RNA-seq Data Handling and Analysis

Kevin Childs

Statistical genetics/genomics journal club

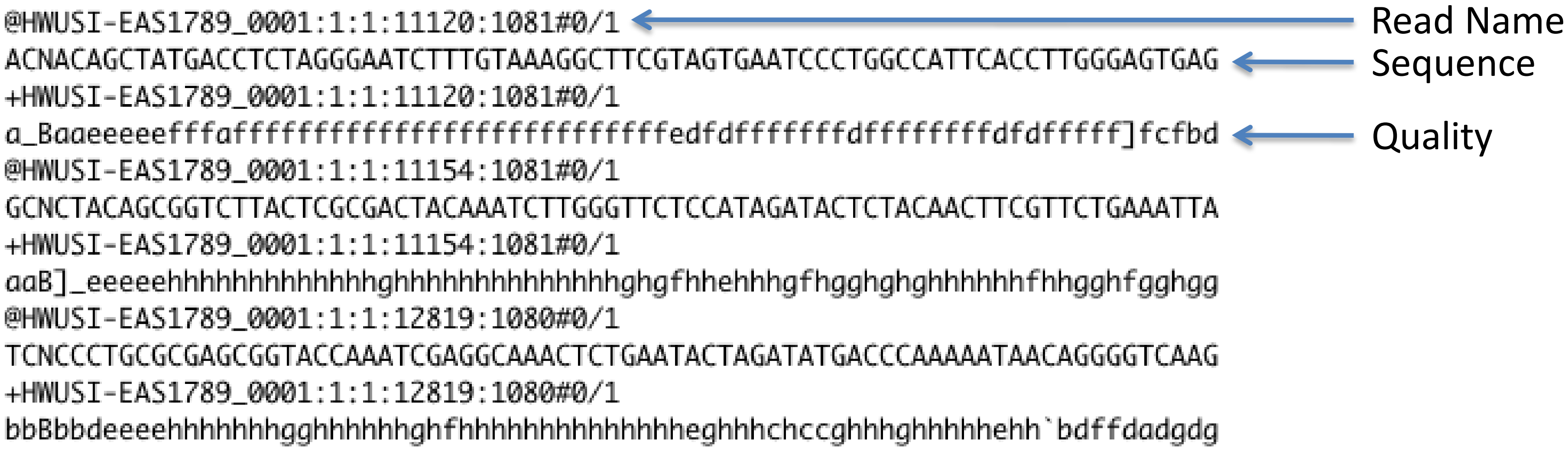
Overview

- Fasta/Fastq file formats
- NCBI SRA
- Data preparation
- Bowtie/Tophat/Cufflinks
- Velvet/Oases
- Trinity

Fasta File Format

```
##FASTA
>gi|1800214|gb|U56729.1|SBU56729 Sorghum bicolor phytochrome A
CGCATCCTTCCGCGCCGGGCATGGGCACCGCGTCGGCGCGCGCCCCTACCCAGTCGTCGACTTGATGCTG
CTCACTCGCACTCGTCGCAGCGCCCCACGCCCCGCTATTTATGCGTACTTGCTTGCCGGGAGAGTCGCTG
GAGGTGGGCGTCCTCCTCCCGCTCCAGAGCTCGCTGCTTCGCTCCACCCACCCTTAAGCAGGAGTGATAT
CTGGTGGTTTTTCAAAAGAAGACAAAAATGTCTTCCTCGAGGCCTGCCCACTCTTCCAGTTCATCCAGTA
GGACTCGCCAGAGCTCCCAGGCAAGGATATTAGCACAAACAACCCTTGATGCTGAACTCAATGCAGAGTA
TGAAGAATCTGGTGATTCCTTTGATTACTCCAAGTTGGTTGAAGCACAGCGGAGCACTCCATCTGAGCAG
CAAGGGCGATCAGGAAAGGTCATAGCCTACTTGCAGCATATTCAAAGAGGAAAGCTAATCCAACCATTTG
GTTGCTTGTTGGCCCTTGACGAGAAGAGCTTCAGGGTCATTGCATTCAGTGAGAATGCACCTGAAATGCT
CACAACGGTCAGCCATGCTGTGCCAAACGTTGATGATCCCCCAAAGCTAGGAATTGGTACCAATGTGCGC
TCCCTTTTCACTGACCCTGGTGCTACAGCACTGCAGAAGGCACTAGGATTTGCTGATGTTTCTTTGCTGA
ATCCTATCCTAGTTCAATGCAAGACCTCAGGCAAGCCATTCTATGCCATTGTTCATAGGGCAACTGGTTG
TCTGGTGGTTGATTTTGAGCCTGTGAAGCCTACAGAATTTCCTGCCACTGCTGCTGGGGCTTTGCAGTCT
```

Fastq File Format



```
@HWUSI-EAS1789_0001:1:1:11120:1081#0/1
ACNACAGCTATGACCTCTAGGGAATCTTTGTAAAGGCTTCGTAGTGAATCCCTGGCCATTCACCTTGGGAGTGAG
+HWUSI-EAS1789_0001:1:1:11120:1081#0/1
a_Baaeeeeefffafffffffffffffffffffffffffffedfdfffffffdffffffffdfdfffff]fcfbd
@HWUSI-EAS1789_0001:1:1:11154:1081#0/1
GCNCTACAGCGGTCTTACTCGCGACTACAAATCTTGGGTTCTCCATAGATACTCTACAACTTCGTTCTGAAATTA
+HWUSI-EAS1789_0001:1:1:11154:1081#0/1
aaB]_eeeeehhhhhhhhhhhhghhhhhhhhhhhhhhhghgfhhehhhgfhgghghghhhhhhfhhgghfgghgg
@HWUSI-EAS1789_0001:1:1:12819:1080#0/1
TCNCCCTGCGCGAGCGGTACCAAATCGAGGCAAACTCTGAATACTAGATATGACCCAAAAATAACAGGGGTCAAG
+HWUSI-EAS1789_0001:1:1:12819:1080#0/1
bbBbbdeeeehhhhhhhgghhhhhhghfhhhhhhhhhhhhhheghhhchccghhhghhhhhehh`bdffdadgdg
```

Read Name

Sequence

Quality

Quality scores are in ASCII characters representing coded Phred scores.

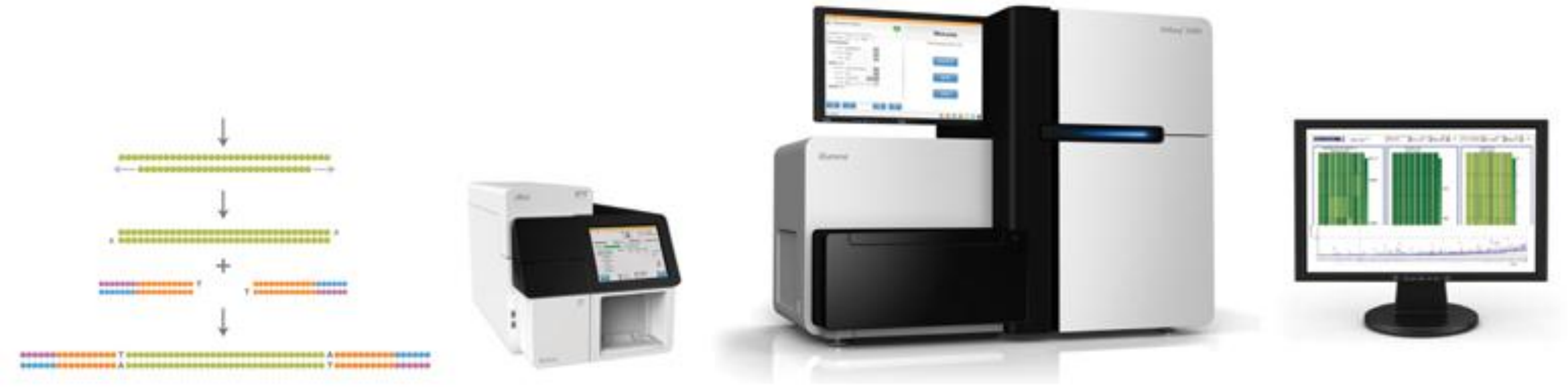
ASCII codes start at ASCII 33 or ASCII 64. All SRA codes converted to ASCII 33

These scores provide a likelihood that the base was called incorrectly.
10 – 1 in 10 chance the base call is incorrect
20 – 1 in 100 chance the base call is incorrect
30 – 1 in 1000 chance the base call is incorrect

High Throughput Sequencing Platforms

- Illumina HiSeq 1000 and HiSeq 2000
- Illumina Genome Analyzer IIx *
- Life Sciences/Roche 454 pyrosequencing
- ABI Solid Sequencing System *
- Pacific Biosciences *
- Ion Torrent
- Cambridge Nannopore (late 2012?)

High Throughput Sequencing



Library Preparation
~2 h [15 min hands-on (Nextera)]
< 6 h [< 3 h hands-on (TruSeq)]

Cluster Generation
~5 h (<10 min hands-on)

Sequencing by Synthesis
~1.5 to 8.5 days

CASAVA
2 days (30 min hands-on)

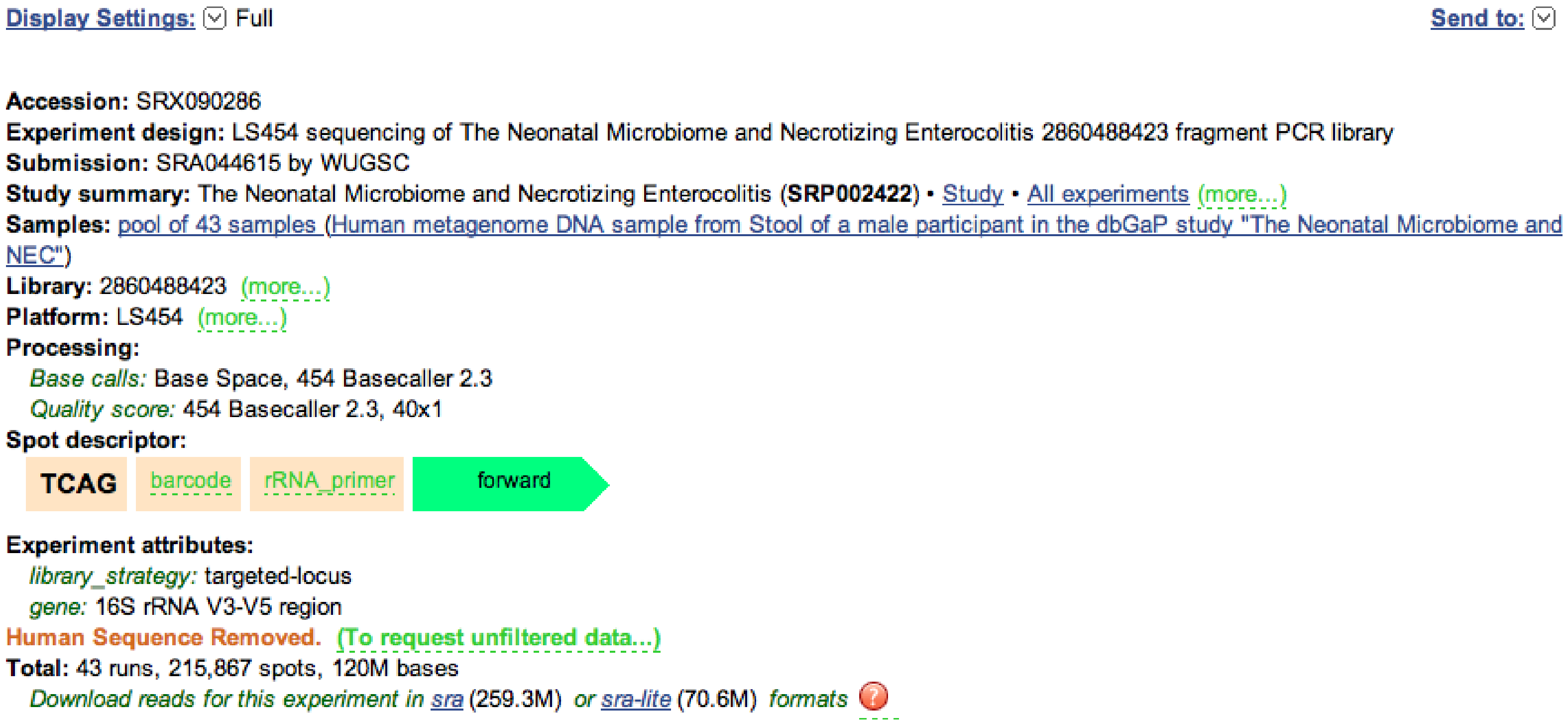
- HiSeq 2000
- Highly parallel sequencing by synthesis
- Single and paired-end reads between 50 bp and 100 bp
- 187 million single end or 374 million paired-end reads per lane
- High error rate in the 3’ end

NCBI SRA

- SRA toolkit
- fastq-dump

-
  ```
  /opt/sratoolkit/fastq-dump SRR373821.lite.sra
  ```

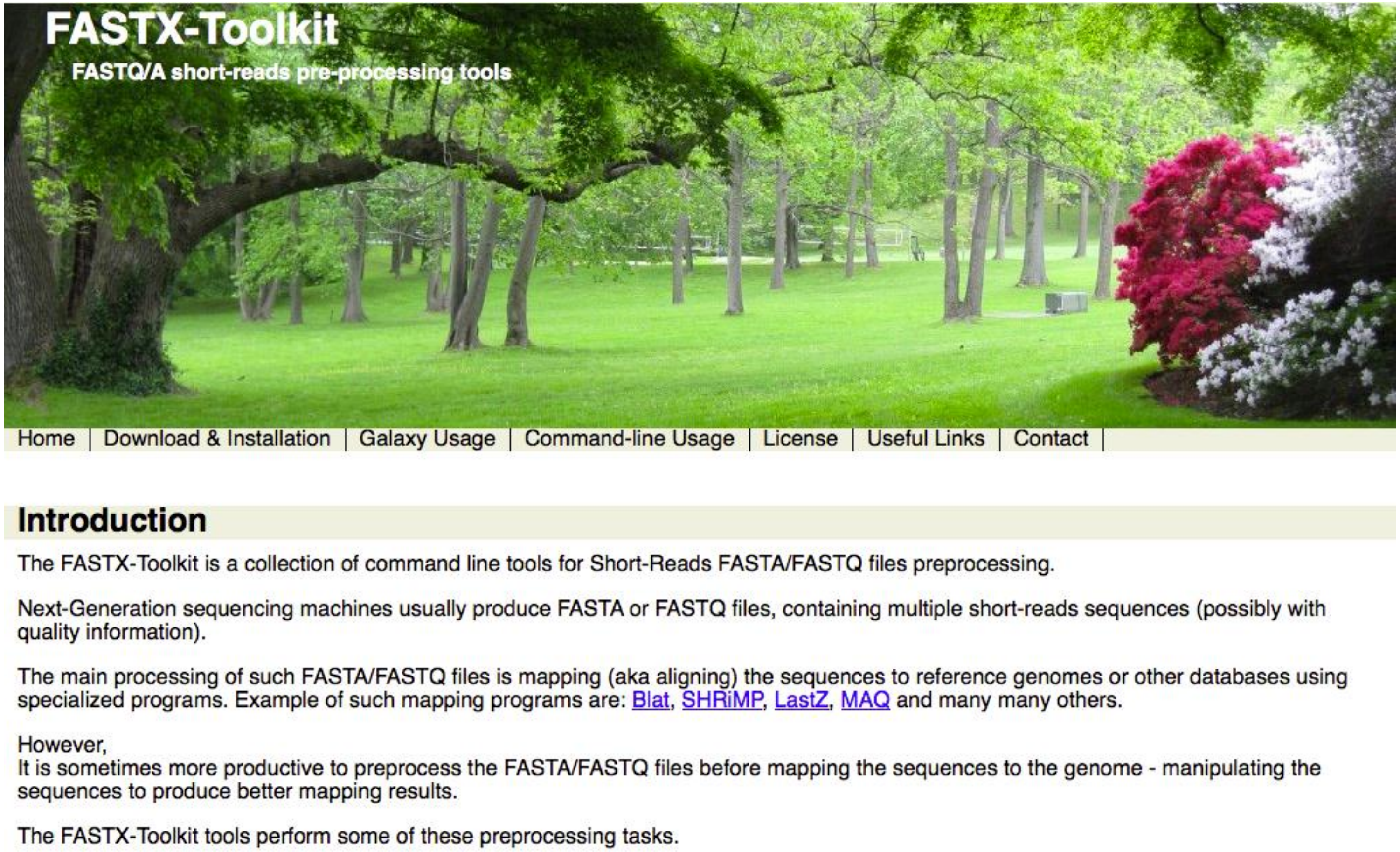
-
  ```
  /opt/sratoolkit/fastq-dump --split-files SRR329070.lite.sra
  ```



Accession: SRX090286
Experiment design: LS454 sequencing of The Neonatal Microbiome and Necrotizing Enterocolitis 2860488423 fragment PCR library
Submission: SRA044615 by WUGSC
Study summary: The Neonatal Microbiome and Necrotizing Enterocolitis (**SRP002422**) • Study • All experiments (more...)
Samples: pool of 43 samples (Human metagenome DNA sample from Stool of a male participant in the dbGaP study "The Neonatal Microbiome and NEC")
Library: 2860488423 (more...)
Platform: LS454 (more...)
Processing:
Base calls: Base Space, 454 Basecaller 2.3
Quality score: 454 Basecaller 2.3, 40x1
Spot descriptor:
TCAG barcode rRNA_primer forward

Experiment attributes:
library_strategy: targeted-locus
gene: 16S rRNA V3-V5 region
Human Sequence Removed. (To request unfiltered data...)
Total: 43 runs, 215,867 spots, 120M bases
Download reads for this experiment in sra (259.3M) *or* sra-lite (70.6M) *formats*

Read Quality with the FASTX-Toolkit



FASTX-Toolkit
FASTQ/A short-reads pre-processing tools
Home | Download & Installation | Galaxy Usage | Command-line Usage | License | Useful Links | Contact |

Introduction

The FASTX-Toolkit is a collection of command line tools for Short-Reads FASTA/FASTQ files preprocessing.

Next-Generation sequencing machines usually produce FASTA or FASTQ files, containing multiple short-reads sequences (possibly with quality information).

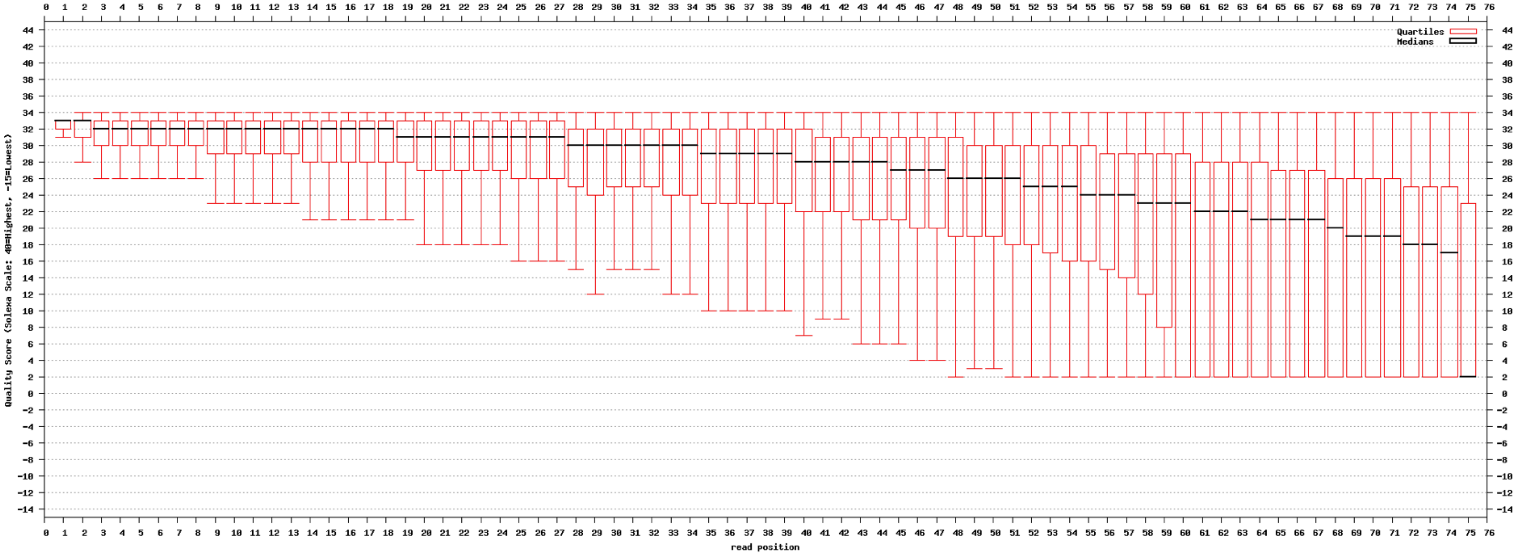
The main processing of such FASTA/FASTQ files is mapping (aka aligning) the sequences to reference genomes or other databases using specialized programs. Example of such mapping programs are: Blat, SHRiMP, LastZ, MAQ and many many others.

However,
It is sometimes more productive to preprocess the FASTA/FASTQ files before mapping the sequences to the genome - manipulating the sequences to produce better mapping results.

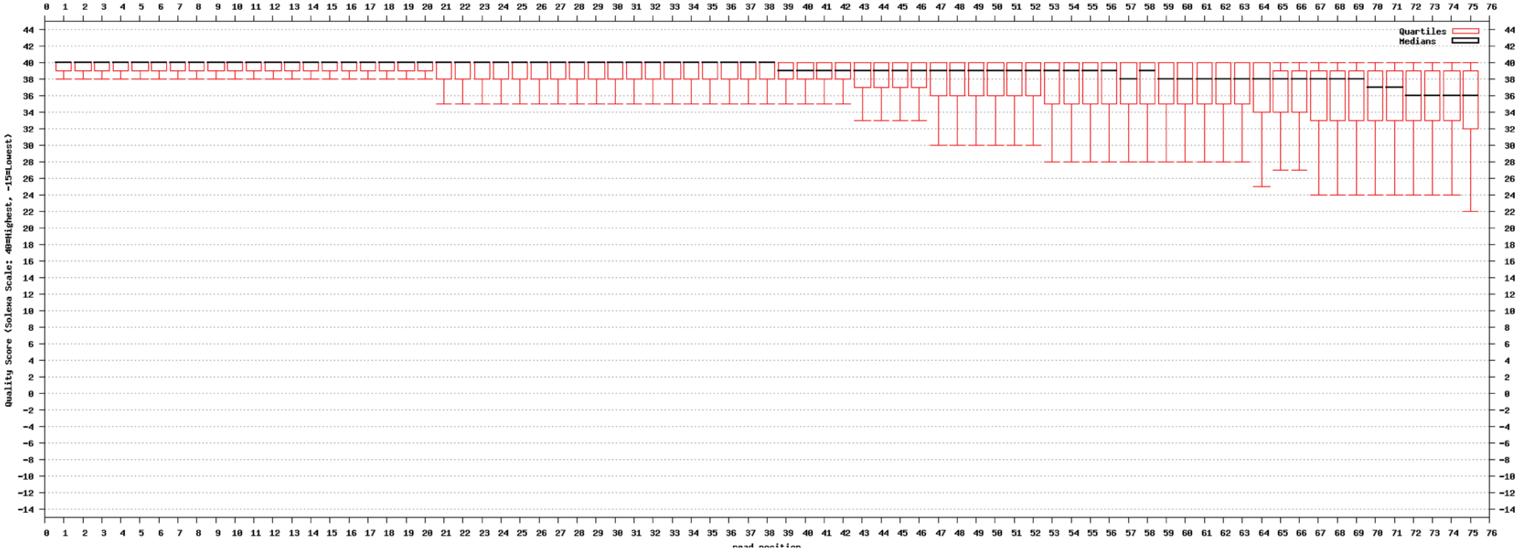
The FASTX-Toolkit tools perform some of these preprocessing tasks.

http://hannonlab.cshl.edu/fastx_toolkit/

Read Quality with the FASTX-Toolkit



Bad Sequence



Good Sequence

FASTX Toolkit

```
fastx_quality_stats -Q 33 –i initial_fastq_file.fastq –o stats.txt
```

```
fastx_quality_boxplot_graph.sh -Q 33 –i stats.txt –t Title –o
quality.png
```

```
fastx_clipper -Q 33  -v  -a
AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTACACGACGCTCTTCCGATCT  -i
initial_fastq_file.fastq  -o fastq_file_clipped.fastq
```

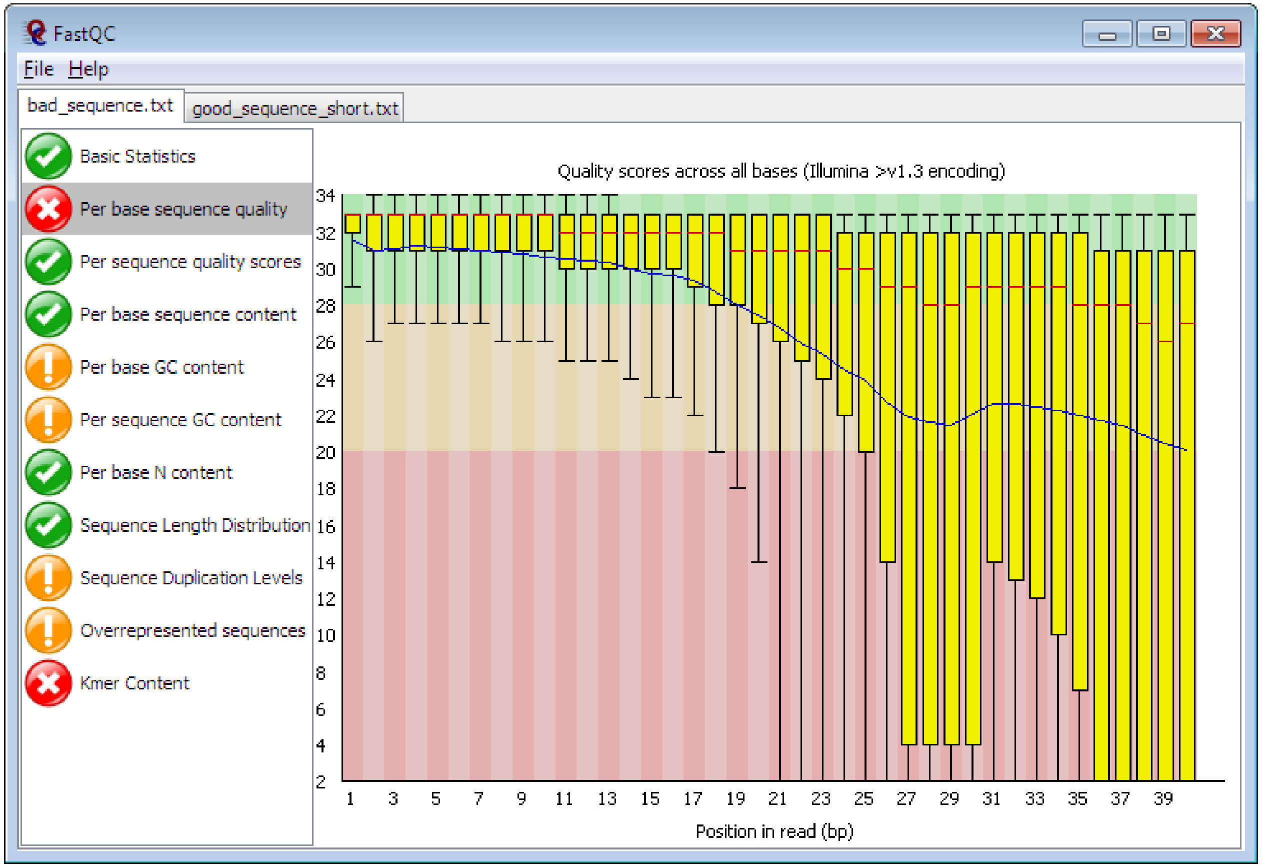
```
fastx_artifacts_filter  -Q 33  -v  -i fastq_file_clipped.fastq  -o
fastq_file_artifact_filtered.fastq
```

```
fastq_quality_trimmer  -Q 33 -v  -t 20  -l 30  -i
fastq_file_artifact_filtered.fastq -o fastq_file_cleaned.fastq
```

-Q is an undocumented parameter to indicate that quality values use ASCII 33 encoding.

FastQC

A quality control tool for high throughput sequence data.


FastQC
File Help
bad_sequence.txt
good_sequence_short.txt
✓ Basic Statistics
✗ Per base sequence quality
✓ Per sequence quality scores
✓ Per base sequence content
! Per base GC content
! Per sequence GC content
✓ Per base N content
✓ Sequence Length Distribution
! Sequence Duplication Levels
! Overrepresented sequences
✗ Kmer Content
Quality scores across all bases (Illumina >v1.3 encoding)
34
32
30
28
26
24
22
20
18
16
14
12
10
8
6
4
2
1
3
5
7
9
11
13
15
17
19
21
23
25
27
29
31
33
35
37
39
Position in read (bp)

http://www.bioinformatics.bbsrc.ac.uk/projects/fastqc/

Samtools

- Package of programs for manipulating sam and bam files
- sam – sequence alignment map
- bam – binary alignment map
  - compressed form of sam file
- http://samtools.sourceforge.net/samtools-c.shtml

Tuxedo Suite

- Bowtie – fast and quality aware short read aligner for aligning DNA and RNA sequence reads
- TopHat – fast, splice junction mapper for RNA-Seq reads built on the Bowtie aligner
- Cufflinks – assembles transcripts, estimates their abundances, and test for differential expression and regulation using the alignments from Bowtie and TopHat

Bowtie

Bowtie

An ultrafast memory-efficient short read aligner

- Aligns short reads to large genomes
- Forms the basis for TopHat, Cufflinks, Crossbow, and Myrna
- Unless you are working with genomic DNA derived short reads, you will not directly use Bowtie
- With the exception of using bowtie-build to create an genomic sequence index file

TopHat

TopHat

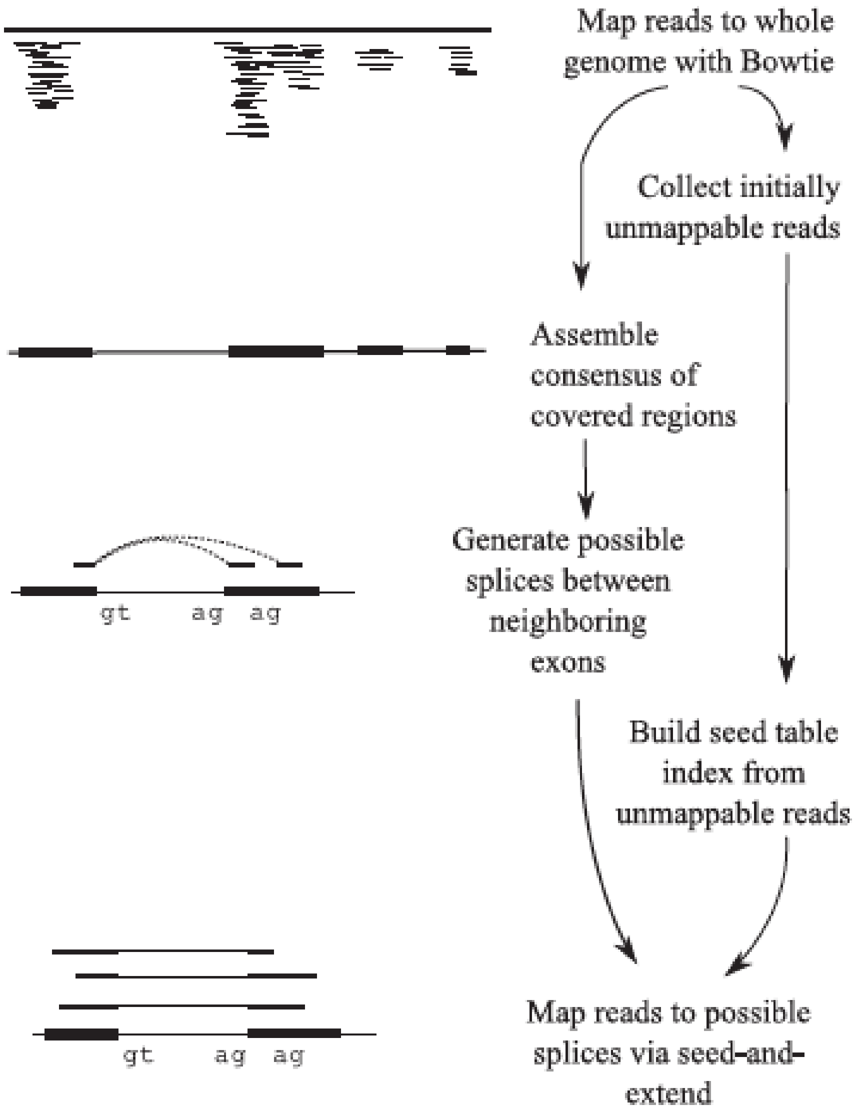
A spliced read mapper for RNA-Seq

- Built on Bowtie and uses the same genome index
- Used for alignment of RNA-Seq reads to a genome
- Optimized for paired-end, Illumina sequence reads >70bp

TopHat

TopHat

A spliced read mapper for RNA-Seq



Map reads to whole genome with Bowtie

Collect initially unmappable reads

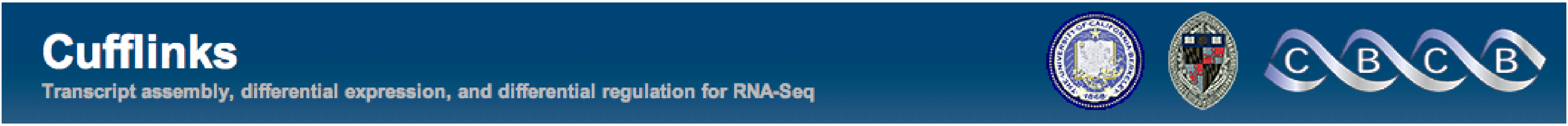
Assemble consensus of covered regions

Generate possible splices between neighboring exons

Build seed table index from unmappable reads

Map reads to possible splices via seed-and-extend

Cufflinks

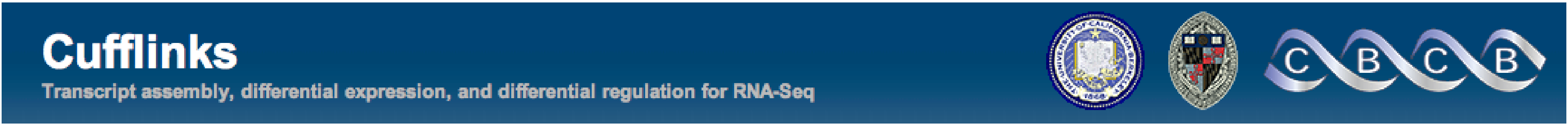


Cufflinks

Transcript assembly, differential expression, and differential regulation for RNA-Seq

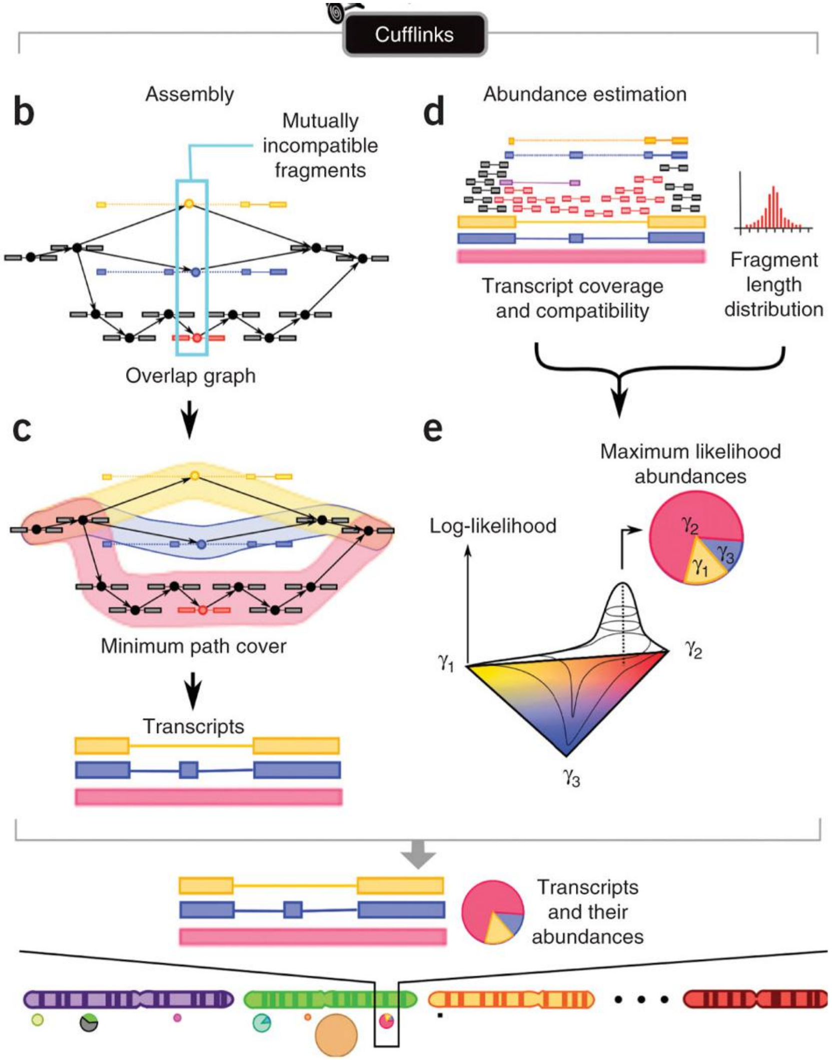
- Quantification of gene expression using RNA-seq reads
- Tests for differential expression
- Uses output from bowtie/tophat
- Assembles read alignments into transcripts
- Uses cufflinks-predicted transcripts or user-supplied gene models for quantification
- Estimates transcript abundance balanced across transcript isoforms

Cufflinks



Cufflinks

Transcript assembly, differential expression, and differential regulation for RNA-Seq



Cufflinks
b
Assembly
Mutually incompatible fragments
Overlap graph
c
Minimum path cover
Transcripts
d
Abundance estimation
Transcript coverage and compatibility
Fragment length distribution
e
Maximum likelihood abundances
Log-likelihood
γ_1
γ_2
γ_3
Transcripts and their abundances

Bowtie/Tophat/Cufflinks

```
bowtie-build  pseudomolecule.fa  pseudomolecule.index

tophat -p 6 --solexa1.3-quals -i 5  -I 1000  -r 100  --no-novel-
juncs  --GTF pseudomolecule.gtf  -o /output/directory
pseudomolecule.index  purified_reads.fastq

samtools sort tophat_output_pairs.bam  tophat_output_pairs_sorted

samtools view  -o tophat_output_pairs_sorted.sam
tophat_output_pairs_sorted.bam

cufflinks  -q  -o /output/directory/  -p 4  -G
pseudomolecule_corrected.gtf  tophat_output_pairs_sorted.sam
```

Velvet/Oases

- Genome/transcriptome assembly package
- Velveth/velvetg work well for genomes but produce fragmented transcriptomes assemblies. Its modules explicitly assume linearity and uniform coverage distribution. Velvet was designed to assemble genomic reads
- Oases was designed to assemble transcriptomes of new species.
- Oases takes the preliminary assembly produced by velvetg, and clusters the contigs into small groups, called loci.
- It then exploits the read sequence and pairing information, when available, to produce transcript isoforms.

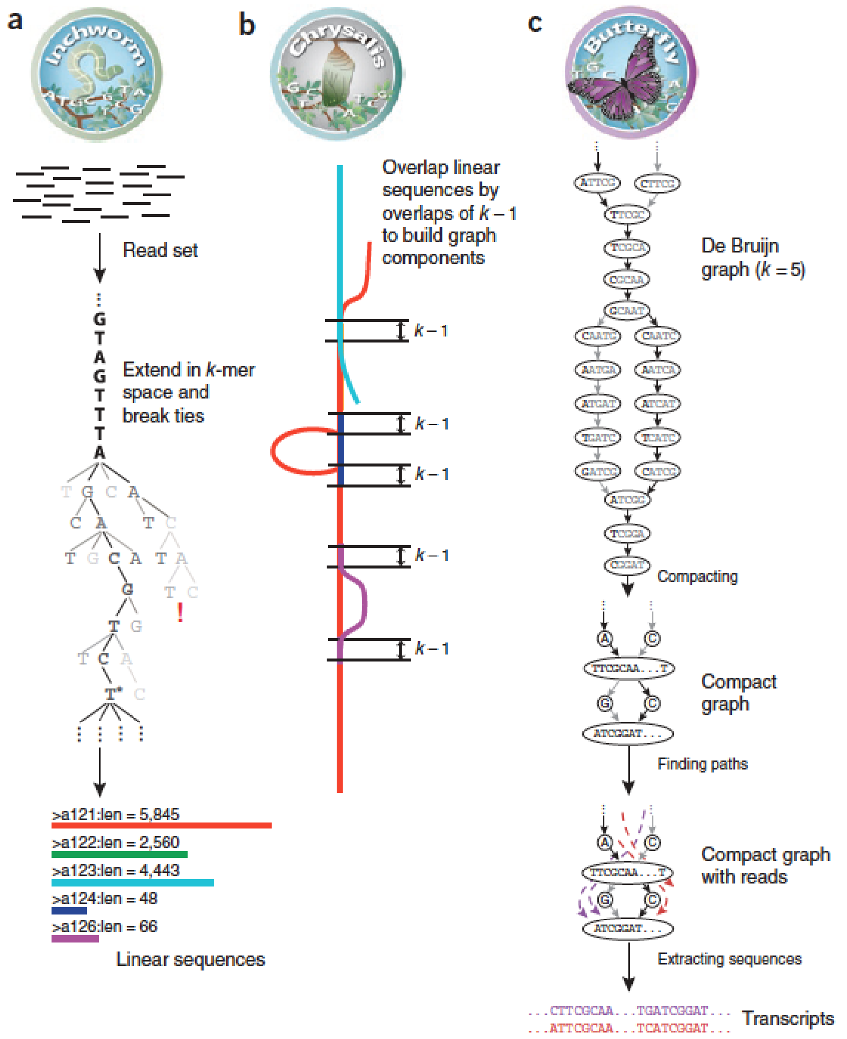
Velvet/Oases

```
velveth  /working/directory/  31 -fastq –shortPaired
switchgrass_purified_1.fastq  switchgrass_purified_2.fastq

velvetg  /working/directory/ -read_trkg yes -ins_length 190
-ins_length_sd 44

oases  /working/directory/  -ins_length 190  -ins_length_sd 44
-min_trans_lgth 250
```

Trinity

- Inchworm, Chrysalis, Butterfly
- Transcript assembly package

Trinity



```
Trinity.pl  --seqType fa  –left ABA_RC_1.fa  –right ABA_RC_2.fa  –single se_ABA_RC.fa  --paired_fragment_length 400  --run_butterfly  --output /output/directory  --CPU 2
```

TRINITY

pe_ABA_RC, se_ABA_RC

5,985,516 sequences

num_of_transcripts= 25069
max_len_trans= 2707
min_len_trans= 300
N50= 6492168.5
Avrg size of contigs= 517.943954685069
N50_contig= 544

Length (bp)	distribution
< 150	0
150 to 250	0
251 to 500	14467
501 to 750	7437
751 to 1000	2323
1001 to 1250	628
1251 to 1500	162
1501 to 2000	49
2001 to 3000	3
3001 to 4000	0
4001 to 5000	0
> 5000	0

VELVET

pe_ABA_RC, pe_ABA_RB, se_ABA_RC, se_ABA_RC.

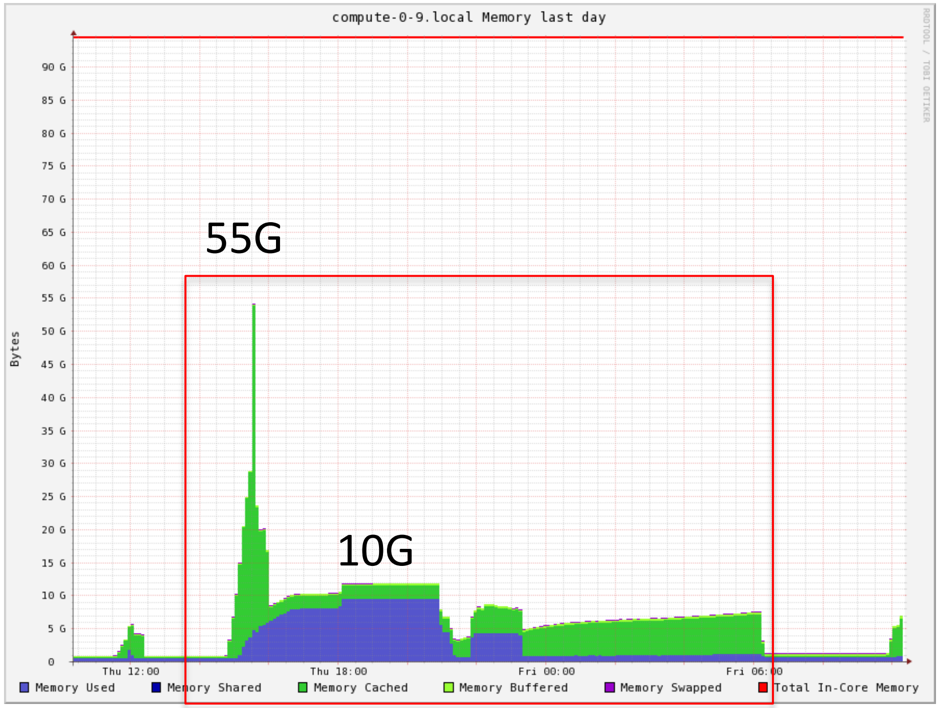
Assembly: used 41385139/46322456 reads
Assembled reads= 89.3414179075479%

num_of_transcripts= 86458
max_len_trans= 3597
min_len_trans= 251
N50= 39127035.5
Avrg size of contigs= 905.110816812788
N50_contig= 1136

Length (bp)	distribution
< 150	0
150 to 250	0
251 to 500	21809
501 to 750	16486
751 to 1000	14946
1001 to 1250	12875
1251 to 1500	9613
1501 to 2000	8990
2001 to 3000	1684
3001 to 4000	55
4001 to 5000	0
> 5000	0

Memory requested and technical notes

5,985,516 sequences



A basic recommendation is to have 1G of RAM per 1M pairs of Illumina reads.

Our experience is that the entire process can require about 1 hour per million pairs of reads in the current implementation.

qsub_time	Thu Jun 23 15:07:59 2011
start_time	Thu Jun 23 15:08:08 2011
end_time	Fri Jun 24 05:54:35 2011

*Memory requirements have improved in more recent updates.