



GenEpi-BioTrain

Raw data and Assembly

Nabil-Fareed Alikhan

28.05.2024

Handout and further reading

Online handout of this material is available at:

<https://bioinformatics-handbook.netlify.app/genepi-biotrain/00-welcome/>

bit.ly/genepi-assembly

I may defer specific/complicated questions to the end



About me



@happy_khan



@happy_khan@mstdn.science

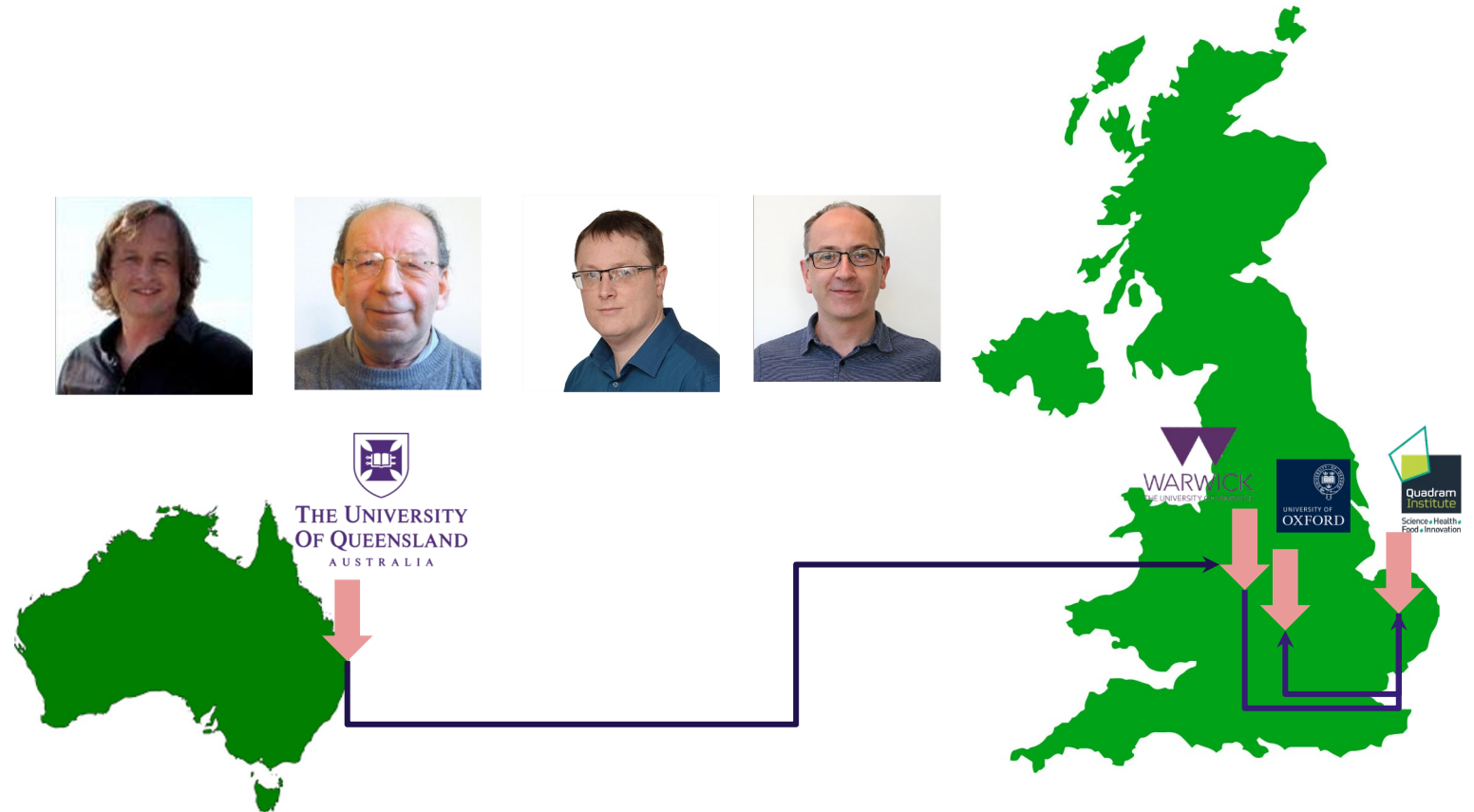


Qualifications

- Doctor of Philosophy (Microbiology). 2010 – 2015. University of Queensland, Australia
- Bachelor of Science (Honours – 1st Class) (Microbiology). 2004 - 2009. University of Queensland, Australia
- Bachelor of Information Technology. 2004 - 2008. University of Queensland, Australia

My Interests

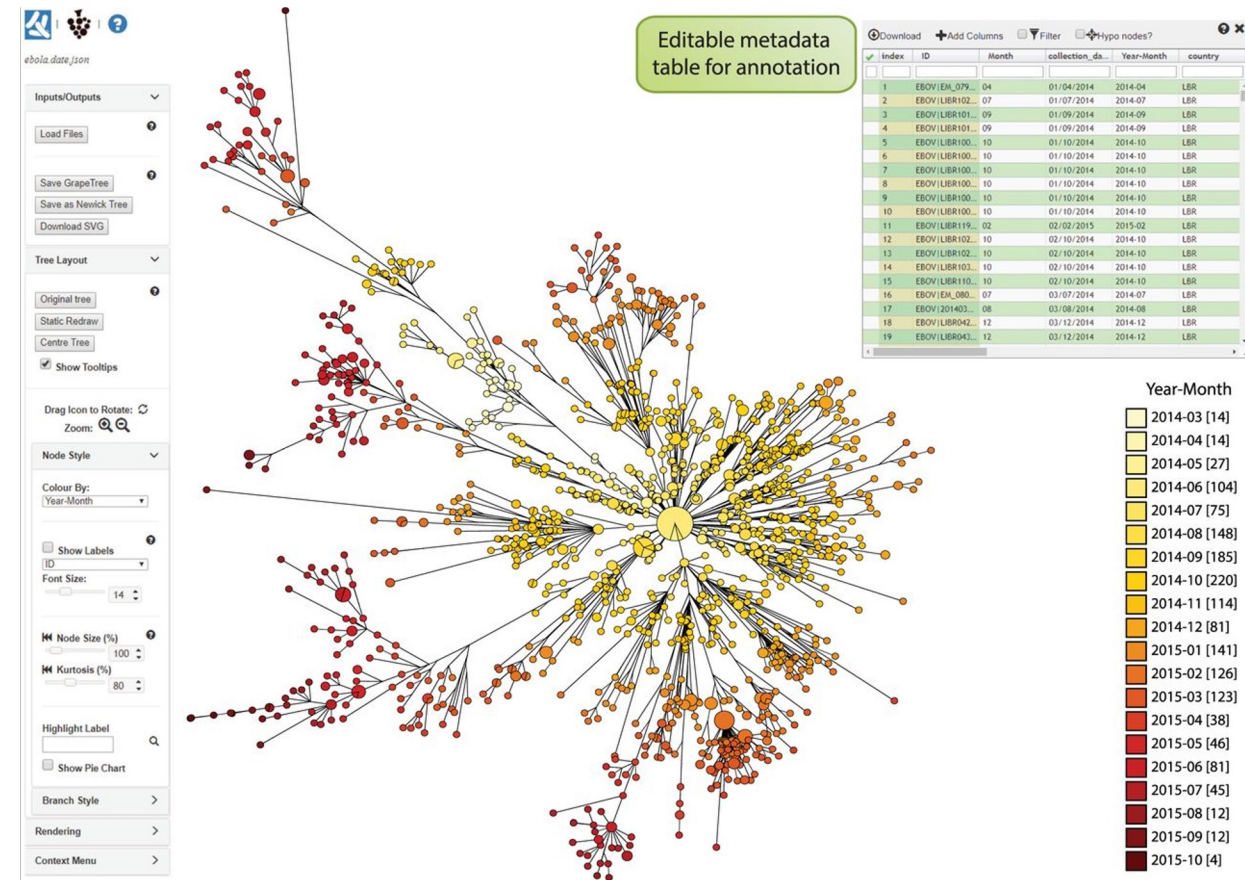
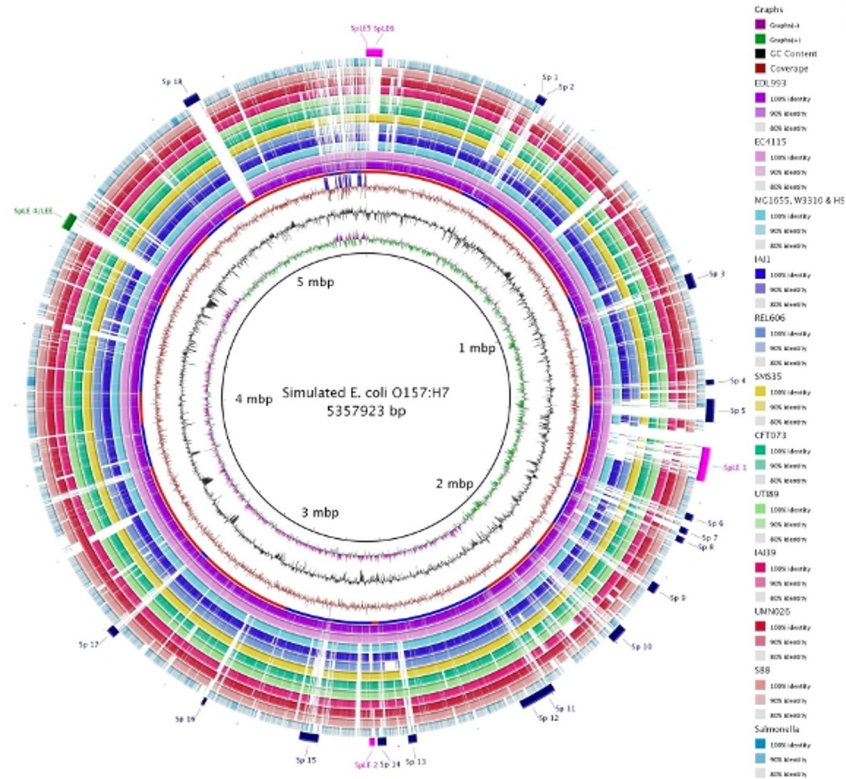
- Population genetics and pathogenesis of enteric pathogens, including *Salmonella* and *E. coli*
- I did some SARS-CoV-2
- Software development, and bioinformatics infrastructure.
- Piña Coladas and long walks on the beach



I (co)write software

Enterobase

<https://enterobase.warwick.ac.uk/>



Intended Learning Objectives

- **Learn to handle raw sequence data, perform quality assessment using fastQC**
- **Know about processing steps such as merging, error correction, and trimming to improve data quality**
- **How to read and interpret a QC report**
- Explore assembly visualization, and **methods for assessing assembly quality**
- **Gain insights into contamination detection**
- Learn to recognize false SNPs and **poor quality assemblies**
- Understanding the impact of poor data quality on epidemiological inferences

Intended Learning Objectives

- **Bioinformaticians** should master raw data quality assessment and processing, including tools like fastQC, to ensure precise genomic data analysis. They must also learn about genome assembly tools and techniques for comprehensive genomic analysis.
- **Microbiologists** must understand the impact of DNA library quality and sequencing quality for accurate data generation. They should use visualization tools to assess raw read quality and ensure accurate interpretation of genomic data.
- **Epidemiologists** must familiarize themselves with these concepts to ensure integration of high-quality NGS data with public health information. They should grasp how raw data or assembly quality influences genotyping data interpretation for effective surveillance.

Outline - Content to cover

1. Why is quality control important? - Issues arising
2. A framework for quality control of whole genome sequencing data
3. Dealing with sequence read data - What is a FASTQ?
4. Quality assessment of sequence read data
5. Basics of genome assembly
6. The 4C's for quality assessment of genome assemblies

Outline - Practicals

- Read classification of our sequenced data: bit.ly/biotrain-readclass
- Quality control for short reads: bit.ly/biotrain-readqc
- Genome assembly QC: bit.ly/biotrain-assemblyqc

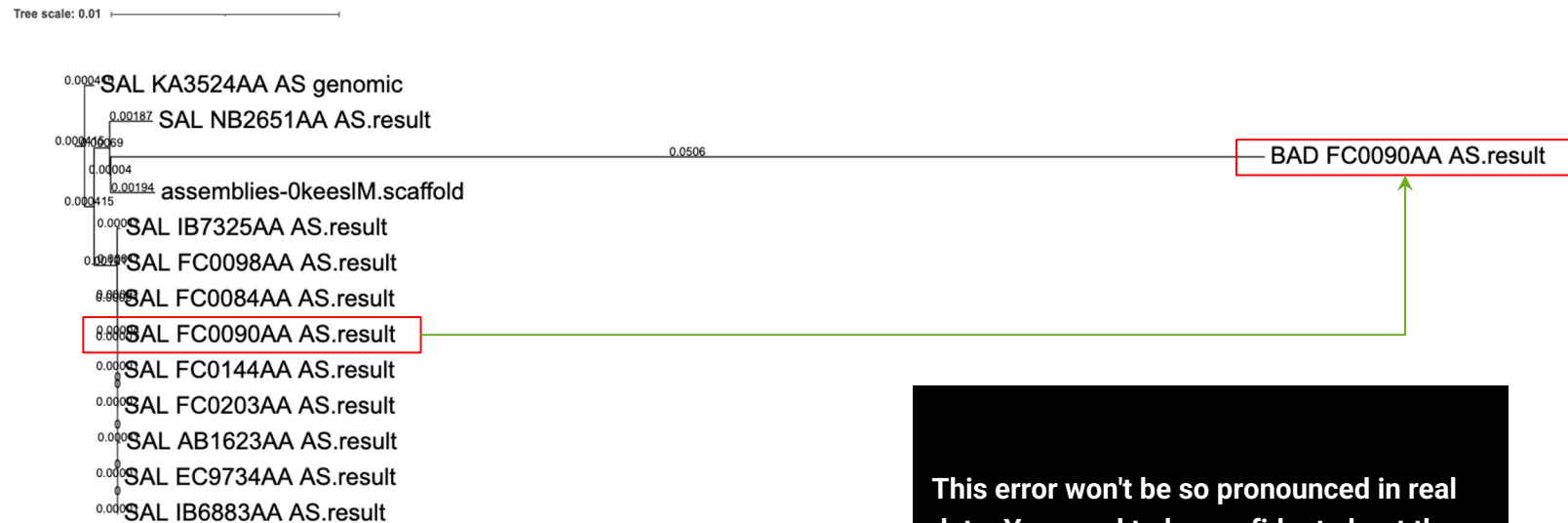
Impact of Errors on Epidemiological Interpretations

Errors can distort understanding of pathogen dynamics, leading to misleading conclusions.

- **Inflating Genetic Diversity**
 - Artificial nucleotide differences due to sequencing errors.
 - Overestimation of genetic diversity.
 - False appearance of multiple sources of infection instead of a single outbreak.
- **False Exclusion of Isolates and Incorrect Phylogenetic Placement**
 - Isolates placed on incorrect branches of phylogenetic trees.
 - Erroneous conclusions about isolates' involvement in outbreaks.
 - Complicated efforts to trace transmission pathways.
- **Misinterpretation of Transmission Dynamics**
 - Disruption of transmission chain reconstruction.
 - Inaccurate mapping of pathogen spread.
 - Misidentification of cases as new introductions rather than continuations of existing chains.

The real effect of poor data

- Selected 12 *Salmonella enterica* ser. Choleraesuis samples
- Created a poorly assembled genome from one sample (SAL_FC0090AA_AS)
- Generated a neighbour joining tree using mashtree based on average nucleotide identity
- BAD_FC0090AA_AS is a clear outlier
- Would BAD_FC0090AA_AS be considered part of the outbreak?



A very special *Salmonella* Typhi

- A researcher presented results indicating the presence of antimicrobial resistance (AMR) determinants in *Salmonella enterica* serovar Typhi.
- The table used black to indicate the presence of AMR determinants and white for their absence.
- The researcher was excited about a unique profile with additional AMR mechanisms in one of their samples.

	Fluoroquinolones	Cephalosporins (3rd gen.)	Aminoglycosides	Carbapenems
Sample 1	[Black]	[Black]		
Sample 2				
Sample 3				
Sample 4				
Sample 5	[White]	[Black]	[Black]	
Sample 6	[Black]	[Black]		
Sample 7				
Sample 8				
Sample 9				
Sample 10				

I did a basic check of the taxonomic classification of the sample and it came back - *Klebsiella pneumoniae*.

It was not a special Typhi, but a run of the mill *Klebsiella* that had been picked up by mistake.

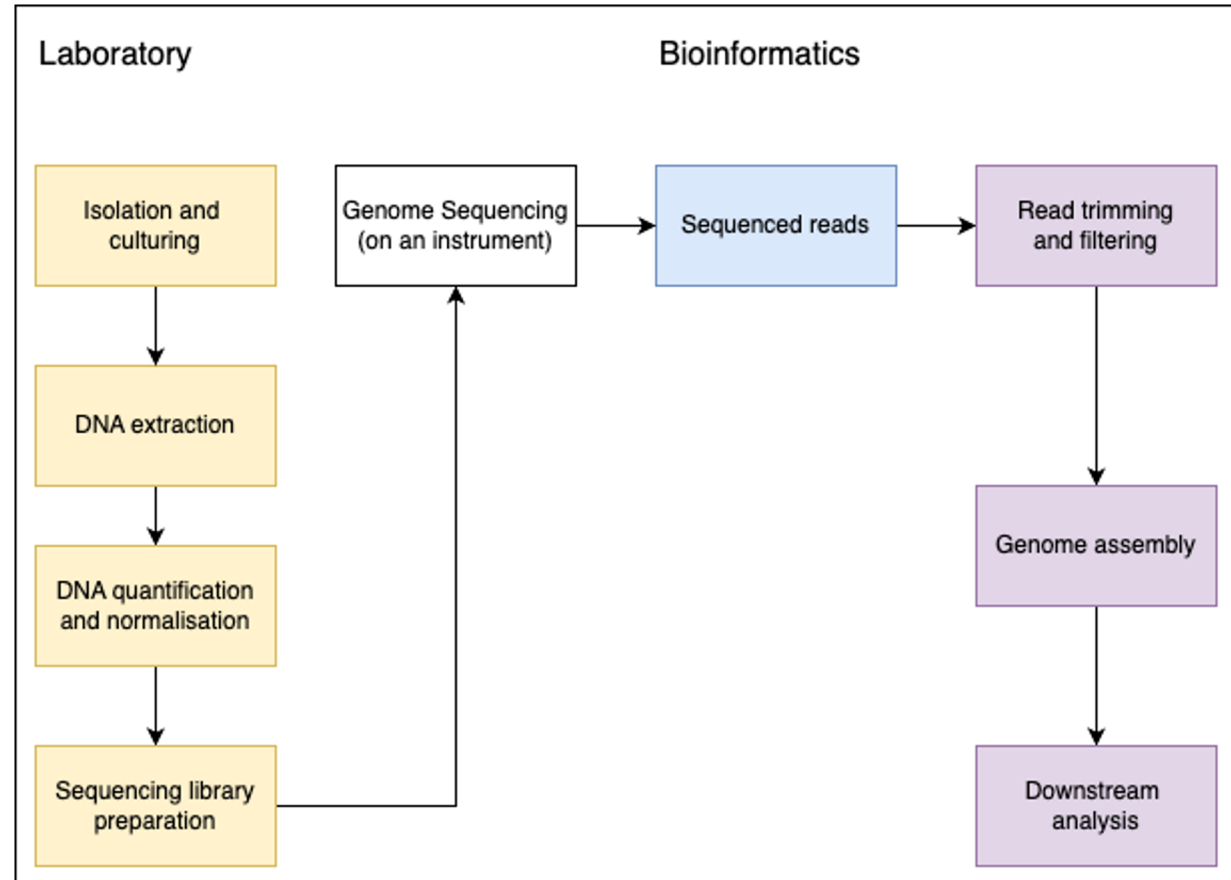
The null result

The most common manifestation of sequencing and genome assembly errors is that a downstream tool just doesn't work. Here is an error thrown by snp calling tool “snippy” when given poor quality data.

```
Opening: core.tab
Opening: core.vcf
Processing contig: NZ_CP030111.1
Processing contig: NZ_CP030112.1
Processing contig: NZ_CP030113.1
Processing contig: NZ_CP030114.1
Processing contig: NZ_CP030115.1
Processing contig: NZ_CP030116.1
Generating core.full.aln
Creating TSV file: core.txt
Running: snp-sites -c -o core.aln core.full.aln
Warning: No SNPs were detected so there is nothing to output.
ERROR: Could not run: snp-sites -c -o core.aln core.full.aln
```

Remember the context

When receiving results from bacterial genomics analyses such as genotyping, *in silico* serotyping, clustering, phylogenetic inference, and predicting antimicrobial resistance (AMR) determinants, you should remember that your data has traversed a laborious and exhaustive journey.

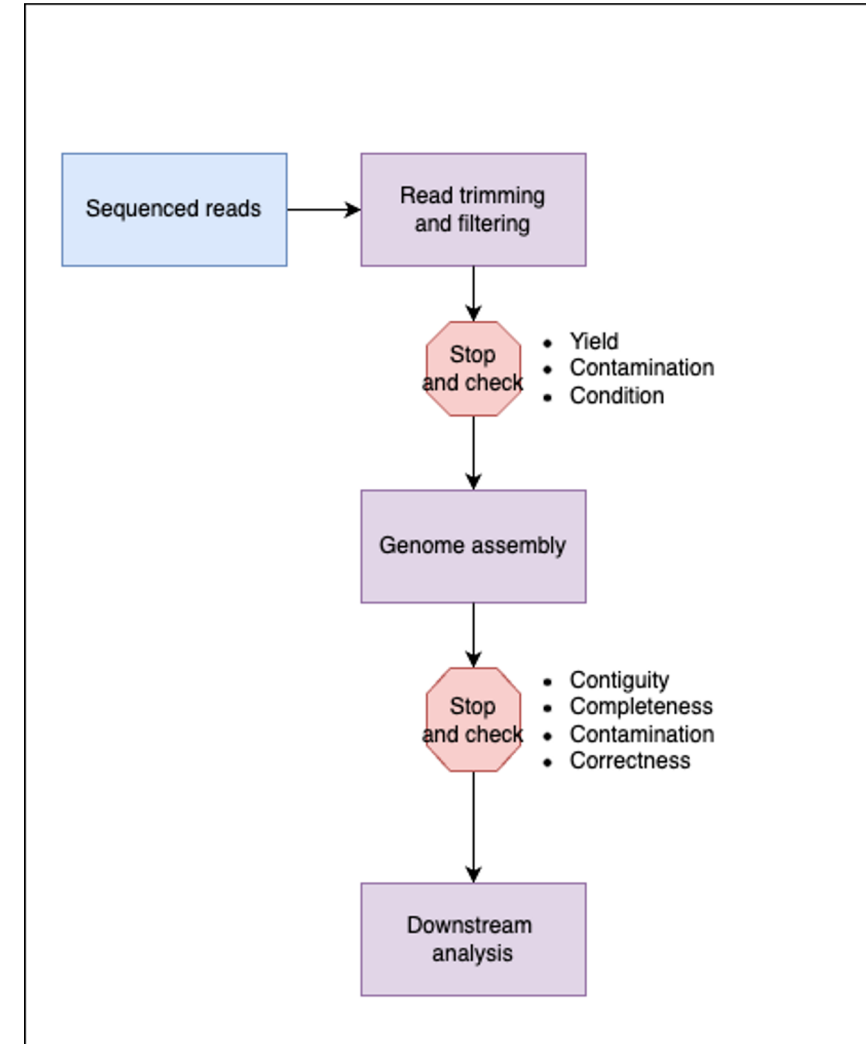


Remember the context

Each step in the process has error potential, which could significantly impact the final interpretation.

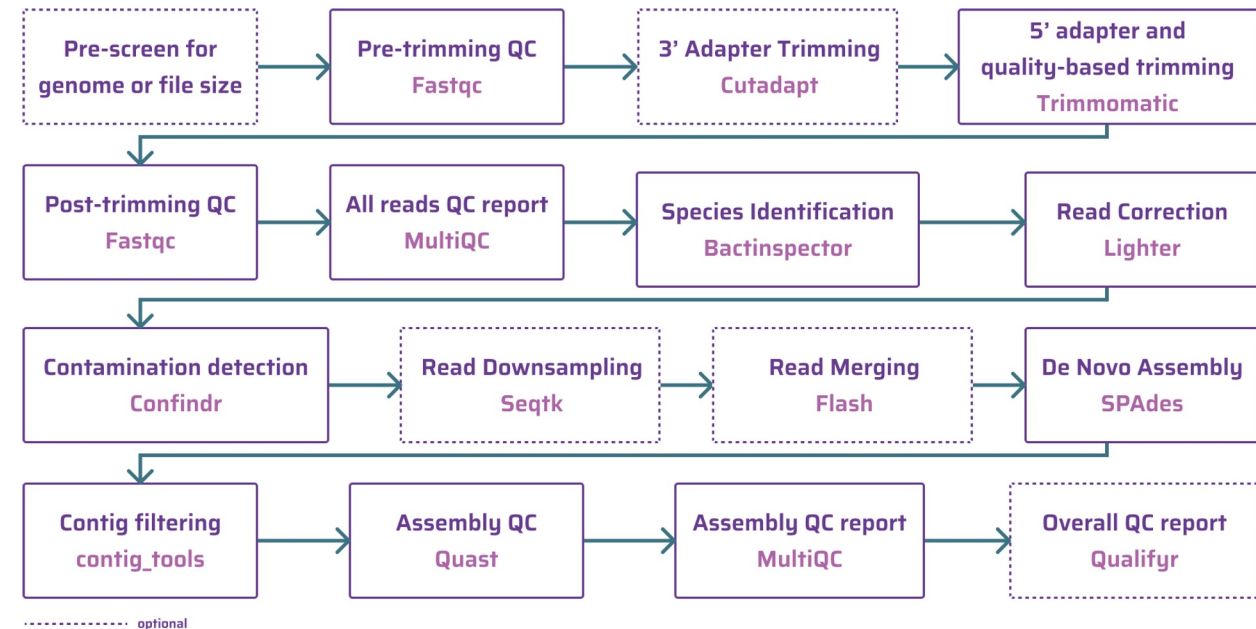
As a Bioinformatician post-sequencing, there are two crucial opportunities to assess data quality in our workflow:

- Check the sequenced reads directly (FASTQ output) after basic filtering like adapter removal.
- Evaluate the quality of the assembled genome derived from the sequenced reads.



Remember the context

- Bioinformaticians often have varied methods for assessing data quality, typically listing tools without detailed explanations of selection criteria.
- Here I will emphasize a problem-solving approach over personal tool preferences.



GHRU SPAdes Assembly workflow.

<https://gitlab.com/cgps/ghru/pipelines/dsl2/pipelines/assembly>

Quality control key questions

Genomic data quality control tools address key questions:

- *Do I have enough sequenced reads for my work?*
- *Are the sequenced reads from the organism I am expecting?*
- *Does the quality scoring, provided by the instrument, meet my expectations?*
- *Does the genome assembly look like an intact genome from the organism I am expecting?*

Quality control key criteria

Four key questions for genomic data quality control break down into seven criteria, plus a bonus:

- *Sequence reads*
 - *Yield*
 - *Contamination*
 - *Condition*
- *Genome assembly*
 - *Contiguity*
 - *Completeness*
 - *Contamination*
 - *Correctness*
- *BONUS:*
 - *Circumstantial*

Wake up! Practical
questions start soon

Just breathe

- Don't feel overwhelmed by the multitude of tools available for assessing genomic data.
- Focus on the seven criteria listed and select an approach that covers each criterion.
- Tool selection and threshold determination depend on your specific research question.
- Regularly used tools for typical scenarios will be discussed.



Criterion: Yield (for Sequenced reads)

- Sufficient sequenced reads ensure representation of every genomic position and confidence in base calls.
- Oversampling compensates for:
 - Non-random DNA selection, leading to coverage variation across the genome.
 - Errors introduced during sequencing, allowing correction through consensus base calling.

Calculating genome coverage

- A quick calculation of the average coverage across a genome.
- We need to know the length of the original genome (G), the number of reads (N), and the average read length (L) to calculate the coverage which is:

$$N \times L / G$$

Try this yourself:

For a bacterial genome of 5 megabase pairs (5,000,000 bp) and an average read length of 150 bases. How many reads do you need to have 30 times

Calculating genome coverage

If our sequencing platform has 20 gigabase pairs (20,000,000,000 bp) yield per sequencing run, how many isolates could we sequence; given the values

- Optimal yield depends on organism and use case.
- Recommendations vary:
 - Read Mapping Analysis: Aim for at least five times coverage.
 - Genome Assembly: Aim for at least twenty times coverage for reasonable results.
- For most use cases:
 - I recommended Genome coverage between 40 to 100

Criterion: Contamination (for sequenced reads)

- One approach: Confirming organism consistency involves aligning sequenced reads to an expected reference genome.
- Steps include fetching a reference genome from GenBank and mapping reads to it.
- Assessment of mapped versus unmapped reads provides insight into organism match.
- Preferred read mappers include minimap2 for its speed and versatility across sequencing technologies, and Bowtie2 as another viable option.

For simple data checks, prioritize tools that are appropriate, reliable, fast, and familiar. Avoid overthinking tool selection for

Criterion: Contamination (for sequenced reads)

- Another approach: Taxonomic classification tools compare sequenced reads to a reference genome database to assign taxonomy.
- Numbers are summarized into an abundance breakdown of detected taxa.
- Popular tools include Kraken2, favored for Illumina data due to its appropriateness, reliability, speed, and familiarity.
- Alternative contamination detection approaches will be discussed later.
- **Proceed to "Practical - Read classification of our sequenced data."** bit.ly/biotrain-readclass

Practical: Contamination (for sequenced reads)

For this exercise, we have three of the same samples that were processed in three different labs, for a total of nine samples. Usually, you know which organisms have been sent to you, but in this case I will let you figure that out from the data provided.

Proceed to "Practical - Read classification of our sequenced data." bit.ly/biotrain-readclass

Remember that there are three original isolates (Sample-1, Sample-3, Sample-8), that have been processed by three different groups (Lab-1, Lab-2, Lab-3); This means that we expect "Lab-1-Sample-1", "Lab-2-Sample-1", "Lab-3-Sample-1" to be the same.

Practical: Contamination (for sequenced reads)

Nabil, you don't have enough time to run Kraken2! Just use the prepared results.

Your tasks are:

- View the Kraken2 reports (prepared results on webpage - under "Help! I'm stuck")
- View the Krona results (prepared html on webpage - under "Help! I'm stuck")
- Visualise the result in Pavian
- Instructions to run on Galaxy are there if you want to try later. Do not do this *now*.
- The webpage has many tips to help you! Or ask me.

Then use this information to answer the following questions:

- **Which species were each sample supposed to be?**
- **Are there indications of contamination?**
- **If there is contamination, what are the top three (in terms of abundance) other species identified?**
- **For each sample, how many reads were unclassified?**
- **Consider the typical genome size for each species, and calculate whether the samples have enough coverage for genome assembly.**
- **What are some possible sources of contamination (if any)? You can simply speculate.**

Remember that there are three original isolates (Sample-1, Sample-3, Sample-8), that have been processed by three different groups (Lab-1, Lab-2, Lab-3); This means that we expect "Lab-1-Sample-1", "Lab-2-Sample-1", "Lab-3-Sample-1" to be the same.

Time for the answers!

- Which species were each sample supposed to be?
- Are there indications of contamination?
- If there is contamination, what are the top three (in terms of abundance) other species identified?
- For each sample, how many reads were unclassified?
- Consider the typical genome size for each species, and calculate whether the samples have enough coverage for genome assembly.
- What are some possible sources of contamination (if any)? You can simply speculate.

Criterion: Condition (for sequenced reads)

- Condition assessment focuses on intrinsic sequencing data quality, detecting errors like poor read quality, short read lengths, or unexpected artefacts.
- Illumina platforms (bcl2fastq) offer detailed reporting, aiding analysis.
- Considerations for assessing sequencing data condition include:
 - Total number and average length of sequences
 - GC content
 - Base-by-base sequence quality
 - Proportion of each nucleotide at each position
 - Percentage of ambiguous base calls (N)
 - Sequence duplication
 - Overrepresented sequences
 - Adapter content
- **Base quality interpretation and values are explained in "Dealing with sequence read data - What is a FASTQ?".**
- Utilize FASTQC to analyze example data and understand these considerations further.

Practical - Quality control for short reads

Go to Quality control for short reads: bit.ly/biotrain-readqc

We will be using some example data to assess the quality of short reads. We will use the tool, FASTQC. We will also use some tools to trim poor quality reads (or parts of reads).

Good vs bad

- pKP1-NDM-1: reads are simulated reads, with minimal error.
- female_oral2: This is a microbiome sample (16S) from a snake

Practical - run FASTQC

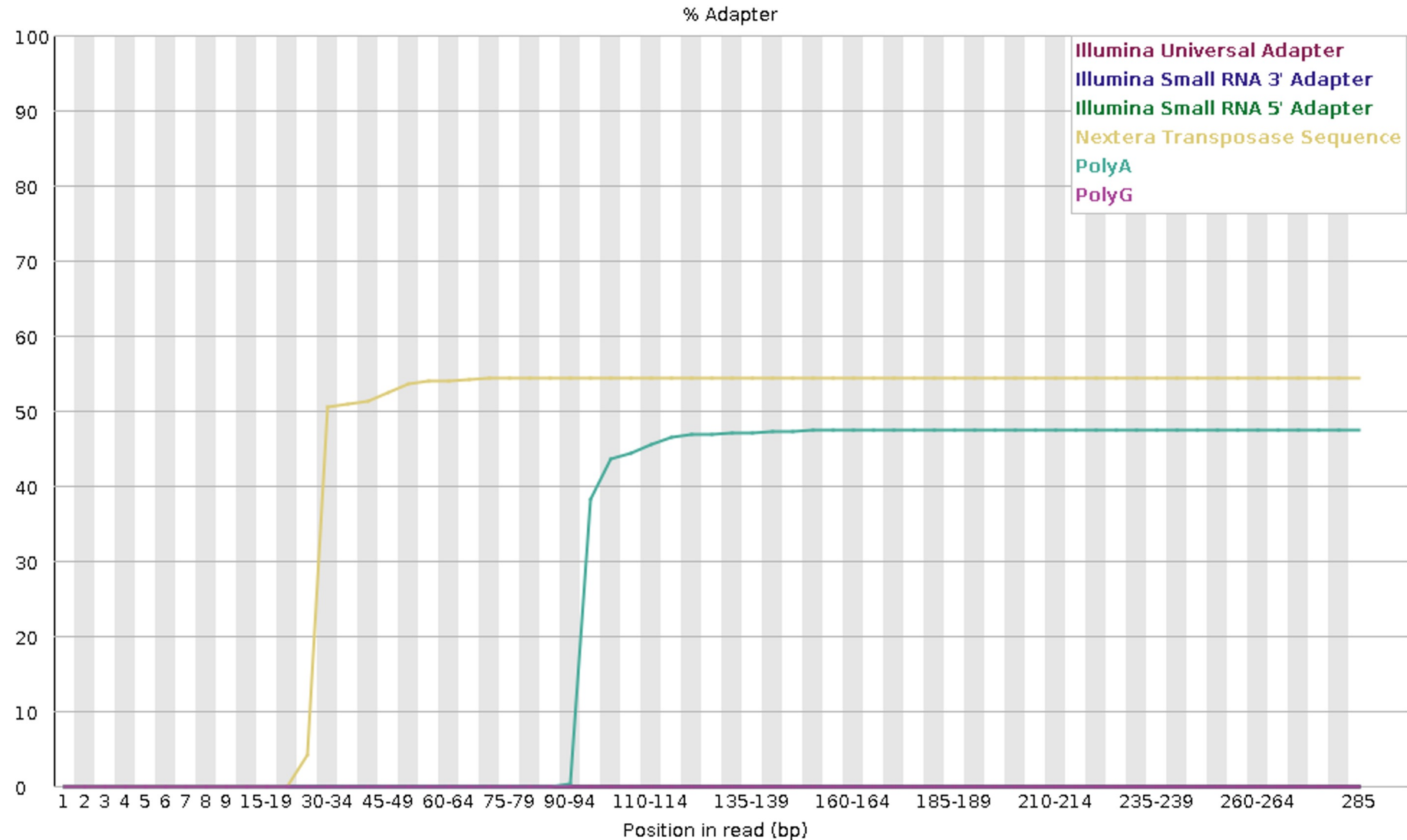
Go to Quality control for short reads: bit.ly/biotrain-readqc

- Run FASTQC on female_oral2.fastq.gz.
- Run FASTQC on pKP1-NDM-1_R1.fastq.gz and pKP1-NDM-1_R2.fastq.gz together.
- Review and compare the HTML reports.
- Precalculated results on website if you are stuck
- Which metrics are a major difference between the two reports?
- Review each metric for female_oral2.fastq.gz, what part of each plot suggests there is a problem?
- female_oral2.fastq.gz data looks terrible, we should probably resequence it, but if we had to; how could we improve the quality?

Time for the answers!

- Which metrics are a major difference between the two reports?
- Review each metric for female_oral2.fastq.gz, what part of each plot suggests there is a problem?
- female_oral2.fastq.gz data looks terrible, we should probably resequence it, but if we had to; how could we improve the quality?

Did you spot this problem?



Practical continues - Trim and filtering reads

- Quality drops in sequences' middle sections can introduce bias in downstream analyses.
- To mitigate bias, sequences require treatment, typically involving trimming to:
 - Remove low-quality score regions.
 - Address issues at the beginning or end of sequences.
 - Eliminate adapters.
- Additionally, filtering involves:
 - Removing sequences with low mean quality scores.
 - Discarding sequences that are too short or contain too many ambiguous bases (N).

Practical continues - Trim and filtering reads

We will use Cutadapt, a tool that enhances sequence quality by automating adapter trimming as well as quality control.

- Trim low-quality bases from the ends. Quality trimming is done before any adapter trimming. We will set the quality threshold as 20, a commonly used threshold.
- Trim adapter with Cutadapt. For that we need to supply the sequence of the adapter. In this sample, Nextera is the adapter that was detected. We can find the sequence of the Nextera adapter on the Illumina website here [CTGTCTCTTATACACATCT](#). We will trim that sequence from the 3' end of the reads.
- Filter out sequences with length < 20 after trimming

Tasks:

- **Use cutadapt to trim the adapter sequence from the 3' end of the reads, and filter out sequences with a length less than 20 after trimming.**
- **Run FASTQC on the trimmed data and compare to the original file.**
- **Does the per base sequence quality look better?**
- **Is the adapter gone?**
- **What can you say about some of the other metrics?**

Time for the answers!

- **Does the per base sequence quality look better?**
- **Is the adapter gone?**
- **What can you say about some of the other metrics?**

Checkpoint. It's time for genome assemblies

Quality control key criteria

Four key questions for genomic data quality control break down into seven criteria, plus a bonus:

- *Sequence reads*
 - *Yield*
 - *Contamination*
 - *Condition*
- *Genome assembly*
 - *Contiguity*
 - *Completeness*
 - *Contamination*
 - *Correctness*
- *BONUS:*
 - *Circumstantial*

Quality control criteria for genome assemblies

- Genome assembly quality control aims to determine if the assembly resembles the expected organism's intact genome.
- Criteria like Contiguity, Completeness, Contamination, and Correctness assess this in various ways.
- Genome assembly involves multiple steps, each introducing errors, making automatic perfect sequence generation impossible.
- Available tools aid in assessing assembly quality, with some common ones to be discussed.

Criterion: Contiguity (Genome assembly)

- Contiguity measures genome assembly continuity, with fewer and longer contigs preferred.
- Assessment methods include:
 - Tracking less contigs and longer average contig lengths.
 - Utilizing metrics like N50 and average contig length.
- Consider using QUAST for contiguity assessment.

A "contig" (short for contiguous sequence) is a set of overlapping DNA segments that together represent a consensus region of DNA. In the context of genome assembly, contigs are created by piecing together shorter sequences, called reads, that have been obtained from sequencing

Criterion: Completeness (Genome assembly)

- Assess genome completeness by comparing assembly to a reference genome or through essential gene analysis.
- Methods include:
 - Comparing to a reference genome using web BLAST or QUAST metrics.
 - Aligning genomes with tools like Mauve or Artemis.
 - Verifying presence of single-copy essential genes using MLST, BUSCO, or CheckM panels.

Criterion: Contamination (Genome assembly)

- Contamination in genome assembly arises from extraneous DNA sources, compromising assembly accuracy.
- Common contamination sources:
 - Sample cross-contamination
 - Contaminated reagents, kits, or environment
 - Human contamination or cross-talk in multiplexed sequencing
 - Lab equipment contamination or library preparation artifacts
- Kraken2 can detect contamination in assembled genomes, similar to its use with sequence reads.

Criterion: Correctness (Genome assembly)

- Assess assembly correctness by checking for errors like mis-joins, collapsed repeats, and duplication artifacts.
- Verify absence of false SNPs or InDels and ensure plasmids are correctly assembled.
- Compare to other assemblies or map original reads back to identify discrepancies.
- Utilize visualization tools like Artemis or Bandage for comprehensive assessment.

Practical - Genome assembly QC

Go to Genome assembly QC: bit.ly/biotrain-assemblyqc

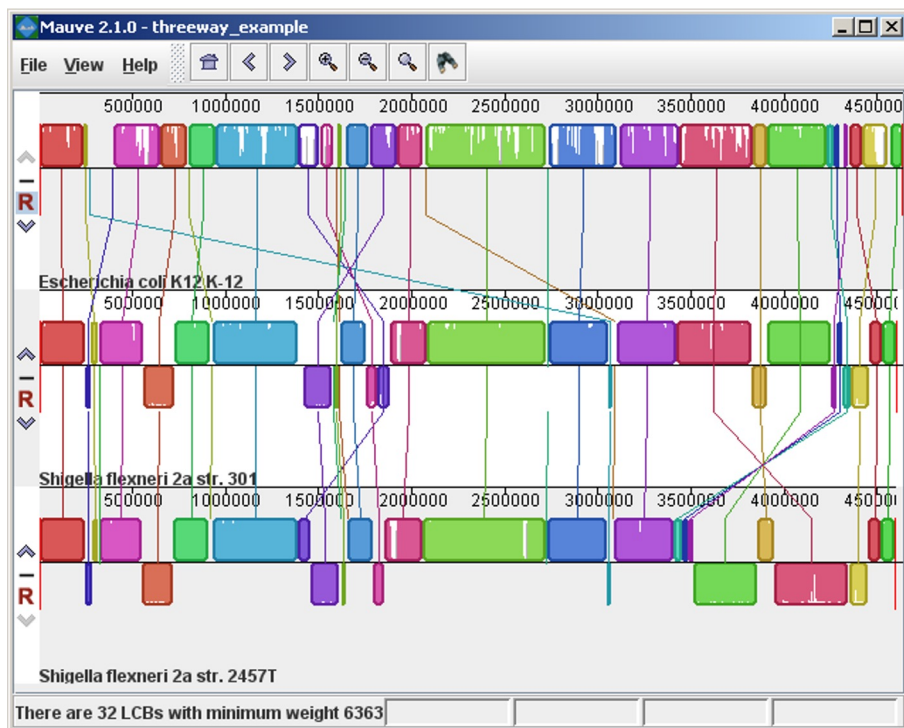
- You have 8 *E. coli* genome assemblies to assess. You need to decide if they pass or fail, using the provided prepared output.
- You will use output from four tools; BUSCO, Kraken2, MLST, QUAST. What criteria can these tools address?

BONUS: Circumstantial evidence

Circumstantial evidence of a good genome includes:

- GC Content matching species expectations.
- Assessment of repeat content to avoid fragmented or misassembled genomes.
- Examination of raw sequencing reads for high quality and minimal errors.
- Evaluation of coverage depth for uniformity across the genome.
- Visualization using tools like Artemis, IGV, or Tablet for confirmation.
- Genome assembly quality may vary based on sequencing technology, software parameters, and DNA quality.
- Careful evaluation and validation are crucial for accuracy.

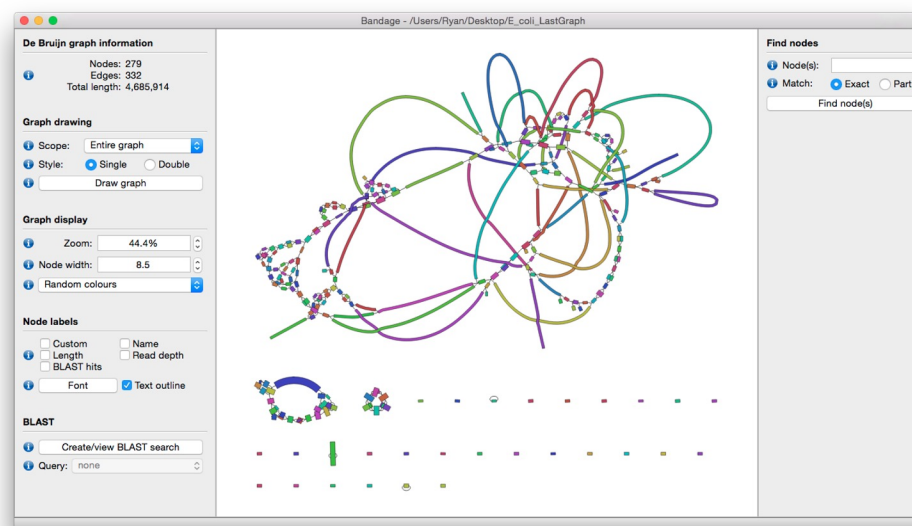
Genome assembly visualization



Mauve



Artemis



Bandage

Acknowledgements

The creation of this training material was commissioned by ECDC to Institut Pasteur with the direct involvement of Nabil-Fareed Alikhan