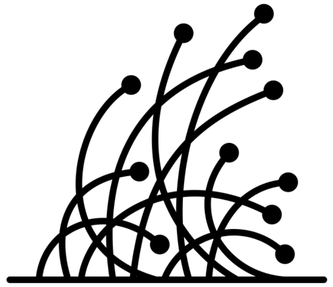


Command line and mapping

Using the Terminal

Dr Johanna Rhodes

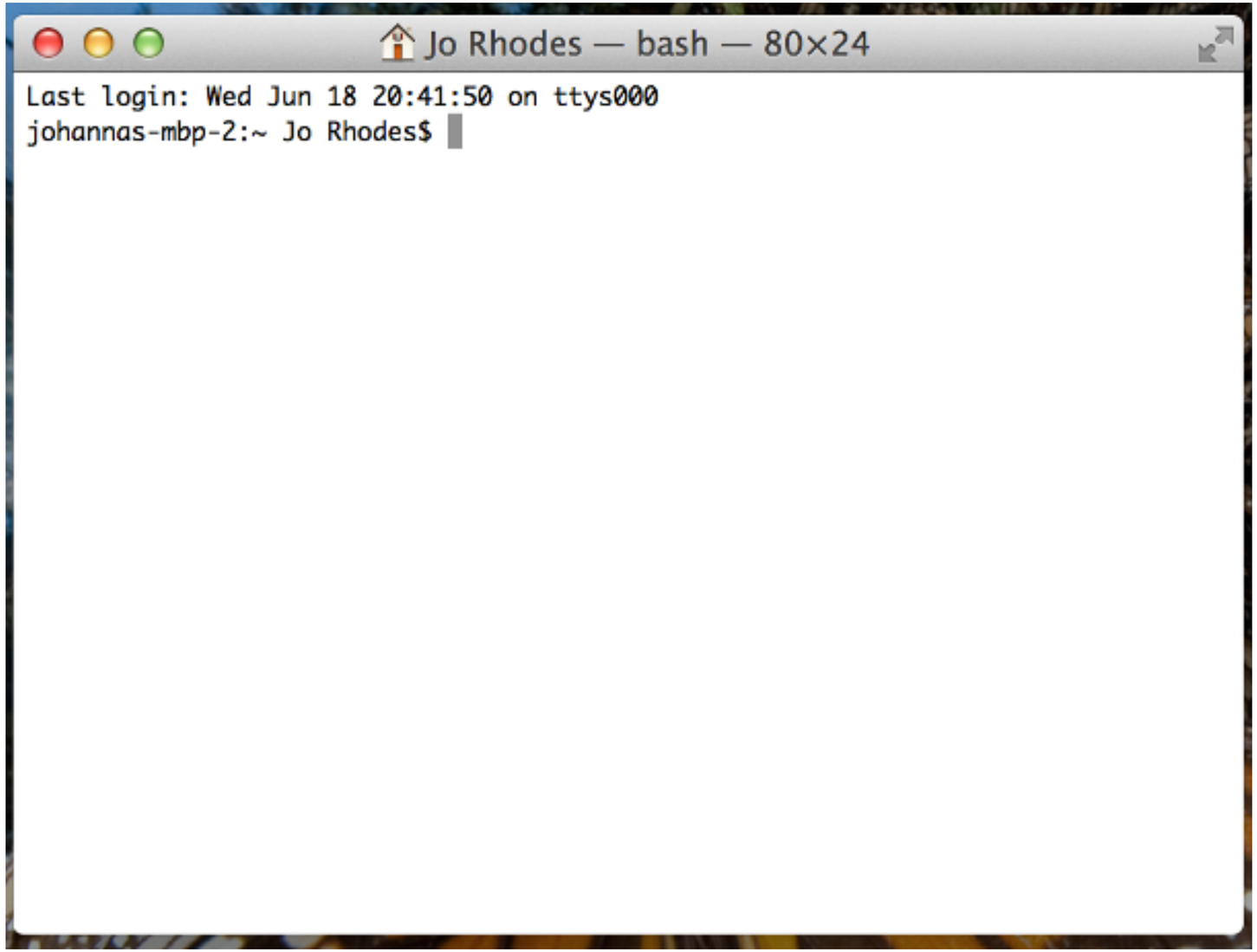
Learning outcomes

- **Understand how Terminal works**
  - Learn basic commands for navigating Terminal
  - Learn basic commands for working with NGS data
- **Sequence data formats**
  - FASTA, FASTQ, md5
  - PHRED encoding for quality
- **Mapping short read sequence data**
  - Quality control
  - Tools
  - Advantages and disadvantages to *de novo* assembly
- **Reference genomes**

Command line basics

Terminal

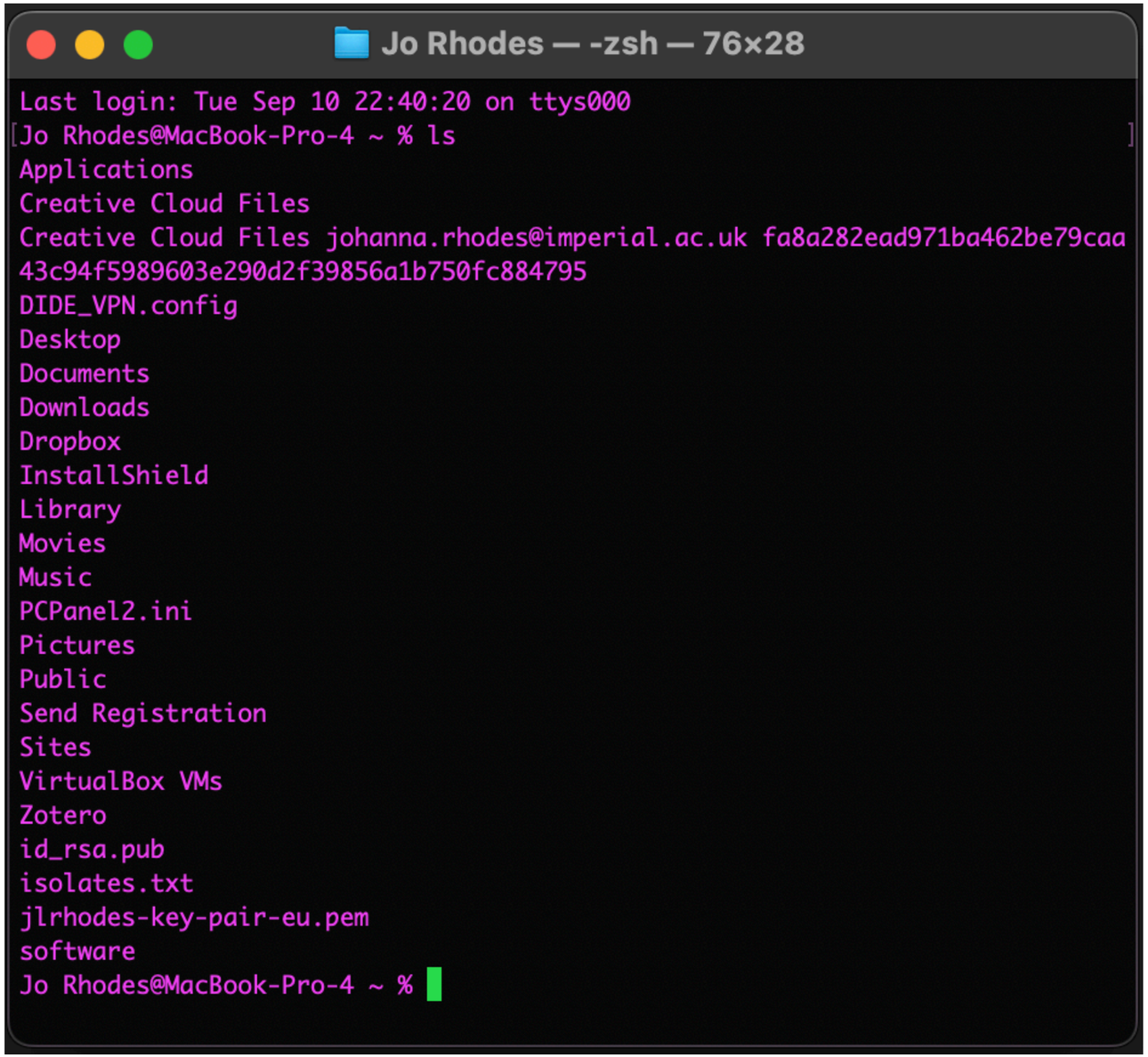
Command-line interface (CLI)



Terminal

ls

Writes standard output of content of directory



ls

```
Jo Rhodes@MacBook-Pro-4 Wadworth % ls -lh
total 0
drwxr-xr-x  6 Jo Rhodes  staff   192B 18 Jun 23:34 Nanopore
drwxr-xr-x  5 Jo Rhodes  staff   160B 10 Sep 21:54 RNAseq
Jo Rhodes@MacBook-Pro-4 Wadworth %
```

Options include -l or -lh for more information

Terminal

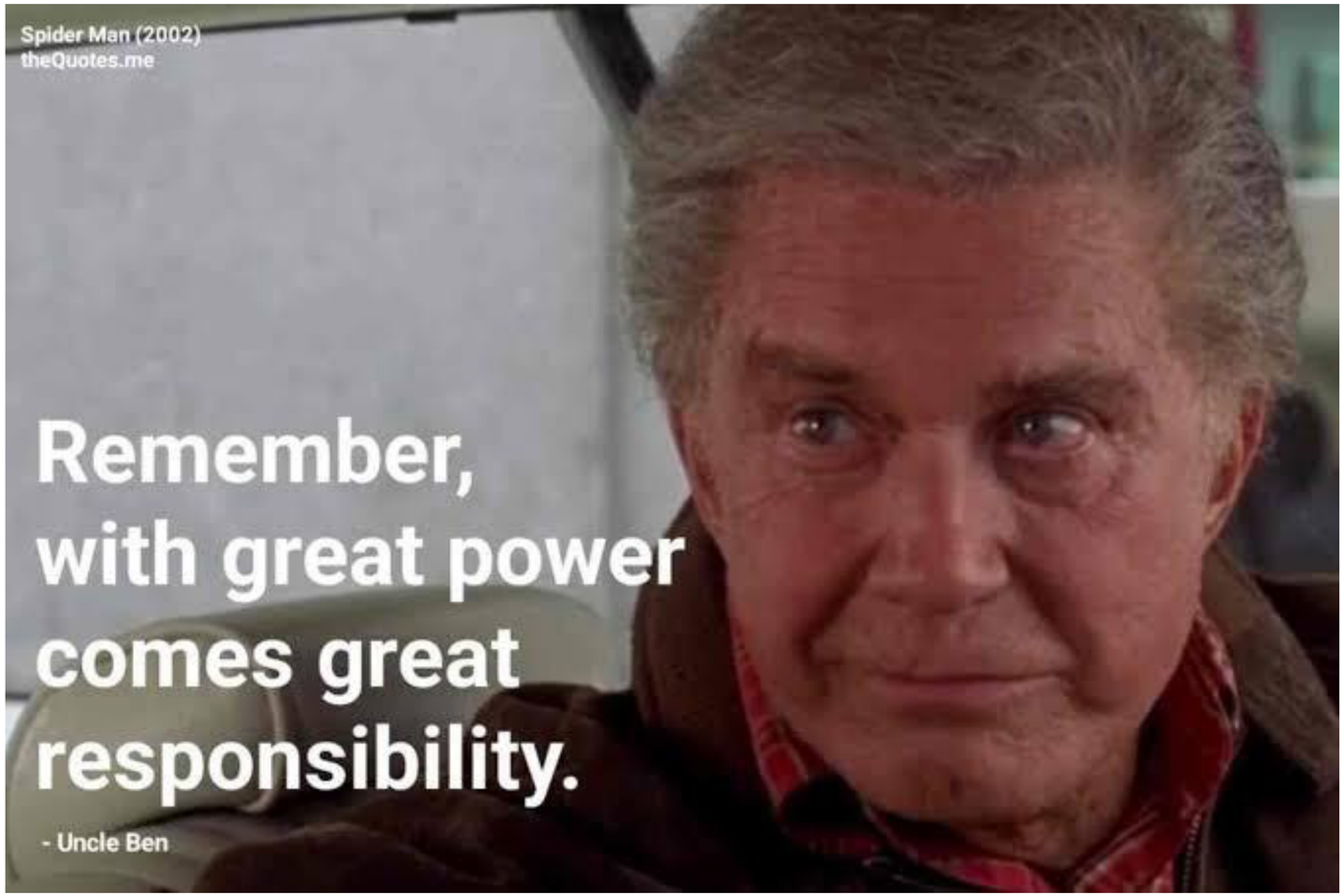
Commands

- **cd**
  - Change directory
- **mkdir namefolder**
  - Create a new directory called ‘namefolder’
- **rmdir namefolder**
  - Remove folder (as long as it’s empty)
- **pwd**
  - Print current directory and path

Terminal

Commands

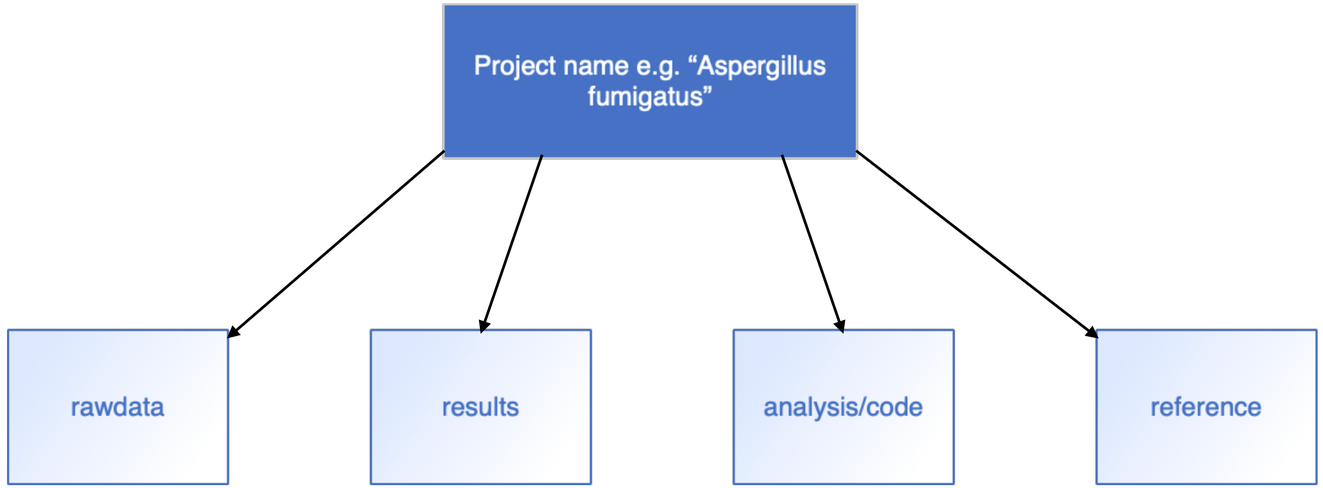
- Case sensitive!
  - ‘pwd’ not the same as ‘PWD’ etc



Terminal

Project directory structure

- Personal preference!
- But good to keep organised
- Each project can have a ‘readme’ file containing details of start/end of project, collborators, funders, basics of project

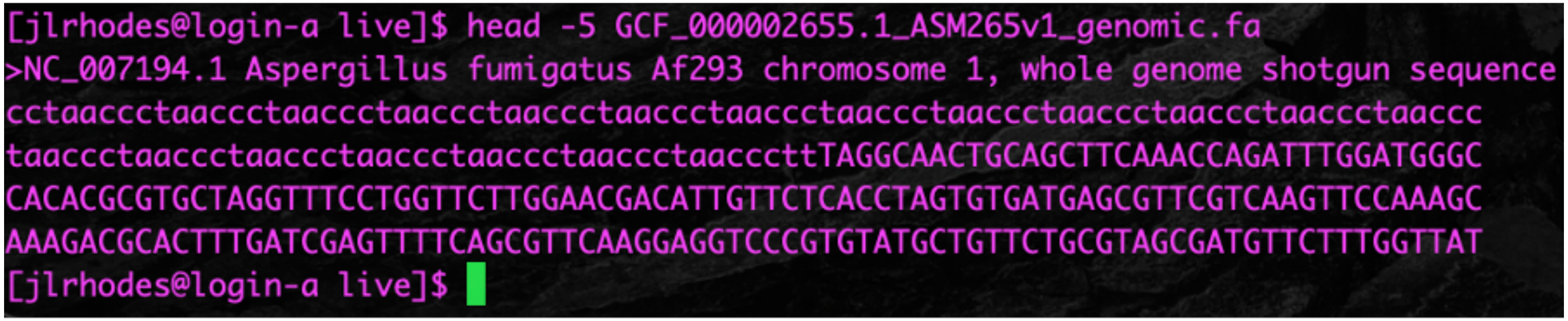


Data formats

FASTA format

Nucleotide or AA sequences

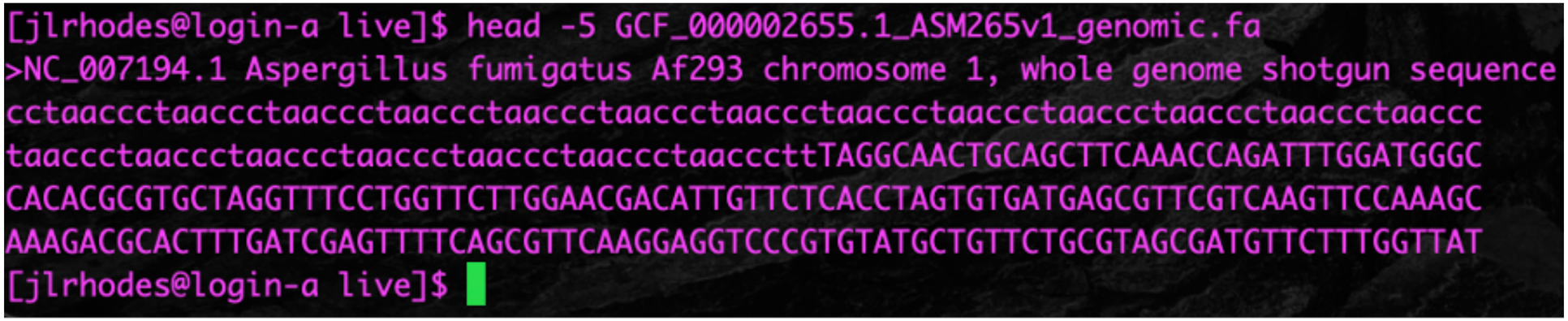
- Always starts with “>” followed by an identifier. The rest of the line (a description) is optional.
- Nucleotide or animo acid sequence follows on subsequent lines
- This format is used for reference genomes, and reference-based alignment workflows



FASTA format

References

- Reference genomes can be found on GenBank, and new versions are released periodically.
- Reference genomes used for alignment workflows must be indexed for use by three separate pieces of software: BWA, Samtools and GATK
  - indexing allows the aligner software to narrow down the potential origin of a query sequence within the genome, saving time and computational memory



FASTA format

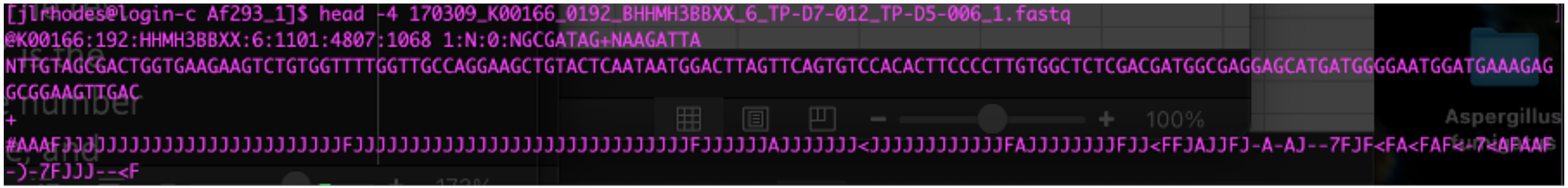
Indexing a reference genome for alignment

- Index using BWA
  - `bwa index reference.fa`
  - Type ‘ls -lh’ to see the files created
- Index using Samtools
  - `samtools faidx reference.fa`
  - Creates a file with extension ‘.fai’, which contains one record per line for each of the costings (contiguous DNA segments) in the fasta file
- Index for GATK using Picard
  - Creates a ‘dictionary’ (.dict file) using Picard in order to call SNPs using GATK. This .dict file describes the contents of your reference fasta file
  - `picard CreateSequenceDictionary -R reference.fa -O reference.dict`

FASTQ format

Raw sequence data

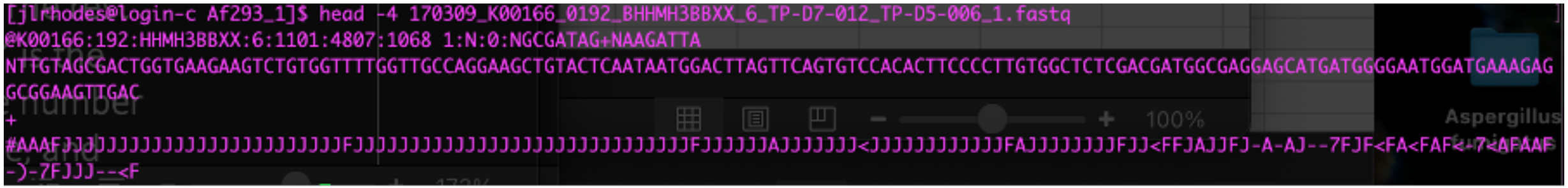
- FASTQ files are compressed with ‘gzip’, and have the .gz extension to save space. We use ‘gunzip’ to decompress and look at the contents, but all software will use compressed fastq files, so there is no need to decompress



FASTQ format

Raw sequence data

- FASTQ files contain the sequence data and associated quality data over four lines
- Line 1: here, ‘K00166’ is the name of the sequencing machine used. The rest of the data on this line is optional.
  - This line is in ‘Casava 1.8’ format, which is used in Illumina 1.9 and beyond.
  - ‘192’ is the run ID, HHMH3BBXX is the flow cell ID; ‘6’ is the flow cell lane; ‘1101’ is the tile number within the flow cell lane; 4807 is the ‘x’-coodinate of the cluster within the tile, and 1068 is the ‘y’-coorindate
  - After the space, the ‘1’ means paired end data (it would be ‘2’ for mate pair), ’N’ means reads are not filtered, and the letters at the end are the index sequence
- Line 2 is the raw sequence
- Line 3 begins with ‘+’ and can contain sequence identifiers, but this is optional
- Line 4 is the associated quality data for sequence in line 2 - note there is one quality score per nucleotide

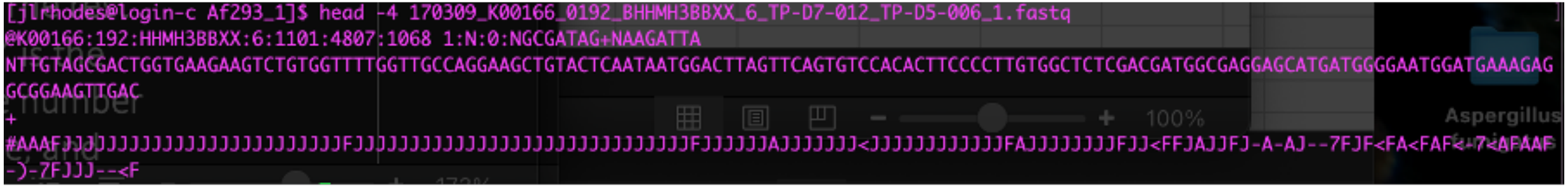


FASTQ format

Quality scores

- Line 4 is the associated quality data for sequence in line 2 - note there is one quality score per nucleotide
- These quality scores are what's known as 'PHRED+33' encoding
  - ! represents the lowest quality
  - J represents the highest quality

```
!"#$%&'()*+,./0123456789:;<=>?@ABCDEFGHIJ
```



ONT long read data format

fast5 format

- Native container for data from ONT platforms, containing raw electrical signal levels measured by the nanopores
  - “squiggle” signals
  - hdf5-based
- Basecalling on fast5 files to acquire FASTQ
  - Do not have sufficient information for methylation calling, so important to keep raw signals
- No formal specification for fast5 format

PacBio long read data format

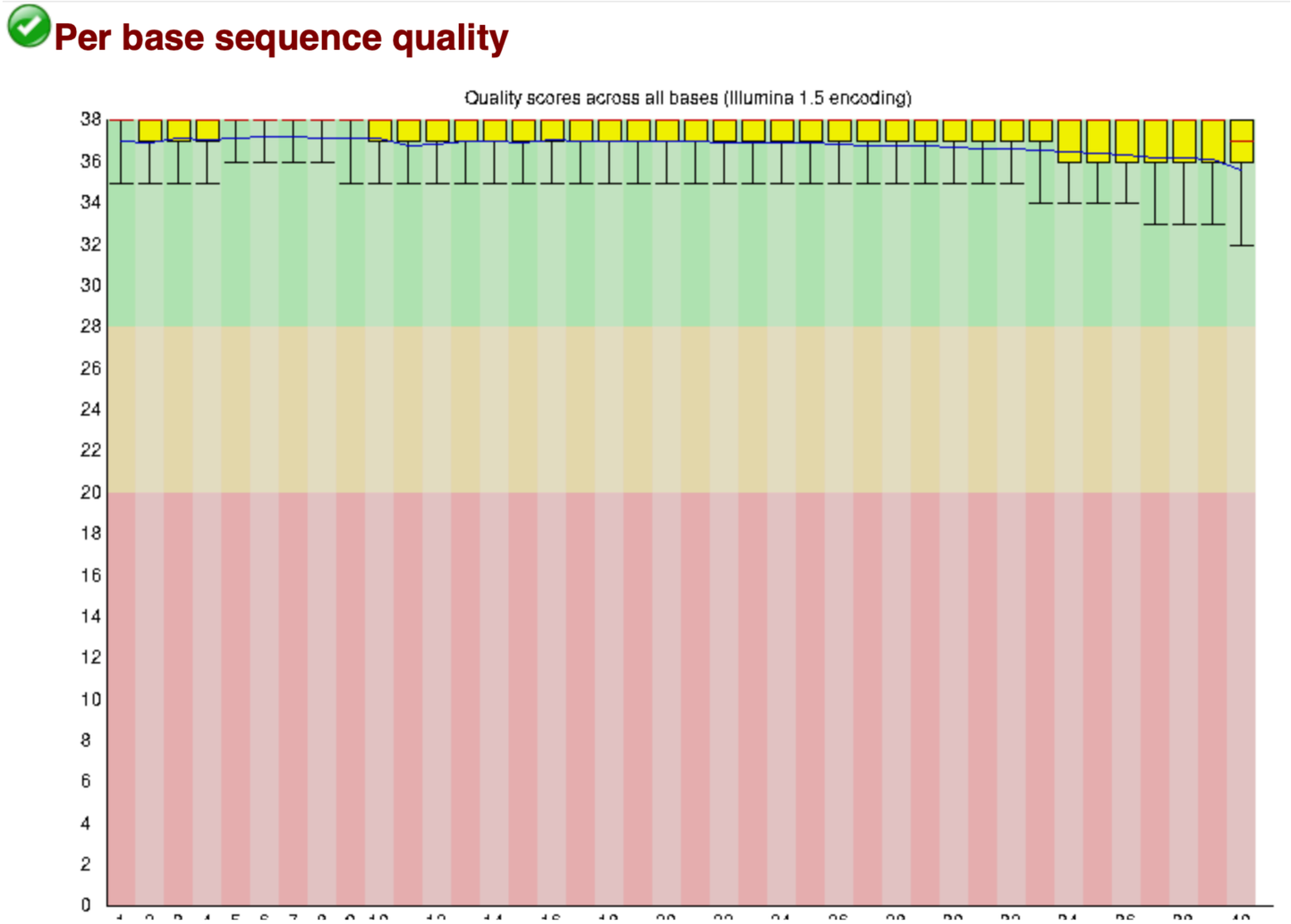
‘movie’ files

- Raw data from sequel platform comes as BAM, FASTA and FASTQ
- FASTQ same as Illumina files, except that each read will be very long
- HiFi reads contain PHRED scores and use a similar encoding to Illuminate 1.8+

Mapping short read data

Quality control

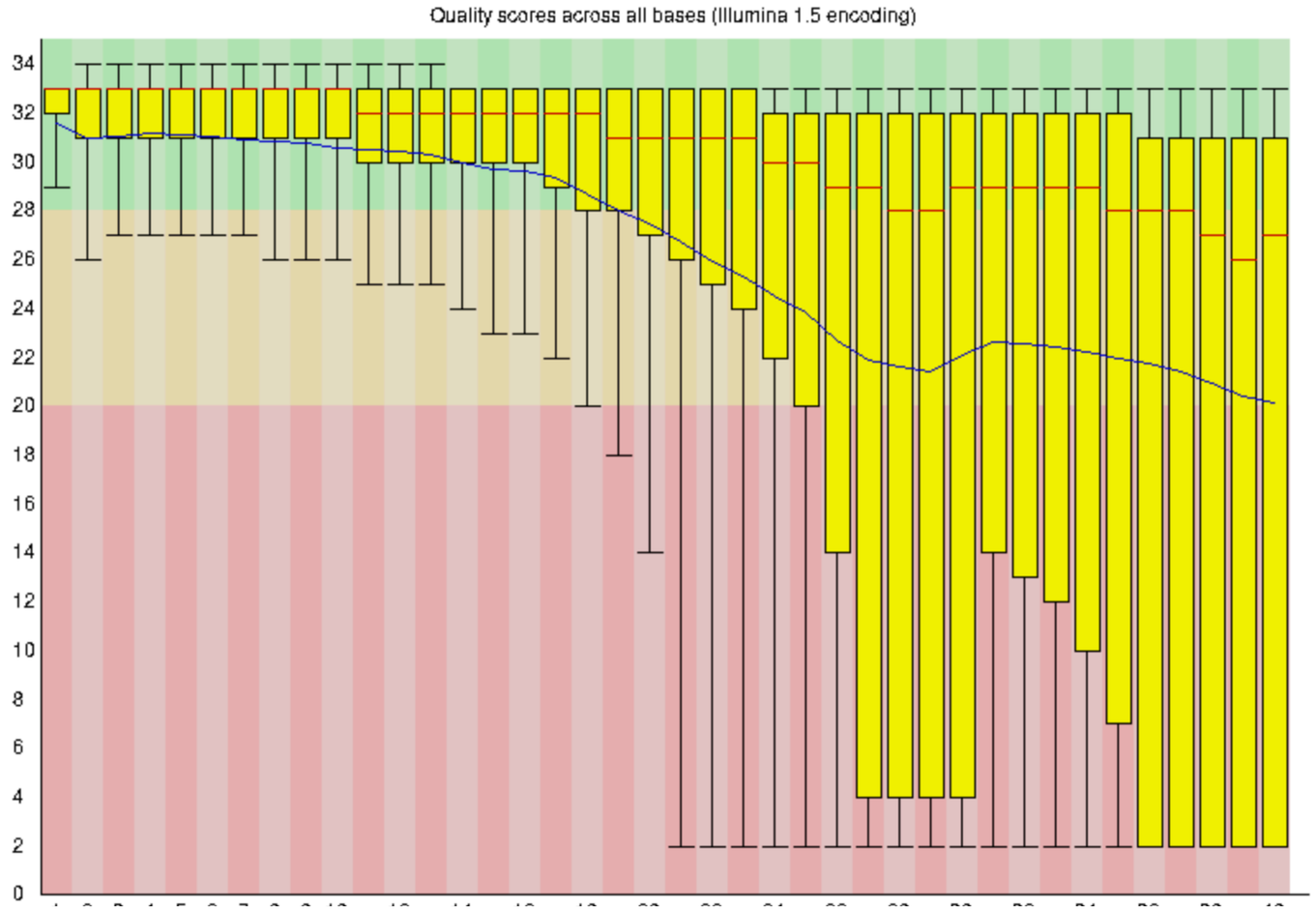
FASTQ reads with FastQC (Babraham Institute)



Quality control

FASTQ reads with FastQC (Babraham Institute)

Per base sequence quality



Quality control

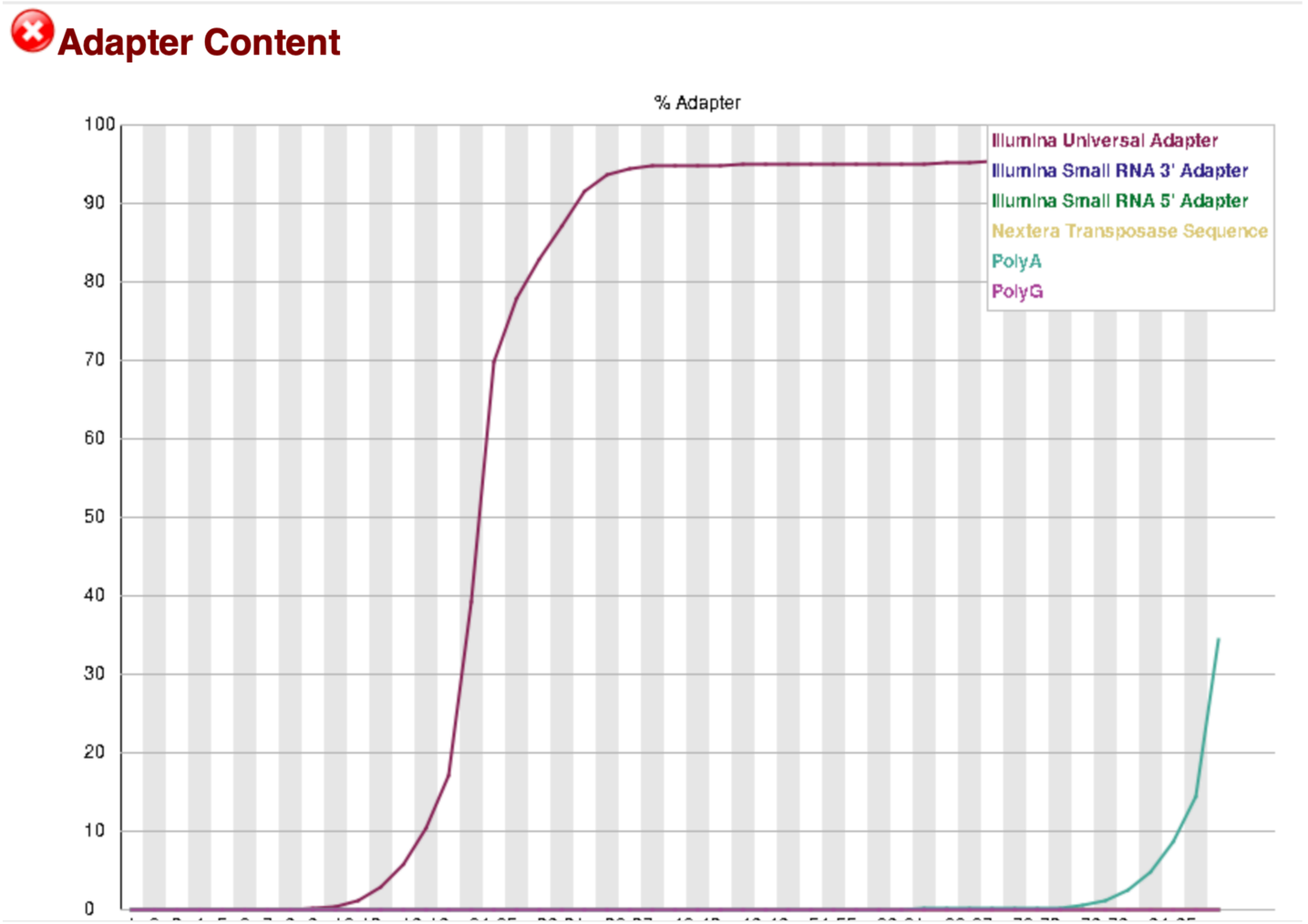
FASTQ reads with FastQC (Babraham Institute) - adapter dimer contamination

Overrepresented sequences

Sequence	Count	Percentage	Possible Source
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCGATCT	8122	8.122	Illumina Paired End PCR Primer 2 (100% over 40bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGATCGGAAG	5086	5.086	Illumina Paired End PCR Primer 2 (97% over 36bp)
AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTAC	1085	1.085	Illumina Single End PCR Primer 1 (100% over 40bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCGGAAG	508	0.508	Illumina Paired End Sequencing Primer 2 (100% over 36bp)
AATTATACGGCGACCACCGAGATCTACACTCTTTCCCTAC	242	0.242	Illumina Single End PCR Primer 1 (97% over 40bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAAGATCGGAA	235	0.23500000000000001	Illumina Paired End Adapter 2 (96% over 31bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCGAGATCGGAAGA	228	0.22799999999999998	Illumina Paired End Adapter 2 (96% over 28bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCGGACG	205	0.20500000000000002	Illumina Paired End Sequencing Primer 2 (100% over 36bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGGTCGGAAG	183	0.183	Illumina Paired End Adapter 2 (100% over 32bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGGATCGGAA	183	0.183	Illumina Paired End Adapter 2 (100% over 32bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCGAACT	164	0.164	Illumina Paired End PCR Primer 2 (97% over 40bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCGGTCT	129	0.129	Illumina Paired End PCR Primer 2 (97% over 40bp)
AATTATACTTCTACCACCTATATCTACACTCTTTCCCTAC	123	0.123	No Hit
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCGGACT	122	0.122	Illumina Paired End Sequencing Primer 2 (100% over 36bp)
CGGTTCAGCAGGAATGCCGAGATCGGAAGAGCGGTTCAGC	113	0.11299999999999999	Illumina Paired End PCR Primer 2 (96% over 25bp)

Quality control

FASTQ reads with FastQC (Babraham Institute) - small RNA with read-through adaptor



Quality control

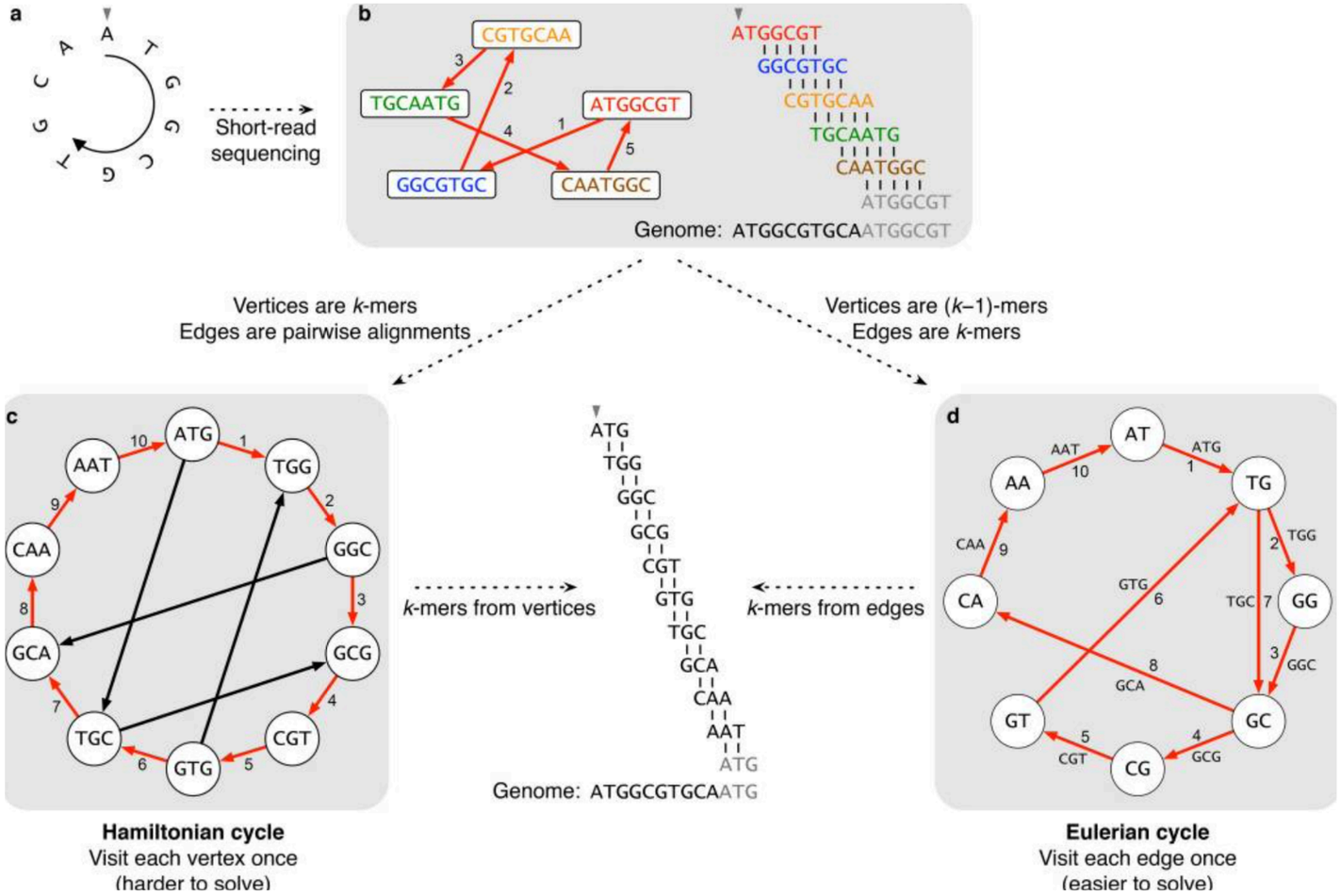
Adaptor trimming reads

- Tools such as ‘cutadapt’ and ‘trimmomatic’ can be used to remove adaptor and/or overrepresented sequences
    - Trimmomatic will also cut low quality sequences

De novo assembly

No reference required

- Converts reads into a detailed set of sequences corresponding to the chromosome of an organism
- Needed when sequencing an organism for the first time
- Longer reads better to resolve repetitive regions
  - SPAdes is a short read assembler
    - Capable of hybrid assembly
  - Canu uses ONT long reads
  - Hifiasm uses PacBio HiFi long reads



Compeau, P.E., *et al. Nat Biotechnol* **29(11)**, 987-991 (2011)

Assembly vs. alignment

Pros and cons

- Pros:
  - Assembly - find structural variants and large copy number variation (CNV)
  - Alignment - good for small variants (indels, SNPs, CNVs)
- Cons:
  - Assembly - loose information about variants or depth of coverage; difficult to resolve repetitive regions without long reads
  - Alignment - errors/false positive variants if high levels of repetitive regions

Mapping reads to reference genome

Like a jigsaw puzzle

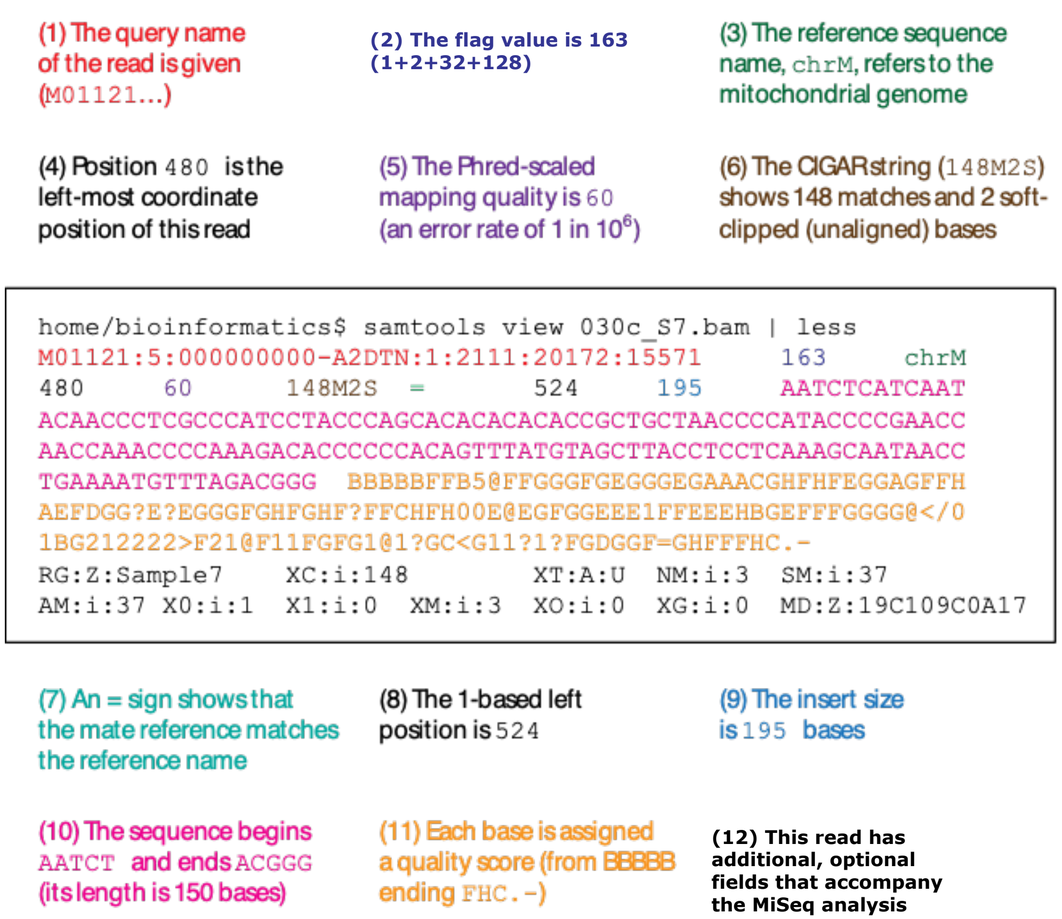
- Alignment is generally quicker and computationally less expensive than assembly, as we have a guide (the reference)
- Burrows-Wheeler algorithm (BWA) reconstructs the short read data in our FASTQ files using the reference genome as a guide
  - BWA ‘mem’ algorithm is the fastest and most accurate in the BWA package, and can handle short reads up to 250 bp.

```
bwa mem –M reference.fa R1.fastq.gz R2.fastq.gz > isolate_name.sam
```

The SAM file format

Sequence Alignment/Map

- Binary version = BAM
  - convert using samtools: `samtools view -bS isolate_name.sam > isolate_name.bam`
- Basic alignment!
  - PCR duplication not considered, no co-ordinate sorting, no indel consideration



Flag Value	Meaning	Flag Sum
1	read is paired	1
32	read2 was reverse complemented	33
64	read1	97
2048	Supplementary alignment	2145

Improving the alignment

Co-ordinate sort

- `samtools sort isolate_name.bam -o isolate_name.sorted.bam`
- Syntax dependent on version!
- All BAM files need to be indexed, which creates a new file allowing faster look-up of data
  - `samtools index isolate_name.sorted.bam`
  - This creates a file with suffix ‘bam.bai’

Improving the alignment
Fix the read groups

- We use ‘AddOrReplaceReadGroups’ from Picard to ‘fix’ the BAM file.
    - Downstream analyses (e.g. GATK) require the BAM file to contain read group information:
        - Read group identifier - this is the sequencer flow cell, lane number and name.
        - Read group platform - e.g. Illumina
        - Read group library - this is needed for marking duplicates to determine which read groups contain molecular duplicates
        - Read group platform unit - this is the run ‘barcode’ or the adaptor sequence
- The command may look a bit like this:

```
picard AddOrReplaceReadGroups -I isolate_name.sorted.bam -O isolate_name.fixed.sorted.bam --SORT_ORDER coordinate
--RGID K00166 --RGLB dnaseq --RGPL illumina --RGSM ’WGS’ --CREATE_INDEX TRUE --RGPU unknown --VALIDATION_STRINGENCY
SILENT
```

Improving the alignment

Marking duplicates

- Use Picard to mark any duplicated reads due to sequencing errors to prevent them being included in variant calling

```
picard MarkDuplicates -I isolate_name.fixed.sorted.bam -O
isolate_name.sorted.marked.bam --CREATE_INDEX TRUE --METRICS_FILE
picard_info.txt --REMOVE_DUPLICATES false --ASSUME_SORTED true --
VALIDATION_STRINGENCY SILENT
```

- This is still a rough ‘global’ alignment
- The rest of the pipeline is entirely GATK for improving the alignment further and variant calling

Improving the alignment
Base Quality Score Recalibration

- BQSR is a pre-processing step that detects systematic errors made by the sequencing machine when it estimates the accuracy of each base call.
    - BQSR adjusts locally according to the quality of each base call
    - Variant callers rely heavily on the quality score assigned to individual bases
- Two passes required
    - First pass generates a recalibration table using a ‘known sites’ vcf of SNPs and indels (these are not gold standard!) using the ‘BaseRecalibrator’ tool
    - Second pass uses ‘ApplyBQSR’ to adjust the base quality scores based on the model, producing the final BAM file
- Recommended to build two models i.e. run this twice for quality control

Improving the alignment

Base Quality Score Recalibration - commands

- First we need to provide a rough guide of variants (SNPs and indels) for BQSR. HaplotypeCaller (GATK) calls variants, and by default assumes a diploid organism so we need to specify ploidy is 1 (haploid)

```
gatk HaplotypeCaller –R reference.fa –I isolate_name.sorted.marked.bam –ploidy 1 –O
isolate_name.raw_variants.vcf
```

- This file contains both SNPs and indels, and can be used for BQSR:

```
gatk BaseRecalibrator –R reference.fa –I isolate_name.sorted.marked.bam –-known-sites
isolate_name.raw_variants.vcf –O isolate_name.recal_data.table
gatk ApplyBQSR –R reference.fa –I isolate_name.sorted.marked.bam –-bqsr-recal-file
isolate_name.recal_data.table –O isolate_name.recal_reads.bam
```

- Do this again (be careful with file names!) to acquire final BAM file with your final alignment

Is the alignment any good?

Quality control

- Two basic ways to check the quality of your alignment at this stage:
  - Coverage
  - Mapping statistics
- Coverage: how many reads map to a certain area
  - Useful to see as a good alignment has an even coverage across the whole genome in order to accurately/confidently call variants.
  - Depth of coverage can be used to observe copy number variation (CNV) or ploidy events

```
picard CollectWgsMetrics -R reference.fa -I isolate_name.post_recal_reads.bam -O
isolate_name.metrics.txt
```

```
samtools depth isolate_name.post_recal_reads.bam > isolate_name.coverage.txt
```

Is the alignment any good?

Coverage

- Samtools command gives you per base coverage (useful for CNV detection)
- Picard command gives a summary
  - Average coverage
  - Number of positions with zero coverage

Is the alignment any good?

Mapping stats

- Samtools
  - ‘flagstat’ command
  - Want:
    - a high percentage of mapped reads (>95%)
    - a high percentage of properly paired (if paired end reads)
    - low percentage of singletons

```
7417232 + 0 in total (QC-passed reads + QC-failed reads)
287618 + 0 duplicates
4534962 + 0 mapped (61.14%:-nan%)
7417232 + 0 paired in sequencing
3708616 + 0 read1
3708616 + 0 read2
4528278 + 0 properly paired (61.05%:-nan%)
4534962 + 0 with itself and mate mapped
```

Reference genomes

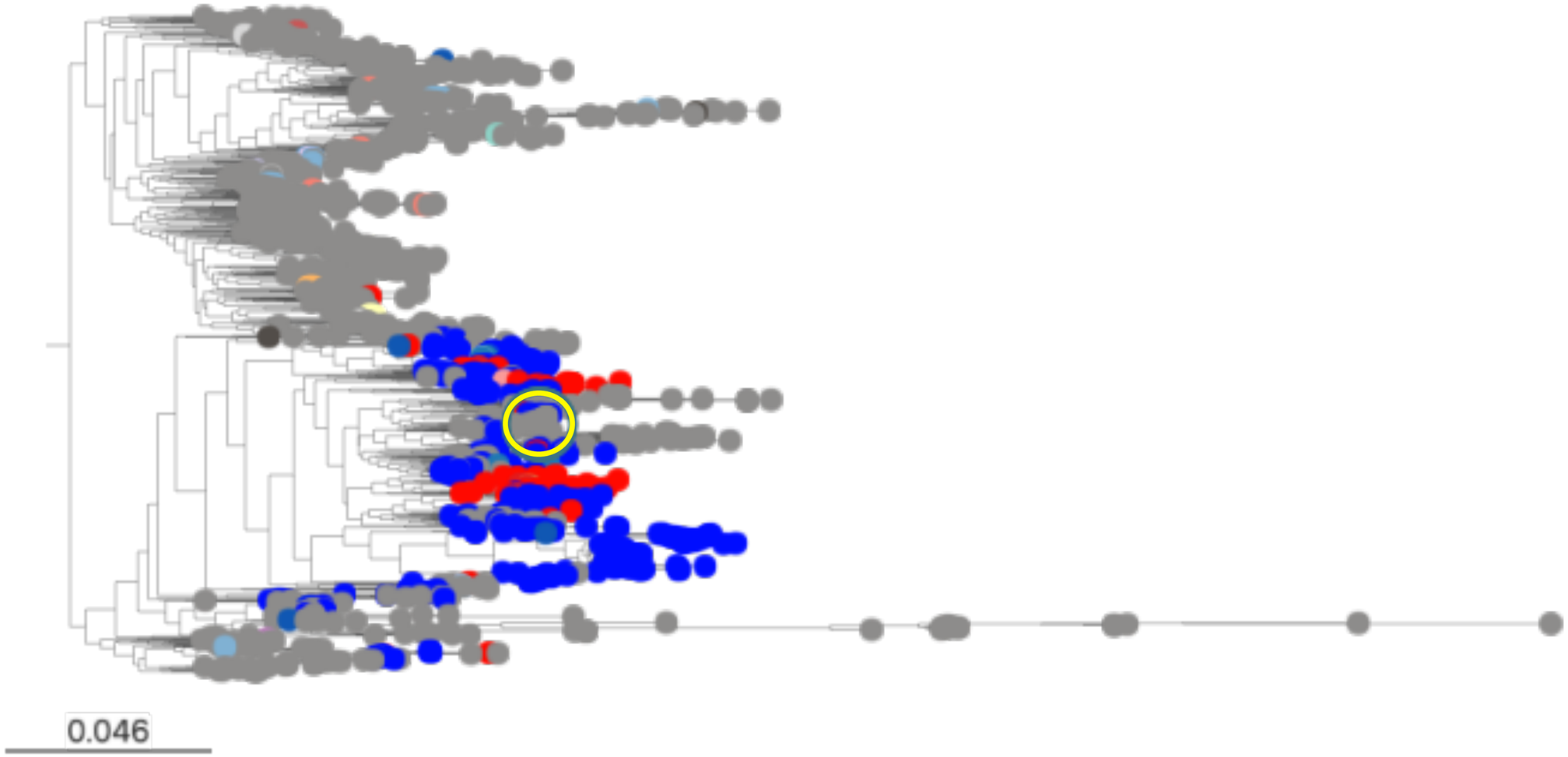
Reference genomes

What makes a good reference genome?

- Reference genomes just tend to be the first whole genome sequence project of that particular species
  - Whole genome sequence available, annotation, lab-based tools
  - As we learn more about the biology of the organism we learn that the reference isolate may be biologically a bit odd….

Odd reference fungi

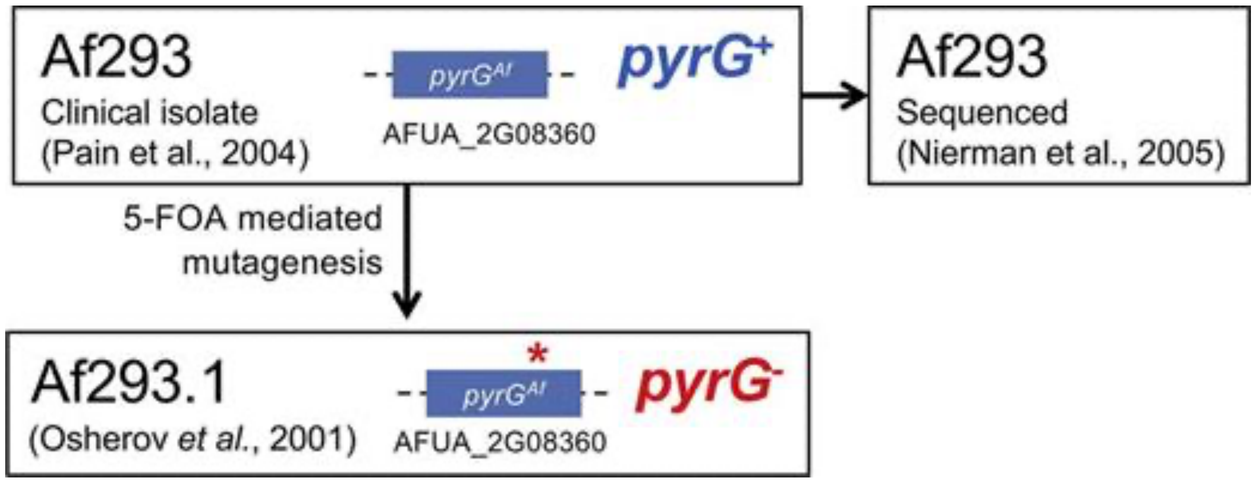
Aspergillus fumigatus



Odd reference fungi

Aspergillus fumigatus

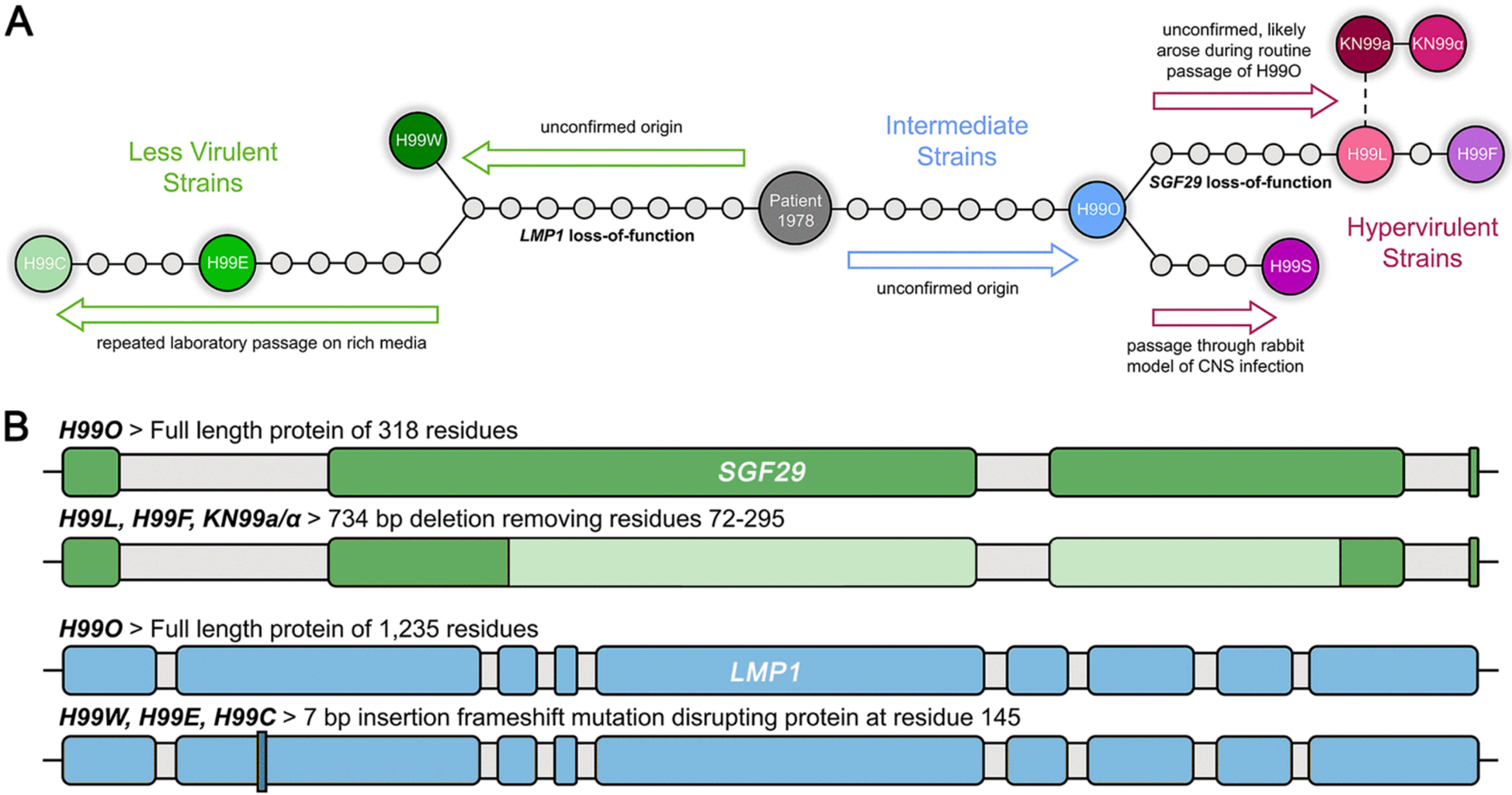
- Significant interstrain variability observed with respect to survival during *in vivo* infection
  - Due to multiple passaging
  - Different labs have different version of the ‘same’ strain



Bertuzzi *et al.* (2021) Medical Mycology

Odd reference fungi

Cryptococcus neoformans



Fernandes *et al.* (2022) mBio

Reference genomes

Are they any good?

- Highly contiguous sequences = good
  - telomere-to-telomere is gold standard
- Are annotations available?
- What is good now might be better in a few years
  - improving sequence technology
  - improved understanding

Reference genomes

RefSeq and GenBank

- New reference genomes are released periodically
  - e.g. *A. fumigatus* Af293 and CEA10
    - recent updates - Af293: ASM265v1 (GCF_000002655.1) and CEA10: ASM2422042v1 (GCA_024220425.1)
      - new technology leads to better sequencing and finding large chunks of sequence previously missed