Short lecture
– How to make a transcriptome

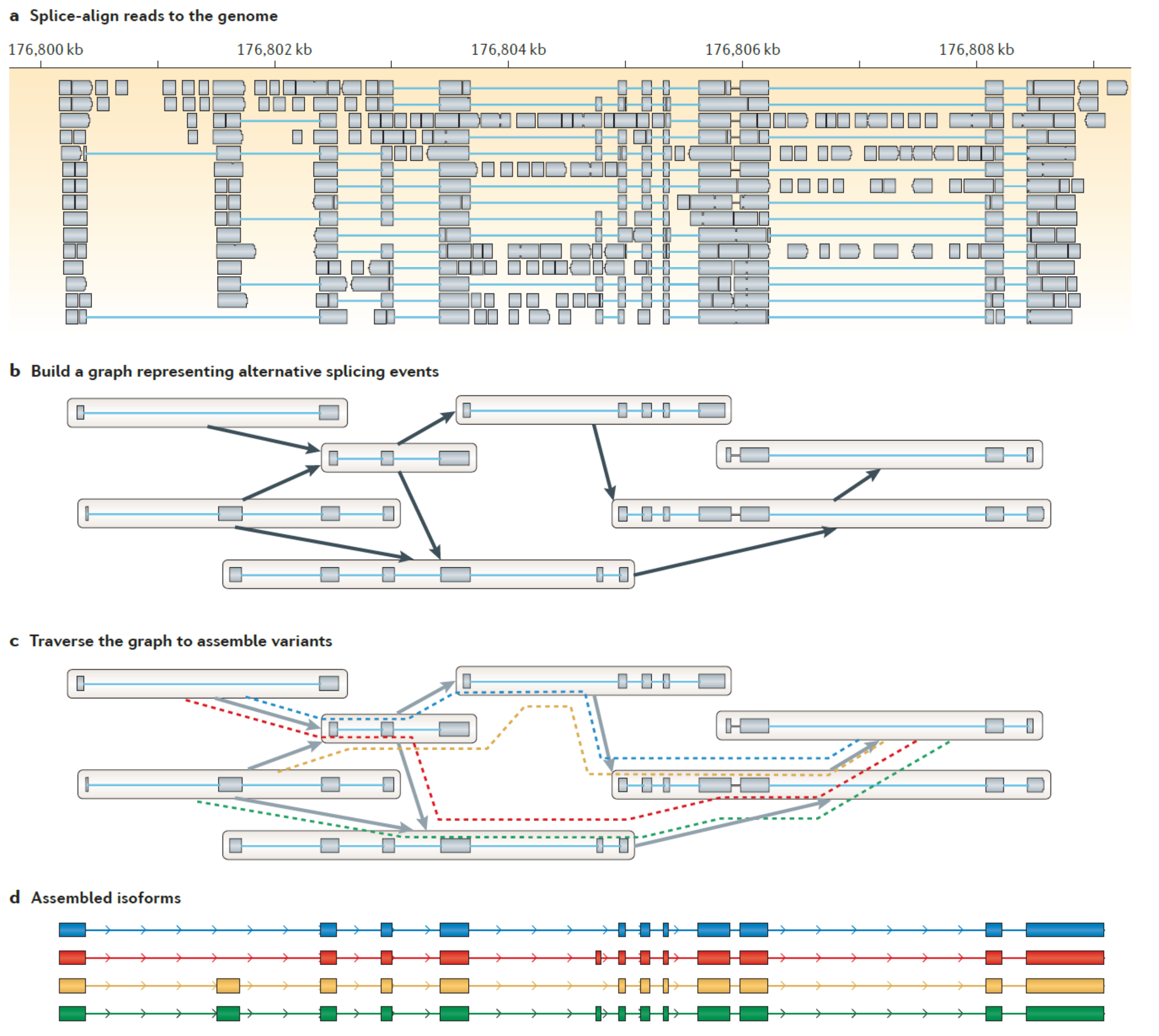
What to consider I

- What do I want?
- What will I use it for?
- Which resources are available for your species (very closely related species)?
- What kind of data do I have?
- 2n or xn ploidy?

The strategies I

- Reference based (*ab initio*)
  - Maps RNAseq reads back towards reference genome and builds transcripts
  - Needs a certain amount of splice-junction covering reads
- *De novo* (with/without genome guiding)
  - Assembly of RNAseq reads only
  - Guided: reads are clustered according to chromosome / scaffold prior to assembly
- Mixed approach
  - Merging several assemblies to one

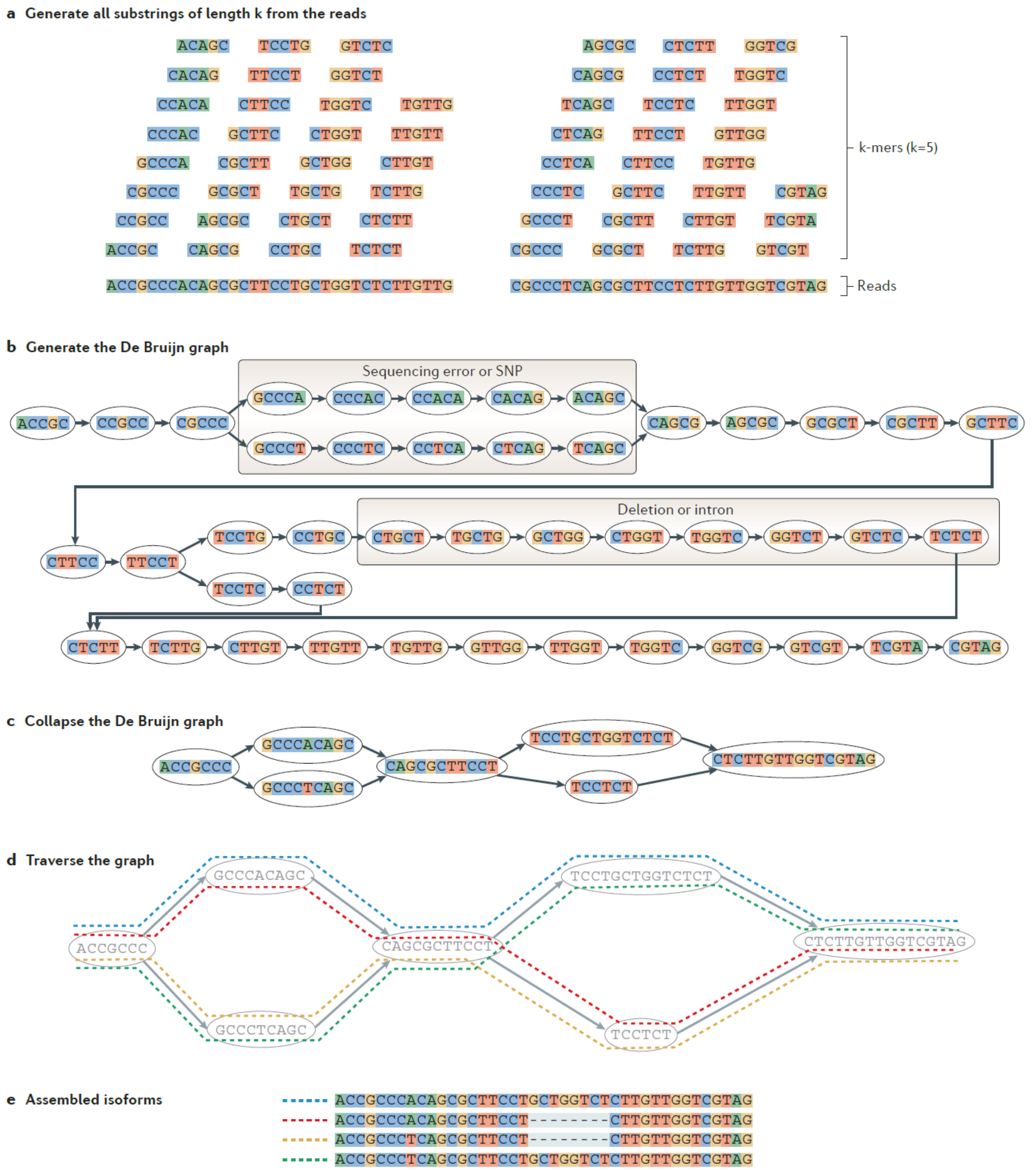
Reference based


a Splice-align reads to the genome
176,800 kb
176,802 kb
176,804 kb
176,806 kb
176,808 kb
b Build a graph representing alternative splicing events
c Traverse the graph to assemble variants
d Assembled isoforms

Reference based II

- Benefits
  - Time efficient / single computer job
  - Requires less coverage of samples
  - Artefacts / contaminations does not align to the reference
  - Low abundance / novel isoforms are resolved
- Complications
  - Depends on quality of reference
  - Gene dense organisms
  - Higher eukaryotes with complex splice variants – especially *trans*-splicing
  - Software settings may discard splice variants / transcripts
  - Different treatment of multi-mapping reads

De novo


a Generate all substrings of length k from the reads
ACAGC TCCTG GTCTC
CACAG TTCCT GGTCT
CCACA CTTCC TGGTC TGTTG
CCCAC GCTTC CTGGT TTGTT
GCCCA CGCTT GCTGG CTTGT
CGCCC GCGCT TGCTG TCTTG
CCGCC AGCGC CTGCT CTCTT
ACCGC CAGCG CCTGC TCTCT
AGCGC CTCTT GGTCG
CAGCG CCTCT TGGTC
TCAGC TCCTC TTGGT
CTCAG TTCCT GTTGG
CCTCA CTTCC TGTTG
CCCTC GCTTC TTGTT CGTAG
GCCCT CGCTT CTTGT TCGTA
CGCCC GCGCT TCTTG GTCGT
k-mers (k=5)
ACCGCCCACAGCGCTTCCTGCTGGTCTCTTGTTG
CGCCCTCAGCGCTTCCTCTTGTTGGTCGTAG
Reads
b Generate the De Bruijn graph
Sequencing error or SNP
ACCGC CCGCC CGCCC
GCCCA CCCAC CCACA CACAG ACAGC
GCCCT CCCTC CCTCA CTCAG TCAGC
CAGCG AGCGC GCGCT CGCTT GCTTC
Deletion or intron
CTTCC TTCCT
TCCTG CCTGC
CTGCT TGCTG GCTGG CTGGT TGGTC GGTCT GTCTC TCTCT
TCCTC CCTCT
CTCTT TCTTG CTTGT TTGTT TGTTG GTTGG TTGGT TGGTC GGTCG GTCGT TCGTA CGTAG
c Collapse the De Bruijn graph
ACCGCCC
GCCCACAGC
GCCCTCAGC
CAGCGCTTCCT
TCCTGCTGGTCTCT
TCCTCT
CTCTTGTTGGTCGTAG
d Traverse the graph
ACCGCCC
GCCCACAGC
GCCCTCAGC
CAGCGCTTCCT
TCCTGCTGGTCTCT
TCCTCT
CTCTTGTTGGTCGTAG
e Assembled isoforms
ACCGCCCACAGCGCTTCCTGCTGGTCTCTTGTTGGTCGTAG
ACCGCCCACAGCGCTTCCT--------CTTGTTGGTCGTAG
ACCGCCCTCAGCGCTTCCT--------CTTGTTGGTCGTAG
ACCGCCCTCAGCGCTTCCTGCTGGTCTCTTGTTGGTCGTAG

De novo II

- Benefits
  - No reference needed
  - Detects all transcripts (coverage dependent)
  - No knowledge/prediction of splice sites needed
  - Complex splice patterns can be resolved
- Complications
  - Requires lots of computing power
  - Requires more coverage to resolve transcripts
  - Sensitive to read errors and artefacts / contaminations
  - Paralog (“gene copies”) resolution is an issue

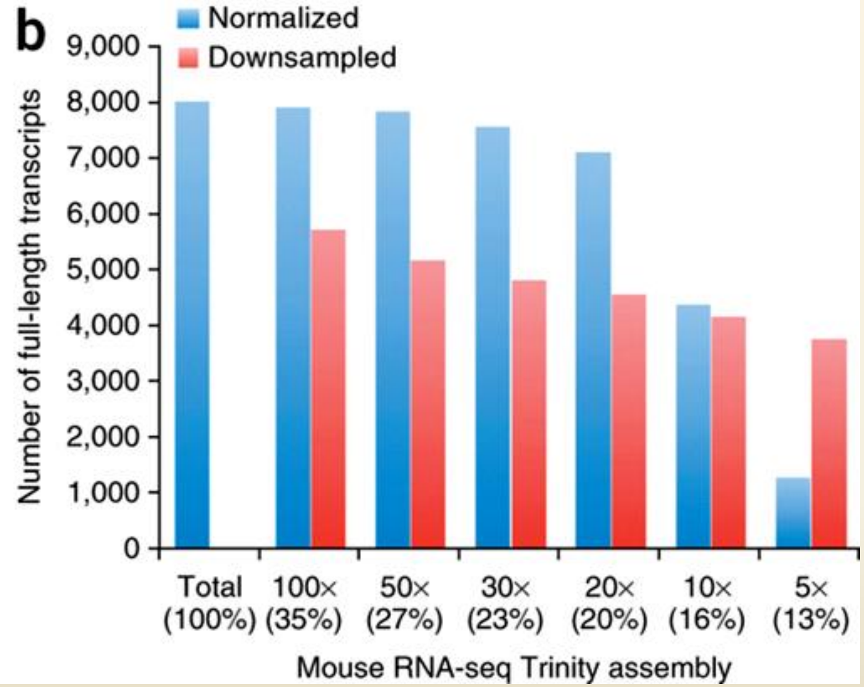
Mixed approach

- *De novo* and *ab initio* assembly concatenation
- Multiple kmer strategy

- Who benefits from a mixed approach?
  - Gene dense eukaryotes
  - Polyploid species
  - When the aim is to make a really good reference transcriptome

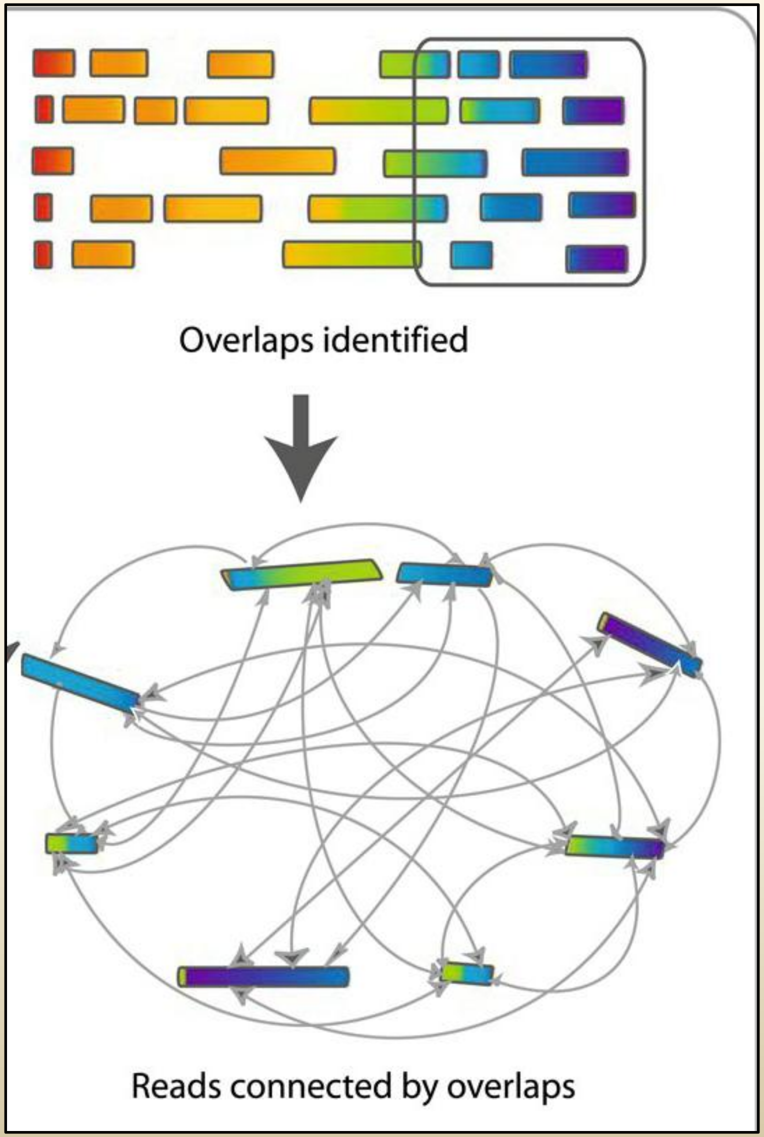
How to make it I

- Use all available data
- Consider normalization to shorten computation time and increase chances of resolving less abundant transcripts



How to make it II

- Consider the assembly algorithm
  - Large datasets with short reads benefits from using De bruijn graph based assembly programs (more than a hundred million read pairs)
  - Small datasets with short reads benefits from using Overlap-Layout-Consensus (OLC) based assembly programs



www.biologicalproceduresonline.com

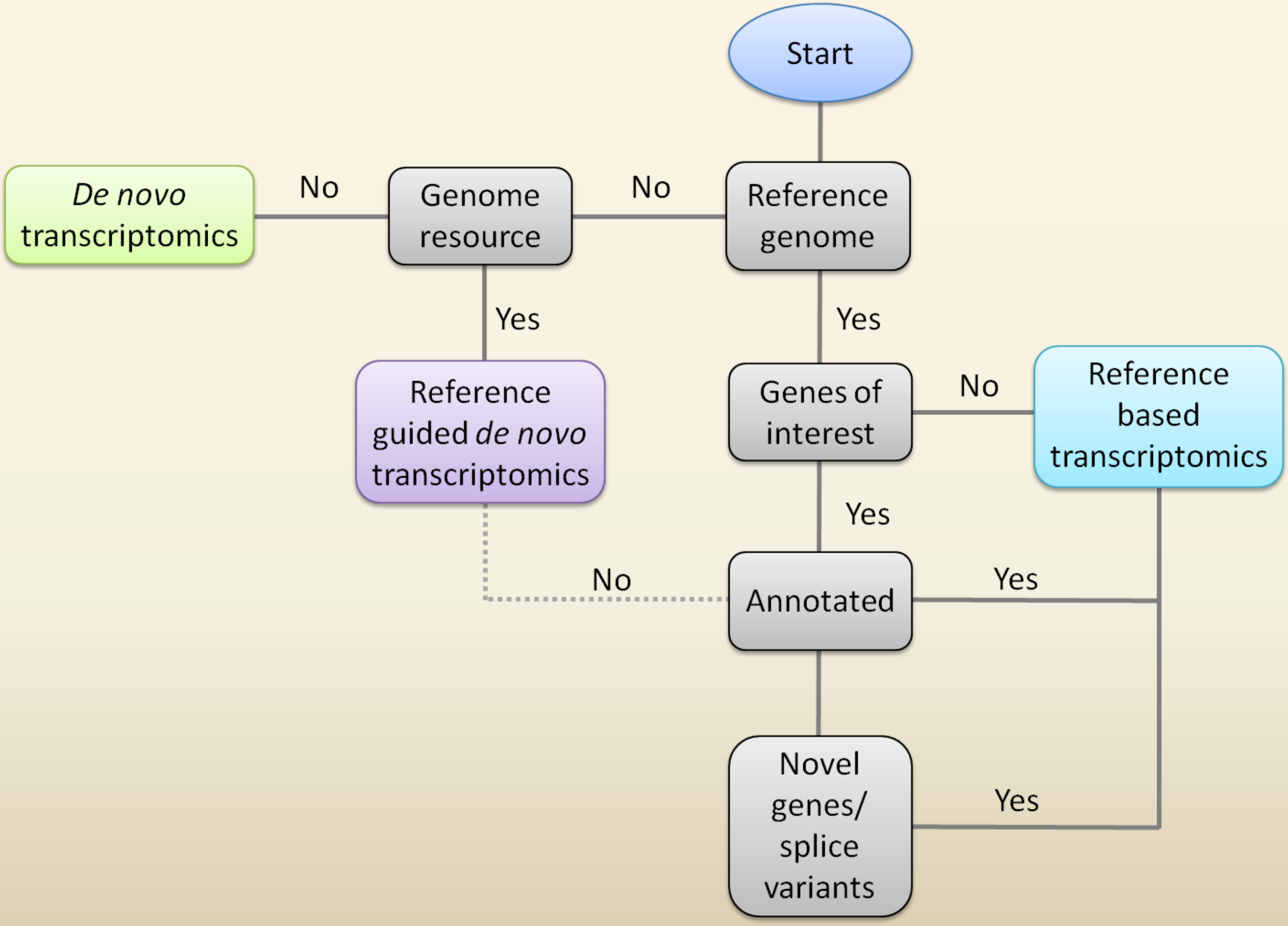
How to make III

- Do you have a an organism known to be gene dense with overlapping UTRs?
  - Select a program with options like jaccard clip to improve algorithm
  - The cost is more computation time so do not use it unless necessary

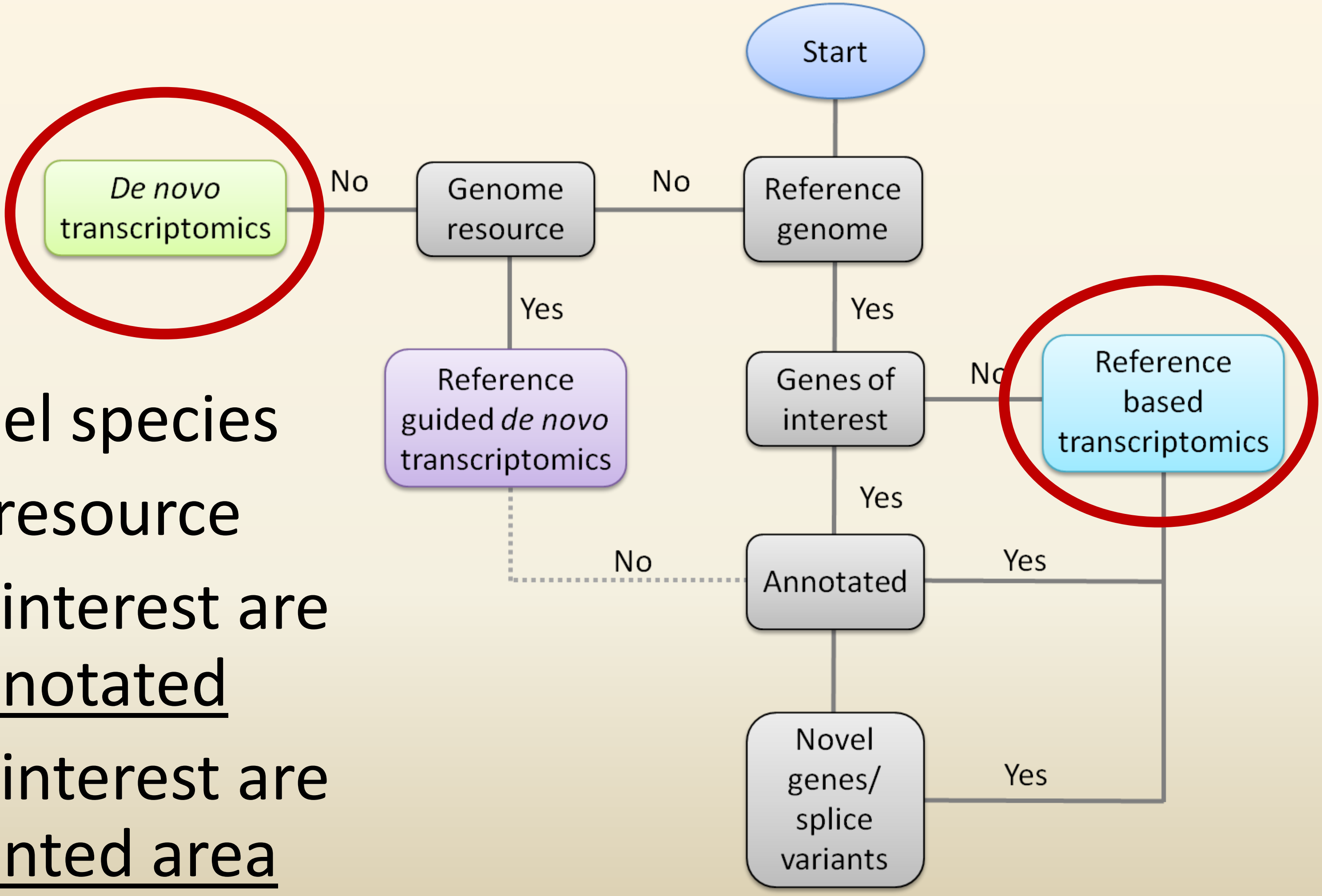
What to consider – INFBIO case

- What do I want? – transcriptome
- What will I use it for? – differential expression
- Which resources are available for your species (very closely related species)? – genome
- What kind of data do I have? – Illumina PE
- 2n or ploidy? – 2n

Choosing our strategy


Start
Reference genome
No
Genome resource
No
De novo transcriptomics
Yes
Reference guided *de novo* transcriptomics
Yes
Genes of interest
No
Reference based transcriptomics
Yes
No
Annotated
Yes
Novel genes/ splice variants
Yes

Choosing our strategy


Start
Reference genome
No
Genome resource
No
De novo transcriptomics
Yes
Reference guided *de novo* transcriptomics
Yes
Genes of interest
No
Reference based transcriptomics
Yes
Annotated
No
Yes
Novel genes/ splice variants
Yes

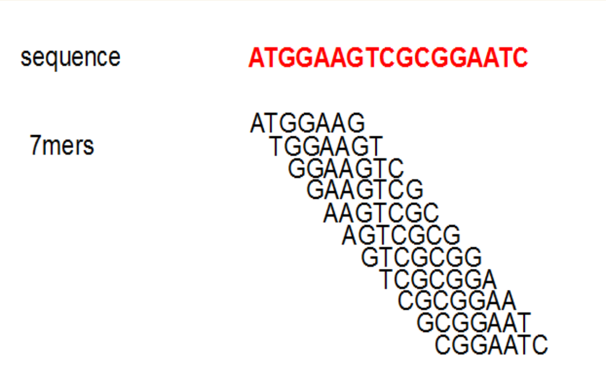
- Non-model species
- Genome resource
- Genes of interest are <u>poorly annotated</u>
- Genes of interest are <u>in fragmented area</u> of genome

Trinity assembler

- Trinity is the best single parameter *de novo* RNA assembly pipeline available
- Good on splice variants, full length transcripts and resolution of lowly expressed transcripts
- Contains tools to help with visualizations

Trinity pipeline - Inchworm

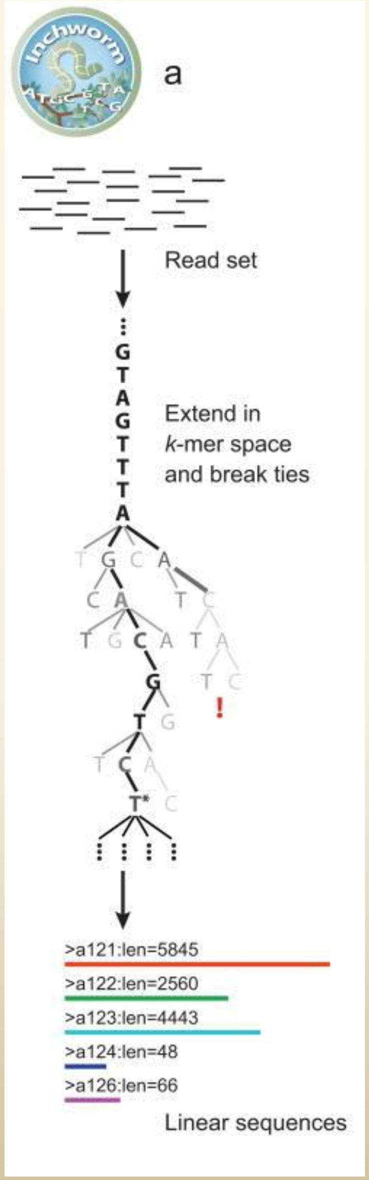
- It employs a greedy kmer based approach to reconstruct the best representative for a transcriptionally active region (often full-length dominant isoform).

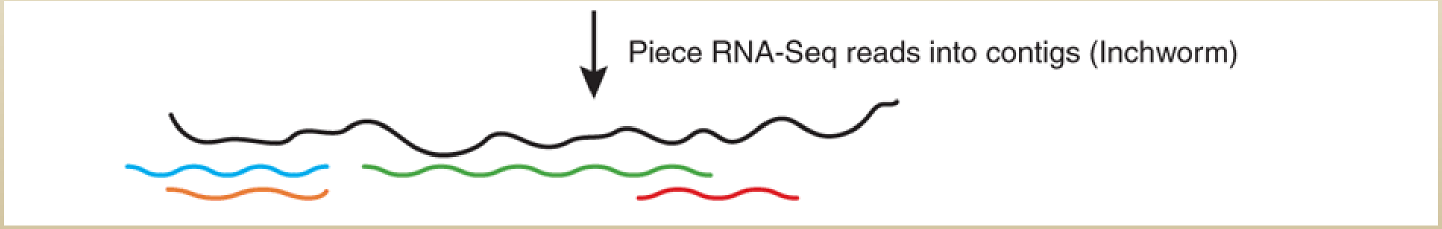


sequence ATGGAAGTCGCGGAATC

7mers

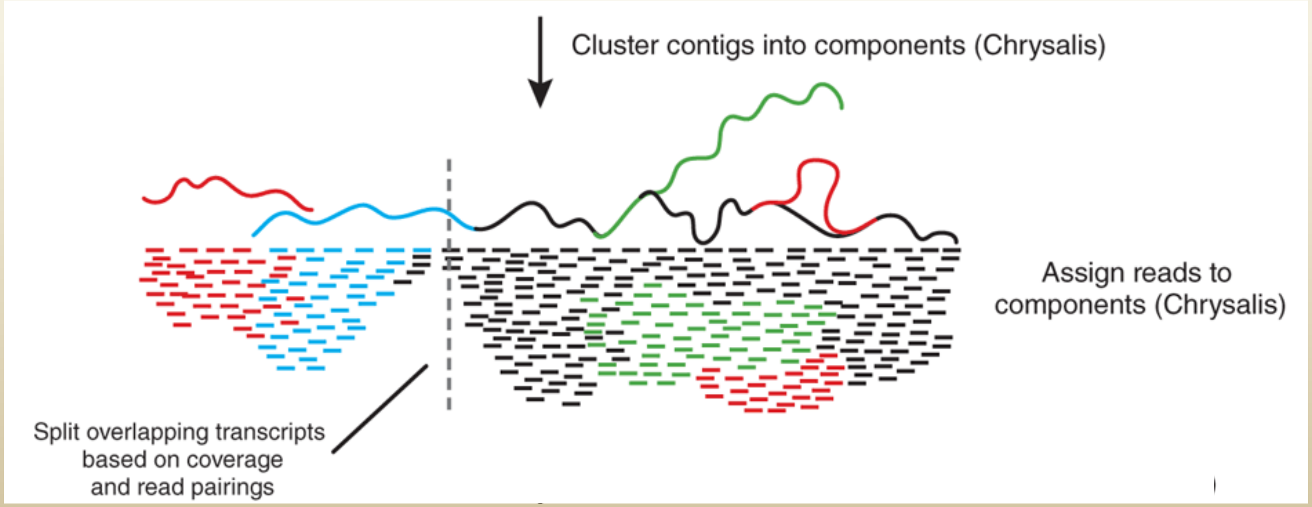
```
ATGGAAG
 TGGAAGT
  GGAAGTC
   GAAGTCG
    AAGTCGC
     AGTCGCG
      GTCGCGG
       TCGCGGA
        CGCGGAA
         GCGGAAT
          CGGAATC
```

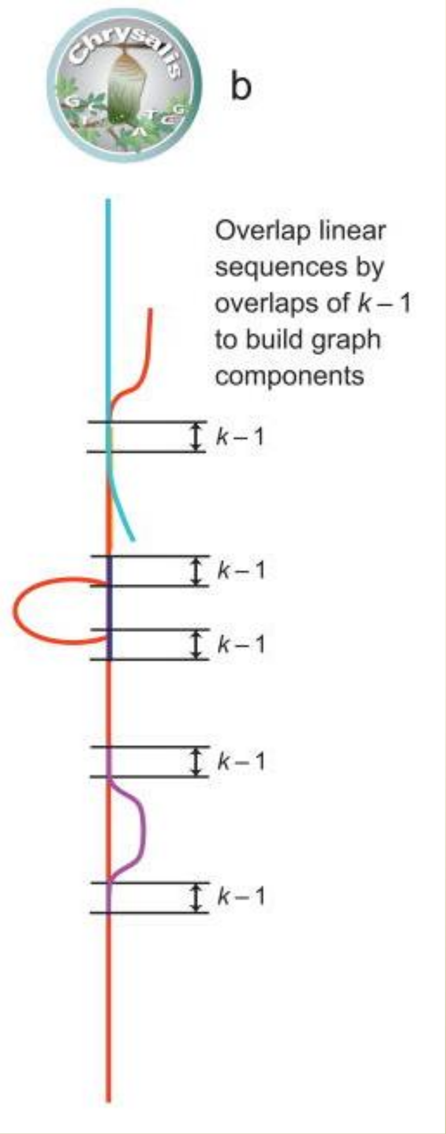




Trinity pipeline

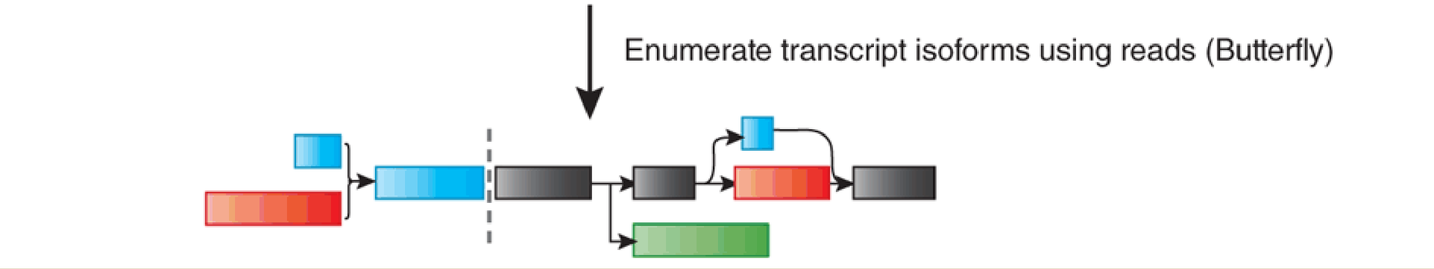
- Chrysalis clusters Inchworm related contigs into components (alternatively spliced variants)
- Then a De Bruijn graph is made for each component

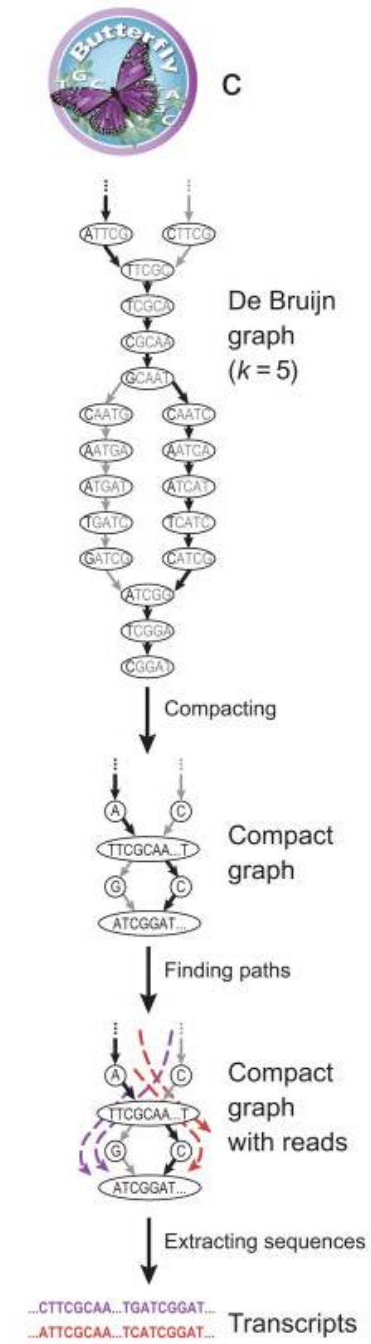

Cluster contigs into components (Chrysalis)
Assign reads to components (Chrysalis)
Split overlapping transcripts based on coverage and read pairings


Chrysalis
b
Overlap linear sequences by overlaps of k – 1 to build graph components
k – 1
k – 1
k – 1
k – 1
k – 1

Trinity pipeline

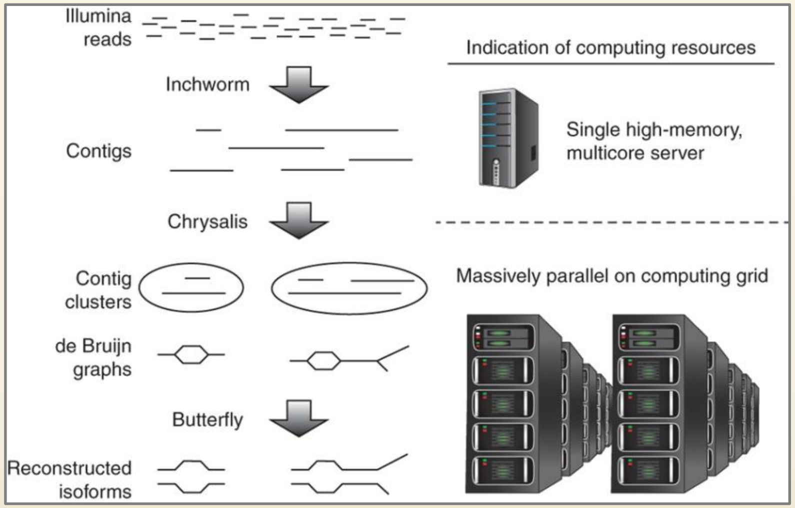
- Butterfly analyzes the paths taken by reads and read pairings in the graphs and reports all plausible transcripts including splice variants and transcripts derived from paralogs (duplicated genes)





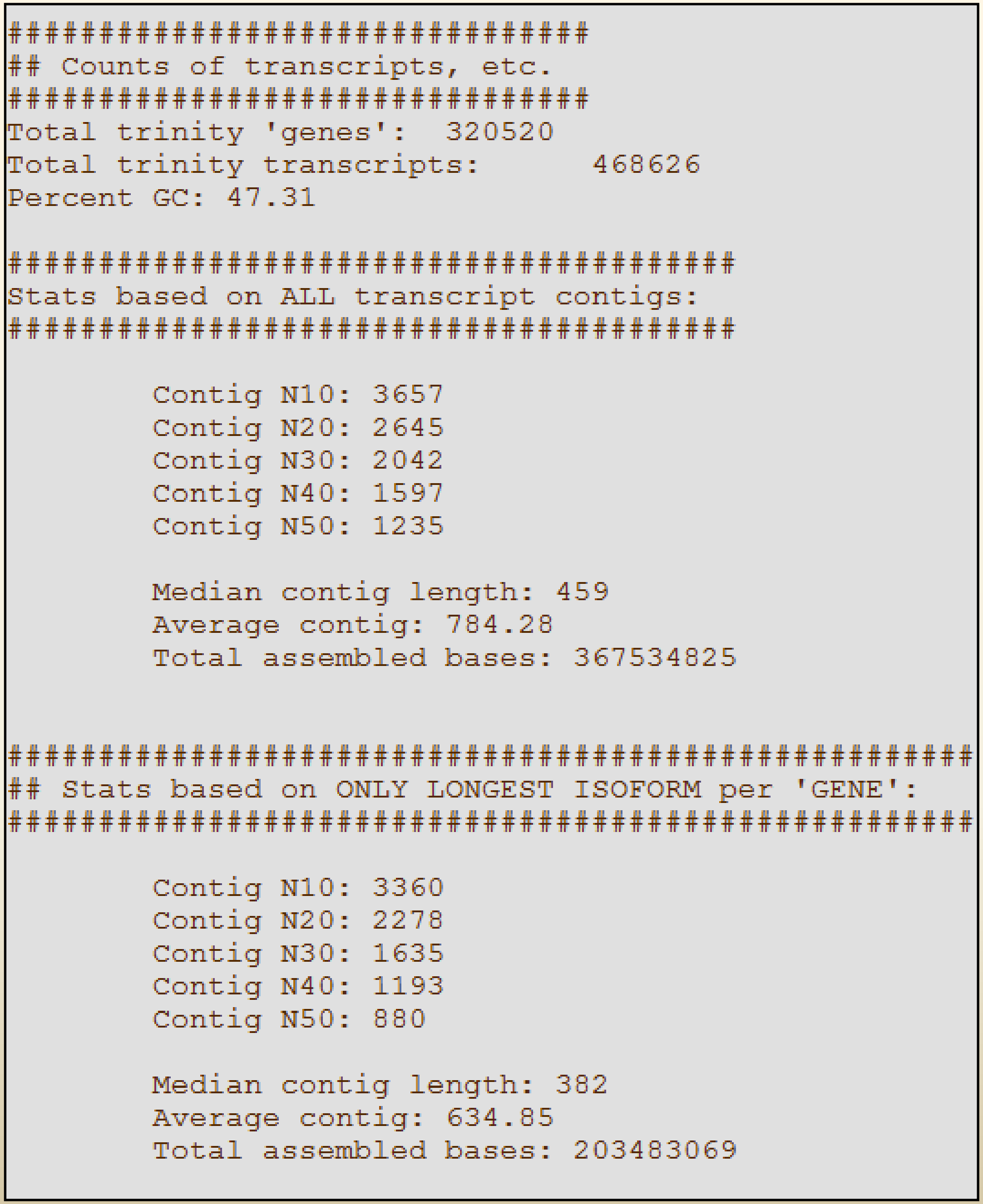
Trinity computation requirements

- Assembly algorithms require large amounts of memory
- 2/3rds of Trinity is parallelized to save computation time
- Estimate at least 1 week of trial/error/final computation
- Remember to calculate memory/time requirements before starting!
  - 1Gb RAM / million reads
  - ½ - 1 hour / million reads


Illumina reads
Indication of computing resources
Inchworm
Contigs
Single high-memory, multicore server
Chrysalis
Contig clusters
Massively parallel on computing grid
de Bruijn graphs
Butterfly
Reconstructed isoforms

What to expect

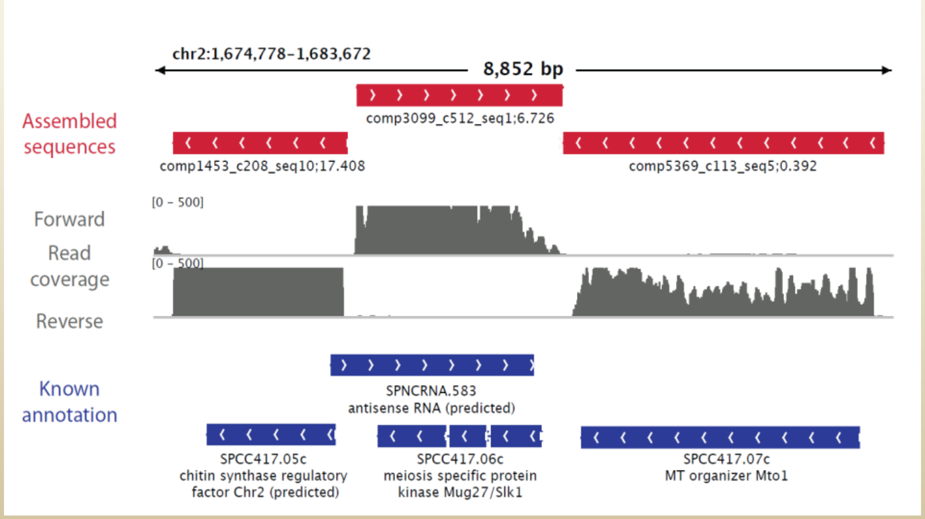
- Significantly more transcripts than predicted in the same or closely related species!
- Low coverage over splice junctions, sequencing errors and heterozygosity restricts full-length transcript reconstruction



```
################################
## Counts of transcripts, etc.
################################
Total trinity 'genes':  320520
Total trinity transcripts:      468626
Percent GC: 47.31

########################################
Stats based on ALL transcript contigs:
########################################

        Contig N10: 3657
        Contig N20: 2645
        Contig N30: 2042
        Contig N40: 1597
        Contig N50: 1235

        Median contig length: 459
        Average contig: 784.28
        Total assembled bases: 367534825


#####################################################
## Stats based on ONLY LONGEST ISOFORM per 'GENE':
#####################################################

        Contig N10: 3360
        Contig N20: 2278
        Contig N30: 1635
        Contig N40: 1193
        Contig N50: 880

        Median contig length: 382
        Average contig: 634.85
        Total assembled bases: 203483069
```

How to make it comparable

- Trinity comes with:
  - Full length estimation (BLAST based)
  - Abundance estimation (simple expression analysis)
- Consider mapping transcripts towards a reference genome if available


chr2:1,674,778-1,683,672
8,852 bp
Assembled sequences
comp3099_c512_seq1;6.726
comp1453_c208_seq10;17.408
comp5369_c113_seq5;0.392
Forward
Read coverage
Reverse
[0 - 500]
[0 - 500]
Known annotation
SPNCRNA.583
antisense RNA (predicted)
SPCC417.05c
chitin synthase regulatory factor Chr2 (predicted)
SPCC417.06c
meiosis specific protein kinase Mug27/Slk1
SPCC417.07c
MT organizer Mto1