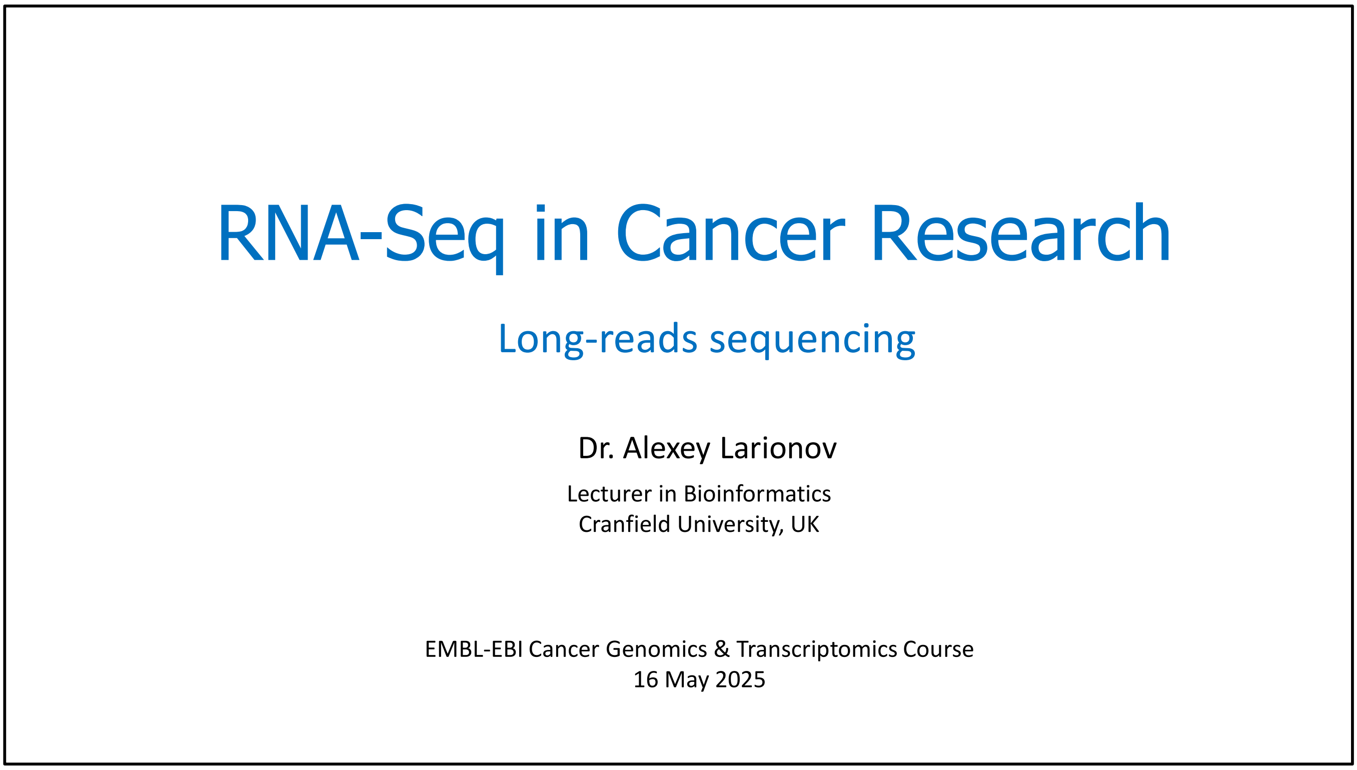


RNA-Seq in Cancer Research

Long-reads sequencing

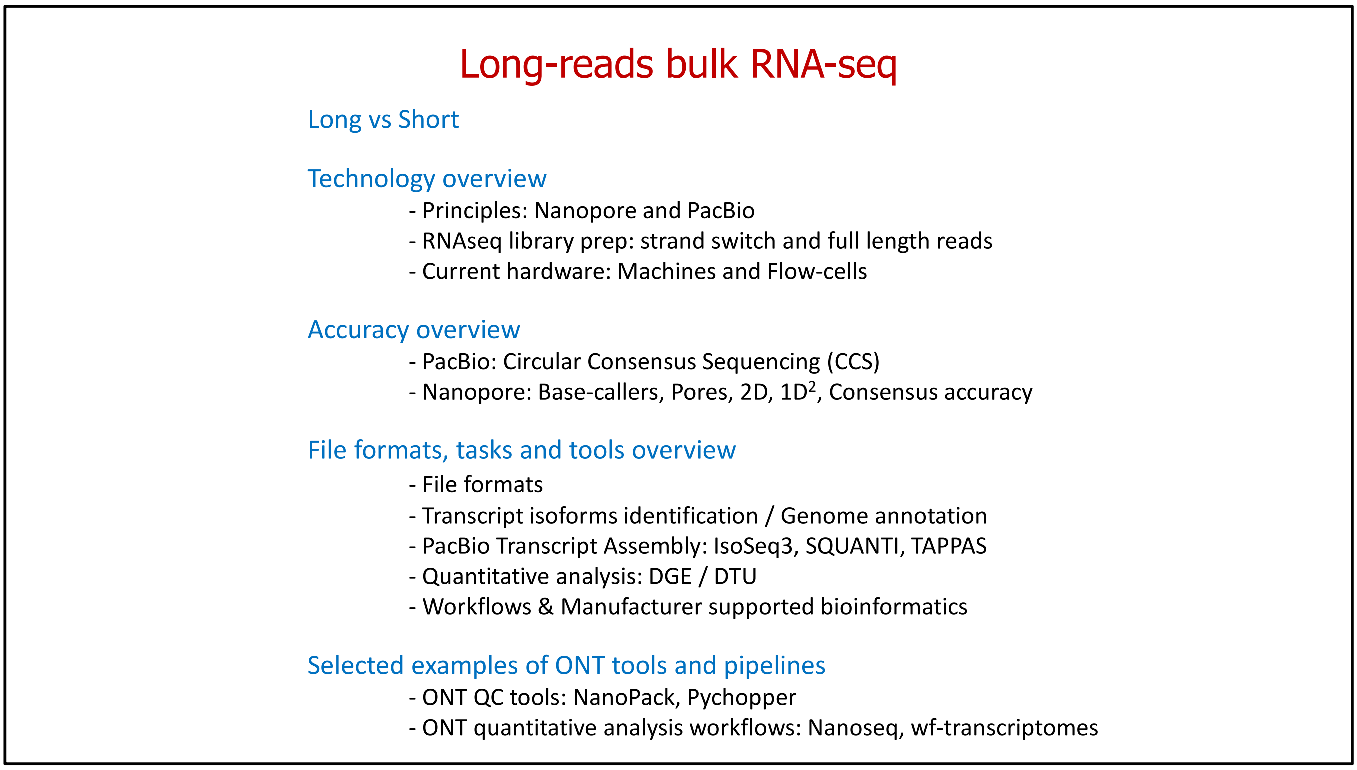
Dr. Alexey Larionov

Lecturer in Bioinformatics
Cranfield University, UK

EMBL-EBI Cancer Genomics & Transcriptomics Course
16 May 2025

Because of the broad scope of the course, and the limited time, this lecture will provide only a very high-level review of long-read RNA sequencing.

The participants who already have experience with long-reads RNA-seq are ***very welcome*** to contribute their comments along the lecture:
just switch on your microphone and contribute!



Long-reads bulk RNA-seq

Long vs Short

Technology overview

- Principles: Nanopore and PacBio
- RNAseq library prep: strand switch and full length reads
- Current hardware: Machines and Flow-cells

Accuracy overview

- PacBio: Circular Consensus Sequencing (CCS)
- Nanopore: Base-callers, Pores, 2D, $1D^2$, Consensus accuracy

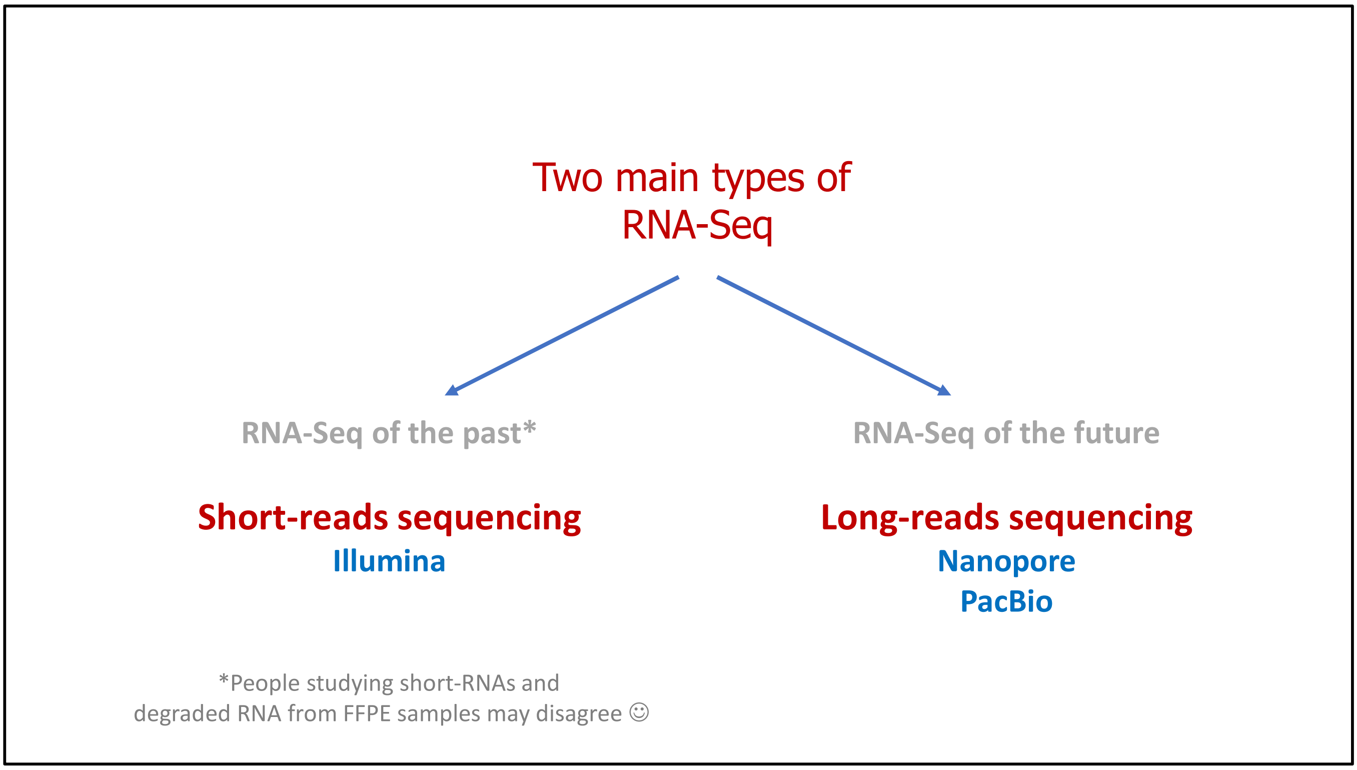
File formats, tasks and tools overview

- File formats
- Transcript isoforms identification / Genome annotation
- PacBio Transcript Assembly: IsoSeq3, SQUANTI, TAPPAS
- Quantitative analysis: DGE / DTU
- Workflows & Manufacturer supported bioinformatics

Selected examples of ONT tools and pipelines

- ONT QC tools: NanoPack, Pychopper
- ONT quantitative analysis workflows: Nanoseq, wf-transcriptomes

This is the plan of the lecture.
I will start with an overview of long reads sequencing technologies,
then focus on some RNA-seq specific aspects of long reads sequencing library preparation, and
then I will go to bioinformatics aspects: discussing selected tasks and tools.



Two main types of RNA-Seq

RNA-Seq of the past*

Short-reads sequencing

Illumina

RNA-Seq of the future

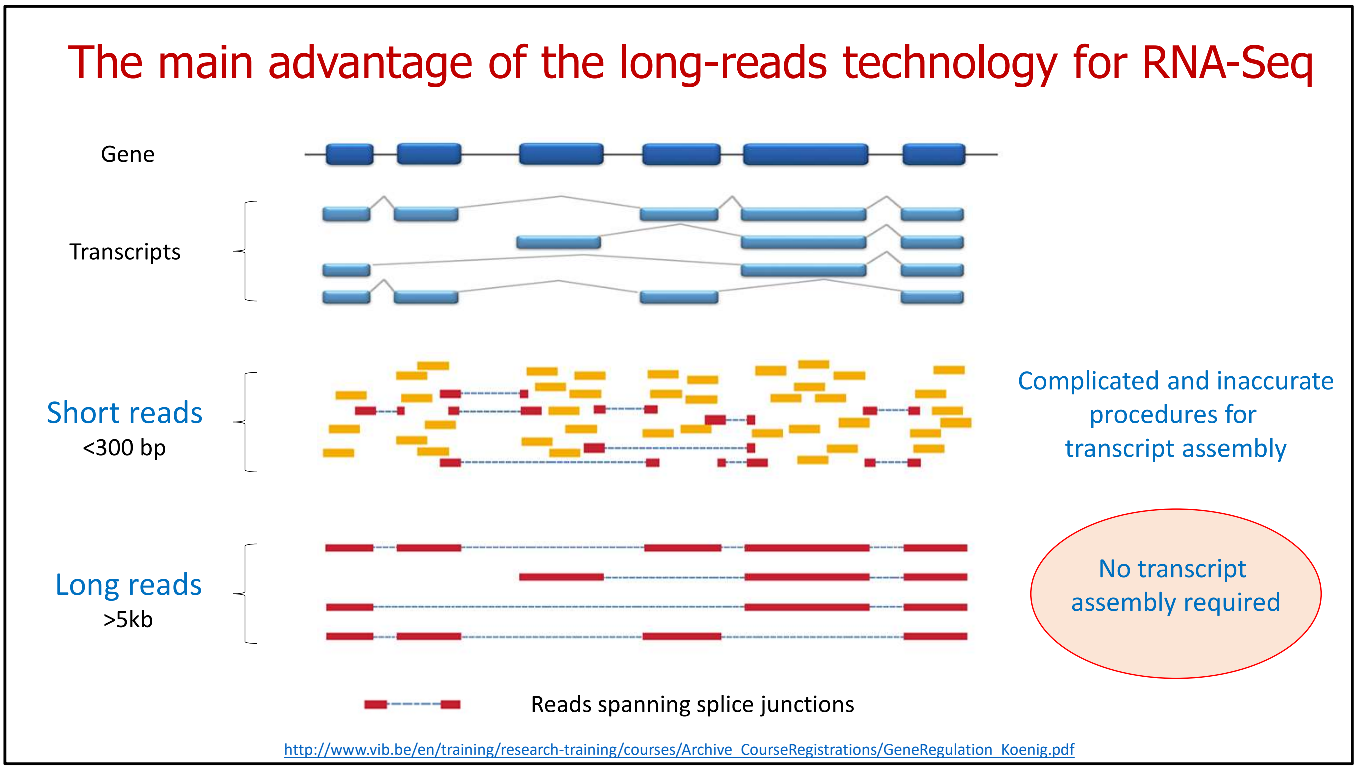
Long-reads sequencing

Nanopore

PacBio

*People studying short-RNAs and degraded RNA from FFPE samples may disagree ☺

As I already mentioned earlier that there are two main types of RNA-sequencing: short-read and long-read sequencing ☺



The main advantage of the long-reads technology for RNA-Seq

Gene

Transcripts

Short reads
<300 bp

Complicated and inaccurate procedures for transcript assembly

Long reads
>5kb

No transcript assembly required

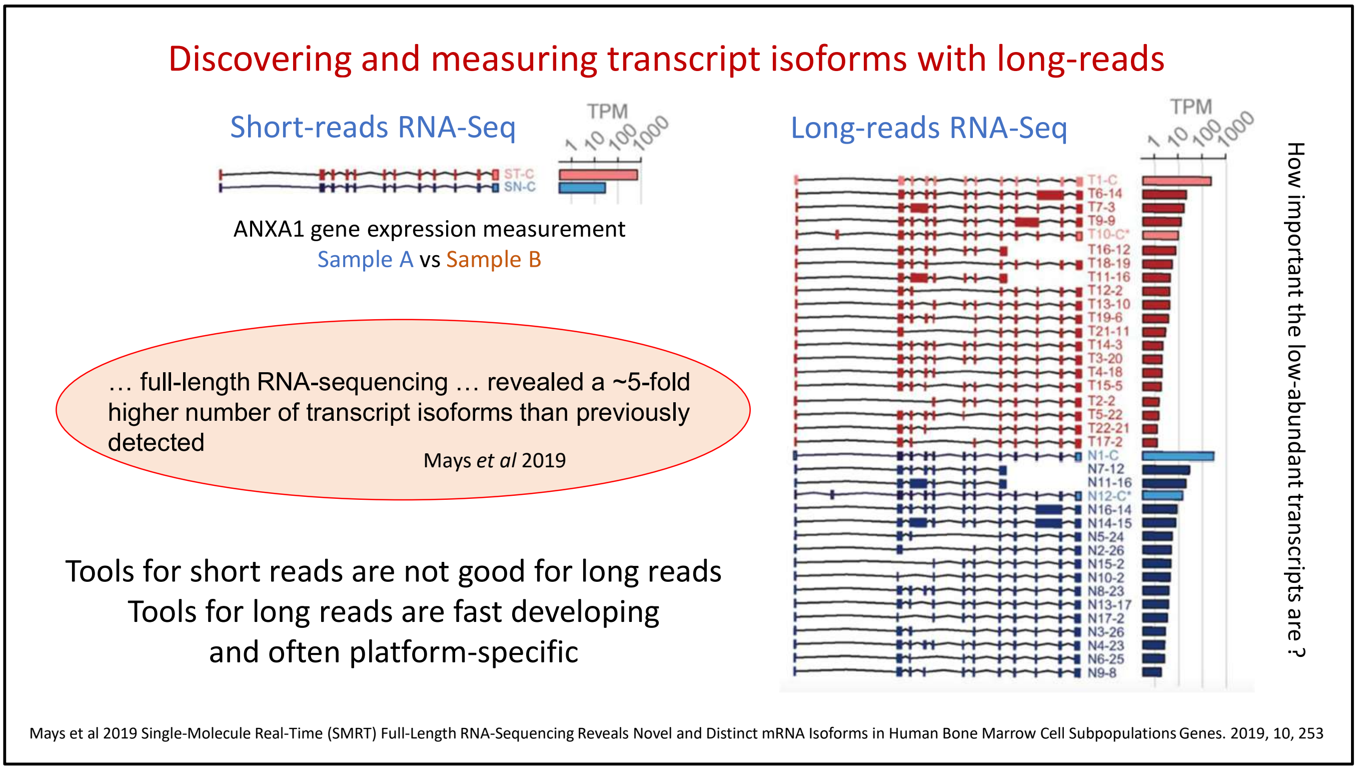
Reads spanning splice junctions

http://www.vib.be/en/training/research-training/courses/Archive_CourseRegistrations/GeneRegulation_Koenig.pdf

The main advantage of the long-reads technology for RNA-Seq is that the long reads can span the ***entire transcripts***, eliminating the need in the transcripts' assembly, which was the bottleneck of the short-reads technology.

Of course, long-reads also have their own limitations. Thus, for a long time, the long-reads sequencing was associated with high cost and low accuracy. However, over the recent years the cost of long reads sequencing was significantly reduced.

Also, it should be noted here, that the high accuracy of sequencing may not be necessary for RNA-seq expression measurements: two or three errors per hundred nucleotides do not complicate identifying the transcript; the length is much more important! Furthermore, he accuracy of long-reads sequencing has significantly improved over the recent years (as we will discuss in more detail later).



Discovering and measuring transcript isoforms with long-reads

Short-reads RNA-Seq

ANXA1 gene expression measurement
Sample A vs Sample B

… full-length RNA-sequencing … revealed a ~5-fold higher number of transcript isoforms than previously detected

Mays *et al* 2019

Tools for short reads are not good for long reads
Tools for long reads are fast developing
and often platform-specific

Long-reads RNA-Seq

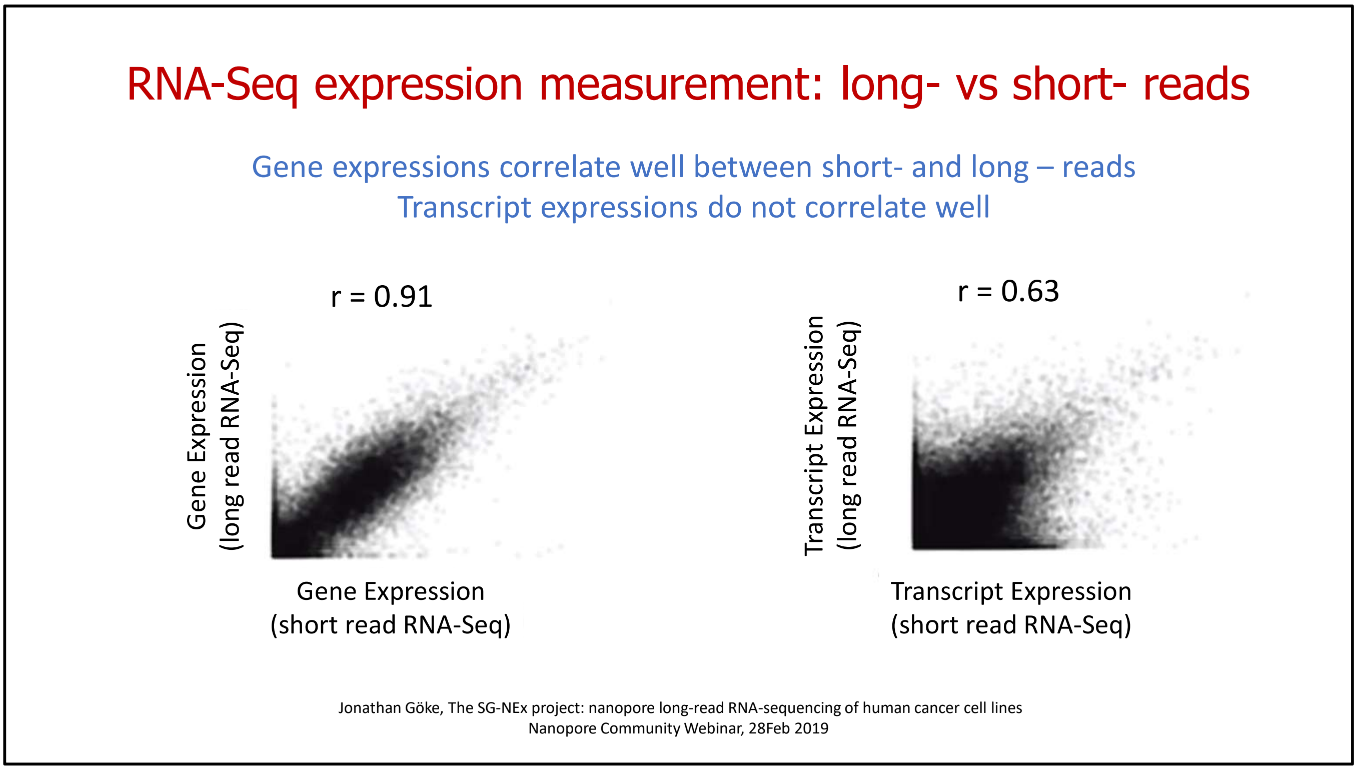
How important the low-abundant transcripts are ?

Mays et al 2019 Single-Molecule Real-Time (SMRT) Full-Length RNA-Sequencing Reveals Novel and Distinct mRNA Isoforms in Human Bone Marrow Cell Subpopulations Genes. 2019, 10, 253

Because of the ability to read the entire transcripts, the long-reads RNA-seq reveals several times more transcripts than it was previously detectable by the short reads.

Interestingly, many of the new detected transcripts are of low-abundancy. So, we still don't know their biologic relevance.

Another question is:
If so many transcripts were missed by the short reads, does it mean that all the previous results obtained with the short reads were wrong ?



RNA-Seq expression measurement: long- vs short- reads

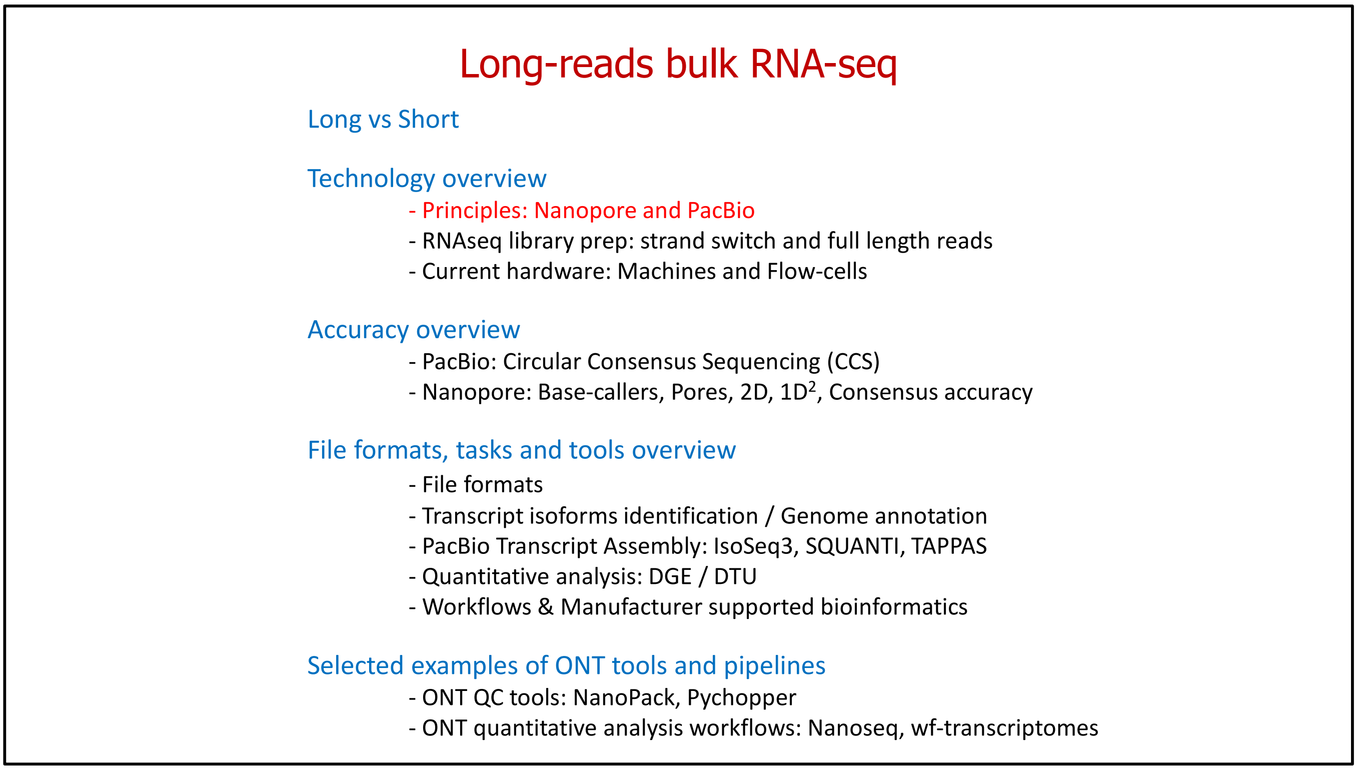
Gene expressions correlate well between short- and long – reads
Transcript expressions do not correlate well

Jonathan Göke, The SG-NEx project: nanopore long-read RNA-sequencing of human cancer cell lines
Nanopore Community Webinar, 28Feb 2019

No, it doesn’t.

On the left panel you can see that after aggregation per ***GENE*** the results of short-reads RNA-sequencing correlate well with the long-reads. Of course, the right panel shows much worse correlation at the ***TRANSCRIPT*** level.

The point is that most of the gene expression results reported from the short reads were reported per gene, not per transcript. So, most of the previously published short-reads results are valid. Of course, the rare short-reads *TRANSCRIPT*-specific data may need to be validated.



Long-reads bulk RNA-seq

Long vs Short

Technology overview

- Principles: Nanopore and PacBio
- RNAseq library prep: strand switch and full length reads
- Current hardware: Machines and Flow-cells

Accuracy overview

- PacBio: Circular Consensus Sequencing (CCS)
- Nanopore: Base-callers, Pores, 2D, 1D^2, Consensus accuracy

File formats, tasks and tools overview

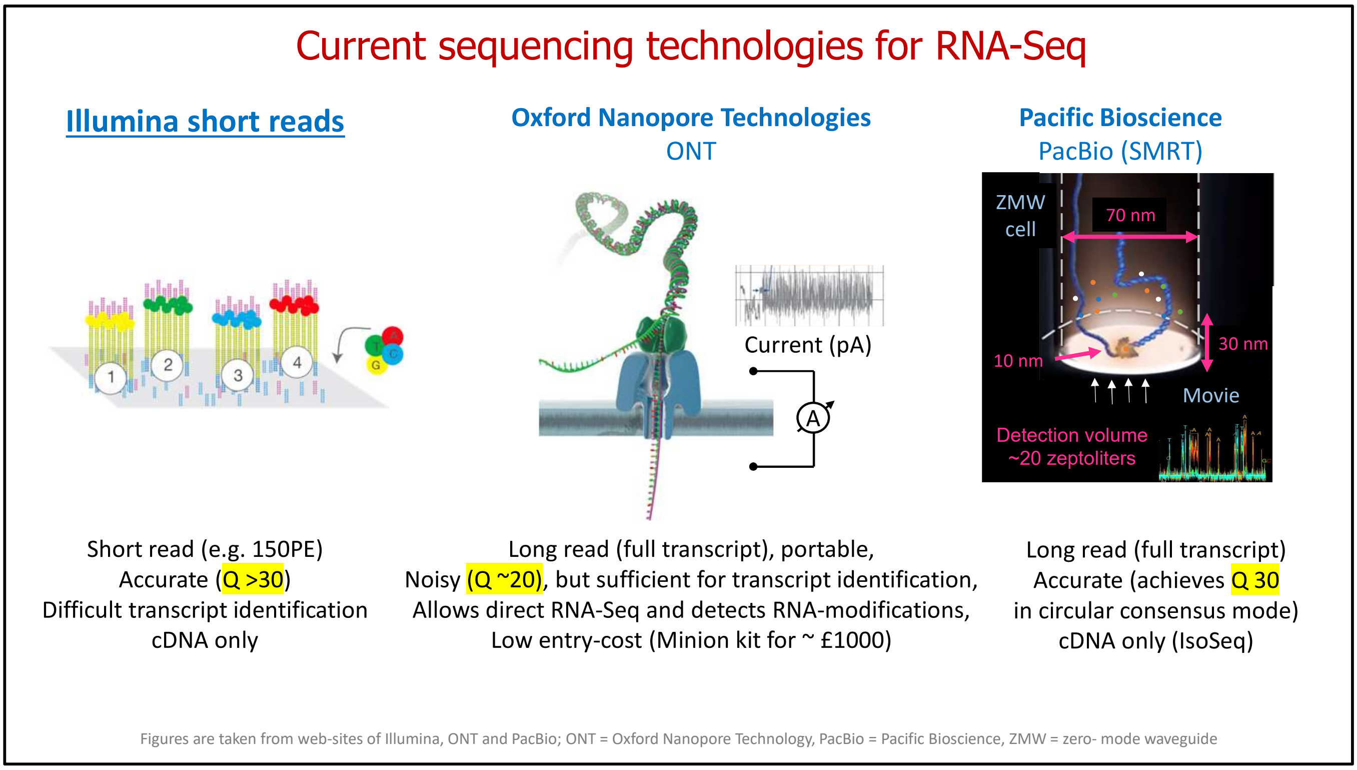
- File formats
- Transcript isoforms identification / Genome annotation
- PacBio Transcript Assembly: IsoSeq3, SQUANTI, TAPPAS
- Quantitative analysis: DGE / DTU
- Workflows & Manufacturer supported bioinformatics

Selected examples of ONT tools and pipelines

- ONT QC tools: NanoPack, Pychopper
- ONT quantitative analysis workflows: Nanoseq, wf-transcriptomes

Now, lets review how the long reads sequencing technologies work.
At the moment, there are two long-read sequencing technologies: ***Oxford Nanopore*** and ***Pacific Bioscience***.

We will not discuss ***Synthetic long reads*** here, except for a single slide to acknowledge the existence of such alternative to the true long-reads sequencing.



Current sequencing technologies for RNA-Seq

Illumina short reads

Oxford Nanopore Technologies
ONT

Pacific Bioscience
PacBio (SMRT)

Short read (e.g. 150PE)
Accurate (Q >30)
Difficult transcript identification
cDNA only

Long read (full transcript), portable,
Noisy (Q ~20), but sufficient for transcript identification,
Allows direct RNA-Seq and detects RNA-modifications,
Low entry-cost (Minion kit for ~ £1000)

Long read (full transcript)
Accurate (achieves Q 30
in circular consensus mode)
cDNA only (IsoSeq)

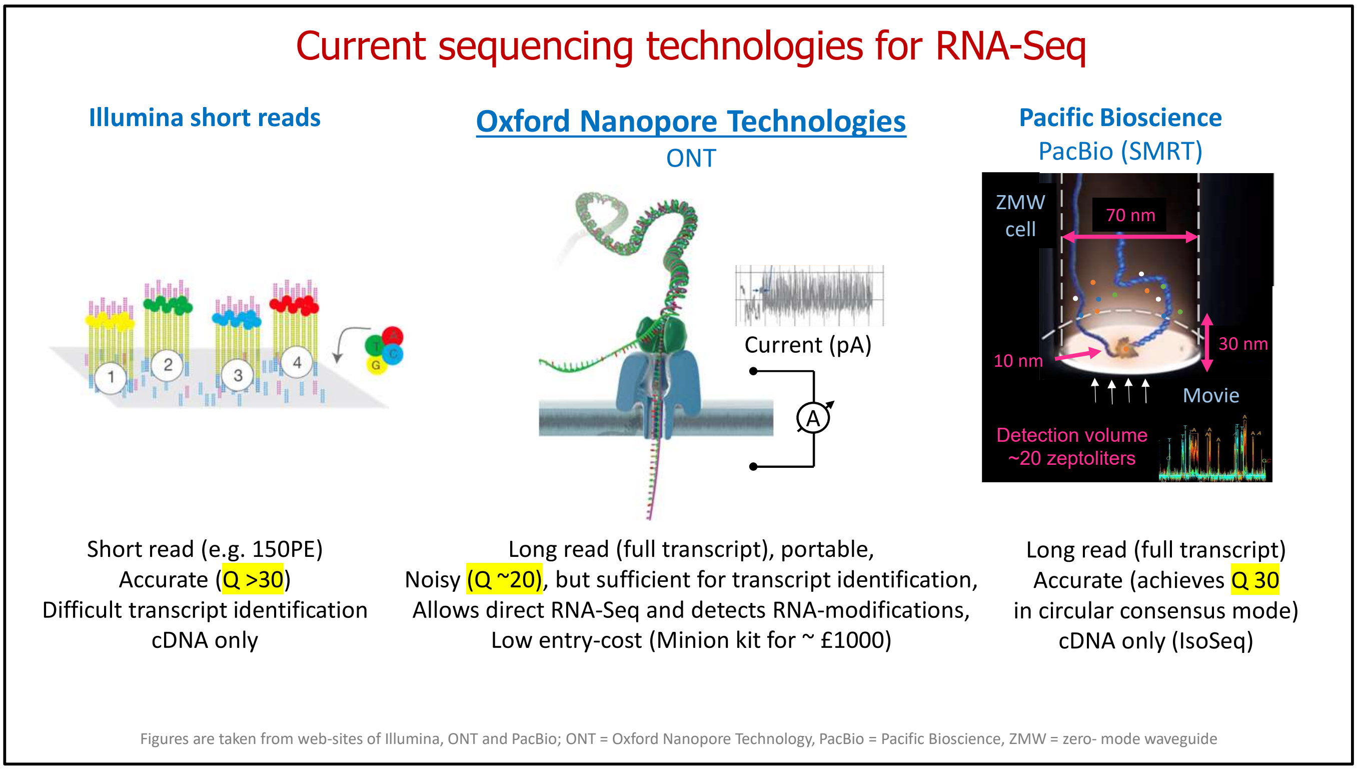
Figures are taken from web-sites of Illumina, ONT and PacBio; ONT = Oxford Nanopore Technology, PacBio = Pacific Bioscience, ZMW = zero- mode waveguide

I included the Illumina short-reads technology on this slide (on the left) just to explain why it can not produce longer reads.

The Illumuna sequencing is based on amplifying each DNA fragment to a cluster. Then it adds one nucleotide per cycle, with different nucleotides coded by different colors. Because each cluster includes hundreds of copies of the same initial sequence, the signal is strong and unambiguous. So, the probability of error is low (in other words, the accuracy is high).

Unfortunately, some individual DNA fragments in clusters occasionally miss a cycle, and after 2 or 3 hundred cycles the molecules in the clusters are going out of sync, preventing the reading of longer fragments.

The advantage of the Illumina technology is that, until the clusters go out of phase, the sequencing is very accurate: less than one error in a thousand bases.



Current sequencing technologies for RNA-Seq

Illumina short reads

1
2
3
4
A
C
T
G

Short read (e.g. 150PE)
Accurate (Q >30)
Difficult transcript identification
cDNA only

Oxford Nanopore Technologies
ONT

Current (pA)
A

Long read (full transcript), portable,
Noisy (Q ~20), but sufficient for transcript identification,
Allows direct RNA-Seq and detects RNA-modifications,
Low entry-cost (Minion kit for ~ £1000)

Pacific Bioscience
PacBio (SMRT)

Long read (full transcript)
Accurate (achieves Q 30
in circular consensus mode)
cDNA only (IsoSeq)

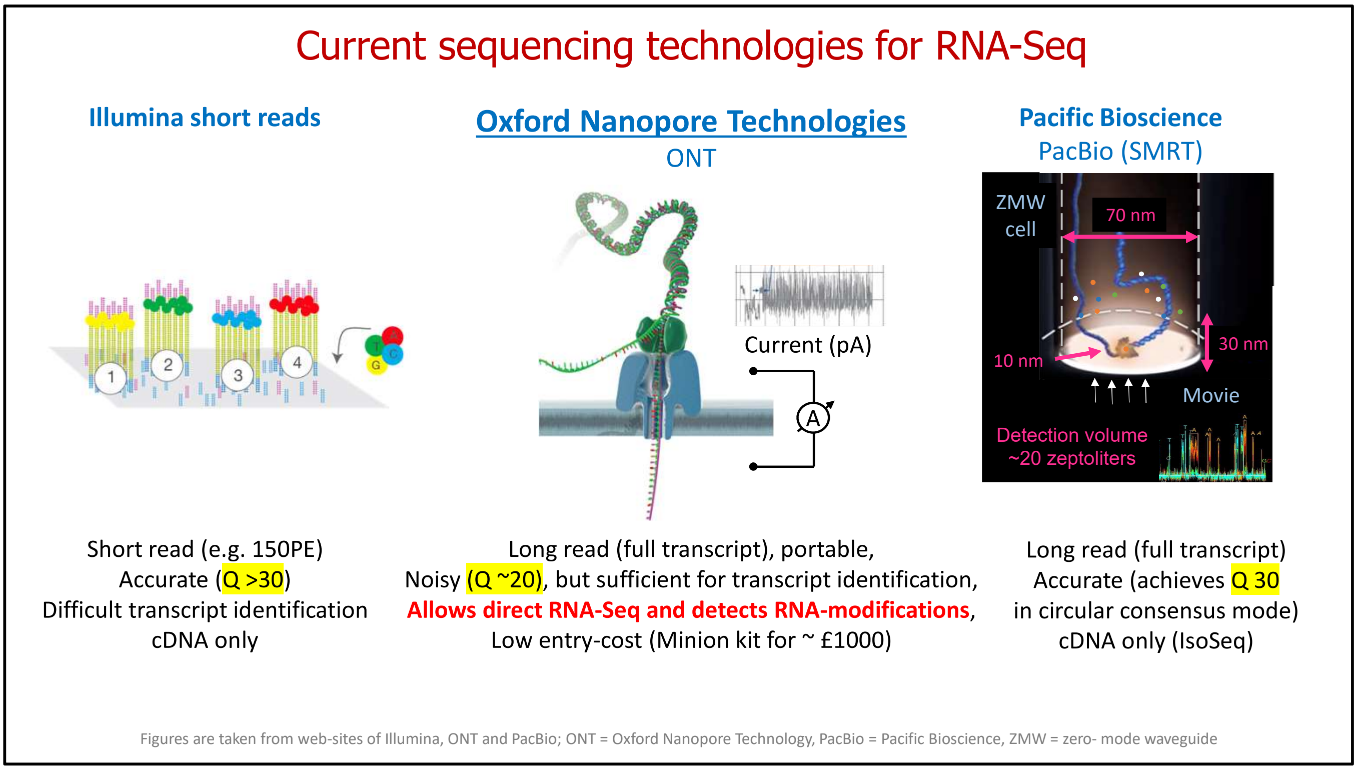
Figures are taken from web-sites of Illumina, ONT and PacBio; ONT = Oxford Nanopore Technology, PacBio = Pacific Bioscience, ZMW = zero- mode waveguide

The Nanopore technology is shown here in the middle.

It just passes DNA fragment through a pore, while measuring the ***ion current*** passing through the same pore at the same time. Because different nucleotides have different size and charge, they block the ions' passage in a different way, and so they have distinct ***current signatures***.

The length of Nanopore sequencing is limited just by the length of the fragment, reaching hundreds of thousand or even millions of bases.

The negative side of this technology is that it is less accurate than the short-reads sequencing: with some errors per each hundred of nucleotides. However, for many RNAseq applications such accuracy is absolutely sufficient.



Current sequencing technologies for RNA-Seq

Illumina short reads

Short read (e.g. 150PE)
Accurate (Q >30)
Difficult transcript identification
cDNA only

Oxford Nanopore Technologies
ONT

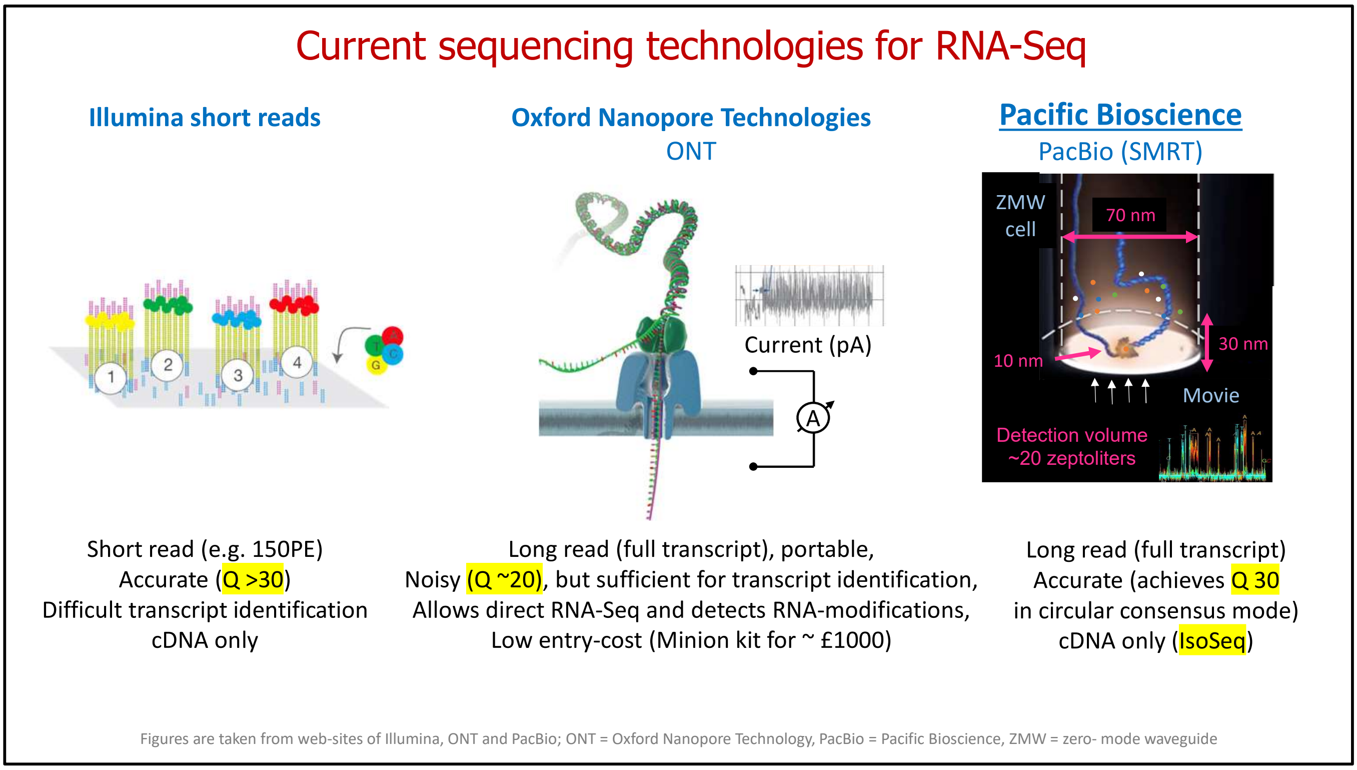
Long read (full transcript), portable,
Noisy (Q ~20), but sufficient for transcript identification,
Allows direct RNA-Seq and detects RNA-modifications,
Low entry-cost (Minion kit for ~ £1000)

Pacific Bioscience
PacBio (SMRT)

Long read (full transcript)
Accurate (achieves Q 30
in circular consensus mode)
cDNA only (IsoSeq)

Figures are taken from web-sites of Illumina, ONT and PacBio; ONT = Oxford Nanopore Technology, PacBio = Pacific Bioscience, ZMW = zero- mode waveguide

Importantly, Nanopore can sequence RNA directly, without converting it to cDNA before sequencing.



Current sequencing technologies for RNA-Seq

Illumina short reads

Short read (e.g. 150PE)
Accurate (Q >30)
Difficult transcript identification
cDNA only

Oxford Nanopore Technologies
ONT

Long read (full transcript), portable,
Noisy (Q ~20), but sufficient for transcript identification,
Allows direct RNA-Seq and detects RNA-modifications,
Low entry-cost (Minion kit for ~ £1000)

Pacific Bioscience
PacBio (SMRT)

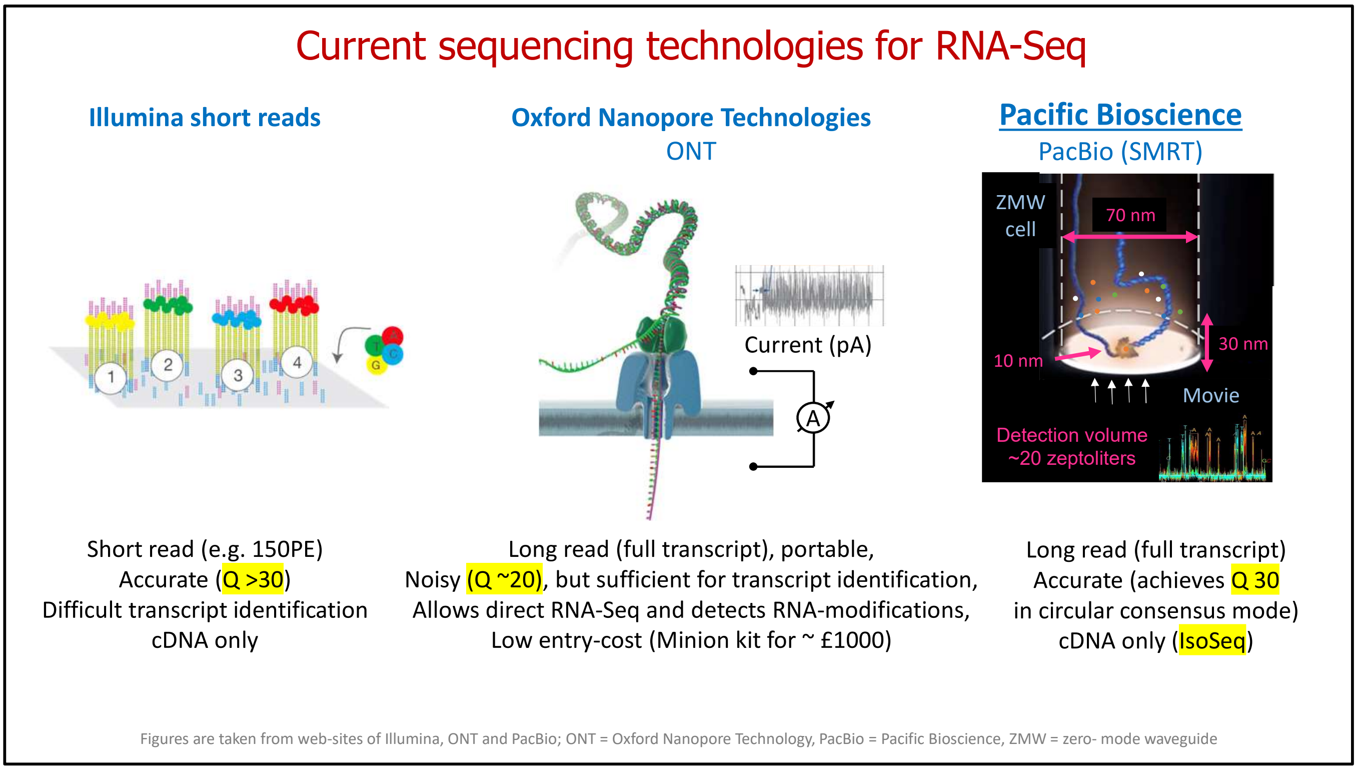
Long read (full transcript)
Accurate (achieves Q 30
in circular consensus mode)
cDNA only (IsoSeq)

Figures are taken from web-sites of Illumina, ONT and PacBio; ONT = Oxford Nanopore Technology, PacBio = Pacific Bioscience, ZMW = zero- mode waveguide

Finally, the PacBio technology is shown here on the right. PacBio is also marketed as SMRT sequencing, meaning that it reads Single Molecule in Real Time.

All what PacBio does is
- Place a single DNA-polymerase at the bottom of a tiny well
- Then film DNA synthesis by a tiny camera in real time ☺

When a new nucleotide is being added to the DNA, the DNA-polymerase retains it for a certain time, which is recorded as a peak in the movie. Different nucleotides are labelled by different colors.



Current sequencing technologies for RNA-Seq

Illumina short reads

Oxford Nanopore Technologies
ONT

Pacific Bioscience
PacBio (SMRT)

Short read (e.g. 150PE)
Accurate (Q >30)
Difficult transcript identification
cDNA only

Long read (full transcript), portable,
Noisy (Q ~20), but sufficient for transcript identification,
Allows direct RNA-Seq and detects RNA-modifications,
Low entry-cost (Minion kit for ~ £1000)

Long read (full transcript)
Accurate (achieves Q 30
in circular consensus mode)
cDNA only (IsoSeq)

Figures are taken from web-sites of Illumina, ONT and PacBio; ONT = Oxford Nanopore Technology, PacBio = Pacific Bioscience, ZMW = zero- mode waveguide

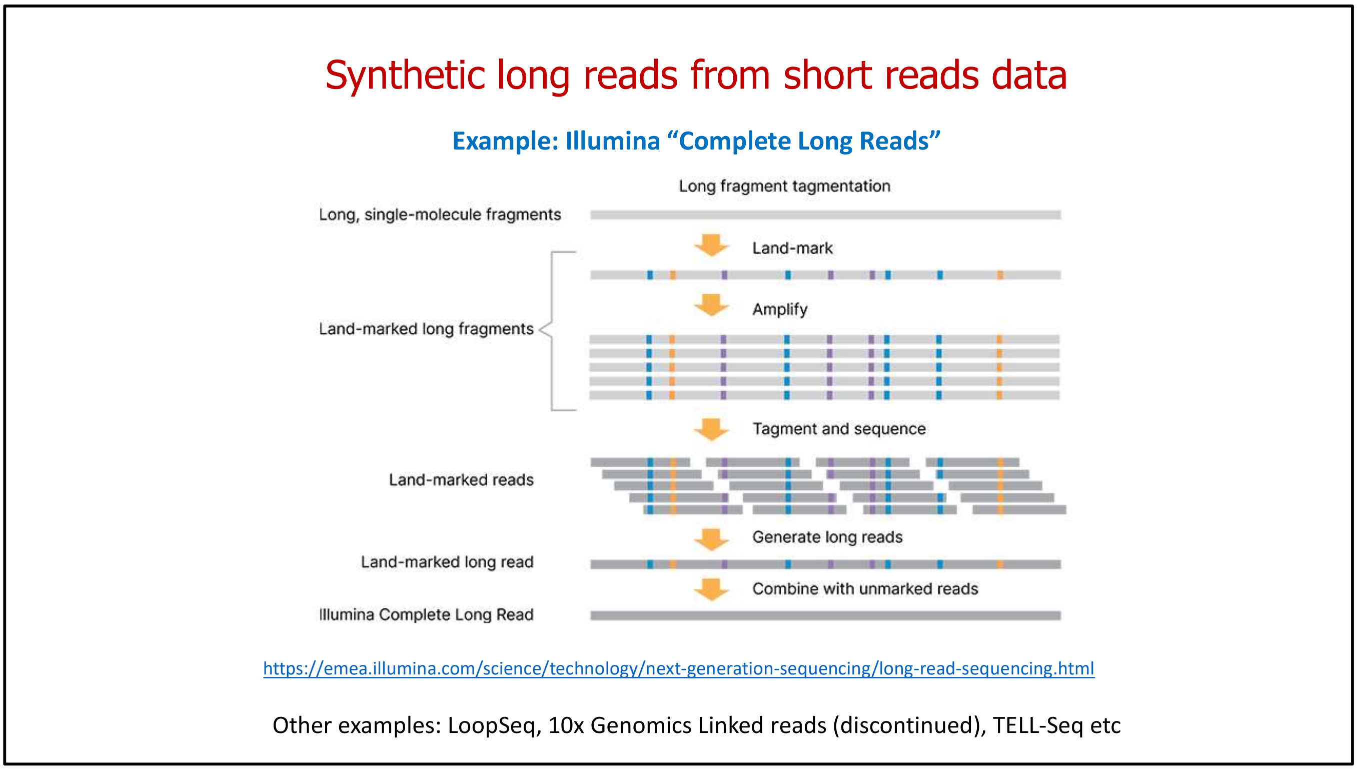
Sounds simple ?

However, to make such movie possible, it’s important to exclude filming the other nucleotides, which are not being retained by DNA-polymerase, but still are present in the solution!

This is made by illuminating only a ***very*** small volume around the DNA-polymerase. Because of the Brownian movement, the non-retained nucleotides quickly cross the volume, so they and are not registered in the movie.

Selective illumination of such a small volume is done by some physical miracle, called Zero-Mode Waveguide (ZMW). Somehow, when the diameter of the well is small enough (relatively to the light wavelength), the light does not go through the well, but only propagates to a certain depth. A well with about 70 nm in diameter allows to illuminate a volume of just 20 ***zepto***-liters.

Honestly, illuminating and filming a single DNA-polymerase molecule in action sounds like SciFi to me. However, somehow it works.



Synthetic long reads from short reads data

Example: Illumina "Complete Long Reads"

Long fragment tagmentation
Long, single-molecule fragments
Land-mark
Land-marked long fragments
Amplify
Tagment and sequence
Land-marked reads
Generate long reads
Land-marked long read
Combine with unmarked reads
Illumina Complete Long Read

https://emea.illumina.com/science/technology/next-generation-sequencing/long-read-sequencing.html

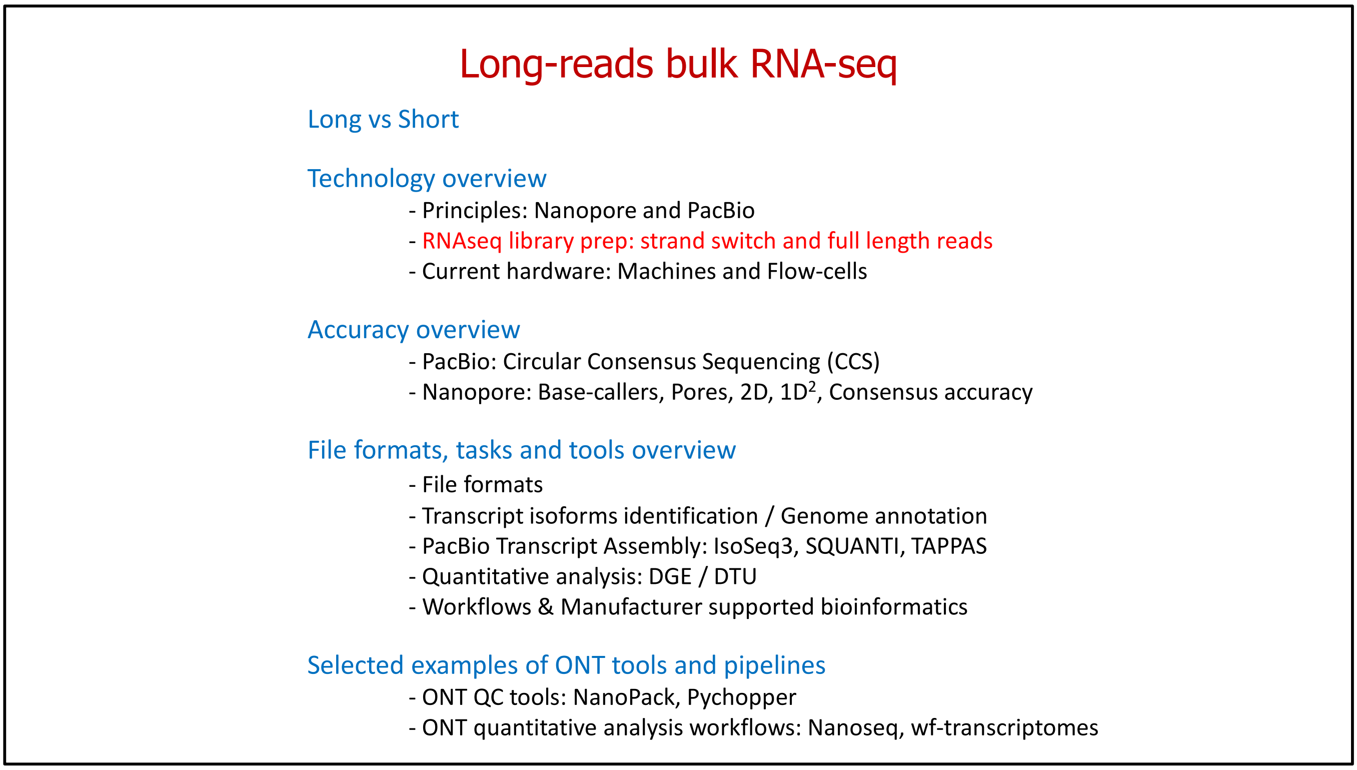
Other examples: LoopSeq, 10x Genomics Linked reads (discontinued), TELL-Seq etc

Finally, I would like to mention "assembled" or "synthetic" long reads. They are not actually long reads, but they use some ways to assemble individual long sequences from short reads.

For instance, Illumina developed a method, where special "land-marks" are placed to long DNA fragments before they are shredded and sequenced by short reads. Then the initial long DNA fragments can be computationally re-assembled from the "land-marks" information.

There are other similar technologies that allow to assemble synthetic long reads from the short reads data. Potentially, this could be a cheaper alternative to the true long-reads technologies. However, because the cost of true long reads sequencing is going down, it may not be necessary to look for more affordable (and more complicated) alternatives in future.

You may see a comparison between PacBio and synthetic long reads following this link: https://www.pacb.com/blog/the-hifi-difference-true-long-reads-vs-synthetic-long-reads/



Long-reads bulk RNA-seq

Long vs Short

Technology overview

- Principles: Nanopore and PacBio
- RNAseq library prep: strand switch and full length reads
- Current hardware: Machines and Flow-cells

Accuracy overview

- PacBio: Circular Consensus Sequencing (CCS)
- Nanopore: Base-callers, Pores, 2D, 1D^2, Consensus accuracy

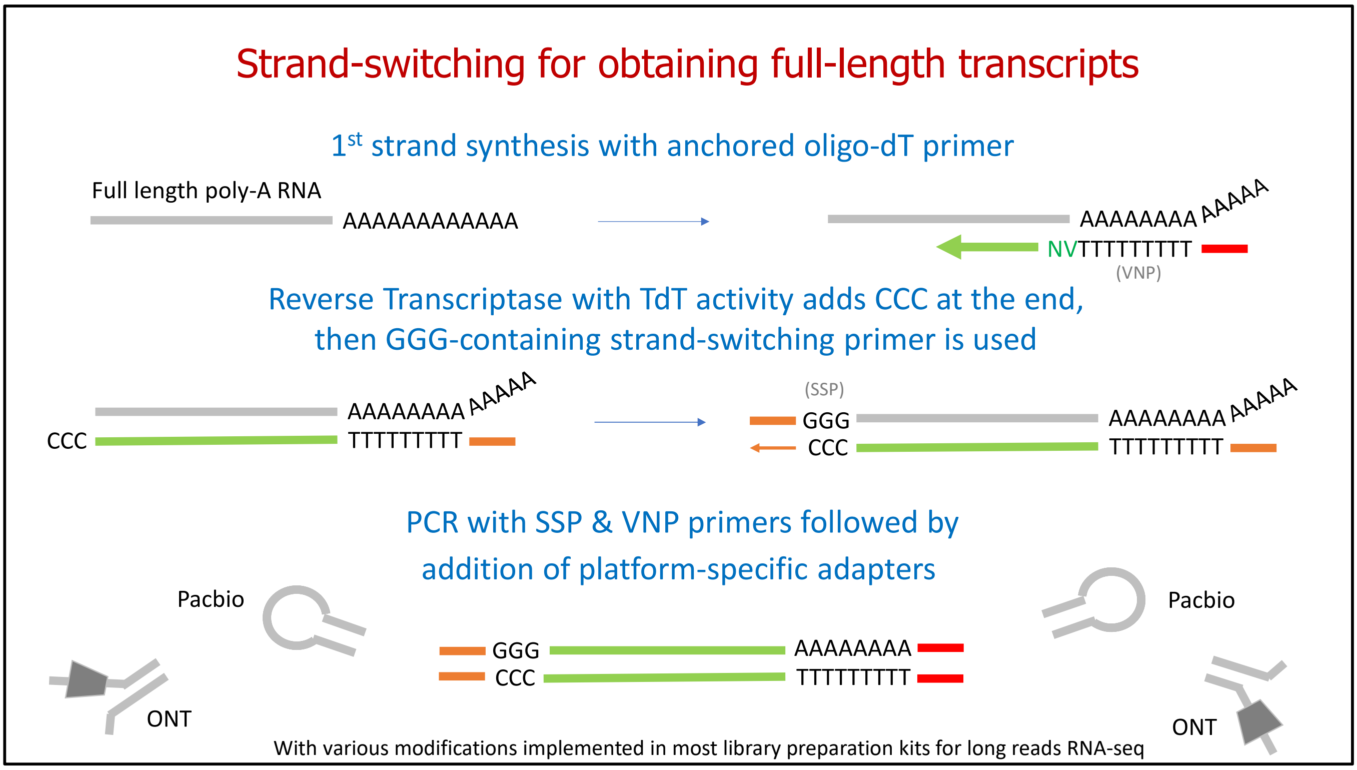
File formats, tasks and tools overview

- File formats
- Transcript isoforms identification / Genome annotation
- PacBio Transcript Assembly: IsoSeq3, SQUANTI, TAPPAS
- Quantitative analysis: DGE / DTU
- Workflows & Manufacturer supported bioinformatics

Selected examples of ONT tools and pipelines

- ONT QC tools: NanoPack, Pychopper
- ONT quantitative analysis workflows: Nanoseq, wf-transcriptomes

Now let’s talk a bit about long-reads RNA-seq library preparation.



Strand-switching for obtaining full-length transcripts

1st strand synthesis with anchored oligo-dT primer

Reverse Transcriptase with TdT activity adds CCC at the end, then GGG-containing strand-switching primer is used

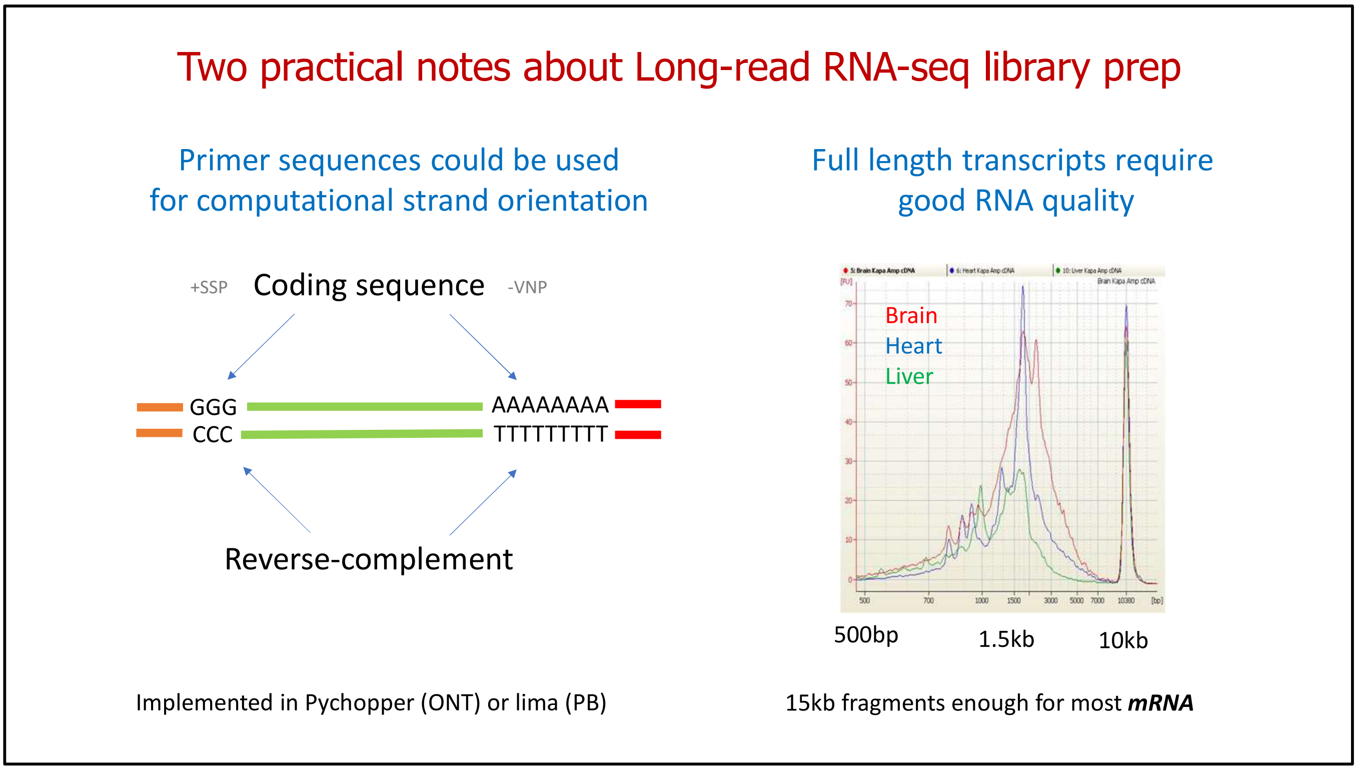
PCR with SSP & VNP primers followed by addition of platform-specific adapters

With various modifications implemented in most library preparation kits for long reads RNA-seq

This is a simplified scheme explaining the widely used technique for making cDNA libraries for long-reads sequencing.

An ***anchored*** oligo-dT primer is used to ensure annealing at the beginning of poly-A tail.
The ***reverse transcriptase with TdT activity*** is used for the 1st strand synthesis to add CCC motif at the end of the 1st strand.
Then a ***strand-switching primer*** is used for the next strand synthesis, ensuring that only the full-length transcripts are sequenced.
Finally, the technology-specific adapters are added (e.g. motor & tether for ONT, or hairpins for PacBio HiFi)
Of course, this procedure assumes non-degraded RNA in the first place.

VNP: V = A or G or C ; N = A or G or C or T ; P= Primer
TdT : Terminal deoxynucleotidyl Transferase
SSP: Strand Switching Primer (sometime called template-switching)



Two practical notes about Long-read RNA-seq library prep

Primer sequences could be used for computational strand orientation

+SSP
Coding sequence
-VNP
GGG
CCC
AAAAAAAA
TTTTTTTTT
Reverse-complement

Implemented in Pychopper (ONT) or lima (PB)

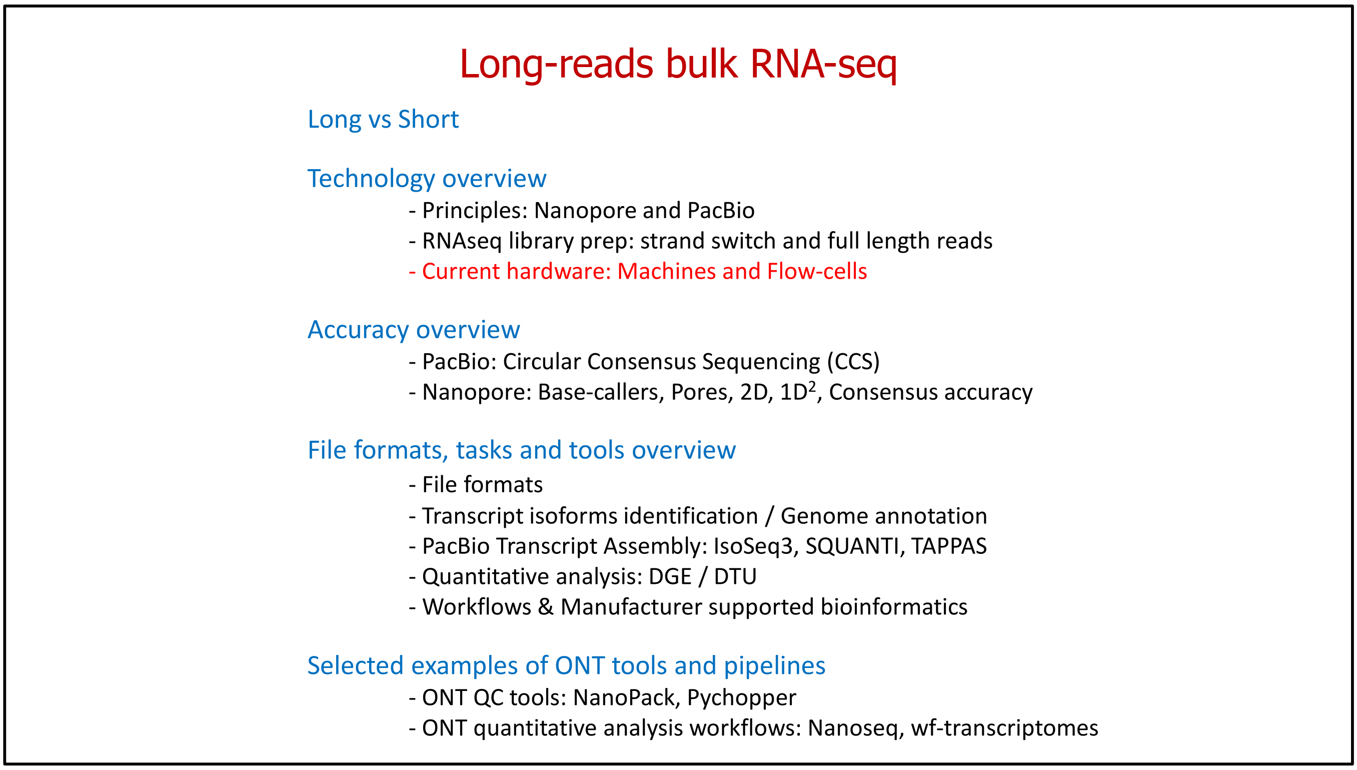
Full length transcripts require good RNA quality

5: Brain Kapa Amp cDNA
6: Heart Kapa Amp cDNA
10: Liver Kapa Amp cDNA
Brain Kapa Amp cDNA
[FU]
70
60
50
40
30
20
10
0
500
700
1000
1500
3000
5000
7000
10380
[bp]
Brain
Heart
Liver
500bp
1.5kb
10kb

15kb fragments enough for most ***mRNA***

An additional advantage of using the Strand-switching technique is that it allows computational detection of the strand orientation. So, all the long-reads libraries are always stranded: the coding sequence is flanked by ***direct*** SSP and ***reverse-complemented*** oligo-dT primers

The right panel illustrates non-degraded RNA extracted from different tissues, suggesting that ~15kb read length should be sufficient to sequence most of the human transcripts.



Long-reads bulk RNA-seq

Long vs Short

Technology overview

- Principles: Nanopore and PacBio
- RNAseq library prep: strand switch and full length reads
- Current hardware: Machines and Flow-cells

Accuracy overview

- PacBio: Circular Consensus Sequencing (CCS)
- Nanopore: Base-callers, Pores, 2D, 1D^2, Consensus accuracy

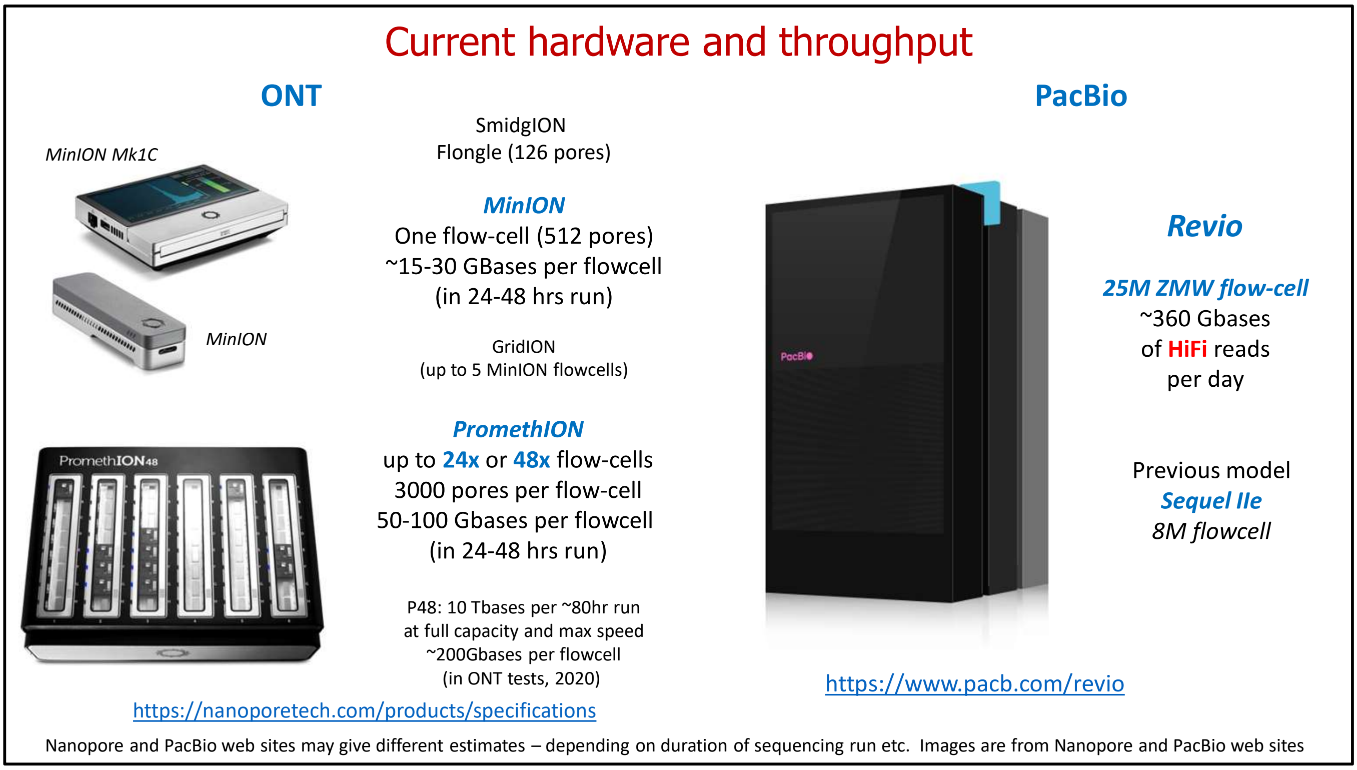
File formats, tasks and tools overview

- File formats
- Transcript isoforms identification / Genome annotation
- PacBio Transcript Assembly: IsoSeq3, SQUANTI, TAPPAS
- Quantitative analysis: DGE / DTU
- Workflows & Manufacturer supported bioinformatics

Selected examples of ONT tools and pipelines

- ONT QC tools: NanoPack, Pychopper
- ONT quantitative analysis workflows: Nanoseq, wf-transcriptomes

Then the prepared libraries go to the sequencing machines.



Current hardware and throughput

ONT

MinION Mk1C

MinION

SmidgION
Flongle (126 pores)

MinION
One flow-cell (512 pores)
~15-30 GBases per flowcell
(in 24-48 hrs run)

GridION
(up to 5 MinION flowcells)

PromethION
up to **24x** or **48x** flow-cells
3000 pores per flow-cell
50-100 Gbases per flowcell
(in 24-48 hrs run)

P48: 10 Tbases per ~80hr run
at full capacity and max speed
~200Gbases per flowcell
(in ONT tests, 2020)

https://nanoporetech.com/products/specifications

PacBio

Revio

25M ZMW flow-cell
~360 Gbases
of **HiFi** reads
per day

Previous model
Sequel IIe
8M flowcell

https://www.pacb.com/revio

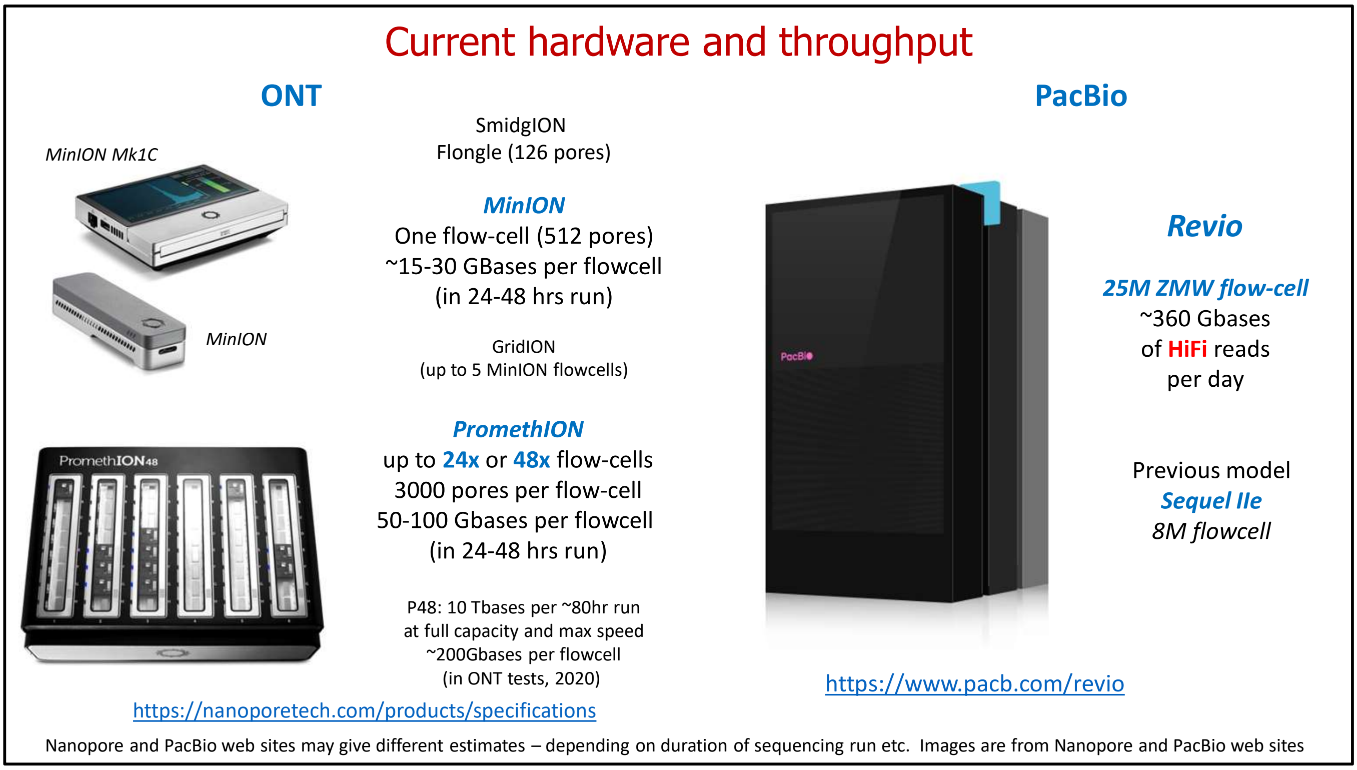
Nanopore and PacBio web sites may give different estimates – depending on duration of sequencing run etc. Images are from Nanopore and PacBio web sites

Nanopore sells many models of sequencing machines.

For a long time, the entry-level Nanopore model was **Minion**. Its size is about ~10x2x3 cm, it requires connection to a laptop by USB cable for saving and analyzing data, and its flowcell contains ~500 pores.

The highest end of Nanopore models is **Promethion**, it has much larger flowcells (with ~3000 pores). Thus, a Promethion flow-cell allows sequencing of a human genome with 50-75x coverage, potentially reaching >30 consensus accuracy (discussed later). Promethion may run up to 48 such flow-cells simultaneously.

There are intermediate and smaller models of Nanopore instruments.



Current hardware and throughput

ONT

MinION Mk1C

MinION

SmidgION
Flongle (126 pores)

MinION
One flow-cell (512 pores)
~15-30 GBases per flowcell
(in 24-48 hrs run)

GridION
(up to 5 MinION flowcells)

PromethION
up to **24x** or **48x** flow-cells
3000 pores per flow-cell
50-100 Gbases per flowcell
(in 24-48 hrs run)

P48: 10 Tbases per ~80hr run
at full capacity and max speed
~200Gbases per flowcell
(in ONT tests, 2020)

https://nanoporetech.com/products/specifications

PacBio

Revio

25M ZMW flow-cell
~360 Gbases
of **HiFi** reads
per day

Previous model
Sequel IIe
8M flowcell

https://www.pacb.com/revio

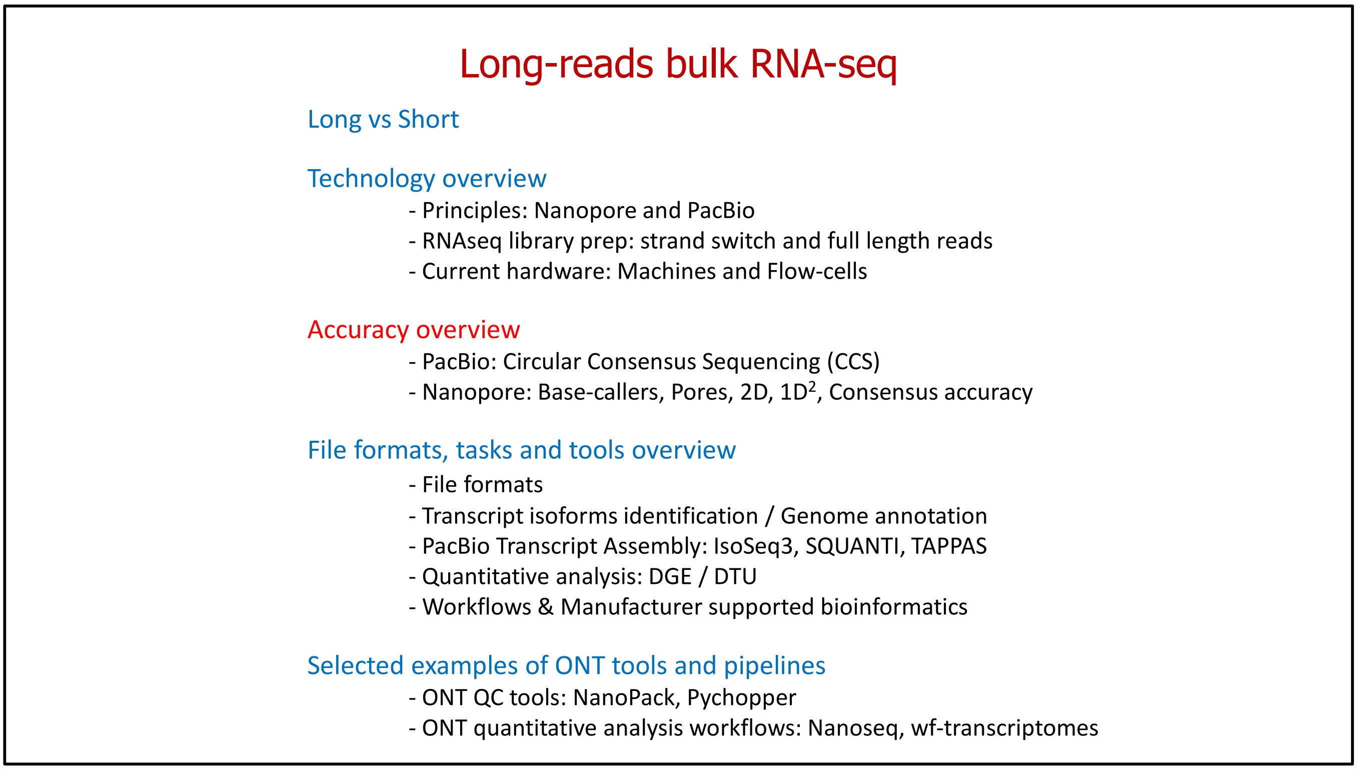
Nanopore and PacBio web sites may give different estimates – depending on duration of sequencing run etc. Images are from Nanopore and PacBio web sites

The "latest and greatest" model of PacBio is **Revio**.

Introduced in 2023, it was a large step forward against the previous models.

Importantly PacBio can produce so called "HiFi" reads – their accuracy could easily exceed the Illumina accuracy (discussed later).

Although PacBio is still more expensive than Nanopore per base, **Revio** already allows the sequencing of a human genome for USD1000 (with long accurate phased reads!). Combined with **Kinnex** library preparation (a new technique discussed later) **Revio** may make PacBio RNA-seq more affordable.



Long-reads bulk RNA-seq

Long vs Short

Technology overview

- Principles: Nanopore and PacBio
- RNAseq library prep: strand switch and full length reads
- Current hardware: Machines and Flow-cells

Accuracy overview

- PacBio: Circular Consensus Sequencing (CCS)
- Nanopore: Base-callers, Pores, 2D, $1D^2$, Consensus accuracy

File formats, tasks and tools overview

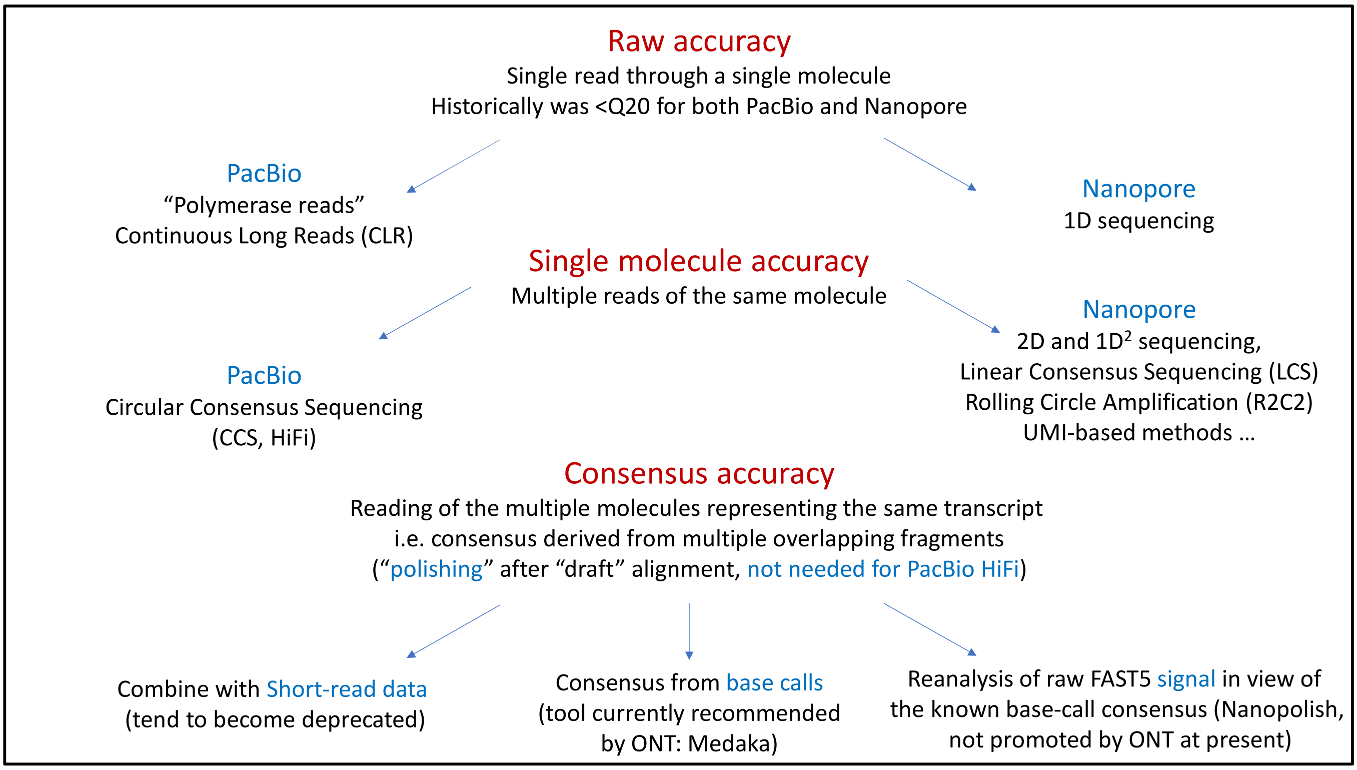
- File formats
- Transcript isoforms identification / Genome annotation
- PacBio Transcript Assembly: IsoSeq3, SQUANTI, TAPPAS
- Quantitative analysis: DGE / DTU
- Workflows & Manufacturer supported bioinformatics

Selected examples of ONT tools and pipelines

- ONT QC tools: NanoPack, Pychopper
- ONT quantitative analysis workflows: Nanoseq, wf-transcriptomes

I have already mentioned previously “consensus accuracy” or “HiFi reads”.

These terms relate to the ways how long reads sequencing technologies improve their accuracy.



Raw accuracy

Single read through a single molecule

Historically was <Q20 for both PacBio and Nanopore

PacBio

"Polymerase reads"

Continuous Long Reads (CLR)

Nanopore

1D sequencing

Single molecule accuracy

Multiple reads of the same molecule

PacBio

Circular Consensus Sequencing

(CCS, HiFi)

Nanopore

2D and $1D^2$ sequencing,

Linear Consensus Sequencing (LCS)

Rolling Circle Amplification (R2C2)

UMI-based methods …

Consensus accuracy

Reading of the multiple molecules representing the same transcript

i.e. consensus derived from multiple overlapping fragments

("polishing" after "draft" alignment, not needed for PacBio HiFi)

Combine with Short-read data

(tend to become deprecated)

Consensus from base calls

(tool currently recommended by ONT: Medaka)

Reanalysis of raw FAST5 signal in view of the known base-call consensus (Nanopolish, not promoted by ONT at present)

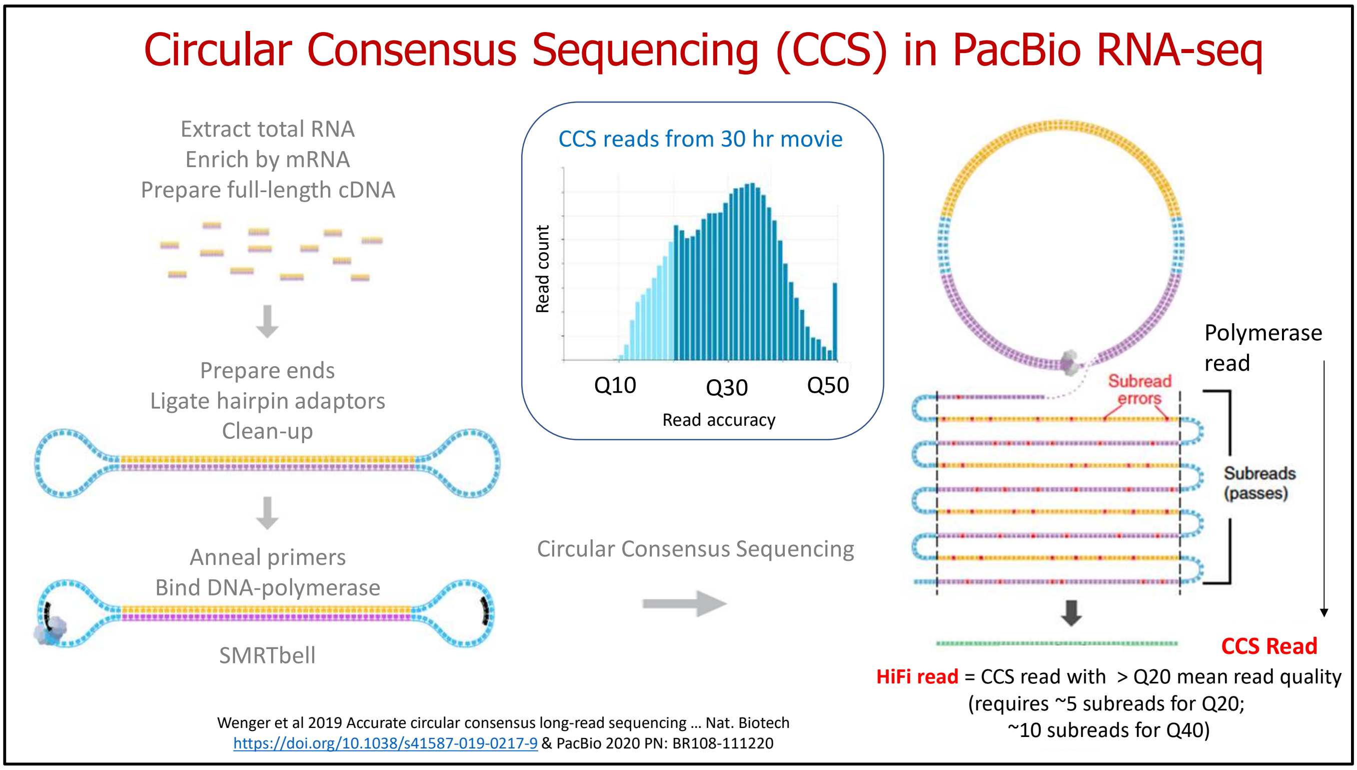
Typical ***raw*** Illumina base quality is Q30-40 (which is less than 1 error per 1000 bases). The ***raw*** long-reads sequencing is significantly less accurate (<20 for both PB and ONT).

This slide contains many new terms. I will explain some of them now, and some later. For PacBio the ***raw*** data may be called "Polymerase reads" (or "Continuous Long Reads", CLR). For Nanopore the ***raw*** data may be called "1D" sequencing.

One of the ways to improve accuracy of the long-reads was to read the same sequence several times.

This solved the accuracy problem for PacBio: when a DNA fragment is circularized and red many times. This technique is called "Circular Consensus Reads" (CCS), and it will be illustrated on the next slide. CSS accuracy easily exceeds the Illumina base quality if the same sequence is red 5 times or more.

Nanopore also tried reading the same molecule twice (called "2D" and "1D2"). Unfortunately, because of the non-random distribution of Nanopore errors, this was less successful than in PacBio. So, Nanopore still uses consensus accuracy: reading multiple fragments from the same gene instead of reading the same fragment. With sufficient depth (and price) Nanopore claims that it may match Illumina accuracy. However, it's still looks like an ambitious task for Nanopore (in 2025).



Circular Consensus Sequencing (CCS) in PacBio RNA-seq

Extract total RNA
Enrich by mRNA
Prepare full-length cDNA

Prepare ends
Ligate hairpin adaptors
Clean-up

Anneal primers
Bind DNA-polymerase

SMRTbell

CCS reads from 30 hr movie

Circular Consensus Sequencing

CCS Read

HiFi read = CCS read with > Q20 mean read quality
(requires ~5 subreads for Q20;
~10 subreads for Q40)

Wenger et al 2019 Accurate circular consensus long-read sequencing … Nat. Biotech
https://doi.org/10.1038/s41587-019-0217-9 & PacBio 2020 PN: BR108-111220

This slide explains the PacBio's ***Circular Consensus Sequencing*** (CCS).

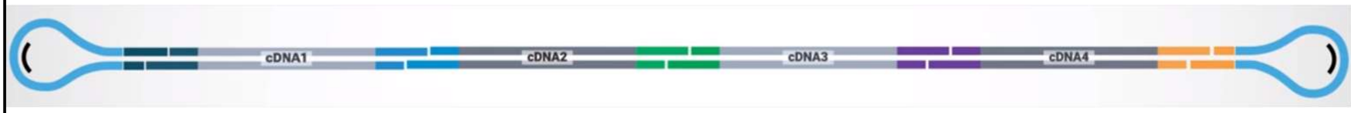
Because PacBio ***raw*** data (so called ***Polymerase Reads*** or ***Continuous Long Reads***) have low accuracy (<20), PacBio came up with a nice trick to improve the accuracy. During the library preparation they make the fragment circular. Then, during the sequencing the same fragment is red again and again many times. Because the errors are random, the consensus sequence after multiple reads becomes as accurate as the Illumina short reads (or even more accurate). The length of the circular consensus reads in PabBio may easily achieve 10-15 kilobases, which is enough for most of human full-length RNAs.

The CCS reads with average base quality >20 are called "HiFi" reads

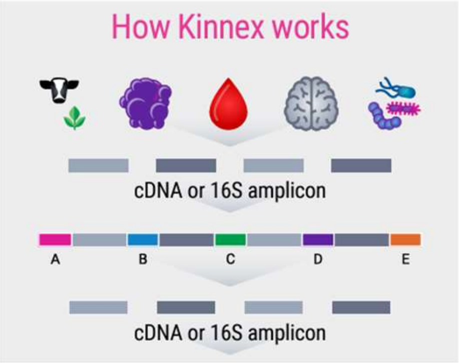
In the recent machines, PacBio implements CCS workflow directly on sequencer, by default outputting only CCS reads, and discarding the sub-reads information to reduce the data size.

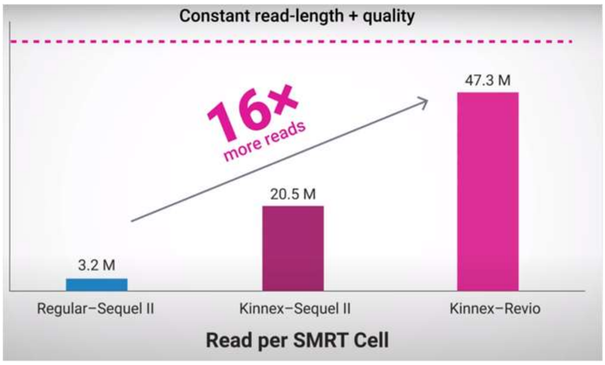
Kinnex technology in PacBio RNA-seq

A new library prep method than can significantly increase yield and decrease price of PB RNA-seq



Kinnex (formerly MAS-seq) concatenates several cDNAs into one array before HiFi sequencing



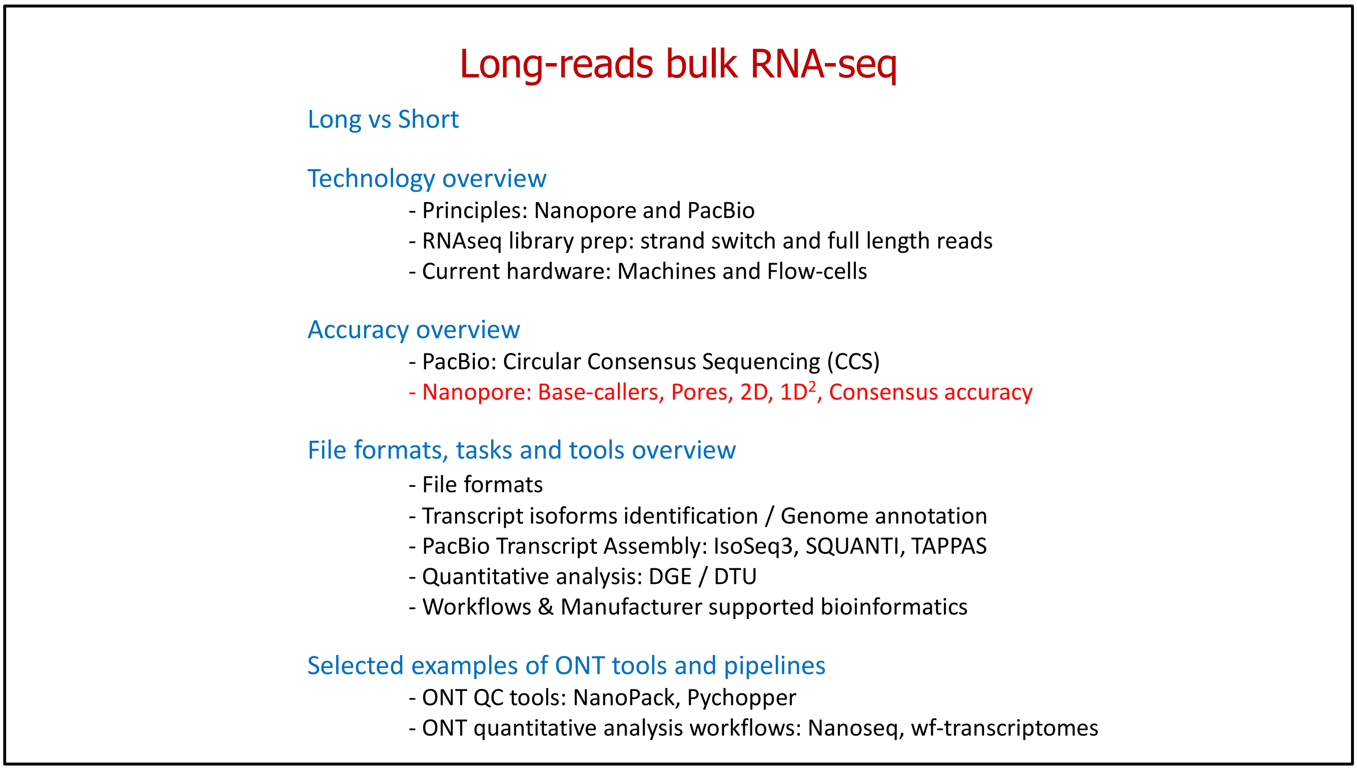


https://www.pacb.com/wp-content/uploads/Kinnex-brochure.pdf

https://www.youtube.com/watch?v=NICUp8C6rms

Kinnex (formerly MAS-seq) is a new PacBio’s library preparation method that concatenates several cDNAs into one array before using it for CCS.
This techniques can be used for cDNA or for amplicon sequencing (e.g. in metagenomics).

Combined with the high output of the Revio sequencer, Kinnex increases PacBio yield by 16 folds, paving the way for more affordable PacBio RNA-seq.



Long-reads bulk RNA-seq

Long vs Short

Technology overview

- Principles: Nanopore and PacBio
- RNAseq library prep: strand switch and full length reads
- Current hardware: Machines and Flow-cells

Accuracy overview

- PacBio: Circular Consensus Sequencing (CCS)
- Nanopore: Base-callers, Pores, 2D, $1D^2$, Consensus accuracy

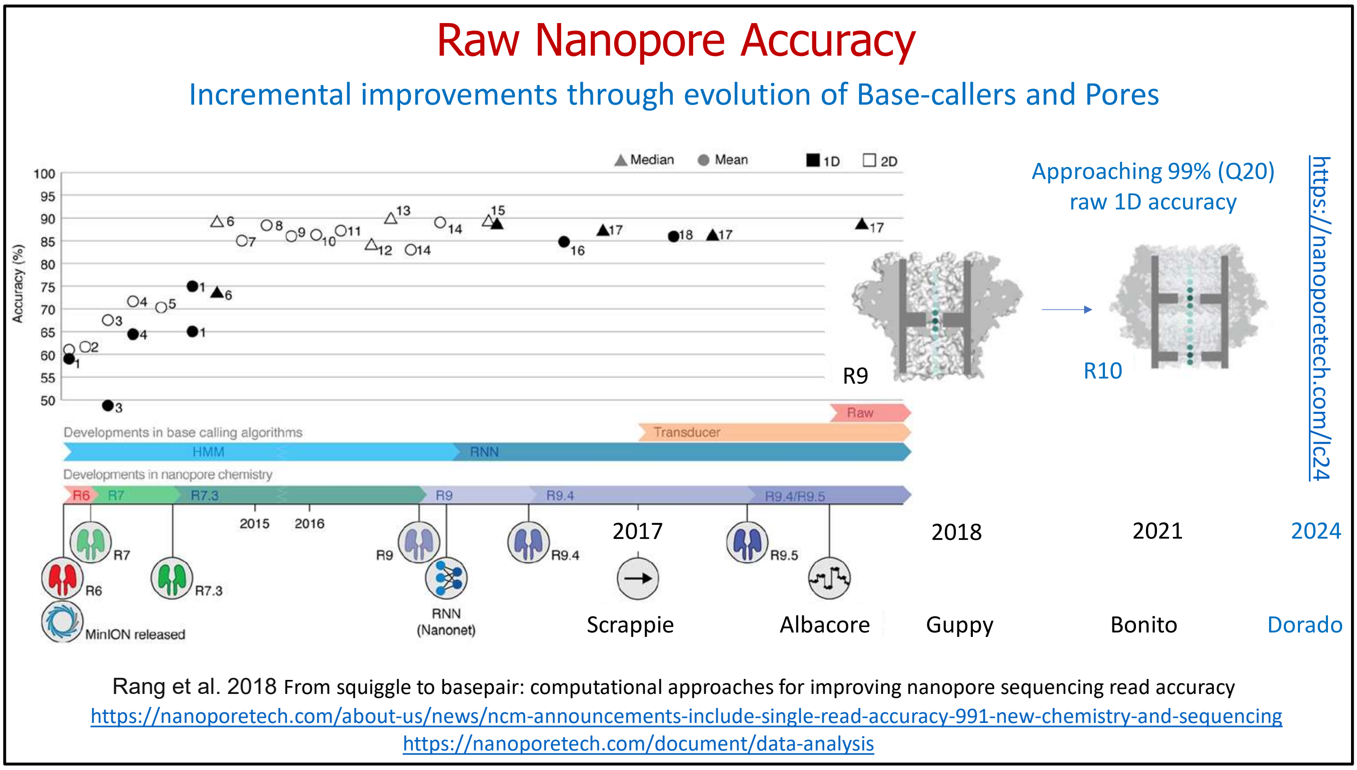
File formats, tasks and tools overview

- File formats
- Transcript isoforms identification / Genome annotation
- PacBio Transcript Assembly: IsoSeq3, SQUANTI, TAPPAS
- Quantitative analysis: DGE / DTU
- Workflows & Manufacturer supported bioinformatics

Selected examples of ONT tools and pipelines

- ONT QC tools: NanoPack, Pychopper
- ONT quantitative analysis workflows: Nanoseq, wf-transcriptomes

Now, let’s talk about the accuracy of Nanopore sequencing.



Raw Nanopore Accuracy

Incremental improvements through evolution of Base-callers and Pores

Rang et al. 2018 From squiggle to basepair: computational approaches for improving nanopore sequencing read accuracy
https://nanoporetech.com/about-us/news/ncm-announcements-include-single-read-accuracy-991-new-chemistry-and-sequencing
https://nanoporetech.com/document/data-analysis

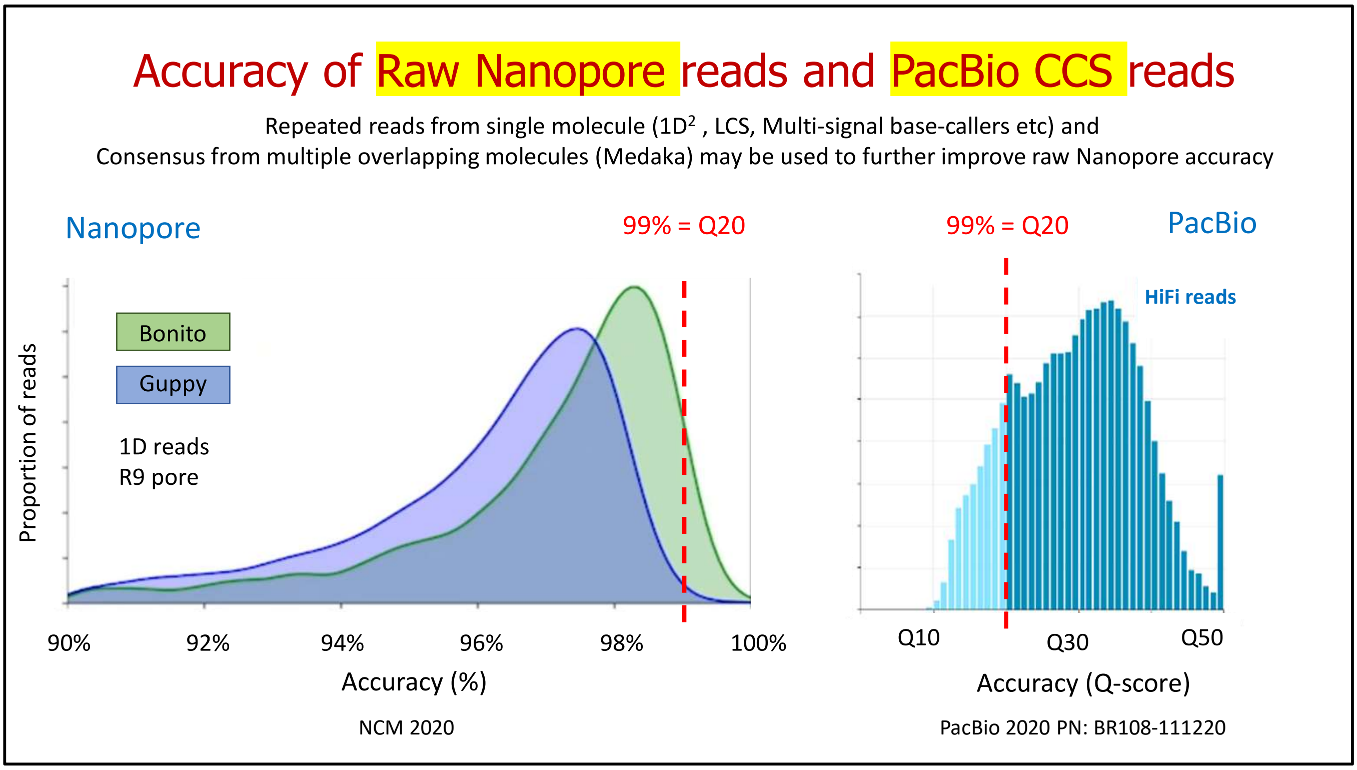
Once upon a time (about ten years ago :) the accuracy of Nanopore reads was awful: 50% errors.

However, it was dramatically improved since that by developing of the new pores, and most importantly, by the new algorithms for calling bases from ***squiggles*** (it’s a term to describe the ionic current fluctuations recorded by the pore).

The breakthrough in the ***basecalling*** algorithms development happened when Nanopore decided to use machine-learning for this.

Practically, this means that the training sets and even some hyperparameters (such as depth of the network etc) may vary even between different models of the same ***basecaller***. For instance, model trained on human DNA may not be sub-optimal for bacteria or plants …

A couple of years ago, Nanopore announced that the latest at the time version of basecaller ***Bonito*** with data from ***R10 pores*** approached ***99% raw accuracy.*** This is a remarkable progress comparing with initial 50% accuracy just 10 year ago!



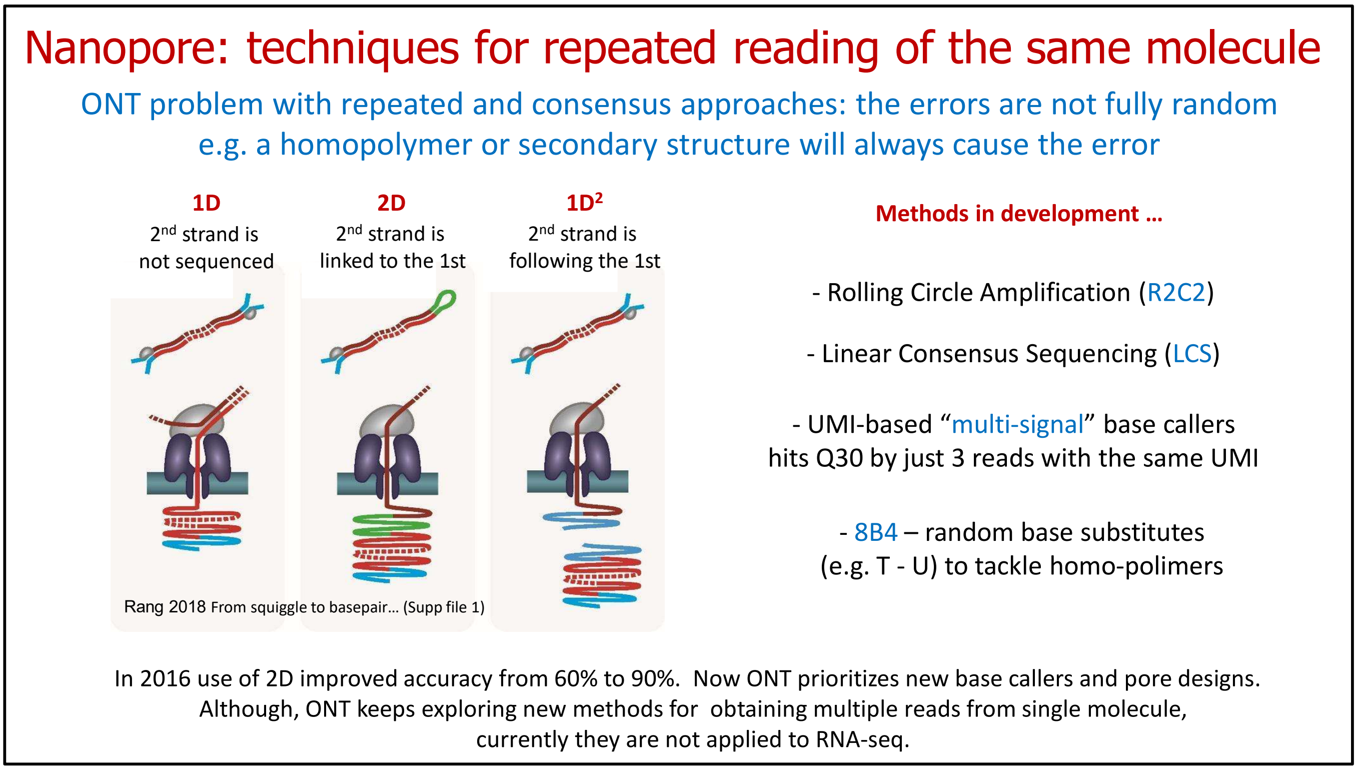
Accuracy of Raw Nanopore reads and PacBio CCS reads

Repeated reads from single molecule (1D^2 , LCS, Multi-signal base-callers etc) and
Consensus from multiple overlapping molecules (Medaka) may be used to further improve raw Nanopore accuracy

NCM 2020

PacBio 2020 PN: BR108-111220

Still, after development of PacBio CCS technology (in about 2020) even the latest Nanopore’s basecallers and pores are far behind the PacBio ***HiFi*** reads quality.



Nanopore: techniques for repeated reading of the same molecule

ONT problem with repeated and consensus approaches: the errors are not fully random
e.g. a homopolymer or secondary structure will always cause the error

1D
2nd strand is not sequenced

2D
2nd strand is linked to the 1st

1D^2
2nd strand is following the 1st

Rang 2018 From squiggle to basepair… (Supp file 1)

Methods in development …

- Rolling Circle Amplification (R2C2)

- Linear Consensus Sequencing (LCS)

- UMI-based "multi-signal" base callers
hits Q30 by just 3 reads with the same UMI

- 8B4 – random base substitutes
(e.g. T - U) to tackle homo-polimers

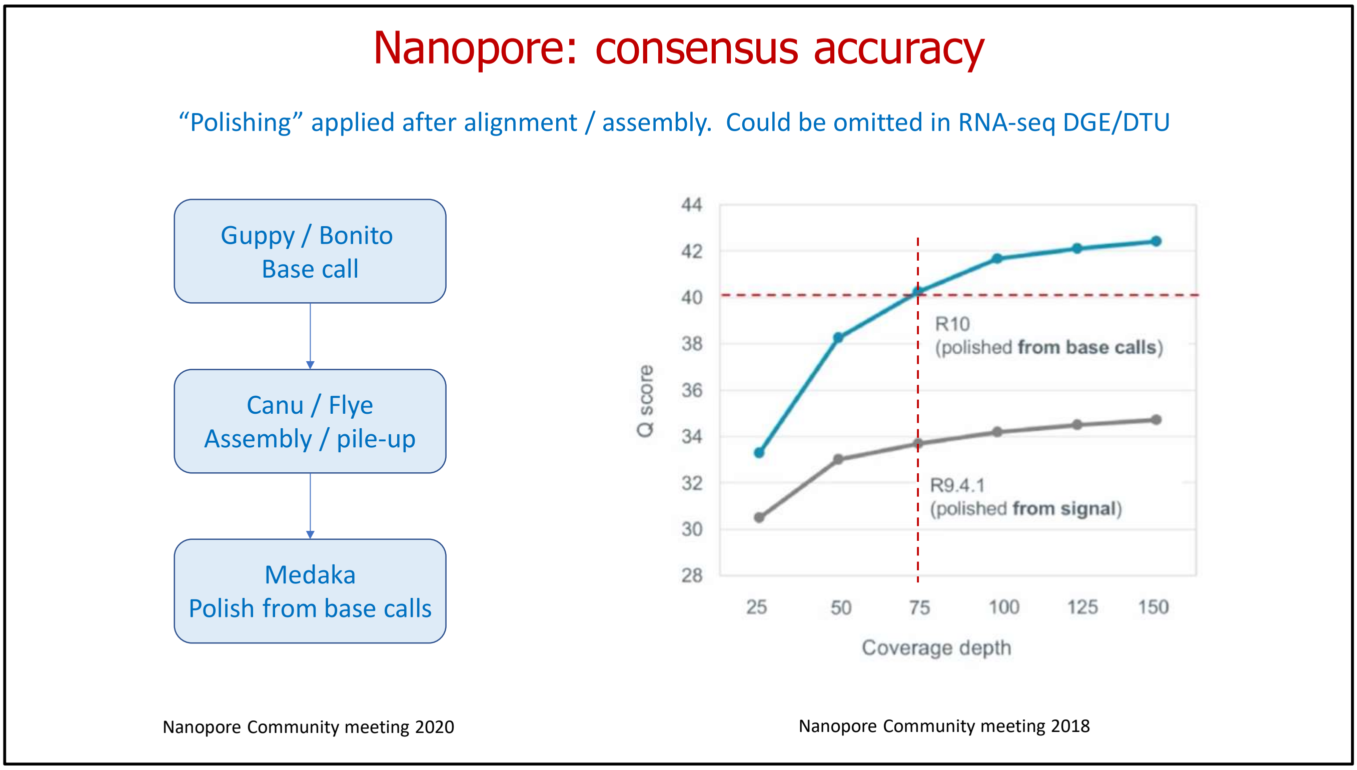
In 2016 use of 2D improved accuracy from 60% to 90%. Now ONT prioritizes new base callers and pore designs.
Although, ONT keeps exploring new methods for obtaining multiple reads from single molecule,
currently they are not applied to RNA-seq.

Nanopore also experimented with the multiple reads from single molecule:

- "2D" libraries sequenced the same molecule twice by adding a hairpin
- "1D^2" libraries did it even without the hairpin because the 2nd strand is still in vicinity of the pore, in ~75% cases it may follow the first strand even w/o hairpin.

These methods gave a modest improvement, but nowhere near the CCS reads in Pacbio. So, their development looks abandoned at the moment.

Hower, Nanopore constantly experiments with new methods. If you want to hear about the greatest and latest Nanopore achievements, I would strongly recommend you watch the presentations on Nanopore's annual London Calling conference (this year it's 20-23 May):
https://nanoporetech.com/about/events/conferences/lc25
Usually, the online access is free.



Nanopore: consensus accuracy

“Polishing” applied after alignment / assembly. Could be omitted in RNA-seq DGE/DTU

Guppy / Bonito
Base call

Canu / Flye
Assembly / pile-up

Medaka
Polish from base calls

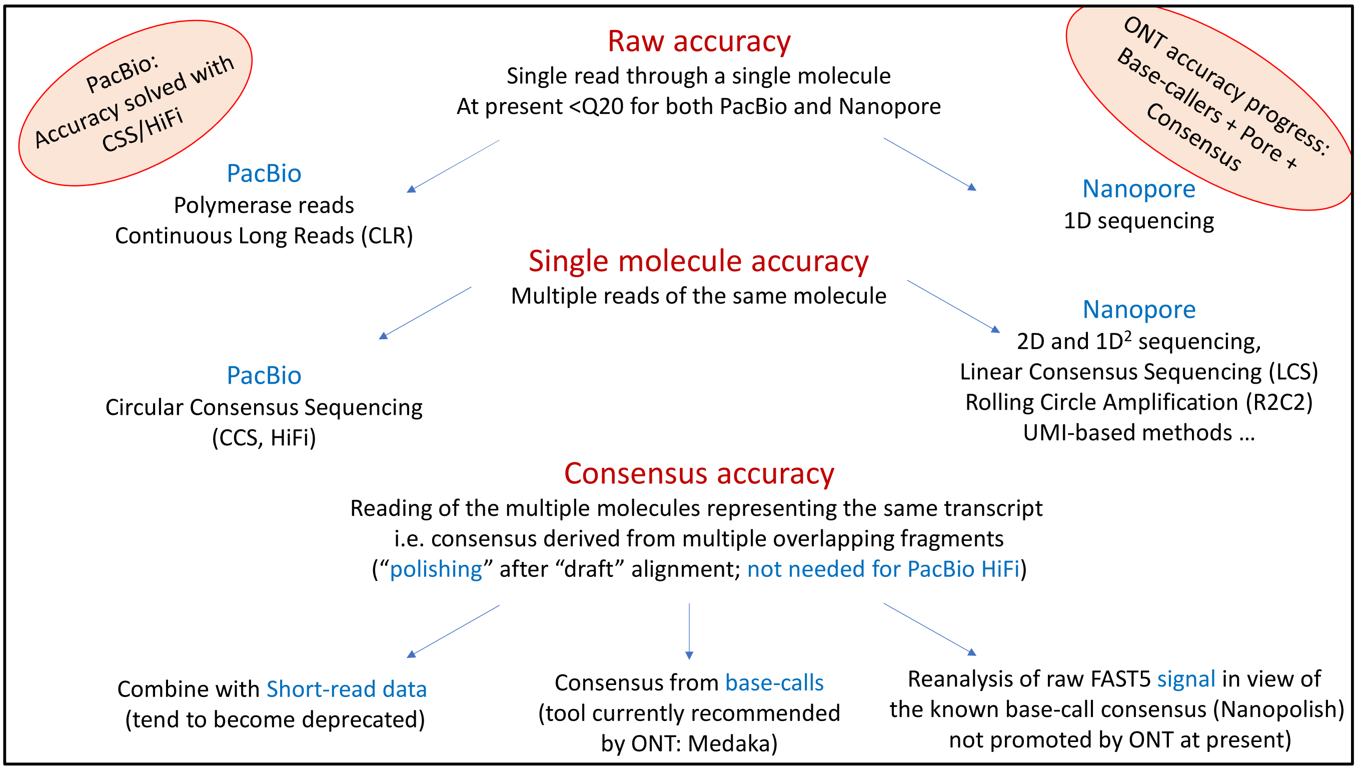
Nanopore Community meeting 2020

Nanopore Community meeting 2018

Along with the 2D, 1D^2 and other techniques aimed at multiple reads from the same fragment, Nanopore also develop data analysis tools to improve accuracy by ***consensus*** between multiple fragments originated from the same genomic region. In some contexts, it works. Thus, Nanopore made marketing clams that with sufficient depth of sequencing (e.g. 75x) Nanopore may reach ***consensus base quality*** in 30-40 range.

The main issue complicating Nanopore basecalling is that the errors are not random. If some sequence produces a squiggle that is hard to decipher at the first pass, it is also hard at the second pass.

Interestingly, most Nanopore errors are in homopolymers (repeats of the same nucleotide), which is not a common pattern in coding sequences and mRNA (except for the poly-A tail). Also, as I mentioned previously, the very high accuracy may not be required for some RNA-seq tasks: a couple of errors per 100 bases are not relevant for a transcript identification.



PacBio: Accuracy solved with CSS/HiFi

Raw accuracy

Single read through a single molecule
At present <Q20 for both PacBio and Nanopore

ONT accuracy progress: Base-callers + Pore + Consensus

PacBio
Polymerase reads
Continuous Long Reads (CLR)

Nanopore
1D sequencing

Single molecule accuracy

Multiple reads of the same molecule

PacBio
Circular Consensus Sequencing
(CCS, HiFi)

Nanopore
2D and $1D^2$ sequencing,
Linear Consensus Sequencing (LCS)
Rolling Circle Amplification (R2C2)
UMI-based methods …

Consensus accuracy

Reading of the multiple molecules representing the same transcript
i.e. consensus derived from multiple overlapping fragments
("polishing" after "draft" alignment; not needed for PacBio HiFi)

Combine with Short-read data
(tend to become deprecated)

Consensus from base-calls
(tool currently recommended
by ONT: Medaka)

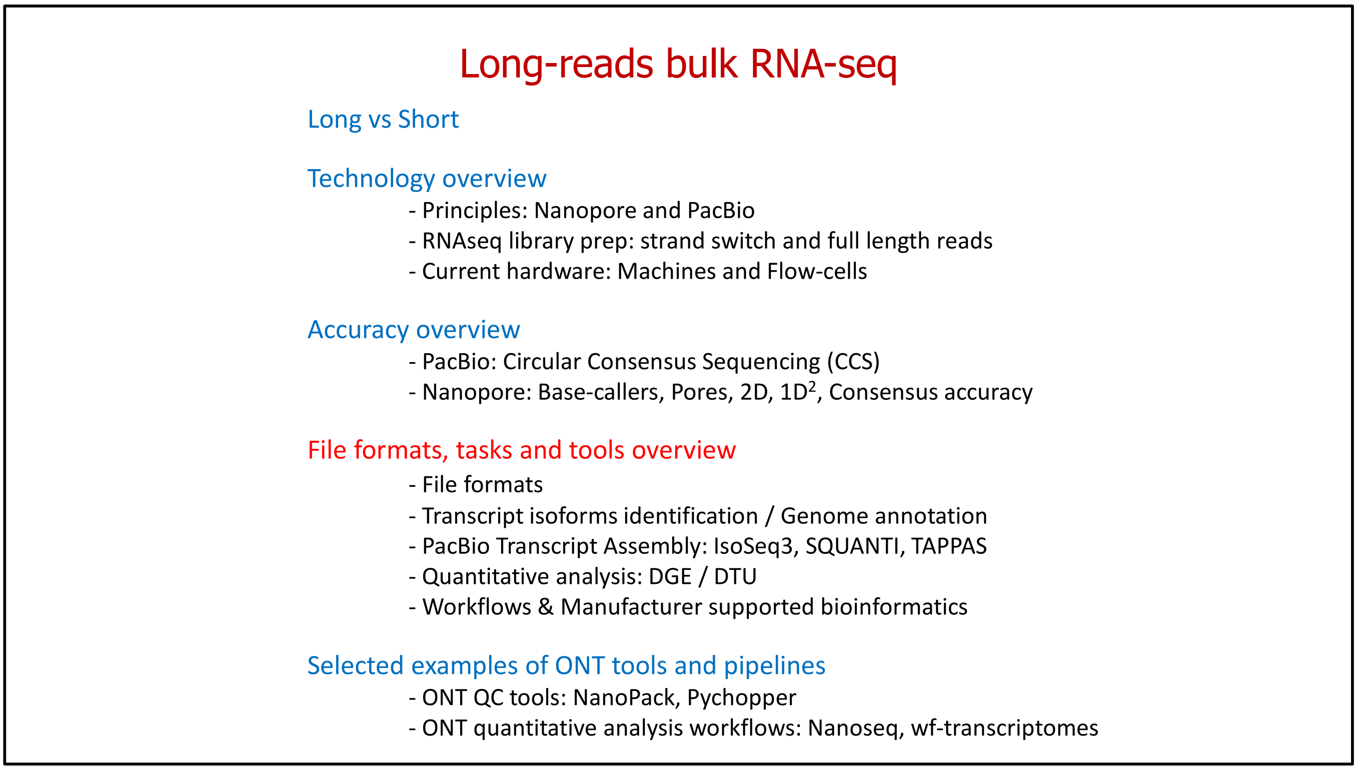
Reanalysis of raw FAST5 signal in view of
the known base-call consensus (Nanopolish)
not promoted by ONT at present)

To summarize, The ***raw*** reads accuracy is low for both PacBio and Nanopore.

PacBio solved the accuracy problem by Circular Consensus Sequencing (***CCS***), easily reaching Q40 and above for reads of 15kb.

Nanopore achieved quality Q15-20 by new ***basecallers and pores design***. This accuracy is enough for many RNAseq applications.

Importantly, Nanopore is still cheaper, which allows to get higher ***depth*** of sequencing for the same price, which may be beneficial for transcripts quantification and differential expression studies, while PacBio could be better for accurate transcript isoforms discovery (Pardo-Palacios et al 2023, https://www.biorxiv.org/content/10.1101/2023.07.25.550582v1)



Long-reads bulk RNA-seq

Long vs Short

Technology overview

- Principles: Nanopore and PacBio
- RNAseq library prep: strand switch and full length reads
- Current hardware: Machines and Flow-cells

Accuracy overview

- PacBio: Circular Consensus Sequencing (CCS)
- Nanopore: Base-callers, Pores, 2D, $1D^2$, Consensus accuracy

File formats, tasks and tools overview

- File formats
- Transcript isoforms identification / Genome annotation
- PacBio Transcript Assembly: IsoSeq3, SQUANTI, TAPPAS
- Quantitative analysis: DGE / DTU
- Workflows & Manufacturer supported bioinformatics

Selected examples of ONT tools and pipelines

- ONT QC tools: NanoPack, Pychopper
- ONT quantitative analysis workflows: Nanoseq, wf-transcriptomes

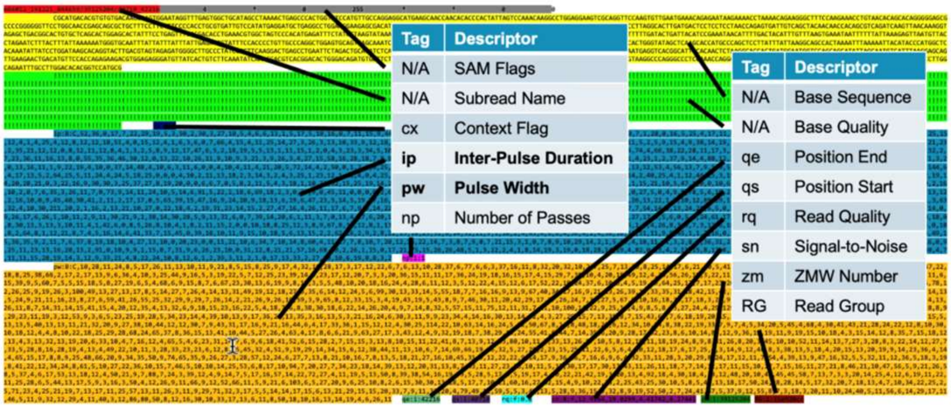
Now, let’s focus on bioinformatics, starting from file formats.

PacBio CCS file formats : BAM-s everywhere

- PacBio machine produces raw data as a BAM of **unaligned subreads** (with all base qual=0 & all read qual=0.8) *
- The CCS workflow produces a BAM with **unaligned consensus reads** with meaningful base & read qualities
- Alignment (mapping) programs will produce **aligned BAM** files that *retain PacBio tags*

* the latest versions of sequencers (starting from Sequel IIe) output the consensus BAM-s to reduce data size

Additional tags in PacBio BAM-s



Tag	Descriptor
N/A	SAM Flags
N/A	Subread Name
cx	Context Flag
ip	**Inter-Pulse Duration**
pw	**Pulse Width**
np	Number of Passes

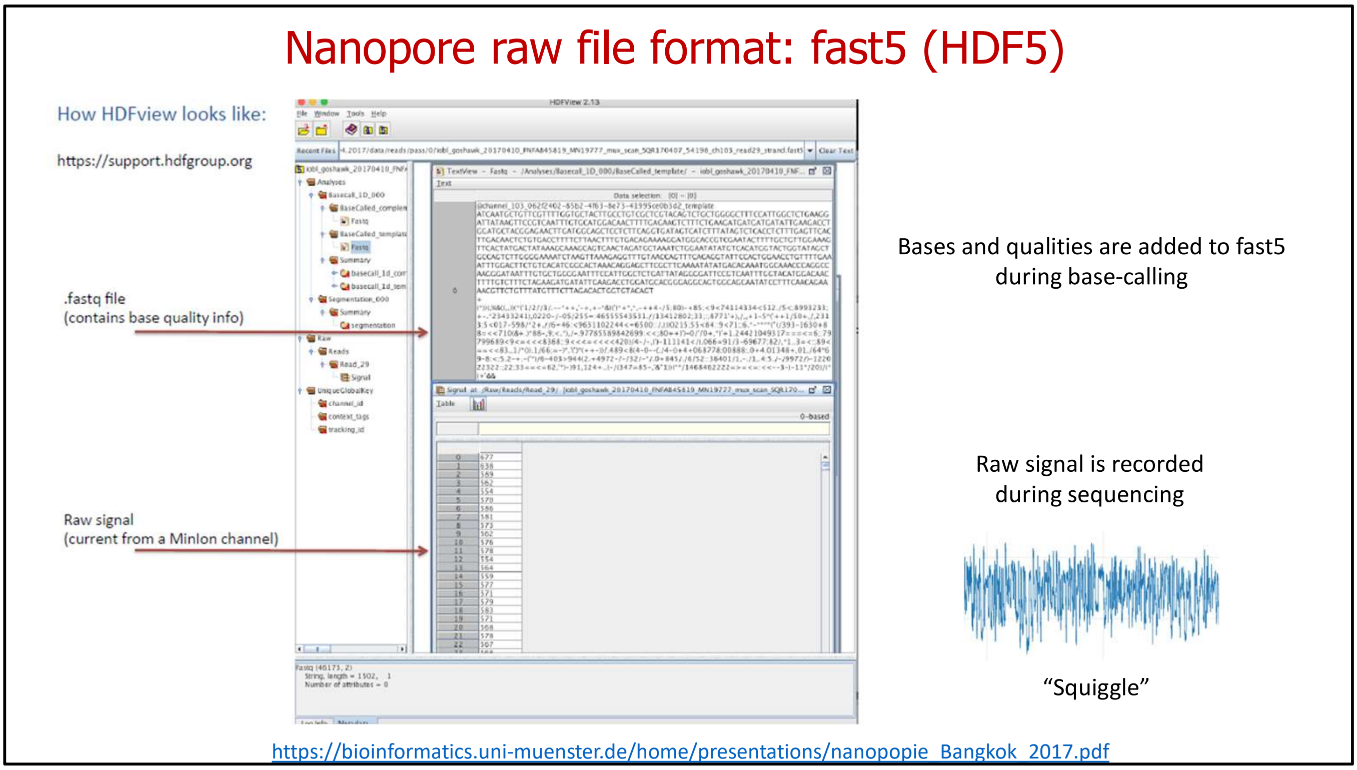
Tag	Descriptor
N/A	Base Sequence
N/A	Base Quality
qe	Position End
qs	Position Start
rq	Read Quality
sn	Signal-to-Noise
zm	ZMW Number
RG	Read Group

PacBio Webinar: PacBio Data Deep Dive: A Closer Look at HiFi Sequencing, 24 March 2021
https://pacbiofileformats.readthedocs.io/en/12.0/index.html

From the beginning, PacBio did not use FASTQ file format for its raw data. In addition to the base sequence and base quality PacBio needs to record the kinetic information (pulse width, inter-pulse duration) and some other information about their movies. The FASTQ file format was not good for recording all this data. So, PacBio developed their own versions of BAM files for raw data:
- Unaligned BAM for subreads (with nominal base qualities), then
- Unaligned BAM for consensus reads (with meaningful base quality information), then
- BAM file for aligned consensus reads (with CIGAR string etc)

Along with the BAM files, PacBio also uses an alternative file format (a special sort of XML) described here:
https://pacbiofileformats.readthedocs.io/en/13.1/DataSet.html

You may see more details about different PacBio file formats here:
https://pacbiofileformats.readthedocs.io/



Nanopore raw file format: fast5 (HDF5)

How HDFview looks like:

https://support.hdfgroup.org

.fastq file
(contains base quality info)

Raw signal
(current from a MinIon channel)

```
4.2017/data/reads/pass/0/iobl_goshawk_20170410_FNFA845819_MN19777_mux_scan_SQR170407_54198_ch103_read29_strand.fast5
```

Data selection: [0] ~ [0]

```
@channel_103_062f2402-85b2-4f63-8e73-41995ce0b3d2_template
ATCAATGCTGTTCGTTTTGGTGCTACTTGCCTGTCGCTCGTACAGTCTGCTGGGGCTTTCCATTGGCTCTGAAGG
ATTATAAGTTCCGTCAATTTGTGCATGGACAACTTTTGACAAGTCTTTCTGAACATGATCATGATATTGAAGACCT
GCATGCTACGGAGAACTTGATGGCAGCTCCTCTTCAGGTGATAGTCATCTTTATAGTCTCACCTCTTTGAGTTCAC
TTGACAACTCTGTGACCTTTTCTTAACTTTGTGACACAAAAGCATGGCACCGTCGAATACTTTTGCTGTTGGAAAG
TTCACTATGACTATAAAGCAAAGCAGTCAACTAGATGCTAAATCTGGAATATATCTCACATGGTACTGGTATAGCT
GCCAGTCTTGGGGAAAATCTAAGTTAAAGAGGTTTGTAACCAGTTTCACAGCTATTCCACTGGAACCTGTTTTGAA
ATTTGGACTTCTGTCACATCGGCACTAAACAGGAGCTTCGCTTCAAAATATATGACACAAATGGCAAACCCAGGCC
AACGGATAATTTGTGCTGGGGAATTTCCATTGGCTCTGATTATAGGCGATTCCCTCAATTTGCTACATGGACAAC
TTTTGTCTTTCTACAAGATGATATTCAAGACCTGGATGCACGGGAGGCACTGGCAGCAATATCCTTTCAACACAA
AACGTTCTGTTTATGTTTCTTAGACACTCGTCTACACT
+
("))(,%&((,,))("(1/2//3/,--"++,'-+,+-"&(("+","-+++-/5:80)-+85,<9<74114334<512./5<.8993233;
+-,"23433241),0220-/-05/255=46555543531.//13412802;33;;8771'+),/.,+1-5"('++1/50+,/,231
3:5<017-598/"2+,//6=46:<9631102244<=6500::/,))0215:55<84::9<71:6."-""""("(/393-1630+8
8=<<710(&+.)"88-,9:<,"),/-.97785589842699.<<;80++))-0/"/0+,"/'+1.24421049317====<=6::79
799689<9<==<<<8368::9<<<==<<<<420)/4-/-,))-111141</).066=91/3-69677;82/,"1..3=<::89<
==<<83..1/"0).1/66;=-)",)")"(++-))/.489<8(4-0--(./4-0++063778.00888:.0+4.01348+,01./64"6
9-8:<:5.2-+,-("")/6-403>944(2.+4972-/-/32/-"/.0+845/./6/52::36401/1.-./1..4:5./-/9972/)-1220
22322:;22:33==<=62,")-)91,124+..(-/(347=85-,'&"1)("'/1468462222=>=<=:<<--3-)-11"/20)/)"
(+'&&
```

Signal at /Raw/Reads/Read_29/ [iobl_goshawk_20170410_FNFA845819_MN19777_mux_scan_SQR120...

```
0   677
1   638
2   589
3   562
4   554
5   570
6   586
7   581
8   573
9   562
10  576
11  578
12  554
13  564
14  559
15  577
16  571
17  579
18  583
19  571
20  568
21  578
22  567
```

```
Fastq (46173, 2)
  String, length = 1502,   1
  Number of attributes = 0
```

Bases and qualities are added to fast5 during base-calling

Raw signal is recorded during sequencing

"Squiggle"

https://bioinformatics.uni-muenster.de/home/presentations/nanopopie_Bangkok_2017.pdf

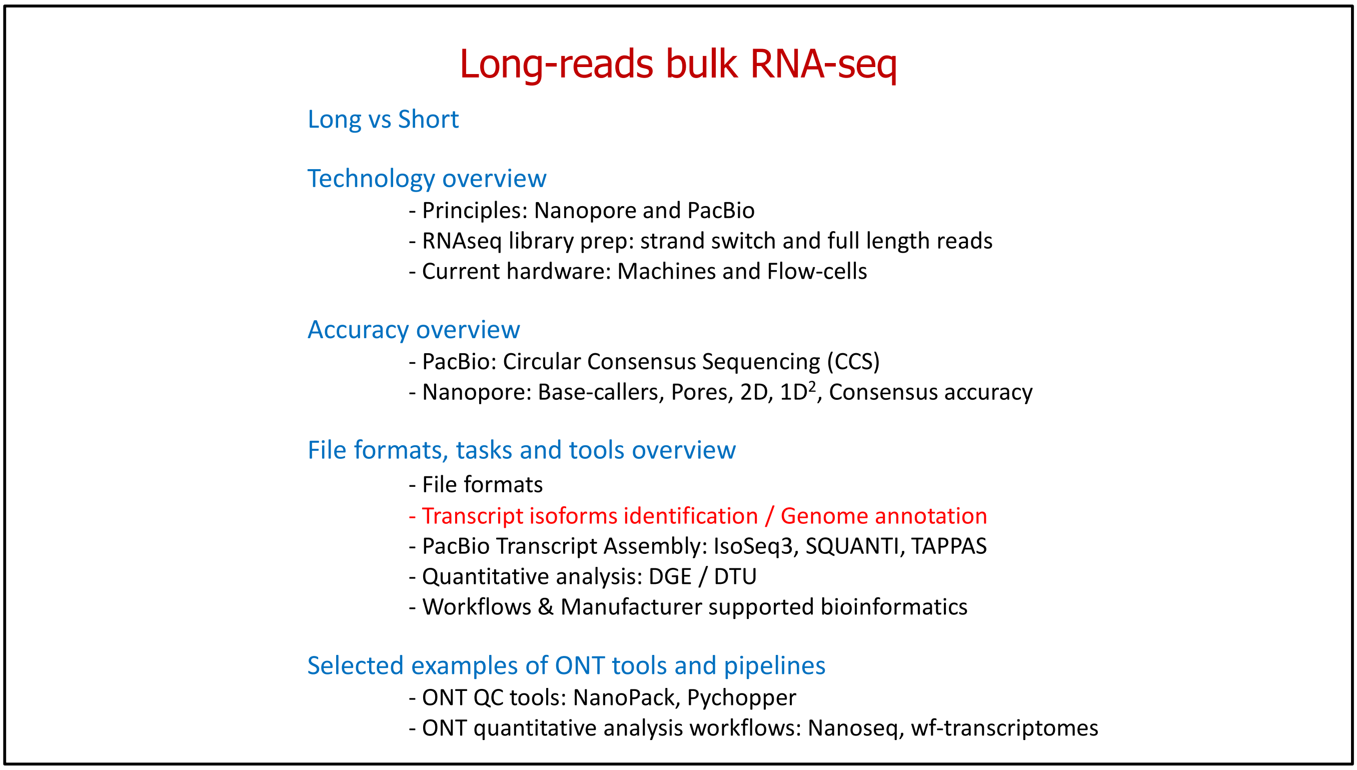
For the same reasons as PacBio Nanopore also could not use FASTQ file for its raw data.

From the very beginning, Nanopore used a sort of XML data format, called ***fast5.*** Before the basecalling, ***fast5*** file contains only the raw data ("squiggle"), along with some additional metadata. After the basecalling, ***fast5*** may contain the sequence and base qualities. At this stage, the sequences from ***fast5 file*** could be exported to FASTQ (with loss of the squiggle information). In our practical session we will analyse Nanopore data using such FASTQ files.

This slide illustrates the structure of ***fast5*** file, and some data contained in different tags, as presented by a ***fast5*** viewer.

Importantly, the ***fast5*** files could be really big (many hundreds of gigabytes), and processing ***fast5*** files may be slow.
A newer version of the Nanopore data files is called ***POD5***: it allows faster data processing.

You may find more information about the Nanopore data formats here:
https://nanoporetech.com/document/data-analysis#file-formats



Long-reads bulk RNA-seq

Long vs Short

Technology overview
- Principles: Nanopore and PacBio
- RNAseq library prep: strand switch and full length reads
- Current hardware: Machines and Flow-cells

Accuracy overview
- PacBio: Circular Consensus Sequencing (CCS)
- Nanopore: Base-callers, Pores, 2D, 1D^2, Consensus accuracy

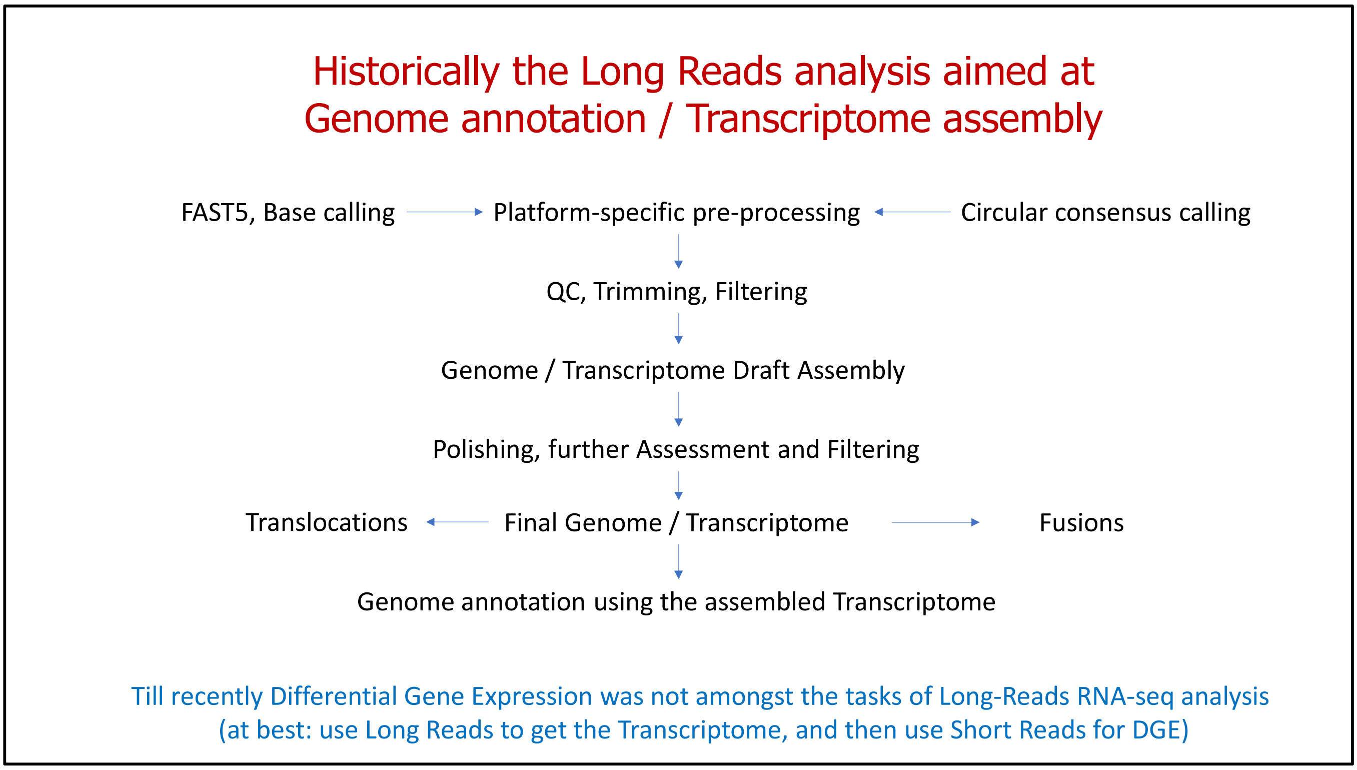
File formats, tasks and tools overview
- File formats
- Transcript isoforms identification / Genome annotation
- PacBio Transcript Assembly: IsoSeq3, SQUANTI, TAPPAS
- Quantitative analysis: DGE / DTU
- Workflows & Manufacturer supported bioinformatics

Selected examples of ONT tools and pipelines
- ONT QC tools: NanoPack, Pychopper
- ONT quantitative analysis workflows: Nanoseq, wf-transcriptomes

We have just seen that the ***file formats*** used in the long-reads sequencing data analysis are quite different from what would be expected after our experience with short-reads.

The same is true about the ***tasks*** that were typically performed by long-reads RNAseq.

During the short-reads RNAseq session we were focusing on quantitative analysis: Differential Gene Expression.
In contrast, till recently, the long-reads RNA-seq was mainly used for ***transcriptome assembly and genome annotation***.



Historically the Long Reads analysis aimed at Genome annotation / Transcriptome assembly

FAST5, Base calling → Platform-specific pre-processing ← Circular consensus calling

↓

QC, Trimming, Filtering

↓

Genome / Transcriptome Draft Assembly

↓

Polishing, further Assessment and Filtering

↓

Translocations ← Final Genome / Transcriptome → Fusions

↓

Genome annotation using the assembled Transcriptome

Till recently Differential Gene Expression was not amongst the tasks of Long-Reads RNA-seq analysis (at best: use Long Reads to get the Transcriptome, and then use Short Reads for DGE)

Till very recently the long reads RNAseq was too expensive to get sufficient depth for reliable transcripts ***quantification***.
So, most of the tools, pipelines and papers published about long reads RNAseq were focused on transcripts ***identification***.

Identification of transcripts could be used for
- Genome annotation
- Transcriptome assembly (even in absence of a reference genome) or
- Fusion detection

Such tasks as “Genome annotation” or “Transcriptome assembly” (especially in absence of a reference genome) may sound alien for a cancer researcher because human genetic analysis enjoys the best annotated reference genome and the best reference transcriptome available (comparatively to other species).

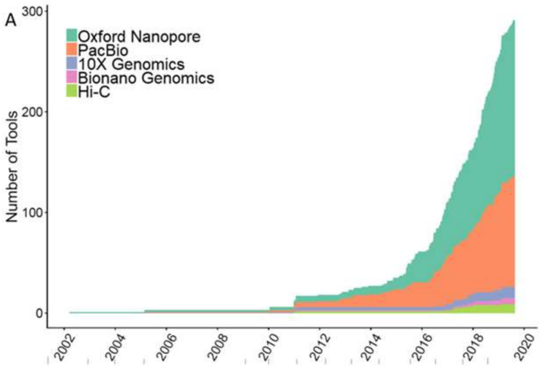
However, after introduction of Nanopore Prometheon (and maybe PacBio Revio+Kinnex in the near future) the long-reads technologies are producing sufficient and affordable depth of sequencing for the quantitative analysis too.

A review of >300 tools for Long Read data analysis in 2020

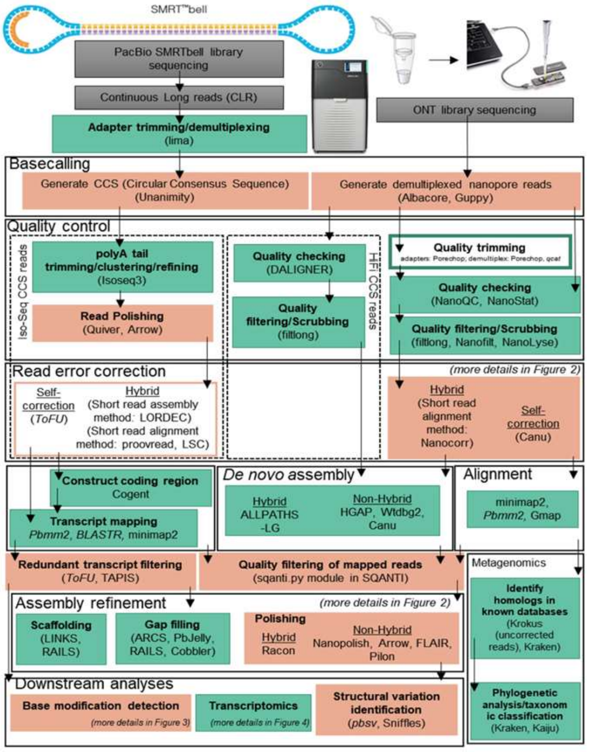
Amarasinghe et al 2020

Opportunities and challenges inlong-read sequencing data analysis

https://doi.org/10.1186/s13059-020-1935-5



Already missing many new tools: Bonito, Medaka, SequelTools etc



SMRT™bell

PacBio SMRTbell library sequencing

Continuous Long reads (CLR)

Adapter trimming/demultiplexing (lima)

ONT library sequencing

Basecalling

Generate CCS (Circular Consensus Sequence) (Unanimity)

Generate demultiplexed nanopore reads (Albacore, Guppy)

Quality control

Iso-Seq CCS reads

polyA tail trimming/clustering/refining (Isoseq3)

Read Polishing (Quiver, Arrow)

Quality checking (DALIGNER)

Quality filtering/Scrubbing (filtlong)

HiFi CCS reads

Quality trimming
adapters: Porechop; demultiplex: Porechop, qcat

Quality checking (NanoQC, NanoStat)

Quality filtering/Scrubbing (filtlong, Nanofilt, NanoLyse)

Read error correction

Self-correction (*ToFU*)

Hybrid (Short read assembly method: LORDEC) (Short read alignment method: proovread, LSC)

(more details in Figure 2)

Hybrid (Short read alignment method: Nanocorr)

Self-correction (Canu)

Construct coding region Cogent

Transcript mapping *Pbmm2, BLASTR,* minimap2

De novo assembly

Hybrid ALLPATHS-LG

Non-Hybrid HGAP, Wtdbg2, Canu

Alignment

minimap2, *Pbmm2*, Gmap

Redundant transcript filtering (*ToFU*, TAPIS)

Quality filtering of mapped reads (sqanti.py module in SQANTI)

Metagenomics

Identify homologs in known databases (Krokus (uncorrected reads), Kraken)

Assembly refinement

(more details in Figure 2)

Scaffolding (LINKS, RAILS)

Gap filling (ARCS, PbJelly, RAILS, Cobbler)

Polishing

Hybrid Racon

Non-Hybrid Nanopolish, Arrow, FLAIR, Pilon

Downstream analyses

Base modification detection (more details in Figure 3)

Transcriptomics (more details in Figure 4)

Structural variation identification (*pbsv*, Sniffles)

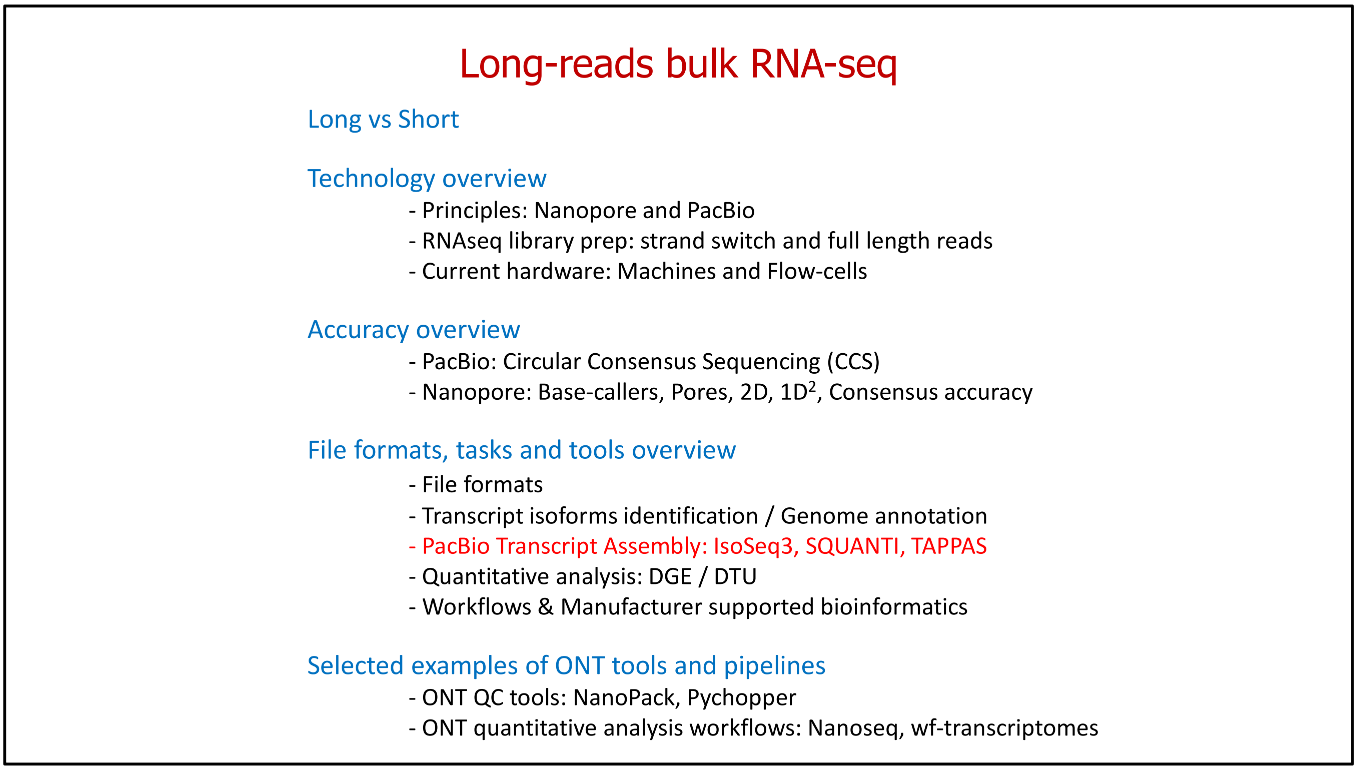
Phylogenetic analysis/taxonomic classification (Kraken, Kaiju)

It looks like the tools for the long-reads data analysis are proliferating even faster than the tools for short-reads analysis :)

This review of 2020 already mentions hundreds of tools (at the time the long reads sequencing was an expensive exercise …)

These are more recent reviews; they compare even more tools and algorithms:
https://www.nature.com/articles/s41592-024-02298-3
https://www.nature.com/articles/s41467-024-48117-3

Comparison of the tools is now a growing branch of bioinformatics!



Long-reads bulk RNA-seq

Long vs Short

Technology overview

- Principles: Nanopore and PacBio
- RNAseq library prep: strand switch and full length reads
- Current hardware: Machines and Flow-cells

Accuracy overview

- PacBio: Circular Consensus Sequencing (CCS)
- Nanopore: Base-callers, Pores, 2D, 1D^2, Consensus accuracy

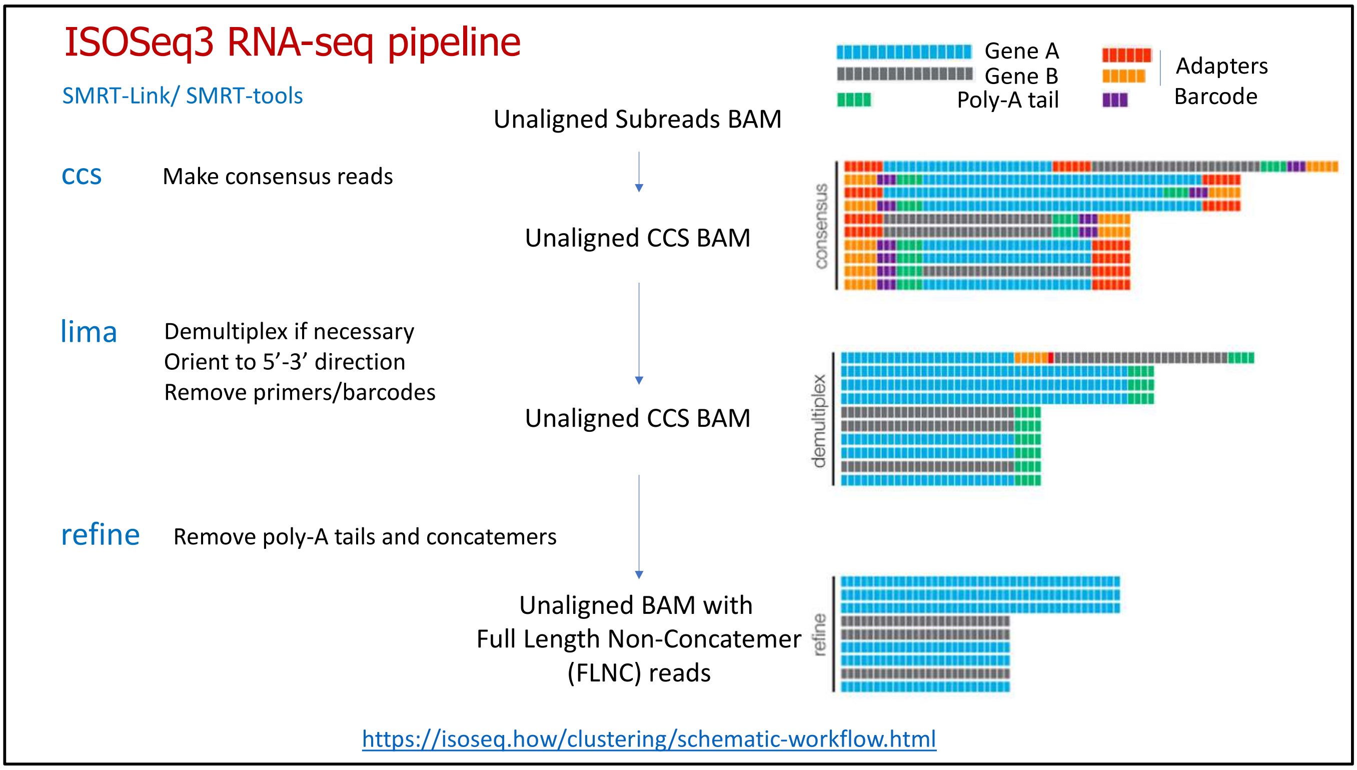
File formats, tasks and tools overview

- File formats
- Transcript isoforms identification / Genome annotation
- PacBio Transcript Assembly: IsoSeq3, SQUANTI, TAPPAS
- Quantitative analysis: DGE / DTU
- Workflows & Manufacturer supported bioinformatics

Selected examples of ONT tools and pipelines

- ONT QC tools: NanoPack, Pychopper
- ONT quantitative analysis workflows: Nanoseq, wf-transcriptomes

I will not compare here different trajectories and tools to generate genome annotation form long-reads RNAseq.
I will only illustrate how this task is performed with one reasonably reputable pipeline: IsoSeq3 from PacBio.



ISOSeq3 RNA-seq pipeline

SMRT-Link/ SMRT-tools

Unaligned Subreads BAM

ccs Make consensus reads

Unaligned CCS BAM

lima Demultiplex if necessary
Orient to 5’-3’ direction
Remove primers/barcodes

Unaligned CCS BAM

refine Remove poly-A tails and concatemers

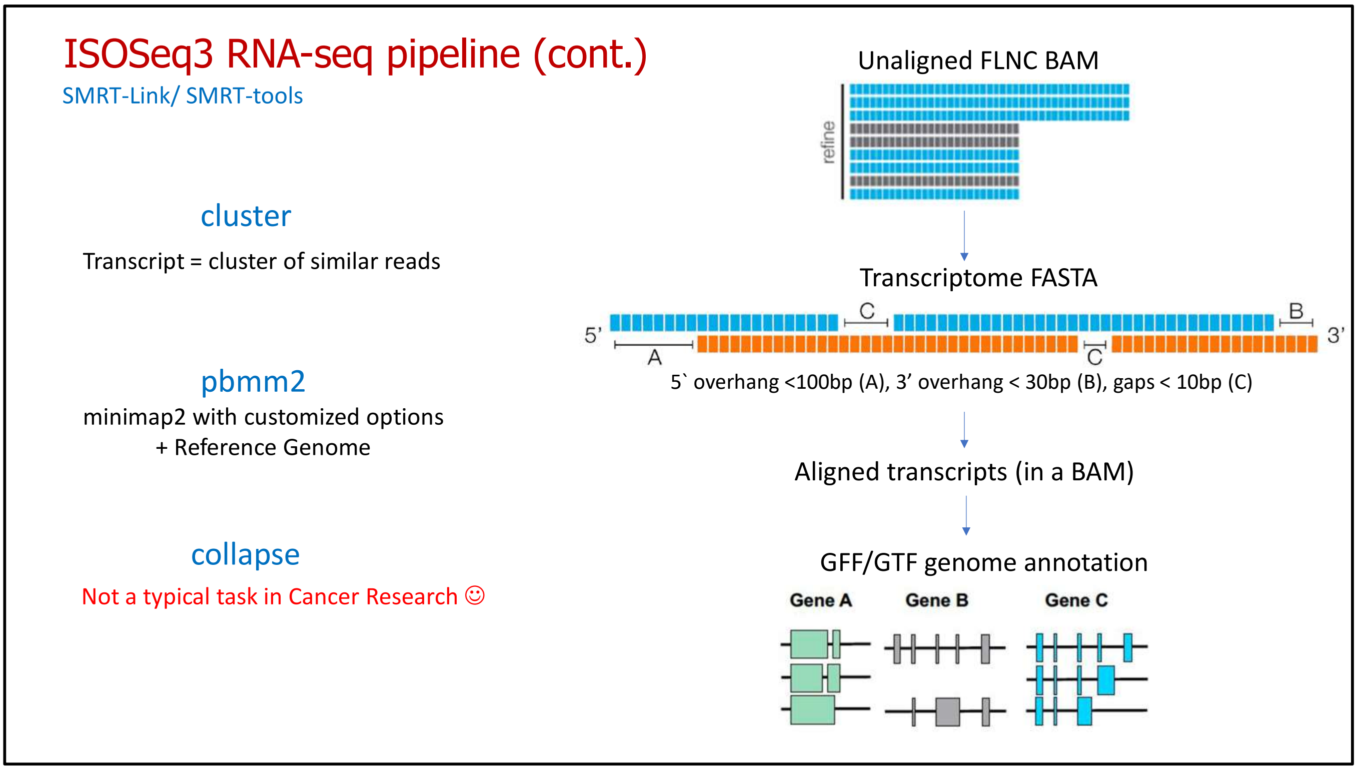
Unaligned BAM with
Full Length Non-Concatemer
(FLNC) reads

https://isoseq.how/clustering/schematic-workflow.html

IsoSeq stands for Isoforms Sequencing.
It was developed by PacBio for transcripts isoforms identification.

It includes preprocessing that required for virtually any PacBio RNAseq task: making consensus reads, orienting reads, removing artefacts and primers, etc.
The pre-processing generates so called Full Length Non-Concatemer (FLNC) reads.

Tools used in this pipeline are available from their pages on PacBio’s GitHub.
Also, they are included into ***SMRT-Link*** software (or ***SMRT-tools*** toolset) available from PacBio.



ISOSeq3 RNA-seq pipeline (cont.)

SMRT-Link/ SMRT-tools

cluster

Transcript = cluster of similar reads

pbmm2

minimap2 with customized options
+ Reference Genome

collapse

Not a typical task in Cancer Research ☺

Unaligned FLNC BAM

Transcriptome FASTA

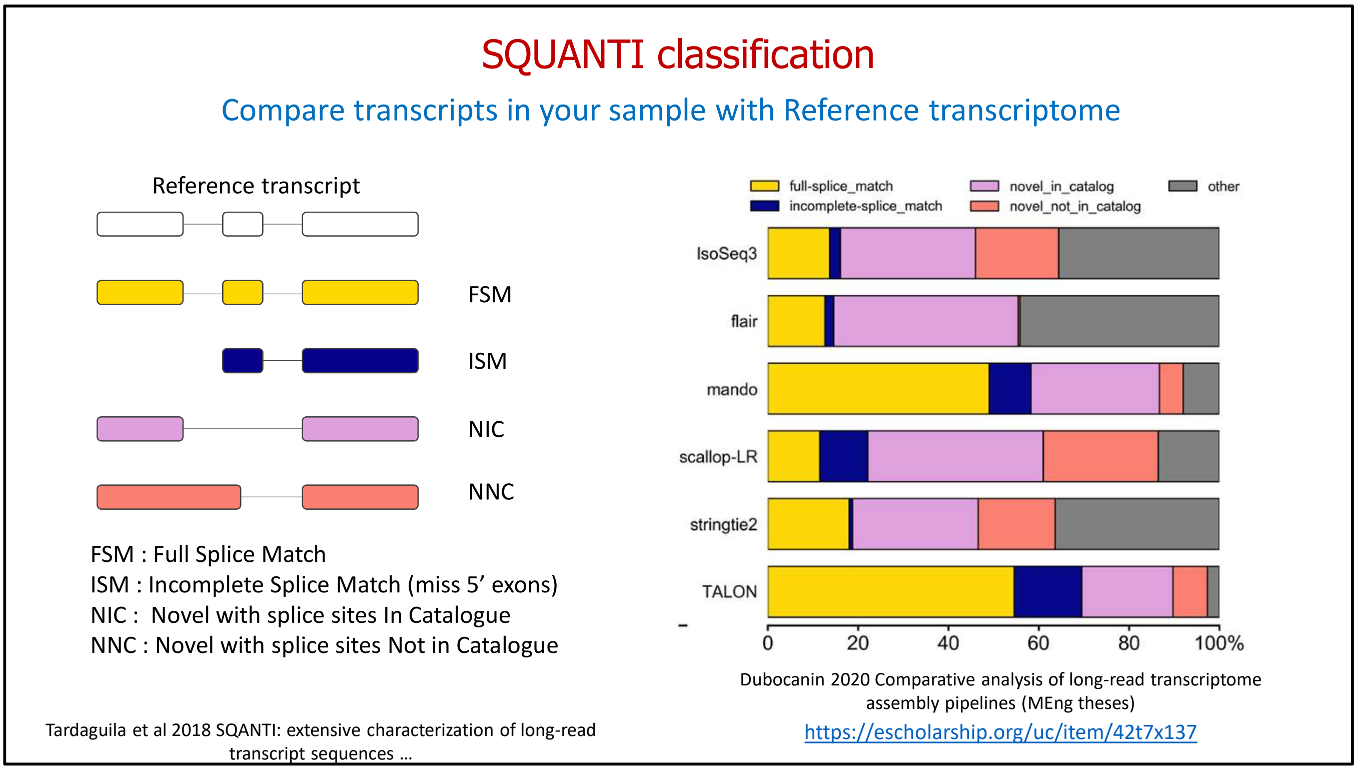
5` overhang <100bp (A), 3’ overhang < 30bp (B), gaps < 10bp (C)

Aligned transcripts (in a BAM)

GFF/GTF genome annotation

Then FLNC reads are clustered (similar reads are combined) and aligned to reference genome with a specifically tuned version of minimap2 (called pbmm2).
Finally, the different identified transcript isoforms are collapsed to genes, and recorded into GFF/GTF file format.

You can see that after alignment IsoSeq pipeline does not count transcripts: instead, it just “collapses” similar transcripts and generates genome annotation. Arguably, this is not a typical task in cancer research…



SQUANTI classification

Compare transcripts in your sample with Reference transcriptome

FSM : Full Splice Match
ISM : Incomplete Splice Match (miss 5' exons)
NIC : Novel with splice sites In Catalogue
NNC : Novel with splice sites Not in Catalogue

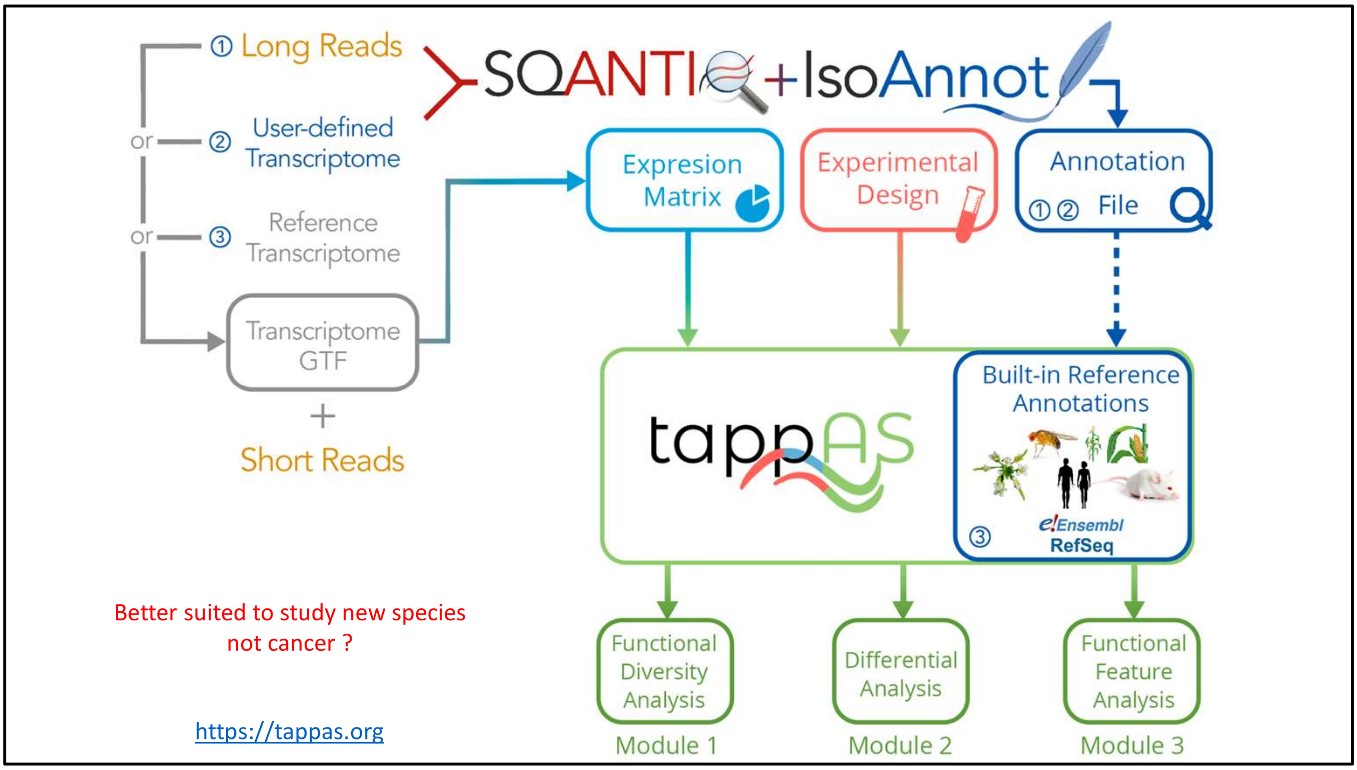
Tardaguila et al 2018 SQANTI: extensive characterization of long-read transcript sequences …

Dubocanin 2020 Comparative analysis of long-read transcriptome assembly pipelines (MEng theses)
https://escholarship.org/uc/item/42t7x137

The down-stream step after transcript isoforms discovery often includes classification of these isoforms.

Squanti is a very popular classification system (and tool). It compares the detected transcripts with the previously available reference transcriptome ("catalogue")..

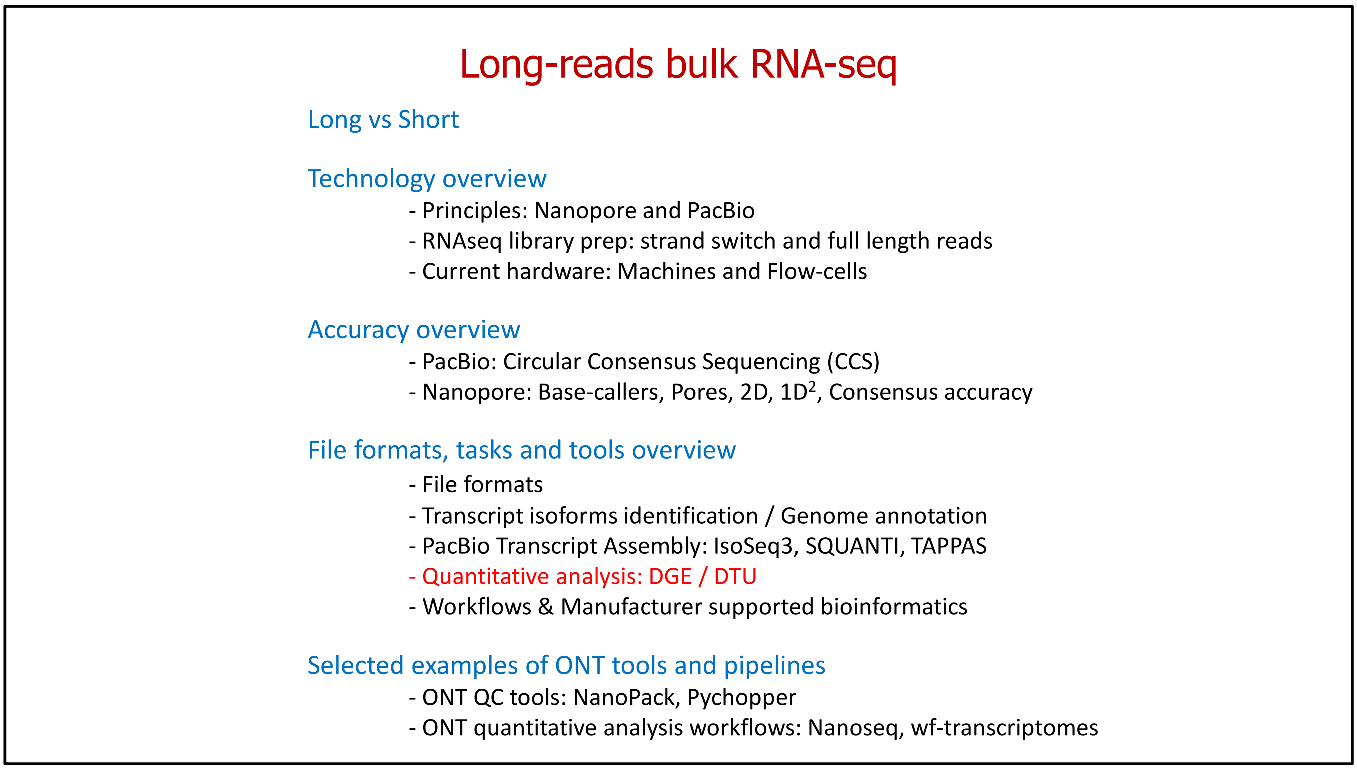
The right panel on this slide compares ***Squanti*** classifications of transcripts identified by different long-reads transcript identification pipelines. Surprisingly to me, despite the long reads supposedly should span entire transcripts, this example suggests strong discrepancies between the transcriptome assembly pipelines…



Better suited to study new species
not cancer ?

https://tappas.org

Squanti is a part of a wider ecosystem of tools for the transcriptome annotation, called ***TAPPAS.***
Apparently, it also includes a differential expression functionality.
However, initially the ***TAPPAS*** quantification module relied on the additional short-reads data.



Long-reads bulk RNA-seq

Long vs Short

Technology overview

- Principles: Nanopore and PacBio
- RNAseq library prep: strand switch and full length reads
- Current hardware: Machines and Flow-cells

Accuracy overview

- PacBio: Circular Consensus Sequencing (CCS)
- Nanopore: Base-callers, Pores, 2D, $1D^2$, Consensus accuracy

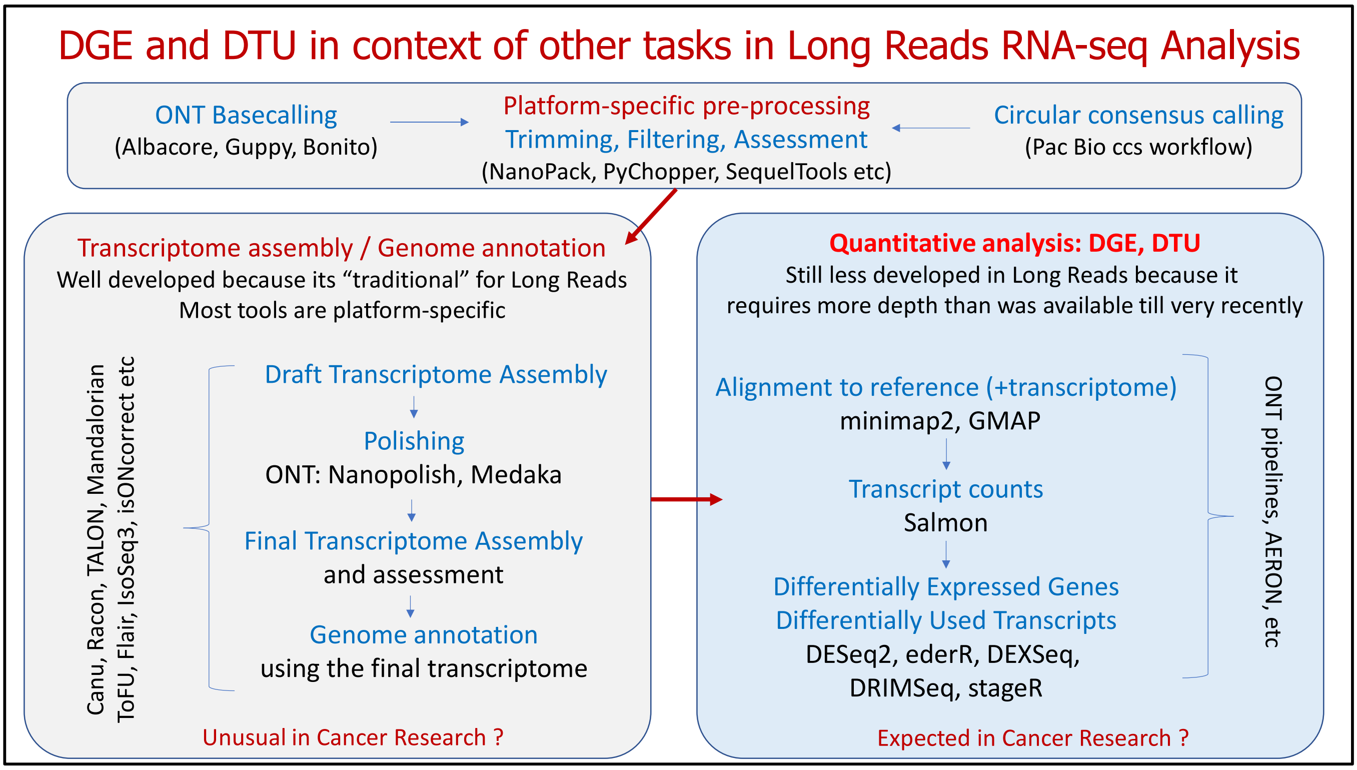
File formats, tasks and tools overview

- File formats
- Transcript isoforms identification / Genome annotation
- PacBio Transcript Assembly: IsoSeq3, SQUANTI, TAPPAS
- Quantitative analysis: DGE / DTU
- Workflows & Manufacturer supported bioinformatics

Selected examples of ONT tools and pipelines

- ONT QC tools: NanoPack, Pychopper
- ONT quantitative analysis workflows: Nanoseq, wf-transcriptomes

Now, when sufficient depth becomes affordable in long reads sequencing, quantitative analysis: differential gene expression (DGE) and differential transcript usage (DTU) can be performed using the long reads data directly.



DGE and DTU in context of other tasks in Long Reads RNA-seq Analysis

ONT Basecalling
(Albacore, Guppy, Bonito)

Platform-specific pre-processing
Trimming, Filtering, Assessment
(NanoPack, PyChopper, SequelTools etc)

Circular consensus calling
(Pac Bio ccs workflow)

Transcriptome assembly / Genome annotation
Well developed because its “traditional” for Long Reads
Most tools are platform-specific

Canu, Racon, TALON, Mandalorian
ToFU, Flair, IsoSeq3, isONcorrect etc

Draft Transcriptome Assembly

Polishing
ONT: Nanopolish, Medaka

Final Transcriptome Assembly
and assessment

Genome annotation
using the final transcriptome

Unusual in Cancer Research ?

Quantitative analysis: DGE, DTU
Still less developed in Long Reads because it
requires more depth than was available till very recently

Alignment to reference (+transcriptome)
minimap2, GMAP

Transcript counts
Salmon

Differentially Expressed Genes
Differentially Used Transcripts
DESeq2, ederR, DEXSeq,
DRIMSeq, stageR

ONT pipelines, AERON, etc

Expected in Cancer Research ?

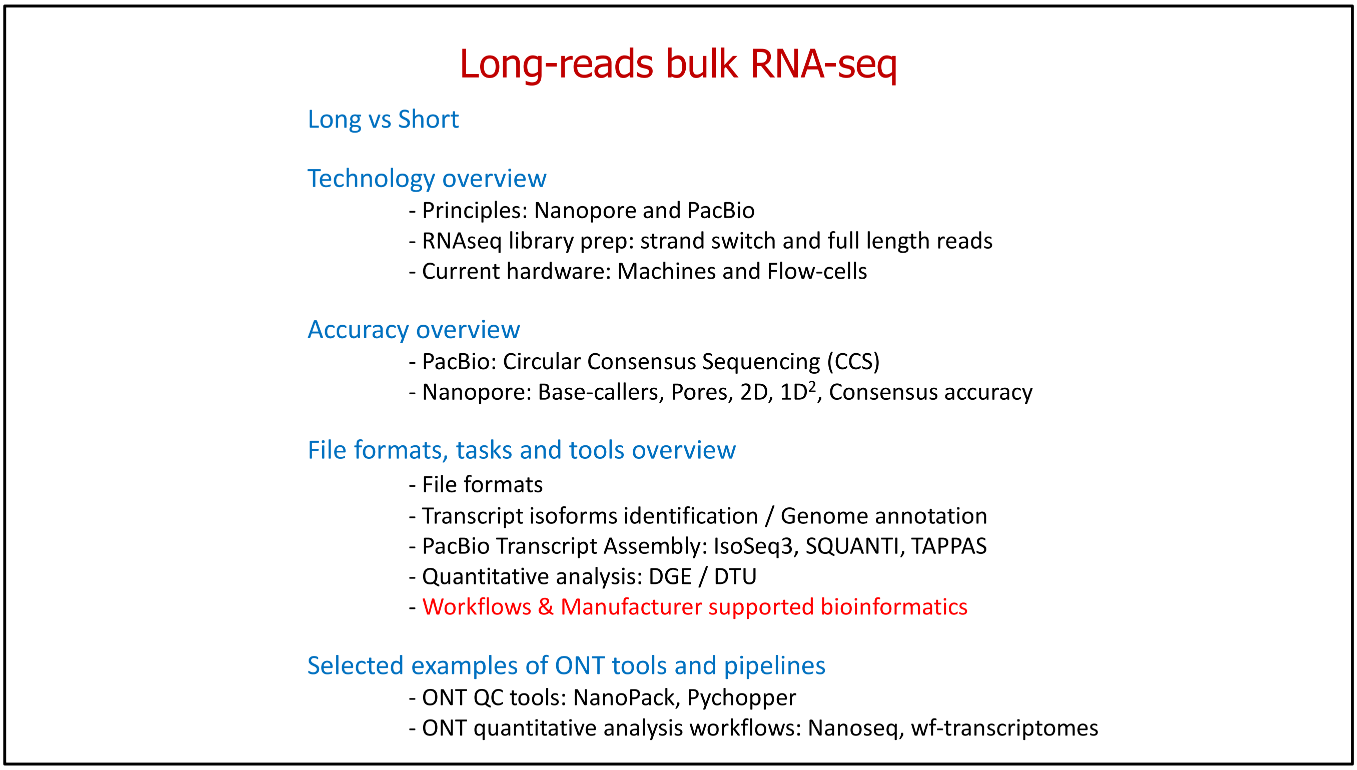
In fact, quantitative RNAseq analysis pipelines often include the transcriptome assembly (=genome annotation) as the initial step of analysis.

On this slide you may see that the assembled transcriptome (the study-specific GTF) could just be used for transcripts count by Salmon, and then the counts can be used in DESeq2 or edgeR, as we did for the short reads.

Of course, the differential transcript usage (DTU) analysis requires additional R packages.

The specific tools mentioned here are just to illustrate what was used in the literature, some of them are already deprecated.
I do not claim that the mentioned tools are the best or the most recent.

And, of course, there are many RNAseq tasks and tools that are not even mentioned here, such as Nanopore Direct RNA-seq, RNA-modifications, poly-A tail length etc



Long-reads bulk RNA-seq

Long vs Short

Technology overview

- Principles: Nanopore and PacBio
- RNAseq library prep: strand switch and full length reads
- Current hardware: Machines and Flow-cells

Accuracy overview

- PacBio: Circular Consensus Sequencing (CCS)
- Nanopore: Base-callers, Pores, 2D, 1D^2, Consensus accuracy

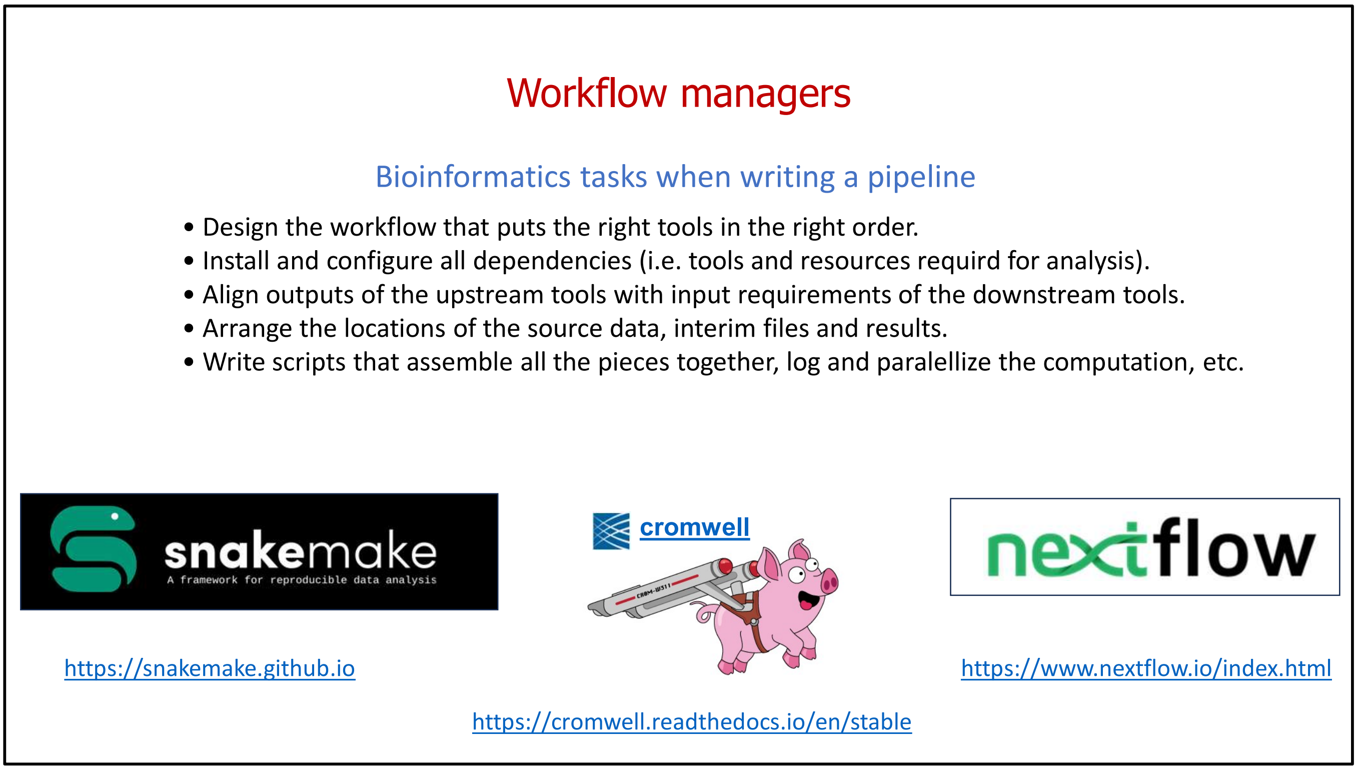
File formats, tasks and tools overview

- File formats
- Transcript isoforms identification / Genome annotation
- PacBio Transcript Assembly: IsoSeq3, SQUANTI, TAPPAS
- Quantitative analysis: DGE / DTU
- Workflows & Manufacturer supported bioinformatics

Selected examples of ONT tools and pipelines

- ONT QC tools: NanoPack, Pychopper
- ONT quantitative analysis workflows: Nanoseq, wf-transcriptomes

Finally, before describing specific tools that we will use in the practical session, I would like to mention workflows and workflow managers.
Writing and publishing pipelines (=workflows) is currently a strong trend in bioinformatics community.
It is very much present in the long-reads RNAseq data analysis field.



Workflow managers

Bioinformatics tasks when writing a pipeline

- Design the workflow that puts the right tools in the right order.
- Install and configure all dependencies (i.e. tools and resources requird for analysis).
- Align outputs of the upstream tools with input requirements of the downstream tools.
- Arrange the locations of the source data, interim files and results.
- Write scripts that assemble all the pieces together, log and paralellize the computation, etc.

https://snakemake.github.io

https://www.nextflow.io/index.html

https://cromwell.readthedocs.io/en/stable

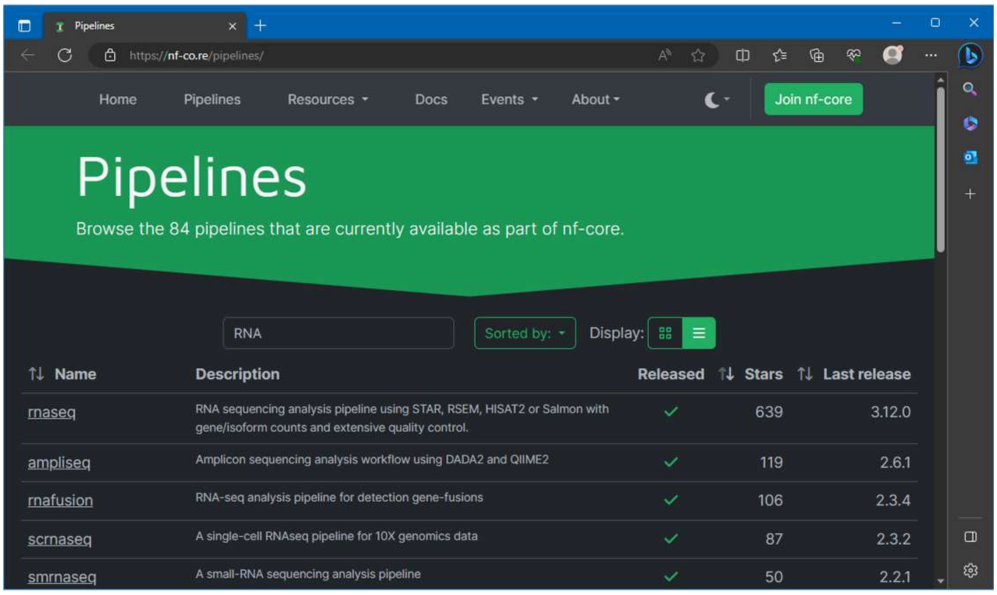
Even a simple bioinformatics task often requires multiple steps and tools.

It takes lots of effort and expertise to connect outputs of the different tools, to perform necessary checks between the steps, to organise the output in pre-specified folder structure, to manage parallelisation (where available), etc.

Workflow managers are special programs that help to connect multiple tools into pipelines.

This slide shows three popular in bioinformatics workflow manages: Snakemake, Cromwell and Nextflow. While all of them are being widely used, my personal opinion is that Nextflow is becoming more popular for long-reads sequencing data analysis, because it is being adopted by both PacBio and Nanopore.

NF-core: Nextflow pipelines repository



Pipelines

Browse the 84 pipelines that are currently available as part of nf-core.

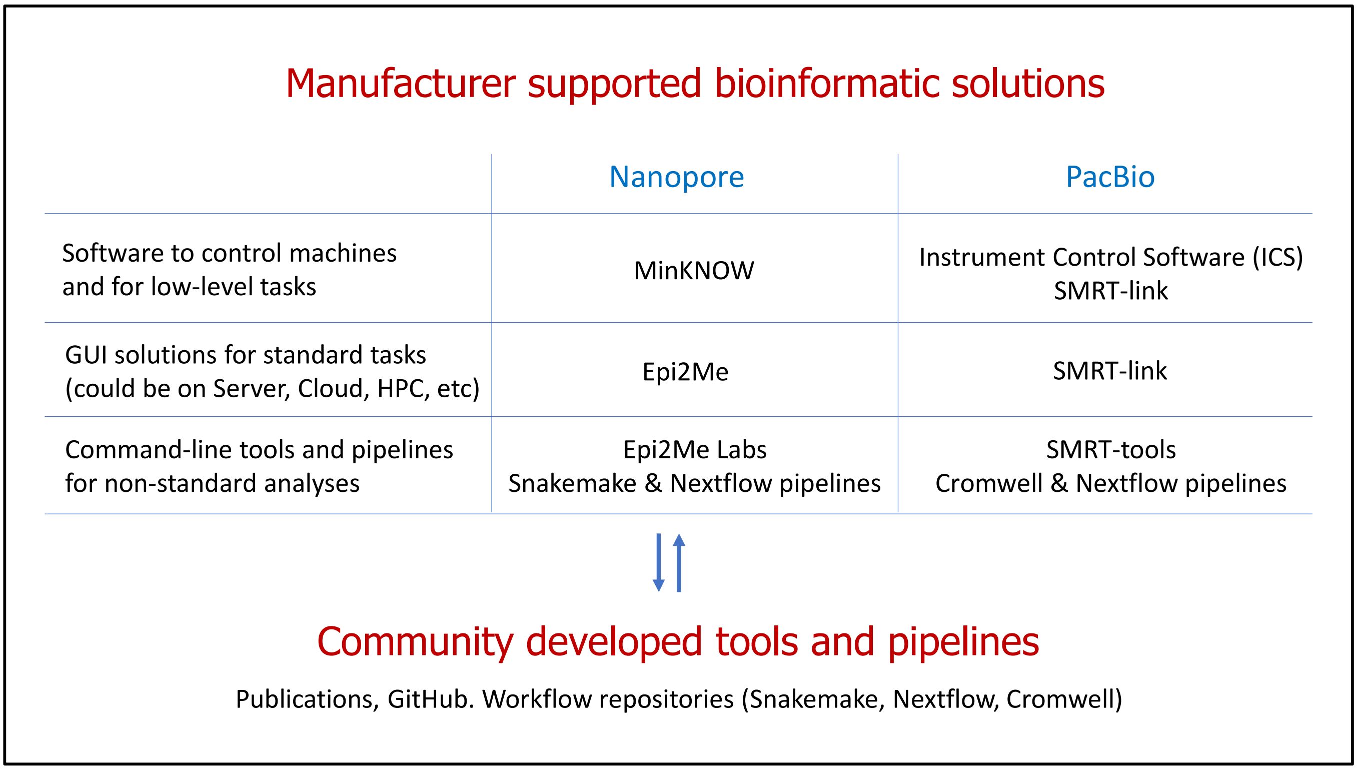
Name	Description	Released	Stars	Last release
rnaseq	RNA sequencing analysis pipeline using STAR, RSEM, HISAT2 or Salmon with gene/isoform counts and extensive quality control.	✓	639	3.12.0
ampliseq	Amplicon sequencing analysis workflow using DADA2 and QIIME2	✓	119	2.6.1
rnafusion	RNA-seq analysis pipeline for detection gene-fusions	✓	106	2.3.4
scrnaseq	A single-cell RNAseq pipeline for 10X genomics data	✓	87	2.3.2
smrnaseq	A small-RNA sequencing analysis pipeline	✓	50	2.2.1

Snakemake workflow catalogue: many hundreds of workflows

https://snakemake.github.io/snakemake-workflow-catalog

NF-core is a catalogue of bioinformatics Nextflow pipelines. You may find there pipelines for many tasks that you studied during this course. It has several pipelines for RNAseq data analysis too ☺

Similar reporsitories are available for other workflow managers too.



Manufacturer supported bioinformatic solutions

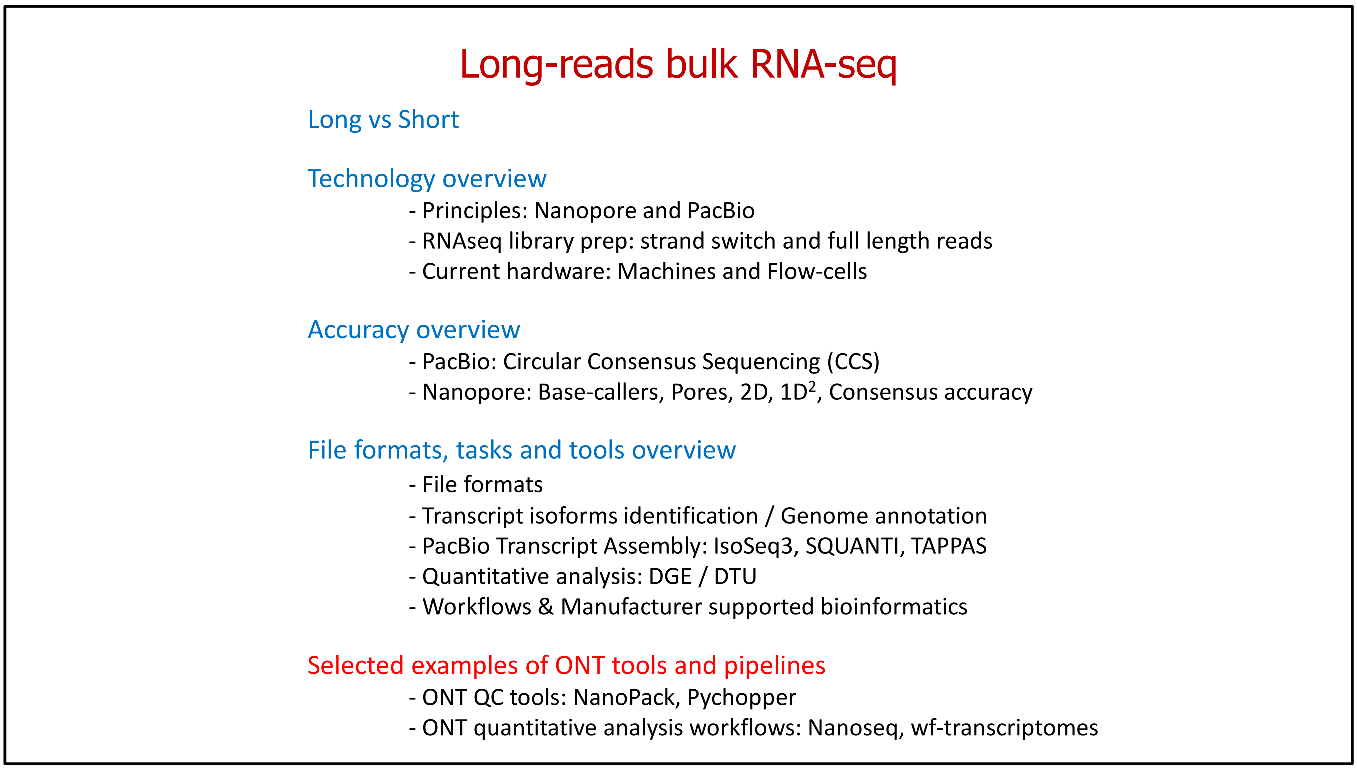
	Nanopore	PacBio
Software to control machines and for low-level tasks	MinKNOW	Instrument Control Software (ICS) SMRT-link
GUI solutions for standard tasks (could be on Server, Cloud, HPC, etc)	Epi2Me	SMRT-link
Command-line tools and pipelines for non-standard analyses	Epi2Me Labs Snakemake & Nextflow pipelines	SMRT-tools Cromwell & Nextflow pipelines

Community developed tools and pipelines

Publications, GitHub. Workflow repositories (Snakemake, Nextflow, Cromwell)

Both Nanopore and PacBio provide extensive Bioinformatics support.
With a pinch of salt, their manufacturer supplied bioinformatics is summarized in such table.

Along with the separate tools, both Nanopore and PacBio develop and publish entire pipelines (=workflows).
We will try one in our practical session …



Long-reads bulk RNA-seq

Long vs Short

Technology overview

- Principles: Nanopore and PacBio
- RNAseq library prep: strand switch and full length reads
- Current hardware: Machines and Flow-cells

Accuracy overview

- PacBio: Circular Consensus Sequencing (CCS)
- Nanopore: Base-callers, Pores, 2D, 1D^2, Consensus accuracy

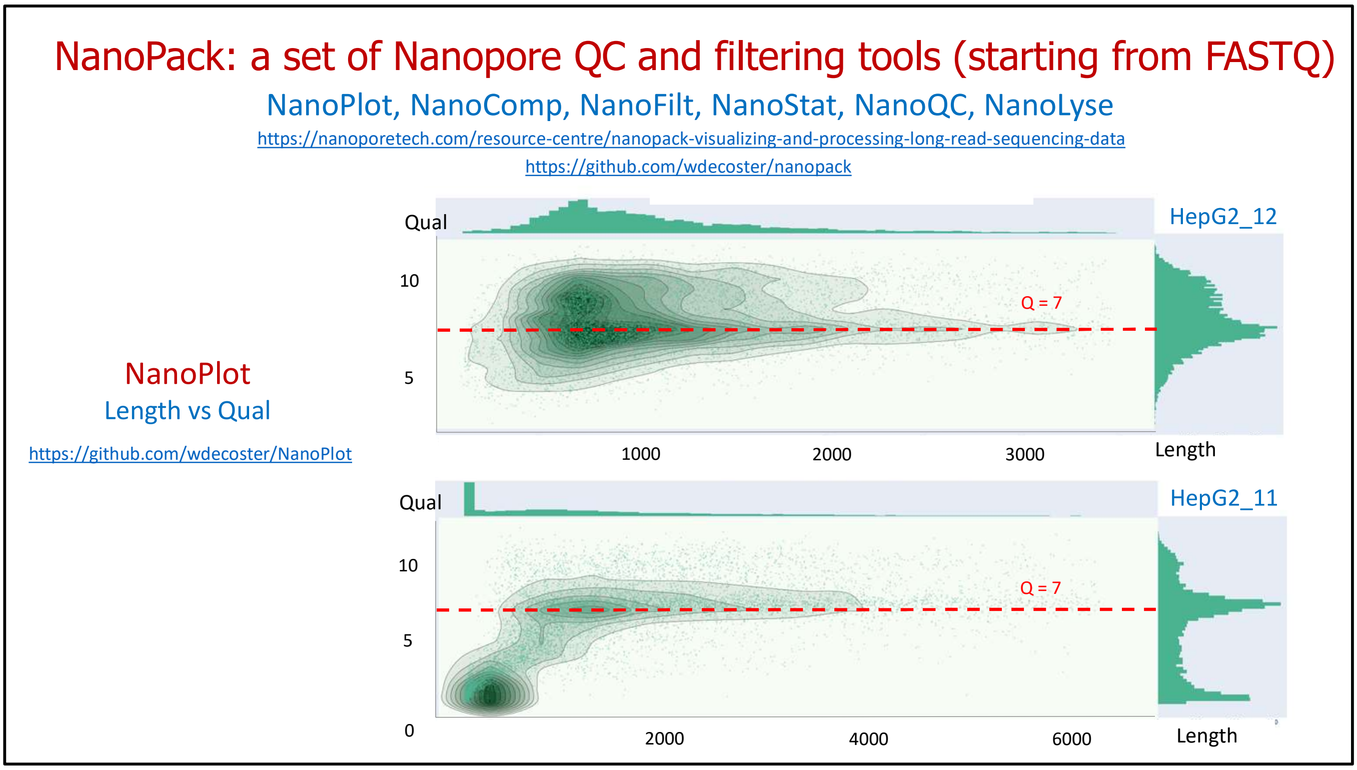
File formats, tasks and tools overview

- File formats
- Transcript isoforms identification / Genome annotation
- PacBio Transcript Assembly: IsoSeq3, SQUANTI, TAPPAS
- Quantitative analysis: DGE / DTU
- Workflows & Manufacturer supported bioinformatics

Selected examples of ONT tools and pipelines

- ONT QC tools: NanoPack, Pychopper
- ONT quantitative analysis workflows: Nanoseq, wf-transcriptomes

The last part of the lecture will illustrate some Oxford Nanopore tools that we will use during the practical session.



NanoPack: a set of Nanopore QC and filtering tools (starting from FASTQ)

NanoPlot, NanoComp, NanoFilt, NanoStat, NanoQC, NanoLyse

https://nanoporetech.com/resource-centre/nanopack-visualizing-and-processing-long-read-sequencing-data

https://github.com/wdecoster/nanopack

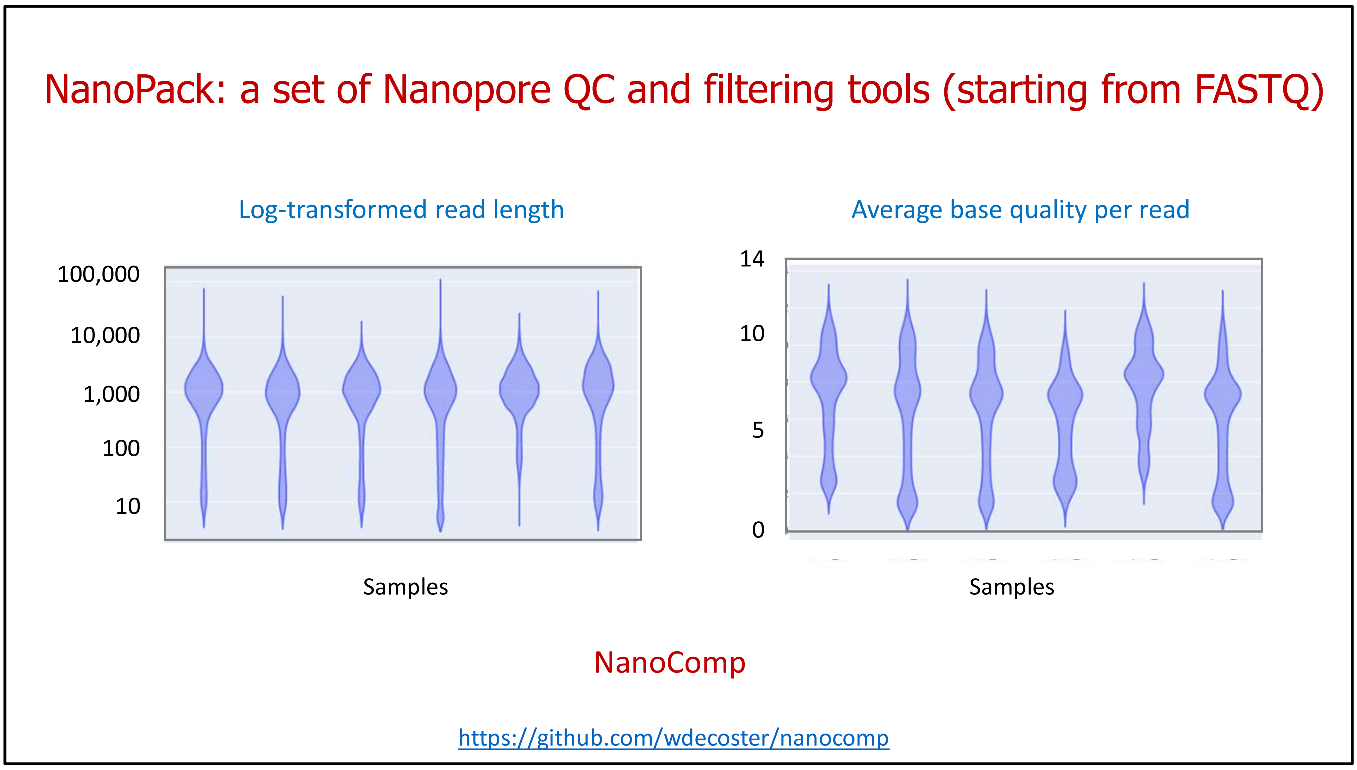
NanoPlot

Length vs Qual

https://github.com/wdecoster/NanoPlot

NanoPack - Simple and straight-to-the-point toolset for QC and filtering Nanopore data (using FASTQ files).

NanoPlot shows a simple and very informative plot of reads Length vs reads Quality.



NanoPack: a set of Nanopore QC and filtering tools (starting from FASTQ)

Log-transformed read length

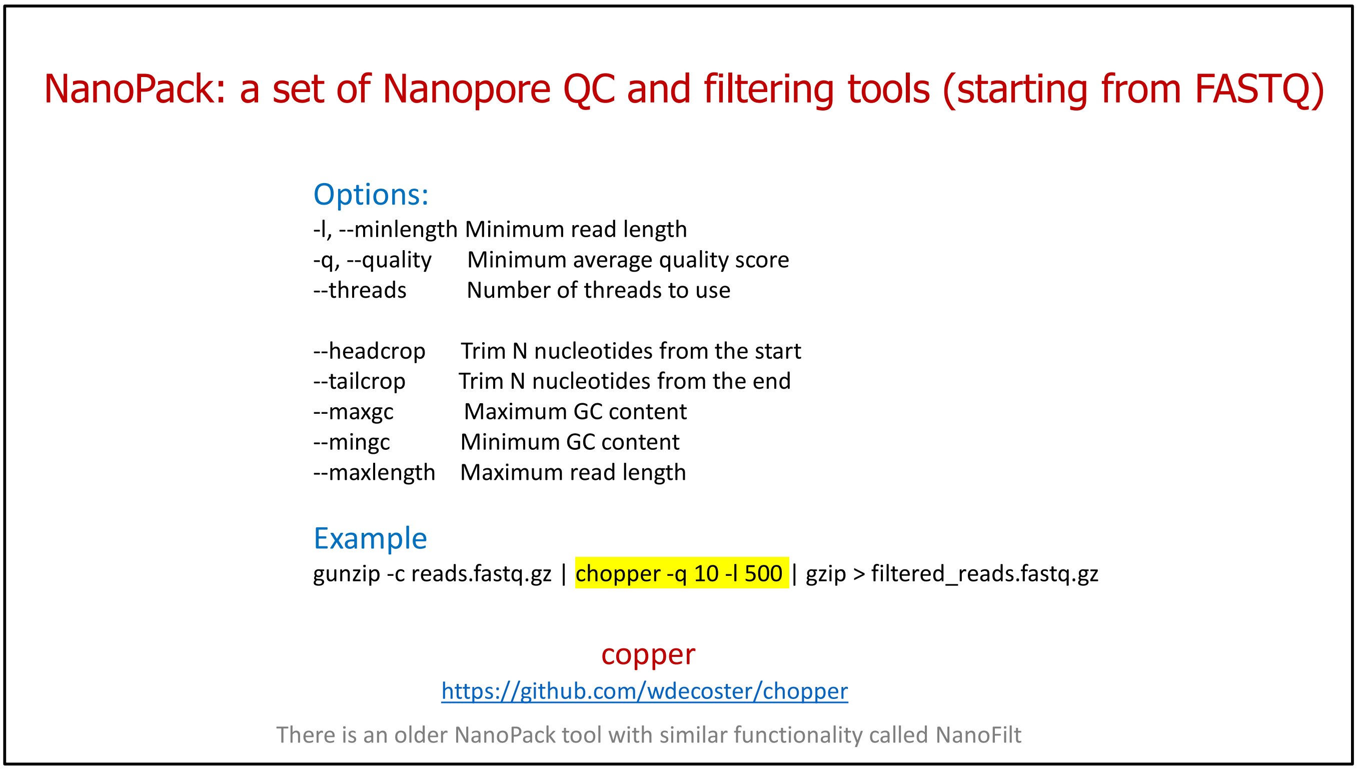
Average base quality per read

NanoComp

https://github.com/wdecoster/nanocomp

NanoComp allows to compare quality metrics between multiple samples.

Actual plots could be interactive (showing additional information when mouse hovers over the plot).



NanoPack: a set of Nanopore QC and filtering tools (starting from FASTQ)

Options:

```
-l, --minlength Minimum read length
-q, --quality     Minimum average quality score
--threads         Number of threads to use

--headcrop      Trim N nucleotides from the start
--tailcrop       Trim N nucleotides from the end
--maxgc          Maximum GC content
--mingc          Minimum GC content
--maxlength    Maximum read length
```

Example

```
gunzip -c reads.fastq.gz | chopper -q 10 -l 500 | gzip > filtered_reads.fastq.gz
```

copper

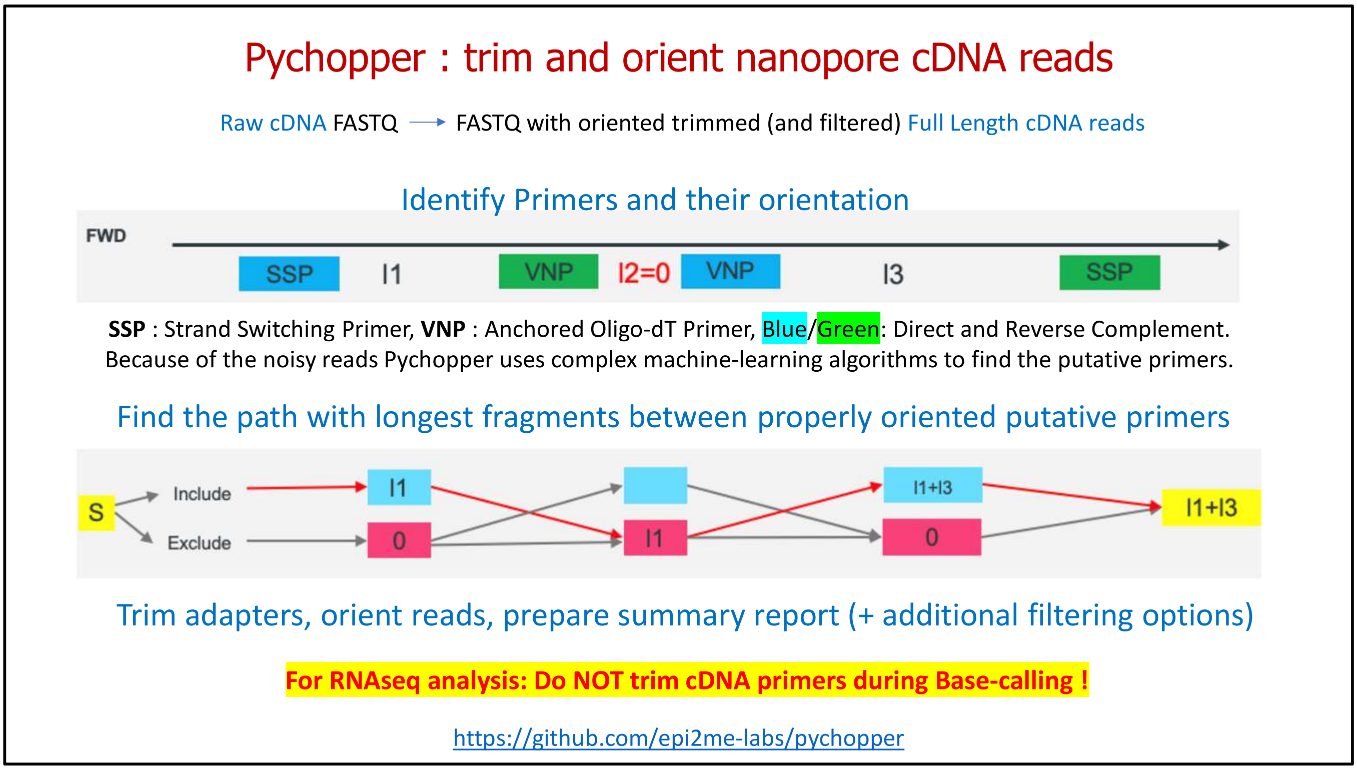
https://github.com/wdecoster/chopper

There is an older NanoPack tool with similar functionality called NanoFilt

After evaluating the quality of the reads, we may need to perform filtering to remove bad quality reads.

A common practice for Nanopore data includes removal of reads by mean qual > 7
(compare to minimal Q20 in PacBio HiFi :)
Copper can filter reads basing on many different parameters too.

Important: NanoPack's ***copper*** is not the same as ***Pychopper*** discussed on the next slide!



Pychopper : trim and orient nanopore cDNA reads

Raw cDNA FASTQ ⟶ FASTQ with oriented trimmed (and filtered) Full Length cDNA reads

Identify Primers and their orientation

SSP : Strand Switching Primer, **VNP** : Anchored Oligo-dT Primer, Blue/Green: Direct and Reverse Complement.
Because of the noisy reads Pychopper uses complex machine-learning algorithms to find the putative primers.

Find the path with longest fragments between properly oriented putative primers

Trim adapters, orient reads, prepare summary report (+ additional filtering options)

For RNAseq analysis: Do NOT trim cDNA primers during Base-calling !

https://github.com/epi2me-labs/pychopper

Pychopper trims, orients and filters Nanopore RNAseq reads.

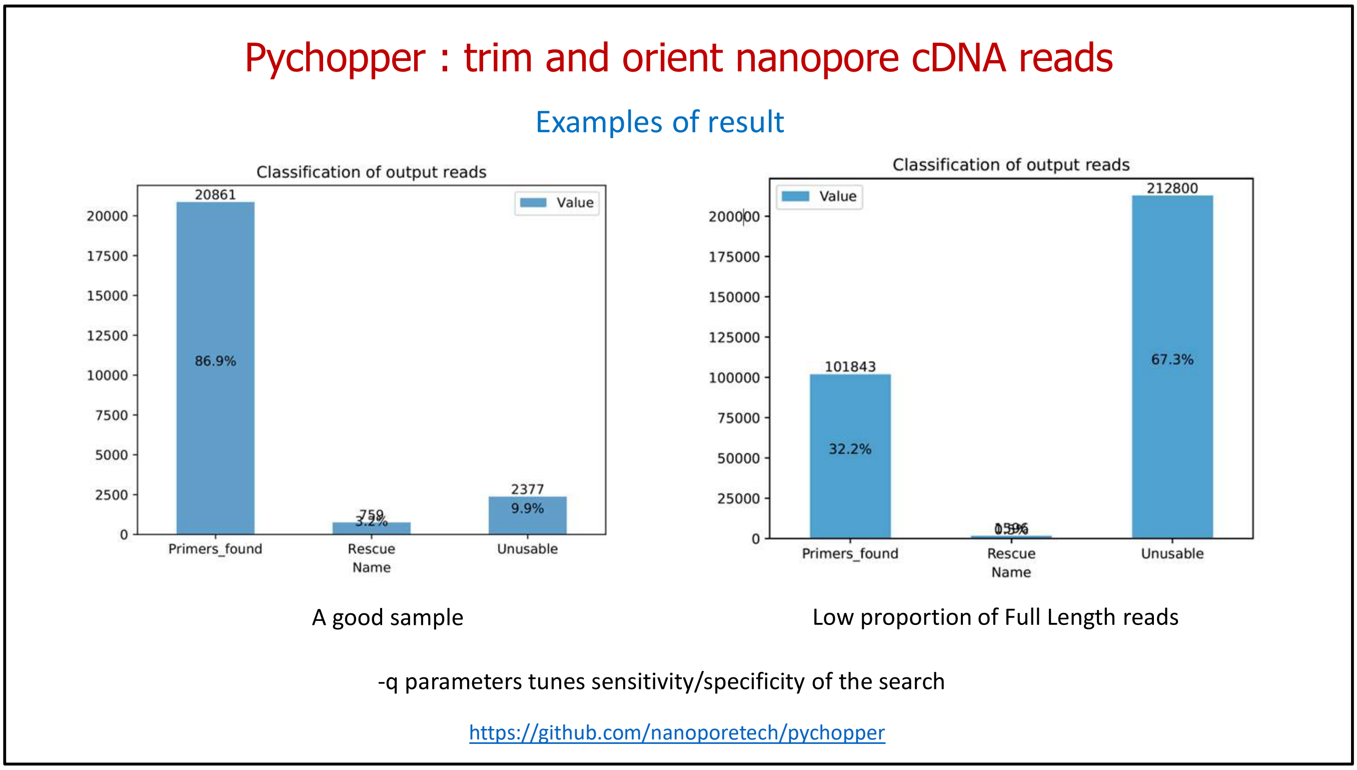
If you remember, initially raw RNAseq long reads include strand switching primer (***SSP***) on one side and anchored oligo-dT primer (***VNP***) on the other.

The strand and orientation can be detected by the position and sequences (***direct*** or ***reverse-complement***) of these primers.

Because of the low base quality of raw Nanopore reads (especially in old Nanopore data) ***Pychopper*** uses complex approximation algorithms to detect the primer sequences.

These are diagrams from the ***Pychopper*** web page, that supposedly should explain some of the employed algorithms.

To be honest, I struggle to understand these diagrams, but you are welcome to try it yourself (the link to web page is provided ☺



Pychopper : trim and orient nanopore cDNA reads

Examples of result

A good sample

Low proportion of Full Length reads

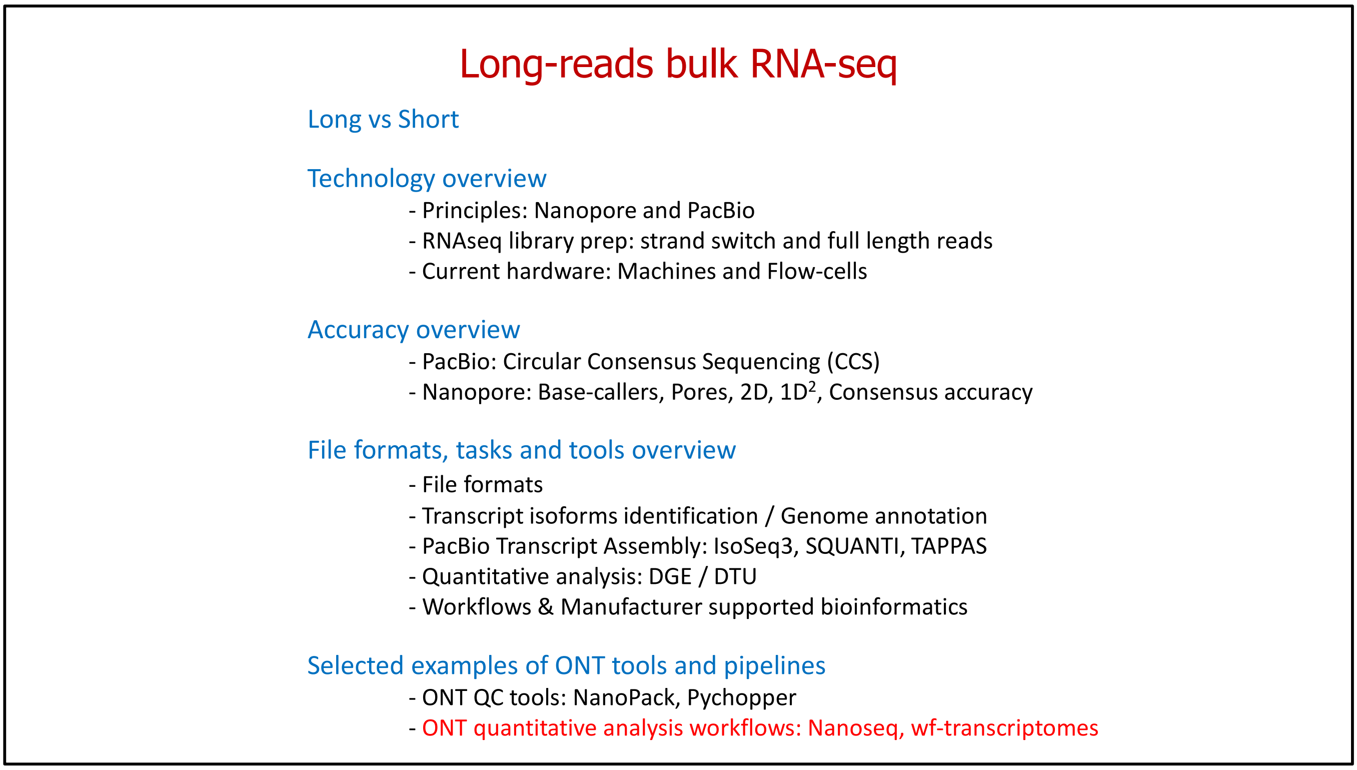
-q parameters tunes sensitivity/specificity of the search

https://github.com/nanoporetech/pychopper

The good thing about ***Pychopper*** is that it provides informative plots about the preprocessing results.

Here you can see a good and a bad example.

The proportion of rejected reads may be tuned by –q parameter when running Pychopper (in the recent versions of Pychopper it is tuned automatically using a sub-set of reads).



Long-reads bulk RNA-seq

Long vs Short

Technology overview

- Principles: Nanopore and PacBio
- RNAseq library prep: strand switch and full length reads
- Current hardware: Machines and Flow-cells

Accuracy overview

- PacBio: Circular Consensus Sequencing (CCS)
- Nanopore: Base-callers, Pores, 2D, 1D^2, Consensus accuracy

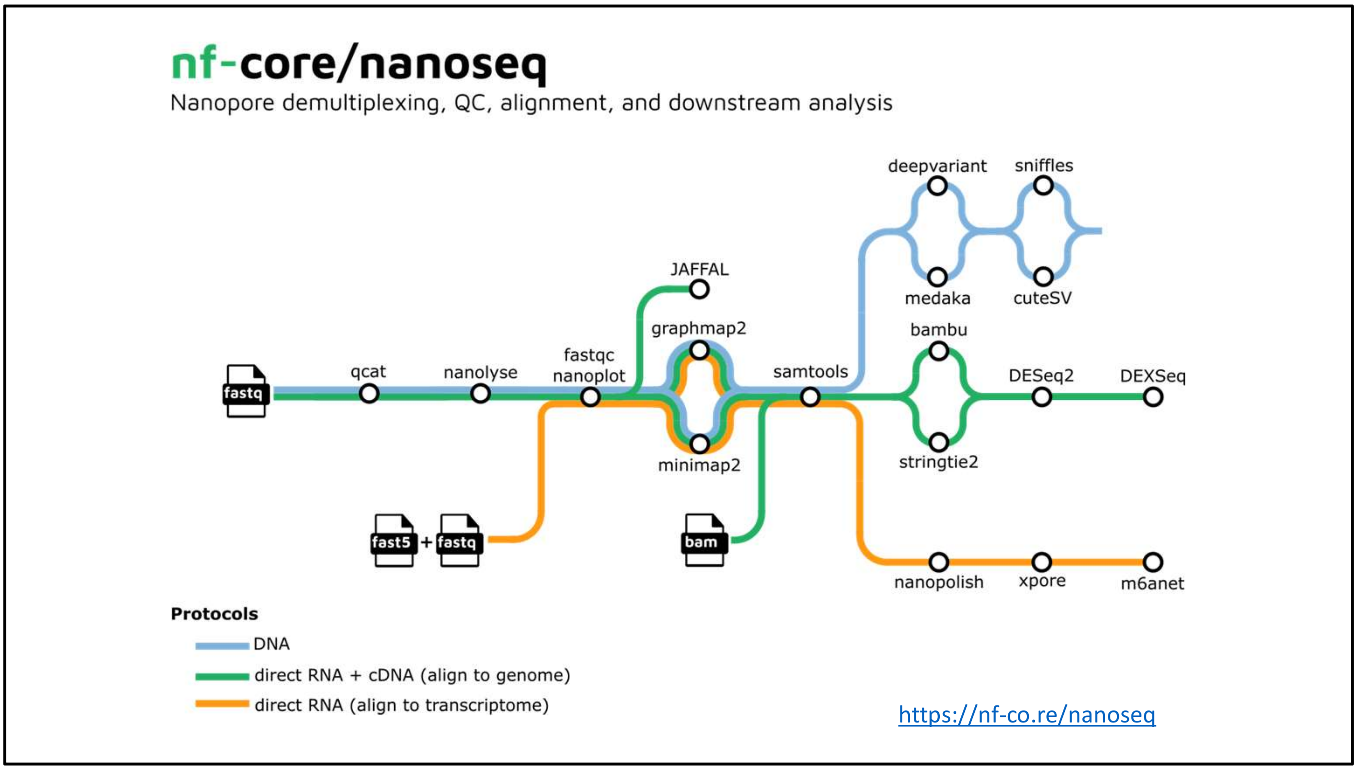
File formats, tasks and tools overview

- File formats
- Transcript isoforms identification / Genome annotation
- PacBio Transcript Assembly: IsoSeq3, SQUANTI, TAPPAS
- Quantitative analysis: DGE / DTU
- Workflows & Manufacturer supported bioinformatics

Selected examples of ONT tools and pipelines

- ONT QC tools: NanoPack, Pychopper
- ONT quantitative analysis workflows: Nanoseq, wf-transcriptomes

Finally, I will mention some workflows, which were developed for analysis of Nanopore RNAseq data.
One of these workflows will be used in our practical session.



nf-core/nanoseq

Nanopore demultiplexing, QC, alignment, and downstream analysis

Protocols

🟦 DNA

🟩 direct RNA + cDNA (align to genome)

🟧 direct RNA (align to transcriptome)

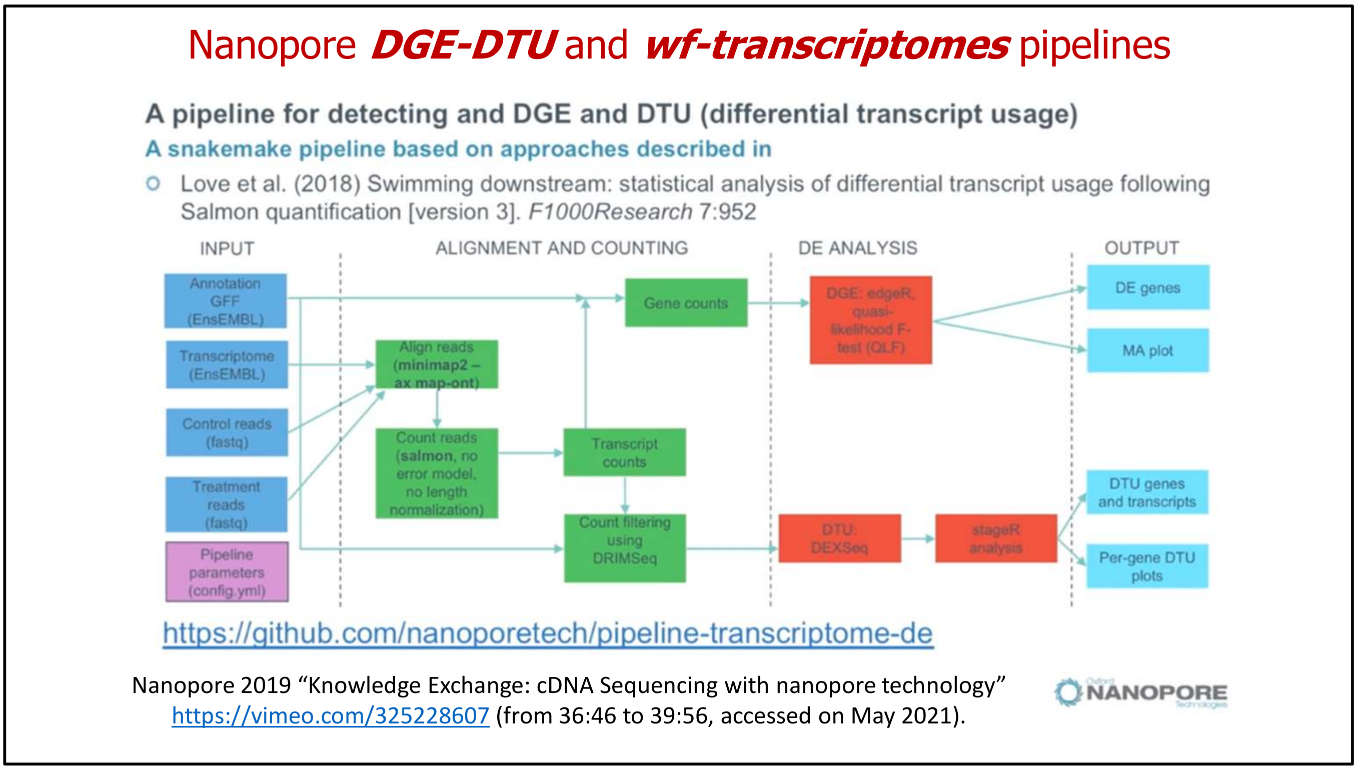
https://nf-co.re/nanoseq

This slide illustrates *Nanoseq* ***pipeline*** developed by an international consortium of academic collaborators and available in the **Nfcore** reporsitory.

The pipeline can perform multiple trajectories of Nanopore data analysis including DNA and RNA sequencing.

Amongst other options, for RNA sequencing this workflow includes
- QC and preprocessing
- Alignment (using minimap2 or graphmap2)
- Transcriptome assembly (using bambu or stringtie2)
- Differential gene and differential transcript use analysis (DESEq2 and DEXSeq)
- Fusion detection (JAFFAL)

The pipeline downloads and installs the required tools from the repository using Docker or Singularity containers.



Nanopore *DGE-DTU* and *wf-transcriptomes* pipelines

A pipeline for detecting and DGE and DTU (differential transcript usage)

A snakemake pipeline based on approaches described in

- Love et al. (2018) Swimming downstream: statistical analysis of differential transcript usage following Salmon quantification [version 3]. *F1000Research* 7:952

INPUT
ALIGNMENT AND COUNTING
DE ANALYSIS
OUTPUT
Annotation GFF (EnsEMBL)
Transcriptome (EnsEMBL)
Control reads (fastq)
Treatment reads (fastq)
Pipeline parameters (config.yml)
Align reads (minimap2 – ax map-ont)
Count reads (salmon, no error model, no length normalization)
Gene counts
Transcript counts
Count filtering using DRIMSeq
DGE: edgeR, quasi-likelihood F-test (QLF)
DTU: DEXSeq
stageR analysis
DE genes
MA plot
DTU genes and transcripts
Per-gene DTU plots

https://github.com/nanoporetech/pipeline-transcriptome-de

Nanopore 2019 “Knowledge Exchange: cDNA Sequencing with nanopore technology”
https://vimeo.com/325228607 (from 36:46 to 39:56, accessed on May 2021).

This slide illustrates the ***DGE-DTU pipeline*** that was developed by Nanopore for quantification of transcript isoforms some years ago. Initially it was implemented in ***Snakemake***. However, now it has been re-written in ***Nextflow*** and named as ***wf-transcriptomes.***

The functionality of ***wf-transcriptomes*** is a bit more narrow than of the ***Nanoseq*** workflow.

Depending on the requested options, ***wf-transcriptomes*** can
- Orient and ***make full-length reads*** by ***Pychopper***
- ***Align reads*** and ***assemble transcriptome*** for each sample (using a reference transcriptome as a guide)
- ***Count transcripts*** with ***Salmon***
- Use these counts for ***DGE*** (Differential Gene Expression) analysis with ***edgeR***
- Perform ***DTU*** (Differential Transcript Usage) analysis with ***DEXSeq*** & ***stageR***

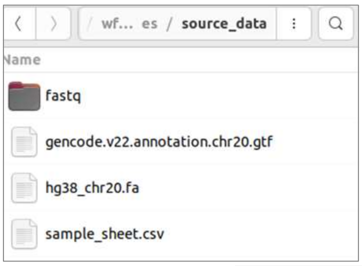
Instead of being hosted on Nfcore, this workflow is available from Nanopore github: https://github.com/epi2me-labs/wf-transcriptomes

We will run the ***wf-transcriptomes*** pipeline during our practical session.

Nanopore wf-transcriptomes pipeline

https://github.com/epi2me-labs/wf-transcriptomes

Input Folder → Output folder



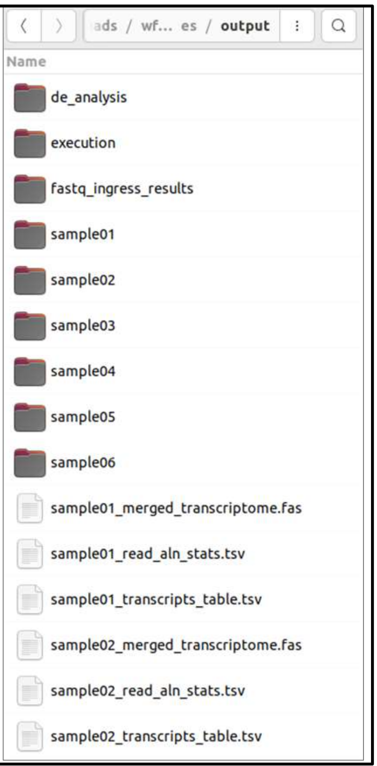
/ wf... es / source_data

Name

- fastq
- gencode.v22.annotation.chr20.gtf
- hg38_chr20.fa
- sample_sheet.csv

1) Install pipeline and tools
2) Copy data & resources in the source folder
3) Run the pipeline

```
nextflow run epi2me-labs/wf-transcriptomes \
--de_analysis \
--direct_rna \
--fastq  source_data/fastq \
--sample_sheet source_data/sample_sheet.csv \
--ref_genome source_data/hg38_chr20.fa \
--ref_annotation source_data/gencode.v22.gtf \
--minimap2_index_opts '-k 15’
-profile singularity
```



ads / wf... es / output

Name

- de_analysis
- execution
- fastq_ingress_results
- sample01
- sample02
- sample03
- sample04
- sample05
- sample06
- sample01_merged_transcriptome.fas
- sample01_read_aln_stats.tsv
- sample01_transcripts_table.tsv
- sample02_merged_transcriptome.fas
- sample02_read_aln_stats.tsv
- sample02_transcripts_table.tsv

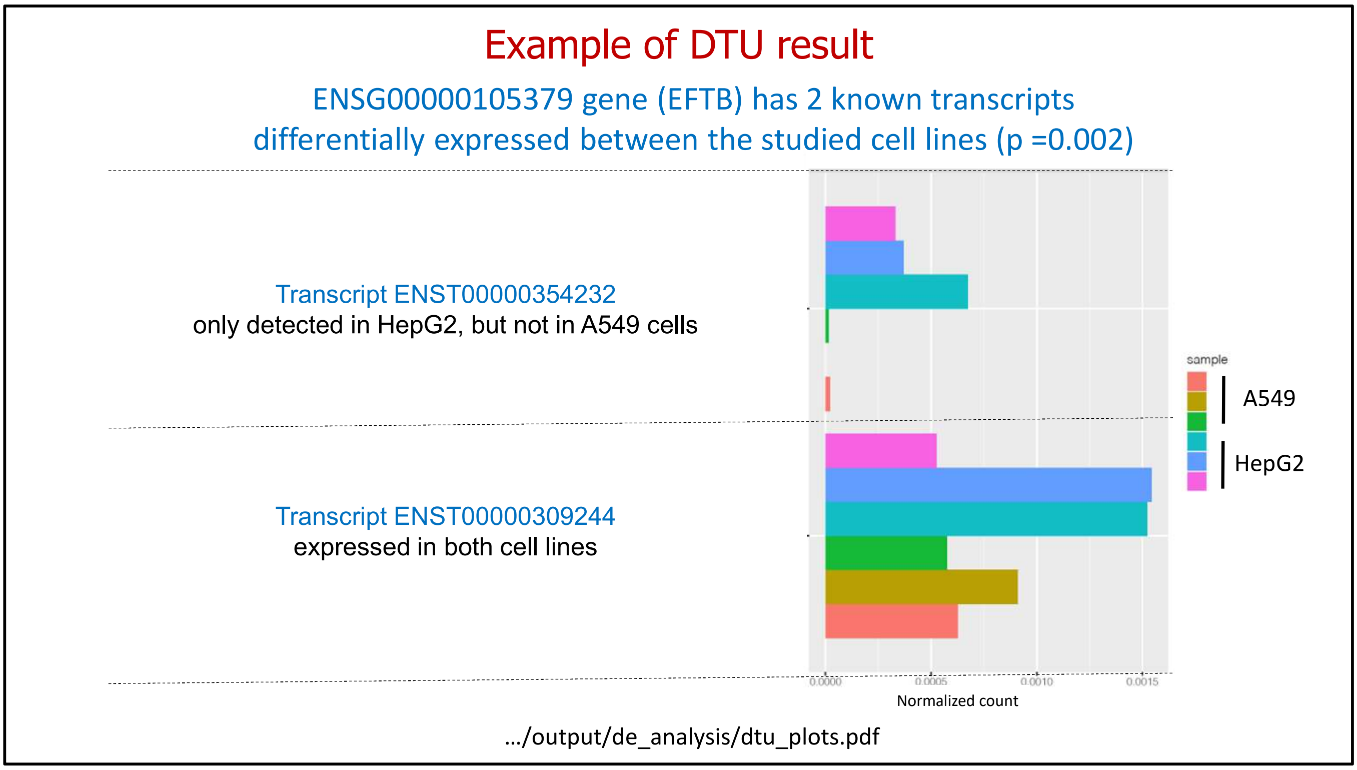
In practice, you first need to install the pipeline and tools following the authors’ recommendations.
This step may require some IT knowledge and admin rights, so this was already done on your VM.

Then, all what you need to is:
- Provide your source data and resources (FASTQ files, reference genome, etc)
- Start the pipeline

After the pipeline completes, the results will appear in the output folder.

Nanopore has developed a user-friendly GUI tool to install and run its Nextflow pipelines: EPI2ME Desktop.
EPI2ME Desktop can run the analysis locally, or can be connected to a a cloud, if the user’s machine does not have enough resources to run the analysis.

A copy of EPI2ME Desktop was installed into your VM. However, we don’t have time to learn the user interface. So, you will run the pipeline using the provided script.



Example of DTU result

ENSG00000105379 gene (EFTB) has 2 known transcripts differentially expressed between the studied cell lines (p =0.002)

Transcript ENST00000354232
only detected in HepG2, but not in A549 cells

Transcript ENST00000309244
expressed in both cell lines

…/output/de_analysis/dtu_plots.pdf

This is an example of DTU analysis produced by the pipeline.
In the ***wf-transcriptomes*** output folder you may find such plots in the …/output/de_analysis/dtu_plots.pdf file.

This example compares two cell lines: A549 and HepG2.
You can see that a specific gene (ENSG00000105379, ETFB) has two alternative transcripts (ENST00000309244 and ENST00000354232).
One of them (ENST00000309244) is strongly expressed in both cell lines.
While the other (ENST00000354232) is only expressed in HepG2.

The output folder contains much more results, including the outputs of every involved tool, such as FASTQs processed by Pychopper (if the analysis was done on cDNA), BAM files produced by minimap2, transcriptomes assembled for each sample, etc. You will be able to explore the content of the ***wf-transcriptomes*** output folder during the practical session.

Long-reads bulk RNA-seq

DONE

Long vs Short

Technology overview

- Principles: Nanopore and PacBio
- RNAseq library prep: strand switch and full length reads
- Current hardware: Machines and Flow-cells

Accuracy overview

- PacBio: Circular Consensus Sequencing (CCS)
- Nanopore: Base-callers, Pores, 2D, 1D^2, Consensus accuracy

File formats, tasks and tools overview

- File formats
- Transcript isoforms identification / Genome annotation
- PacBio Transcript Assembly: IsoSeq3, SQUANTI, TAPPAS
- Quantitative analysis: DGE / DTU
- Workflows & Manufacturer supported bioinformatics

Selected examples of ONT tools and pipelines

- ONT QC tools: NanoPack, Pychopper
- ONT quantitative analysis workflows: Nanoseq, wf-transcriptomes

Selected references

Pardo-Palacios F. *et al.* **2023**: Systematic assessment of long-read RNA-seq methods for transcript identification and quantification. bioRxiv https://www.biorxiv.org/content/10.1101/2023.07.25.550582v1

Kovaka S. *et al.* **2023:** Approaching complete genomes, transcriptomes and epi-omes with accurate long-read sequencing. Nat Methods https://doi.org/10.1038/s41592-022-01716-8

Foord C. *et al.* **2023:** The variables on RNA molecules: concert or cacophony? Answers in long-read sequencing Nat Methods https://doi.org/10.1038/s41592-022-01715-9

Logsdon G. *et al.* **2020:** Long-read human genome sequencing and its applications. *Nat Rev Genet*. https://doi.org/10.1038/s41576-020-0236-x

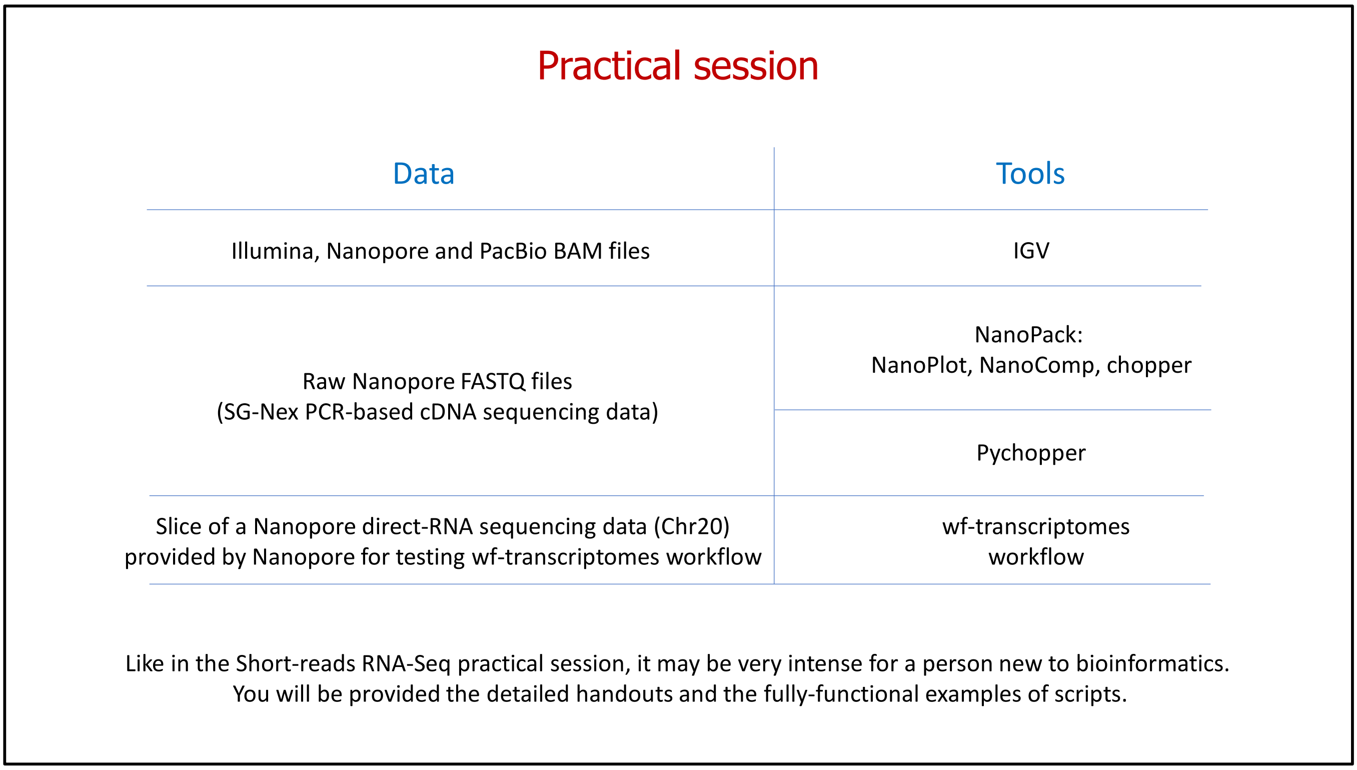
Amarasinghe S. *et al.* **2020:** Opportunities and challenges in long-read sequencing data analysis. *Genome Biol.* https://doi.org/10.1186/s13059-020-1935-5

Stark R. *et al.* **2019:** RNA sequencing: the teenage years. *Nat Rev Genet.* https://doi.org/10.1038/s41576-019-01

Wenger A. *et al.* **2019:** Accurate circular consensus long-read sequencing improves variant detection and assembly of a human genome. *Nat Biotechnol* https://doi.org/10.1038/s41587-019-0217-9

Love M. et al **2018:** Swimming downstream: statistical analysis of differential transcript usage following Salmon quantification *F1000Research,* https://doi.org/10.12688/f1000research.15398.3

See more references on slides.



Practical session

Data	Tools
Illumina, Nanopore and PacBio BAM files	IGV
Raw Nanopore FASTQ files (SG-Nex PCR-based cDNA sequencing data)	NanoPack: NanoPlot, NanoComp, chopper
	Pychopper
Slice of a Nanopore direct-RNA sequencing data (Chr20) provided by Nanopore for testing wf-transcriptomes workflow	wf-transcriptomes workflow

Like in the Short-reads RNA-Seq practical session, it may be very intense for a person new to bioinformatics. You will be provided the detailed handouts and the fully-functional examples of scripts.

This is the plan of our practical session.

All data files are sub-sampled to reduce size, so don't over-interpret data quality and biology during the practical session :)