

# Introduction to Oxford Nanopore Technologies (ONT) sequencing bioinformatics pipeline

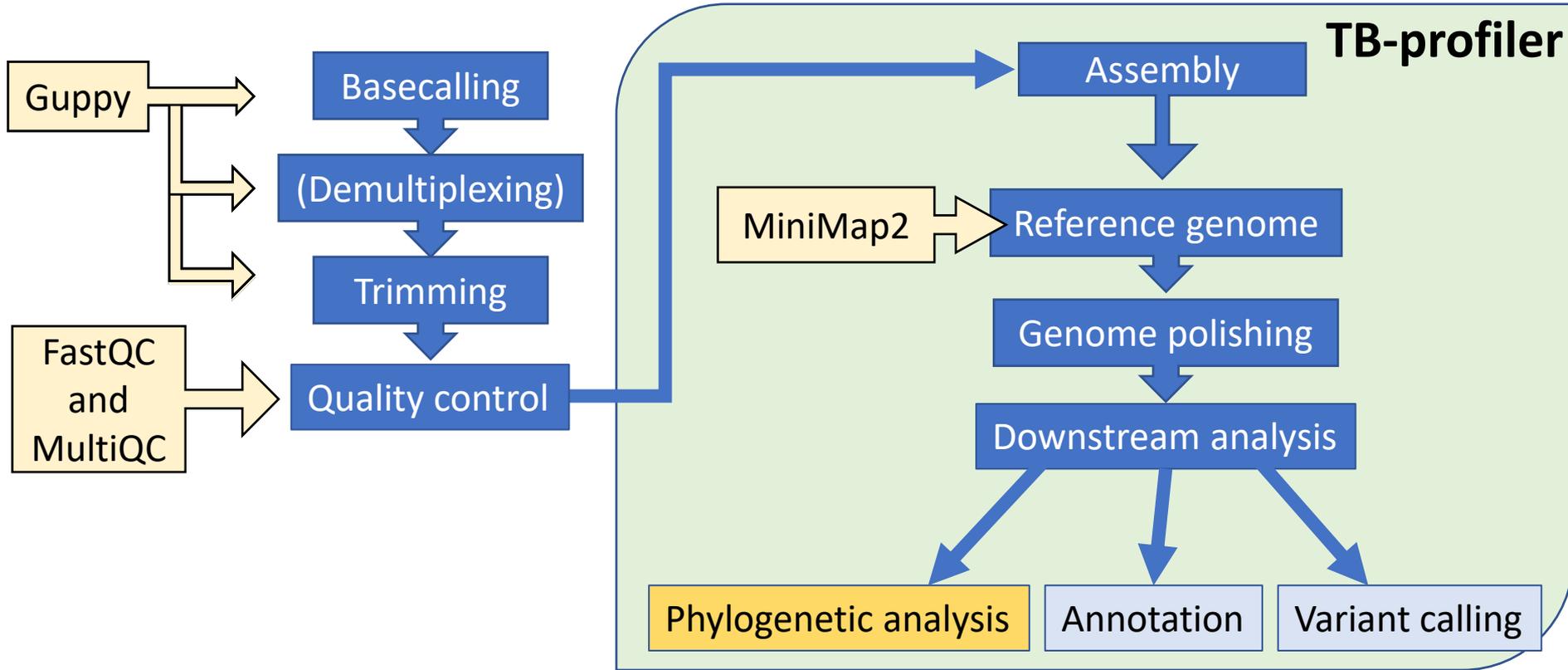
Dr Linzy Elton, Prof Neil Stoker, Dr Sylvia Rofael

**Let's Download the data (5 fast5 files)**

# Before you start

- What operating system are you using? Be aware that most bioinformatics tools only run on Linux and MacOs
- For Windows 10 and above users they can use Windows Sub Linux System (WSL)
- How powerful is your computer? Some programmes require a minimum amount of memory (RAM  $\geq$ 16 GB), Processor (how many CPUs), Disk Space
- Power – processing huge files can take a long time (days!) so you will need to make sure you have a continuous power supply throughout, otherwise you will have to start again
- Internet– CLI's don't generally need internet to run, but if you are uploading to cloud based server, downloading and installing bioinformatics tools you need a good connection

# ONT MTB Bioinformatics Pipeline



Data Analysis

```

1404619      www      idle    4    2:00:00  Tue Nov 28 18:18:19
1404620      www      idle    4    2:00:00  Tue Nov 28 18:18:19
1404621      www      idle    4    2:00:00  Tue Nov 28 18:18:19
1404622      www      idle    4    2:00:00  Tue Nov 28 18:18:19
1404623      www      idle    4    2:00:00  Tue Nov 28 18:18:19
1404624      www      idle    4    2:00:00  Tue Nov 28 18:18:19
1404625      www      idle    4    2:00:00  Tue Nov 28 18:12:73
1404626      www      idle    4    2:00:00  Tue Nov 28 18:12:56
1404627      www      idle    4    2:00:00  Tue Nov 28 18:13:05
1404628      www      idle    4    2:00:00  Tue Nov 28 18:13:27
Mnab.1615G10 atireza BatchHold 10 1:00:43:20 Mon Nov 19 17:31:44

688 blocked jobs
Total jobs: 911

service2[ram] /home/pamfr/cbr/projects/cgp/pamfr/ram
Directory: /home/pamfr/cbr/projects/cgp/pamfr/ram
service2[ram] /home/pamfr/cbr/projects/cgp/pamfr/ram cdpg
Directory: /home/pamfr/cbr/projects/cgp
service2[ram] /home/pamfr/cbr/projects/cgp/cd data/cgp_private/ham_standBG1_2_batch_
data/cgp_private/ham_standBG1_2_batch_ No such file or directory
service2[ram] /home/pamfr/cbr/projects/cgp/cd data/cgp_private/ham_standBG1_2_batch_P1/
Directory: /home/pamfr/cbr/projects/cgp/data/cgp_private/ham_standBG1_2_batch_P1
service2[ram] /home/pamfr/cbr/projects/cgp/data/cgp_private/ham_standBG1_2_batch_P1
total: 488
dramr-crx-x 2 rsm cgp 4096 Oct 8 15:18 82012_01
dramr-crx-x 2 rsm cgp 4096 Oct 8 15:18 82012_02
dramr-crx-x 2 rsm cgp 4096 Oct 8 15:18 82012_03
dramr-crx-x 2 rsm cgp 4096 Oct 8 15:18 82012_04
dramr-crx-x 2 rsm cgp 4096 Oct 8 15:18 82012_05
dramr-crx-x 2 rsm cgp 4096 Oct 8 15:18 82012_06
dramr-crx-x 2 rsm cgp 4096 Oct 8 15:18 82012_07
dramr-crx-x 2 rsm cgp 4096 Oct 8 15:18 82012_08
rsm-crx-x-1 1 rsm ccom 11113 Nov 13 09:57 BG1_batch_P1_assembly_P1_B1a.c
rsm-crx-x-1 1 rsm ccom 11172 Nov 13 09:57 BG1_batch_P1_assembly_P1_B1b.c
dramr-crx-x 2 rsm ccom 4096 Nov 1 18:44 BG1_data
dramr-crx-x 2 rsm cgp 4096 Nov 1 18:18 assemblies
dramr-crx-x 187 rsm cgp 3017 Nov 1 11:38 assembly_map
dramr-crx-x 2 rsm cgp 4096 Oct 18 09:26 processed_reads
dramr-crx-x 5 rsm cgp 4096 Nov 1 11:38 obsv_cfr_01
dramr-crx-x 6 rsm cgp 4096 Nov 13 14:23 obsv_cfr_01a
dramr-crx-x 5 rsm cgp 4096 Nov 13 14:23 obsv_cfr_01b
dramr-crx-x 5 rsm cgp 4096 Nov 13 14:23 obsv_cfr_01c
dramr-crx-x 5 rsm cgp 4096 Nov 13 14:23 obsv_cfr_01d
dramr-crx-x 5 rsm cgp 4096 Nov 1 11:38 obsv_cfr_02
dramr-crx-x 5 rsm cgp 4096 Nov 1 11:38 obsv_cfr_03
dramr-crx-x 5 rsm cgp 4096 Nov 1 11:38 obsv_cfr_04
dramr-crx-x 4 rsm cgp 8152 Nov 1 13:41 raw_read_temp
service2[ram] /home/pamfr/cbr/projects/cgp/data/cgp_private/ham_standBG1_2_batch_P1
    
```

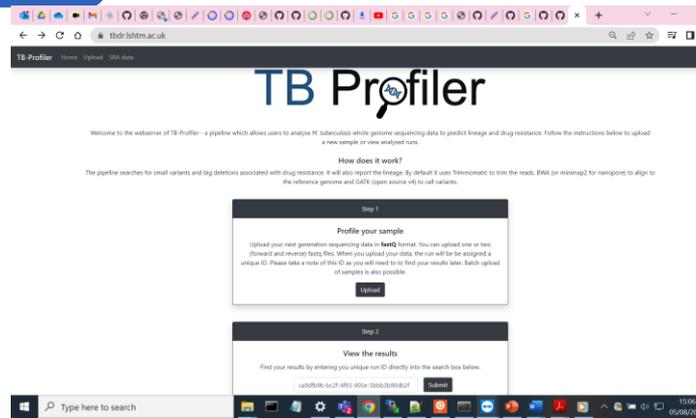
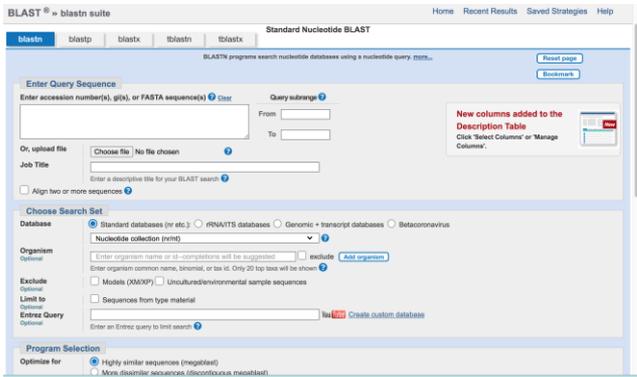
Bioinformatics tools

command line

Python

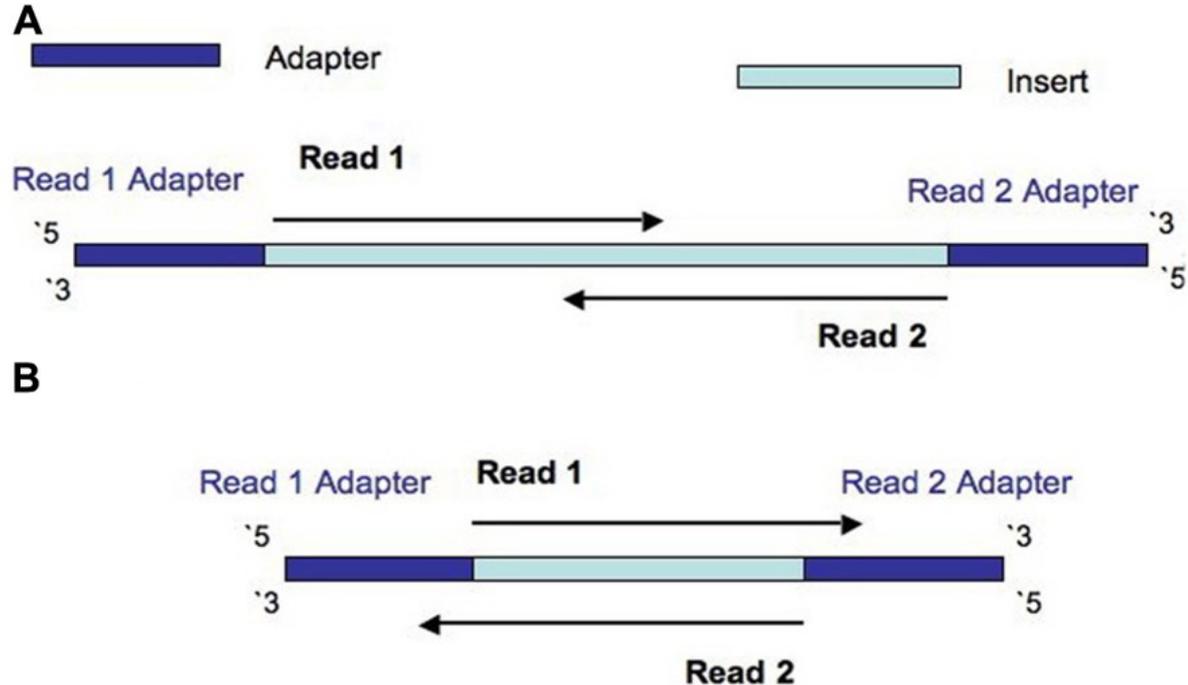
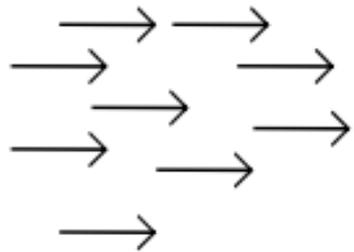
Web-based interface

user friendly  
e.g. NCBI BLAST tools

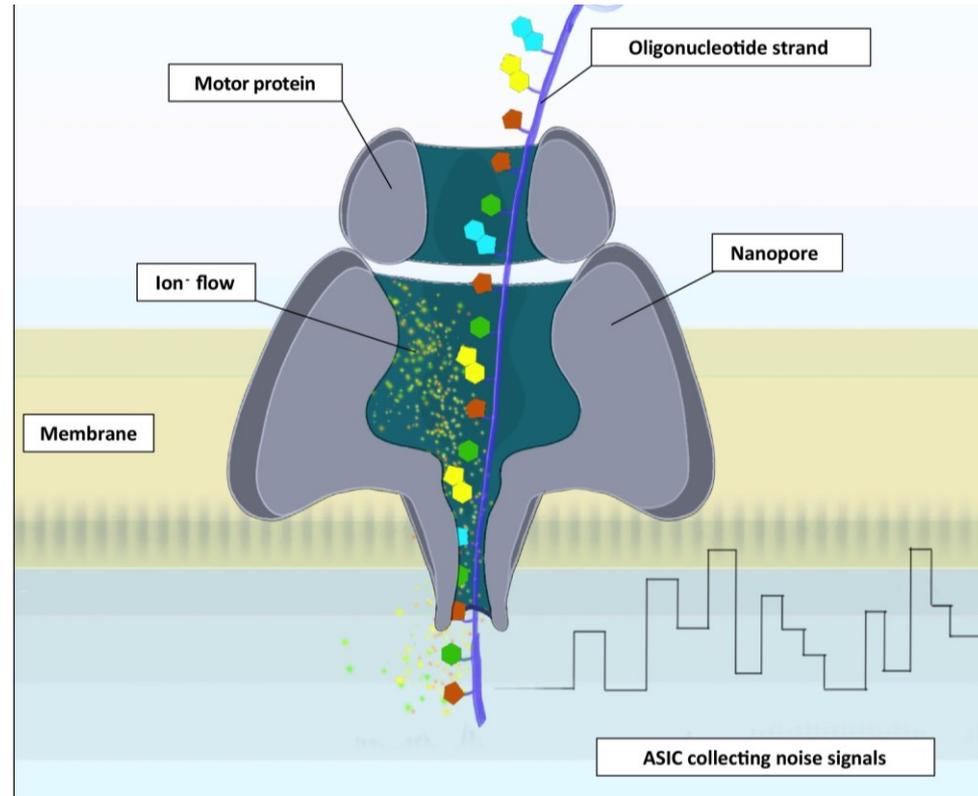


# Paired read

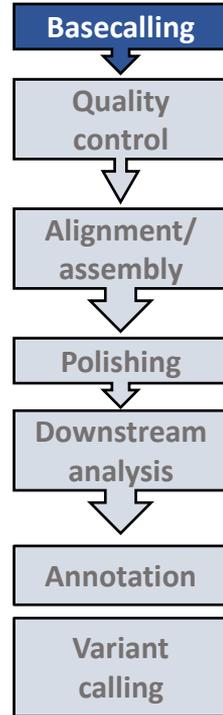
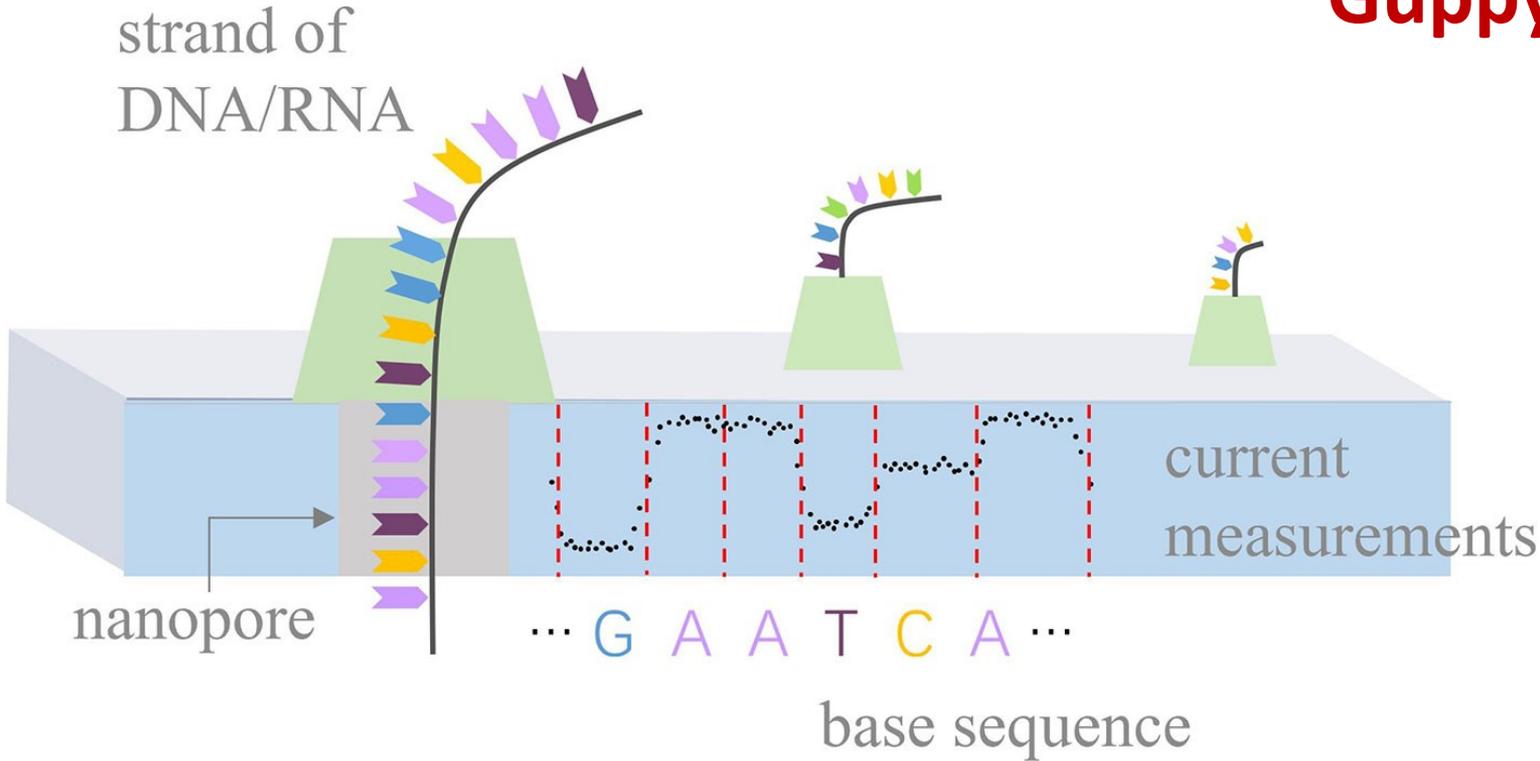
Single end reads



- Fast5 files are used by the MinKNOW instrument software to store the primary raw sequencing data (squiggles data, chromatogram peaks 'raw data' ) from ONT sequencing devices
- Guppy is a basecalling software that converts the primary raw data to Sequences bases data (A, G, T, C)



# Guppy



# Some Tips before we Start

- When you are writing the code script Never use Microsoft word (hidden characters)
- At least use simple text editor (e.g Notepad or TextEdit)
- Recommended install Notepad++ (Windows-Users) or (Xcode for MacOS)
- It's always useful to write down your code (and annotate with #) so you know what to do when you come back to it later
- [What is your working directory? Can you set it to your home ? And How to check what is your work directory?](#)

Use the cd and pwd command

```
cd ~
```

```
pwd
```

```
cd ..    #-> Go up one directory
```

```
mkdir ONT_workshop
```

You can copy and Paste commands from [“Script ONT.txt”](#) shared with you

# Some Tips before we Start

- In any command you will have to provide < Path > of the file / folder : This is the long address of the file or folder location with all the upper directories in the hierarchy separated by a
  - forward slash “/” in MacOS and Linux  
E.g. /Users/sylviarofael/miniconda3/share/tbprofiler/tbdb.fasta
  - Backward slash “\” in Windows  
E.g.  
“C:\Users\WSL\_shared\ONT\_workshop\MTB\_fastq\barcode01\\*.fastq”
    - \* is a wildcard that can replace text in the name
- When to use “ ” if the path has spaces in the file name this can break the Syntax so in this case we can put it between vertical quotation marks
- “.” means current directory

**Let's Install Guppy Please refer to Script\_ONT.txt (line 47)**

**Then Download the data (5 fast5 files)**

**What are Checksums? Please Compare it with the values in the script (line 21)**

**Open Excel spreadsheet, Copy and paste the checksums provided in the script in column A, the checksums you get in column B, in column C use "Exact" function**

**= Exact (A1, B1)**

# How code is structured (Syntax)?

- **Programme command** – the name of the programme you want to execute the action (e.g. guppy\_basecaller)
- **Argument(s) or parameter(s)** – instructions to execute the command in a way you need separated by spaces **(Required) -i (--input) -s (--save\_path) -c (--config)**
- **Flag(s)** – enable you to specify options **(optional) you can see by running the tool with “--help” to see the available flags and options.**
- **Option(s)** – used to explain to the programme what argument/parameter you want it to execute (e.g. filter out all files with less than 4000 reads)

## Other points

- You can 'pipe' code “|” together so that each command is linked together and they follow on
- The path (address) of the tools/programmes can be saved in the \$PATH variable of your working environment this allows you to call the tool/programme by its name only without providing its path each time you need to use it
- **<https://www.baeldung.com/linux/path-variable>**

# How to deal with errors

- Errors are common, especially when you are learning, so don't panic!
- Read through the error – they are designed to help you identify the problem
- Someone will have come across it before: check forums on e.g. Stack Overflow and Biostar and GitHub
- Google! You can copy and paste the error

## Guppy

Guppy is a data processing toolkit that contains ONT basecalling algorithms, and several bioinformatic post-processing features

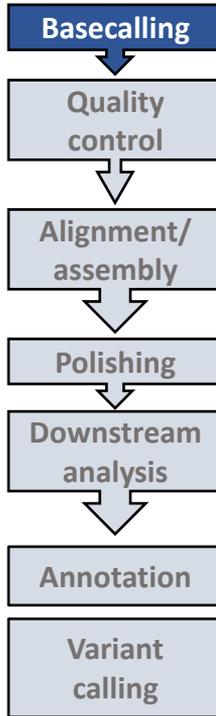
It is integrated with MinKNOW, the ONT device control software

Requirements:

- At least 8 GB RAM
- Windows, MacOS or Linux command line

Guppy has three functions:

- **Basecaller** (converts .fast5 files into .fastq files)
- **Barcoder** (demultiplexes/separates into barcode files, and trims Adapters and barcodes)
- **Aligner** (can align basecalled reads to a reference genome)

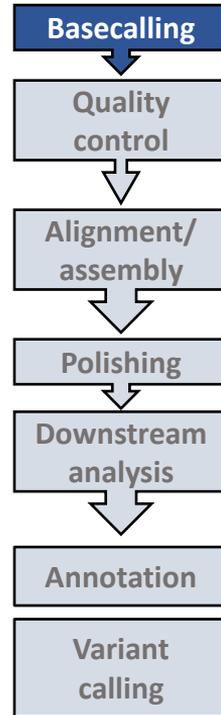


## Today's 'test' data

- Used SQK-RBK004
- Used a R9.4.1 flow cell

# Guppy – how to install

- Windows – download straight from community website
- Can also install directly from the command window (directions can be found in the Guppy document on the community website)



## Guppy basic code – basecalling (Windows)

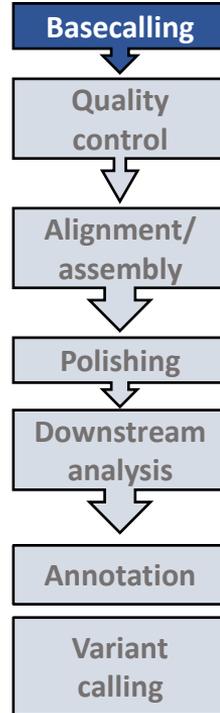
```
<path_guppy_basecaller>  
-i <fast5_skip_folder_address>  
-s <fastq_output_folder_address>  
-c dna_r9.4.1_450bps_fast.cfg
```

Address to where guppy basecaller is stored

Input folder and address

Output folder and address

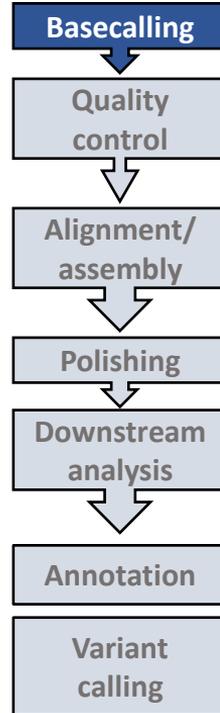
Configuration file and information (flow cell type, kit and fast or high accuracy basecalling)



## Guppy example code (Windows) - basecalling

```
"C:\Program Files\OxfordNanopore\ont-guppy-cpu\bin\guppy_basecaller.exe"
```

```
-i C:\Users\renglel\Documents\Sequencing\tutorial_tests\fast5  
-s C:\Users\renglel\Documents\Sequencing\tutorial_tests\fastq  
-c dna_r9.4.1_450bps_fast.cfg
```



## Guppy basic code – trimming (Windows)

```

<location_of_guppy_barcode>
-i <fast5_skip_folder_address>
-s <fastq_output_folder_address>
-r
-c configuration.cfg
--trim_barcodes
--barcode_kits SQK-RBK004
  
```

-r recursive ('do this to all folders within this one') useful for barcode folders

Tell it what type of barcodes used

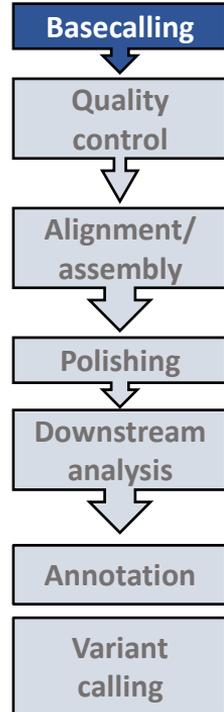
Address to where guppy barcoder is stored

Address of input folder

Address of output folder

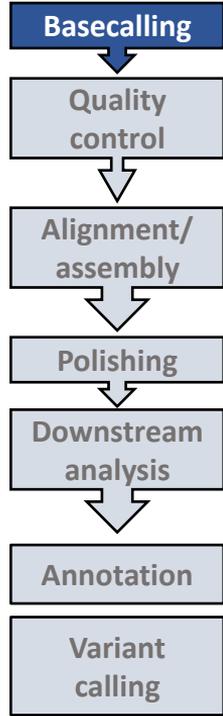
Configuration file

Command to 'trim'



## Guppy example code (Windows) - trimming

```
"C:\Program Files\OxfordNanopore\ont-guppy-cpu\bin\guppy_barcoder.exe" -i  
C:\Users\rengle\Documents\Sequencing\tutorial_tests\fastq -s  
C:\Users\rengle\Documents\Sequencing\tutorial_tests\trimmed -r -c  
configuration.cfg --trim_barcodes --barcode_kits SQK-RBK004
```



**Let's run Guppy Please refer to Script\_ONT.txt (line 57)**

**Please refer to Script\_ONT.txt (line 126)**

**Then Download ready 12 fastq files**

**What are Checksums? Please Compare it with the values in the script**

**Open Excel spreadsheet, Copy and paste the checksums provided in the script in column A, the checksums you get in column B, in column C use “Exact” function**

**= Exact (A1, B1)**

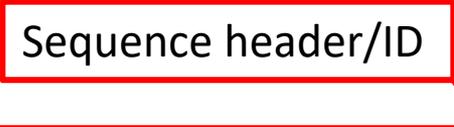
**How many reads in each fastq file?**

**Let’s look into one fastq file barcode01.fastq use the less command**

# FASTA files

- File extension .fna
- This file format is used for reference genomes or reference database sequences
- In this file format, each sequence read is defined in 2 lines: header + sequence code

Sequence header/ID

A red rectangular box with a thin border contains the text 'Sequence header/ID'. A red arrow points from the bottom right corner of the box to the first line of the FASTA sequence example below.

```
>NC_015758.1 Mycobacterium tuberculosis variant africanum GM041182, complete genome  
TTGACCGATGACCCCGGTT CAGGCTTCACCACAGTGTGGAACGCGGTCGTCTCCGAACTTAACGGCGACC  
CTAAGGTTGACGACGGACCCAGCAGTGATGCTAATCTCAGCGCTCCGCTGACCCCTCAGCAAAGGGCTTG
```

# Fastq files

- File extension .fastq
- fastq = fasta + quality scores, result of conversion of squiggles to bases
- Can have multiple sequences in a list
- Each sequence read is defined in 4 lines: Header + sequence code+ Name field (optional, usually empty line starts with "+" sign + 'sequence quality' information each nucleotide base is corresponding to a code that describes its quality )

```

@NS500195:610:HTC5YAFX2:1:11101:3727:1059 1:N:0:TACCTGTG+NTCGGTTG
CAAGGCTGGTCCGGCCTACTCTGATCAGCAATGACCGAACACACCCCGGATATCCCGCTGGGGTCTGGCTGGCCGCTTGTCCAGAGATCGGAAGAGCACACG
CTGGTT
+
AAAAAEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE<EEEEEEEEEEEEEEEEEEEE
/////<
@NS500195:610:HTC5YAFX2:1:11101:9955:1060 1:N:0:TACCTGTG+NTCGGTTG
CGGC GCGACCATTC CCGATCGCCACCGGCGGTGAATG CAGGAAAAA TAGAGCCGCTATCCACAATTCGGCGTCGAGCTCGGCTACCACAAACGGTAGAA
CCGCTC
+
AAAAAEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE/EEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE
EEEE/

```

Example of Fastq File content containing 2 sequence reads; each read is defined in 4 line red square: Header/Sequence ID, green box is the quality scores corresponding to each base

# Anaconda (3 GB) /Miniconda (400 MB)

- A repository for >7,500 open source packages
- Package installing tool (Conda)
- Creating virtual environments (isolated environments for different projects to solve the problem of different tools' versions required for each )



The screenshot shows a web browser displaying the Conda user guide for installation. The browser's address bar shows the URL: [conda.io/projects/conda/en/latest/user-guide/install/index.html](https://conda.io/projects/conda/en/latest/user-guide/install/index.html). The page has a navigation sidebar on the left with a 'User guide' section expanded to show 'Installation' options: 'System requirements', 'Regular installation', 'Installing in silent mode', and 'Installing conda on a system that has other Python installations or packages'. Below this are sections for 'Configuration', 'Tasks', 'Cheat sheet', 'Troubleshooting', 'Conda configuration', 'Conda Python API', 'Command reference', 'Glossary', 'Developer guide', and 'Release notes'. At the bottom of the sidebar is a 'Read the Docs' button and a version selector set to 'v. latest'. The main content area on the right contains text explaining that the fastest way to obtain Conda is to install Miniconda, a mini version of Anaconda. It also recommends installing Anaconda for the local user, which does not require administrator permissions. A 'System requirements' section lists: 32- or 64-bit computer; 400 MB disk space for Miniconda; and a minimum of 3 GB disk space for Anaconda. A 'Note' box at the bottom states that administrative or root permissions are not needed if a user-writable install location is selected.

<https://conda.io/projects/conda/en/latest/user-guide/install/index.html>

<https://repo.anaconda.com/miniconda/>

<https://docs.conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html>

# FastQC

FastQC aims to assess the quality control checks on raw sequence data coming from high throughput sequencing pipelines

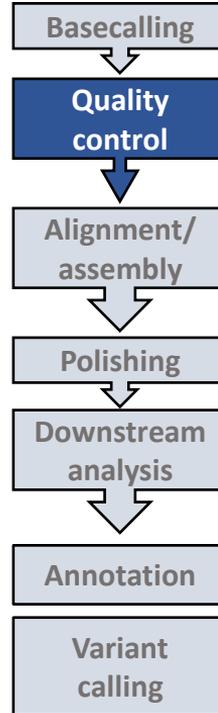
It provides a modular set of analyses which you can use to give a quick impression of whether your data has any problems of which you should be aware before doing any further analysis.

Requirements:

- Java installed
- At least 8 GB RAM
- Operating systems: Linux, MacOS and Windows

A simple way to do some quality control checks on raw sequence data

<https://www.bioinformatics.babraham.ac.uk/projects/download.html#fastqc>



## FastQC – how to install (Linux/MacOs/WSL) line 177

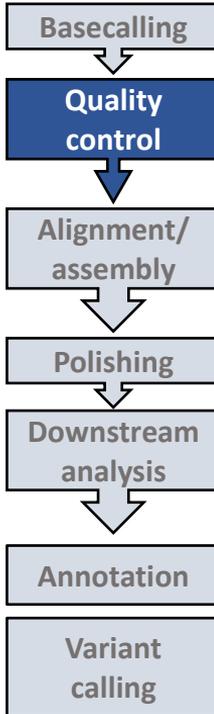
Download the .zip file from the website

You will need to navigate to the folder you downloaded it to and then run:

```
sudo apt install fastqc
```

if you get an error installing it, you may need to update your cache:

```
sudo apt update
```



## FastQC basic code (Linux) (line 191)

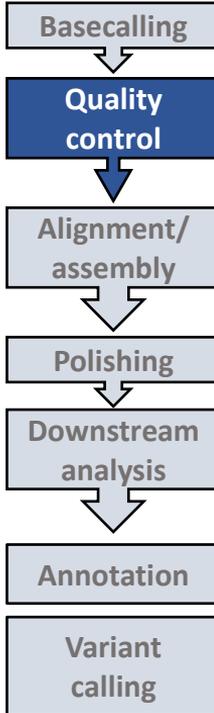
```
fastqc <filepath>/*.fastq -t 2
```

\*.fastq indicates you want to process every file with the extension .fastq

Number of threads you are using

Name of the programme

File address for the .fastq files you want to QC



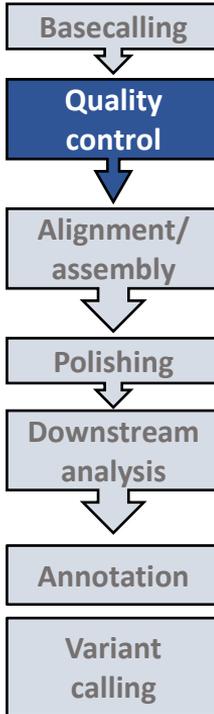
# MultiQC

MultiQC searches a given directory(folder) for analysis logs (e.g. from FastQC) and compiles a HTML report

It's a general use tool for summarising the output from numerous bioinformatics tools, analysing across many samples into a single report

Requirements:

- Java installed
- At least 8 GB RAM
- Operating Systems: Linux, MacOS and Windows



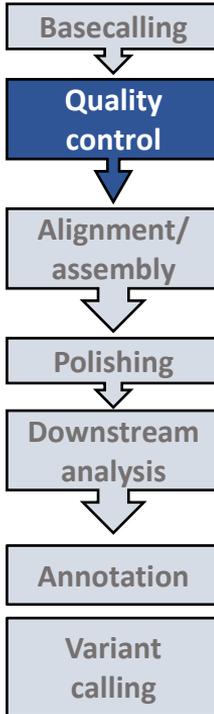
# MultiQC – how to install (Linux/MacOs)

Install multiQC:

(line 198)

```
conda install -c bioconda -c multiqc
```

```
conda install -c bioconda/label/cf201901 multiqc
```

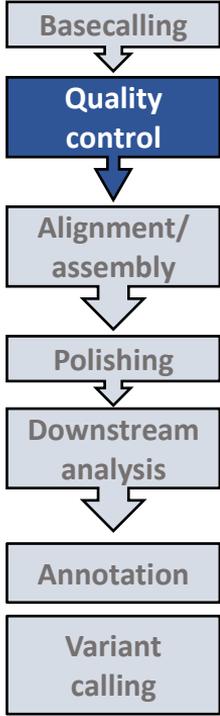


# MultiQC basic code

```
multiqc fastqc
```

'do this to all the files in this folder'

Name of the programme



**Please refer to Script\_ONT.txt (line 174)**

**Let's Install the fastqc**

**Run the fastq files of Barcode 08 and Barcode 09: which is a better Sequence**

**ASANTE  
SANA!**

*Thank You*

The words "Thank You" are written in a dark blue, elegant cursive script. The text is accented with gold dots and starburst shapes, and features gold and black outlines and underlines.