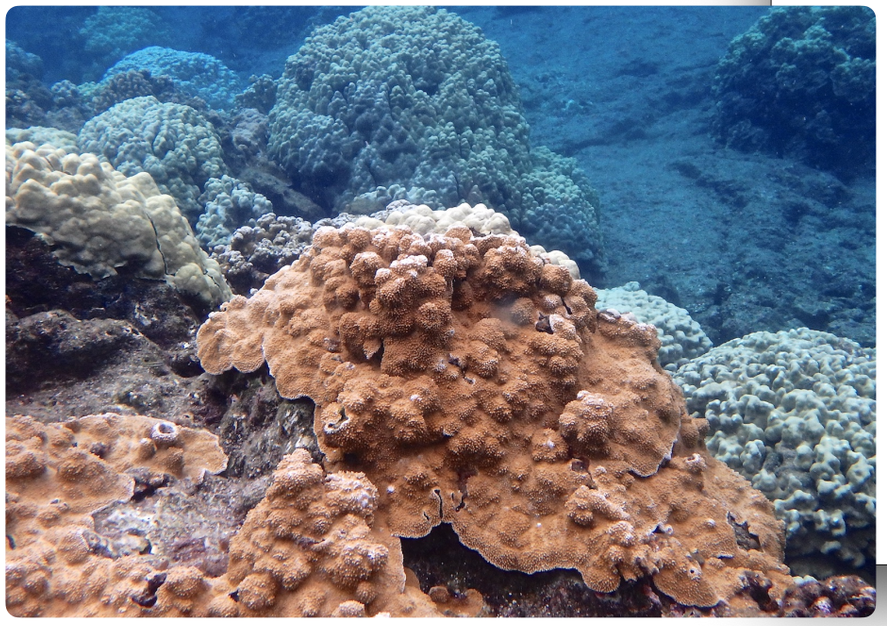
Genomics

An introduction to genome sequencing, data processing, and analysis

#DatAlumni course series @ ZMT, July 6/7, 2021 – part II

Martin Helmkampf

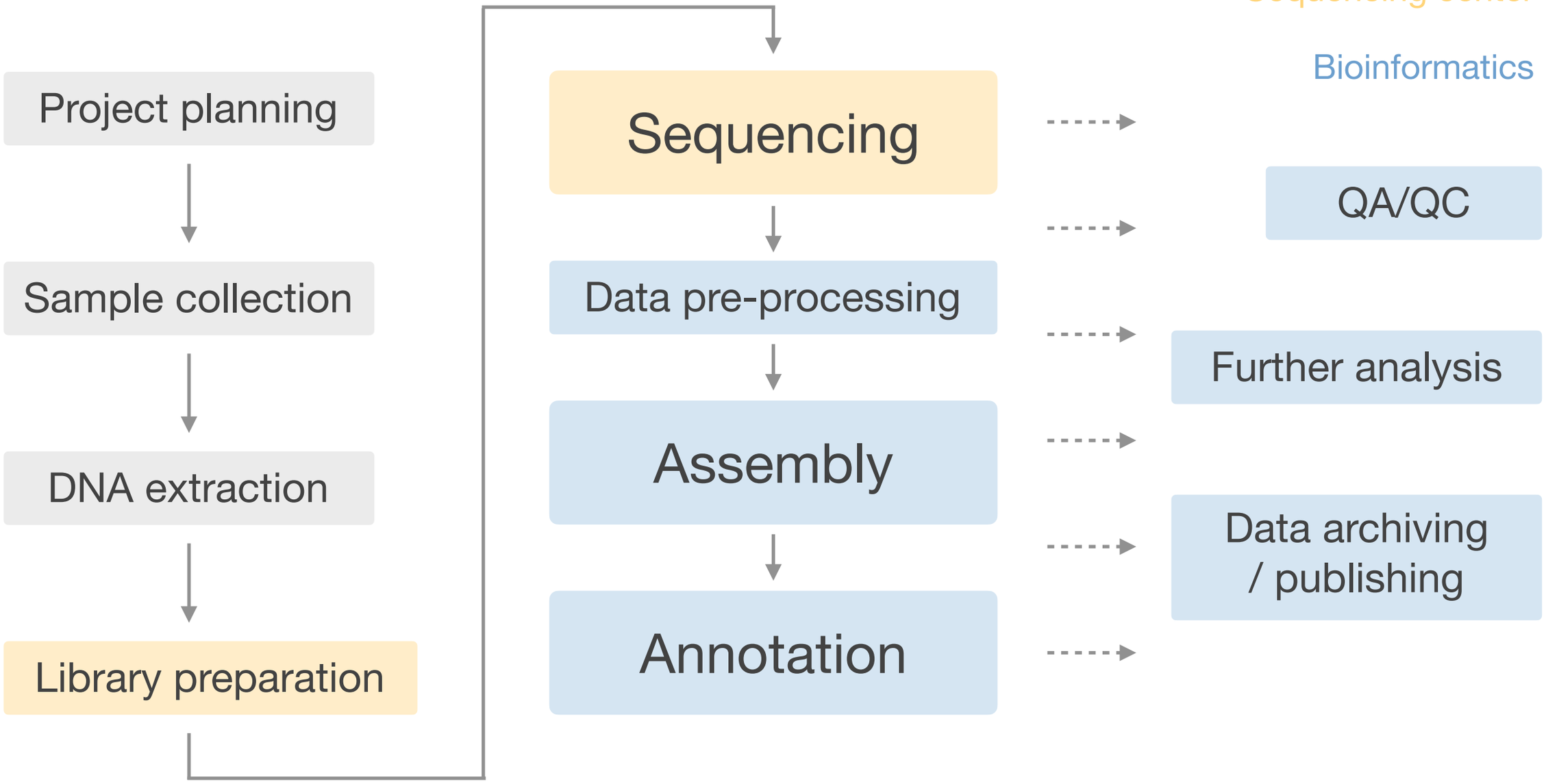


Workshop plan

Time	Block	Topics
Jul 06, 14:00–14:30	1. Background and introduction	– Sequencing techniques and applications – Workflow overview – Planning a genome project
Jul 06, 14:30–16:00	2. Data pre-processing and QA/QC	– Introduction to bioinformatics (exercise) – Raw sequencing data, trimming & filtering – Quality assurance / control
Jul 07, 14:00–15:00	3. Assembly and annotation	– Genome assembly (exercise) – Genome re-sequencing – Genome annotation
Jul 06, 15:00–16:00	4. Data management	– Documentation – Data organization – Publishing / sharing genome data

Objective: Convey basic knowledge of the steps required to sequence a genome, from project planning to data publication

De novo genome sequencing workflow



Recap: raw data

The FASTQ format

```
head -n 4 Hpue_raw300_F.fastq                    # display first 4 lines of file
```

```
@HWI-ST1293:199:HA9JHADXX:1:1101:1044:1603 1:N:0:CGATGT
NCCCTGTTAAAGGATCATCTCTGACCTATCATTGTGGTGTAATCACATTTAACACAATCACGATGTGCTTTACCTGCAGCATCTTTACAG
CAGGGCTGGGAGATATGACCAAAAACAGTTATGATAATATGTTTATTTCTATTGAAAATCA
+
#1=DDFFFHHHHHHJJJJJJJJJJJJJJJJJJJJJGHIJJJJJIIIJJJJHJJJJJJJJJJJJIJIJJJJJJJHHHHHFFFFFFEEEE
EEDDDDDDDDD@DDDEDEDDDBDDDDDDCDCEDDEEDDEEEDEECDDEDEDEEDDDDDDDC
```

1. @ followed by sequence id and optional info (e.g. instrument/run id, multiplex barcode)
2. DNA sequence
3. +, sometimes followed by sequence id
4. per-base quality score (same length as sequence)

Recap: raw data

The FASTQ format

Phred quality score:

$$Q = -10 \log_{10} P$$

Common benchmark:

% bases with Q ≥ 30

Quality score	*P* incorrect base call	Base call accuracy
10	1 in 10	90%
20	1 in 100	99%
30	1 in 1000	99.9%
40	1 in 10000	99.99%

FASTQ encoding (Illumina 1.8+):

```
ASCII Symbol:   !”#$%&’()*+,-./0123456789:;<=>?@ABCDEFGHIJ
                | |       |          |          |          |
Quality Score:  0.2......10........20........30.........41
```

Recap: bash basics

Open shell (bash)

```
cd <dir>                                    # change into workshop_zmt directory
git pull                                    # update workshop example data via GitHub

cd apps                                     # change into apps directory
ls -l                                       # display directory content in long format
```

```
-rw-r--r--@ 1 martin  staff   7902501 Jun 21 13:31 cutadapt-3.4.exe
-rw-r--r--@ 1 martin  staff  10249221 Jun 18 14:12 fastqc_v0.11.9.zip
drwxr-xr-x  3 martin  staff        96 Jul  7 10:44 test
```

d = directory, permissions	owner	group	file size (bytes)	modification time	file / dir name

Raw read quality check

Using the FastQC software

```
unzip fastqc_v0.11.9.zip                          # decompress fastqc software
FastQC/fastqc -h                                  # confirm software works, show help

cd ../project/1_rawdata                           # change into raw data folder
../../apps/FastQC/fastqc Hpue_raw300_F.fastq      # run FastQC on file
```

```
Started analysis of Hpue_raw300_F.fastq
Approx 5% complete for Hpue_raw300_F.fastq
Approx 10% complete for Hpue_raw300_F.fastq
…
Analysis complete for Hpue_raw300_F.fastq
```

Results can also be found in `workshop_zmt/project/2_processing/qc`

Raw read quality check

Using the FastQC software

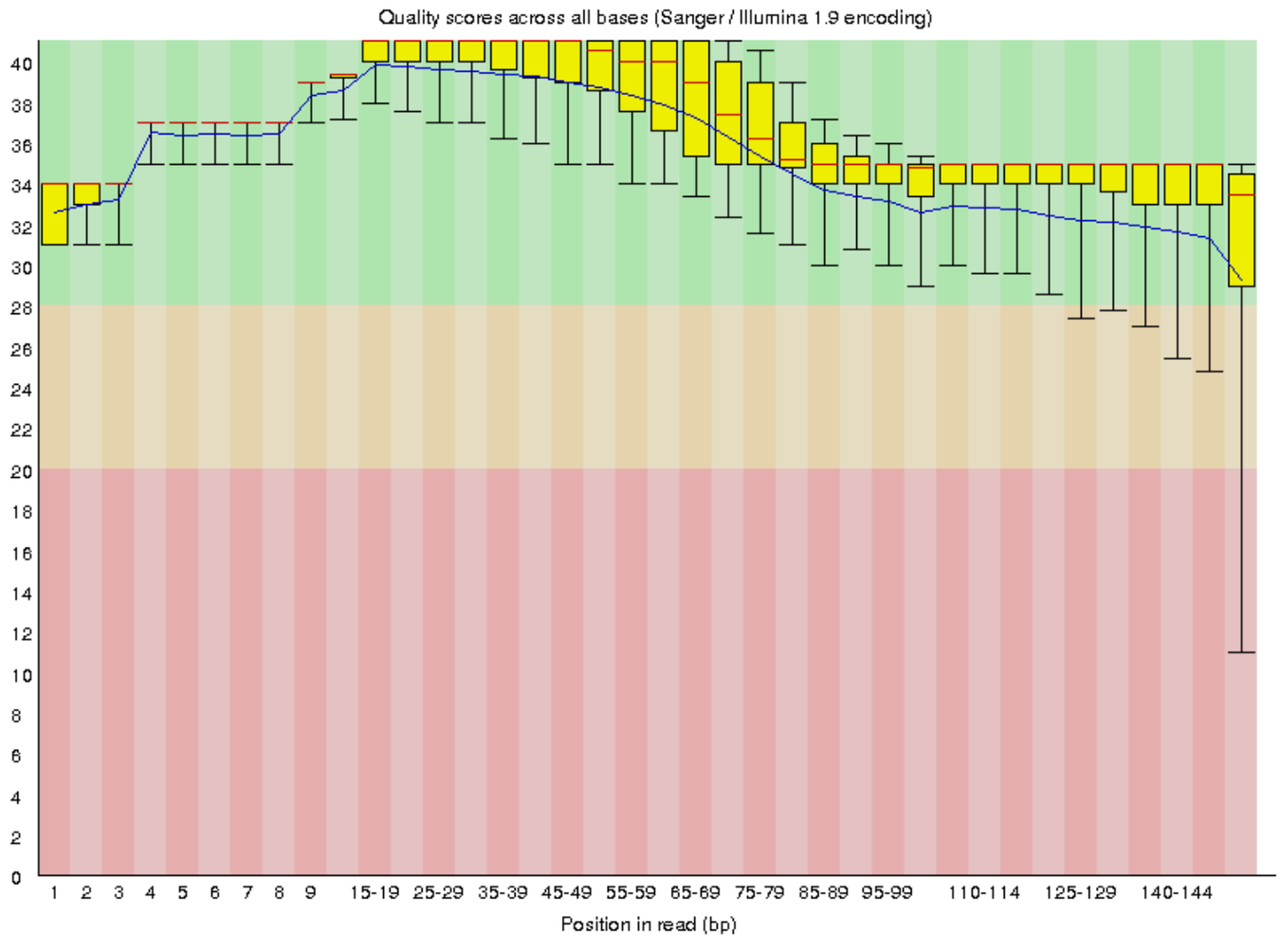
Basic Statistics

Measure	Value
Filename	Hpue_raw300_F.fastq
File type	Conventional base calls
Encoding	Sanger / Illumina 1.9
Total Sequences	75000
Sequences flagged as poor quality	0
Sequence length	151
%GC	40

Raw read quality check

Using the FastQC software

Per base sequence quality



Raw read quality check

Using the FastQC software

Overrepresented sequences

Sequence	Count	Percentage	Possible Source
GATCGGAAGAGCACACGTCTGAACTCCAGTCACCGATGTATCTCGTATGC	640	0.8533333333333334	TruSeq Adapter, Index 2 (100% over 50bp)

Example for bad Illumina data:

https://www.bioinformatics.babraham.ac.uk/projects/fastqc/bad_sequence_fastqc.html

Read trimming & filtering

Using cutadapt for adapter removal and quality trimming

```
../../apps/cutadapt-3.4.exe -help          # confirm app works, show help (from 1_rawdata)

../../apps/cutadapt-3.4.exe -a AGATCGGAAGAGCACACGTCTGAACTCCAGTCA -q 10 -o
Hpue_raw300_F_trim.fastq Hpue_raw300_F.fastq
# run FastQC (-q 10: quality-filter last 10 bases, -a: trim Illumina adapter)
```

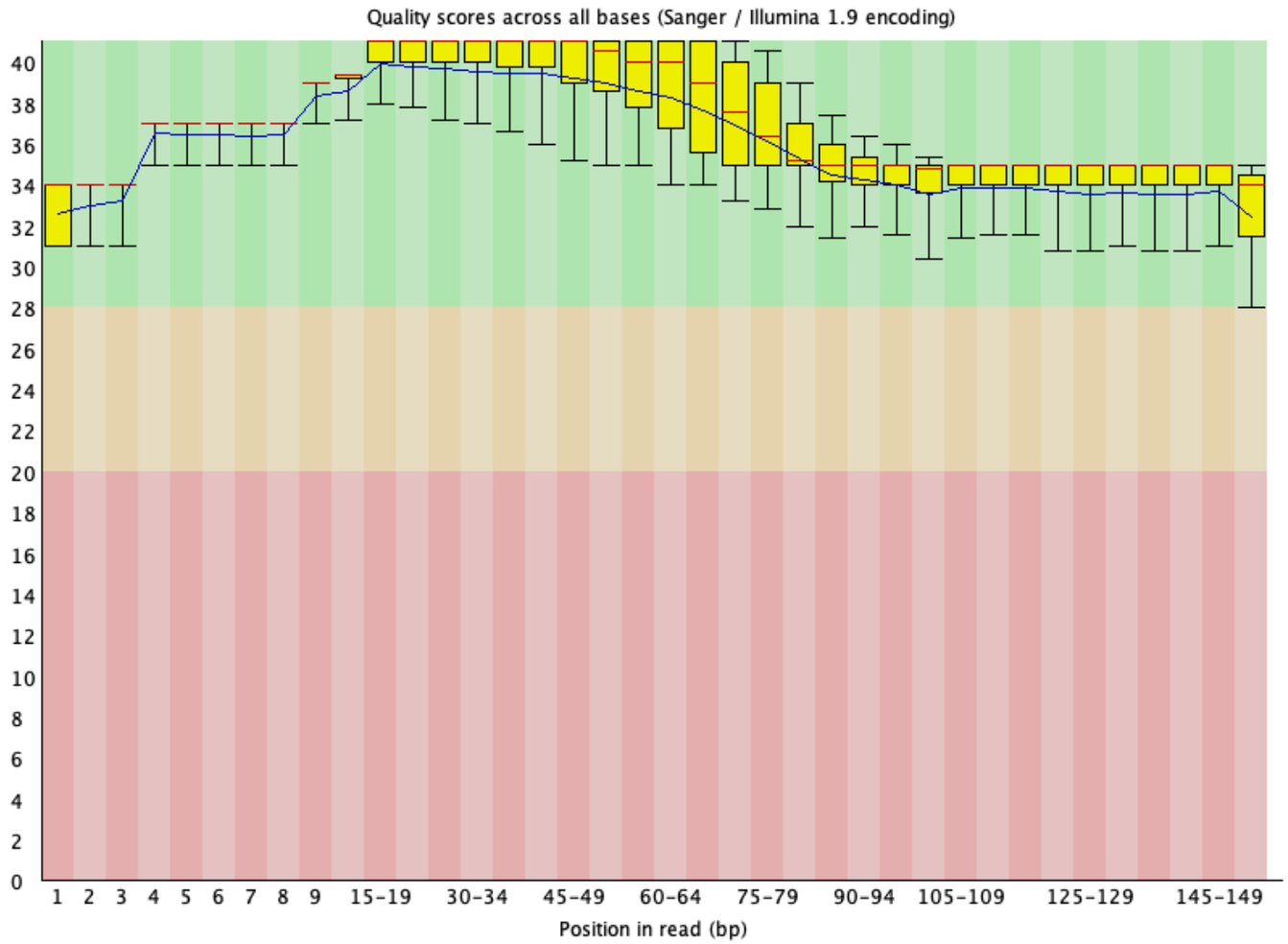
```
This is cutadapt 3.4 with Python 3.9.2
Command line parameters: …
Processsing reads …
```

Rerun FastQC

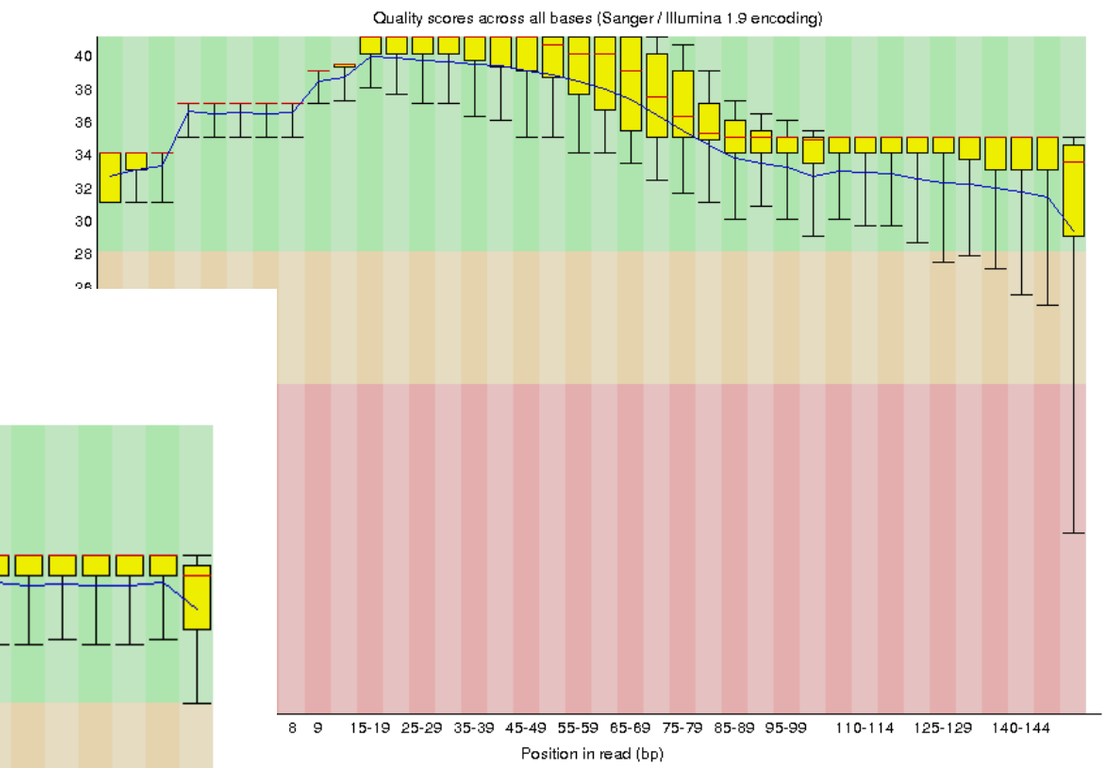
Filtered read quality check

Using the FastQC software

Per base sequence quality



Per base sequence quality



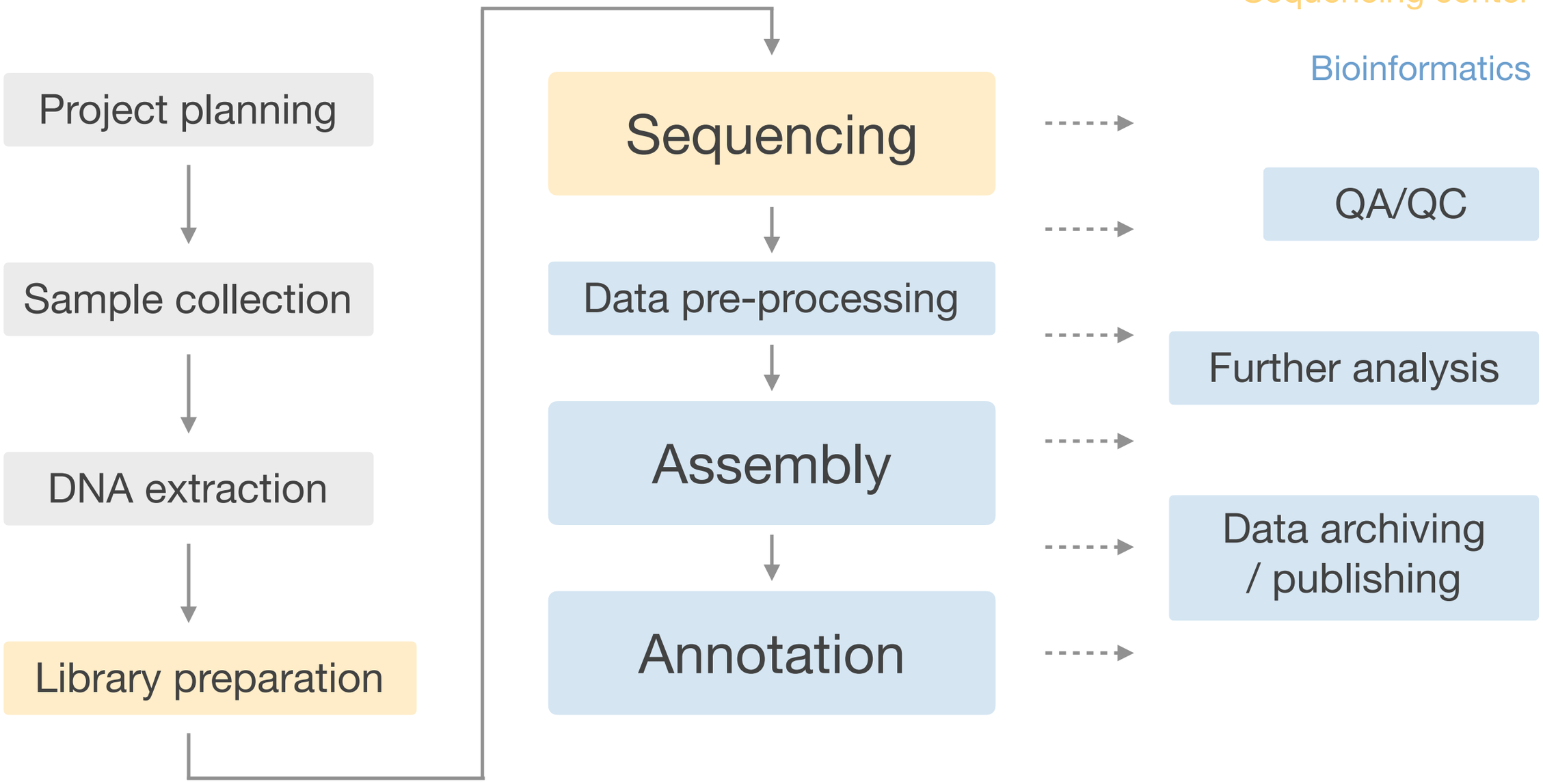
Raw reads

Trimmed reads

Data pre-processing summary

- Raw read quality assessment
- Adapter removal
- Quality filtering (trim low-quality bases / reads)
- Check for contamination (reads of viral, bacterial, human origin)
- Trimmed read quality assessment

De novo genome sequencing workflow

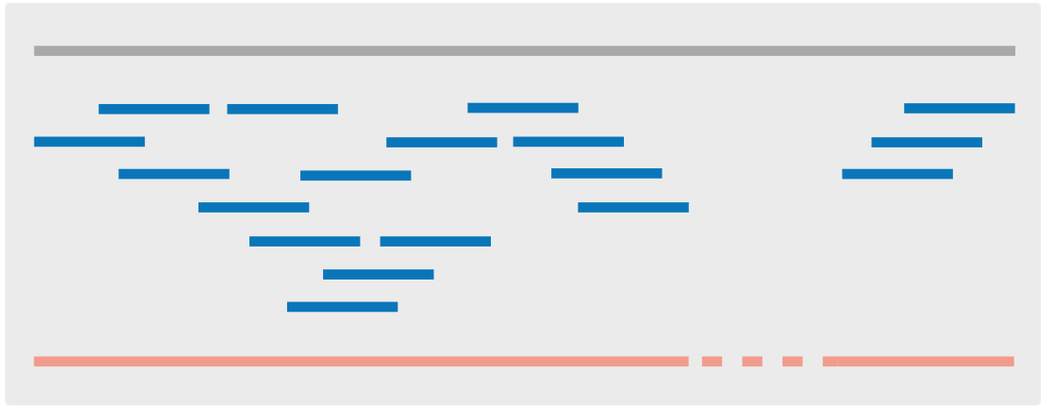


3. Assembly and annotation

- Genome assembly
- Metrics and assembly QA/QC
- Re-sequencing and variant calling
- Annotation

De novo genome assembly

- Reconstructing long, continuous sequence (up to chromosomes) from millions of fragments (reads)

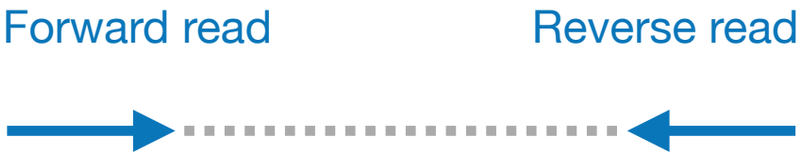


- Computationally difficult due to repetitive (identical or highly similar) DNA
- Levels of assembly: Reads > contigs > scaffolds (> chromosomes)

= draft assembly = chromosome-level

Short read assembly

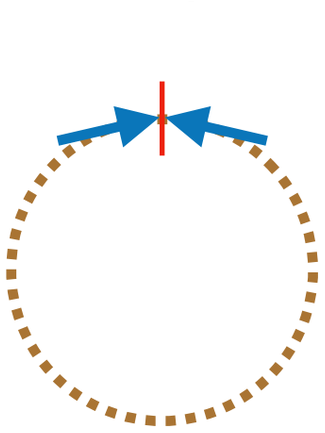
- Paired ends (PE)



Example file names

```
Hpue_raw300_F.fastq
Hpue_raw300_R.fastq
```

- Mate pairs

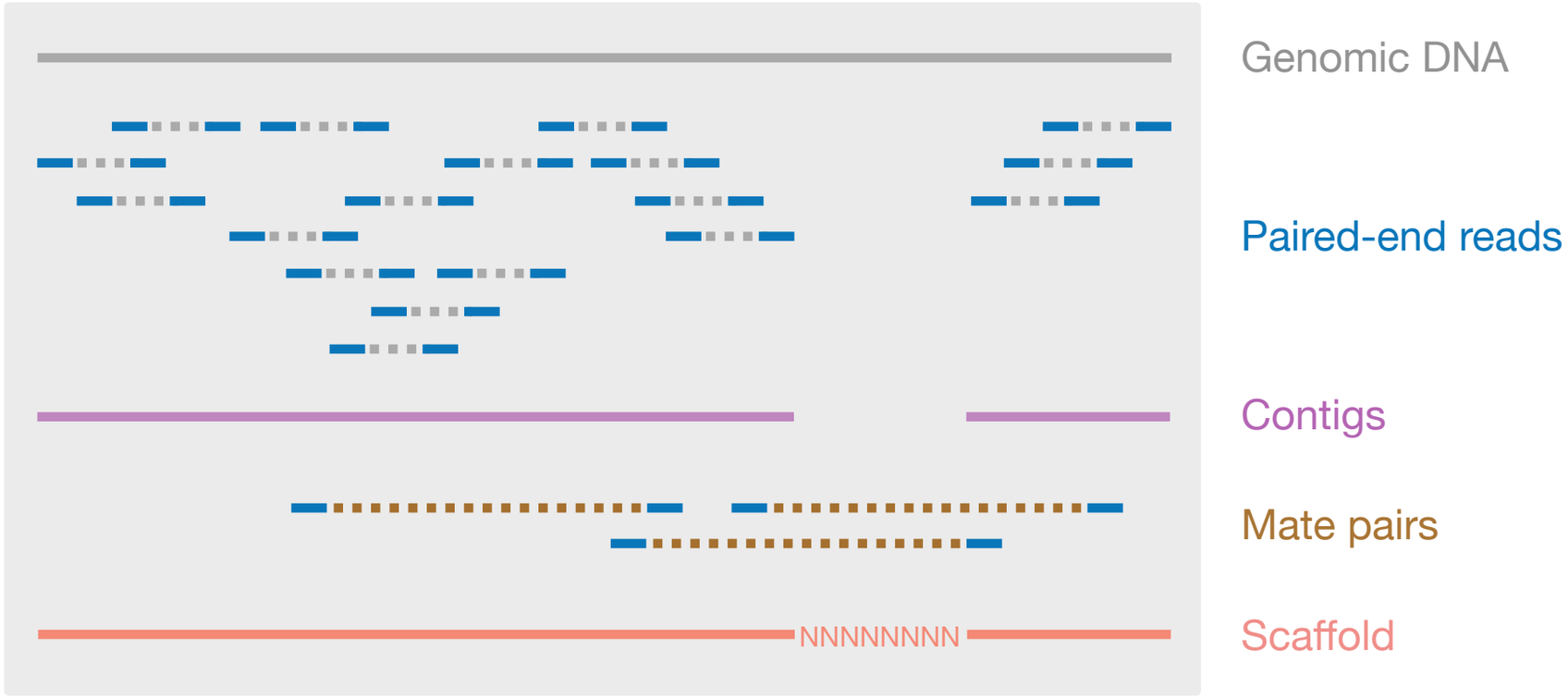


Library type	Insert sizes
5 × pair	250–580 bp
2 × mate	2.5, 4.3 kb

Example sequencing strategy

Short read assembly

- Overlapping reads > contigs
- Ordered, oriented contigs with gaps of known position, length > scaffolds (gaps spanned by mate pairs)



Hybrid strategy

- Combine Illumina with long read sequencing
- Correct long reads (low coverage) with high coverage Illumina data
- Fill gaps in Illumina-based scaffolds with long reads (> up to chromosomes)
- Current state-of-the-art

Assemblers

- Illumina: ALLPATHS, ABySS, Discovar, Platanus, SOAPdenovo
- PacBio: Falcon, Peregrine, Canu (also Nanopore)
- 10x Genomics: Supernova
- Small genomes: SPAdes, Velvet, HGAP, MIRA

- Command-line based, though some graphical user interfaces are available for smaller genomes (< 100 Mb, e.g. Geneious)
- Expect run-times of multiple hours or days, depending on genome

De Bruijn graphs

Example:

6-mers of 10 bp-sequence

AGTCAAGTCT

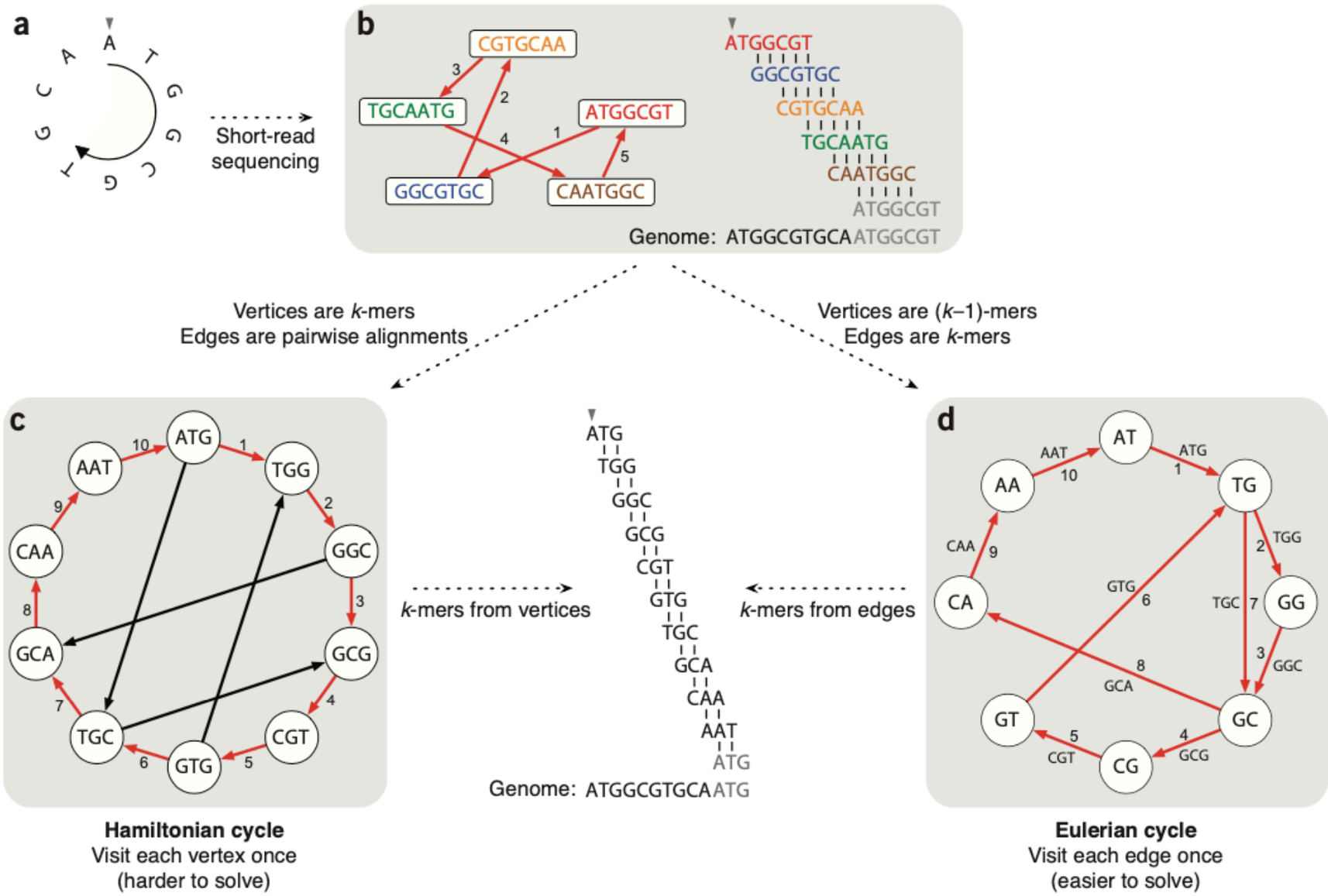
AGTCAA

GTCAAG

TCAAGT

CAAGTC

AAGTCT



Compeau and Tessler 2011 (Nature Biotechnology)

De Bruijn graphs

Key assembly parameter: k-mer size *k*

- larger *k*: more specific, lower coverage, risk of under-assembly
- smaller *k*: less specific, higher coverage, risk of over-assembly

General advice

- Compare alternate assemblies (consider use case)
- Even high-quality assemblies should be considered drafts
- Expect lower quality in difficult (e.g. highly repetitive / heterozygous) regions

Assembly example

The FASTA format

```
cd workshop_zmt/project/3_assembly                    # change directory
zcat < Hpue_assembly_01_LG12.fas.gz | head -n 100     # look at compressed file (zcat),
                                                      # display first 100 lines
```

```
>UXGA01000012.1 Hypoplectrus puella genome assembly, contig: LG12, whole genome shotgun
sequence
CCAAAAGTAGTTACAGTCCATGCGGCAAGTCCAGaactctctgcaccagactttTTTGAAGAAATGAGAGAGTTTATGCG
CCGGTCTAAACAAACTGAGGCTGCACTTTTTGAGCAGATTAAACAGCTCAGAAGGGACATGAGACCAAGATCAAGGCAAG
ACTGGACCTCTGAAGAACCTTTTGAACAACCTGTTTCTTCATCTCCTCAGCTGATAGCCAGTGGACCTGTCTCTTCATCT
…
TGGGGATTAAGAGACGAAACACGTTTAAACTCACATGAAACACTAAACTGATACATAGNNNNNNNNNNNNNNNNNNNNN
NNNTCGGTCAGAAAAGTCTGTGCCTGTTTGTCAAAGATGATTTTTGAACTCATGAGCTTGTCTCACTGTCTCTGGTGTAT
GTTAATCAAGTGTACTTCCCGATAATGGAAACCTCTGATAACAGCTCTTTTGTTGCCGCTGCTGATAAGCCCTGCAAAGC
```

Assessing assembly quality

- Assembly metrics: size distribution of contigs / scaffolds
- Gaps per Gb, unplaced contigs
- Base accuracy (Q score)
- Presence of false duplications
- Gene completeness
- Percentage of assembly assigned to chromosomes

Assembly metrics

- Total size (compare to expected genome size)
- Number of contigs / scaffolds
- Largest scaffold
- N50: contig / scaffold size where 50% of assembly is found on contigs / scaffolds of equal or larger size (measure for sequence continuity)

```
Scaffolds: 530, 760, 1050, 610, 450, 800, 220, and 1200 kb
Reorder: 1200, 1050, 800, 760, 610, 530, 450, 220 kb
Sum / 2: 5620 / 2 = 2810
Add up until reached: 1200 + 1050 + 800 > 2810
N50 = 800 kb
```

Example calculation

Assembly metrics

Using QUAST (command line tool or http://cab.cc.spbu.ru/quast/)

Report Example

E. coli single-cell assemblies

Aligned to "e.coli_reference" | 4 639 675 bp | 50.79% G+C | 884 operons

All statistics are based on contigs of size >= 500 bp, unless otherwise noted (e.g., "# contigs (>= 0 bp)" and "Total length (>= 0 bp)" include all contigs).

Show heatmap

Worst Median Best

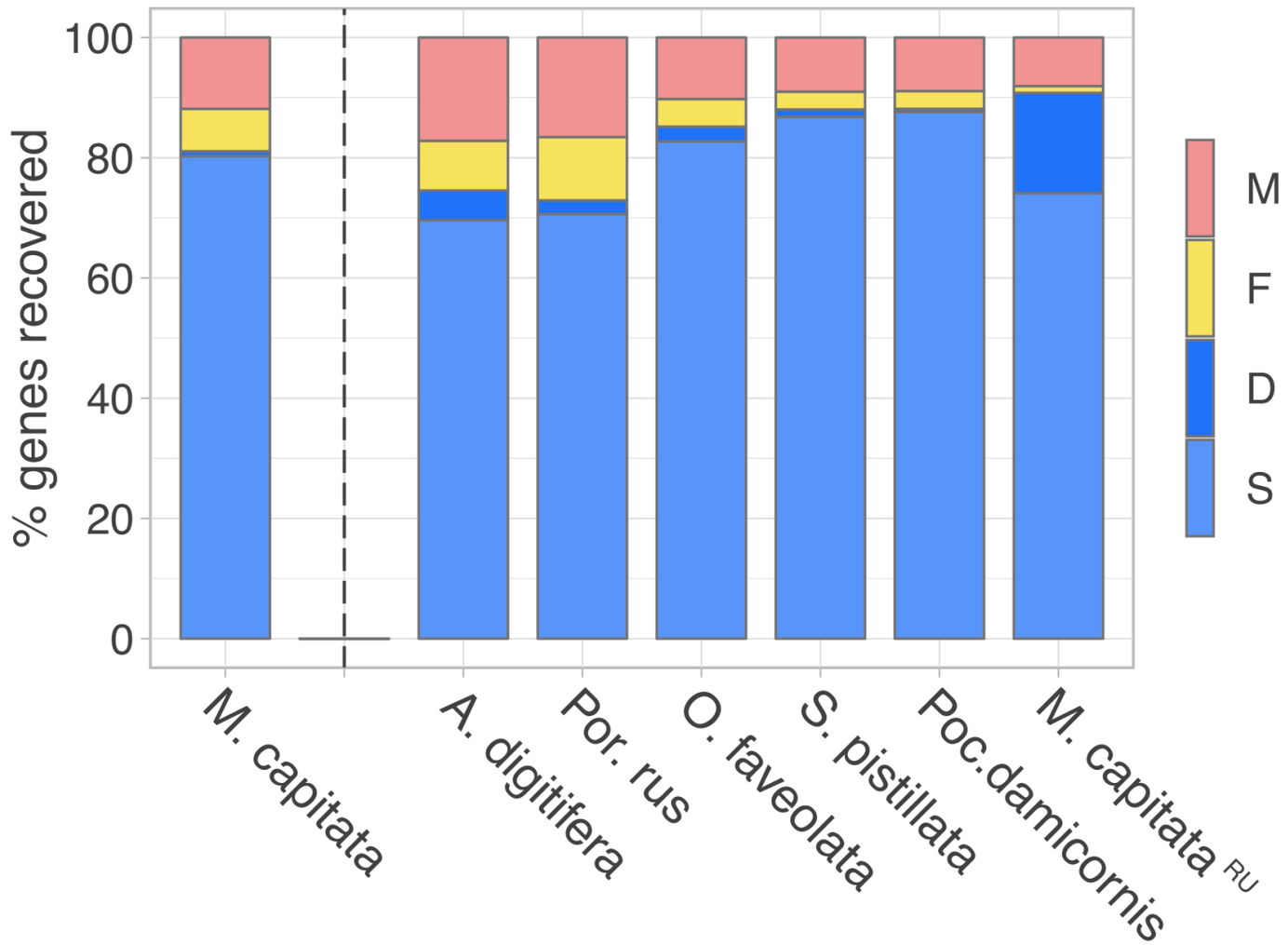
Statistics without reference	E+V-SC	IDBA-UD	SPAdes-2.4	Velvet-SC
# contigs	344	250	277	519
Largest contig	132 865	224 018	269 177	121 367
Total length	4 540 286	4 791 744	4 877 521	4 526 656
N50	33 616	96 947	106 927	20 445
Misassemblies				
# misassemblies	2	9	2	2
Misassembled contigs length	23 485	66 335	26 551	22 359
Mismatches				
# mismatches per 100 kbp	2.26	3.65	5.06	1.77
# indels per 100 kbp	0.7	0.2	0.7	0.92
# N's per 100 kbp	0	0	4.86	0
Genome statistics				
Genome fraction (%)	91.727	94.943	95.759	91.43
Duplication ratio	1.001	1.001	1.004	1.002
# genes	3767 + 160 part	4026 + 80 part	4046 + 102 part	3630 + 288 part
# operons	723 + 87 part	802 + 40 part	809 + 48 part	650 + 158 part
NGA50	32 051	96 947	110 539	19 791
Predicted genes				
# predicted genes (unique)	4258	4394	4417	4331
# predicted genes (>= 0 bp)	4258	4394	4490	4331
# predicted genes (>= 300 bp)	3643	3736	3784	3666
# predicted genes (>= 1500 bp)	524	559	559	515
# predicted genes (>= 3000 bp)	44	49	48	39

Extended report

Gene completeness

Using BUSCO (https://busco.ezlab.org)

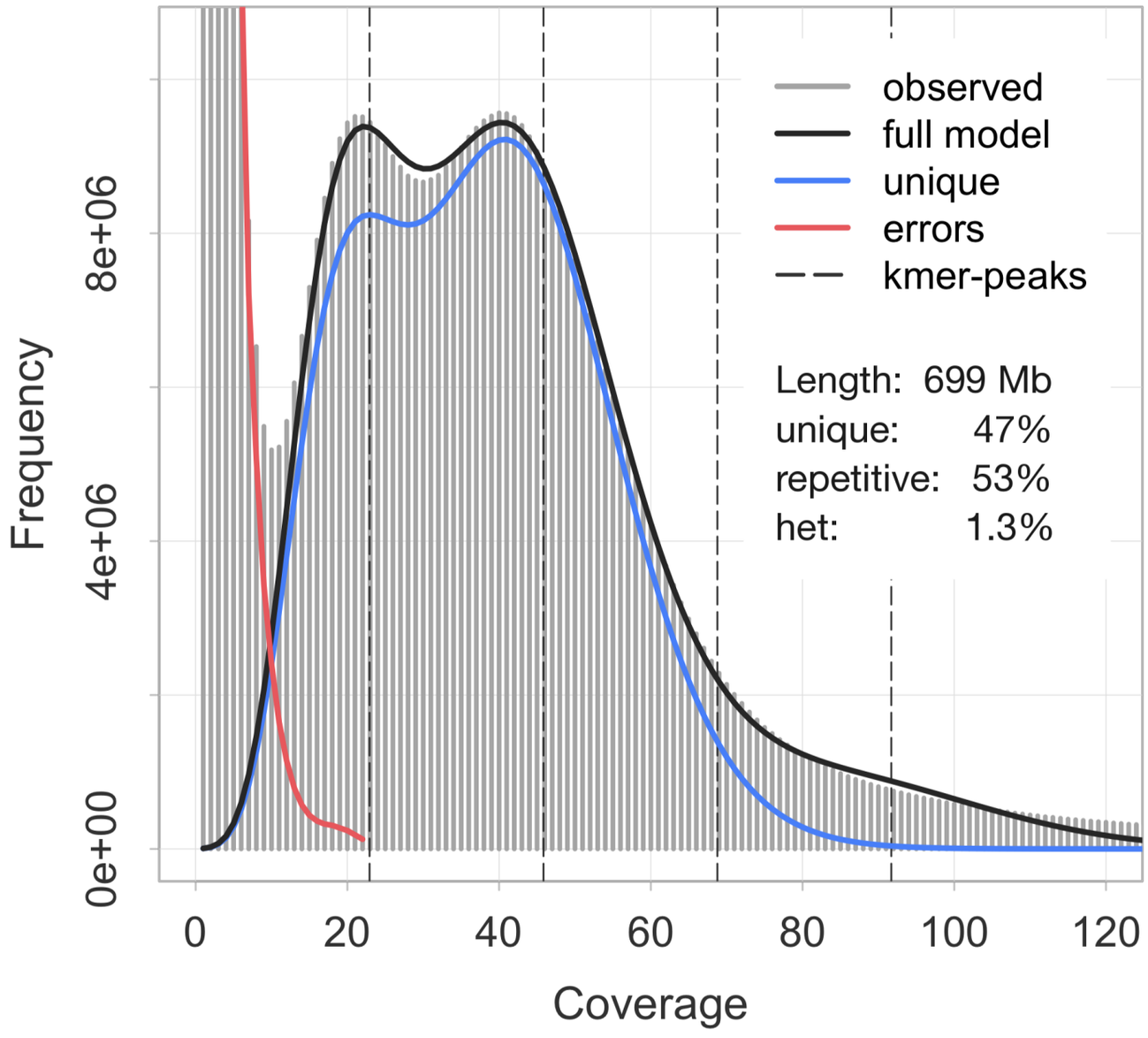
Quantifies assembly completeness based on presence of universal, highly conserved, single-copy genes (e.g. housekeeping genes)



Helmkampf et al. 2019 (Genome Biology and Evolution)

Genome assessment using *k*-mer profiles

Using GenomeScope (http://qb.cshl.edu/genomescope/)



Example:

6-mers of 10 bp-sequence

```
AGTCAAGTCT
AGTCAA
 GTCAAG
  TCAAGT
   CAAGTC
    AAGTCT
```

Helmkampf et al. 2019 (Genome Biology and Evolution)

Re-sequencing

- Prerequisite: high-quality reference genome
- Approach
    - low-coverage sequencing of multiple samples
    - aligning reads to reference (mapping)
    - identify differences > SNPs (variant calling / genotyping)

Re-sequencing

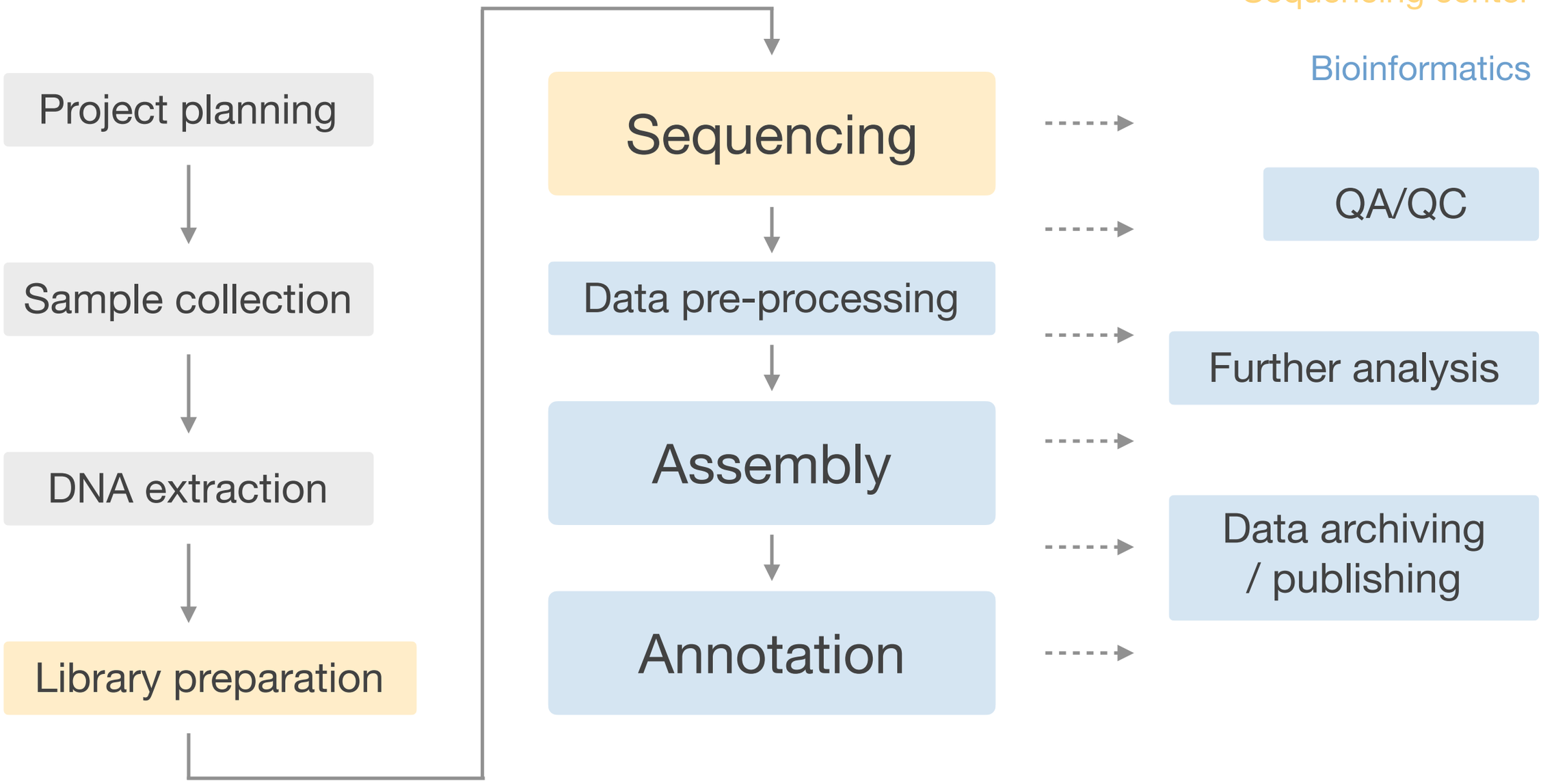
The VCF format

```
##fileformat=VCFv4.1
…
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT sampleA sampleB sampleC sampleD sampleE
LG01 1258 . C T . PASS . GT 0|0 0|0 0|0 0|0 0|0
LG01 1277 . G T . PASS . GT 0|0 0|0 0|0 0|0 0|0
LG01 1292 . G A . PASS . GT 0|0 0|0 0|0 0|0 0|0
LG01 1365 . C A . PASS . GT 0|0 0|0 0|0 0|0 0|0
LG01 1373 . C T . PASS . GT 0|1 0|1 1|1 0|0 1|1
LG01 1398 . G C . PASS . GT 0|1 0|1 1|1 0|0 1|1
LG01 1403 . G C . PASS . GT 0|0 0|0 0|0 0|0 0|0
LG01 1494 . C T . PASS . GT 1|0 0|0 0|0 1|1 0|1
LG01 1495 . G A . PASS . GT 0|1 1|0 1|1 0|0 1|0
…
```

A quick note on metagenomics

- Similar workflow: shotgun sequencing, read pre-processing, QA/AC
- Assembly stage: dedicated software like metaSPAdes, MetaQUAST
- Extra steps: binning, taxonomic profiling
- Annotation geared towards prokaryotes

De novo genome sequencing workflow



Genome annotation

Identifying the location, structure, and putative function of genome features

- repetitive and mobile DNA
- protein-coding genes
- regulatory elements

Repeat annotation

- Most genomes contain large amounts of repetitive and mobile DNA
- Repeat classes:

Repetitive Elements Identified in the *Montipora capitata* Genome Assembly

	Number	Total Length (Mb)	Fraction of Assembly (%)
Tandem repeats	160,985	16.5	2.7
Interspersed repeats	1,028,006	257.4	41.9
DNA elements	31,667	11.6	1.9
LTR elements	14,753	10.0	1.6
Non-LTR elements	110,728	42.6	6.9
Unclassified	870,858	193.2	31.4

- Repeats may have regulatory or structural functions, but can also bias downstream analyses

Helmkampf et al. 2019 (Genome Biology and Evolution)

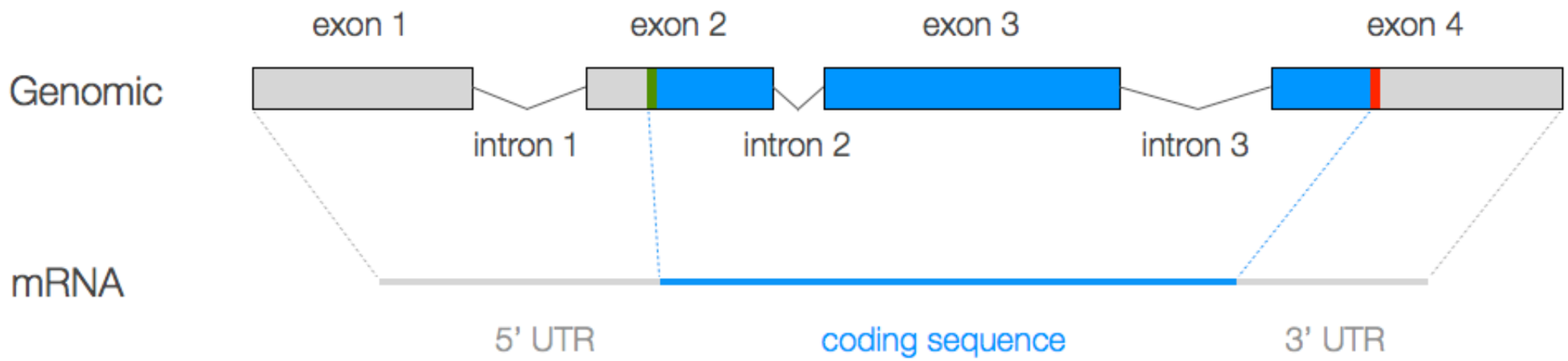
Repeat annotation

- RepeatModeler combines multiple repeat search engines and curated databases (e.g. Repbase) to identify repeats
- RepeatMasker marks repeats in assembly (replacement with Ns or soft-masking)

```
>Sc0000000
TACTGGCAGTCCAATCTGTCACCCTTGATTATAATTTCATTCAAATAATTTCTTTTTTTATCCAATGATCAACCAAATGCATTGCTTGCATTAAAAACTC
TTGTCAGCATTAAGTCCTCAGATATGGATCAACAACACAGTAAAACCATTTAATCACTAAACTTTAAAATAATAATTATTATATTGAAGAATCTAAAtca
tcatttctttcacaggaacacacaagcccaacaaattgacctgttcccaacagattggcttcatagctcagttggttctagcaattgcattcatttggta
tcacagaggtaatgggttcaaatcccattggagctgcctgaatttttgggtgtctTTTAAAGTGACTCAATCAATTCTATCTTCTCCCTGAATGCAGCG
TGTTAAGTATGCTTTTTTGAAACACAAGTTTGAGAATCCTGTACTGCAAACTAAATCCTTTAGTTTGCAGTGTATacacacacacacacacatacTGGAA
AATTACGGGTAAGAAATGTCAATCTCTCCTCATTCCAGTTCTCTCCAAAATAATGTTGAAGTTGAGGTAATGGTGGACCACTGAATAGTGAAATTCCCCT
GTTTTCATCCTTGTTATGTTTTACAATCACATGTCACATTGATTAATGGAAATATAATCCCCCTTTAGAACCAGATTGCAGACAGTCATTCTTTAACTTG
…
```

Gene prediction

In eukaryotes, prediction of protein-coding genes is difficult due to coding sequence (exons) being interrupted by introns



Gene prediction

- Complex pipelines (e.g. AUGUSTUS) combine multiple approaches
    - *ab initio* prediction (pattern-based)
    - extrinsic evidence can include
        - RNA-seq (transcripts)
        - homologous protein sequences
    - Sensitivity and specificity are increased by multiple rounds of training
    - Repeats should be masked
- Computational gene prediction always requires careful quality control

Gene prediction

The GFF format

```
# start gene g29
Sc0000000 AUGUSTUS gene 656020 658053 0.9 - . g29
Sc0000000 AUGUSTUS transcript 656020 658053 0.9 - . g29.t1
Sc0000000 AUGUSTUS stop_codon 656020 656022 . - 0 transcript_id "g29.t1"; gene_id "g29";
Sc0000000 AUGUSTUS intron 656165 657273 1 - . transcript_id "g29.t1"; gene_id "g29";
Sc0000000 AUGUSTUS intron 657509 657956 1 - . transcript_id "g29.t1"; gene_id "g29";
Sc0000000 AUGUSTUS CDS 656020 656164 1 - 1 transcript_id "g29.t1"; gene_id "g29";
Sc0000000 AUGUSTUS CDS 657274 657508 1 - 2 transcript_id "g29.t1"; gene_id "g29";
Sc0000000 AUGUSTUS CDS 657957 658053 0.9 - 0 transcript_id "g29.t1"; gene_id "g29";
Sc0000000 AUGUSTUS start_codon 658051 658053 . - 0 transcript_id "g29.t1"; gene_id "g29";
# coding sequence = [atgacccgcttcgaaagcgacgattctcaattggatgaagatagtgatcatgggaattgttcagcaaaacggttggccc
# ccaaaagaaaaatttcaggtcggttgcgcaggaaatcaaaagaatcatatcgcctgggtcccaaggttgatgtaactccagcgagcagtgacgatgaa
# ctggagcgggaaacgtcttttatgaactctggtacctttaaaccagtcgagggtttttccaaccctatacccgtgagctccggagaggacgactttga
# tgaccttagtgttggagacatgaaaaagcgtcggagctcaaaaacgaaagaattccgacaacgcgaagatacccacgatacacctgatttgacagaca
# atgagggagagctggggtttaacacctttaagcccgcctcttacgtcaacgaaggtgcccacacttcggatgatgccgaatctgaactggaaaaggaa
# aaatga]
# protein sequence = [MTRFESDDSQLDEDSDHGNCSAKRLAPKRKISGRLRRKSKESYRLGPKVDVTPASSDDELERETSFMNSGTFKPVEGF
# SNPIPVSSGEDDFDDLSVGDMKKRRSSKTKEFRQREDTHDTPDLTDNEGELGFNTFKPASYVNEGAHTSDDAESELEKEK]
…
```

Gene prediction

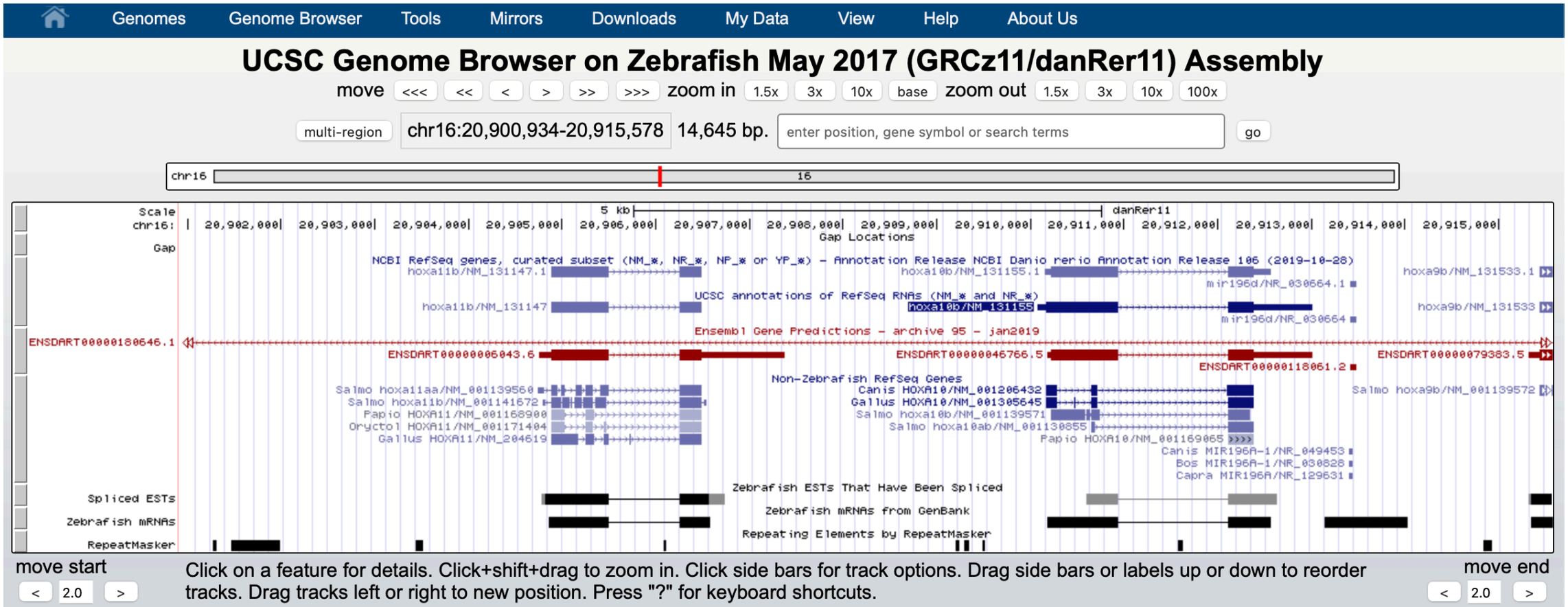
- Gene structure (UTRs, exons, introns)
- Coding sequence
- Transcription status and splice variants
- Putative identity and function

  mostly homology-based, e.g. by BLAST versus UniProt (uniprot.org) database

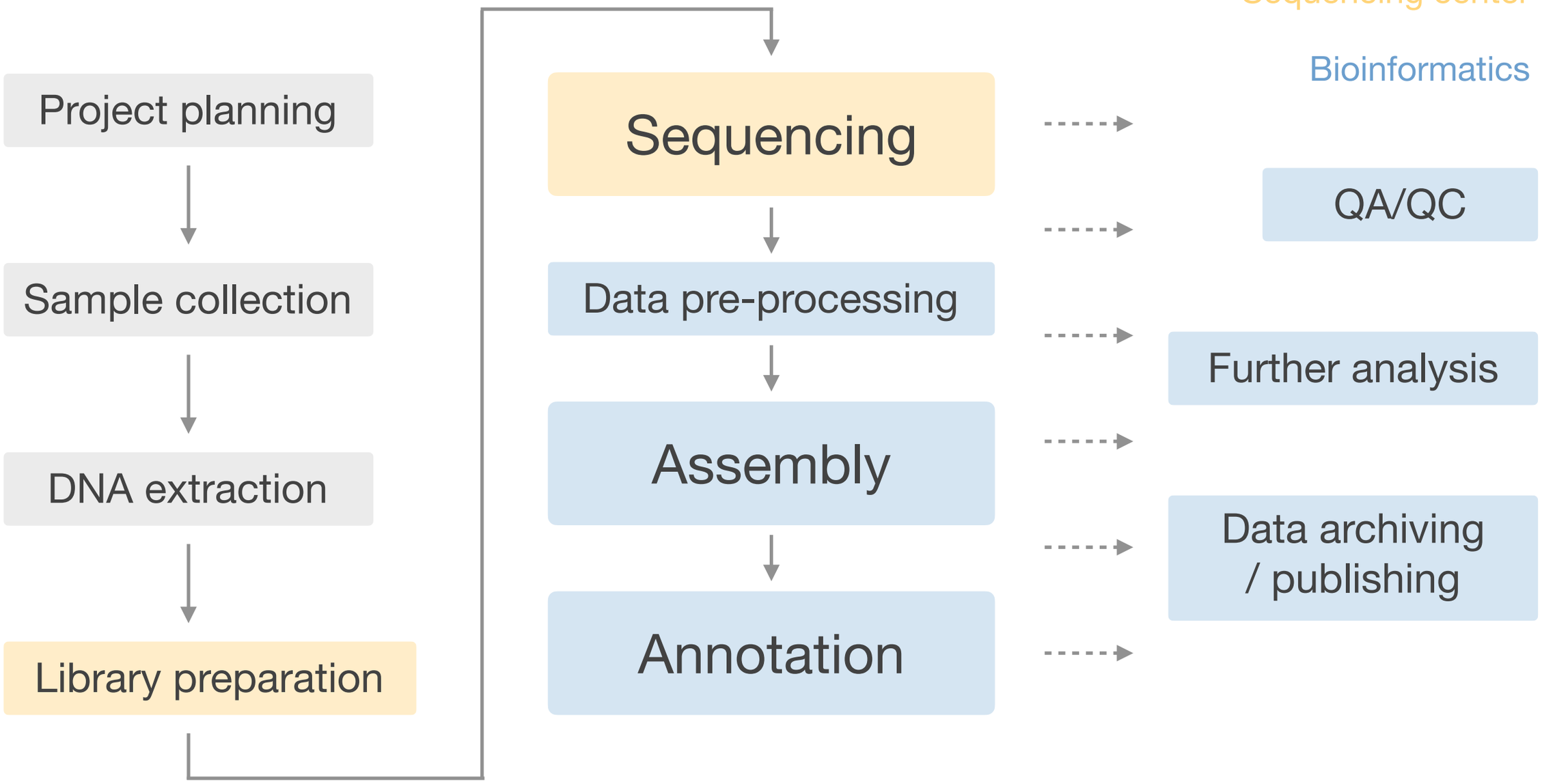
- Computational + manually curated gene models = Official Gene Set

Visualization

GBrowse / JBrowse genome browser



De novo genome sequencing workflow



4. Data management

- Documentation
- Data organization
- Backing up / Version control
- Publishing / Sharing data

Documentation

- Document your workflow in plain text format (Atom / TextEdit / Notepad)
- Include code, software versions, summary results, and comments

Worklog_genome_v4.txt

```
### Assembly v4

## Supernova v2.1.1 (mcap_v4)

supernova run --id=mcap10v4 --fastqs=~/proj/coral/genome/raw_data/raw/ --maxreads='all'

supernova mkoutput --style=pseudohap2 --asmdir=/mcap10v4/outs/assembly --outprefix=ph2

> mcap_v4.fas, mcap_v4h.fas (57431 scaffolds)

supernova upload mhelmkam@hawaii.edu mcap10v4/mcap10v4.mri.tgz

python ~/apps/quast_5.0.0/quast.py -e -s mcap_v4.fas   # no minimum contig length
python ~/apps/quast_5.0.0/quast.py -l -s -k mcap_v4.fas   # imposes -m 3000 -e, adds k-mer
analysis

> quast/mcap_v4

|

### Assembly polishing (v4 and v2 revision)

## Self-Blast to identify duplicates (also performed for v4)

blastn -task megablast -query Mcap_genome_v2.fas -db ../blast_db/Mcap_genome_v2.fas -out v2-
self.blast -evalue 1e-10 -max_target_seqs 2 -outfmt "6 qseqid sseqid pident length qlen slen
qstart qend sstart send evalue" -num_threads 10
blastn -task megablast -query mcap_v4.fas -db ../blast_db/mcap_v4.fas -out v4-self.blast
-evalue 1e-10 -max_target_seqs 2 -outfmt "6 qseqid sseqid pident length qlen slen qstart qend
sstart send evalue" -num_threads 10
```

Data organization

Formats

- Standard file formats (e.g. plain text, pdf, png)
- Standard data formats (e.g. FASTA, FASTQ, VCF, GFF)

Data organization

Naming scheme

- Uniform, persistent
- Comprehensive, includes important information
- No spaces and special characters (use underscores instead)
- Consistent file extensions (e.g. .fastq, .vcf, .gff)
- May include date (YYYYMMDD) or version number

Data organization

Naming scheme examples

What is it?

Species acronym

```
Hpue_raw300_F.fastq.gz
Mcap_genemodels_1.1_aa.fas
hyp155_a_0.33_mac4_5kb.raxml.log
```

File / data format

Dataset name

Version number

Important parameters

Data organization

Folder structure

- Be consistent
- Separate ongoing and completed work
- Consider links instead of duplicating files
- Include documentation (readme or worklog)

Example

```
Project
├── 0_metadata/
├── 1_rawdata/
├── 2_processing/
├── 3_assembly/
├── 4_annotation/
└── README.txt   # or WORKLOG.txt
```

Storing big data

- Keep raw data on dedicated hard drive
- Use file compression (e.g. `gzip`)
- Transfer working copies to computing cluster or cloud service, if possible

Transferring big data

- Use improved copy tools (e.g. `rsync`) to avoid loss of progress or data corruption in case of connection issues
- Confirm data integrity using checksums after transfer

```
rsync -arvz <source_file> <destination_file>   # improved copy (-a: archive mode, -r:
                                               # include subdirectories, -z: compress
                                               # during transfer)

md5 <file>                                     # calculate checksum (not working in Git
                                               # Bash)
```

Version control – git

- Tracks changes in files
- Can be used to back up and share data via GitHub
- Keep track of code, workflow / log files, result files, plots etc.
- Unsuitable to back up large data files

Version control – git

Typical git usage

```
git init                                    # create git repository based on current folder
git status                                  # show repository status
git add *                                   # track all files (exceptions in .gitignore)
git commit -m “title”                       # record changes

git push                                    # push to remote GitHub repository (if set up)
```

Example .gitignore file

```
1_rawdata/
*.fastq.gz
*.fastq
```

Publishing / sharing genome data

FAIR principles — findable, accessible, interoperable, reusable

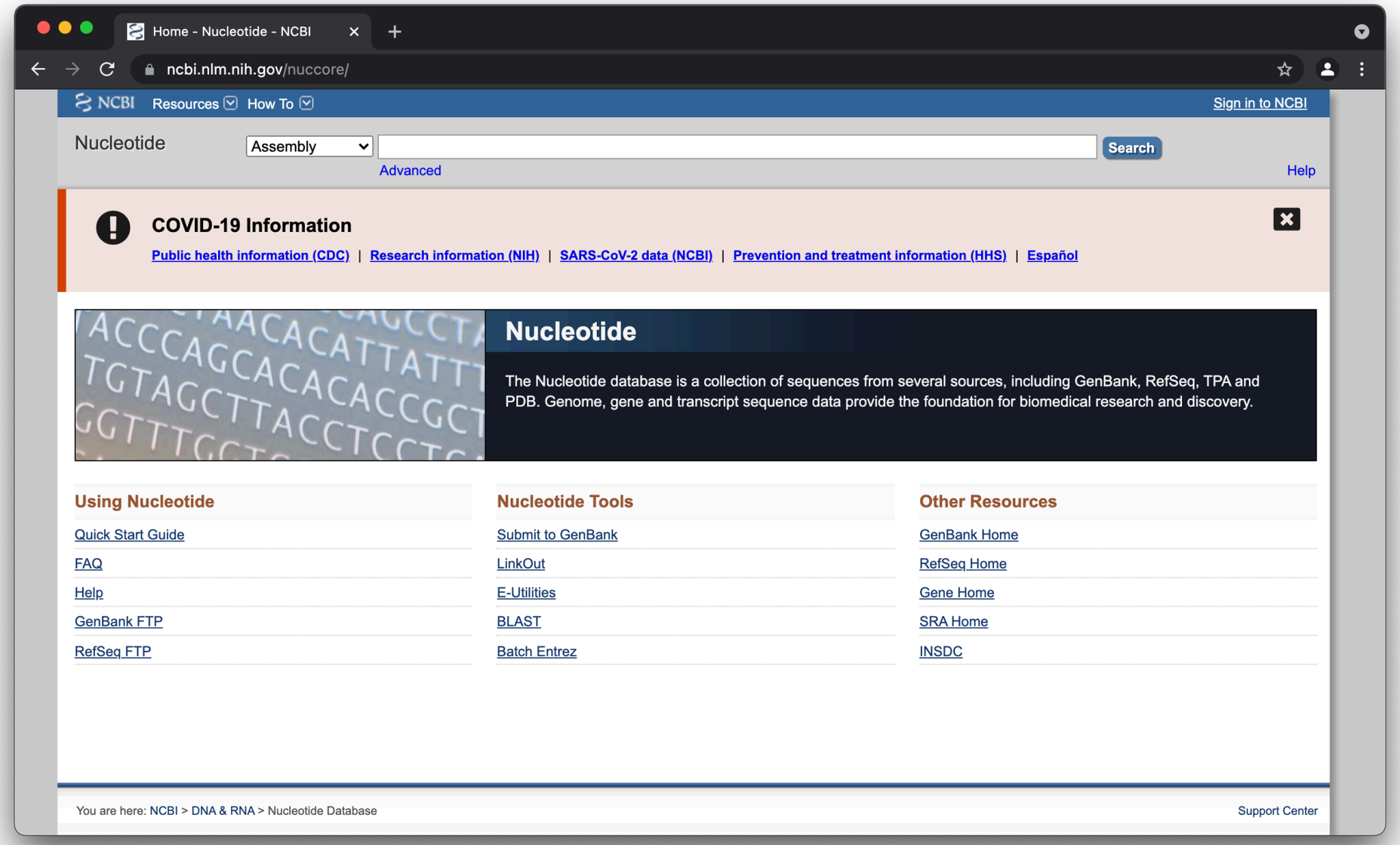
Key attributes:

- unique and persistent identifier
- clear and detailed metadata
- standardized data format and metadata vocabulary
- accessible data usage license (e.g. open access)

Public sequence databases

Public sequence databases

NCBI Genbank (https://www.ncbi.nlm.nih.gov/genbank/)



Select database:

Assembly

Search term:

Scleractinia

Useful search tags

[ACCN]

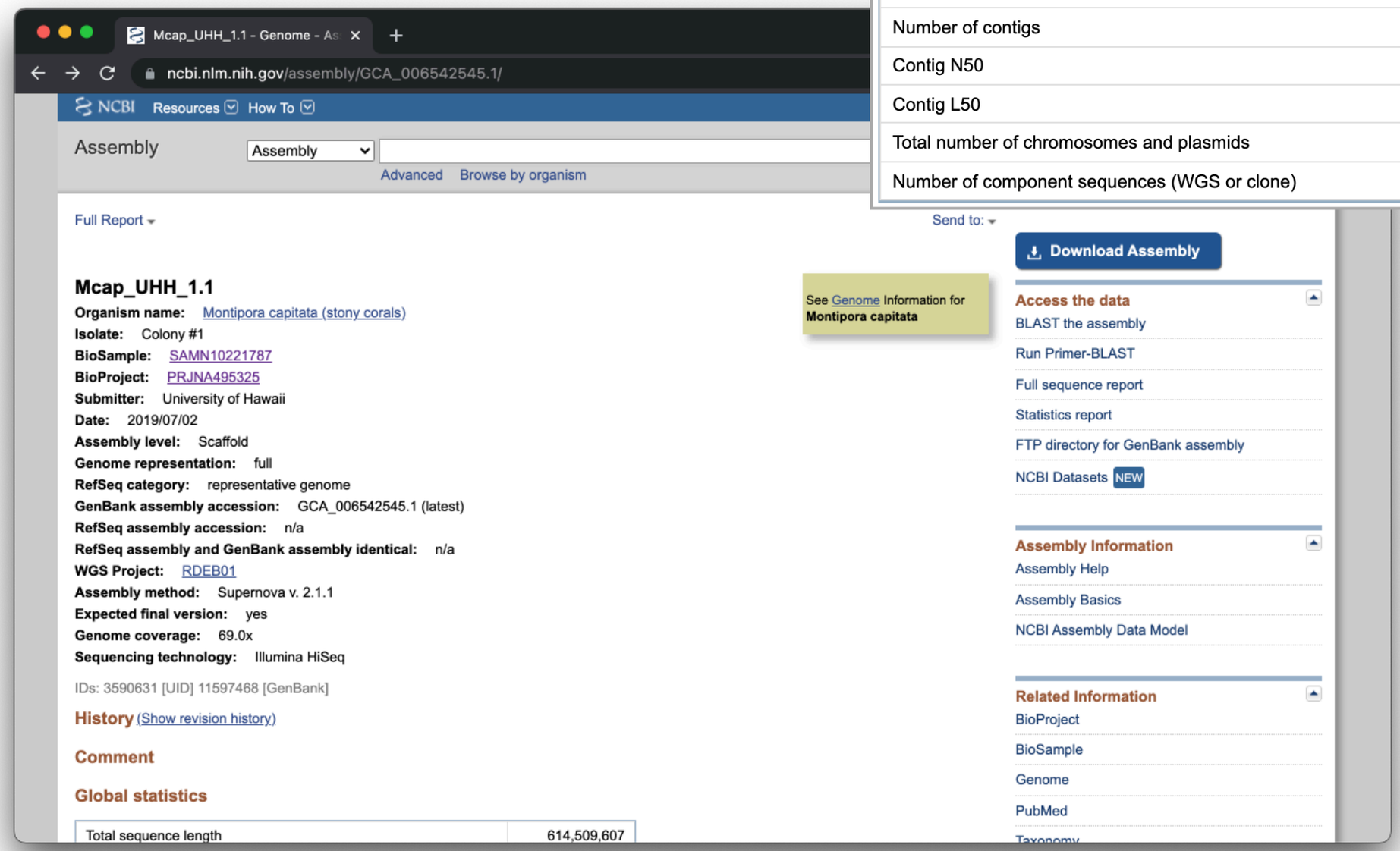
[ORGN]

[AUTH]

[TITL]

Public sequence databases

NCBI Genbank | Assembly report

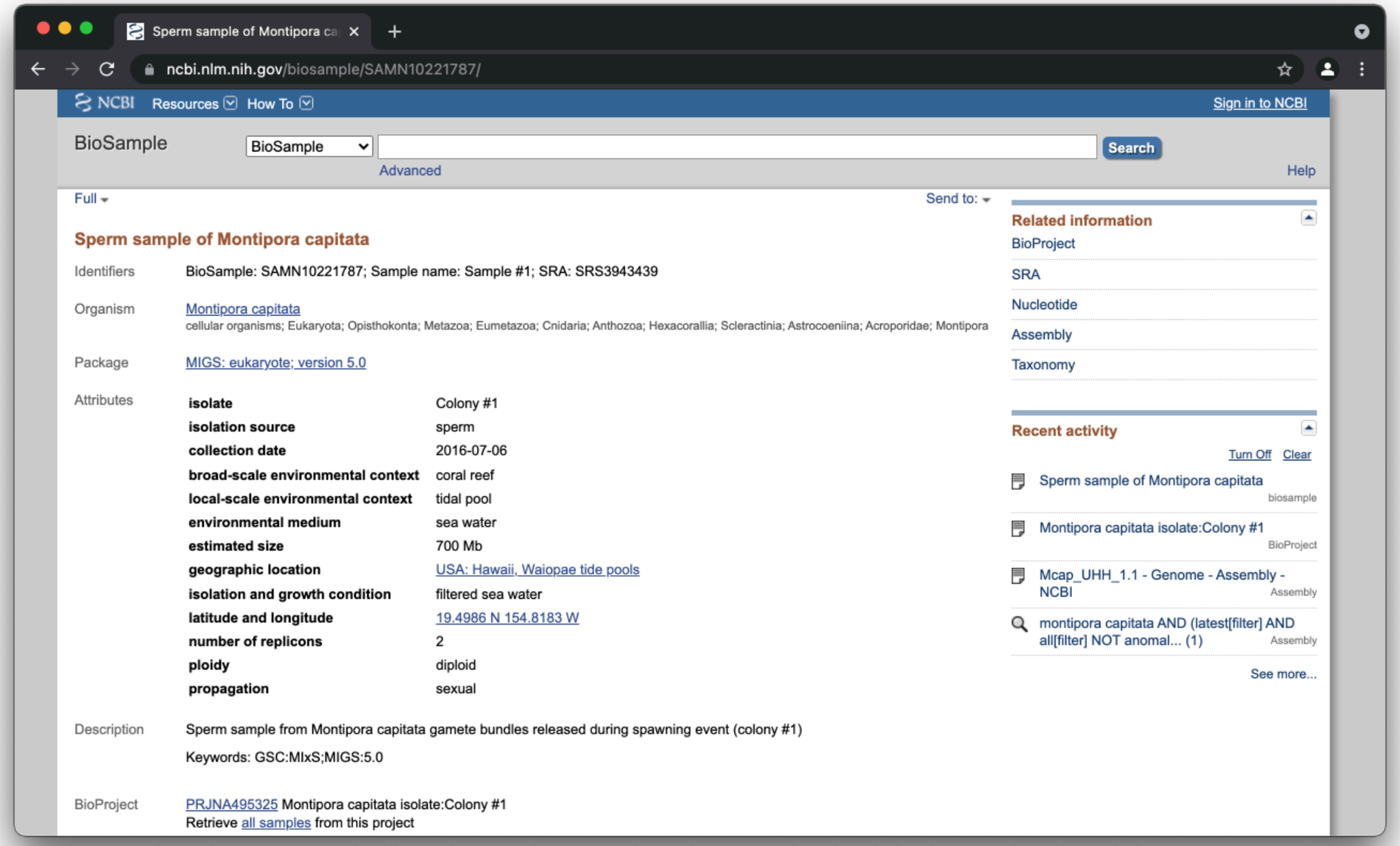


Global statistics

Total sequence length	614,509,607
Total ungapped length	571,886,739
Gaps between scaffolds	0
Number of scaffolds	27,865
Scaffold N50	185,537
Scaffold L50	747
Number of contigs	50,174
Contig N50	24,266
Contig L50	6,689
Total number of chromosomes and plasmids	0
Number of component sequences (WGS or clone)	27,865

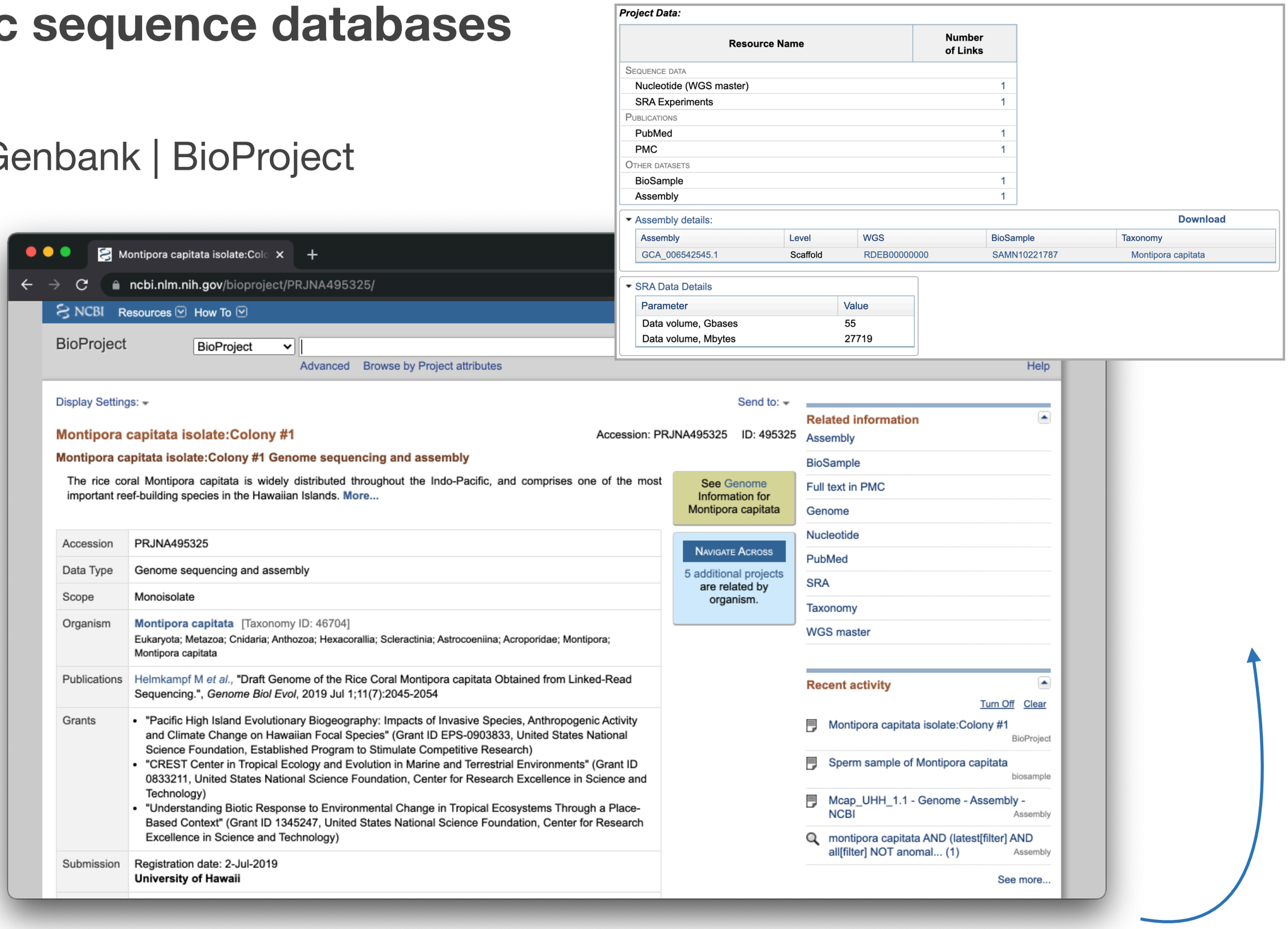
Public sequence databases

NCBI Genbank | BioSample



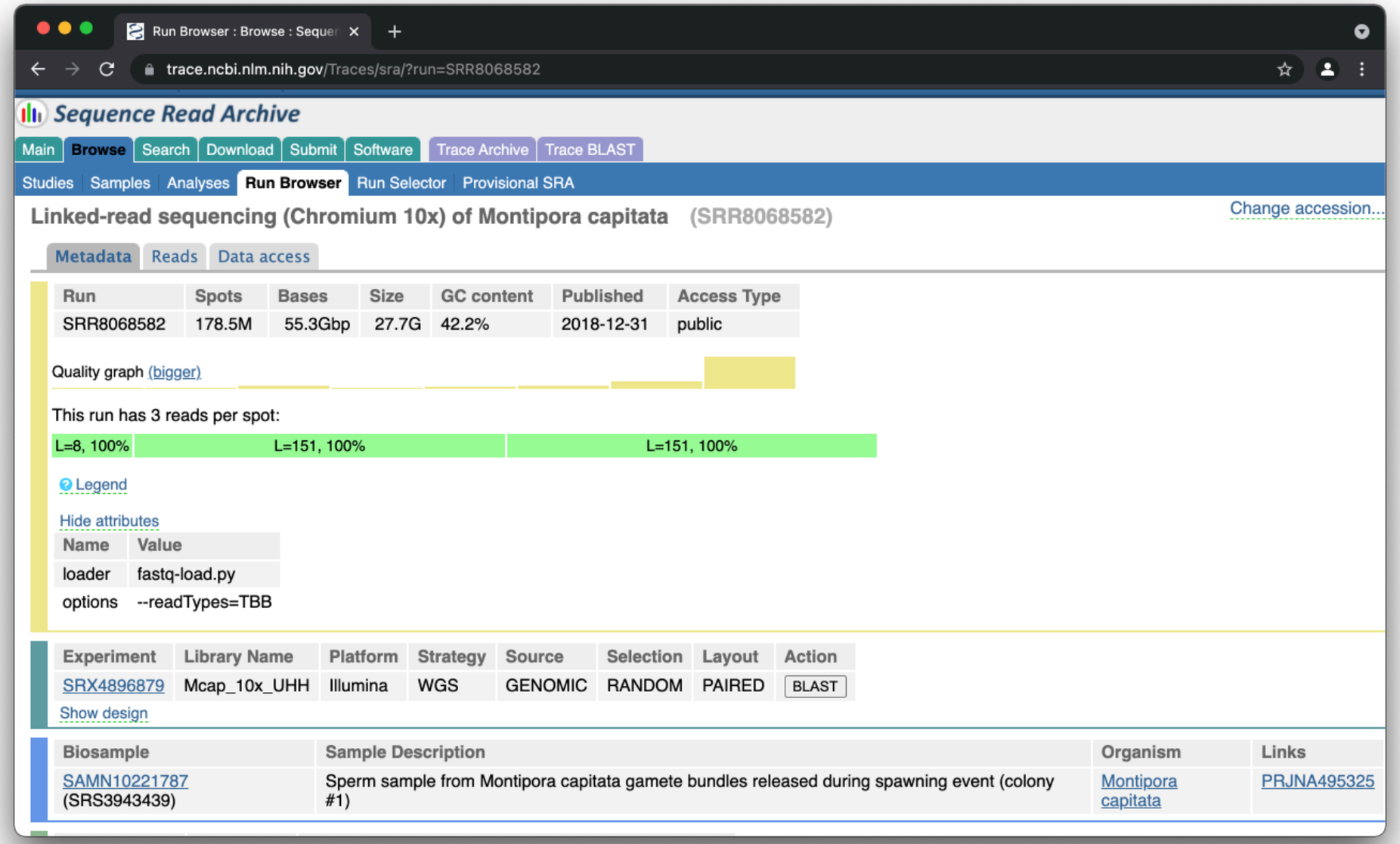
Public sequence databases

NCBI Genbank | BioProject



Public sequence databases

NCBI Genbank | Sequence Read Archive (SRA)



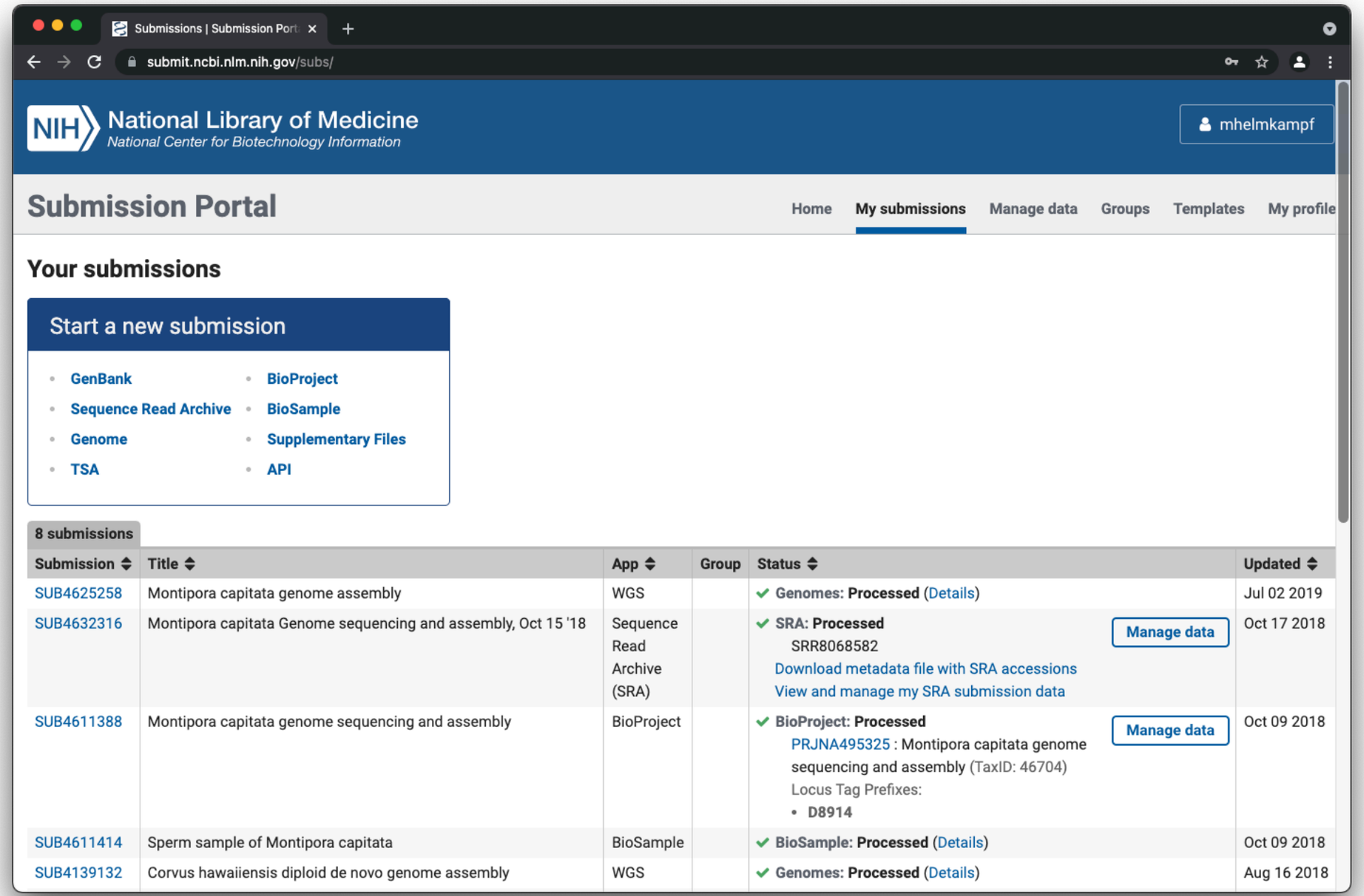
Genome submission

- NCBI genome submission guide:

  https://www.ncbi.nlm.nih.gov/genbank/genomesubmit/

- Submission portal login (https://submit.ncbi.nlm.nih.gov/subs/genome/)
  - Log in with NCBI, Google, or ORCiD etc. account
  - Start new or edit existing submission

Genome submission



New submission / register:

BioProject

BioSample

SRA

Genome (WGS)

Genome submission

- Provide project description, sample metadata, technical details (method, coverage etc.), contact information, publication if applicable
- Upload assembly as single FASTA file (minimum contig length 200 bp)
    - Chromosome-level?
    - What do ’Ns’ represent?
- Upload reads to SRA in FASTQ format
- Batch upload for multiple genomes

Genome submission

- Automated validation regarding:
    - Contamination
    - Adapter sequences
    - Duplicate contigs
- After successful review, accession numbers will be issued
- Public release at requested date or publication date, whichever comes first
- Data may run through database annotation pipelines

Dryad

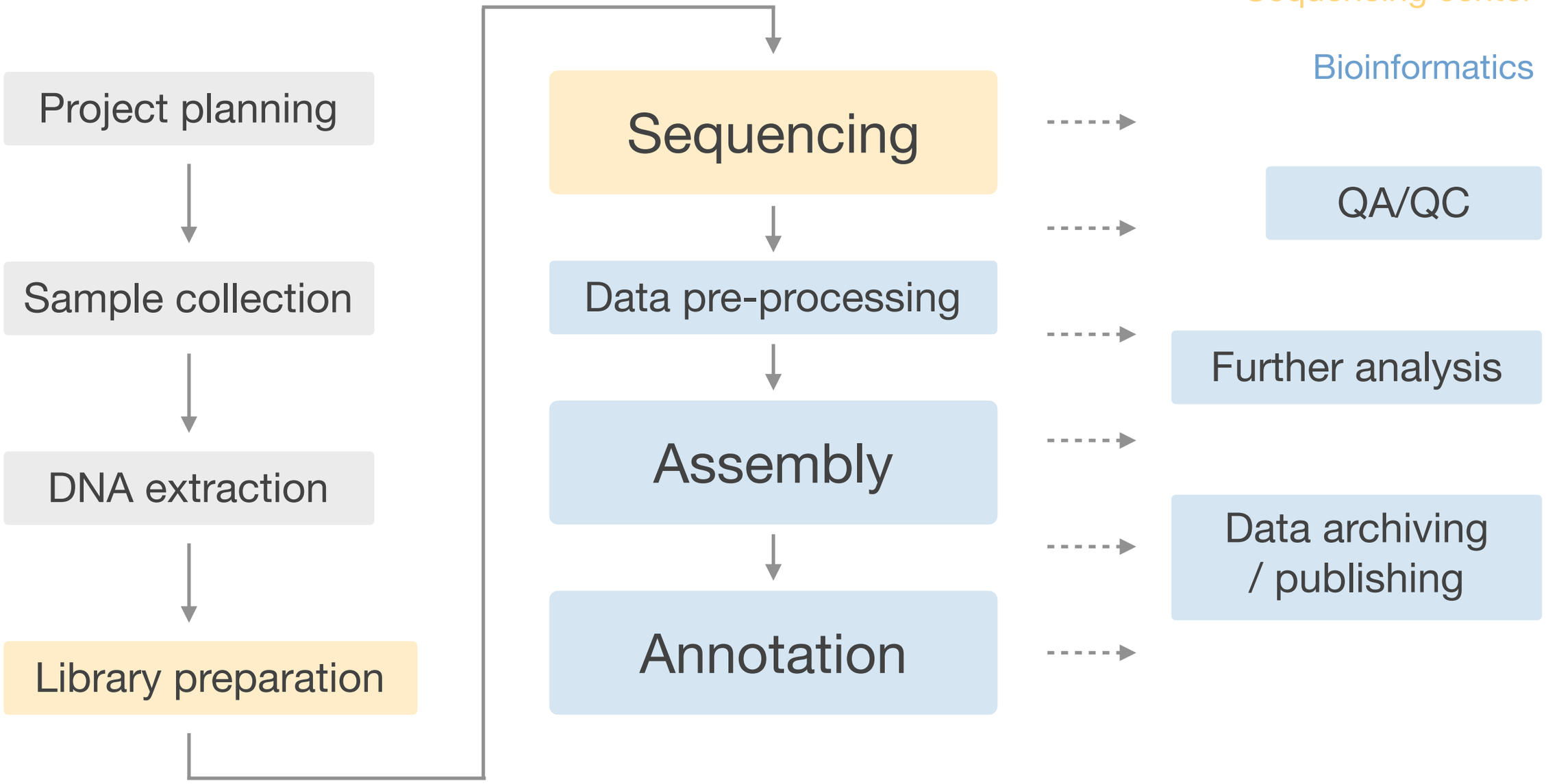
- https://datadryad.org
- Free open-access repository for data in any format

  (e.g. GFF, official gene set FASTA, genotype data in VCF format)
- May or may not be linked to publication
- Data receive permanent DOI

Code and workflow documentation

- Key to reproducibility
- GitHub / GitHub Pages
- Example: https://k-hench.github.io/hamlet_radiation/

De novo genome sequencing workflow



Thank you!

Please leave feedback for this course here (Google form):

https://tinyurl.com/9hucxfvh

Questions?

Email me at martin.helmkampf@leibniz-zmt.de