Introduction to the Linux command line, NGS data formats, read mapping and alignments.

Advanced Bioinformatics

Lecturer: Fernando Pozo

fpozoc@cnio.es

Bioinformatics Unit,
Spanish National Cancer Research Centre (CNIO)

Tuesday 3rd September, 2019

Universidad Francisco de Vitoria UFV Madrid

Schedule

Universidad Francisco de Vitoria UFV Madrid

Next-generation sequencing (NGS)

Universidad Francisco de Vitoria UFV Madrid

Why next-generation sequencing?

Sanger vs. NGS

- For almost 30 years, sequencing of DNA has largely been dependent on the 1st generation Sanger dideoxy sequencing method.
- Sanger sequencing requires each sequence read to be amplified and read individually. Despite considerable improvements in automation and throughput, Sanger sequencing remains relatively expensive and labor intensive.

In both technologies:

- DNA Polymerase adds fluorescent nucleotides one by one onto a growing DNA template strand.
- Each incorporated nucleotide is identified by its fluorescent tag.

Universidad Francisco de Vitoria UFV Madrid

Why next-generation sequencing?

Sanger vs. NGS

- The critical difference between Sanger sequencing and NGS is **sequencing volume**.
- While the Sanger method only sequences a single DNA fragment at a time, **NGS is massively parallel, sequencing millions** of fragments simultaneously per run.
- NGS high-throughput process translates into sequencing hundreds to thousands of genes at one time.

Universidad Francisco de Vitoria UFV Madrid

Why next-generation sequencing?

Sanger vs. NGS

"With Sanger sequencing, we saw a limited DNA snapshot... NGS and its massively parallel sequencing enable us to look at tens to hundreds of thousands of reads per sample."

Professor, Head of TrEnD laboratory, Curtin University

Universidad Francisco de Vitoria UFV Madrid

Why next-generation sequencing?

Sanger vs. NGS

	Sanger Sequencing	Targeted NGS
Benefits	• Fast, cost-effective sequencing for low numbers of targets (1–20 targets) • Familiar workflow	• Higher sequencing depth enables higher sensitivity (down to 1%) • Higher discovery power* • Higher mutation resolution† • More data produced with the same amount of input DNA‡ • Higher sample throughput
Challenges	• Low sensitivity (limit of detection ~15–20%) • Low discovery power • Not as cost-effective for high numbers of targets (> 20 targets) • Low scalability due to increasing sample input requirements	• Less cost-effective for sequencing low numbers of targets (1–20 targets) • Time-consuming for sequencing low numbers of targets (1–20 targets)

* Discovery power is the ability to identify novel variants.
† Mutation resolution is the size of the mutation identified. NGS can identify large chromosomal rearrangements down to single nucleotide variants.
‡ 10 ng DNA will produce ~1 kb with Sanger sequencing or ~300 kb with targeted resequencing (250 bp amplicon length × 1536 amplicons with TruSeq Custom Amplicon workflow)

Figure: Challenges and Benefits of Sanger Sequencing and NGS

Why next-generation sequencing?

Sanger vs. NGS

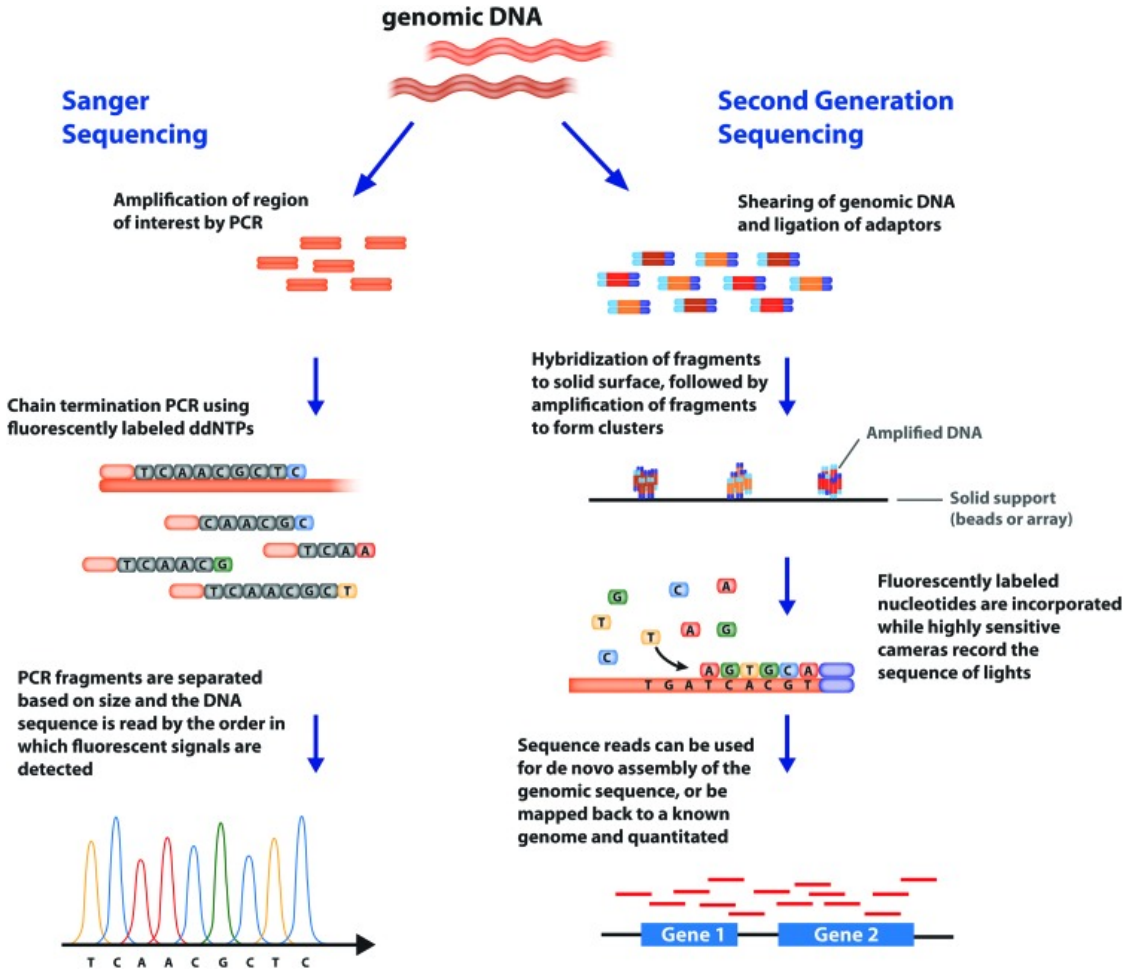
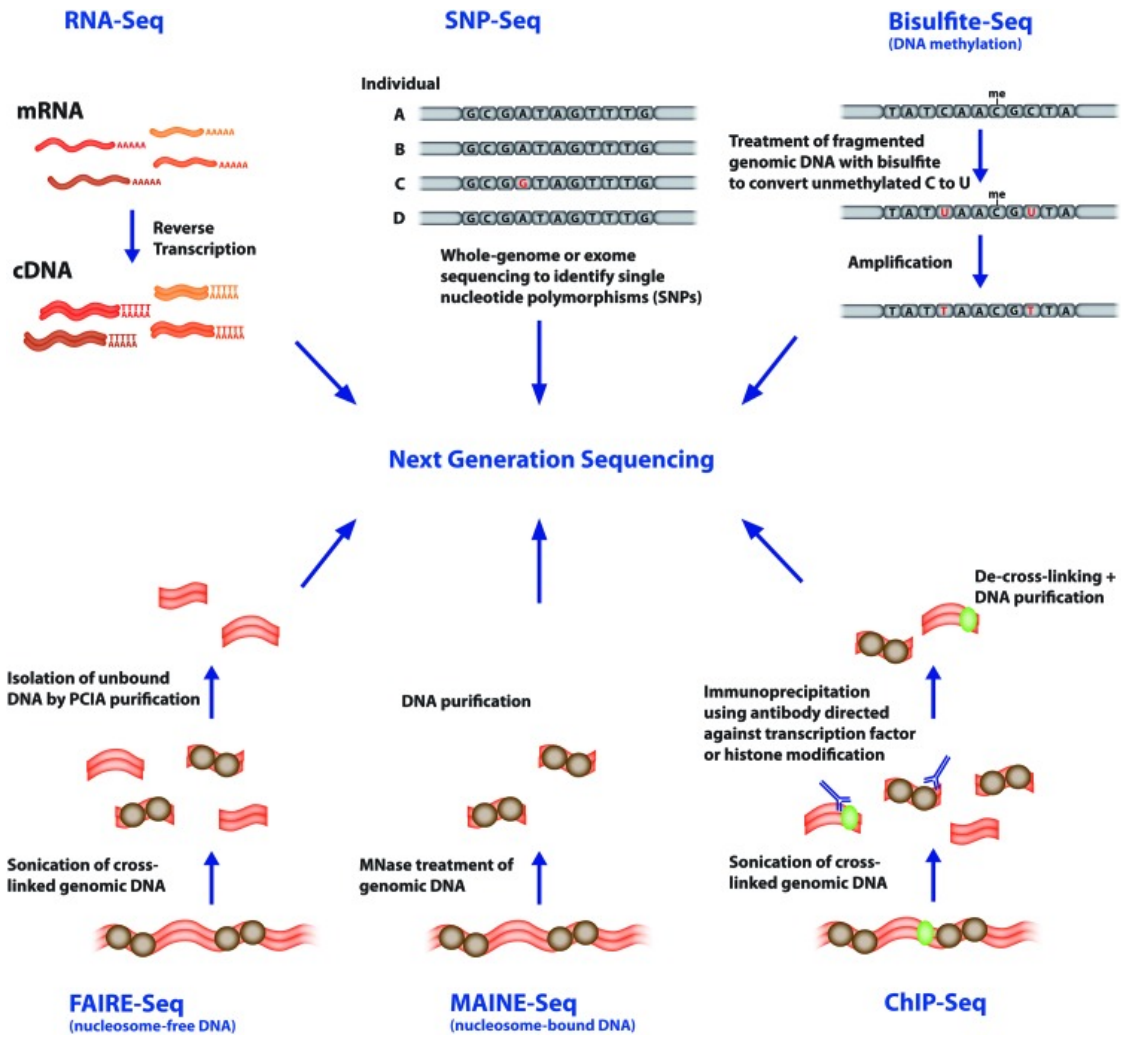


Figure: NGS platforms can sequence millions of DNA fragments in parallel in one reaction

Universidad Francisco de Vitoria UFV Madrid

Why next-generation sequencing?

Sanger vs. NGS



Universidad Francisco de Vitoria UFV Madrid

Figure: The types of experiments that can be performed using NGS are many fold

NGS platforms

- Roche 454 platform (Roche Life Sciences).
- Applied Biosystems SOLiD platform (Applied Biosystems). CIFAR-10
- Illumina Genome Analyzer (formerly known as Solexa) and HiSeq platforms (Illumina).
- Ion Torrent (Termofisher).
- 3rd generation sequencing (Single molecule level & Longer Reads):
  - PacBio Sequencing (PacBio)
  - MinION (Oxford Nanopore).

Universidad Francisco de Vitoria UFV Madrid

NGS protocol design

Quality Scores

Measure the probability that a base is called incorrectly. It uses the phred-like algorithm (similar to that originally developed for Sanger).

Paired-End vs. Single-End

- Paired-end sequencing allows users to sequence both ends of a fragment and generate high-quality, alignable sequence data. It facilitates detection of genomic rearrangements and repetitive sequence elements, as well as gene fusions and novel transcripts.
- Producing twice the number of reads for the same time and effort in library preparation, sequences aligned as read pairs enable more accurate read alignment and the ability to detect insertion-deletion (indel) variants, which is not possible with single-read data. All Illumina next-generation sequencing (NGS) systems are capable of paired-end sequencing.

Universidad Francisco de Vitoria UFV Madrid

NGS protocol design

Multiplex Sequencing

Processing more samples in less time. Sample multiplexing exponentially increases the number of samples sequenced per run

Read Length

Number of base pairs (bp) sequenced from a DNA fragment. The right sequencing read length depends on your sample type, application, and coverage requirements. Examples:

- Long reads: *de novo* assembly and resolving repetitive areas of the genome with greater confidence.
- Other applications: shorter reads are sufficient and more cost-effective than longer ones

Universidad Francisco de Vitoria UFV Madrid

NGS protocol design

Read Length for Different Applications

DNA Sequencing Applications	
Application	**Recommended Read Length**
Whole-genome sequencing	2 × 150 bp
Whole-exome sequencing	2 × 150 bp
Targeted enrichment sequencing	2 × 150 bp
Amplicon sequencing	Length of the entire amplicon insert
De novo sequencing	Ranges from 2 × 150 to 2 × 300 bp

Universidad Francisco de Vitoria UFV Madrid

NGS protocol design

Read Length for Different Applications

RNA Sequencing Applications	
Application	**Recommended Read Length**
Transcriptome analysis	2 × 75 bp
Gene expression profiling	1 × 50 bp
Small RNA sequencing	1 × 50 bp

NGS protocol design

Coverage

Average number of reads that align to known reference bases. Variant discovery can be made with a certain degree of confidence at particular base positions.

Sequencing Method	Recommended Coverage
Whole genome sequencing (WGS)	30× to 50× for human WGS (depending on application and statistical model)
Whole-exome sequencing	100×
RNA sequencing	Usually calculated in terms of numbers of millions of reads to be sampled. Detecting rarely expressed genes often requires an increase in the depth of coverage.
ChIP-Seq	100×

Figure: Sequencing Coverage Requirements

Universidad Francisco de Vitoria UFV Madrid

NGS protocol design

Deep Sequencing

Sequencing a Genomic region multiple times, sometimes hundreds or even thousands of times.

The case of Cancer Research: Required sequencing depth increases for low purity tumors, highly polyclonal tumors, and applications that require high sensitivity (identifying low frequency clones). Cancer sequencing depth typically ranges from 80 to up to thousands-fold coverage.

Factors Impacting Cancer Sequencing Depth:

- Purity of the tumor.
- Heterogeneity of the tumor.
- Sensivity required.

Universidad Francisco de Vitoria UFV Madrid

Sequencing technology overview

Illumina

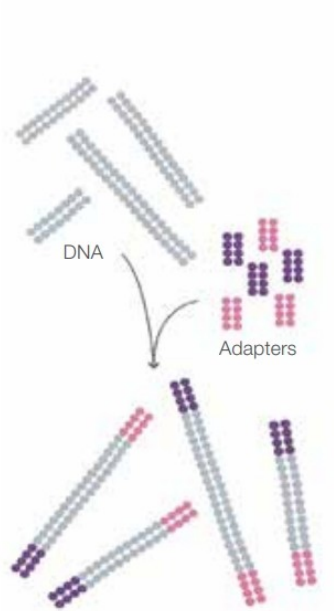
- Length is in range of 50 to 300 nt.
- It uses a glass *flowcell*, about the size of a microscope slide, with 8 separate *lanes*.

Universidad Francisco de Vitoria UFV Madrid

Sequencing technology overview

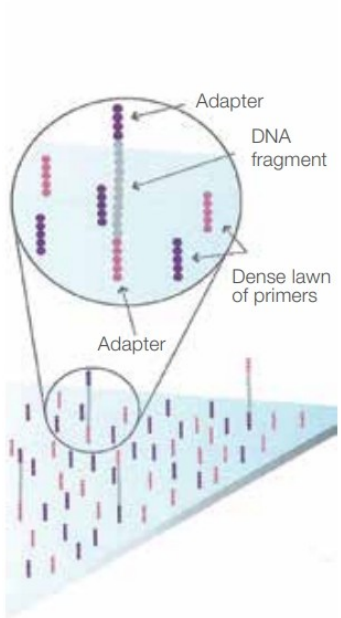
Illumina

Figure 2: Prepare Genomic DNA Sample



Randomly fragment genomic DNA and ligate adapters to both ends of the fragments.

Figure 3: Attach DNA to Surface



Bind single-stranded fragments randomly to the inside surface of the flow cell channels.

Figure: Fragment DNA, ligate adaptor and oligos

Universidad Francisco de Vitoria UFV Madrid

Sequencing technology overview

Illumina

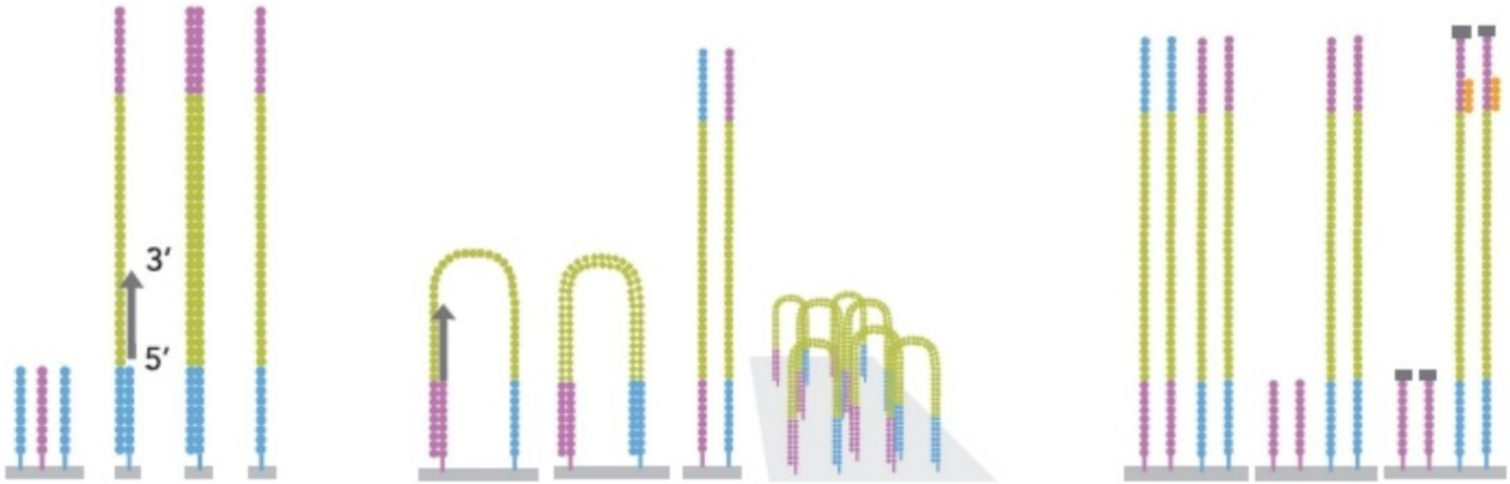


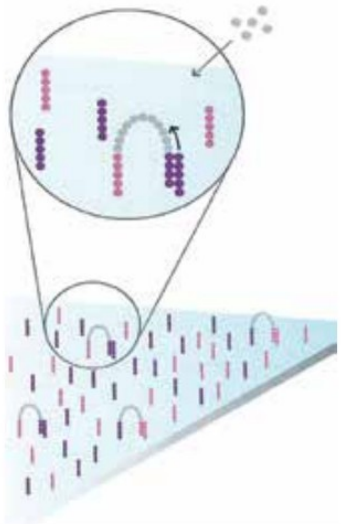
Figure: Surface-bound primers are extended by DNA polymerase across annealed ssDNA molecules, the DNA is denatured back to single strands, and the free ends of immobilized strands anneal again to oligos bound on surface of flowcell. This 'bridge PCR' continues until a cluster of approximately 1000 molecules is produced on the surface of the flowcell, all descended from the single molecule that bound at that site. After PCR, the free ends of all DNA strands are blocked.

Universidad Francisco de Vitoria UFV Madrid

Sequencing technology overview

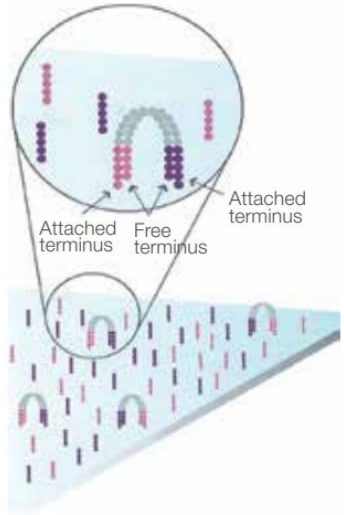
Illumina

Figure 4: Bridge Amplification



Add unlabeled nucleotides and enzyme to initiate solid-phase bridge amplification.

Figure 5: Fragments Become Double Stranded



The enzyme incorporates nucleotides to build double-stranded bridges on the solid-phase substrate.

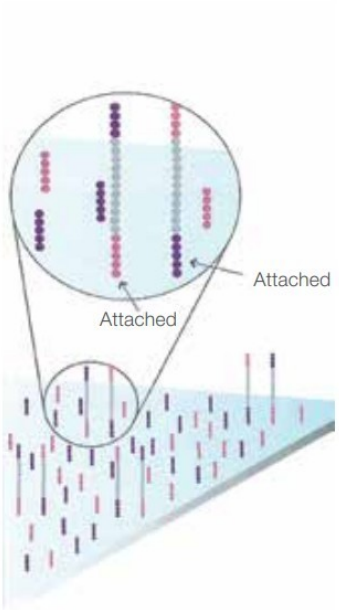
Figure: Denaturalization and lusters of products

Universidad Francisco de Vitoria UFV Madrid

Sequencing technology overview

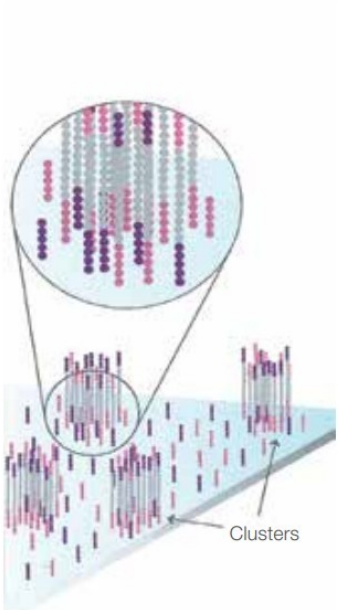
Illumina

Figure 6: Denature the Double-Standed Molecules



Denaturation leaves single-stranded templates anchored to the substrate.

Figure 7: Complete Amplification



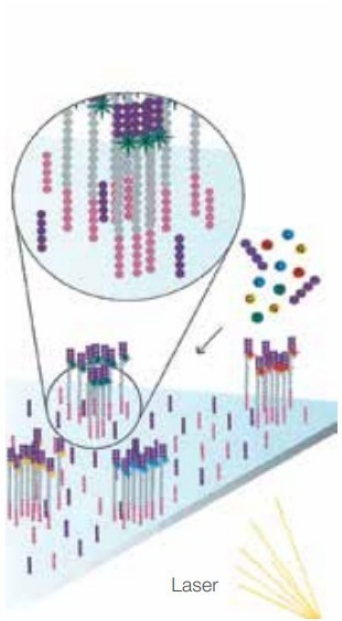
Several million dense clusters of double-stranded DNA are generated in each channel of the flow cell.

Universidad Francisco de Vitoria UFV Madrid

Sequencing technology overview

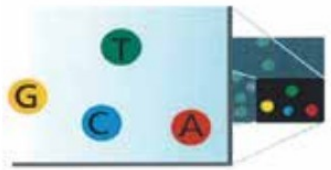
Illumina

Figure 8: Determine First Base



The first sequencing cycle begins by adding four labeled reversible terminators, primers, and DNA polymerase.

Figure 9: Image First Base



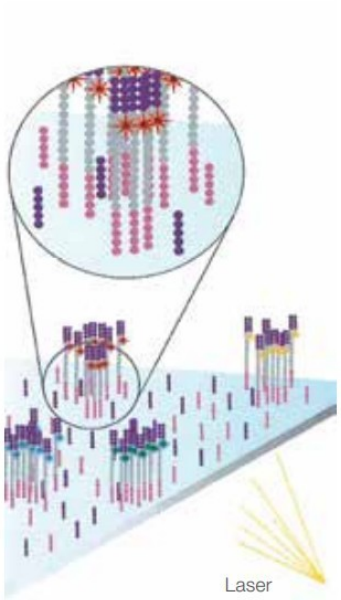
After laser excitation, the emitted fluorescence from each cluster is captured and the first base is identified.

Universidad Francisco de Vitoria UFV Madrid

Sequencing technology overview

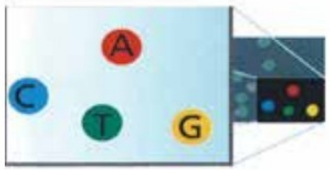
Illumina

Figure 10: Determine Second Base



The next cycle repeats the incorporation of four labeled reversible terminators, primers, and DNA polymerase.

Figure 11: Image Second Chemistry Cycle



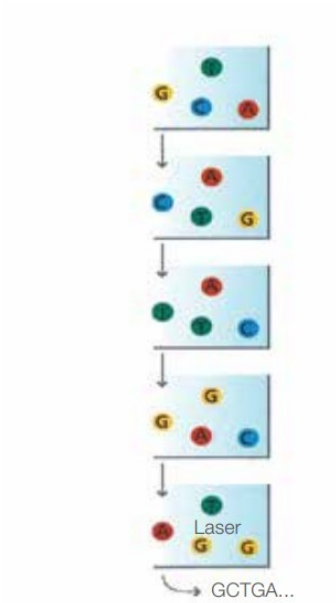
After laser excitation, the image is captured as before, and the identity of the second base is recorded.

Universidad Francisco de Vitoria UFV Madrid

Sequencing technology overview

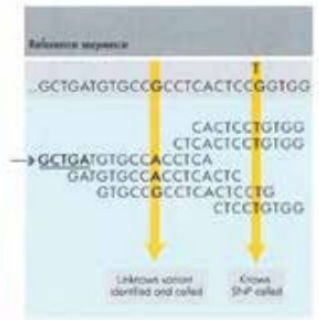
Illumina

Figure 12: Sequencing Over Multiple Chemistry Cycles



The sequencing cycles are repeated to determine the sequence of bases in a fragment, one base at a time.

Figure 13: Align Data



The data are aligned and compared to a reference, and sequencing differences are identified.

Universidad Francisco de Vitoria UFV Madrid

NGS File Formats

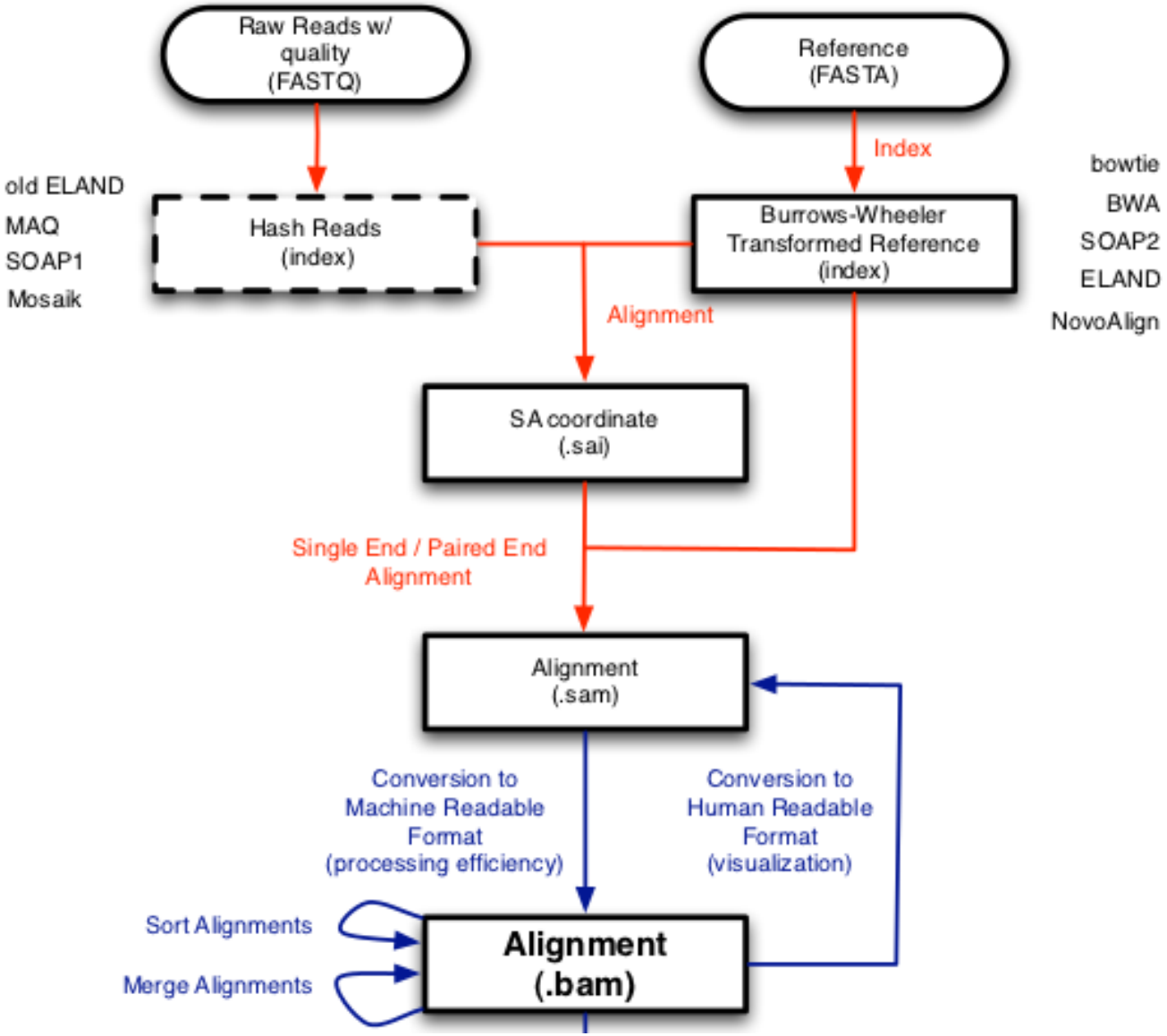
Universidad Francisco de Vitoria UFV Madrid

NGS Bioinformatics Pipeline

- Quality Control of FASTQ sentence files.
- Read mapping against some Reference Genome.
- Analysis of the mapped reads:
  - Variant Calling (Exome, genome...)
  - Differential Expression (RNA-seq)
  - Peak calling (ChIP-seq)
- Visualization.
- Biomedical interpretation.

Universidad Francisco de Vitoria UFV Madrid

NGS Bioinformatics Pipeline



Universidad Francisco de Vitoria UFV Madrid

NGS File Formats

- Many different file formats that reflect the various steps of analysis.
- We are going to introduce the most common formats today.
- Some of them are going to be used in our Hands-On
- Remaining lectures will fill in details and lead into another types of analysis with another formats.

Universidad Francisco de Vitoria UFV Madrid

NGS File Formats

Sequence data output format

- DNA sequence data are typically provided with *quality scores*, either as paired files or combined in a FASTQ file.
- In separate files, DNA sequences are in FASTA format

```
>FQSOZHZ01ASD8U rank=0159502 x=206.0 y=1164.5 length=65
TACCTCTCCGCGTAGGCGCTCGTTGGTCCAGCAGAGGCGGCCGCCTTCGTCGCGAGCAGAATAGG
```

- and quality scores are numbers from 0 to 40 (SCARF format)

```
>FQSOZHZ01ASD8U rank=0159502 x=206.0 y=1164.5 length=65
37 28 28 28 37 37 37 28 28 28 37 39 36 33 33 33 37 37 40 40 39 39
39 39 39 39 40 40 39 39 39 39 39 39 40 38 37 37 35 35 35 33 33 23
23 23 19 17 19 21 21 17 17 17 19 17 19 14 14 12 12 14 16 12 12
```

- In FASTQ format , DNA sequences look similar, but quality scores are encoded as single text characters rather than as numbers

```
@FQSOZHZ01ASD8U rank=0159502 x=206.0 y=1164.5 length=65
TACCTCTCCGCGTAGGCGCTCGTTGGTCCAGCAGAGGCGGCCGCCTTCGTCGCGAGCAGAATAGG
+
F===FFF===FHEBBBFFIIHHHHHHIIHHHHHHIGFFDDDBB88842466222424//--/1--
```

Universidad Francisco de Vitoria UFV Madrid

NGS File Formats

Understanding FASTQ format

- Most recent Illumina Sequences are reported in FASTQ format

```
@M00825:185:000000000-B5NDY:1:1101:17248:1026 1:N:0:6
NCCTTTTCCCACCCCAGCAGGAAAATCGTCAGAACCCTGCACGTTTTTCATTTCTTGTAGGTTTTTTTTTTTTCC
+
#8ACCFGGGFCFGGG7@F<<8,,@<FF;FF8;9<EFEE8E8,,CFFFFF,CCEF,,,,,,,,;,,+88@@>>>,,
@M00825:185:000000000-B5NDY:1:1101:14316:1046 1:N:0:6
CTAACTGTAGCAAAACAACTTCAAACAGAGTTAGTTAAAGCCGGTTTTGAAGTGTTAATGACAATTACAAATTATT
+
8A6@BFDGD9EADCCFCFFFGGAA9F@<,CEF<@FF<9C9FF77FFGG8,,,C,,,,,<,,;,,,,,<,,,;,,<,
@M00825:185:000000000-B5NDY:1:1101:17565:1052 1:N:0:6
CCAGAAATCGTTAATATCGAAACCAACCCAGCGCTTCTACGTGTGACATCACCGCTCTCCATTATTTCCCTTTTCC
+
8866-@-EE;FFC<EAFG@88,CFC@CEF>8FCEFEFG9F<F,C89F9CC,CE7++8,,:,<,,9,,<,999,:9,
@M00825:185:000000000-B5NDY:1:1101:10843:1053 1:N:0:6
CTAAAAGTTACTTCTTCTGCTTCTATGGCAGATTTTTATGGTGTTTCCGAACAACAAATCTTTCTCCTTTTCTTCT
+
```

1. Read identifier
   - Unique instrument name
   - Run id
   - Flowcell id
   - Flowcell lane
   - Number within the flowcell lane
   - x-coordinate of the cluster within the tile
   - y-coordinate of the cluster within the tile
   - the member of a pair, 1 or 2 (paired-end or mate-pair reads only)
   - Y if the read is filtered, N otherwise
   - 0 when none of the control bits are on, otherwise it is an even number
   - sample number

Universidad Francisco de Vitoria UFV Madrid

NGS File Formats

Understanding FASTQ format

- Most recent Illumina Sequences are reported in FASTQ format
  1. Read identifier
  2. Raw sequence reported by the machine
  3. '+' (can optionally include a sequence description)
  4. The FASTQ format encodes PHRED scores as ASCII characters alongside the read sequences.
- Quality scores are numbers which represent the probability that the given base call is an error.
- These probabilities are always less than 1, so the value is given as -10x(log10) of the probability.
- An error probability of 0.001 is represented as a quality of score of 30.
- The numbers are converted into text characters so they occupy less space. A single character is as meaningful as 2 numbers plus a space between adjacent values.

Universidad Francisco de Vitoria UFV Madrid

NGS File Formats

Understanding FASTQ format

- Unfortunately, at least 4 different ways of **converting numbers** to characters have been widely used, and **header line formats** have also changed, so one aspect of data analysis is knowing what you have.

```
SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS..................................................
RRRRRRRRR...........................................................................
  ........................XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX.....................
  ............................IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII..........................
  ..............................JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ.....................
  !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
  |                         |    |        |                              |                     |
 33                        59   64       73                            104                   126

 S - Sanger, Illumina 1.8+ Phred+33,  raw reads typically (0, 40)
 R - Roche 454             Phred, numeric coding typically (0, 40)
 X - Solexa                Solexa+64, raw reads typically (-5, 40)
 I - Illumina 1.3+         Phred+64,  raw reads typically (0, 40)
 J - Illumina 1.5+         Phred+64,  raw reads typically (3, 40) - 0,1,2 = reserved
```

Figure: Reference table

NGS File Formats

Understanding FASTQ format

- Unfortunately, at least 4 different ways of **converting numbers** to characters have been widely used, and **header line formats** have also changed, so one aspect of data analysis is knowing what you have.

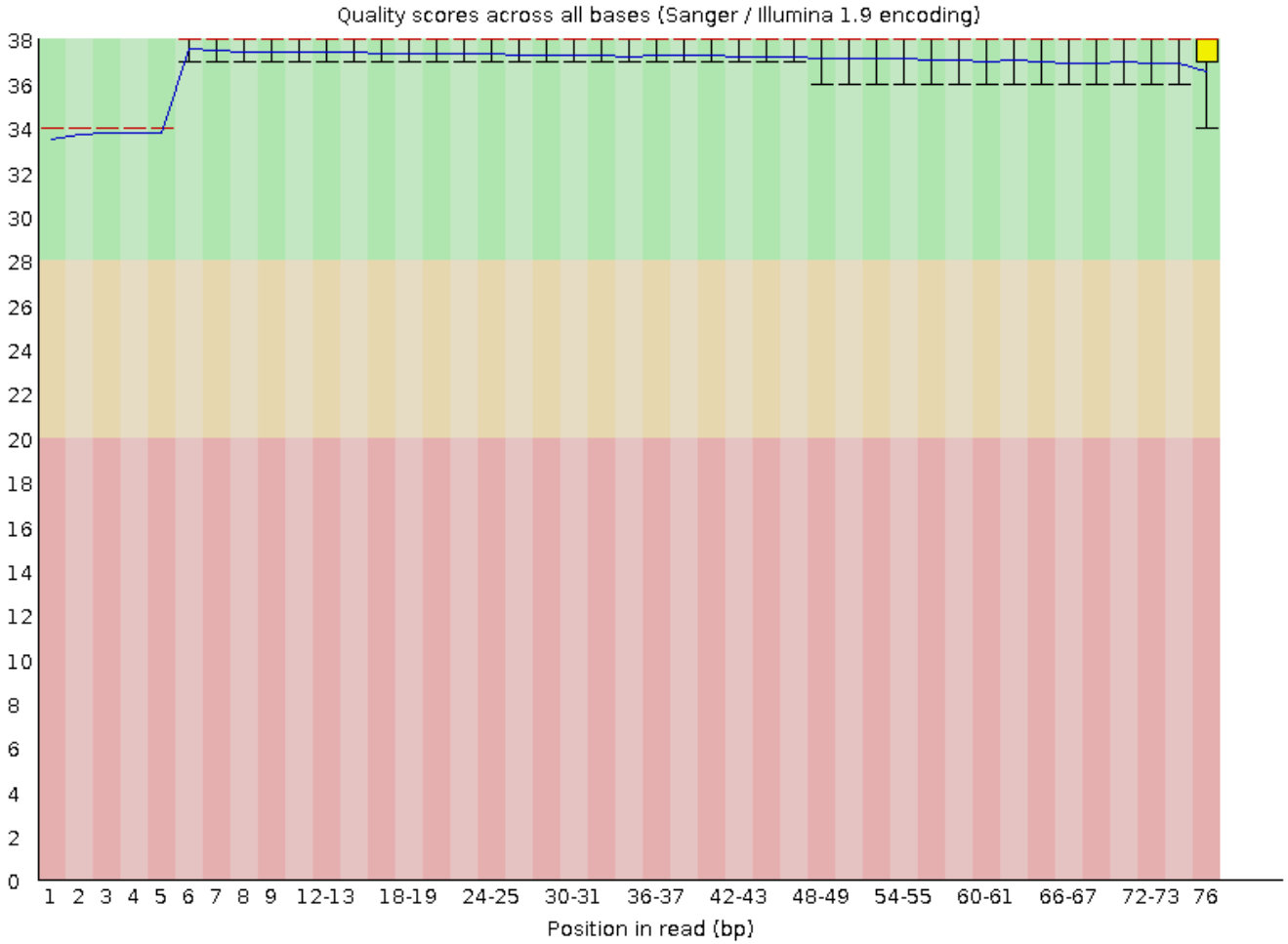
Qphred	Solexa 1.3+ offset=64	Sanger offset=33	454 Numeric Q	P	%
0	@	!	0	1.00000	100.000%
1	A	"	1	0.79433	79.433%
2	B	#	2	0.63096	63.096%
3	C	$	3	0.50119	50.119%
4	D	%	4	0.39811	39.811%
5	E	&	5	0.31623	31.623%
6	F	'	6	0.25119	25.119%
7	G	(	7	0.19953	19.953%
8	H	)	8	0.15849	15.849%
9	I	*	9	0.12589	12.589%
10	J	+	10	0.10000	10.000%
11	K	,	11	0.07943	7.943%
12	L	-	12	0.06310	6.310%
13	M	.	13	0.05012	5.012%
14	N	/	14	0.03981	3.981%
15	O	0	15	0.03162	3.162%
16	P	1	16	0.02512	2.512%
17	Q	2	17	0.01995	1.995%
18	R	3	18	0.01585	1.585%
19	S	4	19	0.01259	1.259%
20	T	5	20	0.01000	1.000%
21	U	6	21	0.00794	0.794%
22	V	7	22	0.00631	0.631%
23	W	8	23	0.00501	0.501%
24	X	9	24	0.00398	0.398%
25	Y	:	25	0.00316	0.316%
26	Z	;	26	0.00251	0.251%
27	[	<	27	0.00200	0.200%
28	\	=	28	0.00158	0.158%
29	]	>	29	0.00126	0.126%
30	^	?	30	0.00100	0.100%
31	_	@	31	0.00079	0.079%
32	`	A	32	0.00063	0.063%
33	a	B	33	0.00050	0.050%
34	b	C	34	0.00040	0.040%
35	c	D	35	0.00032	0.032%
36	d	E	36	0.00025	0.025%
37	e	F	37	0.00020	0.020%
38	f	G	38	0.00016	0.016%
39	g	H	39	0.00013	0.013%
40	h	I	40	0.00010	0.010%

Figure: Reference table

Universidad Francisco de Vitoria UFV Madrid

NGS File Formats

Understanding FASTQ format



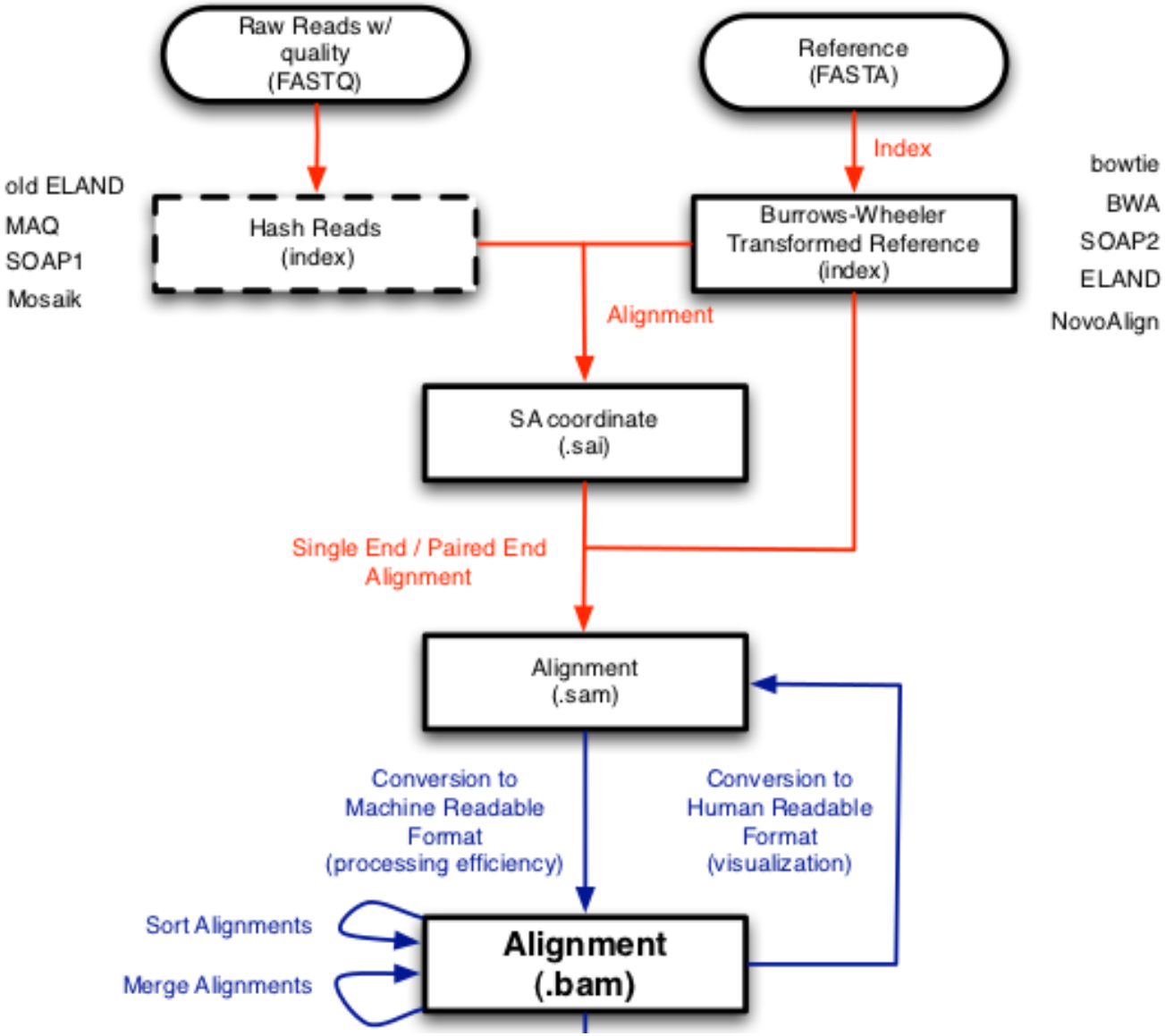
Universidad Francisco de Vitoria UFV Madrid

Figure: Practical per base quality view generated with FastQC package

Read Mapping and Alignments

Universidad Francisco de Vitoria UFV Madrid

NGS Bioinformatics Pipeline



Universidad Francisco de Vitoria UFV Madrid

Read Mapping and Alignments

Once high-quality data are obtained from preprocessing, the next step is the read mapping or alignment.

There are two main options depending on the availability of a genome sequence

- When studying an organism with a **reference genome**, it is possible to infer which transcripts are expressed by mapping the reads to the reference genome (**genome mapping**) or transcriptome (**transcriptome mapping**). Mapping reads to the genome requires no knowledge of the set of transcribed regions or the way in which exons are spliced together. This approach allows the discovery of new, unannotated transcripts.
- When working on an organism without a reference genome, reads need to be assembled first into longer contigs (***de novo assembly***). These contigs can then be considered as the expressed transcriptome to which reads are re-mapped for quantification. *De novo assembly* algorithms are constructed with de Bruijn graphs.

Universidad Francisco de Vitoria UFV Madrid

Read Mapping and Alignments

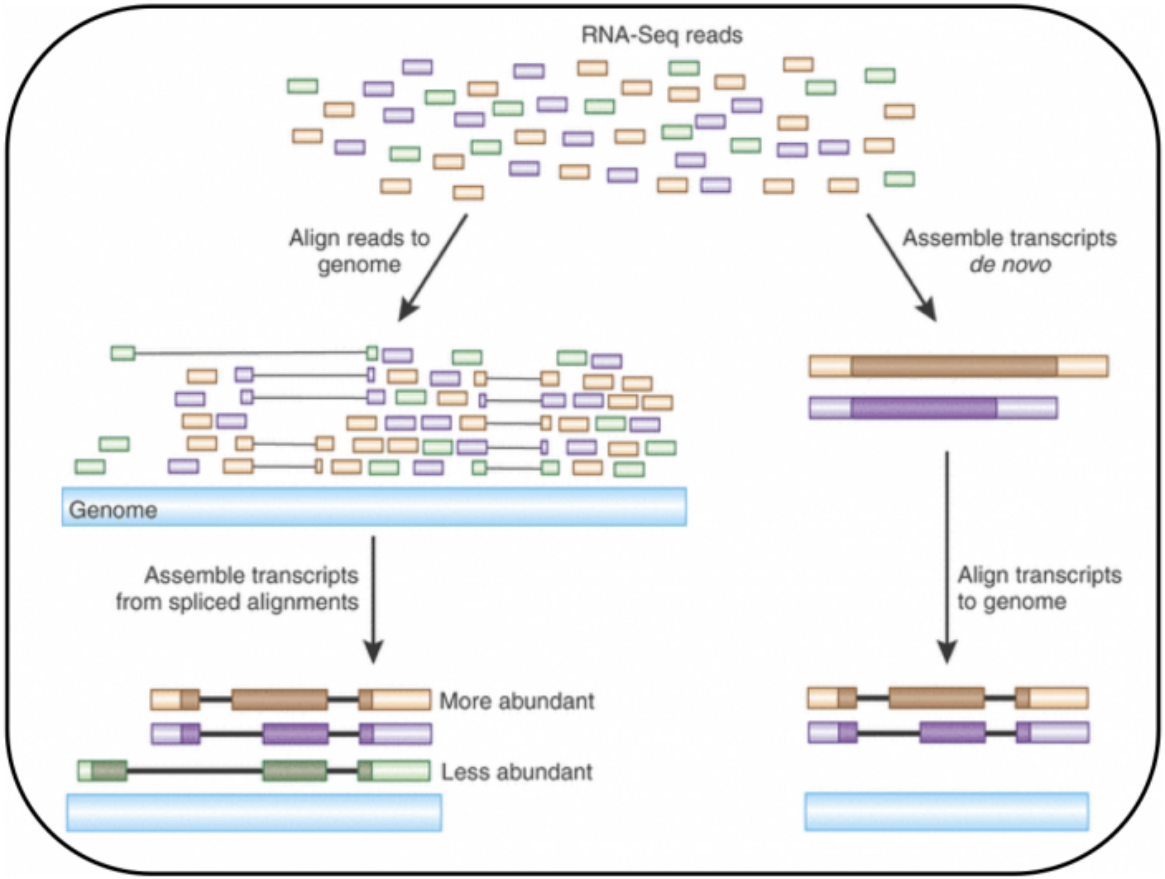
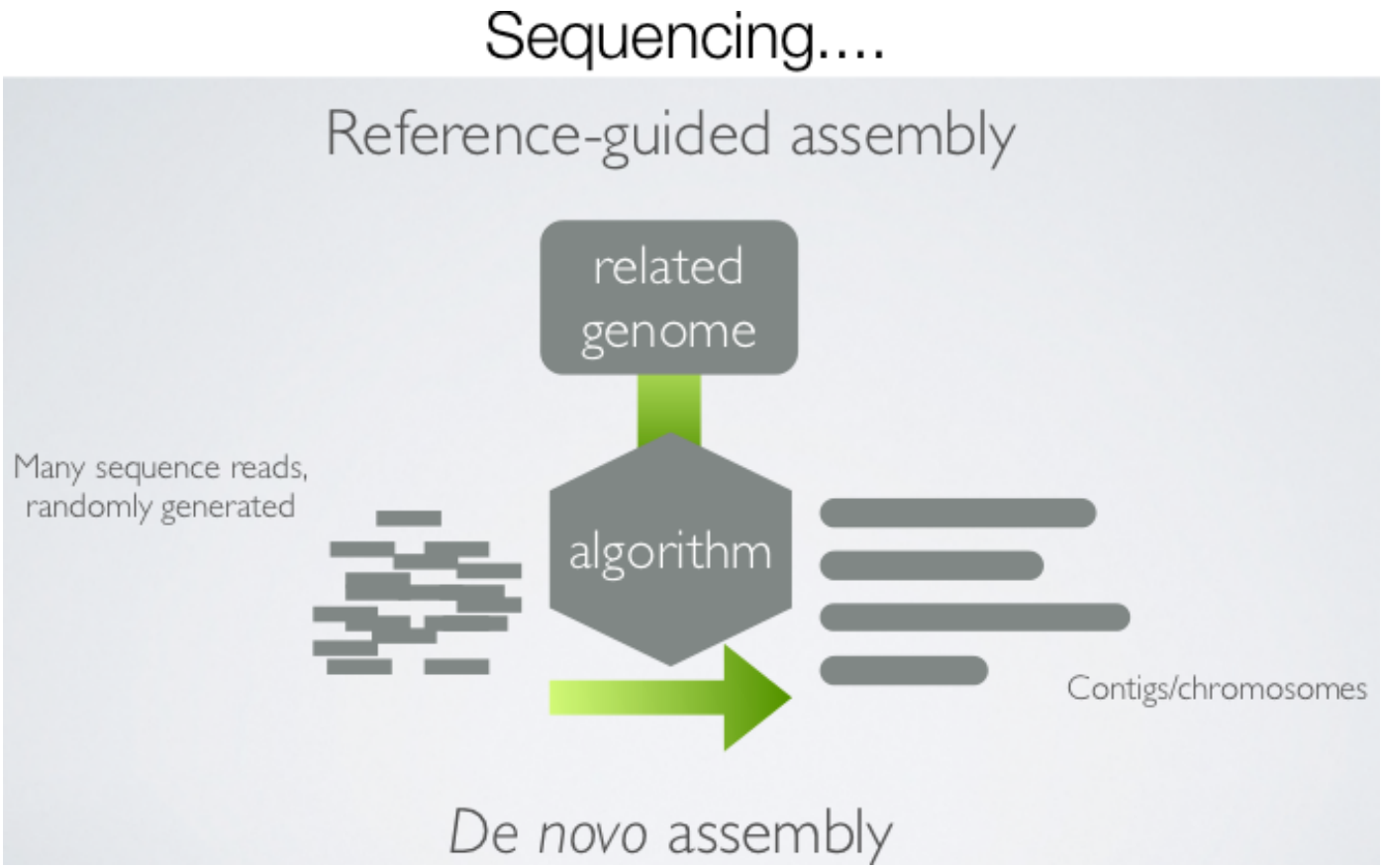


Figure: Mapping reads to a reference or de novo assembly

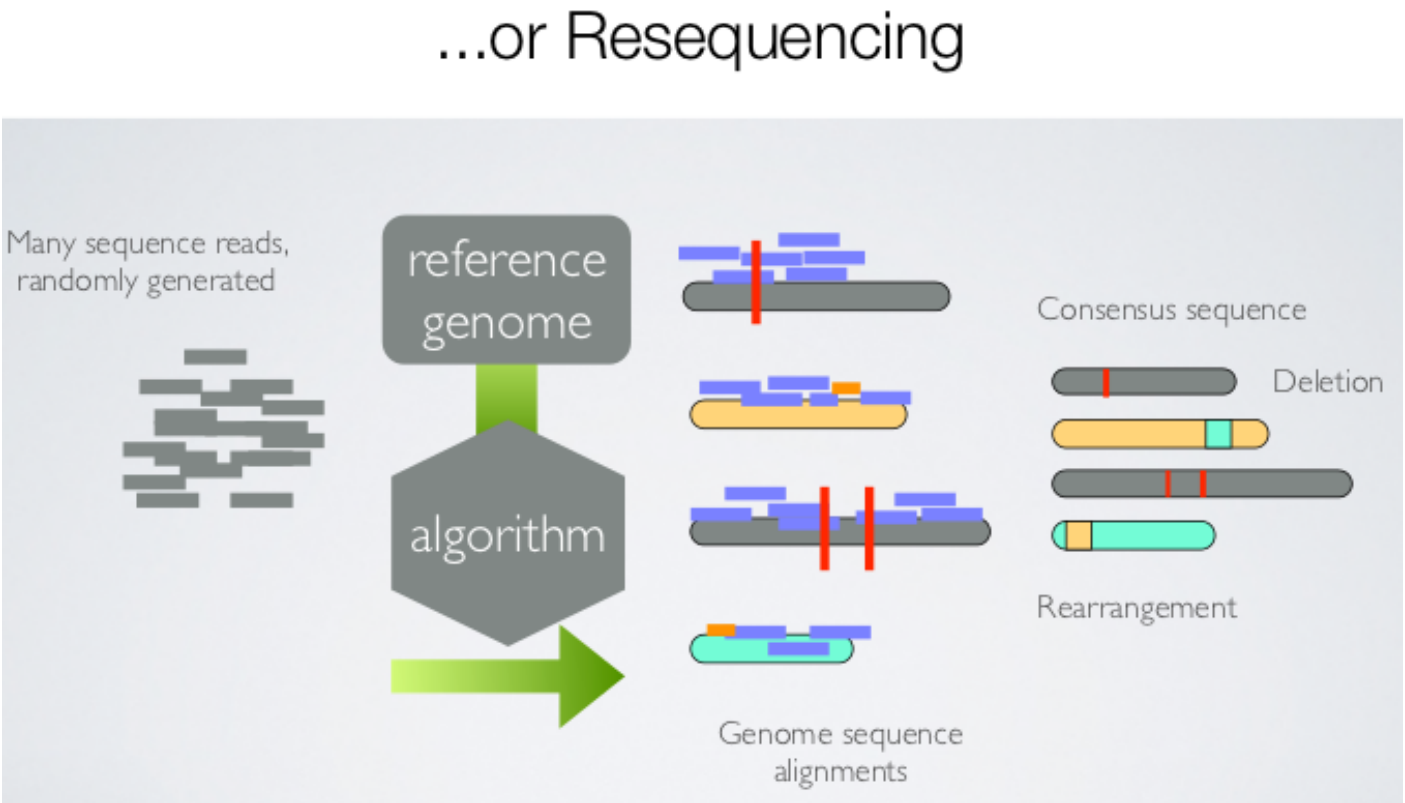
Universidad Francisco de Vitoria UFV Madrid

Read Mapping and Alignments



Universidad Francisco de Vitoria UFV Madrid

Read Mapping and Alignments



Universidad Francisco de Vitoria UFV Madrid

Read Mapping and Alignments

How to map billions of short reads onto genomes

Features supported by the tools

	Bowtie	Bowtie2	BWA	SOAP2	MAQ	RMAP	GSNAP	FANGS	Novoalign	mrFAST	mrsFAST
Seed mm.	Up to 3		Any	Up to 2	Any	Any					
Non-seed mm.	QS	AS	Count	Count	QS	Count	Count	Count	QS	Count	Count
Var. seed len.	> 5		Any	> 28							
Mapping qual.		Yes	Yes		Yes				Yes		
Gapped align.		Yes	Yes	PE	PE		Yes	Yes	Yes	Yes	
Colorspace	Yes		Yes		Yes				Yes		
Splicing							Yes				
SNP tolerance							Yes				
Bisulphite reads						Yes	Yes		Yes	Yes	

PE: paired-end only, mm.: mismatches, QS: base quality score, count: total count of mismatches in the read, AS: alignment score, and empty cells mean not supported.

Universidad Francisco de Vitoria UFV Madrid

Reference Based Assembly

The Burrows–Wheeler transform

Transformation				
1. Input	2. All rotations	3. Sort into lexical order	4. Take the last column	5. Output
^BANANA|	^BANANA| |^BANANA A|^BANAN NA|^BANA ANA|^BAN NANA|^BA ANANA|^B BANANA|^	ANANA|^B ANA|^BAN A|^BANAN BANANA|^ NANA|^BA NA|^BANA ^BANANA| |^BANANA	ANANA|^B ANA|^BAN A|^BANAN BANANA|^ NANA|^BA NA|^BANA ^BANANA| |^BANANA	BNN^AA|A

Figure: The **Burrows–Wheeler transform** (BWT, also called block-sorting compression) rearranges a character string into runs of similar characters. This is useful for compression, since it tends to be easy to compress a string that has runs of repeated characters by techniques such as move-to-front transform and run-length encoding. More importantly, the transformation is **reversible**, without needing to store any additional data except the position of the first original character. The BWT is thus a "free" method of improving the efficiency of text compression algorithms, costing only some extra computation.

Universidad Francisco de Vitoria UFV Madrid

Reference Based Assembly

The Burrows–Wheeler transform

- When a string of characters is transformed by the BWT, none of its characters change the value (it is a **lossless compression algorithm**).
- The transformation changes the order of the characters. If the original string had several **substrings** that occurred frequently, then the transformed string has several sites where a single character is repeated consecutively.
- This is very useful in **compression**: it is easier to compress a string that has several characters repeated together with techniques such as RLE encoding (run-length encoding).

Universidad Francisco de Vitoria UFV Madrid

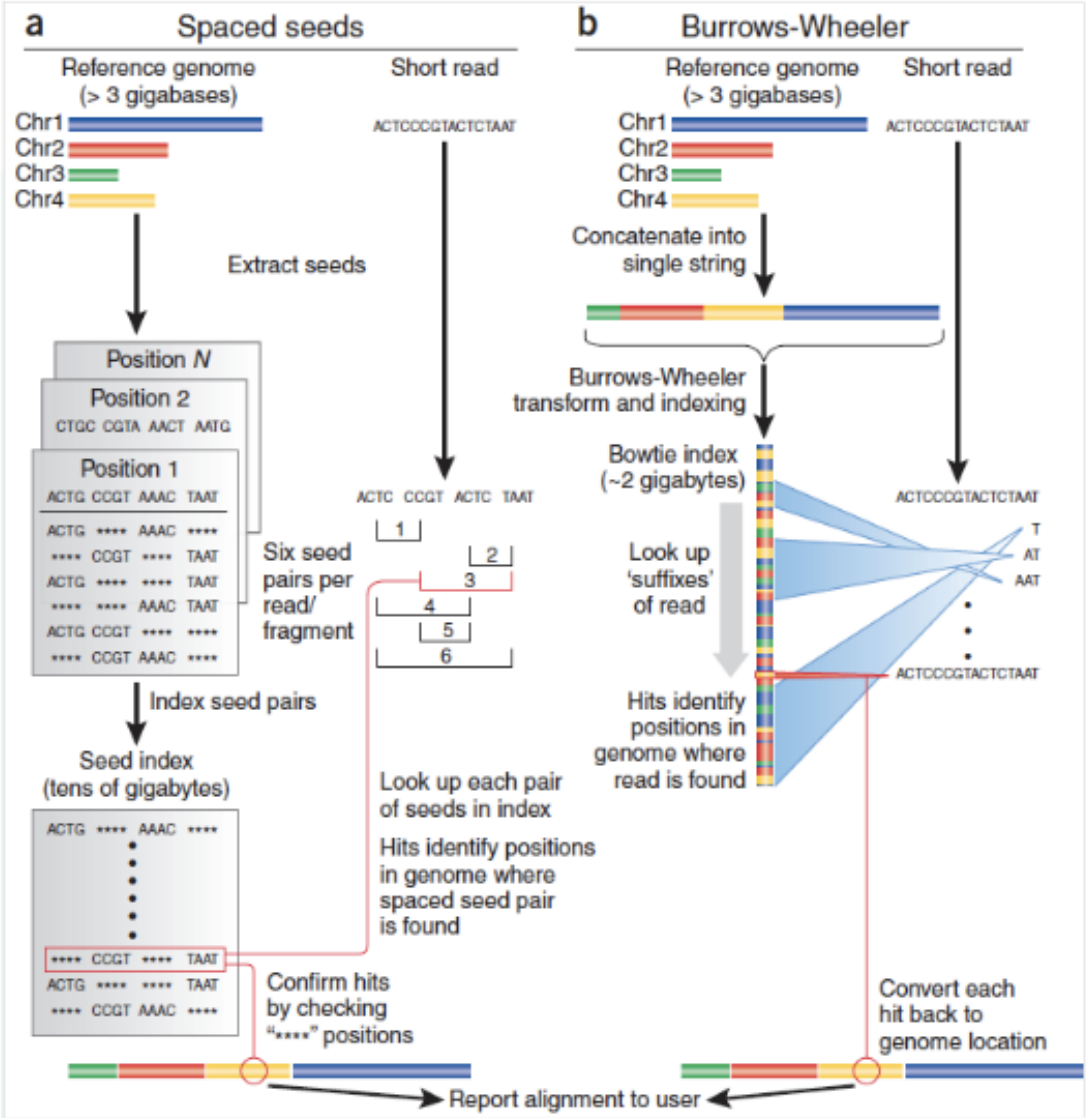
Reference Based Assembly

The Burrows–Wheeler transform

Figure: BWT example with DNA. From 54 to 45 characters with this transformation

Universidad Francisco de Vitoria UFV Madrid

Reference Based Assembly



NGS File Formats

SAM (Sequence Alignment/Map) format

- **SAM (Sequence Alignment/Map)** is a generic format for storing large nucleotide sequence alignments.
- SAM is is the human readable, scriptable format. A **BAM** file is essentially a binary (gzip-compressed) version of a SAM file.
- SAM/BAM files are usually sorted and indexed to streamline data processing. Both contains exactly the same information, and are interconvertible

```
@HD     VN:1.0  SO:unsorted
@SQ     SN:NC_003210.1  LN:2944528
@PG     ID:bowtie2      PN:bowtie2      VN:2.3.5.1      CL:"/SOFTWARE/bowtie2-2.3.5.1-sra-linux-x86_64/bowtie2-align-s --wrapp
M00825:185:000000000-B5NDY:1:1101:19478:1210    83      NC_003210.1     240679  1       58M     =       240621  -116    GTAGGC
M00825:185:000000000-B5NDY:1:1101:19478:1210    163     NC_003210.1     240621  1       4M      =       240679  116     TCCT
M00825:185:000000000-B5NDY:1:1101:11002:1224    99      NC_003210.1     514004  42      60M     =       514182  183     ATGAAA
M00825:185:000000000-B5NDY:1:1101:11002:1224    147     NC_003210.1     514182  42      5M      =       514004  -183    GGAAA
M00825:185:000000000-B5NDY:1:1101:13862:1242    83      NC_003210.1     1707300 42      60M     =       1707090 -270    AGCGTC
M00825:185:000000000-B5NDY:1:1101:13862:1242    163     NC_003210.1     1707090 42      12M     =       1707300 270     TATTTT
M00825:185:000000000-B5NDY:1:1101:13352:1256    83      NC_003210.1     2728804 1       66M     =       2728778 -92     AATATG
M00825:185:000000000-B5NDY:1:1101:13352:1256    163     NC_003210.1     2728778 1       3M      =       2728804 92      TCT
M00825:185:000000000-B5NDY:1:1101:10201:1263    137     NC_003210.1     1832522 1       5M      =       1832522 0       TCCGT
M00825:185:000000000-B5NDY:1:1101:10201:1263    69      NC_003210.1     1832522 0       *       =       1832522 0       GTCCGG
M00825:185:000000000-B5NDY:1:1101:10535:1266    83      NC_003210.1     2100680 1       63M     =       2100623 -120    TTCGTC
M00825:185:000000000-B5NDY:1:1101:10535:1266    163     NC_003210.1     2100623 1       9M      =       2100680 120     TTTTTC
M00825:185:000000000-B5NDY:1:1101:21296:1267    99      NC_003210.1     10118   42      56M     =       10296   184     CTGCTG
M00825:185:000000000-B5NDY:1:1101:21296:1267    147     NC_003210.1     10296   42      6M      =       10118   -184    AGAGAA
M00825:185:000000000-B5NDY:1:1101:9728:1273     99      NC_003210.1     749337  1       56M     =       749792  460     AAGAAG
M00825:185:000000000-B5NDY:1:1101:9728:1273     147     NC_003210.1     749792  1       5M      =       749337  -460    AAGAA
M00825:185:000000000-B5NDY:1:1101:13881:1282    153     NC_003210.1     1029168 1       8M      =       1029168 0       AAAAAA
M00825:185:000000000-B5NDY:1:1101:13881:1282    101     NC_003210.1     1029168 0       *       =       1029168 0       AAACAA
M00825:185:000000000-B5NDY:1:1101:13016:1289    83      NC_003210.1     2727127 1       70M     =       2727081 -116    ATGGTC
M00825:185:000000000-B5NDY:1:1101:13016:1289    163     NC_003210.1     2727081 1       4M      =       2727127 116     TTTT
```

Figure: SAM file format sample file

Universidad Francisco de Vitoria UFV Madrid

NGS File Formats

SAM (Sequence Alignment/Map) format

```
@HD   VN:1.0      SO:coordinate
@SQ   SN:1  LN:249250621      AS:NCBI37   UR:file:/data/local/ref/GATK/human_g1k_v37.fasta      M5:1b22b98cdeb4a9304cb5d48026a85128
@SQ   SN:2  LN:243199373      AS:NCBI37   UR:file:/data/local/ref/GATK/human_g1k_v37.fasta      M5:a0d9851da00400dec1098a9255ac712e
@SQ   SN:3  LN:198022430      AS:NCBI37   UR:file:/data/local/ref/GATK/human_g1k_v37.fasta      M5:fdfd811849cc2fadebc929bb925902e5
@RG   ID:UM0098:1 PL:ILLUMINA PU:HWUSI-EAS1707-615LHAAXX-L001     LB:80 DT:2010-05-05T20:00:00-0400   SM:SD37743  CN:UMCORE
@RG   ID:UM0098:2 PL:ILLUMINA PU:HWUSI-EAS1707-615LHAAXX-L002     LB:80 DT:2010-05-05T20:00:00-0400   SM:SD37743  CN:UMCORE
@PG   ID:bwa      VN:0.5.4
@PG   ID:GATK TableRecalibration    VN:1.0.3471 CL:Covariates=[ReadGroupCovariate, QualityScoreCovariate, CycleCovariate, DinucCovariate, TileCovariate],
default_read_group=null, default_platform=null, force_read_group=null, force_platform=null, solid_recal_mode=SET_Q_ZERO, window_size_nqs=5, homopolymer_nback=7,
exception_if_no_tile=false, ignore_nocall_colorspace=false, pQ=5, maxQ=40, smoothing=1
```

Figure: SAM file format Header

```
1:497:R:-272+13M17D24M   113  1    497  37   37M  15   100338662 0
CGGGTCTGACCTGAGGAGAACTGTGCTCCGCCTTCAG   0;==-==9;>>>>>=>>>>>>>>>>>=>>>>>>>>>>
XT:A:U    NM:i:0    SM:i:37   AM:i:0    X0:i:1    X1:i:0    XM:i:0    XO:i:0
XG:i:0    MD:Z:37
19:20389:F:275+18M2D19M  99   1    17644     0    37M  =    17919     314
TATGACTGCTAATAATACCTACACATGTTAGAACCAT   >>>>>>>>>>>>>>>>>>>><<>>><<>>4::>>:<9
RG:Z:UM0098:1  XT:A:R    NM:i:0    SM:i:0    AM:i:0    X0:i:4    X1:i:0    XM:i:0
     XO:i:0    XG:i:0    MD:Z:37
19:20389:F:275+18M2D19M  147  1    17919     0    18M2D19M  =    17644     -314
GTAGTACCAACTGTAAGTCCTTATCTTCATACTTTGT   ;44999;499<8<8<<<8<<><<<<<>7<;<<<>><<
XT:A:R    NM:i:2    SM:i:0    AM:i:0    X0:i:4    X1:i:0    XM:i:0    XO:i:1
XG:i:2    MD:Z:18^CA19
9:21597+10M2I25M:R:-209  83   1    21678     0    8M2I27M   =    21469     -244
CACCACATCACATATACCAAGCCTGGCTGTGTCTTCT   <;9<<5><<<<><<<><>><<><><>><9>><><>>9>>><>
XT:A:R    NM:i:2    SM:i:0    AM:i:0    X0:i:5    X1:i:0    XM:i:0    XO:i:1
XG:i:2    MD:Z:35
```

Universidad Francisco de Vitoria UFV Madrid

Figure: SAM file format Alignment

NGS File Formats

SAM (Sequence Alignment/Map) format

Col	Field	Type	Brief description
1	QNAME	String	Query template NAME
2	FLAG	Int	bitwise FLAG
3	RNAME	String	References sequence NAME
4	POS	Int	1- based leftmost mapping POSition
5	MAPQ	Int	MAPping Quality
6	CIGAR	String	CIGAR String
7	RNEXT	String	Ref. name of the mate/next read
8	PNEXT	Int	Position of the mate/next read
9	TLEN	Int	observed Template LENgth
10	SEQ	String	segment SEQuence
11	QUAL	String	ASCII of Phred-scaled base QUALity+33

Figure: Alignment sections have 11 mandatory fields

Universidad Francisco de Vitoria UFV Madrid

NGS File Formats

SAM (Sequence Alignment/Map) format

What is a CIGAR?

The CIGAR (Compact Idiosyncratic Gapped Alignment Report) string is how the SAM/BAM format represents spliced alignments. Understanding the CIGAR string will help you **understand how your query sequence aligns to the reference genome**.

```
RefPos:     1  2  3  4  5  6  7     8  9 10 11 12 13 14 15 16 17 18 19
Reference:  C  C  A  T  A  C  T     G  A  A  C  T  G  A  C  T  A  A  C
Read:                  A  C  T  A  G  A  A  -  T  G  G  C  T
```

POS: 5

CIGAR: 3M1I3M1D2M1X2M

Op	BAM	Description
M	0	alignment match (can be a sequence match or mismatch)
I	1	insertion to the reference
D	2	deletion from the reference
N	3	skipped region from the reference
S	4	soft clipping (clipped sequences present in SEQ)
H	5	hard clipping (clipped sequences NOT present in SEQ)
P	6	padding (silent deletion from padded reference)
=	7	sequence match
X	8	sequence mismatch

Universidad Francisco de Vitoria UFV Madrid

NGS File Formats

Other interesting NGS data format files

Annotation files

GFF (General Feature Format), GTF (Gene Transfer format), GFF3 or BED (Browser Extensible Data)

```
 0  ##gff-version 3.2.1
 1  ##sequence-region ctg123 1 1497228
 2  ctg123 . gene            1000  9000  .  +  .  ID=gene00001;Name=EDEN
 3  ctg123 . TF_binding_site 1000  1012  .  +  .  ID=tfbs00001;Parent=gene00001
 4  ctg123 . mRNA            1050  9000  .  +  .  ID=mRNA00001;Parent=gene00001;Name=EDEN.1
 5  ctg123 . mRNA            1050  9000  .  +  .  ID=mRNA00002;Parent=gene00001;Name=EDEN.2
 6  ctg123 . mRNA            1300  9000  .  +  .  ID=mRNA00003;Parent=gene00001;Name=EDEN.3
 7  ctg123 . exon            1300  1500  .  +  .  ID=exon00001;Parent=mRNA00003
 8  ctg123 . exon            1050  1500  .  +  .  ID=exon00002;Parent=mRNA00001,mRNA00002
 9  ctg123 . exon            3000  3902  .  +  .  ID=exon00003;Parent=mRNA00001,mRNA00003
10  ctg123 . exon            5000  5500  .  +  .  ID=exon00004;Parent=mRNA00001,mRNA00002,mRNA00003
11  ctg123 . exon            7000  9000  .  +  .  ID=exon00005;Parent=mRNA00001,mRNA00002,mRNA00003
12  ctg123 . CDS             1201  1500  .  +  0  ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
13  ctg123 . CDS             3000  3902  .  +  0  ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
14  ctg123 . CDS             5000  5500  .  +  0  ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
15  ctg123 . CDS             7000  7600  .  +  0  ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
16  ctg123 . CDS             1201  1500  .  +  0  ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
17  ctg123 . CDS             5000  5500  .  +  0  ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
18  ctg123 . CDS             7000  7600  .  +  0  ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
19  ctg123 . CDS             3301  3902  .  +  0  ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
20  ctg123 . CDS             5000  5500  .  +  1  ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
21  ctg123 . CDS             7000  7600  .  +  1  ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
22  ctg123 . CDS             3391  3902  .  +  0  ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
23  ctg123 . CDS             5000  5500  .  +  1  ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
24  ctg123 . CDS             7000  7600  .  +  1  ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
```

Figure: GFF3 sample file

Universidad Francisco de Vitoria UFV Madrid

NGS File Formats

Other interesting NGS data format files

The Variant Call Format

VCF

```
 0  ##gff-version 3.2.1
 1  ##sequence-region ctg123 1 1497228
 2  ctg123 . gene            1000  9000  .  +  .  ID=gene00001;Name=EDEN
 3  ctg123 . TF_binding_site 1000  1012  .  +  .  ID=tfbs00001;Parent=gene00001
 4  ctg123 . mRNA            1050  9000  .  +  .  ID=mRNA00001;Parent=gene00001;Name=EDEN.1
 5  ctg123 . mRNA            1050  9000  .  +  .  ID=mRNA00002;Parent=gene00001;Name=EDEN.2
 6  ctg123 . mRNA            1300  9000  .  +  .  ID=mRNA00003;Parent=gene00001;Name=EDEN.3
 7  ctg123 . exon            1300  1500  .  +  .  ID=exon00001;Parent=mRNA00003
 8  ctg123 . exon            1050  1500  .  +  .  ID=exon00002;Parent=mRNA00001,mRNA00002
 9  ctg123 . exon            3000  3902  .  +  .  ID=exon00003;Parent=mRNA00001,mRNA00003
10  ctg123 . exon            5000  5500  .  +  .  ID=exon00004;Parent=mRNA00001,mRNA00002,mRNA00003
11  ctg123 . exon            7000  9000  .  +  .  ID=exon00005;Parent=mRNA00001,mRNA00002,mRNA00003
12  ctg123 . CDS             1201  1500  .  +  0  ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
13  ctg123 . CDS             3000  3902  .  +  0  ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
14  ctg123 . CDS             5000  5500  .  +  0  ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
15  ctg123 . CDS             7000  7600  .  +  0  ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
16  ctg123 . CDS             1201  1500  .  +  0  ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
17  ctg123 . CDS             5000  5500  .  +  0  ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
18  ctg123 . CDS             7000  7600  .  +  0  ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
19  ctg123 . CDS             3301  3902  .  +  0  ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
20  ctg123 . CDS             5000  5500  .  +  1  ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
21  ctg123 . CDS             7000  7600  .  +  1  ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
22  ctg123 . CDS             3391  3902  .  +  0  ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
23  ctg123 . CDS             5000  5500  .  +  1  ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
24  ctg123 . CDS             7000  7600  .  +  1  ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
```

Figure: Variant Call Format (VCF) sample file

Universidad Francisco de Vitoria UFV Madrid

Linux Command-Line Interface

Universidad Francisco de Vitoria UFV Madrid

Why we need an Operating System (OS)?

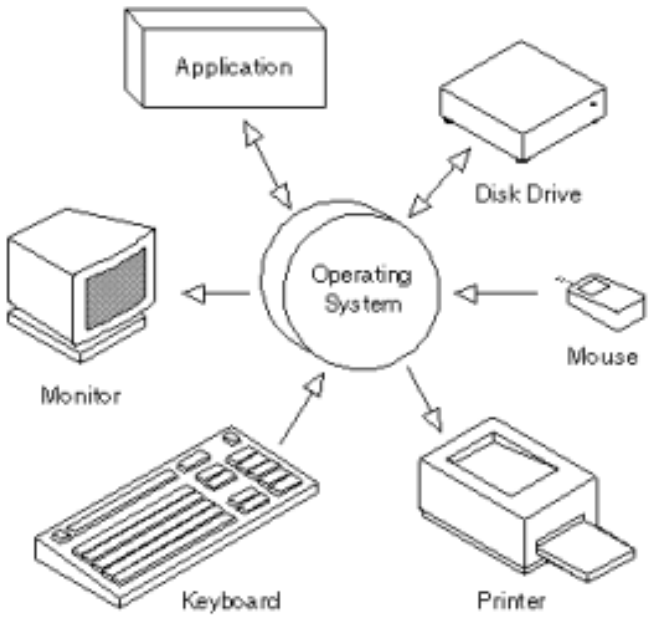


Figure: An operating system (OS) is system software that manages computer hardware and software resources and provides common services for computer programs

Universidad Francisco de Vitoria UFV Madrid

Why we need an Operating System (OS)?

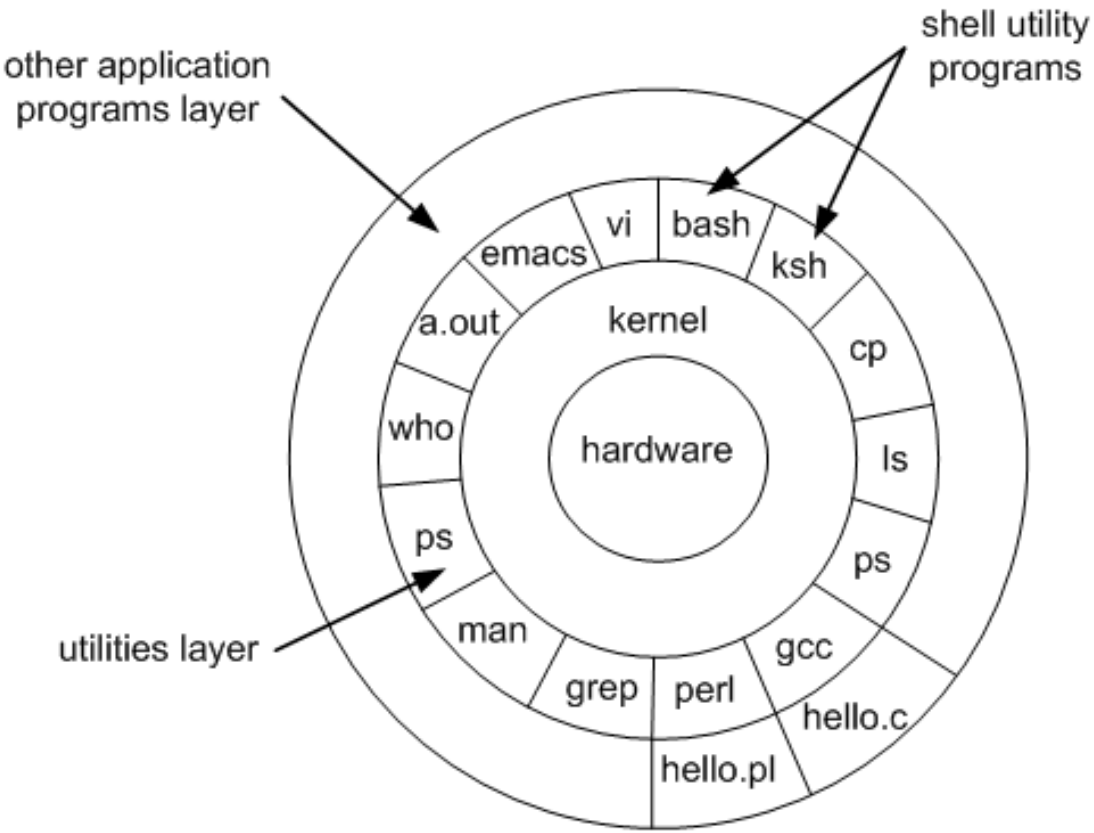


Figure: Linux Layers

Universidad Francisco de Vitoria UFV Madrid

Linux Distributions

Figure: Ubuntu will be our OS to manage the Hands-on session

Universidad Francisco de Vitoria UFV Madrid

Why use Linux for sequencing data?

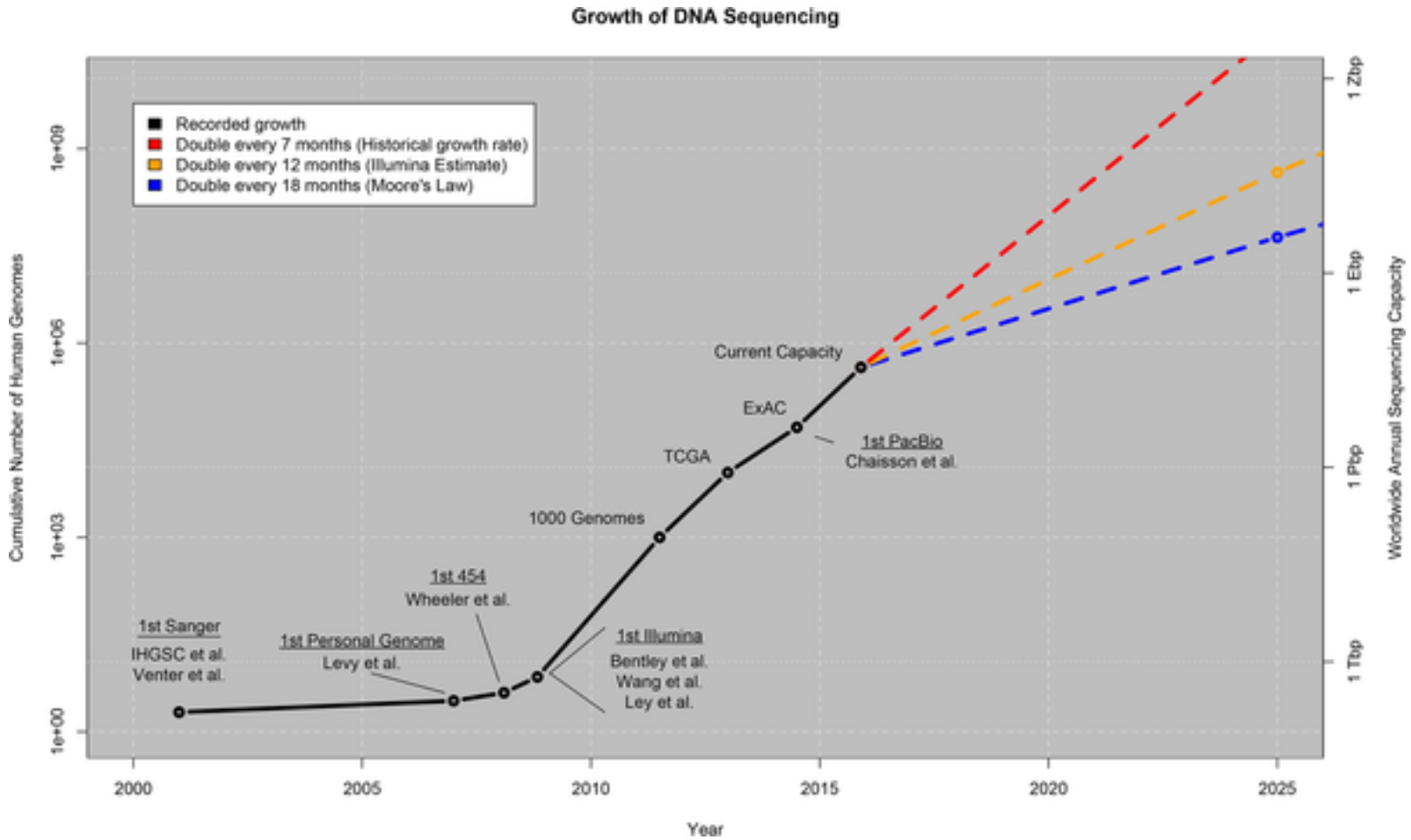


Figure: Growth of DNA Sequencing (Stephens et al. PLoS Biol. 2015)

Universidad Francisco de Vitoria UFV Madrid

Why use Linux for sequencing data?

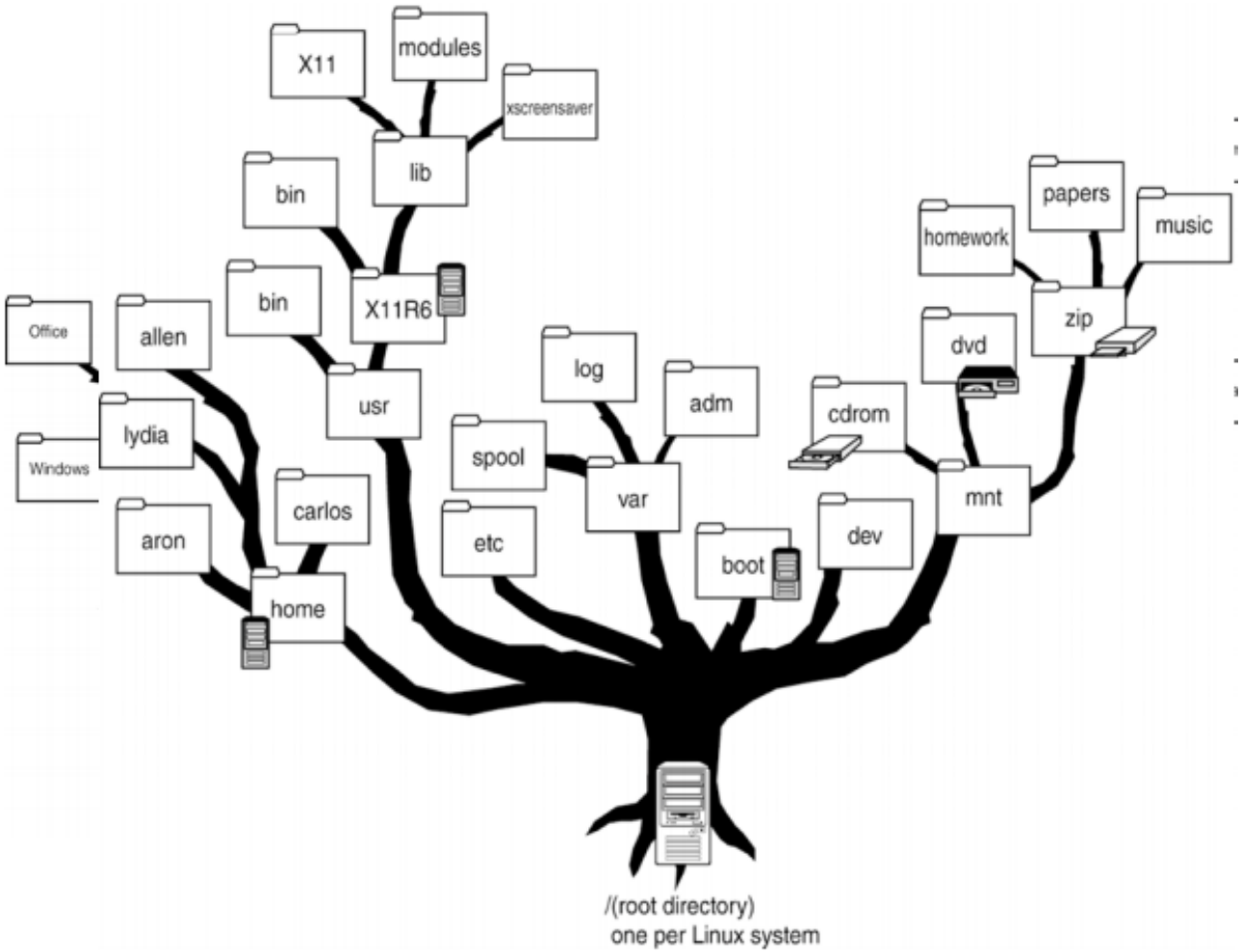
- Thousand of tools that each do simple tasks.
- Preferred development platform for **open-source** software.
- Free.
- Built for **speed**, not for ...

Some alternatives exist:

- Java / C++ programs. Run on any major operating system.
- Mac OS X is Linux based OS with a very nice GUI.
- Commercial software exist.

Universidad Francisco de Vitoria UFV Madrid

Linux Directory Structure



Universidad Francisco de Vitoria UFV Madrid

Sample Linux Paths

Path	Explanation
`/`	Refers to the root directory
`/home`	Refers to the directory `home`, which is contained in the root directory
`/usr/X11R6/lib`	Refers to the directory `lib` in the directory `X11R6`, in the directory `usr`, which is itself in the root directory
`/usr/share/xmms/Skins`	Refers to the directory `Skins` in the directory `xmms`, in the directory `share`, in the directory `usr`, which is itself in the root directory

Universidad Francisco de Vitoria UFV Madrid

Linux Commands

Basic Commands

1. **pwd** - When you first open the terminal, you are in the home directory of your user.
2. **ls** - Use this command to know what files are in the directory you are in. You can see all the hidden files by using the command **ls -a**.
3. **cd** - Use the "cd" command to go to a directory.
4. **mkdir & rmdir** - Use the mkdir command when you need to create a folder or a directory. Use rmdir to delete a directory.
5. **rm** - Use the rm command to delete files and directories. Use **rm -r** to delete just the directory. It deletes both the folder and the files it contains when using only the rm command.
6. **touch** - The touch command is used to create a file.
7. **man & –help** - To know more about a command and how to use it, use the man command.
8. **cp** - Use the cp command to copy files through the command line.
9. **mv** - Use the mv command to move files through the command line.
10. **locate** - The locate command is used to locate a file in a Linux system, just like the search command in Windows.

Linux Commands

Intermediate Commands

1. **echo** - If you want to create a new text file or add to an already made text file, you just need to type in **echo hello, my name is alok » new.txt**.
2. **cat** - Use the cat command to display the contents of a file. It is usually used to easily view programs.
3. **nano, vi, jed** - nano and vi are already installed text editors in the Linux command line.
4. **sudo** - If you want any command to be done with administrative or root privileges, you can use the sudo command.
5. **df** - Use the df command to see the available disk space in each of the partitions in your system.
6. **du** - Use du to know the disk usage of a file in your system.
7. **tar** - Use tar to work with tarballs (or files compressed in a tarball archive) in the Linux command line.
8. **zip, unzip** - Use zip to compress files into a zip archive, and unzip to extract files from a zip archive.
9. **uname** - Use uname to show the information about the system your Linux distro is running.

Linux Commands

Tips and Tricks for Using Linux Command Line

- You can use the **clear** command to clear the terminal if it gets filled up with too many commands.
- **TAB** can be used to fill up in terminal. For example, You just need to type **cd Doc** and then TAB and the terminal fills the rest up and makes it **cd Documents**.
- **Ctrl+C** can be used to stop any command in terminal safely. If it doesn't stop with that, then **Ctrl+Z** can be used to force stop it.
- You can exit from the terminal by using the **exit** command.
- You can power off or reboot the computer by using the command **reboot**.
- Use some Unix Command Line References

Universidad Francisco de Vitoria UFV Madrid

Thank you for your attention !

And now...

Universidad Francisco de Vitoria UFV Madrid

Hands-on

Click on this link to start the Hands-on:

Transcriptome Assembly: Case study of bacteria *Listeria monocytogenes*

Universidad Francisco de Vitoria UFV Madrid