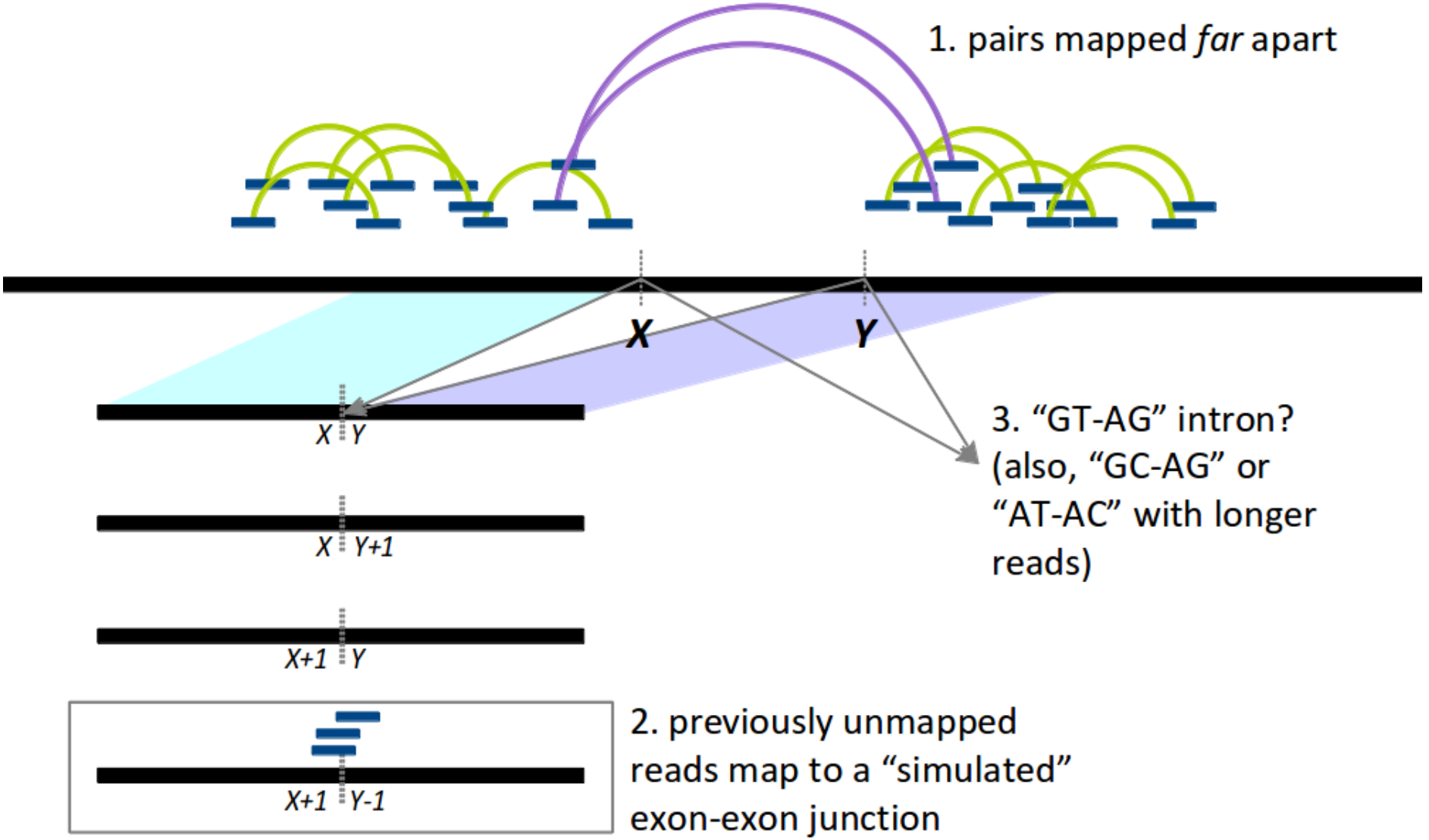
Wu TD and Watanabe CK (2005) “GMAP: a genomic mapping and alignment program for mRNA and EST sequences” Bioinformatics 21:1859 [PMID: 15728110]

GSNAP:

Wu TD and Nacu S (2010) “Fast and SNP-tolerant detection of complex variants and splicing in short reads” Bioinformatics 26:873 [PMID: 20147302]

… new 'gsnap-users' mailing list for GSNAP and GMAP

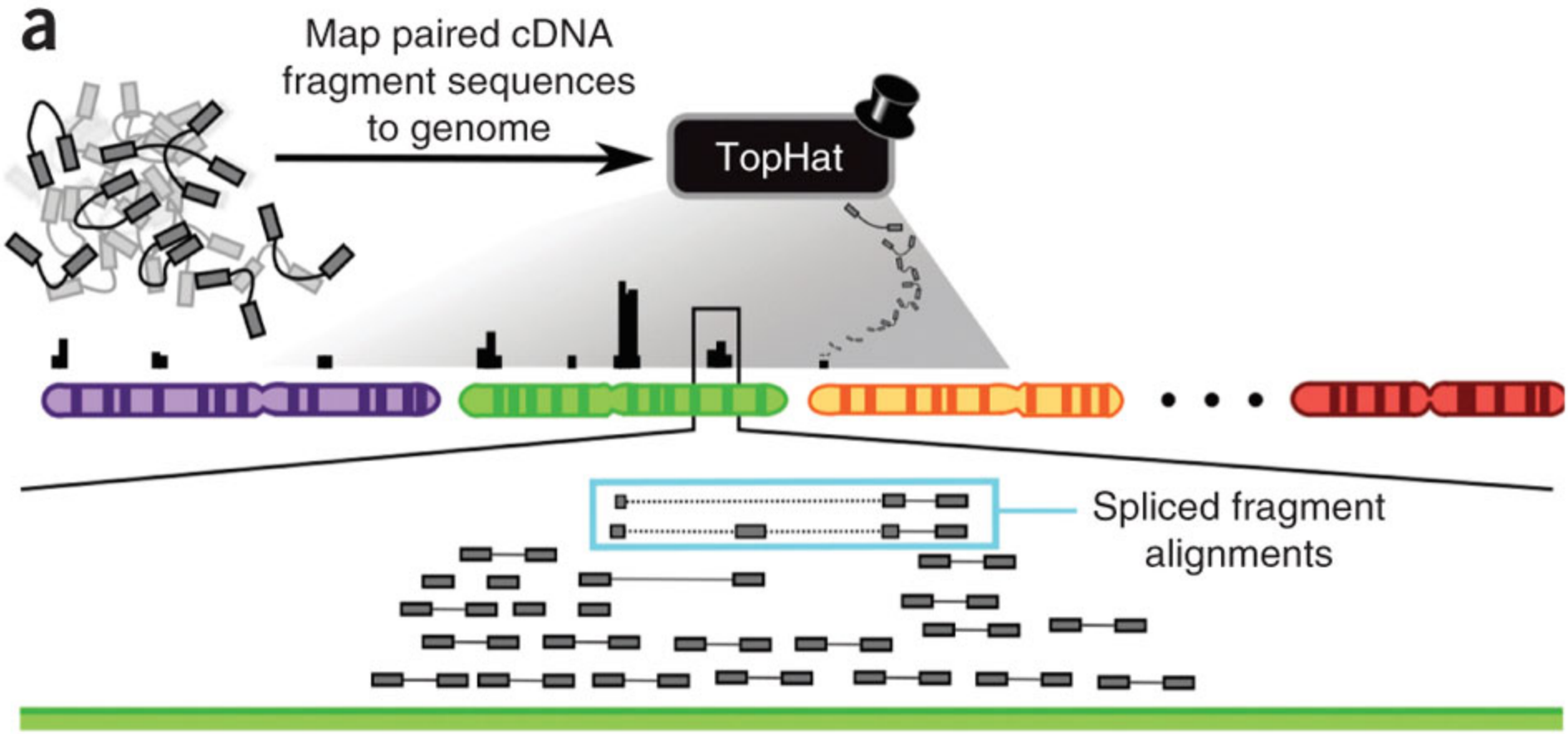
TopHat (using Bowtie)



TopHat (using Bowtie)

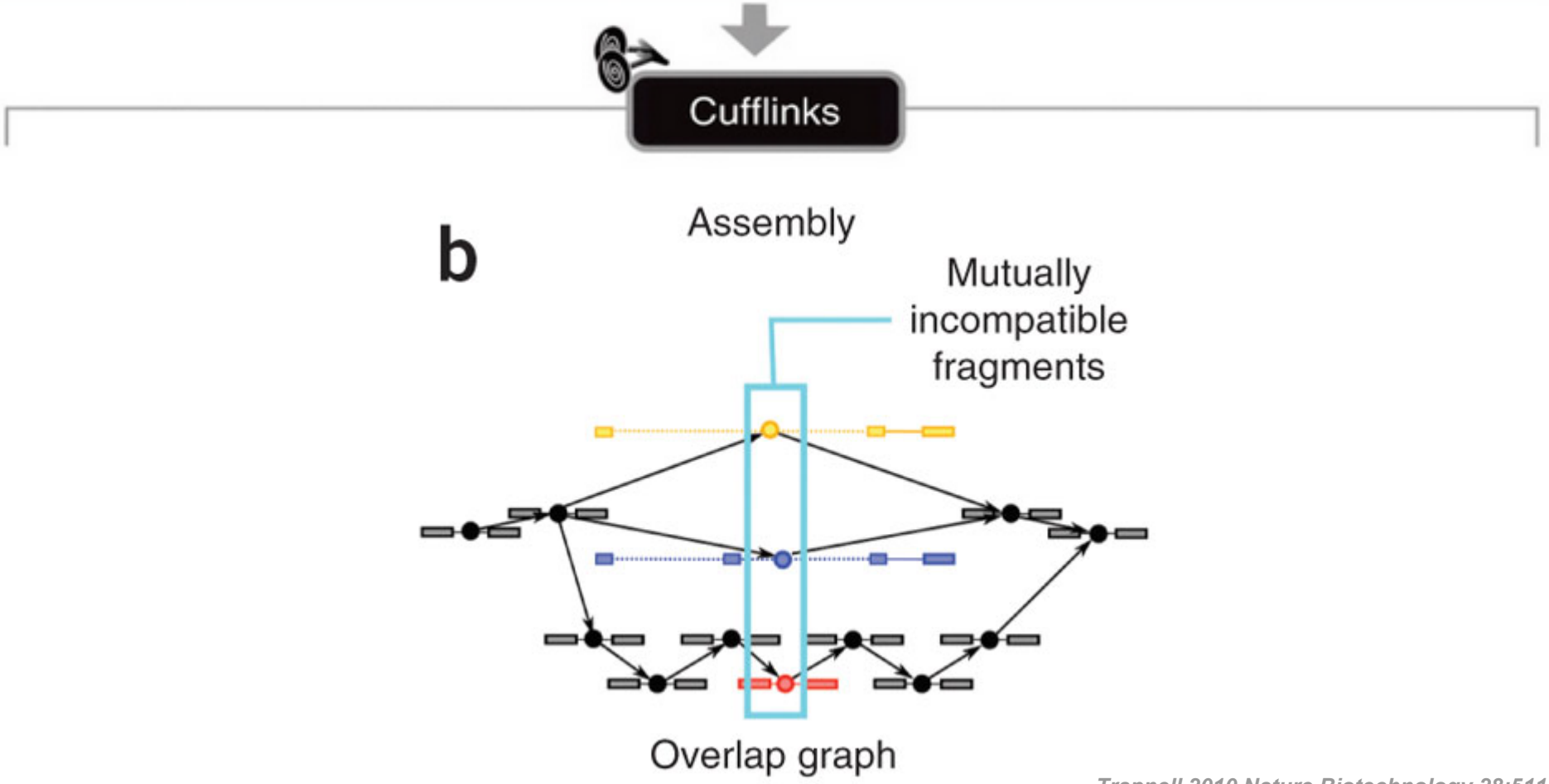
- alignments in a BAM file
- junctions in a BED file
  - see UCSC Genome Browser File Format help

Cufflinks: Transcriptome "Assembler"



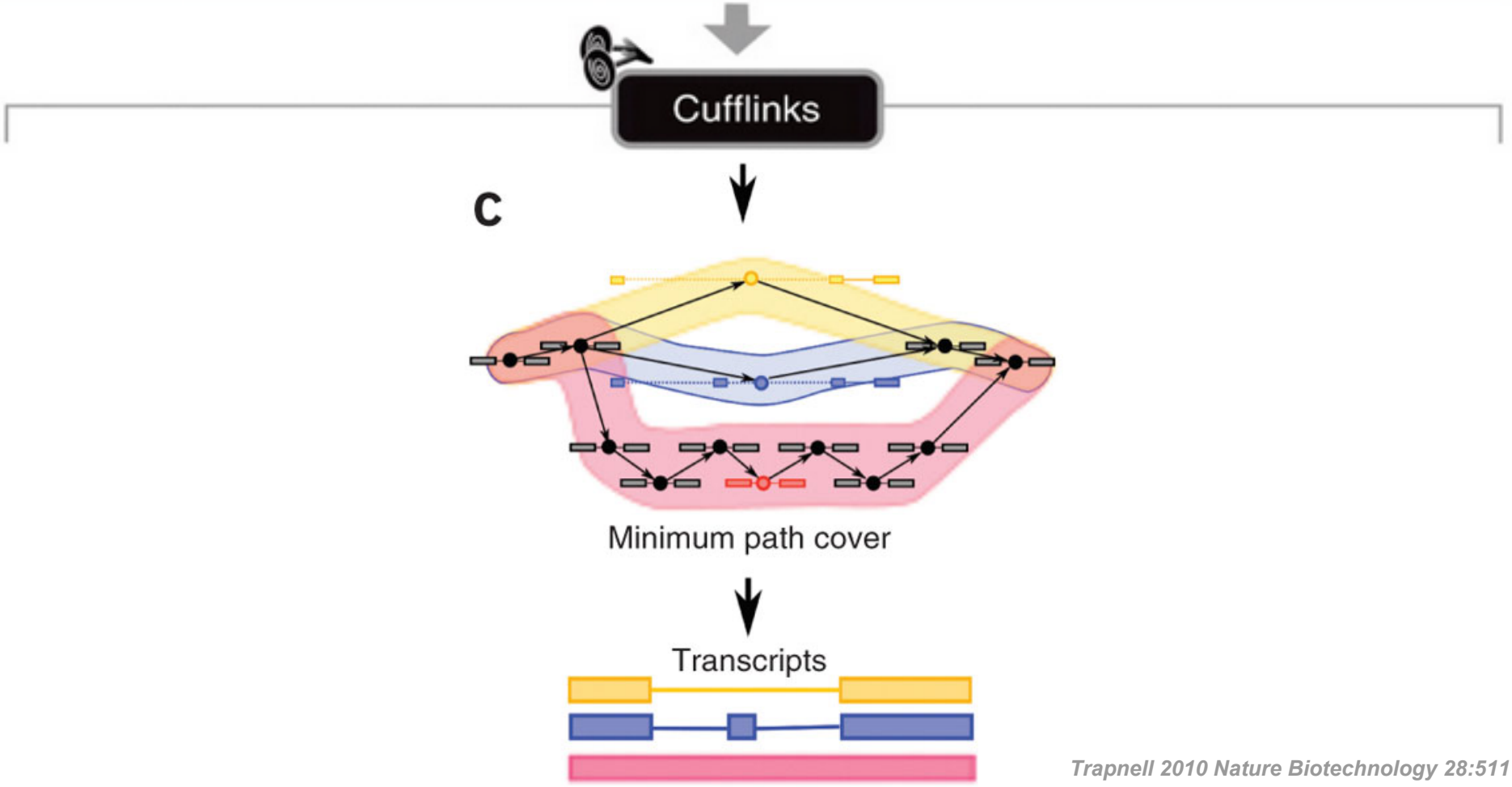
Trapnell 2010 Nature Biotechnology 28:511

Cufflinks: Transcriptome "Assembler"



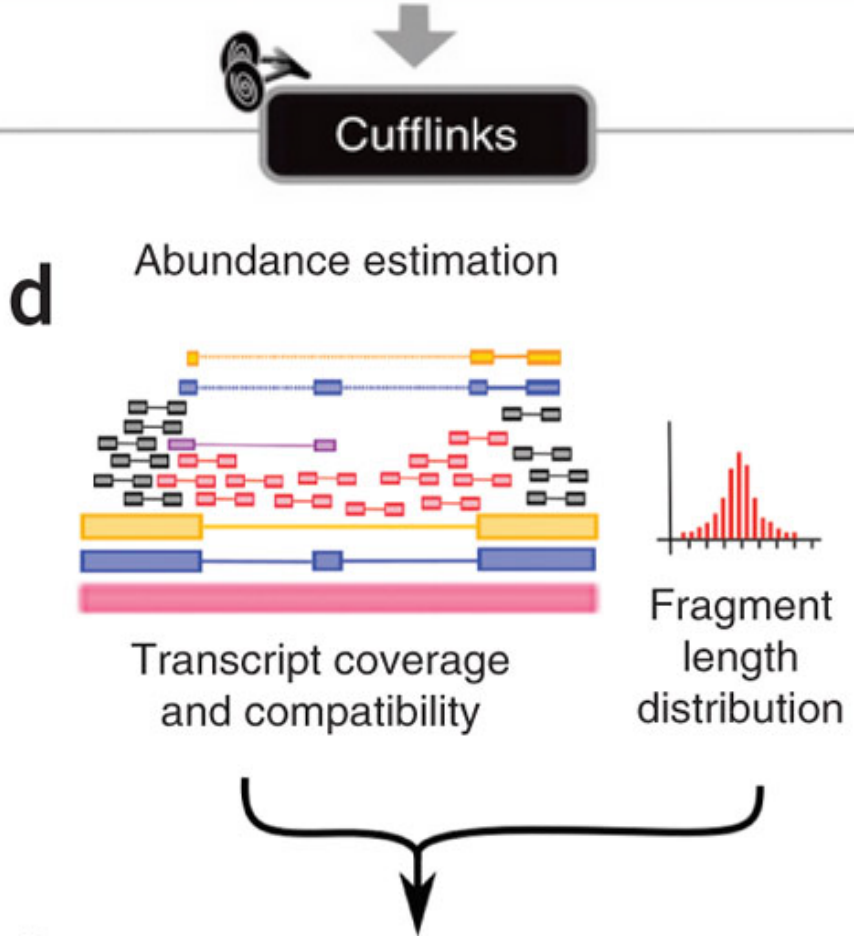
Trapnell 2010 Nature Biotechnology 28:511

Cufflinks: Transcriptome "Assembler"



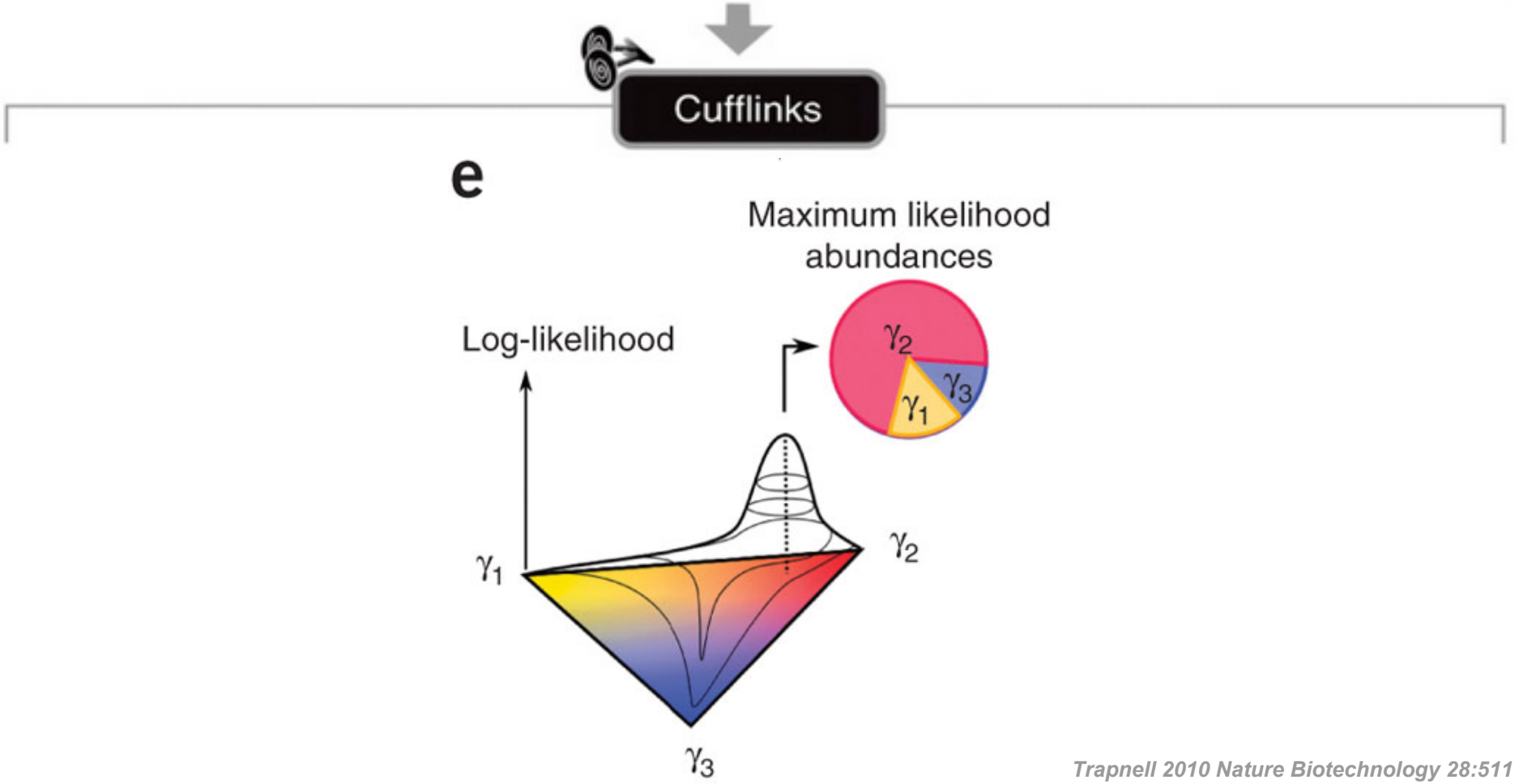
Trapnell 2010 Nature Biotechnology 28:511

Cufflinks: Transcriptome "Assembler"



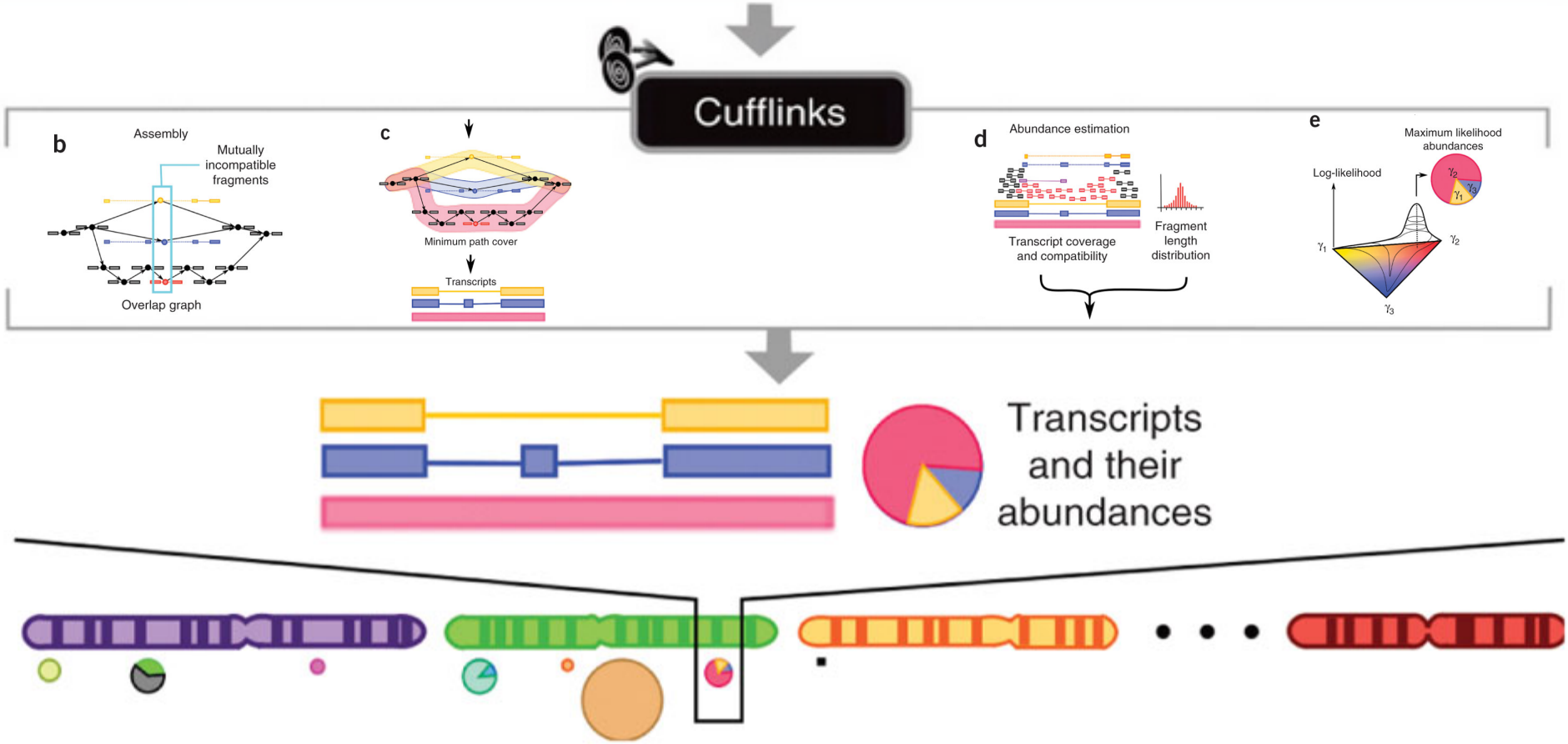
Trapnell 2010 Nature Biotechnology 28:511

Cufflinks: Transcriptome "Assembler"



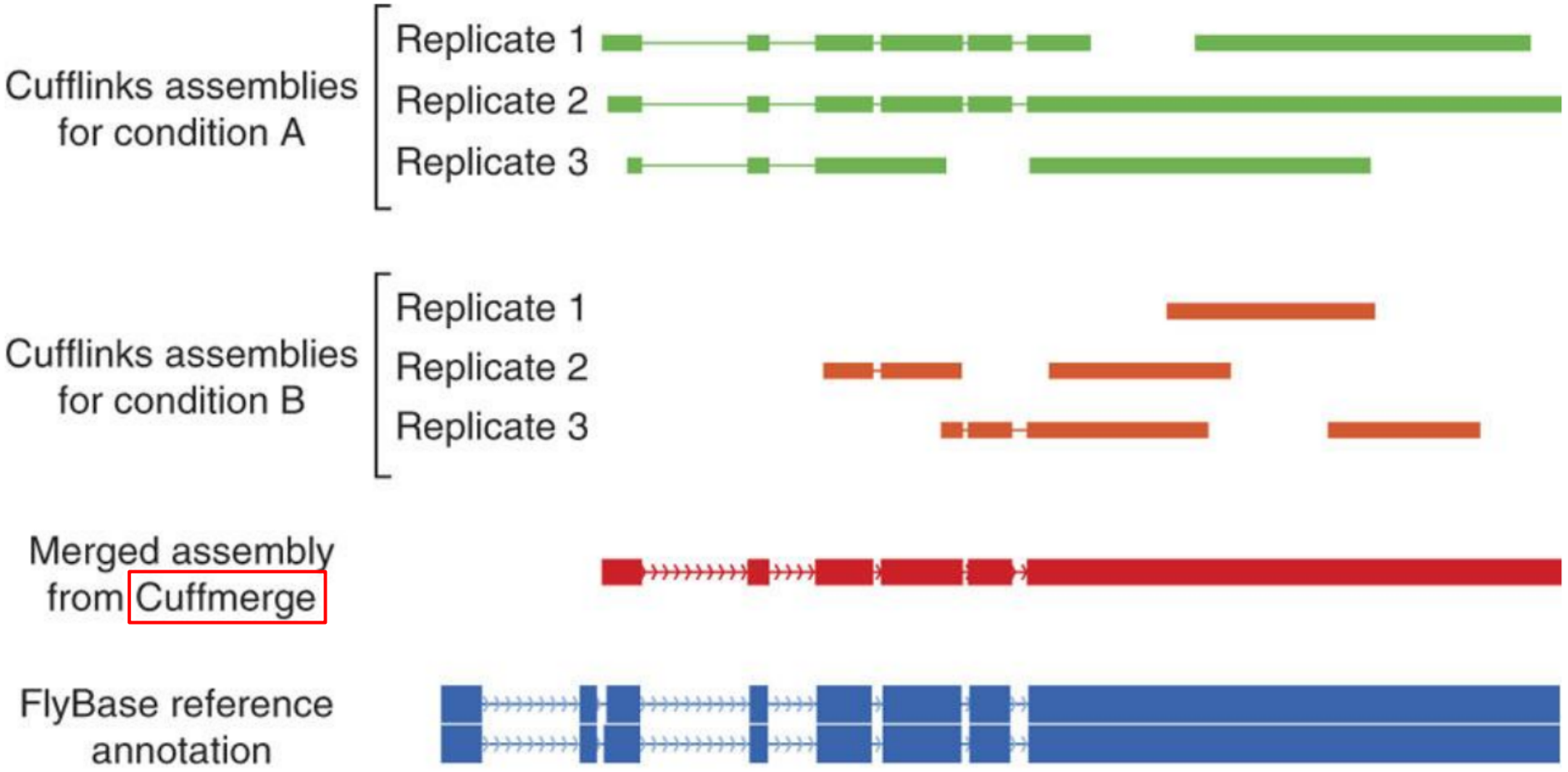
Trapnell 2010 Nature Biotechnology 28:511

Cufflinks: Transcriptome "Assembler"



Trapnell 2010 Nature Biotechnology 28:511

Cufflinks: Transcriptome "Assembler"

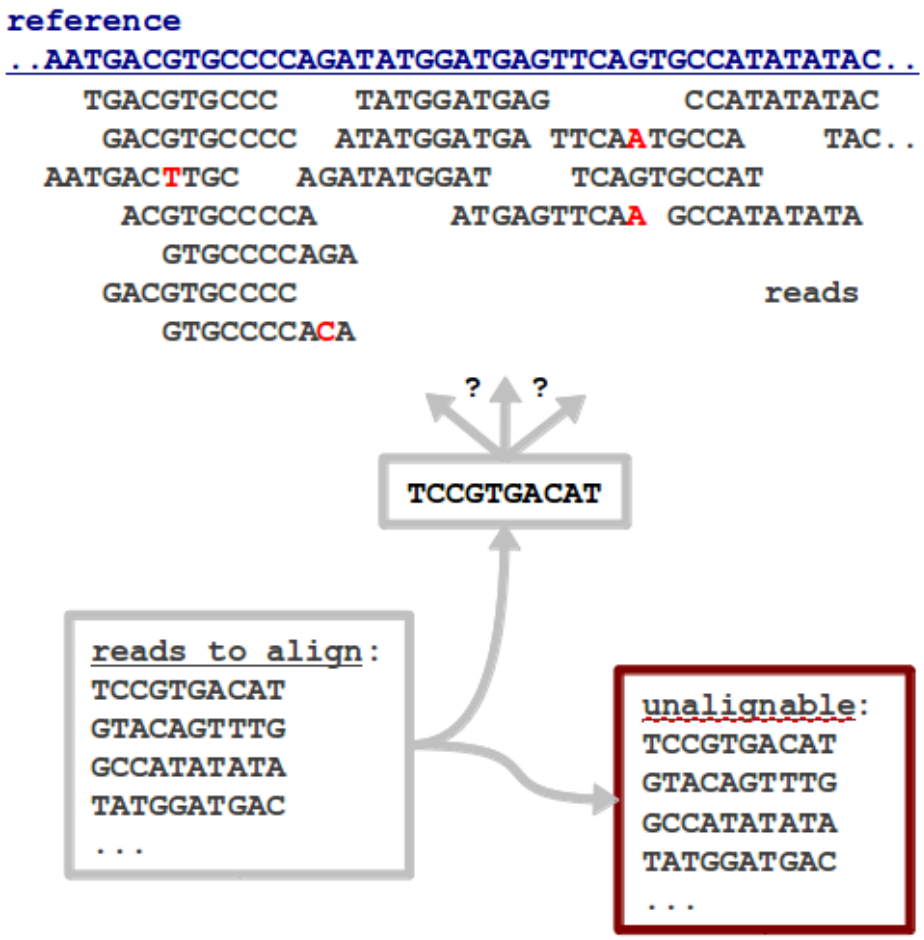


Trapnell 2012 Nature Protocols 7:562

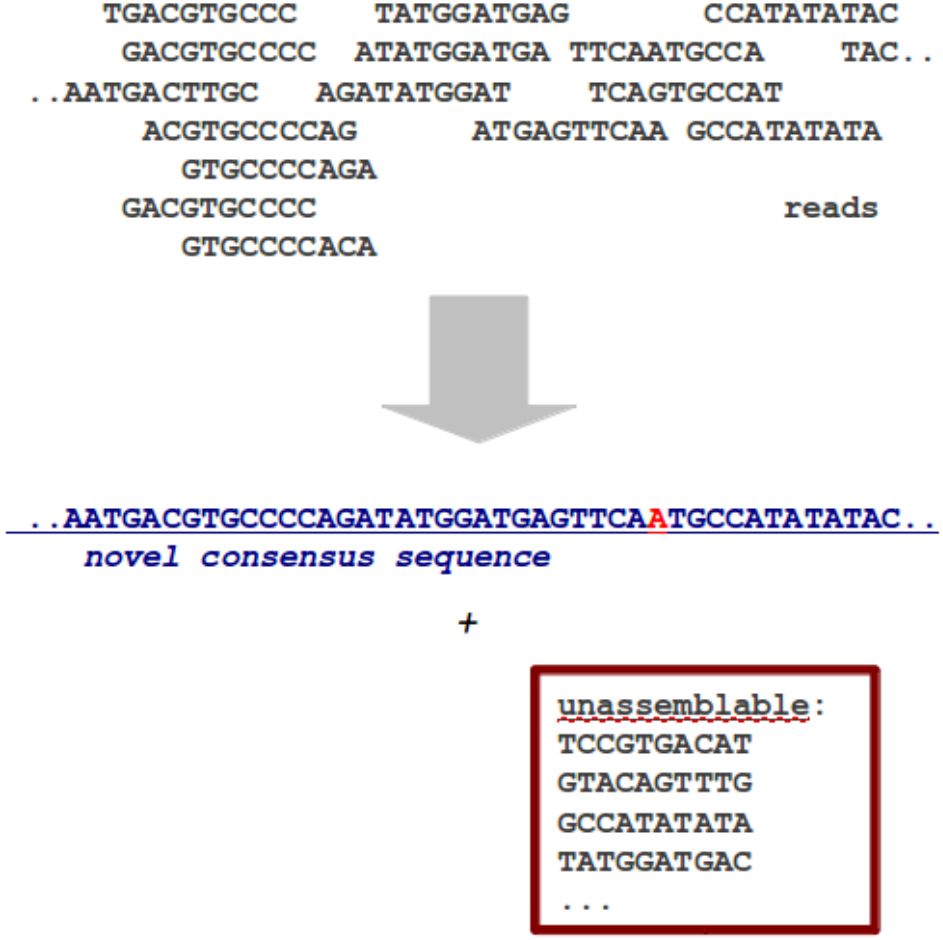
Assembly ... into the unknown

From reads to *molecules*

Alignment



Assembly

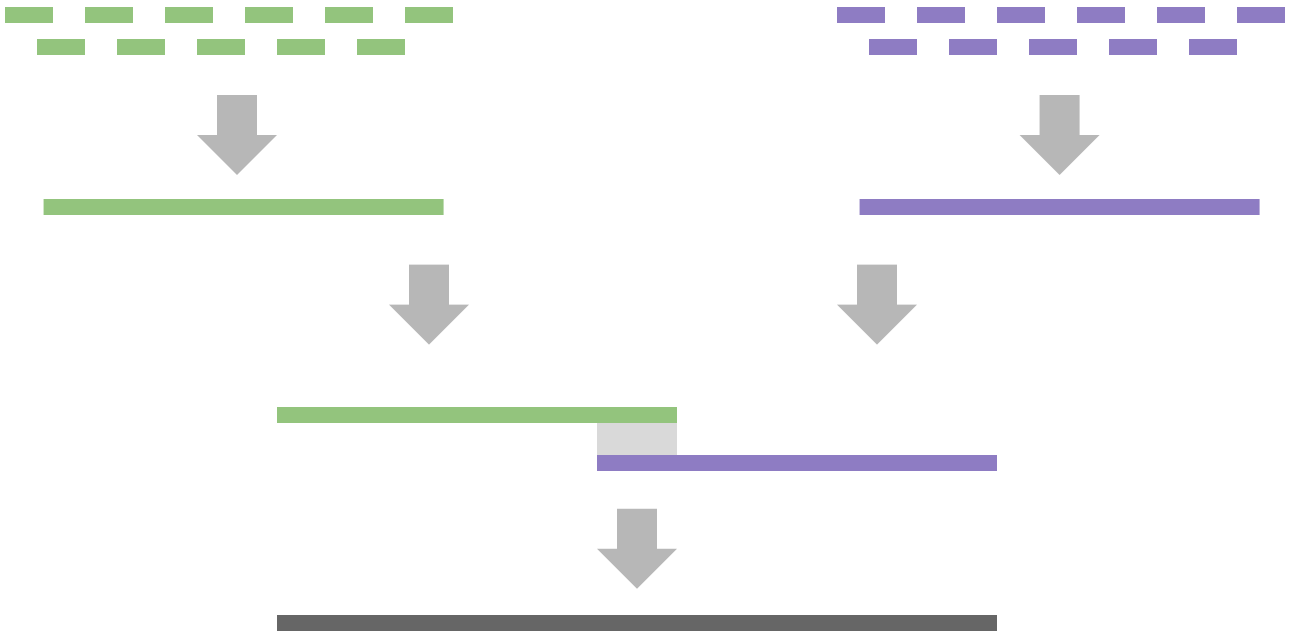


historic Genome Assemblers

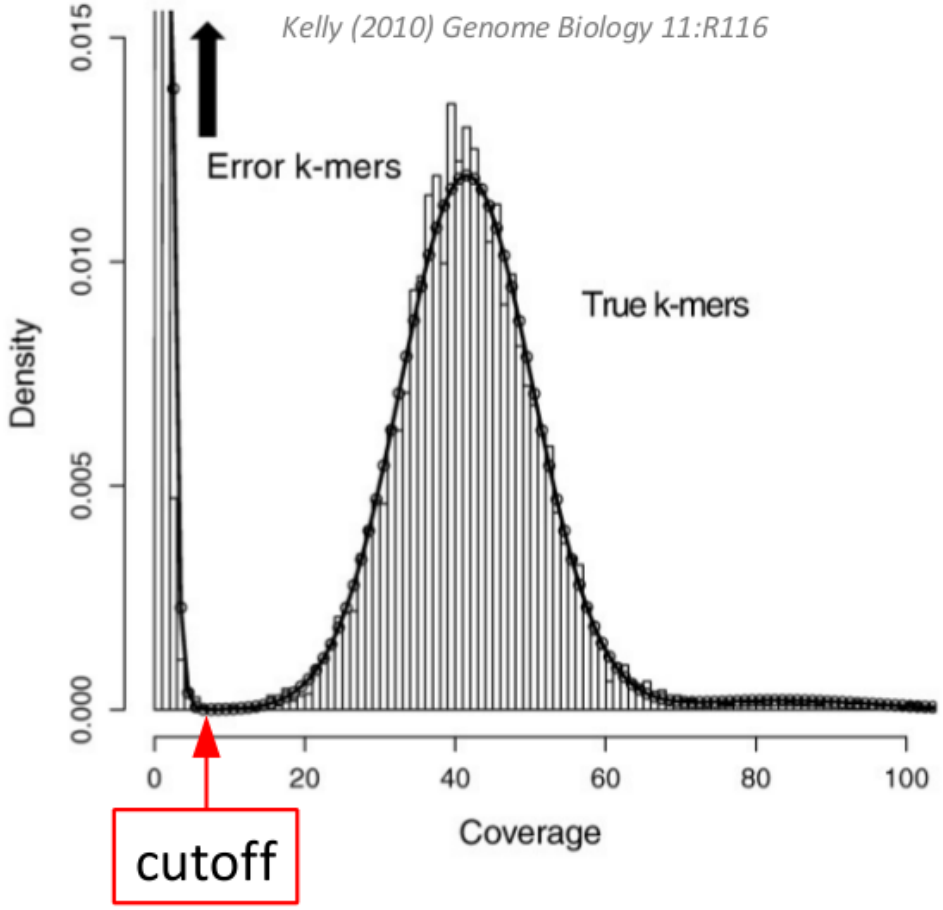
- Celera Assembler (used for whole-genome shotgun human assembly, as opposed to NIH BAC-by-BAC approach) ... now, wgs-assembler (*PBcR!*)
- Velvet (one of 1st de Bruijn graph assemblers)
- ALLPATHS-LG (de Bruijn, recipe-based)
- SGA - String Graph Assembler

Hierarchical Assembly

Amplify **B**acterial **A**rtificial **C**hromosomes, Fosmids, etc.
... sequence, assemble, then assemble the assemblies.



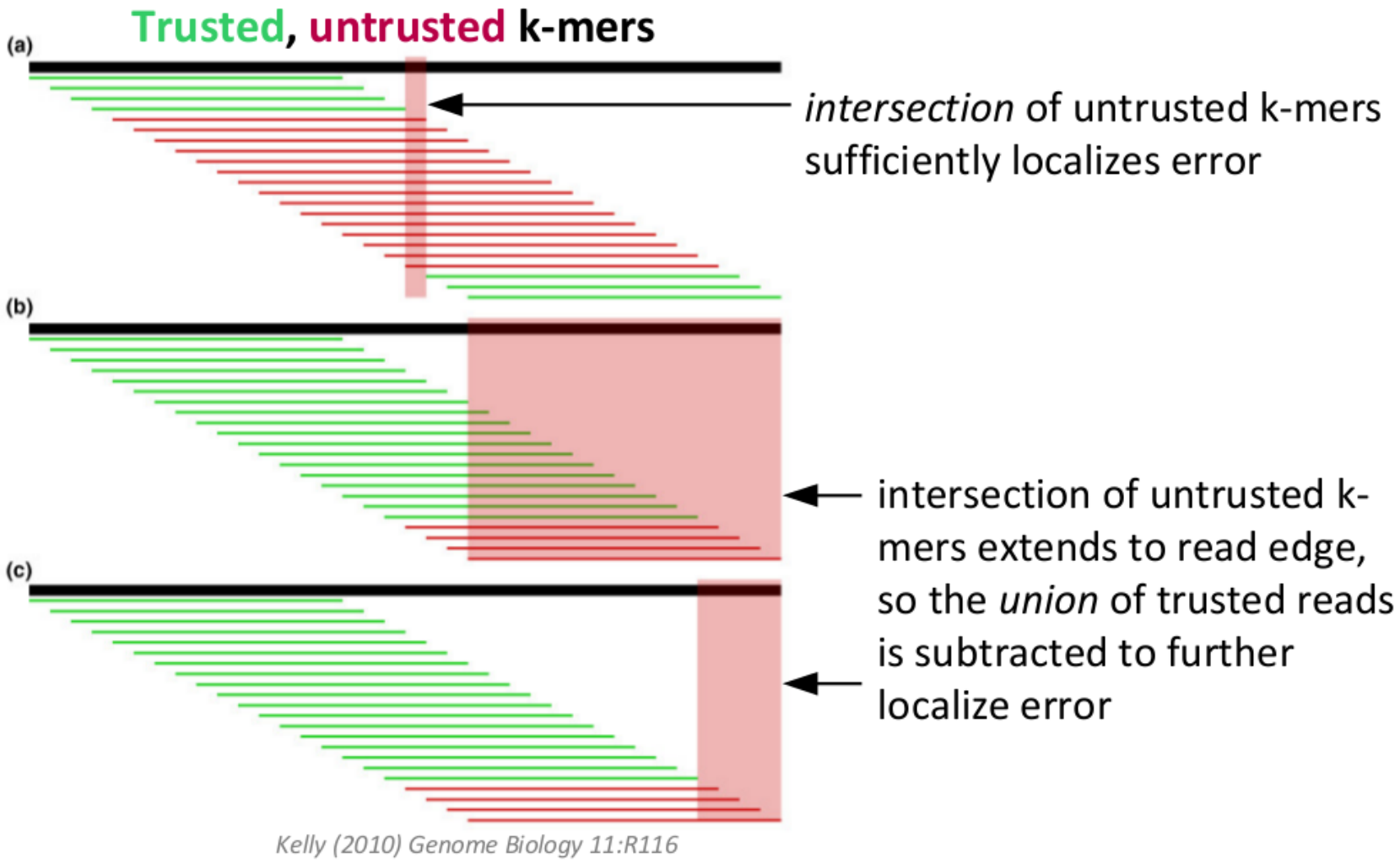
Error Correction (Quake)



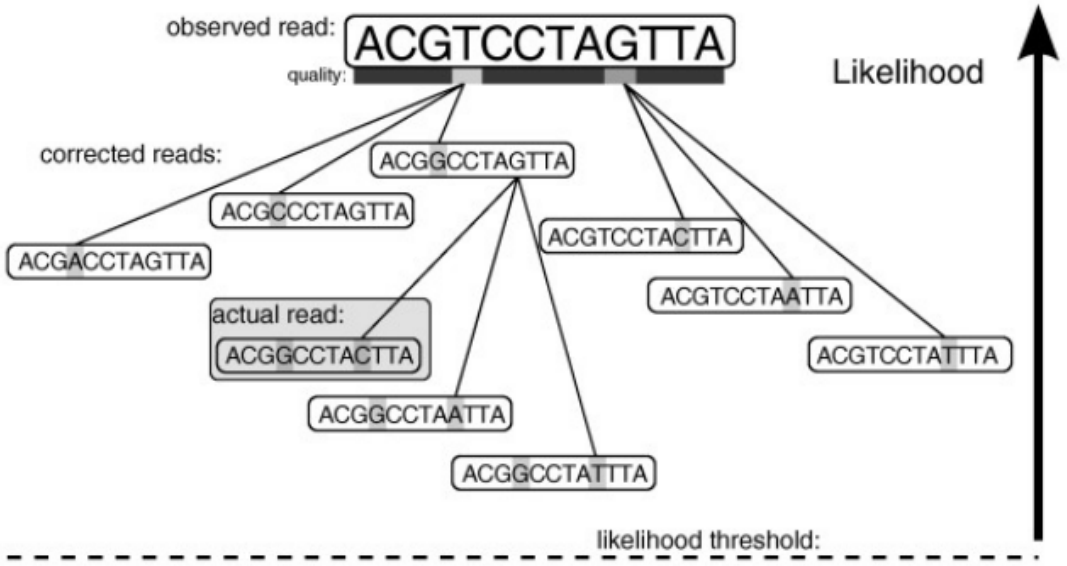
e.g.:

```
CCGGATTTACGCCGAGTGA x 43
CCGGATTTACGCTGAGTGA x 1
                   ^
```

Error Correction (Quake)



Error Correction (Quake)



Searching from highest *likelihood* first, the "actual read" is determined by the first-encountered set of corrections that make all k-mers in the read *trusted k-mers*.

Error Correction

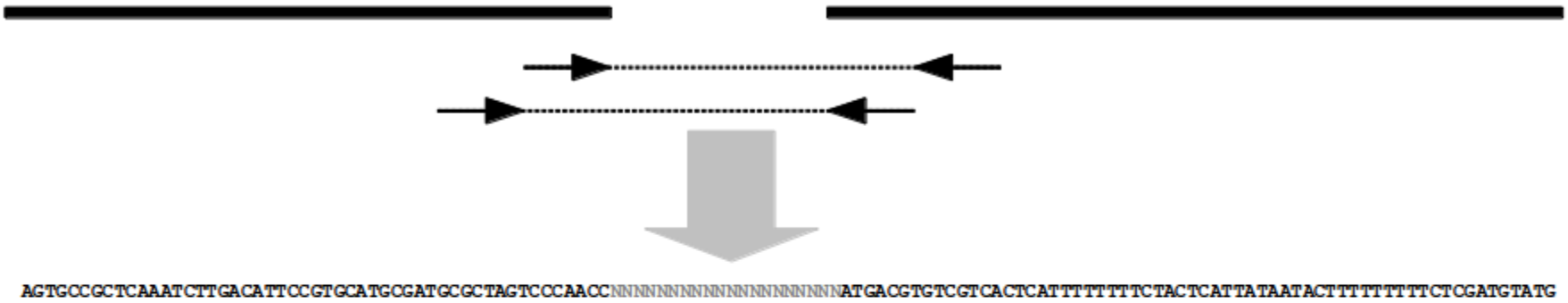
Similar correction methods are incorporated into modern assemblers (like SOAPdenovo, SGA, ALLPATHS), and error *exclusion* (based on k-mer coverage) is an element of some (Velvet ...)

Error Correction (PBcR)

Assuming corrected read sections are still > 1kb, corrected PacBio reads can be extremely helpful in spanning (assembling across) repetitive sequence of that size range.

Scaffolders

What's a scaffold?



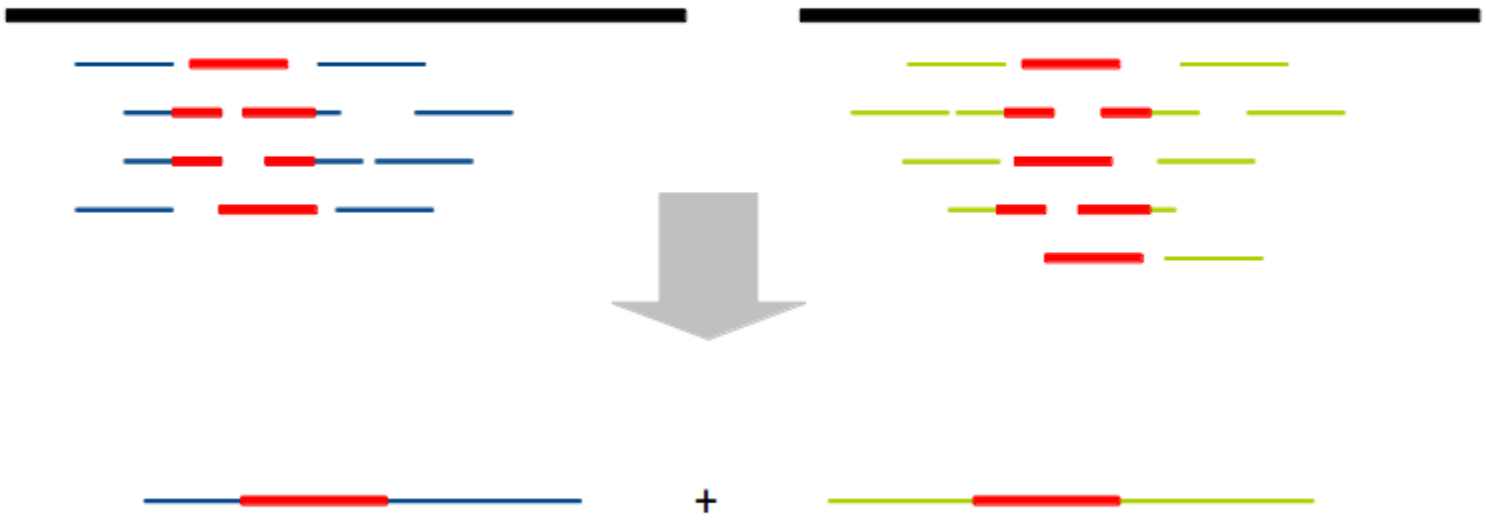
Scaffolding using pairing information in most modern assemblers.

also:
Bambus – scaffolder for AMOS
(http://sourceforge.net/apps/mediawiki/amos)

SSPACE – (**S**tandalone **S**caffolder of **P**re-**A**ssembled **C**ontigs using Paired R**E**ads) …
Boetzer 2010

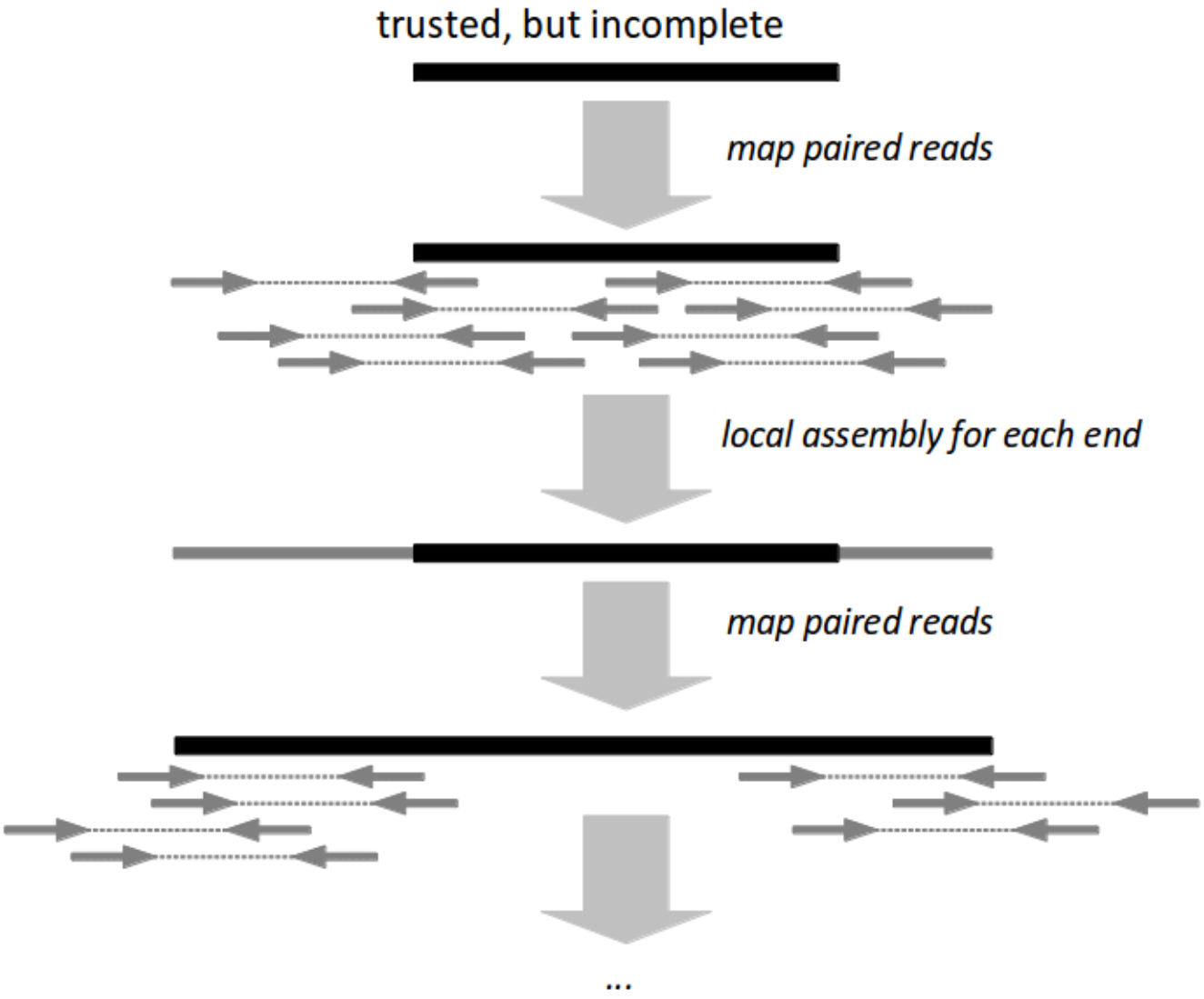
Reference-assisted assembly

When sequence exists that is *close* to what you're trying to assemble, it can be used to guide the assembly, making it easier.



Velvet (Columbus module) ... *makes separate instances of k-mers when they appear in different '-reference' sequences, and doesn't connect across these instances.*

Gap filling / contig extension



Gap filling / contig extension

IMAGE (**I**terative **M**apping and **A**ssembly for **G**ap **E**limination)
Tsai 2010 Genome Biology 11:R41

PRICE (**P**aired **R**ead **I**terative **C**ontig **E**xtension)
DeRisi lab, UCSF

Constructing an assembly "graph"

three “word-osomes”; k-mer size = 3 …

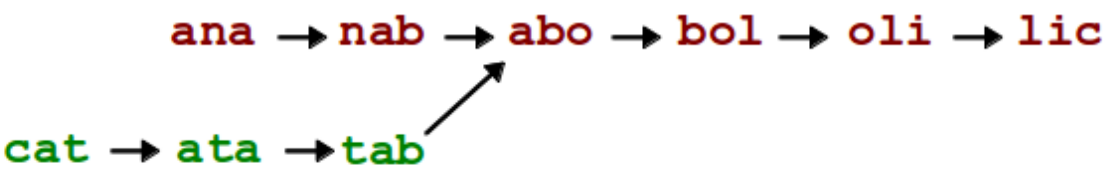
anabolic	catabolic	metabolism
ana	cat	met
nab	ata	eta
abo	tab	tab
bol	abo	abo
oli	bol	bol
lic	oli	oli
	lic	lis
		ism

build graph from 3-mer nodes with 2-mer overlaps …

Constructing an assembly "graph"

three “word-osomes”; k-mer size = 3 …

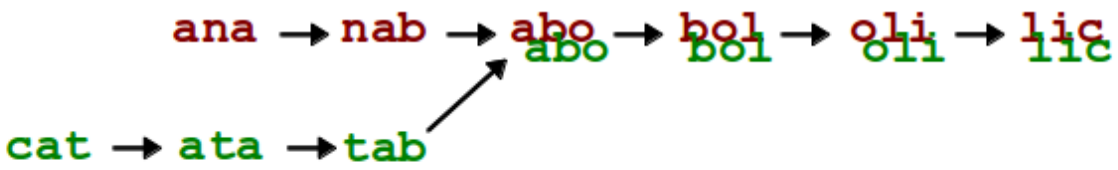
anabolic	catabolic	metabolism
ana	cat	met
nab	ata	eta
abo	tab	tab
bol	abo	abo
oli	bol	bol
lic	oli	oli
	lic	lis
		ism



Constructing an assembly "graph"

three “word-osomes”; k-mer size = 3 …

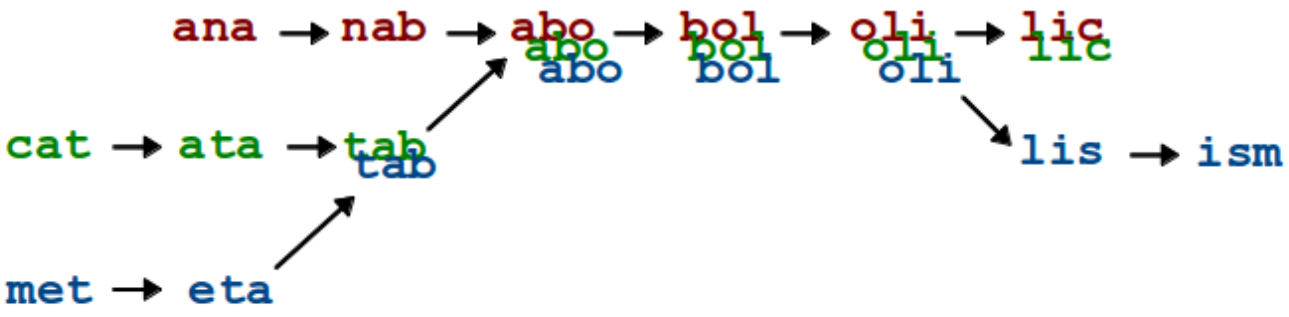
anabolic	catabolic	metabolism
ana	cat	met
nab	ata	eta
abo	tab	tab
bol	abo	abo
oli	bol	bol
lic	oli	oli
	lic	lis
		ism



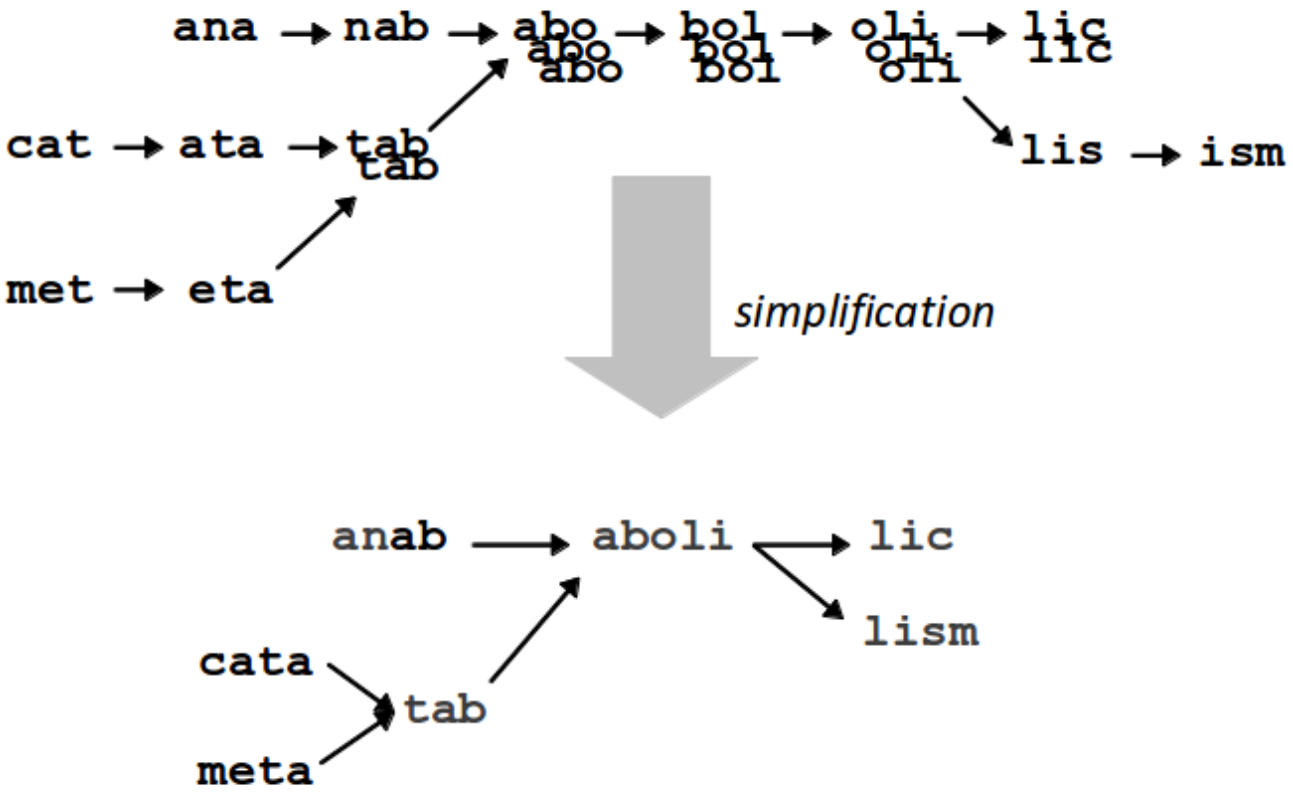
Constructing an assembly "graph"

three “word-osomes”; k-mer size = 3 …

anabolic	catabolic	metabolism
ana	cat	met
nab	ata	eta
abo	tab	tab
bol	abo	abo
oli	bol	bol
lic	oli	oli
	lic	lis
		ism



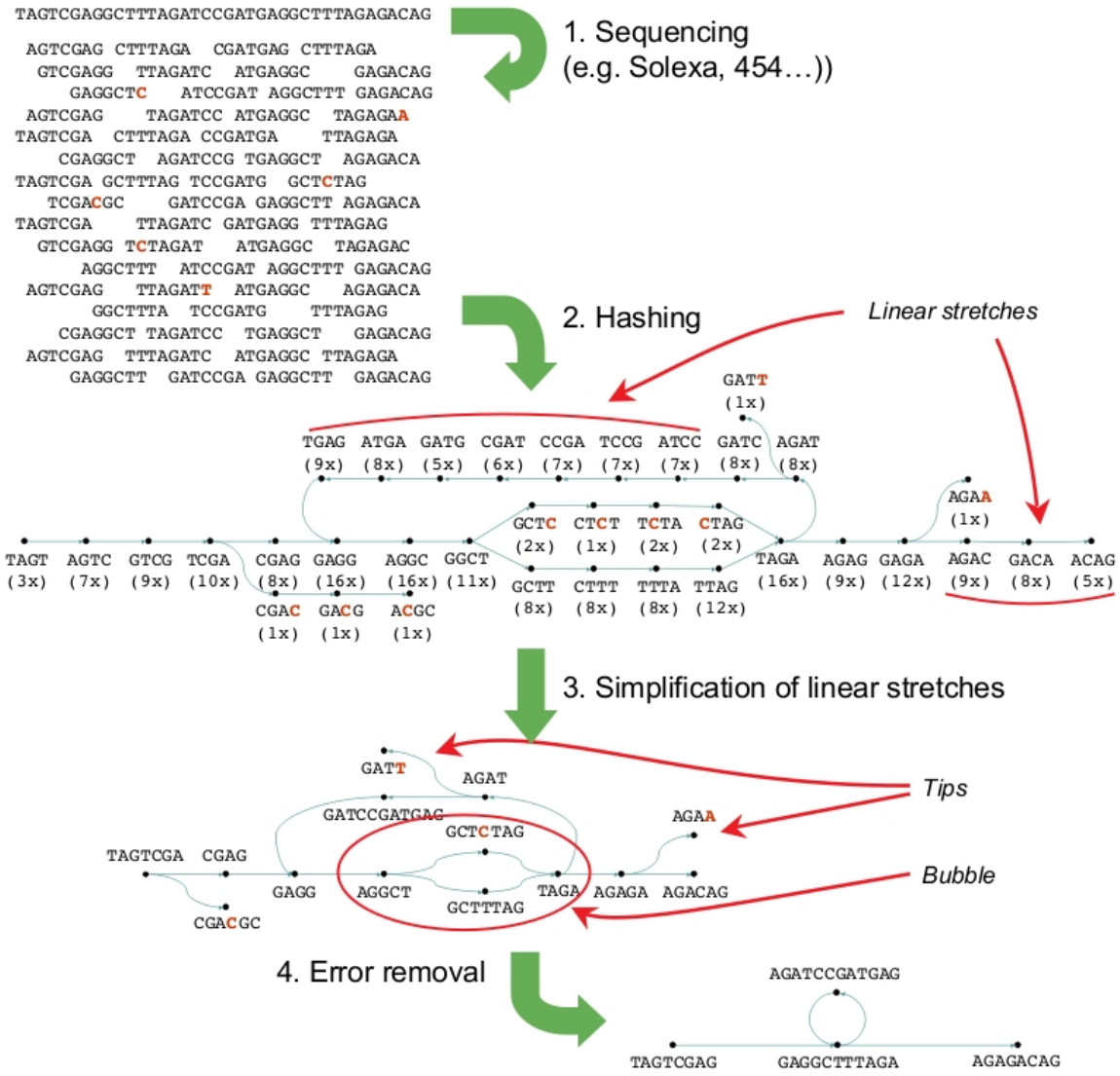
Constructing an assembly "graph"



de Bruijn graph assembler, Velvet

Build graph from 7 bp *reads*, with errors ... using 4 bp *k-mers*

Tracking k-mers, not reads, essentially compresses the data ... important for NextGen era!

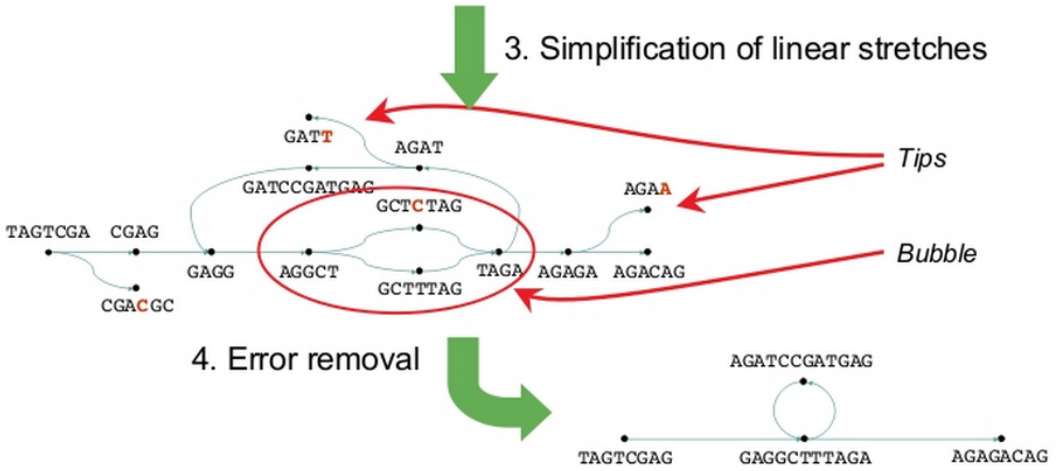


de Bruijn graph assembler, Velvet

Tip Removal

Bubble Popping

(Coverage Constraints)



Cutting at every ambiguity (branch point) yields the final contigs:

```
TAGTCGAG
GAGGCTTAGA
AGATCGGATGAG
AGAGACAG
```

Table 1. Efficiency of the Velvet error-correction pipeline on the BAC data set

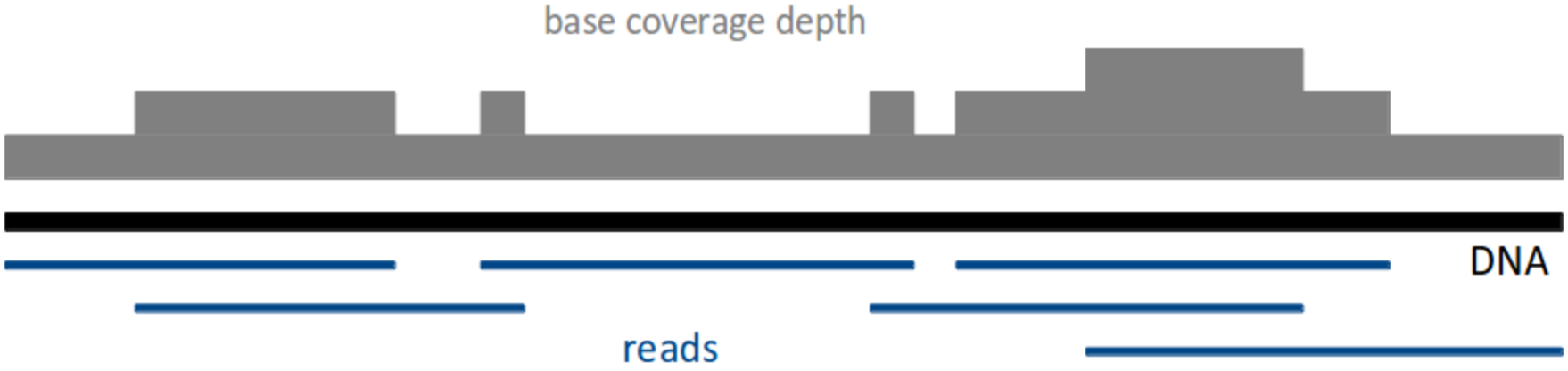
Step	No. of nodes	N50 (bp)	Maximum length (bp)	Coverage (percent >50 bp)	Coverage (percent >100 bp)
Initial	1,353,791	5	7	0	0
Simplified	945,377	5	80	4.3	0.2
Tips clipped	4898	714	5037	93.5	78.7
Tour Bus	1147	1784	7038	93.4	90.1
Coverage cutoff	685	1958	7038	92.0	90.0
Ideal	620	2130	9045	93.7	91.9

Zerbino 2008 Genome Research 18: 821-829

K-mer coverage ... ?

Performance (speed, memory, *effectiveness of assembly*) of de Bruijn-graph assemblers is correlated with k-mer coverage, not base coverage.

Base coverage



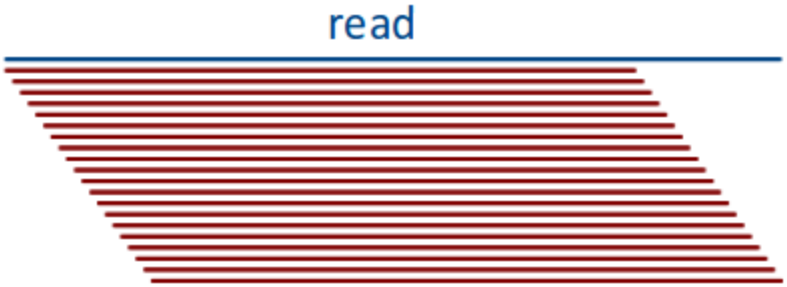
base coverage: $C = \frac{(\sum_{i=1}^{n} L_i)}{G}$

G ... *Genome length*

L_i ... *Length of i^{th} read*

$\hat{L}$ ... *mean read length*

K-mer coverage



k-mers tile across reads

(L - k + 1) k-mers in a read of length L

k-mer coverage: $C_k = \frac{(\sum_{i=1}^{n} K_i)}{G}$

G ... *Genome length*

L_i ... *Length of i^{th} read*

$\hat{L}$ ... *mean read length*

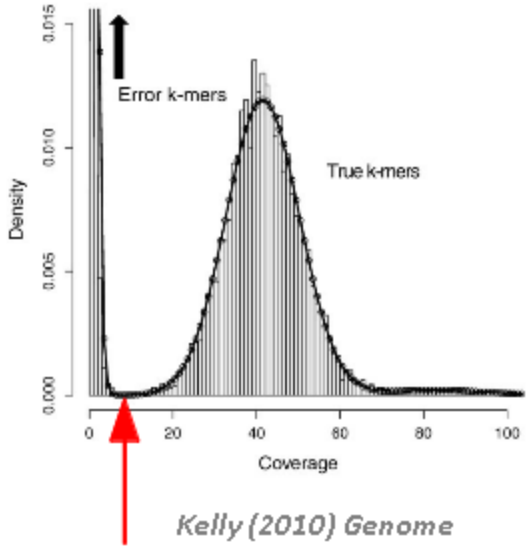
(or, converting from base coverage): $C_k \simeq \frac{C \times (\hat{L} - k + 1)}{\hat{L}}$

Choosing k

”Smaller k-mers increase the connectivity of the graph by simultaneously increasing the chance of observing an overlap between two reads and the number of ambiguous repeats in the graph. There is therefore a balance between sensitivity and specificity determined by k.”

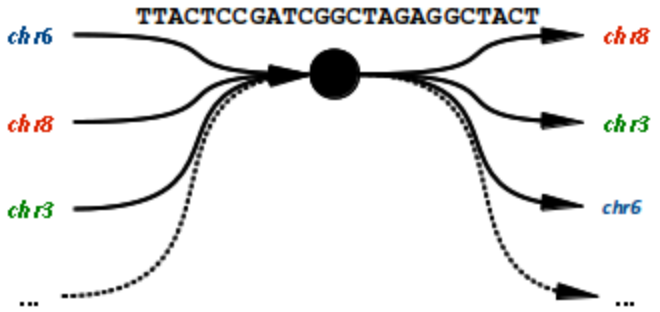
~Zerbino (2008) Genome Research 18:821

Choosing k



Kelly (2010) Genome Biology 11:R116

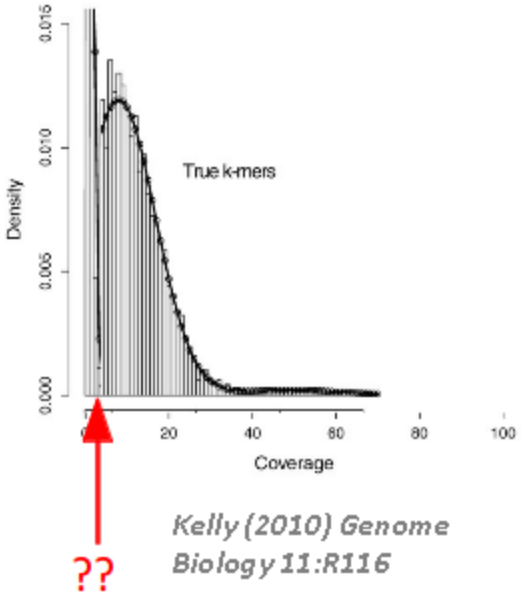
but ...



number of ambiguous repeats in the graph. There is therefore a balance between sensitivity and specificity determined by *k*."

~Zerbino (2008) Genome Research 18:821

Choosing k



Kelly (2010) Genome Biology 11:R116

but ...

chr6...TTCTGGAGTTACTCCGATCGGCTAGAGGCTACTCCGCGA...*chr6*

TTACTCCGATCGGCTAGAGGCTACTAATGGC...*chr8*

chr8...TGCGATACTTACTCCGATCGGCTAGAGGCTACT

....TCCTGCTCTTACTCCGATCGGCTAGAGGCTACT

therefore a balance between sensitivity and specificity determined by *k*."

~Zerbino (2008) Genome Research 18:821

Allpaths-LG ... and its "recipe"

Ribeiro (2012) Genome Research doi: 10.1101/gr.141515.112
Gnerre (2011) PNAS 108:1513

Makes use of a “recipe” of three (or four) different libraries (see below) … can be run without largest scale libraries, but not for best results. Makes sense for an institute that can standardize its sequencing and bioinformatics together.

Gnerre 2011:

- 45x … Overlapping PE reads (180 bp ISIZE, >100bp reads)
- 45x … Short jump / MP (3kb)
- 5x ….. Optional long jump / MP (6kb)
- 1x ….. Optional fosmid jump / MP (40kb)

Ribeiro 2012:

- 50x … Overlapping PE reads (180bp ISIZE, >100bp reads)
- 50x … 1-3kb PacBio reads
- 50x … Long jump / MP (2-10kb)

sga: String Graph Assembler

Simpson, J and Durbin, R (2010) "Efficient construction of an assembly string graph using the FM-index" Bioinformatics 26:i367

String graphs retain the information lost by de Bruijn graphs – full read context – by building graphs based on the ***full overlaps*** between reads (instead of k-mers).

But, this requires all-to-all overlap detection!

sga utilizes ***BWT*** & ***FM-index*** to make this tractable, but graph construction is still the most (computationally) expensive step. Compared to de Bruijn graph assemblers, sga uses less memory, but is significantly slower.

Stay tuned, as sga is developed ...

fermi (Heng Li)

Li (2012) Bioinformatics doi: 10.1093/bioinformatics/bts280

Inspired by sga, fermi follows the OLC paradigm, using the FM-DNA-index. Aims to preserve heterozygotes in final assembly, rather than collapse them to a single consensus. Read information is preserved in the final set of unitigs, so that variant calling can be performed with better sensitivity as compared to mapping to an existing reference, or *de novo* assembly followed by read re-alignment to that assembly.

Stay tuned, as fermi is developed ...

Assembly Competitions

Assemblathon

http://assemblathon.org/

1: Earl 2011 Genome Research 21:2224

2: ArXiv.org - ***http://arxiv.org/abs/1301.5406***

GAGE - **G**enome **A**ssembly **G**old-standard **E**valuations

http://gage.cbcb.umd.edu/

dnGASP - ***d****e* ***n****ovo* **G**enome **A**ssembly **P**roject

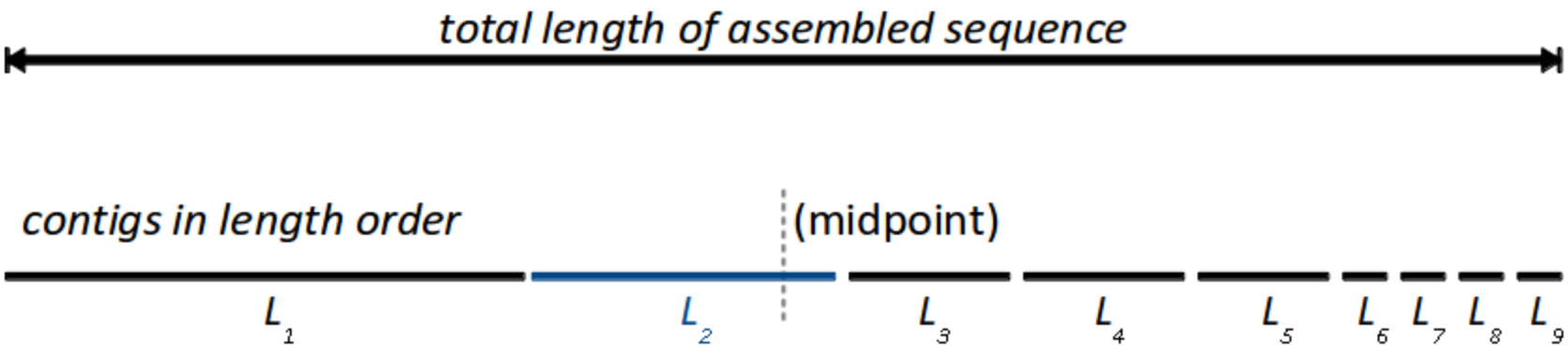
http://cnag.bsc.es/

Assembly Assessment

- N50
- NG50
- Cumulative Length Plots
- Feature Response Curves
- (Alignment) Block NG50 (versus a good? reference)
- Haplotype Block NG50

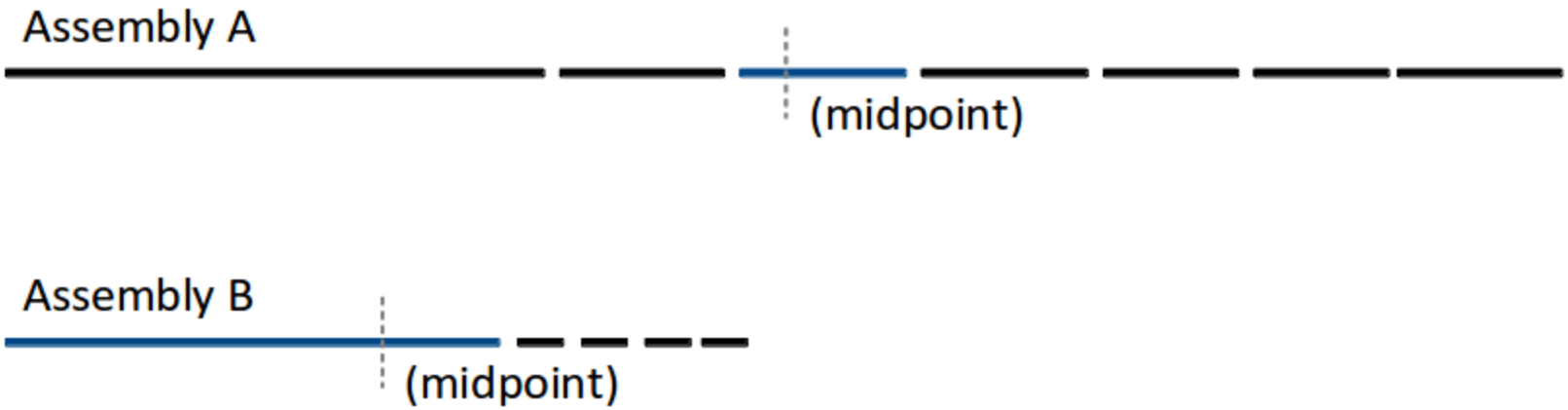
N50

"Half of all the sequence I've assembled is in contigs of at least {N50} bp ... "

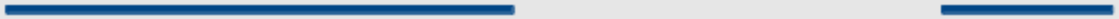


$N50 = L_2$

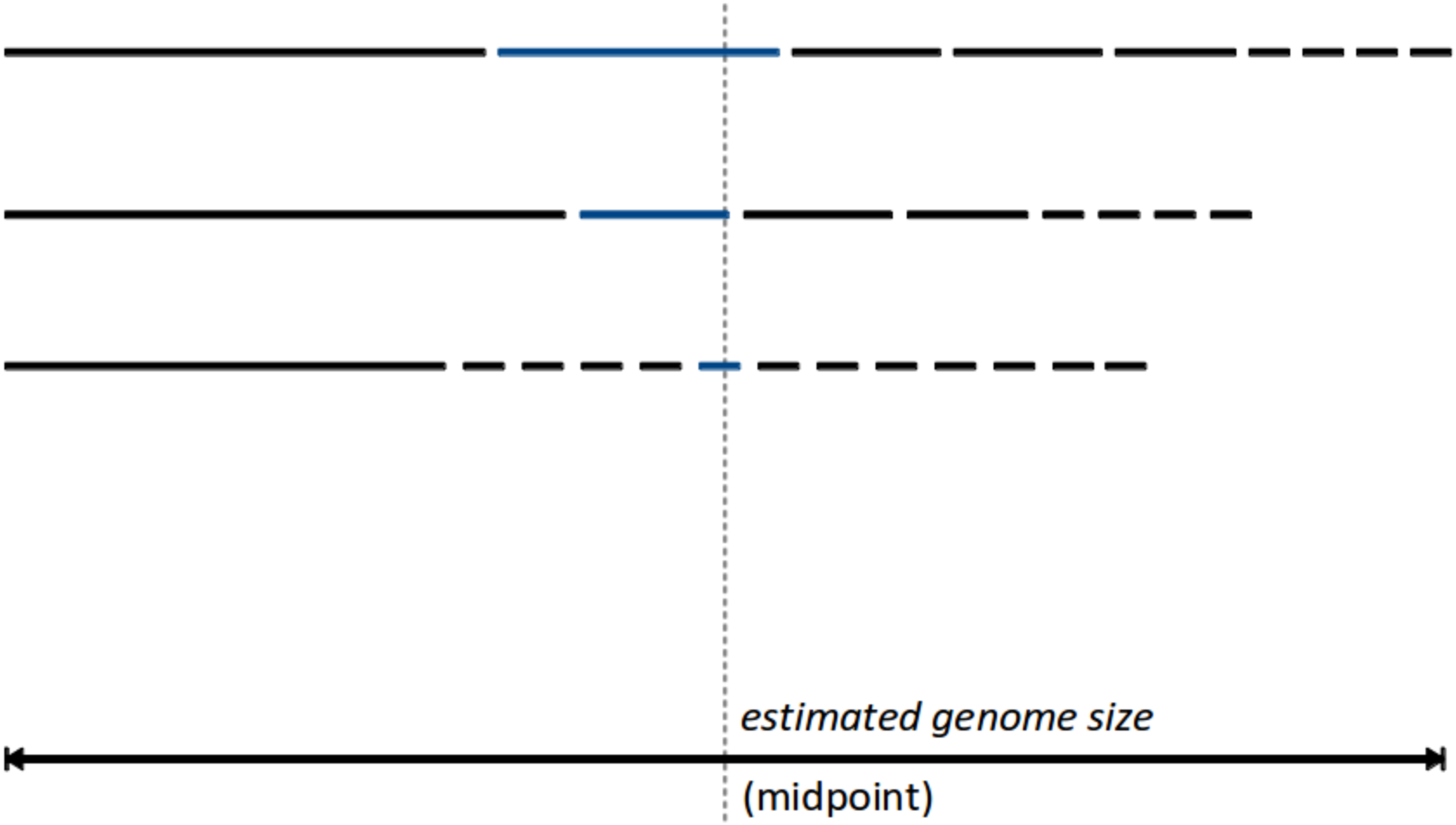
N50, confused



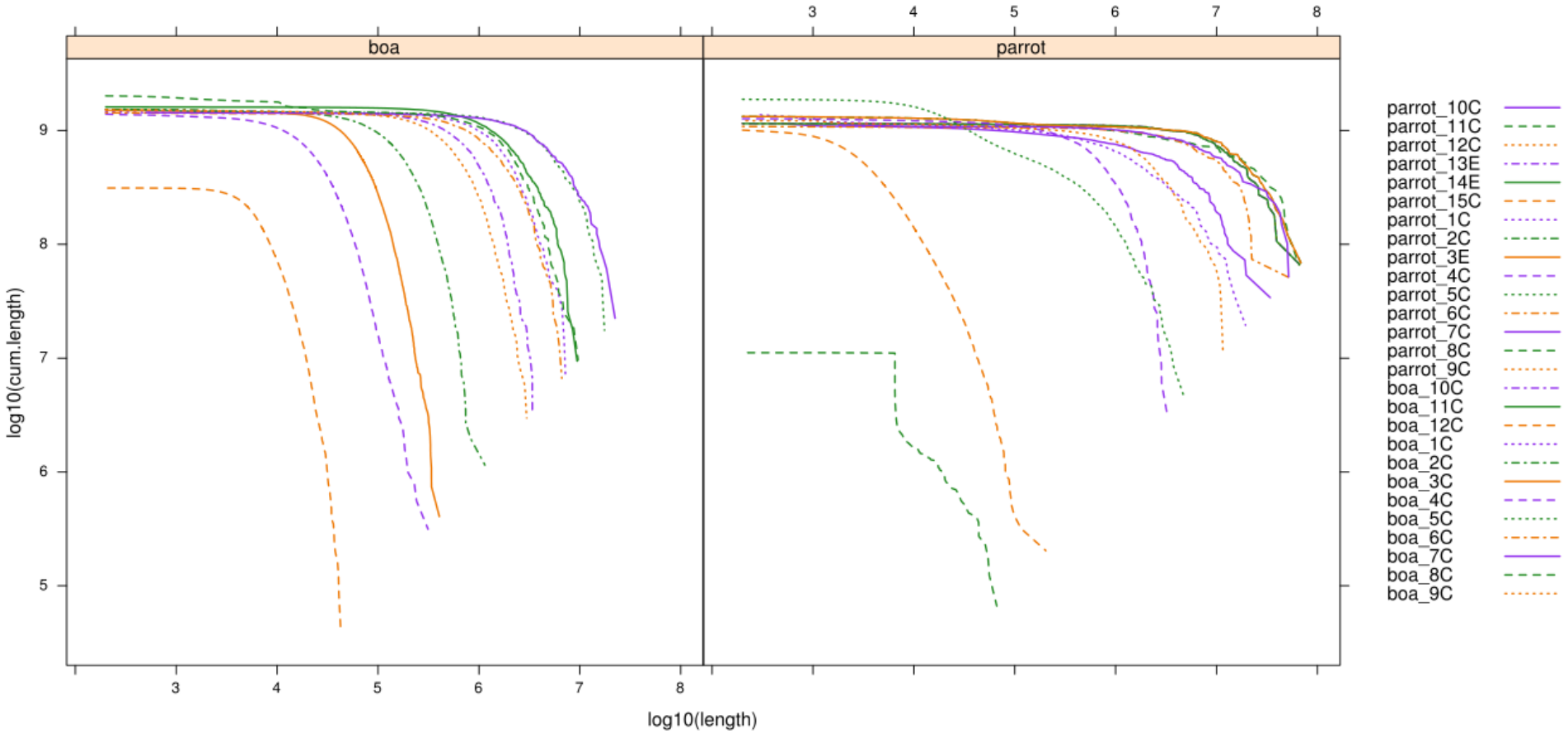
Assembly B's N50 >> Assembly A's N50 (??)



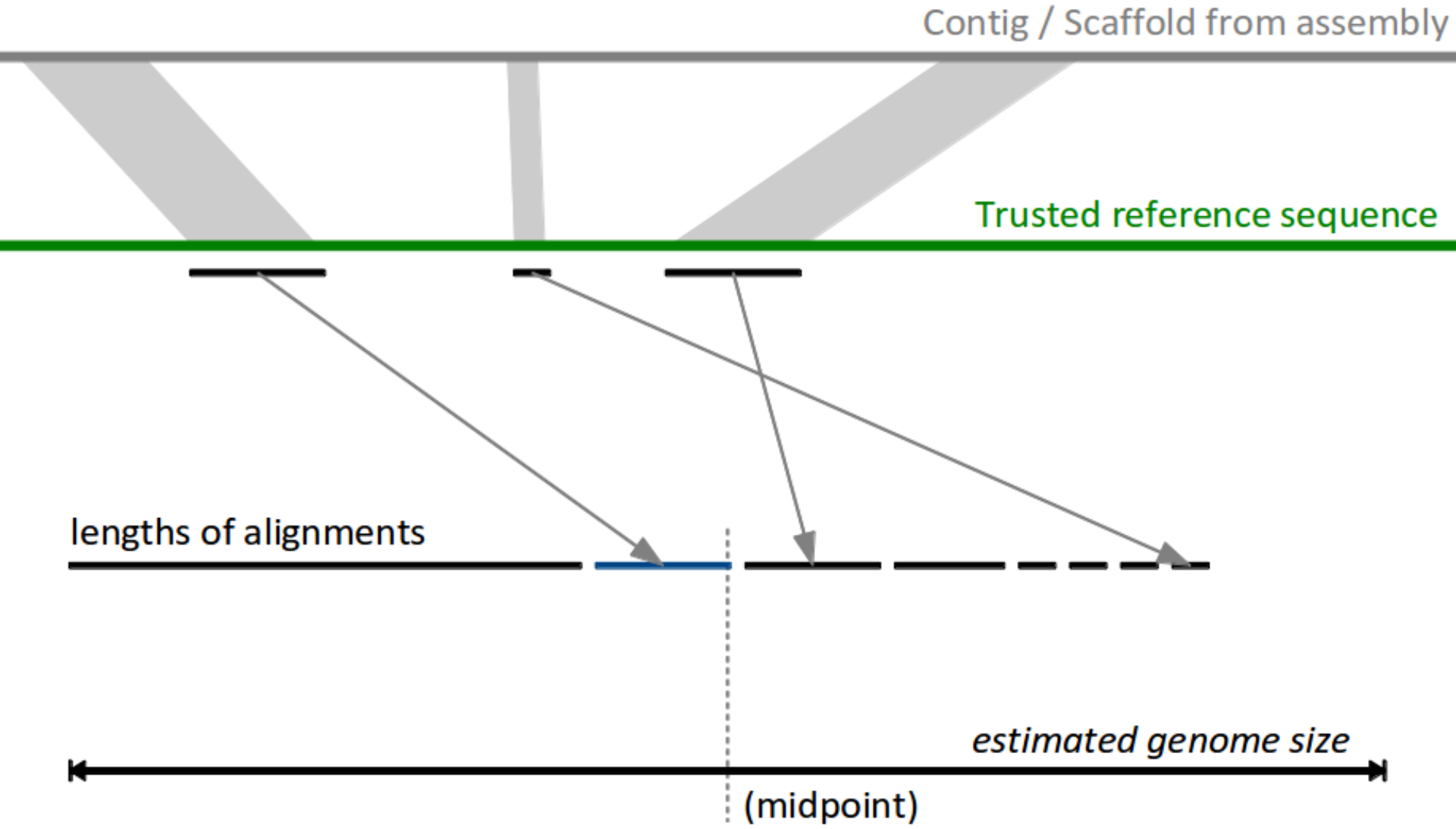
NG50



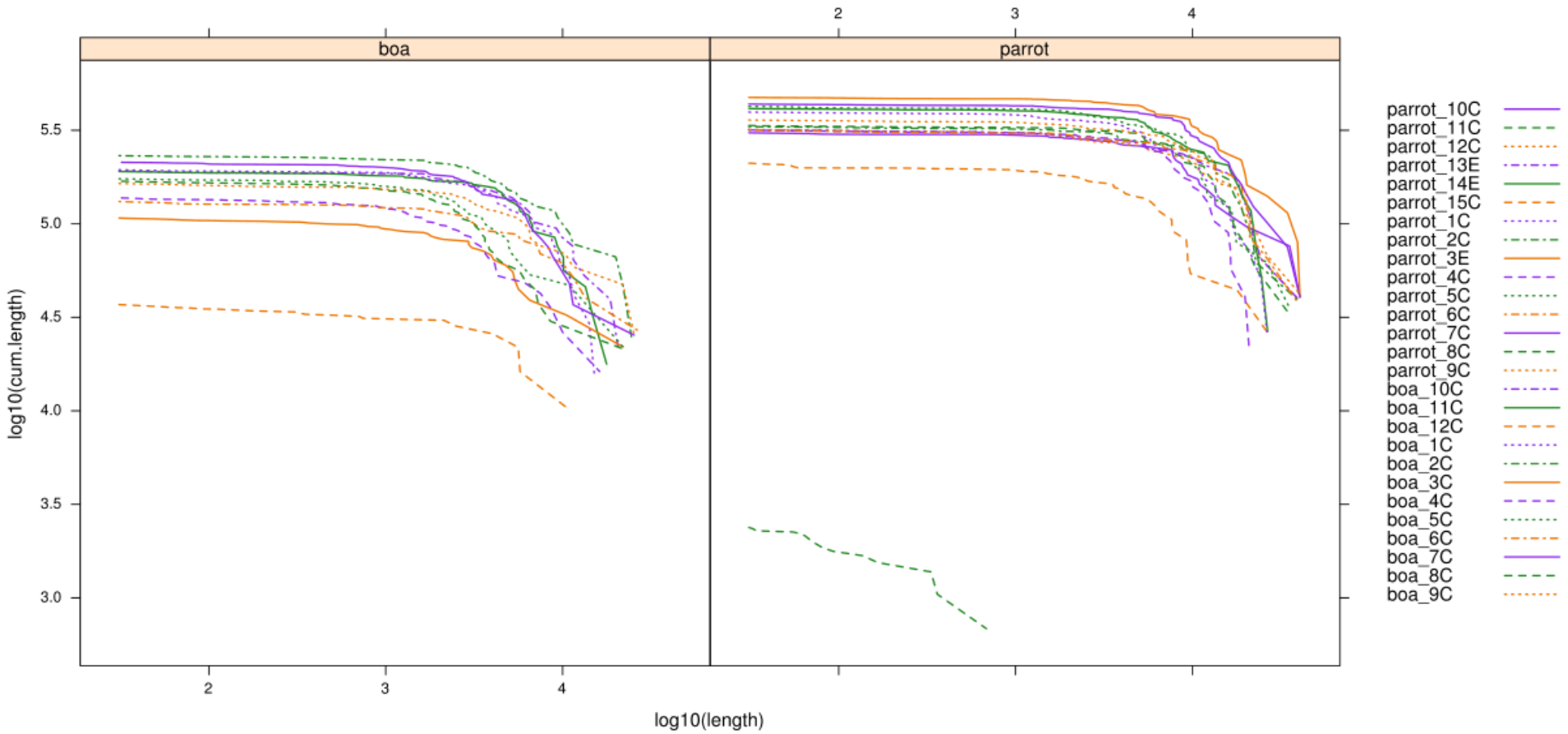
Cumulative Length Plots



(Alignment) Block NG50



Cumulative (Alignment) Length Plots

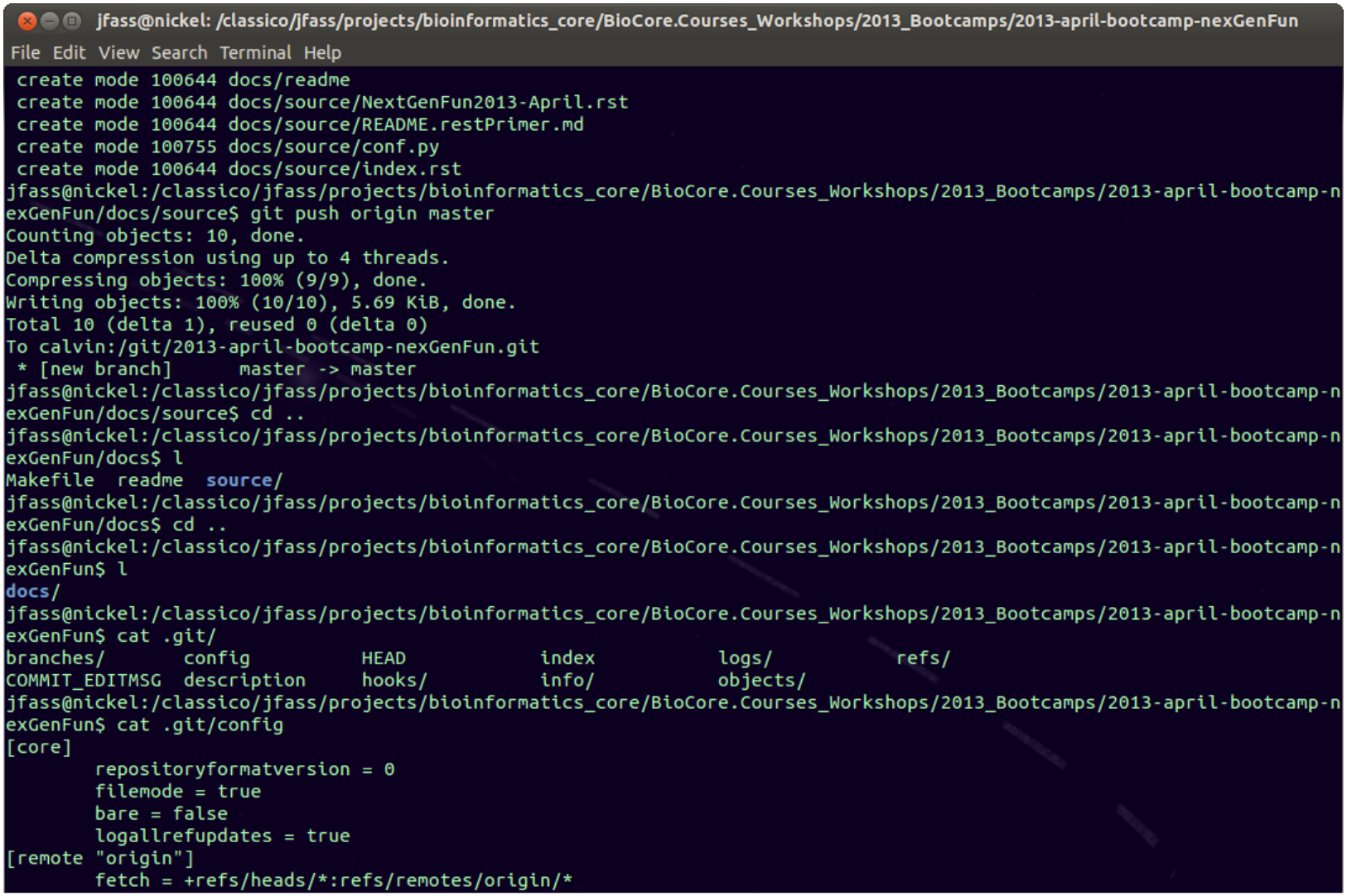


Read-based Assessment

- AMOS, FRCurves, AMOS-validate

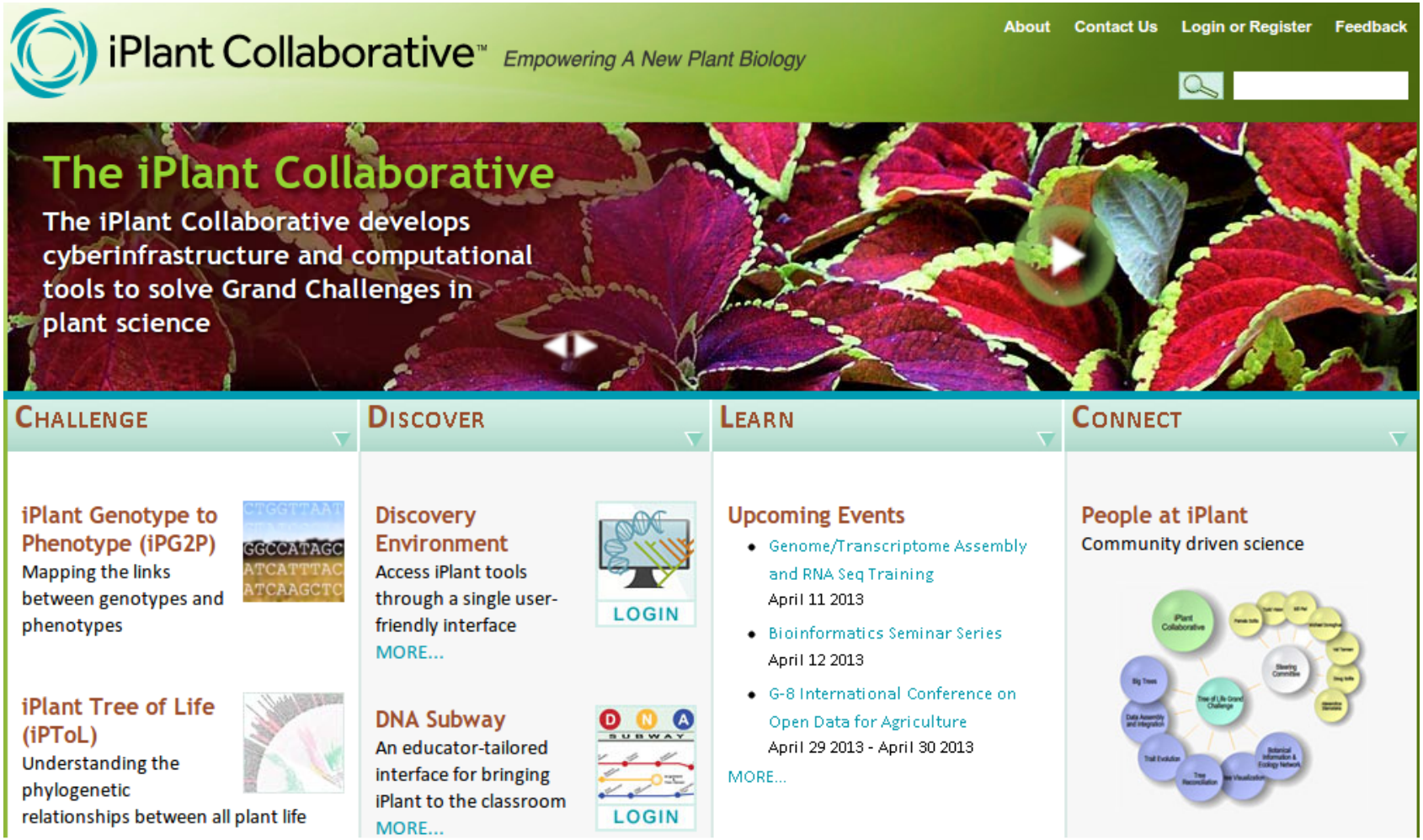
IT Considerations

Linux command line :o



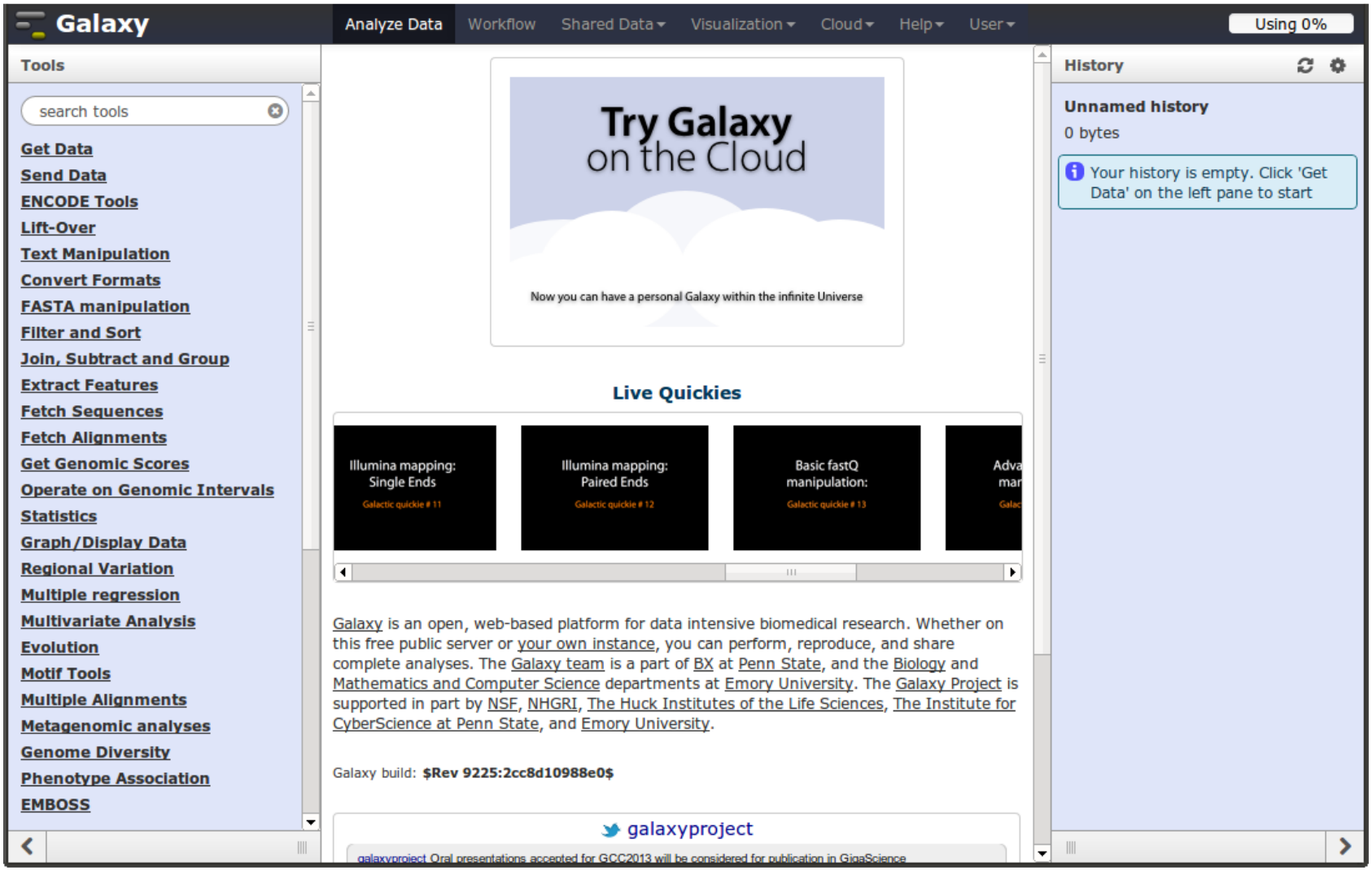
iPlant

http://www.iplantcollaborative.org/



Galaxy

http://usegalaxy.org



Cloud Computing

Cloud computing - either via command line, or stock or customized GUI (e.g. Galaxy) - *may* be a cost effective solution for a smallish lab with light computing requirements.

"If within an entire year one only needs to run their computers for less than 30% of time then cloud computing may be worth it."

~ Istvan Albert (*http://www.biostars.org/p/132/*)

The Big Picture

- Own your own
  - Best for heavy compute user; get what you pay for
  - Fully customizable
  - High training barrier to use
- Public server
  - Often limited compute power, scheduling
  - Not customizable
  - Low training barrier to use
- Cloud servers
  - Some limits on RAM; unlimited nodes / compute
  - Highly customizable
  - High training barrier to use; some "turn-key"-*ish*

Next Gen Project Costs

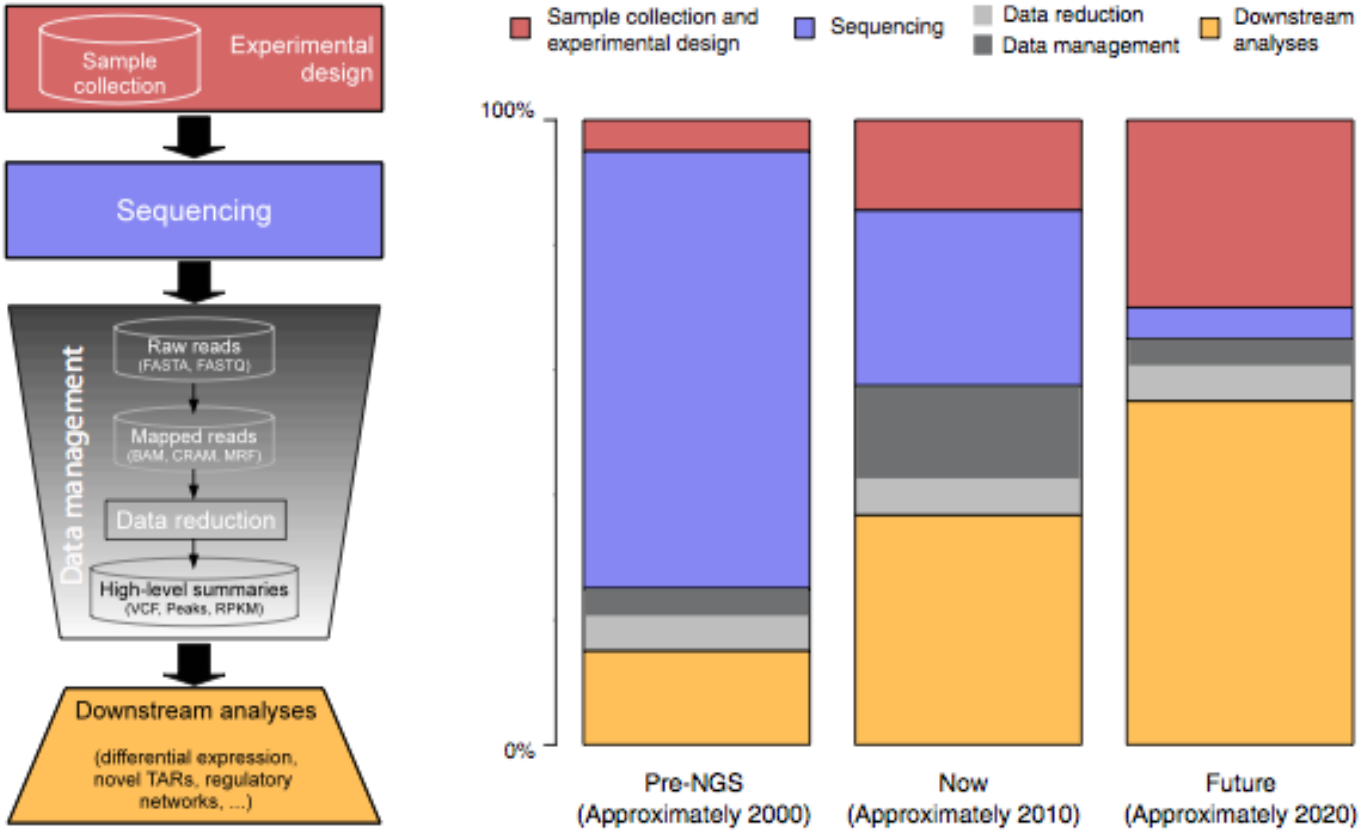


Figure 1. Contribution of different factors to the overall cost of a sequencing project across time. Left, the four-step process: (i) experimental design and sample collection, (ii) sequencing, (iii) data reduction and management, and (iv) downstream analysis. Right, the changes over time of relative impact of these four components of a sequencing experiment. BAM, Binary Sequence Alignment/Map; BED, Browser Extensible Data; CRAM, compression algorithm; MRF, Mapped Read Format; NGS, next-generation sequencing; TAR, transcriptionally active region; VCF, Variant Call Format.

Sboner 2011 Genome Biology 12:125

Project Cost Estimates

- **Don't skimp** on sample replication - biological over technical(?)
- Select technology with limitations / error modes in mind (and cost)
- Coverage
  - Genome *de novo* - 100X; technology dependent (mix long & short?)
  - Transcriptome (euk.) - 30M reads for counting, 100-200M for novel
  - Genome resequencing - 30X
  - ChIP-Seq (euk.) - 10m reads
- Bioinformatics
  - $80/hour (UC) | $123/hour academic | $146 non-academic
  - Preliminary study 20-40 hours (if ~novel application or organism)
  - RNA-Seq two condition ~20 hours
  - Variant discovery w/ effects ~20-40 hours
  - Genome assembly ~50-150 hours
  - Transcriptome assembly ~50-200 hours
- Don't forget about experimental validation!
- Don't forget about data archiving & submission costs (server ~20 hours + 0.5 hour/month) (NCBI submission ~5-15 hours)

Acknowledgements

Bioinformatics Core

Ian Korf (current Core Manager, GC faculty)
Dawei Lin (past Core Manager)

Data Analysis
Joe Fass
Nikhil Joshi
Monica Britton
Jesse Li

Statistical Programming
Blythe Durbin-Johnson

Application Development (Web/DB)
Adam Schaal

System Administration & HPC
Zhi-Wei Lu

Advisory Board

Craig Benham, *chair* (Mathematics)
Gino Cortopassi (Molecular Biosciences)
Vladimir Filkov (Computer Science)
Fredric Gorin (Neurosciences)
Juan Medrano (Animal Sciences)
Jie Peng (Statistics)
David Rocke (Biostatistics)

Genome Center

Director
Richard Michelmore

Associate Directors for Bioinformatics
Ian Korf
Patrice Koehl

Feedback / Evaluation form:

http://tinyurl.com/lg7yehk

Q&A -- fire away!

... and keep in touch

jnfass ~at~ ucdavis ~dot~ edu