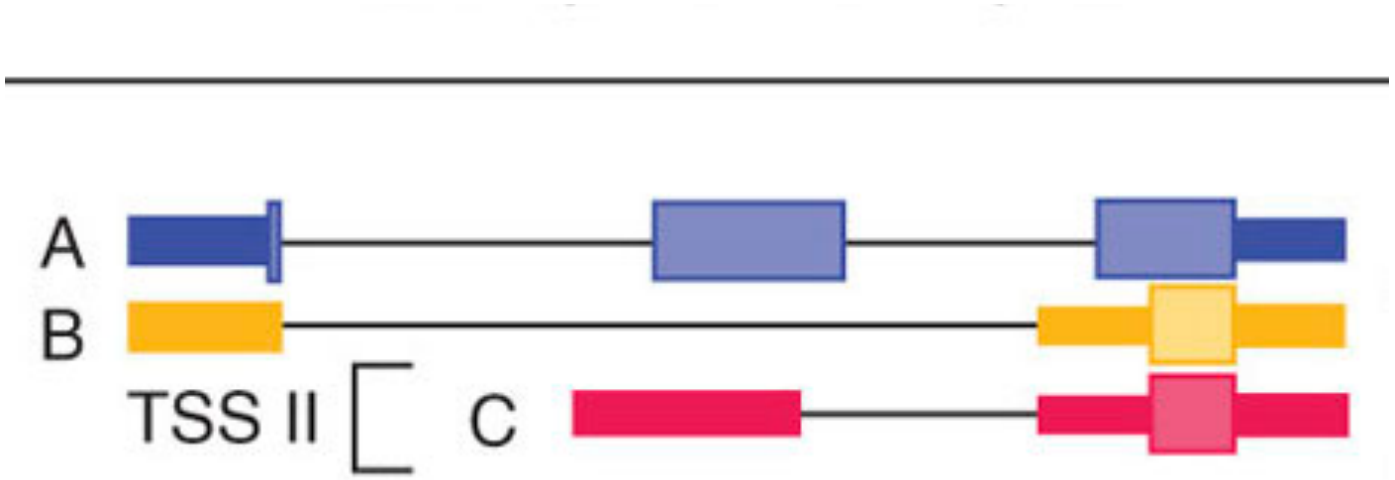
# Tuxedo Pipeline for Novel Transcript Discovery

Dhivya Arasappan

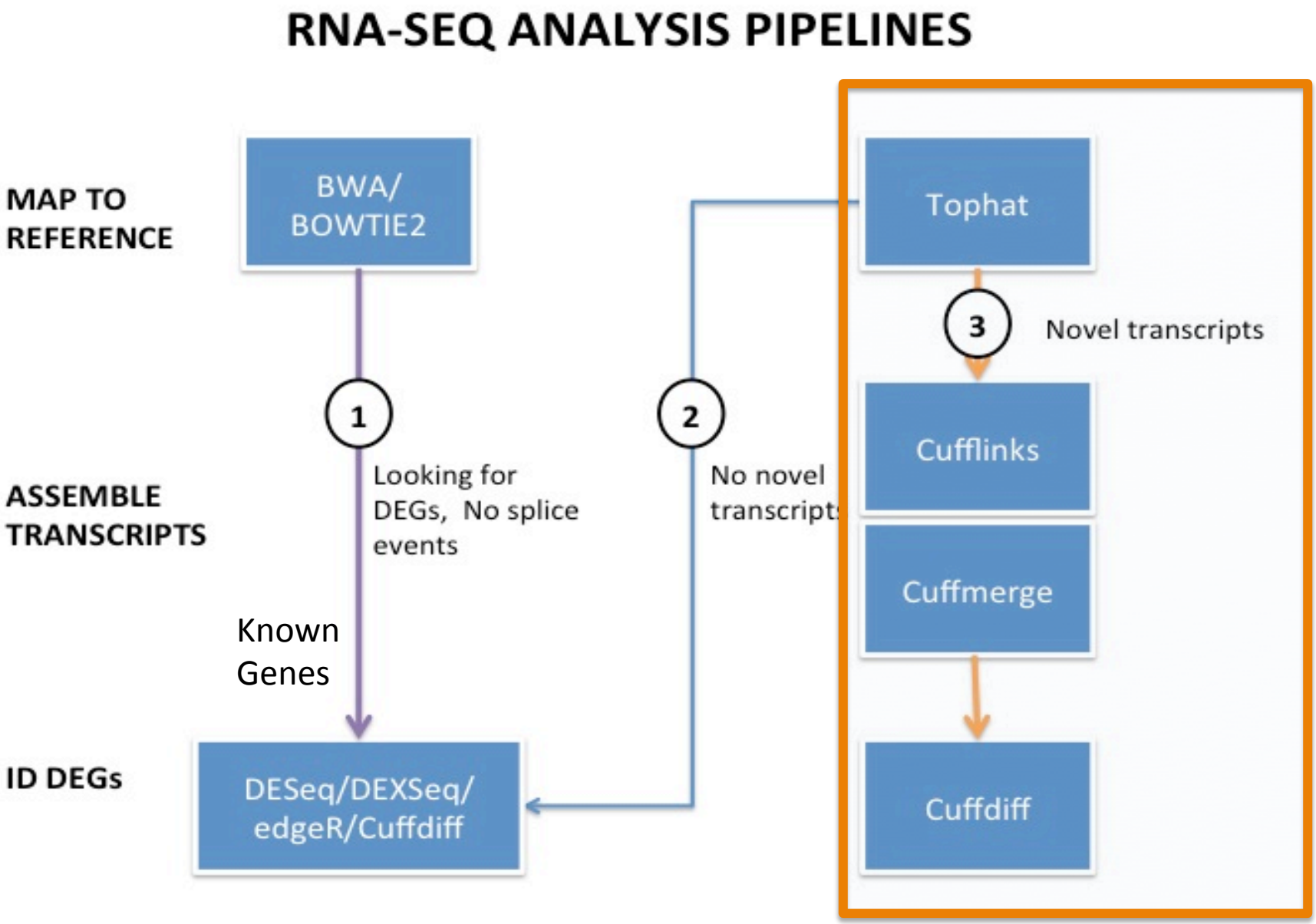
# What is a gene? What is a transcript?

A gene  can have multiple transcripts!

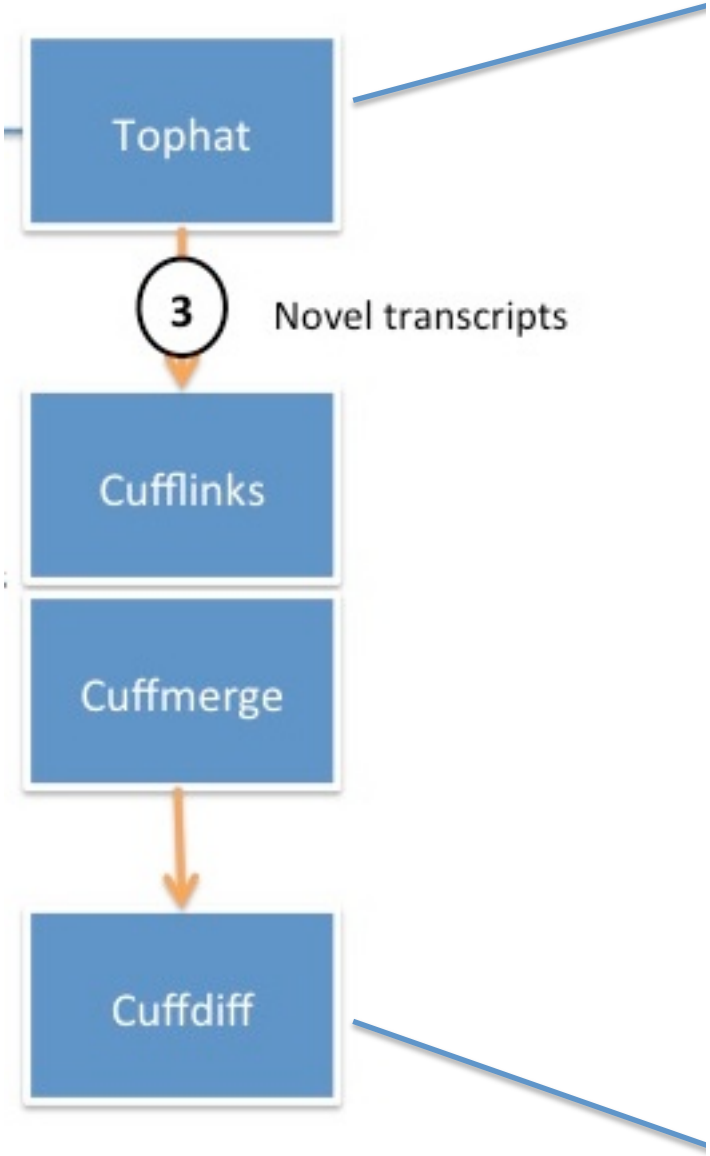



- We want to identify all these transcripts, whether annotated or not.

# Back to the big picture…



# TUXEDO PIPELINE




TopHat
Aligns RNA-Seq reads to the genome using Bowtie
Discovers splice sites

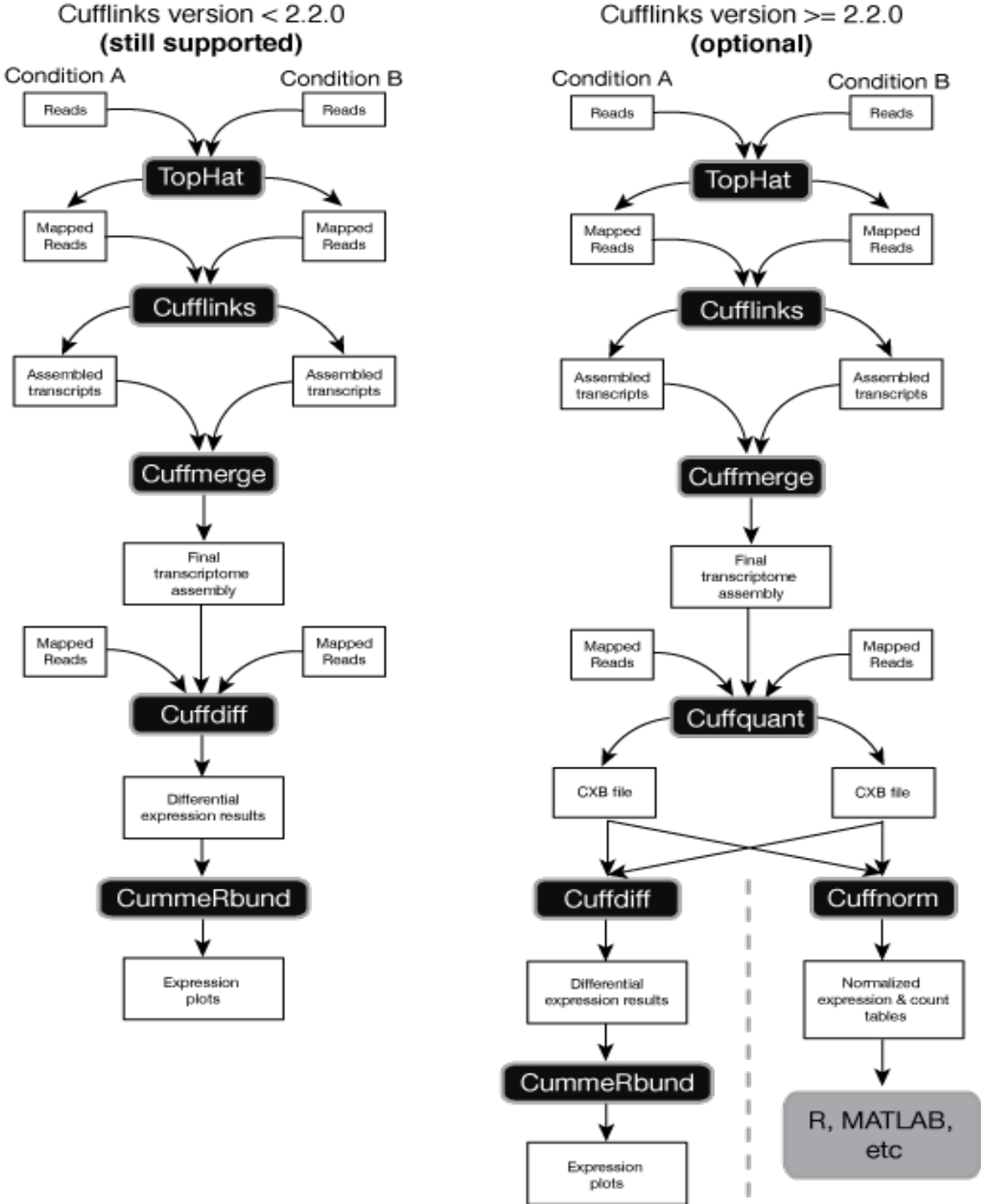
Cufflinks package

Cufflinks
Assembles transcripts

Cuffcompare
Compares transcript assemblies to annotation

Cuffmerge
Merges two or more transcript assemblies

Cuffdiff
Finds differentially expressed genes and transcripts
Detects differential splicing and promoter use




The pipeline is sequential.

Output of one step becomes input of next step.

Figure from:
Trapnell et al, Nature protocols, 2012.

# Of course, Tuxedo Pipeline can be run without looking for novel events

- NO NOVEL JUNCTIONS: Simple differential gene expression analysis against a set of known transcripts.
  - User provides a gff/gtf file containing annotated features. Quantify only the annotated features and id DEGs.

- NOVEL JUNCTIONS ALSO: In addition to known transcripts, novel transcripts should be explored.
  - User provides a gff/gtf file containing annotated features. But you also allow the search for novel variants as well. Both annotated and novel variants are quantified and DEGs are identified.

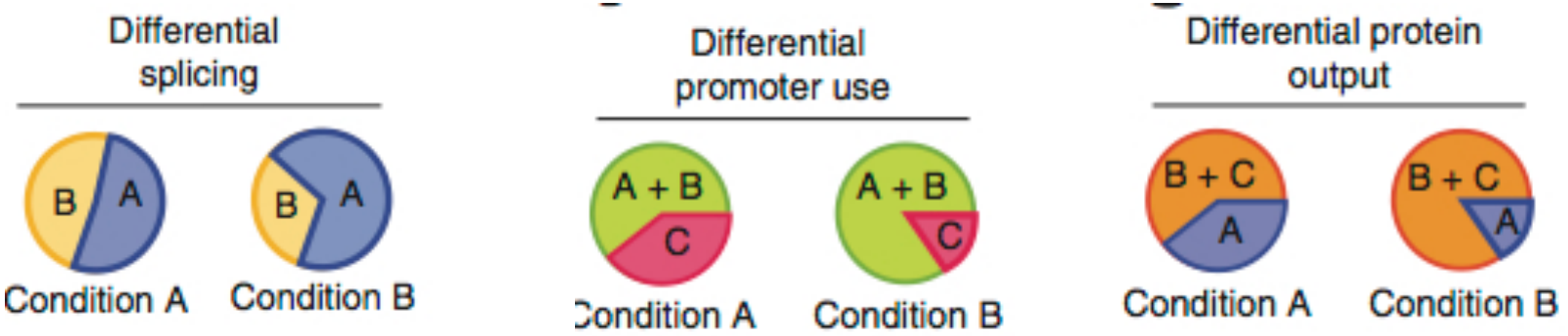
- ONLY NOVEL/*DE NOVO* JUNCTIONS: No gff/gtf file is provided. Using just the read data and the genome reference, construct *de novo* transcripts, quantify them and id DEGs.

# What do we get at the end of running this pipeline?

A view of how the transcriptome is different between condition C1 and condition C2

- Both in terms of annotated genes and transcripts.
- And novel genes and transcripts

Differential gene expression and so much more…



# STEP 1: TOPHAT

What does Tophat do?

Tophat maps your data to your reference in a transcriptome-aware manner, that will also identify junctions. We've already looked at how you can turn on and off its ability to identify novel junctions.

| | |
|---|---|
| **--no-novel-juncs** | Only look for reads across junctions indicated in the supplied GFF file. |
| **-G <GTF/GFF3 file>** | Supply TopHat with a set of gene model annotations and/or known transcripts, as a GTF 2.2 or GFF3 formatted file. |

# STEP 2: CUFFLINKS

- What does cufflinks do?
  - **TRANSCRIPT ASSEMBLY (also referred to as transcriptome reconstruction)**
  - Let's see first what that entails.

# Why transcript assembly?

**Transcript assembly** = assembly of mapped reads into transcriptional units.

**Why?**

- Define a precise map of all transcripts expressed in a sample.
- How does our transcriptome look in comparison to the known transcriptome?
- Look for novel transcripts between conditions/ samples.
- Look for differences in expression for these novel transcripts between conditions/samples.

# Why is transcript assembly hard?

Difficult to tell which read came from which transcript

- Many Short reads, many transcripts!
- Transcripts are expressed in different amounts. So, coverage of reads can be vastly different.
- Reads can come from mature mRNA (exons only) and precursor RNA (containing partial introns).

**Table 1** | Selected list of RNA-seq analysis programs

| Class | Category | Package | Notes | Uses | Input |
|---|---|---|---|---|---|
| **Read mapping** | | | | | |
| Unspliced aligners[a] | Seed methods | Short-read mapping package (SHRiMP)[41] | Smith-Waterman extension | Aligning reads to a reference transcriptome | Reads and reference transcriptome |
| | | Stampy[39] | Probabilistic model | | |
| | Burrows-Wheeler transform methods | Bowtie[43] | | | |
| | | BWA[44] | Incorporates quality scores | | |
| Spliced aligners | Exon-first methods | MapSplice[52] | Works with multiple unspliced aligners | Aligning reads to a reference genome. Allows for the identification of novel splice junctions | Reads and reference genome |
| | | SpliceMap[50] | | | |
| | | TopHat[51] | Uses Bowtie alignments | | |
| | Seed-extend methods | GSNAP[53] | Can use SNP databases | | |
| | | QPALMA[54] | Smith-Waterman for large gaps | | |
| **Transcriptome reconstruction** | | | | | |
| Genome-guided reconstruction | Exon identification | G.Mor.Se | Assembles exons | Identifying novel transcripts using a known reference genome | Alignments to reference genome |
| | Genome-guided assembly | Scripture[28] | Reports all isoforms | | |
| | | Cufflinks[29] | Reports a minimal set of isoforms | | |
| Genome-independent reconstruction | Genome-independent assembly | Velvet[61] | Reports all isoforms | Identifying novel genes and transcript isoforms without a known reference genome | Reads |
| | | TransABySS[56] | | | |
| **Expression quantification** | | | | | |
| Expression quantification | Gene quantification | Alexa-seq[47] | Quantifies using differentially included exons | Quantifying gene expression | Reads and transcript models |
| | | Enhanced read analysis of gene expression (ERANGE)[20] | Quantifies using union of exons | | |
| | | Normalization by expected uniquely mappable area (NEUMA)[82] | Quantifies using unique reads | | |
| | Isoform quantification | Cufflinks[29] | Maximum likelihood estimation of relative isoform expression | Quantifying transcript isoform expression levels | Read alignments to isoforms |
| | | MISO[33] | | | |
| | | RNA-seq by expectaion maximization (RSEM)[69] | | | |
| Differential expression | | Cuffdiff[29] | Uses isoform levels in analysis | Identifying differentially expressed genes or transcript isoforms | Read alignments and transcript models |
| | | DegSeq[79] | Uses a normal distribution | | |
| | | EdgeR[77] | | | |
| | | Differential Expression analysis of count data (DESeq)[78] | | | |
| | | Myrna[75] | Cloud-based permutation method | | |

Figure :
Garber et al, Nature Methods, 2011

**Most commonly used, if you have a genome.**

**Less resource-intensive**

We'll call this graph based approach

| Transcriptome reconstruction | | | | |
|---|---|---|---|---|
| Genome-guided reconstruction | Exon identification | G.Mor.Se | Assembles exons | Identifying novel transcripts using a known reference genome |
| | Genome-guided assembly | Scripture[28] | Reports all isoforms | |
| | | Cufflinks[29] | Reports a minimal set of isoforms | |
| Genome-independent reconstruction | Genome-independent assembly | Velvet[61] | Reports all isoforms | Identifying novel genes and transcript isoforms without a known reference genome |
| | | TransABySS[56] | | |

**If you don't have a genome.**
**If you believe your sample has major rearrangements**

**More CPU and RAM intensive**

Figure :
Garber et al, Nature Methods, 2011

# Genome guided transcript assesmbly

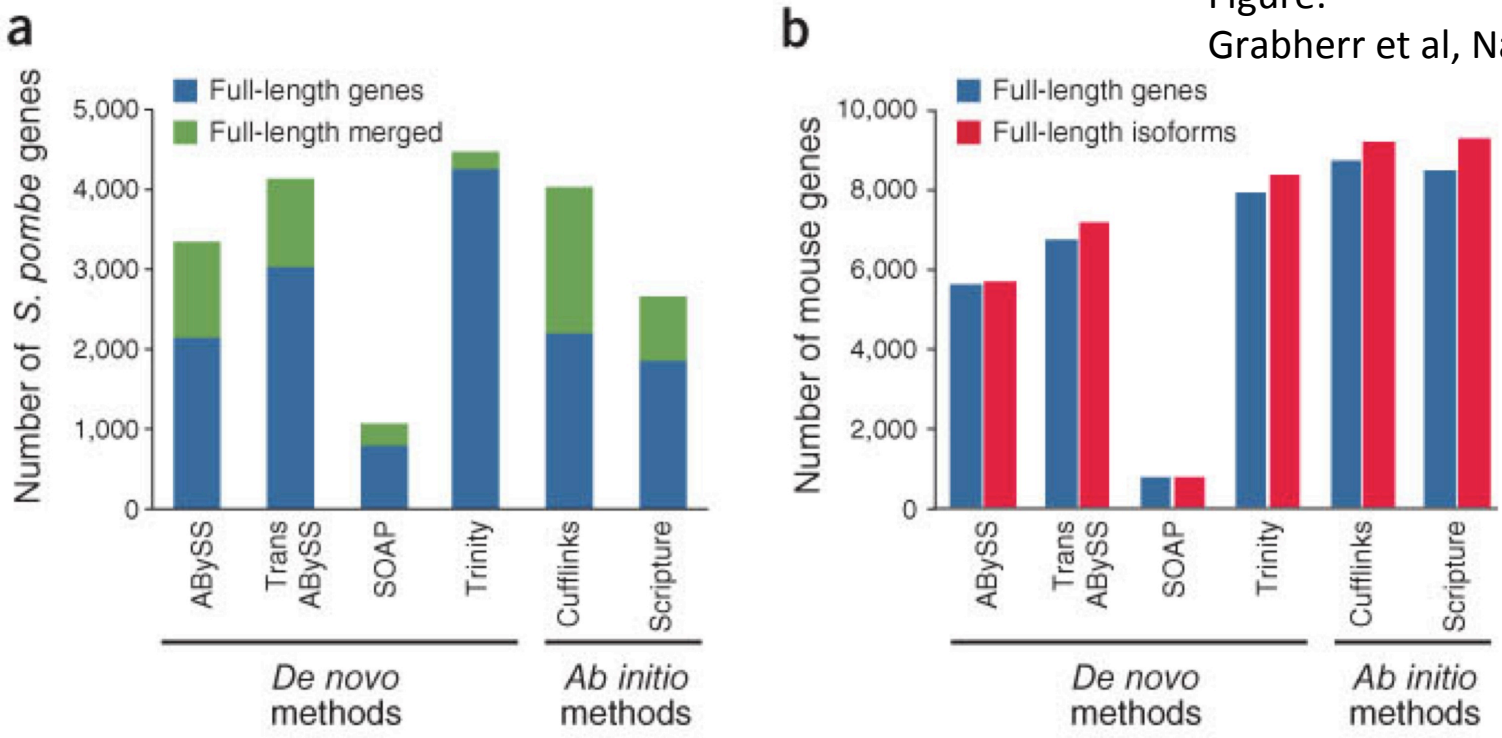
Different assembly methods

- **Exon Identification Method**
  - First ID putative exons by looking for coverage islands.
  - Older method, were meant for shorter read lengths.
  - G.MorSe

- **Graph Based approach**
  - Directly uses mappings of spliced reads to reconstruct transcriptome.
  - Uses graph topology.
  - **Cufflinks (part of tuxedo suite),** scripture

# How do these tools compare?

Figure:
Grabherr et al, Nature Biotech, 2011



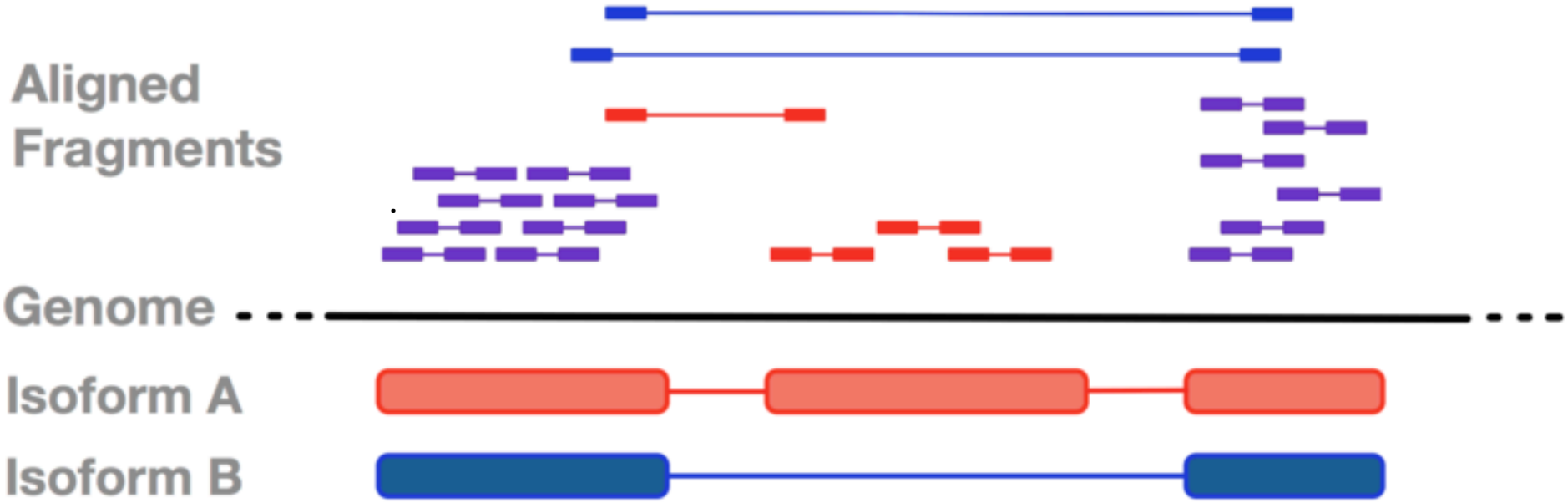

| Program combination | *vs.* orthology annotation | | *vs.* EST annotation | |
|---|---|---|---|---|
| | Base-level accuracy (%)[1] | Confirmed junctions (%)[1] | Base-level accuracy (%)[1] | Confirmed junctions (%)[1] |
| TopHat + Cufflinks | 83.9 | 75.8 | 68.9 | 63.0 |
| GSNAP + Cufflinks | 79.4 | 71.2 | 65.7 | 58.4 |
| GSNAP + Cufflinks (subsample[2]) | 80.3 | 72.7 | 60.2 | 66.3 |
| TopHat + Scripture | 70.3 | 67.9 | 60.8 | 62.5 |

[1]Base level accuracy and percentage of confirmed junctions with different combinations of mapper and assembler on the sample ps94 males compared to the orthology annotation and the EST annotation ([2]based on 48 M reads).
doi:10.1371/journal.pone.0046415.t001

Figure:
Palmieri et al, PLOS One, 2012

# How does Cufflinks do transcript assembly

Graph based approach!



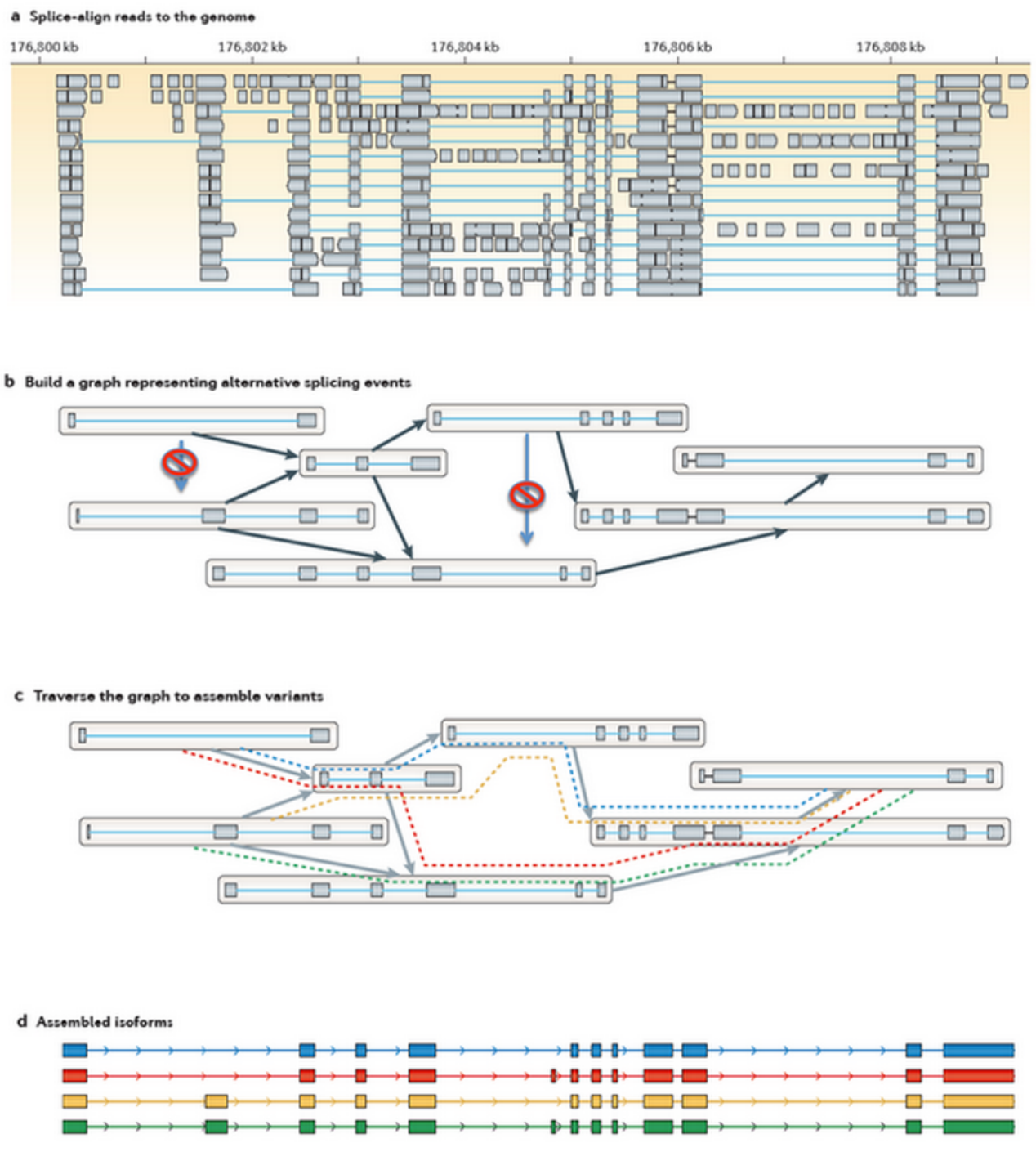
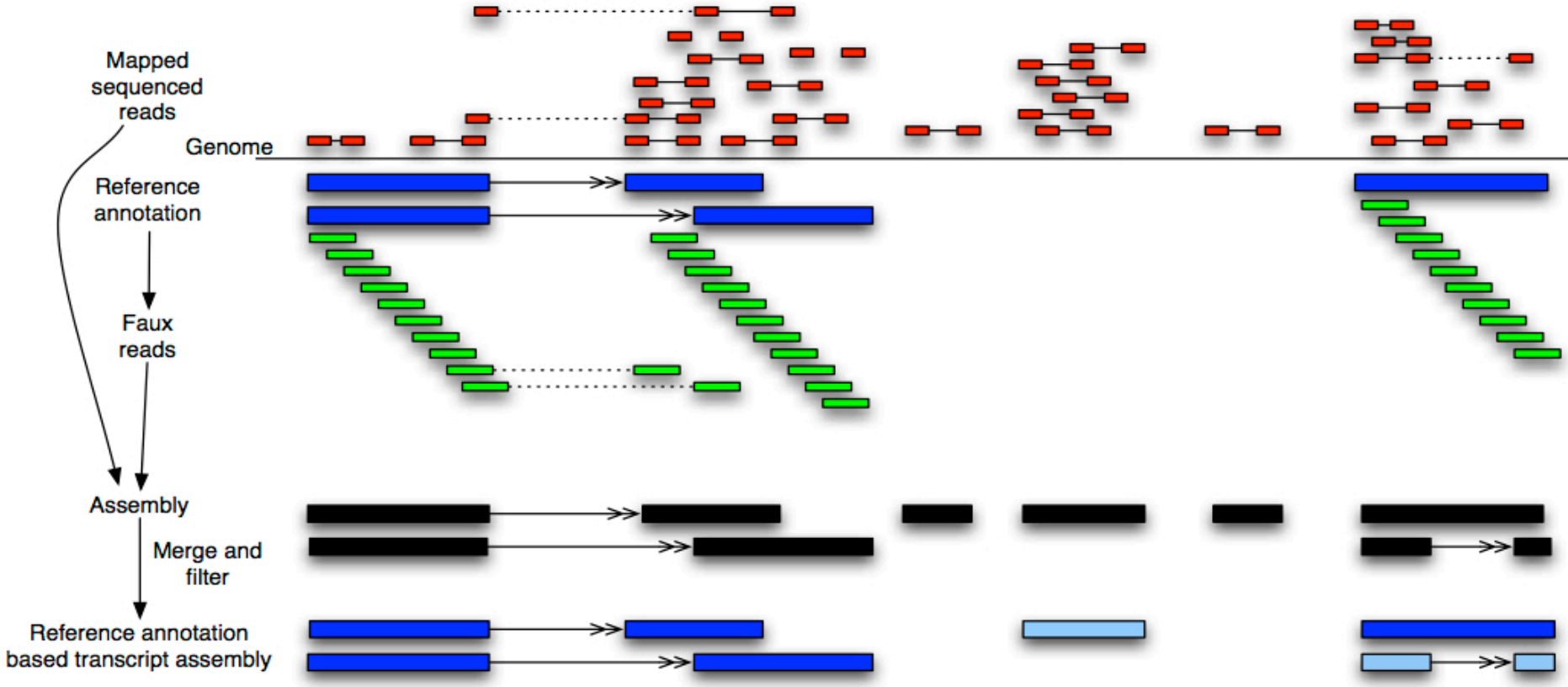



Figure :
http://sourceforge.net/projects/trinityrnaseq/files/misc/RNASEQ_WORKSHOP/rnaseq_workshop_slides.pdf

# RABT

- Reference annotation based transcript assembly (RABT)
  - Uses existing annotation to guide assembly of transcripts.



# After assembly

- Calculates abundance for these assembled transcripts.

- Normalized using FPKM (Fragments Per Kilobase of Exon Per Million) (variation of RPKM)
  - RPKM normalizes for **transcript length variations** and **sequencing depth**.
  - RPKM= (No.of Mapped reads*10^9)/ (**length of transcript *total no.of reads**)
  - FPKM just exchanges reads with fragments.

# General syntax for cufflinks command

cufflinks [options] <accepted_hits.bam>

Some of the important options:

-p/--num-threads

-G/--GTF (quantify only annotated transcripts)

-g/--GTF-guide (both annotated and novel transcripts)

-b/--frag-bias-correct

-u/--multi-read-correct

# General syntax for cufflinks command

-b/--frag-bias-correct

When quantifying abundance, corrects for sequence-specific bias at the ends of reads by 'learning' from the data.

-u/--multi-read-correct

By default, if a read maps to 2 genes it will count as 50% (half a read) towards each gene.

With this flag, it handles this question in a more fine-tuned manner.

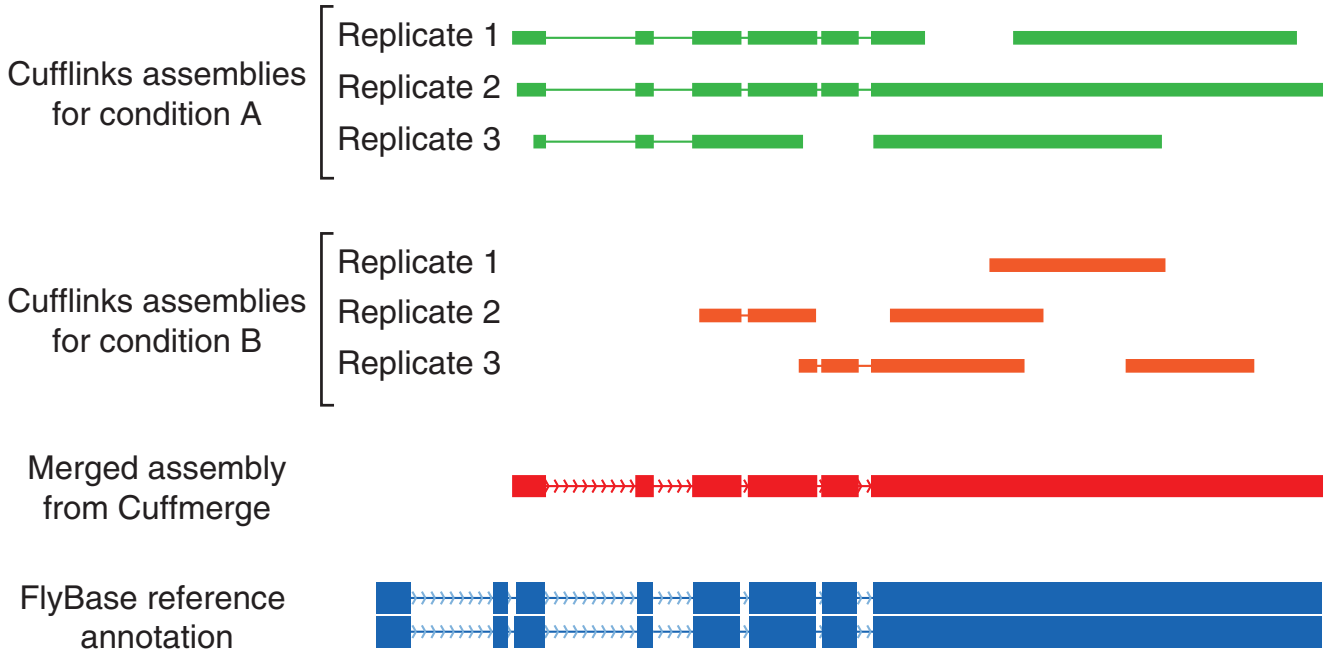
# Let's look at some results from a cufflinks transcript assembly

- Input:
  - Tophat mapped results (bam files)
  - Transcriptome annotation (genes.gtf)

- Let's SWITCH TO THE WIKI for instructions on looking at these results…

# STEP 3: CUFFMERGE

- Cuffmerge is used to merge all the transcripts that cufflinks assembled into one file.



# STEP 3: CUFFMERGE

- Input: All cufflinks assembly files (in gtf format)
- Input: Optionally: Annotated genes (in gff/gtf format)

- Compares your assembled transcripts to a reference annotation.

- Output: merged.gtf
  - Your very own gtf file, containing all the transcripts found in your samples (both novel and otherwise).

- SWITCH TO THE WIKI for instructions on viewing these results

# STEP 4: CUFFQUANT

- Optional but recommended step

- Computes gene and isoform expression quantification values and stores them in a structure that can be used by cuffdiff or cuffnorm.

- Input: result from cuffmerge
- Output: abundances files

- SWITCH TO THE WIKI for instructions on viewing these commands and results

# STEP 5: CUFFDIFF

- Calculates differential expression!

- Input:
  - Our newly created merged.gtf file or  a gtf file we downloaded (genes.gtf)
  - Our newly created cuffquant abundances file

  - Our newly created merged.gtf file or  a gtf file we downloaded (genes.gtf)
  - Mapped bam files

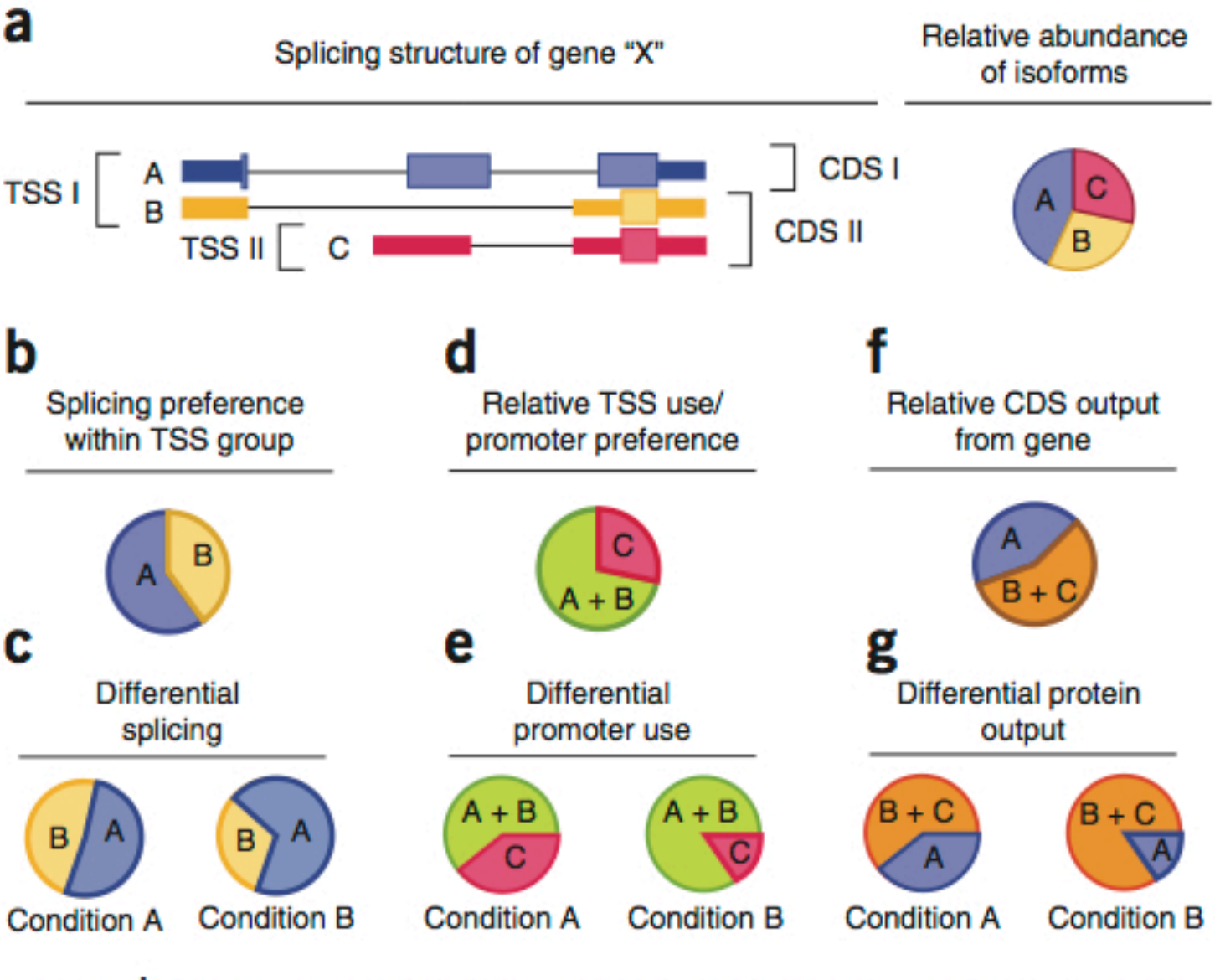
- Counts the number of fragments(reads) generated by each isoform to obtain isoform-level expression.
- Calculates difference in isoform-level expression among conditions.
- If the chance of seeing this difference is small enough under the chosen statistical model, it is deemed  signficantly differentially expressed.

# Other differential expression tools vs cuffdiff

| Others | Cuffdiff |
| --- | --- |
| **Raw count method** for assigning counts to genes | **Isoform deconvolution method** for assigning counts to genes |
| Count the reads mapping to **exons** of each gene/normalization factor = expression for gene | Count the reads that map to each **isoform** of the gene/normalization factor = expression for gene |
| If all isoforms of the gene are up/down, works fine | If all isoforms of the gene are up/down, works fine |
| If some isoforms of the gene are up and some are down, **inaccurate results** | If some isoforms of the gene are up and some are down, **works fine** |

# STEP 5: CUFFDIFF

Figure from: Differential analysis of gene regulation at transcript resolution with rNA-seq, Trapnell et al, Nature Biotechnology, 2013



# STEP 5: CUFFDIFF

- SWITCH TO THE WIKI for instructions on viewing these results

# Limitations of the Tuxedo Pipeline

- A Reference is needed.

- Not quick.
  - For a human dataset with say, 60 million reads, each step can take 12-24 hours on lonestar and on stampede, probably 6-12 hours.
  - Some steps (cufflinks, cuffdiff) can run out of memory on large jobs.

# DESeq/edgeR output vs Tuxedo pipeline output

- Yesterday we generated differential expressed genes too. So, why the big fuss?
  - They were all from annotated genes. So, they all has flybase ids.
  - Now our output has genes with ids 'CUFF…' - they are novel.
  - In addition to differential gene expression, we also have results for differential regulation.
  - We also have results telling us where our novel transcripts are with respect to the annotated ones.

# If You Don't Have A Genome

**Transcriptome reconstruction**

| | | | | |
|---|---|---|---|---|
| Genome-guided reconstruction | Exon identification<br>Genome-guided assembly | G.Mor.Se<br>Scripture[28]<br>Cufflinks[29] | Assembles exons<br>Reports all isoforms<br>Reports a minimal set of isoforms | Identifying novel transcripts using a known reference genome |
| Genome-independent reconstruction | Genome-independent assembly | Velvet[61]<br>TransABySS[56] | Reports all isoforms | Identifying novel genes and transcript isoforms without a known reference genome |

- PRE SEQUENCING: Consider using a normalized cDNA library
- POST SEQUENCING : Trinity for assembly
  - Wrapper script to parallelize some parts of trinity: https://wikis.utexas.edu/display/bioiteam/assemble_trinity
  - assemble_trinity -a <your_allocation> -l <R1_reads.fq> -r <R2_reads.fq> -o <output_directory>
- Annotate using trinotate or Blast2GO
- Map reads to the assembled transcriptome and simply quantify transcripts by parsing the SAM file:
  - Samtools idxstats samfile
  - cut –f 3 samfile|sort|uniq -c

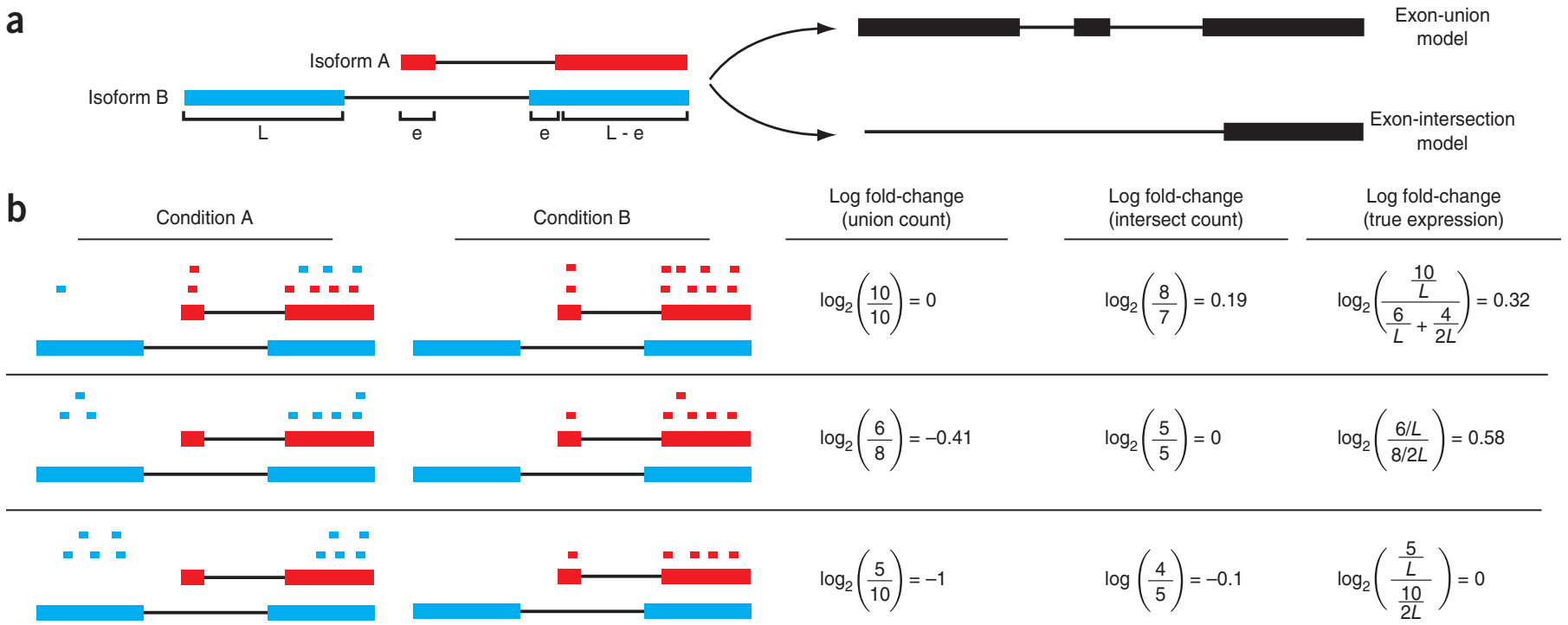



**Figure 1** Changes in fragment count for a gene does not necessarily equal a change in expression. (**a**) Simple read-counting schemes sum the fragments incident on a gene's exons. The exon-union model counts reads falling on any of a gene's exons, whereas the exon-intersection model counts only reads on constitutive exons. (**b**) Both of the exon-union and exon-intersection counting schemes may incorrectly estimate a change in expression in genes with multiple isoforms. The true expression is estimated by the sum of the length-normalized isoform read counts. The discrepancy between a change in the union or intersection count and a change in gene expression is driven by a change in the abundance of the isoforms with respect to one another. In the top row, the gene generates the same number of reads in conditions A and B, but in condition B, all of the reads come from the shorter of the two isoforms, and thus the true expression for the gene is higher in condition B. The intersection count scheme underestimates the true change in gene expression, and the union scheme fails to detect the change entirely. In the middle row, the intersection count fails to detect a change driven by a shift in the dominant isoform for the gene. The union scheme detects a shift in the wrong direction. In the bottom row, the gene's expression is constant, but the isoforms undergo a complete

Figure from: Differential analysis of gene regulation at transcript resolution with rNA-seq, Trapnell et al, Nature Biotechnology, 2013