PRZYRÓWNANIE DO GENOMU REFERENCYJNEGO

Analiza danych NGS
Wykład 5

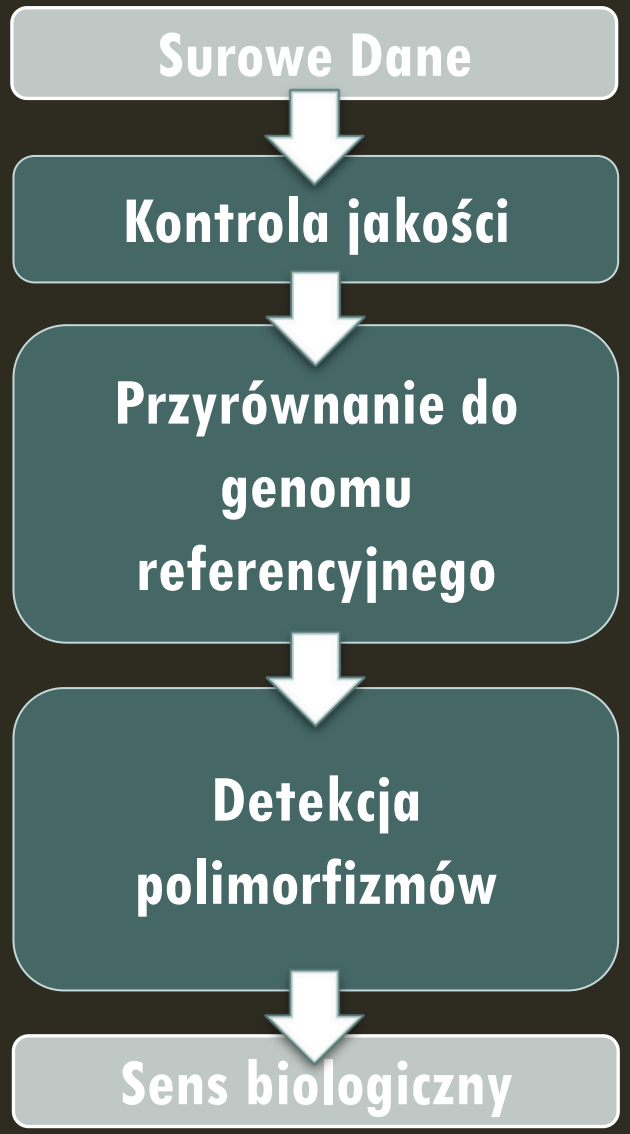
READS MAPPING

After the data is cleaned up, the next step is to align, the reads to a reference genome if it is available or conduct *de novo* assembly.

Most NGS applications require reads mapping to a reference genome prior to conducting further analysis.

READS MAPPING

Detekcja polimorfizmów genetycznych	✓
RNA-Seq: profilowanie transkryptomu	✓
Chip-Seq: Interakcje na linii białko-DNA	✓
Methyl-Seq: Epigenomika i metylacja DNA	✓
Metagenomika	✓
Poznawanie nowych genomów (*de novo* genome assembly)	X



READS MAPPING

After the data is cleaned up, the next step is to align, the reads to a reference genome if it is available or conduct *de novo* assembly.

Most NGS applications require reads mapping to a reference genome prior to conducting further analysis.

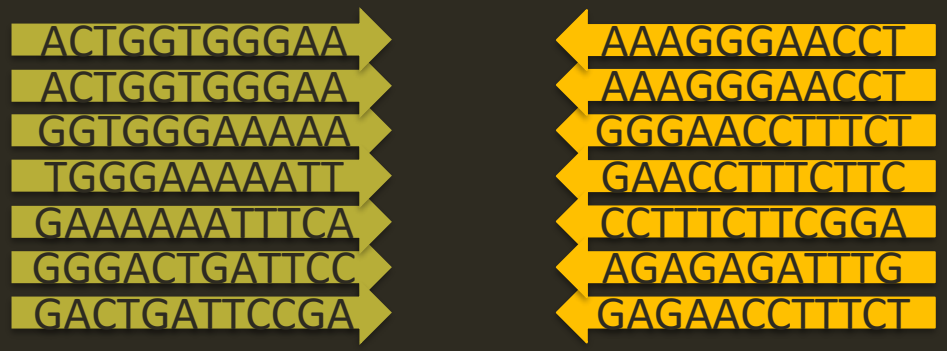
The purpose of this mapping process is to locate origins of the reads in the genome.

Simultaneous mapping of millions of NGS reads, sometimes very short, to a genome is not trivial. A challenge comes from the fact that any particular genome from which NGS reads are derived deviates from the reference genome at many sites because of polymorphisms and mutation. Sequencing errors are often indistinguishable from true sequence deviations.

PIPELINE

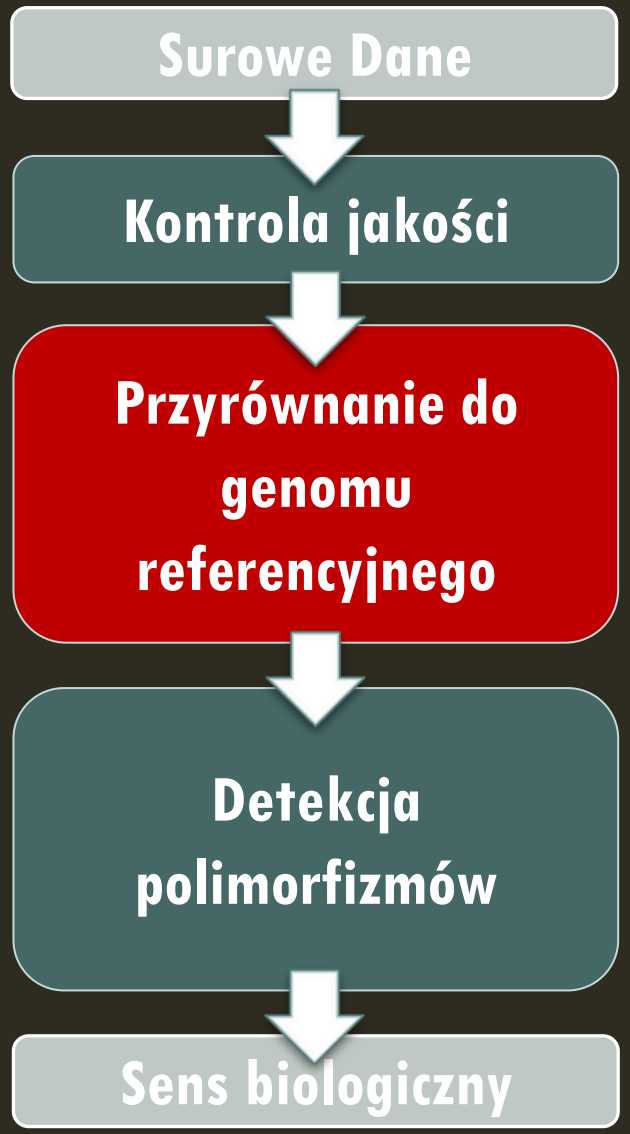

Surowe Dane
Kontrola jakości
Przyrównanie do genomu referencyjnego
Detekcja polimorfizmów
Sens biologiczny

PRZYRÓWNANIE DO GENOMU REFERENCYJNEGO

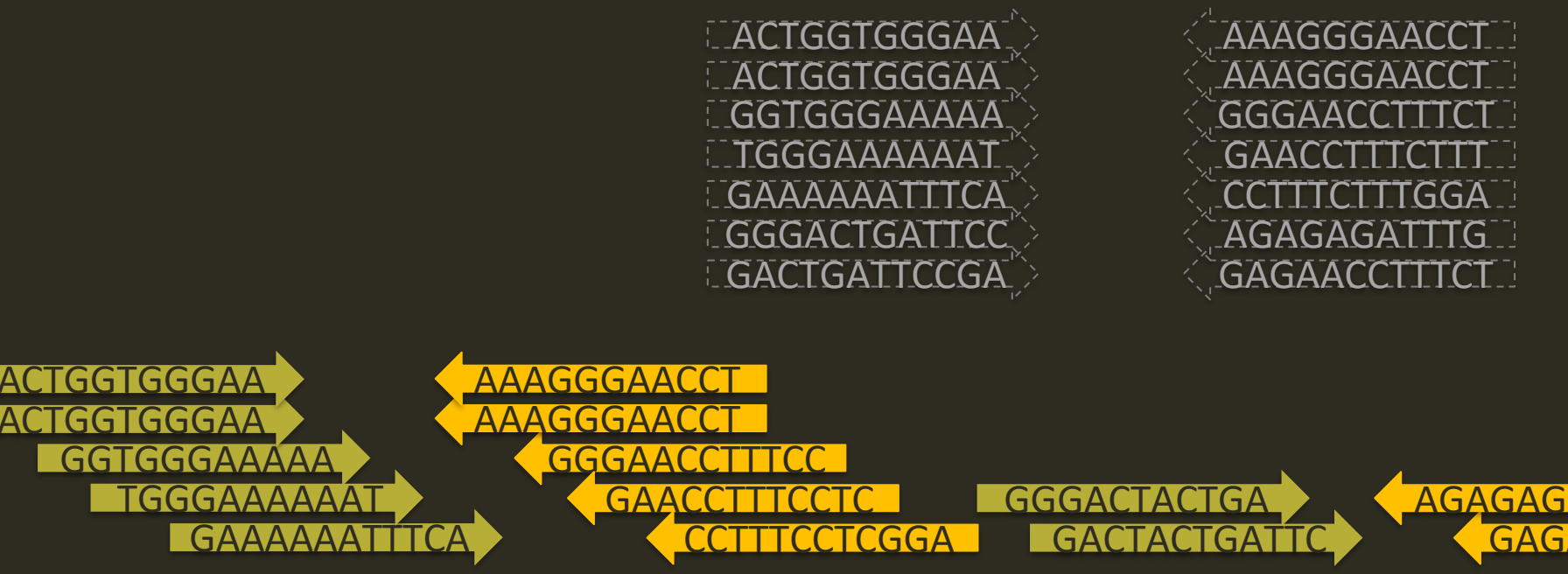


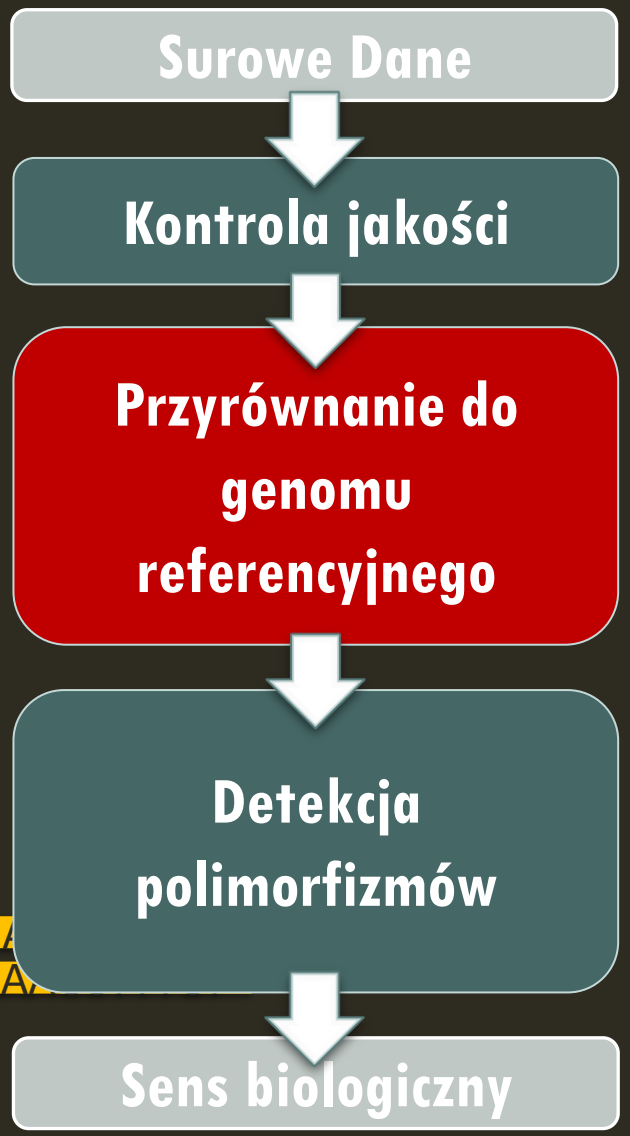
ACTGGTGGGGAAAAATTTCAAAAGGGAACCTTTCTTTGGAGCGGGACTGATTCCGAGAGAGA…

Genom referencyjny



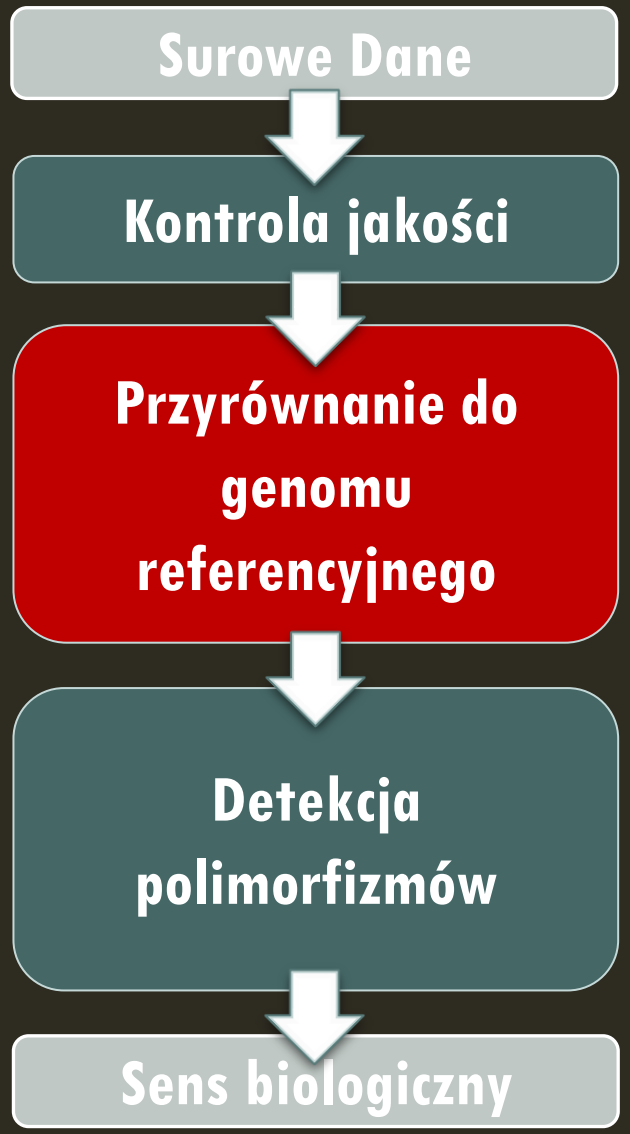
PRZYRÓWNANIE DO GENOMU REFERENCYJNEGO



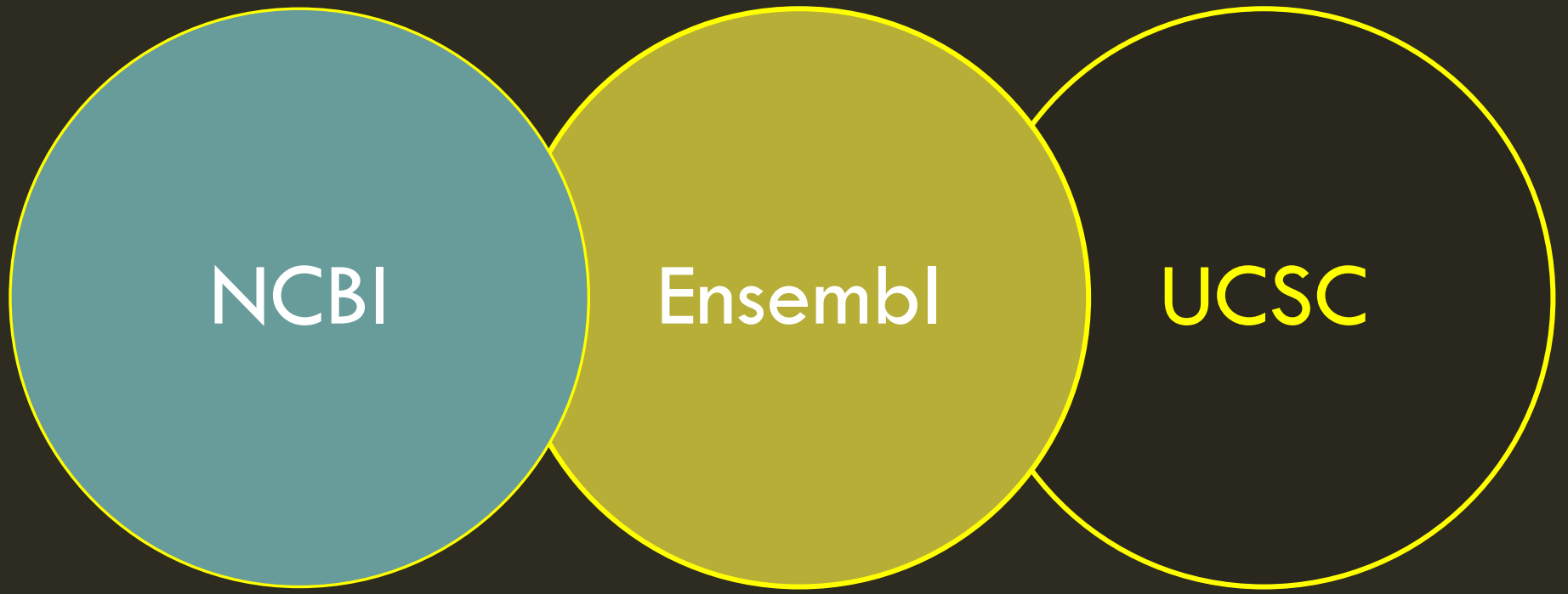


GENOM REFRENCYJNY

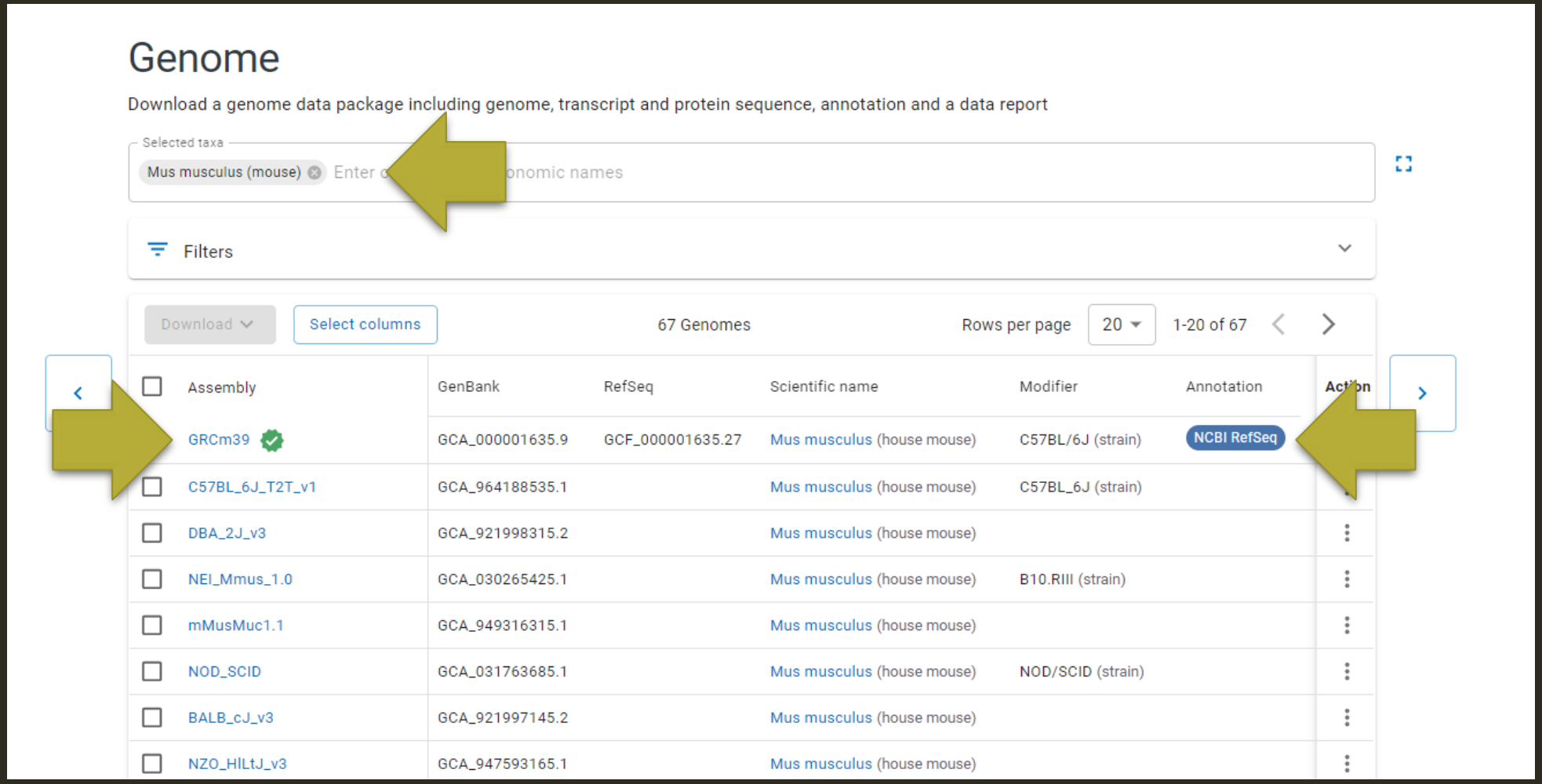
- Czym jest?
- Czy jest dostępny dla każdego gatunku?
- Skąd go pobrać?



POPULARNE PRZEGLĄDARKI GENOMOWE



GENOME


Genome
Download a genome data package including genome, transcript and protein sequence, annotation and a data report
Selected taxa
Mus musculus (mouse)
Filters
Download
Select columns
67 Genomes
Rows per page
20
1-20 of 67
Assembly
GenBank
RefSeq
Scientific name
Modifier
Annotation
GRCm39
GCA_000001635.9
GCF_000001635.27
Mus musculus (house mouse)
C57BL/6J (strain)
NCBI RefSeq
C57BL_6J_T2T_v1
GCA_964188535.1
Mus musculus (house mouse)
C57BL_6J (strain)
DBA_2J_v3
GCA_921998315.2
Mus musculus (house mouse)
NEI_Mmus_1.0
GCA_030265425.1
Mus musculus (house mouse)
B10.RIII (strain)
mMusMuc1.1
GCA_949316315.1
Mus musculus (house mouse)
NOD_SCID
GCA_031763685.1
Mus musculus (house mouse)
NOD/SCID (strain)
BALB_cJ_v3
GCA_921997145.2
Mus musculus (house mouse)
NZO_HlLtJ_v3
GCA_947593165.1
Mus musculus (house mouse)

GENOME

NCBI Datasets | Taxonomy | Genome | Gene | Command-line tools | Documentation

Genome assembly GRCm39 reference

Download | datasets | URL | FTP

		Actions
NCBI RefSeq assembly	GCF_000001635.27	⋮
Submitted GenBank assembly	GCA_000001635.9	⋮
Taxon	Mus musculus (house mouse)	
Strain	C57BL/6J	
Synonym	mm39	
Assembly type	haploid	
Submitter	Genome Refere e Consortium	
Date	Jun 24, 2020	

View annotated genes

See in Genome Data Viewer

Addi

Browse a

BioP

PRJNA2
Genome
musculu
Genome

Publi

Showing 5

PLoS Biol
Moderni
DM Churc

PLoS Biol
Lineage-specific biology revealed by a finished genome assembly of the mouse
DM Church, et al.

Assembly statistics

	RefSeq	GenBank
Genome size	2.7 Gb	2.7 Gb
Total ungapped length	2.7 Gb	2.7 Gb
Gaps between scaffolds	143	143
Number of chromosomes	21	21
Number of organelles	1	1
Number of scaffolds	101	101
Scaffold N50	106.1 Mb	106.1 Mb
Scaffold L50	11	11
Number of contigs	305	305
Contig N50	59.5 Mb	59.5 Mb
Contig L50	15	15
GC percent	42	42
Assembly level	Chromosome	Chromosome
View sequences	view RefSeq sequences	view GenBank sequences

GENOME

Chromosomes

Download

Chromosome	GenBank	RefSeq	Size (bp)	GC		
1	CM000994.3	NC_000067.7	195 154 279			
2	CM000995.3	NC_000068.8	181 755 017			
3	CM000996.3	NC_000069.7	159 745 316			
4	CM000997.3	NC_000070.7	156 860 686			
5	CM000998.3	NC_000071.7	151 758 149	42,5	5	

Annotation details

See full annotation report

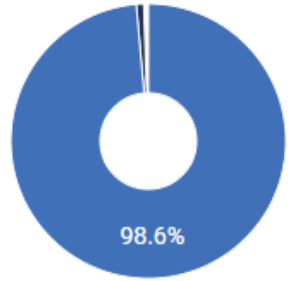
	RefSeq
Provider	NCBI RefSeq
Name	GCF_000001635.27-RS_2024_02
Date	Feb 1, 2024
Genes	50 763
Protein-coding	22 198
Software version	10.2

View RefSeq annotation

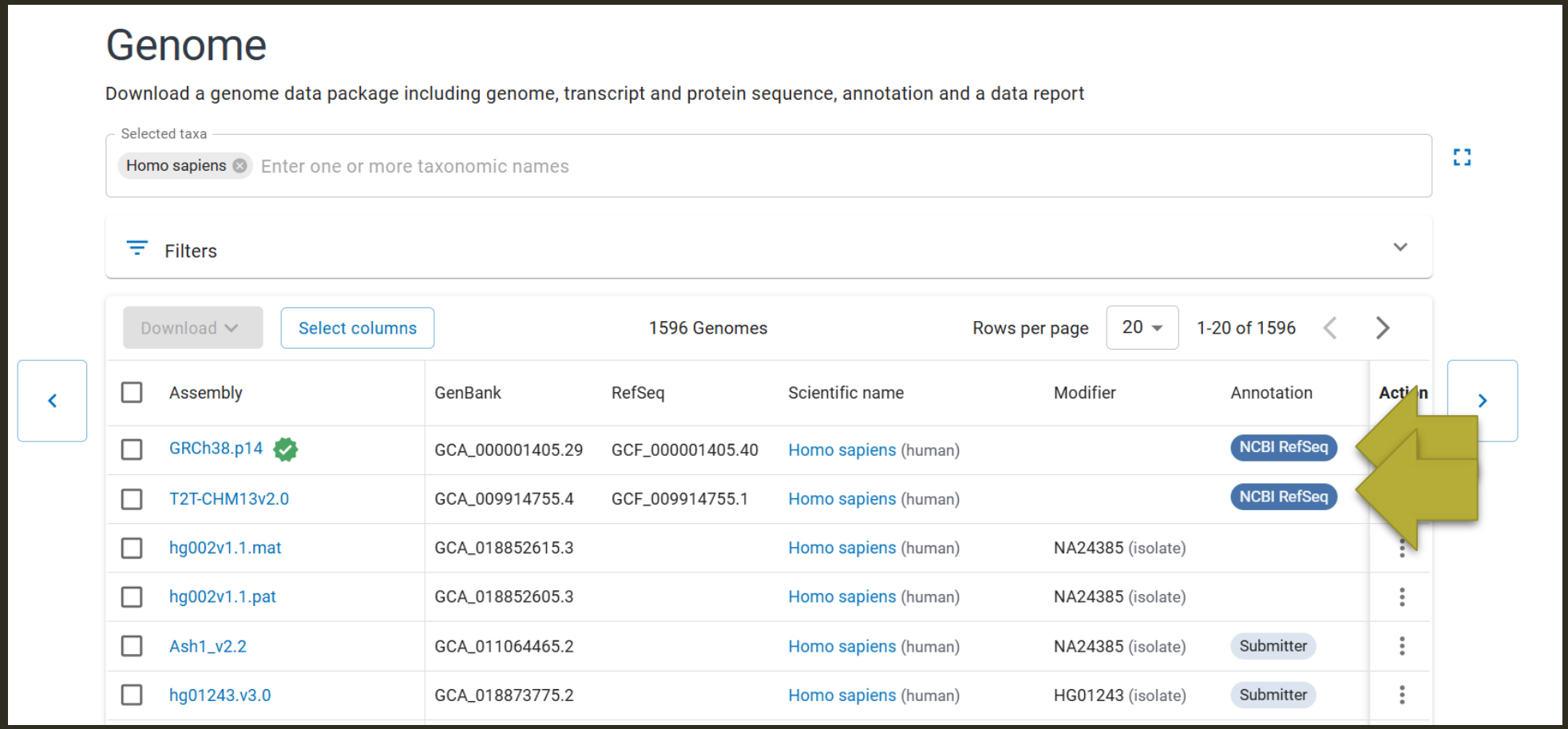
Quality analysis

BUSCO analysis (4.1.4)

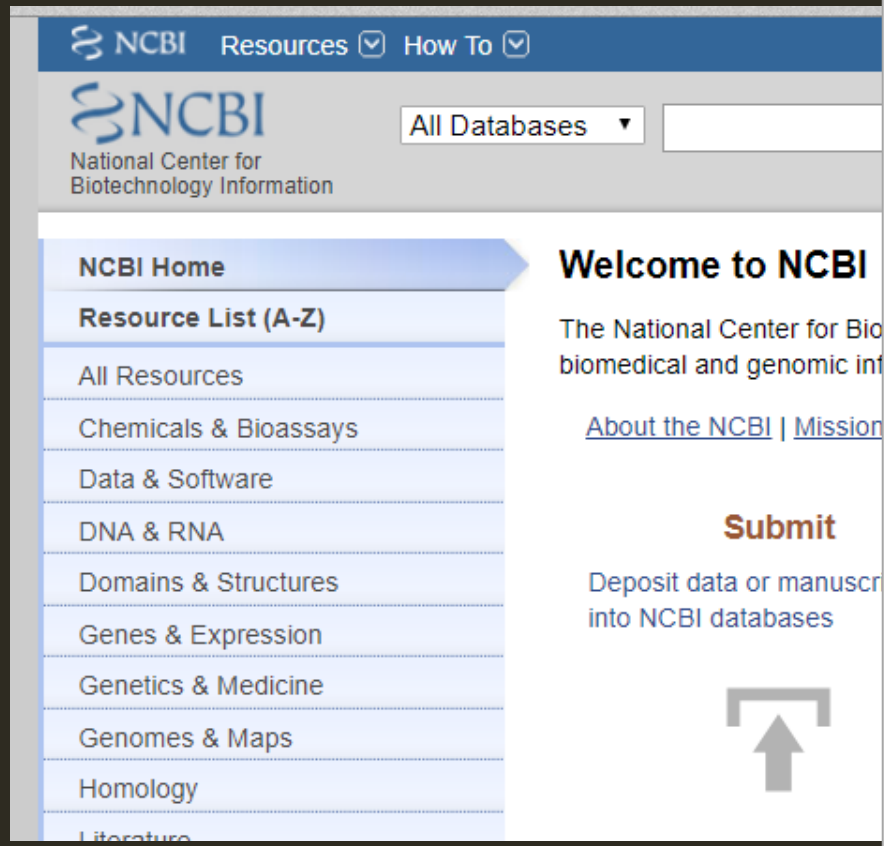
- Single_copy 98.6%
- Duplicated 0.9%
- Fragmented 0.2%
- Missing 0.4%



GENOME


Genome
Download a genome data package including genome, transcript and protein sequence, annotation and a data report
Selected taxa
Homo sapiens
Enter one or more taxonomic names
Filters
Download
Select columns
1596 Genomes
Rows per page
20
1-20 of 1596
Assembly
GenBank
RefSeq
Scientific name
Modifier
Annotation
GRCh38.p14
GCA_000001405.29
GCF_000001405.40
Homo sapiens (human)
NCBI RefSeq
T2T-CHM13v2.0
GCA_009914755.4
GCF_009914755.1
Homo sapiens (human)
NCBI RefSeq
hg002v1.1.mat
GCA_018852615.3
Homo sapiens (human)
NA24385 (isolate)
hg002v1.1.pat
GCA_018852605.3
Homo sapiens (human)
NA24385 (isolate)
Ash1_v2.2
GCA_011064465.2
Homo sapiens (human)
NA24385 (isolate)
Submitter
hg01243.v3.0
GCA_018873775.2
Homo sapiens (human)
HG01243 (isolate)
Submitter

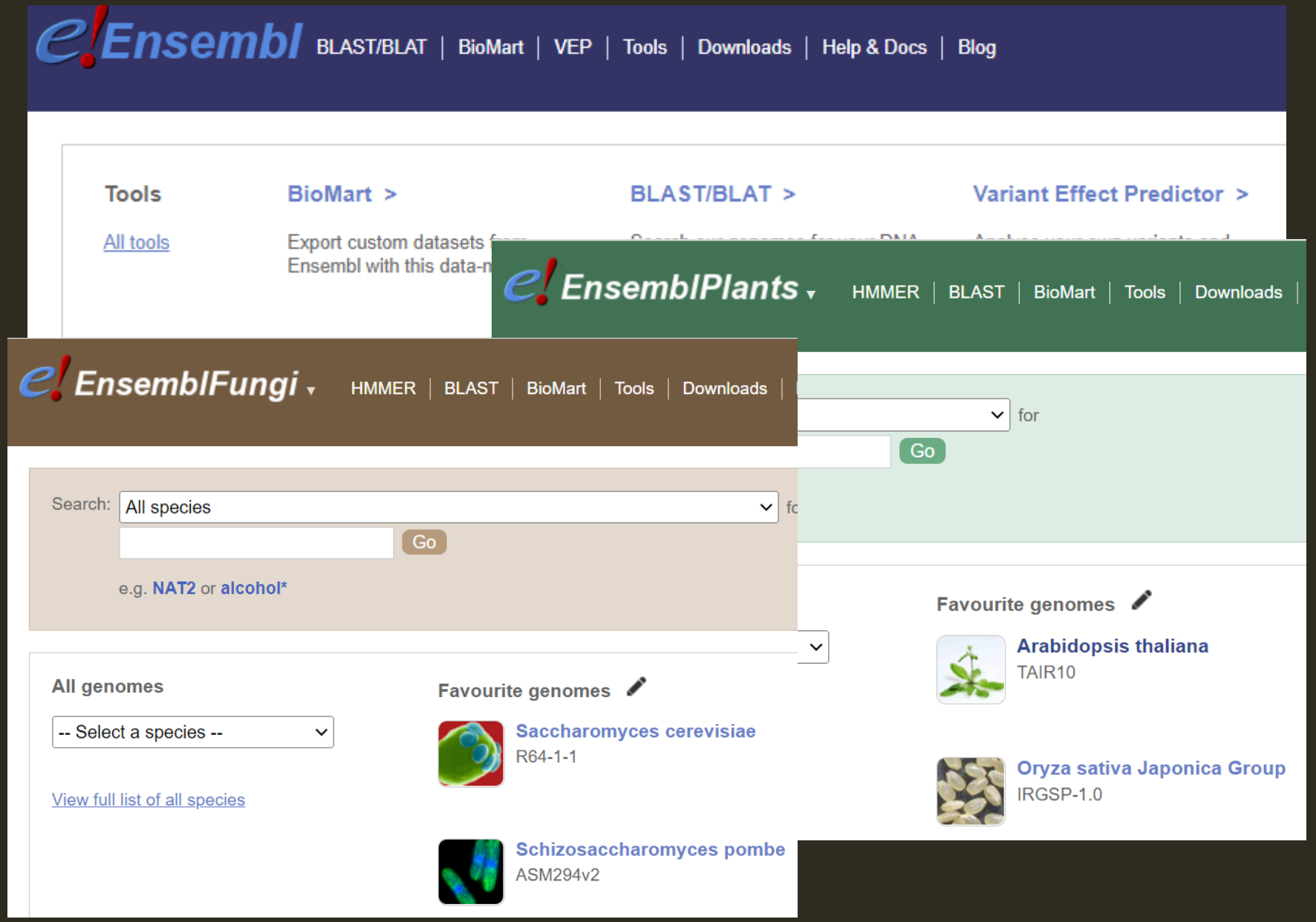
NCBI DOWNLOAD
FTP (FILE TRANSFER PROTOCOL)



Name	Last modified	Size
Parent Directory		-
GCF_002263795.2_ARS-UCD1.3_assembly_structure/	2023-06-24 18:00	-
GCF_002263795.2_ARS-UCD1.3_assembly_report.txt	2022-05-20 23:22	235K
GCF_002263795.2_ARS-UCD1.3_assembly_stats.txt	2022-05-20 23:22	45K
GCF_002263795.2_ARS-UCD1.3_cds_from_genomic.fna.gz	2022-05-20 23:22	18M
GCF_002263795.2_ARS-UCD1.3_feature_count.txt.gz	2022-05-20 23:22	708
GCF_002263795.2_ARS-UCD1.3_feature_table.txt.gz	2022-05-20 23:22	4.0M
GCF_002263795.2_ARS-UCD1.3_genomic.fna.gz	2022-05-20 23:22	801M
GCF_002263795.2_ARS-UCD1.3_genomic.gbff.gz	2023-06-24 18:00	1.1G
GCF_002263795.2_ARS-UCD1.3_genomic.gff.gz	2023-06-24 18:00	23M
GCF_002263795.2_ARS-UCD1.3_genomic.gtf.gz	2023-06-24 18:00	22M
GCF_002263795.2_ARS-UCD1.3_genomic_gaps.txt.gz	2022-05-20 23:22	3.9K
GCF_002263795.2_ARS-UCD1.3_protein.faa.gz	2023-06-24 18:00	15M
GCF_002263795.2_ARS-UCD1.3_protein.gpff.gz	2023-06-24 18:00	39M
GCF_002263795.2_ARS-UCD1.3_rna.fna.gz	2023-06-24 18:00	51M
GCF_002263795.2_ARS-UCD1.3_rna.gbff.gz	2023-06-24 18:00	134M
GCF_002263795.2_ARS-UCD1.3_rna_from_genomic.fna.gz	2023-06-24 18:00	40M
GCF_002263795.2_ARS-UCD1.3_translated_cds.faa.gz	2022-05-20 23:22	12M
README.txt	2023-09-01 13:55	48K
annotation_hashes.txt	2023-06-24 18:00	410
assembly_status.txt	2023-09-12 10:16	14
md5checksums.txt	2023-06-24 18:00	23K

ENSEMBL

- Ensembl
- EnsemblGenomes
- EnsemblFungi
- EnsemblMetazoa
- EnsemblProtists
- EnsemblBacteria
- EnsemblPlants



ENSEMBL

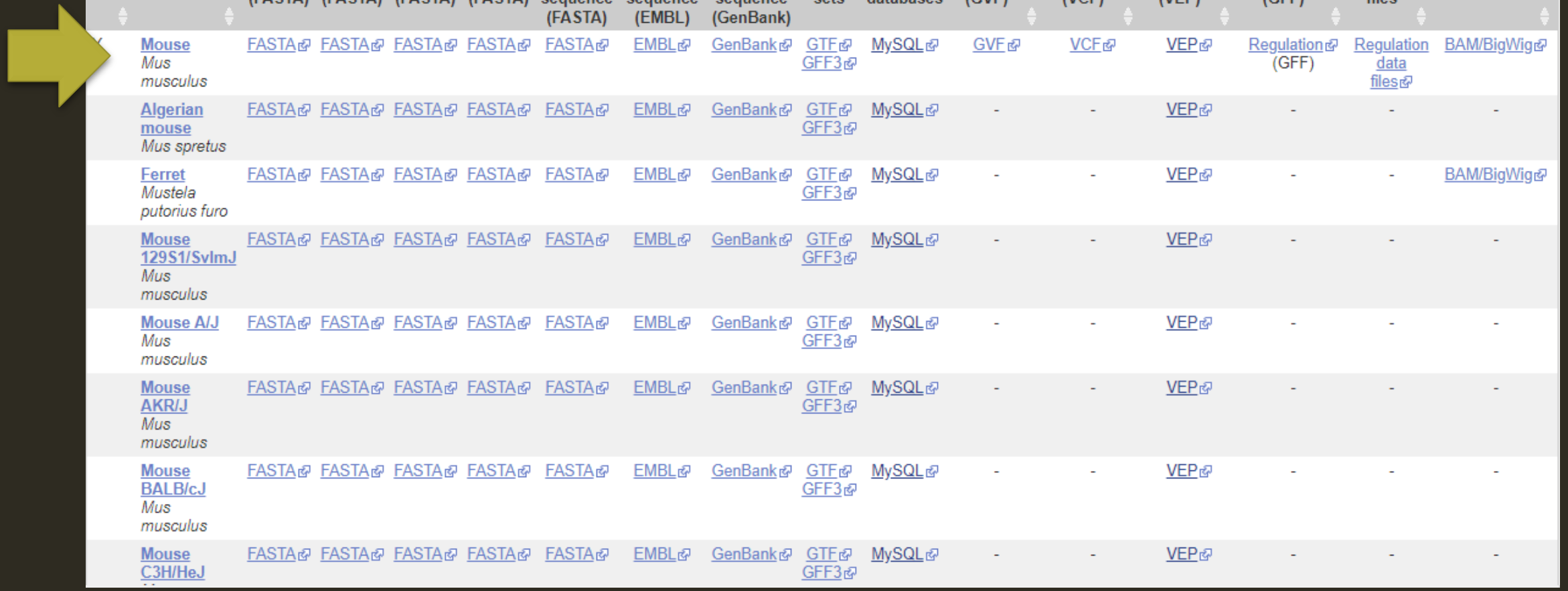
Downloads → FTP Download

Show 10 entries | Show/hide columns | Filter

★	Species	DNA (FASTA)	cDNA (FASTA)	CDS (FASTA)	ncRNA (FASTA)	Protein sequence (FASTA)	Annotated sequence (EMBL)	Annotated sequence (GenBank)	Gene sets	Whole databases	Variation (GVF)	Variation (VCF)	Variation (VEP)	Regulation (GFF)	Data files	BAM/BigWig
Y	Human *Homo sapiens*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	MySQL	GVF	VCF	VEP	Regulation (GFF)	Regulation data files	BAM/BigWig
Y	Mouse *Mus musculus*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	MySQL	GVF	VCF	VEP	Regulation (GFF)	Regulation data files	BAM/BigWig
Y	Zebrafish *Danio rerio*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	MySQL	GVF	VCF	VEP	-	-	BAM/BigWig
	Algerian mouse *Mus spretus*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	MySQL	-	-	VEP	-	-	-
	Alpaca *Vicugna pacos*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	MySQL	-	-	VEP	-	-	-
	Amazon molly *Poecilia formosa*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	MySQL	-	-	VEP	-	-	BAM/BigWig
	Angola colobus *Colobus angolensis palliatus*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	MySQL	-	-	VEP	-	-	-
	Anole lizard *Anolis carolinensis*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	MySQL	-	-	VEP	-	-	BAM/BigWig

Downloads → Download data via FTP

ENSEMBL



ENSEMBL

Masked and unmasked genome sequences associated with the assembly (contigs, chromosomes etc.)
Coordinate-system string: coord_system:version:name:start:end:strand

Show 10 entries | Show/hide columns | mus

★	Species	DNA (FASTA)	cDNA (FASTA)	CDS (FASTA)	ncRNA (FASTA)	Protein sequence (FASTA)	Annotated sequence (EMBL)	Annotated sequence (GenBank)	Gene sets	Whole databases	Variation (GVF)	Variation (VCF)	Variation (VEP)	Regulation (GFF)	Data files	BAM/BigWig
Y	Mouse *Mus musculus*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	MySQL	GVF	VCF	VEP	Regulation (GFF)	Regulation data files	BAM/BigWig
	Algerian mouse *Mus spretus*	-	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	MySQL	-	-	VEP	-	-	-
	Ferret *Mustela putorius furo*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	MySQL	-	-	VEP	-	-	BAM/BigWig
	Mouse 129S1/SvImJ *Mus musculus*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	MySQL	-	-	VEP	-	-	-
	Mouse A/J *Mus musculus*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	MySQL	-	-	VEP	-	-	-
	Mouse AKR/J *Mus musculus*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	MySQL	-	-	VEP	-	-	-
	Mouse BALB/cJ *Mus musculus*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	MySQL	-	-	VEP	-	-	-
	Mouse C3H/HeJ	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	MySQL	-	-	VEP	-	-	-

Index of /pub/release-105/fasta/mus_musculus/dna/

../		
CHECKSUMS	18-Jun-2021 11:36	4339
Mus_musculus.GRCm39.dna.chromosome.1.fa.gz	Jun-2021 09:46	58349084
Mus_musculus.GRCm39.dna.chromosome.10.fa.gz	Jun-2021 10:07	38648477
Mus_musculus.GRCm39.dna.chromosome.11.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.12.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.13.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.14.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.15.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.16.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.17.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.18.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.19.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.2.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.3.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.4.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.5.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.6.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.7.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.8.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.9.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.MT.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.X.fa.gz		
Mus_musculus.GRCm39.dna.chromosome.Y.fa.gz		
Mus_musculus.GRCm39.dna.nonchromosomal.fa.gz		
Mus_musculus.GRCm39.dna.primary_assembly.fa.gz		
Mus_musculus.GRCm39.dna.toplevel.fa.gz		
Mus_musculus.GRCm39.dna_rm.chromosome.1.fa.gz		
Mus_musculus.GRCm39.dna_rm.chromosome.10.fa.gz		
Mus_musculus.GRCm39.dna_rm.toplevel.fa.gz	18-Jun-2021 10:35	503501360
Mus_musculus.GRCm39.dna_sm.chromosome.1.fa.gz	18-Jun-2021 09:47	62448871
Mus_musculus.GRCm39.dna_sm.chromosome.10.fa.gz	18-Jun-2021 10:07	41360392
Mus_musculus.GRCm39.dna_sm.chromosome.11.fa.gz	18-Jun-2021 10:16	38640505
Mus_musculus.GRCm39.dna_sm.chromosome.12.fa.gz	18-Jun-2021 10:21	38037668
Mus_musculus.GRCm39.dna_sm.chromosome.13.fa.gz	18-Jun-2021 10:19	38193482
Mus_musculus.GRCm39.dna_sm.chromosome.14.fa.gz	18-Jun-2021 10:12	39429730
Mus_musculus.GRCm39.dna_sm.chromosome.15.fa.gz	18-Jun-2021 10:23	32832864
Mus_musculus.GRCm39.dna_sm.chromosome.16.fa.gz	18-Jun-2021 10:25	30870870
Mus_musculus.GRCm39.dna_sm.chromosome.17.fa.gz	18-Jun-2021 10:28	29815607
Mus_musculus.GRCm39.dna_sm.chromosome.18.fa.gz	18-Jun-2021 10:32	28513644
Mus_musculus.GRCm39.dna_sm.chromosome.19.fa.gz	18-Jun-2021 10:34	18928258
Mus_musculus.GRCm39.dna_sm.chromosome.2.fa.gz	18-Jun-2021 09:50	58003425
Mus_musculus.GRCm39.dna_sm.chromosome.3.fa.gz	18-Jun-2021 09:55	50897550
Mus_musculus.GRCm39.dna_sm.chromosome.4.fa.gz	18-Jun-2021 09:58	49591679
Mus_musculus.GRCm39.dna_sm.chromosome.5.fa.gz	18-Jun-2021 10:00	48134397
Mus_musculus.GRCm39.dna_sm.chromosome.6.fa.gz	18-Jun-2021 10:03	47551167
Mus_musculus.GRCm39.dna_sm.chromosome.7.fa.gz	18-Jun-2021 10:05	45871463
Mus_musculus.GRCm39.dna_sm.chromosome.8.fa.gz	18-Jun-2021 10:10	40861839
Mus_musculus.GRCm39.dna_sm.chromosome.9.fa.gz	18-Jun-2021 10:14	39450819
Mus_musculus.GRCm39.dna_sm.chromosome.MT.fa.gz	18-Jun-2021 10:34	5305
Mus_musculus.GRCm39.dna_sm.chromosome.X.fa.gz	18-Jun-2021 09:53	52942432
Mus_musculus.GRCm39.dna_sm.chromosome.Y.fa.gz	18-Jun-2021 10:30	28570341
Mus_musculus.GRCm39.dna_sm.nonchromosomal.fa.gz	18-Jun-2021 09:44	1486008
Mus_musculus.GRCm39.dna_sm.primary_assembly.fa.gz	18-Jun-2021 10:35	862438316
Mus_musculus.GRCm39.dna_sm.toplevel.fa.gz	18-Jun-2021 10:35	862438316
README	18-Jun-2021 10:34	4987

UCSC


Genomics Institute
UCSC
Genome Browser
Genomes
Genome Browser
Tools
Mirrors
Downloads
My Data
Projects
Help
About Us
Our tools
Genome Browser
interactively visualize genomic data
COVID-19 Research
use the SARS-CoV-2 genome browser and explore coronavirus datasets
BLAT
rapidly align sequences to the genome
Table Browser
download data from the Genome Browser database
Variant Annotation Integrator
get functional effect predictions for variant calls
Data Integrator
combine data sources from the Genome Browser database
Genome Browser in a Box (GBiB)
run the Genome Browser on your laptop or server
In-Silico PCR
rapidly align PCR primer pairs to the genome
LiftOver
convert genome coordinates between assemblies
Track Hubs
import and view external data tracks
REST API
returns data in JSON format
More tools...
Our story
On June 22, 2000, UCSC and the other members of the International Human Genome Project consortium completed the first working draft of the human genome assembly, forever ensuring free public access to the genome and the information it contains. A few weeks later, on July 7, 2000, the newly assembled genome was released on the web at http://genome.ucsc.edu, along with the initial prototype of a graphical viewing tool, the UCSC Genome Browser. In the ensuing years, the website has grown to include a broad collection of vertebrate and model organism assemblies and annotations, along with a large suite of tools for viewing, analyzing and downloading data. Learn more about our history on the UCSC Genome Browser Project History page and by watching this video.
What's new
Sep. 27, 2021 - JASPAR tracks for human (hg19/hg38)
Sep. 13, 2021 - Cactus 241-way comparative genomics (hg38)
Aug. 31, 2021 - GENCODE Genes VM27 for mouse (mm39)
More news...
Subscribe
The UCSC Genome Browser is developed and maintained by the Genome Bioinformatics Group, a cross-departmental team within the UCSC Genomics Institute.

UCSC Genome Browser Gateway

Genomes | Genome Browser | Tools | Mirrors | Downloads | My Data | Help | About Us

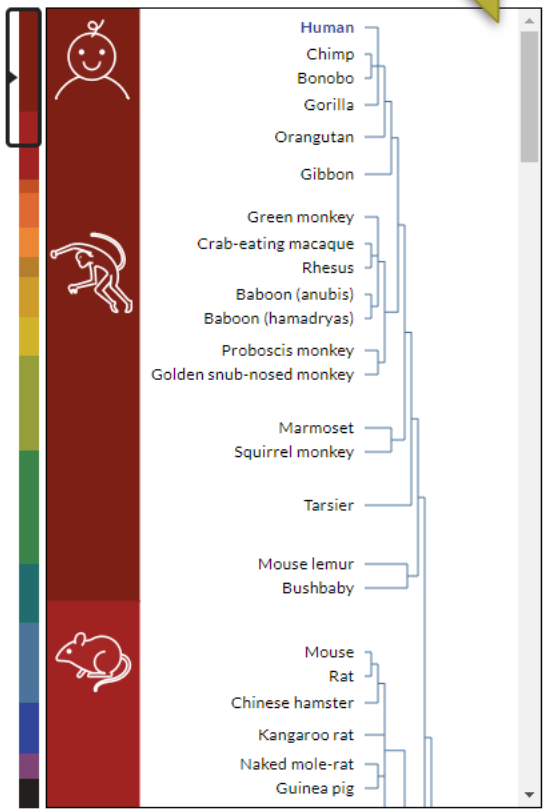
Browse/Select Species

POPULAR SPECIES

Human | Mouse | Rat | Fruitfly | Worm

Enter species or common name

REPRESENTED SPECIES



Find Position

Human Assembly: Dec. 2013 (GRCh38/hg38)

GO

Position/Search Term: Enter position, gene symbol or search terms

Current position: chr1:11,102,837-11,267,747

Human Genome Browser - hg38 assembly

view sequences

UCSC Genome Browser assembly ID: hg38
Sequencing/Assembly provider ID: Genome Reference Consortium Human GRCh38 (GCA_000001405.15)
Assembly date: Dec. 2013
Accession ID: GCA_000001405.15
NCBI Genome ID: 51 (Homo sapiens (human))
NCBI Assembly ID: 883148 (GRCh38, GCA_000001405.15)
BioProject ID: 31257

Homo sapiens
(Graphic courtesy of CBSE)

Search the assembly:

- **By position or search term:** Use the "position or search term" box to find areas of the genome associated with many different attributes, such as a specific chromosomal coordinate range; mRNA, EST, or STS marker names; or keywords from the GenBank description of an mRNA. More information, including sample queries.
- **By gene name:** Type a gene name into the "search term" box, choose your gene from the drop-down list, then press "submit" to go directly to the assembly location associated with that gene. More information.
- **By track type:** Click the "track search" button to find Genome Browser tracks that match specific selection criteria. More information.

Download sequence and annotation data:

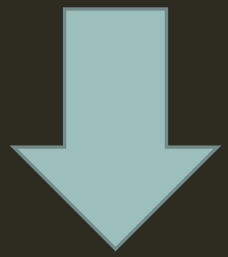
- Using rsync (recommended)
- Using FTP
- Using HTTP
- Data use conditions and restrictions
- Acknowledgments

Assembly Details

The GRCh38 assembly is the first major revision of the human genome released in more than four years. As with the previ…sembly, the Genome Reference Consortium (GRC) is now the primary source for human genome assembly data submitted to GenBank. Beginning with this release, the UCSC Genome Browser versi…he human assemblies now match those of the GRC to minimize version confusion. Hence, the GRCh38 assembly is referred to as "hg38" in the Genome Browser datasets and documentation. Fo…ssembly-related terms, see the GRC Assembly Terminology page.

FASTA

Prosty i popularny → czytelny dla wielu programów do analizy bioinformatycznej
Zapis sekwencji kwasów nukleinowych oraz białek



Identyfikator sekwencji

Sekwencja

>gi|52693750|dbj|AB175071.1
ATGACCAACTTTCGAAAAACCCATCCATTAATAAAAATTCTTAACAACTCATTCATCGAC
CATCAAACATTTCATCATGATGAAATTTCGGGTCCCTTCTAGGATTGTGCCTAGTAATC

FASTA

```
mielczarek@eagle.man.poznan.pl: ~/genomy
>1 dna:chromosome chromosome:GRCm39:1:1:195154279:1 REF
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
```

https://antropocene.it/

FASTA

mielczarek@eagle.man.poznan.pl: ~/genomy

```
>2L dna:primary_assembly primary_assembly:BDGP6.32:2L:1:23513712:1 REF
CGACAATGCACGACAGAGGAAGCAGAACAGATATTTAGATTGCCTCTCATTTTCTCTCCC
ATATTATAGGGAGAAATATGATCGCGTATGCGAGAGTAGTGCCAACATATTGTGCTCTTT
GATTTTTTGGCAACCCAAAATGGTGGCGGATGAACGAGATGATAATATATTCAAGTTGCC
GCTAATCAGAAATAAATTCATTGCAACGTTAAATACAGCACAATATATGATCGCGTATGC
GAGAGTAGTGCCAACATATTGTGCTAATGAGTGCCTCTCGTTCTCTGTCTTATATTACCG
CAAACCCAAAAAGACAATACACGACAGAGAGAGAGAGCAGCGGAGATATTTAGATTGCCT
ATTAAATATGATCGCGTATGCGAGAGTAGTGCCAACATATTGTGCTCTCTATATAATGAC
TGCCTCTCATTCTGTCTTATTTTACCGCAAACCCAAATCGACAATGCACGACAGAGGAAG
CAGAACAGATATTTAGATTGCCTCTCATTTTCTCTCCCATATTATAGGGAGAAATATGAT
CGCGTATGCGAGAGTAGTGCCAACATATTGTGCTCTTTGATTTTTTGGCAACCCAAAATG
GTGGCGGATGAACGAGATGATAATATATTCAAGTTGCCGCTAATCAGAAATAAATTCATT
GCAACGTTAAATACAGCACAATATATGATCGCGTATGCGAGAGTAGTGCCAACATATTGT
GCTAATGAGTGCCTCTCGTTCTCTGTCTTATATTACCGCAAACCCAAAAAGACAATACAC
GACAGAGAGAGAGAGCAGCGGAGATATTTAGATTGCCTATTAAATATGATCGCGTATGCG
AGAGTAGTGCCAACATATTGTGCTCTCTATATAATGACTGCCTCTCATTCTGTCTTATTT
TACCGCAAACCCAAATCGACAATGCACGACAGAGGAAGCAGAACAGATATTTAGATTGCC
TCTCATTTTCTCTCCCATATTATAGGGAGAAATATGATCGCGTATGCGAGAGTAGTGCCA
ACATATTGTGCTCTTTGATTTTTTGGCAACCCAAAATGGTGGCGGATGAACGAGATGATA
```

https://www.medianauka.pl/

FASTA

mielczarek@eagle.man.poznan.pl: ~/genomy

```
>CM009931.2 dna:primary_assembly primary_assembly:Amel_HAv3.1:CM009931.2:1:27754200:1 REF
GAGAC ATTAACTACCTTAACCTGAACCTAAACCTACCGATAACCTAACTCTAAACTATA
CCT CCCTAAACCCTACACCTAAGTCCTAAACCAATAACCTTAACCCTAACAACTA
TATA CACTAACCTATAACCTAATCCCCTAACTACTAAACTACTAACCTAACCTAAAA
CTAT CCTAACCTAAACCTTACCCTAACCATAACCTATTACTCTAACCCTACCAAGAG
CCTAAACCTAGAAACTTAACCCCTACAACCCTTAACCTTAACCTACACCTAACTACCTAA
TCCTACCTAACCTATACCTAACCTAACCTATACCTAACACCTAACCTAATCCCTTAAGCG
TAGGACTTACACCCTAACCTAACCTAACCTAACCTAACCTCTACCTAACCTTAACCTAAC
CTAACACTACACCTGGAAGACCTGTAACCTAAACCTAACCATGTAAACTCTAACGCTATA
CCTAACCTAACAATCAAGCCTATACCGTAACCTACAACAACACTGAACCTATACTCCTAT
ACTACACCTAAACCTAACCTAACCTAACCTAAGCCTAACACCTTACGCTAACTCACCCTA
ACCTAACCTAAACCTAACCCAAACCTAGACCTAACTAACCTAAACACGTAACCTATTATC
CTAACCTAACCTAAACCTCAACCTAACCTAAACCCTAACCTTAAACCTAACCTAACCTAA
TAACCTAACTGAACCGTACAACCTAACCTACGACCTTAATACGCTAACCTAAATAACTAC
TAACCTATAATAATAGCCTAAACTACCTATACCTAAACTAACCGTTAACCTAACTCTAAA
CCTAACCTACTAACCTAACCTACCTAACCTAACCACCTGAAACCTAAACCGTAACCTAAC
CTAACGCTAGACCTACCTAGACCTAAACCTAACCCACTGTGAATCGCCTCGCTAAACCTA
CCTACACAACCTAACCTAACCAACCTAAACCTCACGACCACTTAACCTAACTAACCTAAC
CAAACCTAACCTTAACCTAACCTAAAACCTAACCTAAACCACCTAACCTAACCTAACAAC
```

www.inaturalist.org

PRZYRÓWNANIE DO GENOMU REFERENCYJNEGO

KROKI:

1. Obróbka genomu referencyjnego/odczytów
   - „(…) computational strategy known as ‘indexing’ to speed up the mapping algorithms. Like the index at the end of a book, an index of a large DNA sequence allows one to rapidly find shorter sequences embedded within it.” (Trapnell and Salzberg 2009, doi:10.1038/nbt0509-455).
   - Jedna operacja na 1 genom referencyjny

2. (właściwe) Przyrównanie do genomu referencyjnego

PRZYRÓWNANIE DO GENOMU REFERENCYJNEGO

ALGORYTMY:

1. **Tablica z haszowaniem:**
   - „Hash table on the set of input reads”
   - „Hash table on the reference genome”

2. **Transformata Burrowsa-Wheelera (BWT)**

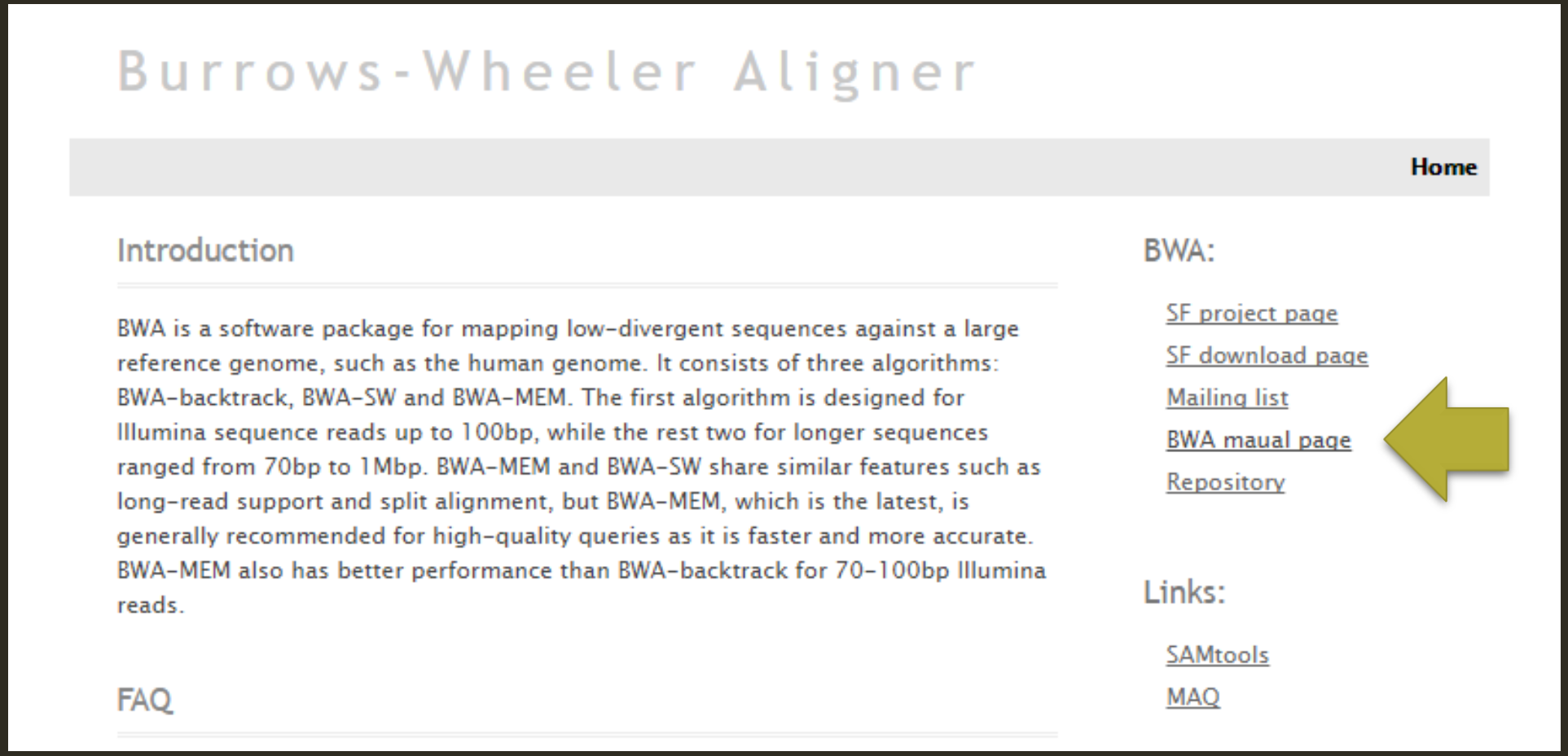
BURROWS-WHEELER TRANSFORM

To increase speed and reduce demands on computational resources, a novel approach is developed on the basis of Burrows-Wheeler transform and suffix trees (or arrays). BWT achieves better reference genome compression to enable more efficient indexing and faster searching.

The human genome indexed with BWT only takes 2-3 GB of computer memory, whereas the spaced-seed indexing approach can take over 50 GB memory. Through the use of BWT and suffix trees (or arrays), the run time needed for aligning million of reads to a large and complex genome, like the human genome, is cut from hours to minutes.

Xinkun Wang. Next Generation Sequencing Data Analysis. 2016, CRC PRESS

PROGRAM BWA


Burrows-Wheeler Aligner
Home
Introduction
BWA is a software package for mapping low-divergent sequences against a large reference genome, such as the human genome. It consists of three algorithms: BWA-backtrack, BWA-SW and BWA-MEM. The first algorithm is designed for Illumina sequence reads up to 100bp, while the rest two for longer sequences ranged from 70bp to 1Mbp. BWA-MEM and BWA-SW share similar features such as long-read support and split alignment, but BWA-MEM, which is the latest, is generally recommended for high-quality queries as it is faster and more accurate. BWA-MEM also has better performance than BWA-backtrack for 70-100bp Illumina reads.
FAQ
BWA:
SF project page
SF download page
Mailing list
BWA maual page
Repository
Links:
SAMtools
MAQ

PROGRAM BWA

DESCRIPTION

BWA is a software package for mapping low-divergent sequences against a large reference genome, such as the human genome. It consists of three algorithms: BWA-backtrack, BWA-SW and BWA-MEM. The first algorithm is designed for Illumina sequence reads up to 100bp, while the rest two for longer sequences ranged from 70bp to 1Mbp. BWA-MEM and BWA-SW share similar features such as long-read support and split alignment, but BWA-MEM, which is the latest, is generally recommended for high-quality queries as it is faster and more accurate. BWA-MEM also has better performance than BWA-backtrack for 70-100bp Illumina reads.

For all the algorithms, BWA first needs to construct the FM-index for the reference genome (the **index** command). Alignment algorithms are invoked with different sub-commands: **aln/samse/sampe** for BWA-backtrack, **bwasw** for BWA-SW and **mem** for the BWA-MEM algorithm.

PROGRAM BWA

SYNOPSIS

```
bwa index ref.fa
bwa mem ref.fa reads.fq > aln-se.sam
bwa mem ref.fa read1.fq read2.fq > aln-pe.sam
bwa aln ref.fa short_read.fq > aln_sa.sai
bwa samse ref.fa aln_sa.sai short_read.fq > aln-se.sam
bwa sampe ref.fa aln_sa1.sai aln_sa2.sai read1.fq read2.fq > aln-pe.sam
bwa bwasw ref.fa long_read.fq > aln.sam
```

https://gatk.broadinstitute.org/hc/en-us/articles/360035890671-Read-groups

READ GROUPS

GATK / Technical Documentation / Glossary

Read groups Follow

Derek Caetano-Anolles
October 17, 2023 00:11 · Updated

There is no formal definition of what a 'read group' is, however in practice this term refers to a set of reads that are generated from a single run of a sequencing instrument.

In the simple case where a single library preparation derived from a single biological sample was run on a single lane of a *flow cell*, all the reads from that lane run belong to the same read group. When multiplexing is involved, then each subset of reads originating from a separate library run on that lane will constitute a separate read group.

https://gatk.broadinstitute.org/hc/en-us/articles/360035890671-Read-groups

READ GROUPS

@LH00124:42:223YYTLT3:1:1101:1832:1032 1:N:0:TCTGTTGG+CCATTCGA
ANTAGTGATTTGAGGTCTATGGTGAAAAAGAAAATATCTTCACATGAAAACTAGACAGACTGTTTCTGAG
AAACTGCTTTGTGATGTATTCATTCATCTCACATACGTAAGCATTTCTTCTTATTGACCAGATTGGAAACTCTG
TTCTTGT
+
-#-FFFF--FF-FFF-FFFFFFFFFFF-FFFFFFFFFFFFFFFFFFFFFFFFF-FFFFFFFFFFFFFFFFFFFFFFFFF5FFFFFFF-
FFFFFFFFFFFFFFFFFFF-FFFFFF-FFFFFFFFFFFFFFFFFFFFFFFFFFFFF-FFF

@RG\tID:223YYTLT3_1\tPL:ILLUMINA\tPU:223YYTLT3.1\tSM:wgs_S9188Nr2\tLB:unknown

https://gatk.broadinstitute.org/hc/en-us/articles/360035890671-Read-groups

READ GROUPS

@LH00124:42:223YYTLT3:1:1101:1832:103
ANTAGTGATTTGAGGTCTATGGTGAAAAAGA
AAACTGCTTTGTGATGTATTCATTCATCTCACAT
TTCTTGT
+
-#-FFFF--FF-FFF-FFFFFFFFFFF-FFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFF-FFFFFF-FFFFFFFFFFFFFFF

@RG\tID:223YYTLT3_1\tPL:ILLUMINA\tPU:22

the unique instrument name
the run id
the flowcell id
flowcell lane
tile 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 (did not pass), N otherwise
0 when none of the control bits are on, otherwise it is an even number
index sequence

PROGRAM BWA MEM & READ GROUPS

SYNOPSIS

```
bwa index ref.fa

bwa mem ref.fa reads.fq > aln-se.sam

bwa mem ref.fa read1.fq read2.fq > aln-pe.sam

bwa aln ref.fa short_read.fq > aln_sa.sai

bwa samse ref.fa aln_sa.sai short_read.fq > aln-se.sam

bwa sampe ref.fa aln_sa1.sai aln_sa2.sai read1.fq read2.fq > aln-pe.sam

bwa bwasw ref.fa long_read.fq > aln.sam
```

-R *STR* Complete read group header line. '\t' can be used in *STR* and will be converted to a TAB in the output SAM. The read group ID will be attached to every read in the output. An example is '@RG\tID:foo\tSM:bar'. [null]

http://chagall.med.cornell.edu/NGScourse/SAM.pdf

SAM (SEQUENCE ALIGNMENT/MAP FORMAT)

header section

```
@HD    VN:1.0
@SQ    SN:chr20 LN:62435964
@RG    ID:L1 PU:SC_1_10 LB:SC_1 SM:NA12891
@RG    ID:L2 PU:SC_2_12 LB:SC_2 SM:NA12891
read_28833_29006_6945 99  chr20 28833 20 10M1D25M = 28993 195 \
       AGCTTAGCTAGCTACCTATATCTTGGTCTTGGCCG <<<<<<<<<<<<<<<<<<<<<<<<:<9/,&,22;;<<< \
       NM:i:1 RG:Z:L1
read_28701_28881_323b 147 chr20 28834 30 35M      = 28701 -168 \
       ACCTATATCTTGGCCTTGGCCGATGCGGCCTTGCA <<<<<;<<<<7;:<<<6;<<<<<<<<<<<7<<<< \
       MF:i:18 RG:Z:L2
```

alignment section

```
read_28833_29006_6945 99  chr20 28833 20 10M1D25M = 28993 195 \
      AGCTTAGCTAGCTACCTATATCTTGGTCTTGGCCG <<<<<<<<<<<<<<<<<<<<:<9/,&,22;;<<< \
      NM:i:1 RG:Z:L1
read_28701_28881_323b 147 chr20 28834 30 35M      = 28701 -168 \
      ACCTATATCTTGGCCTTGGCCGATGCGGCCTTGCA <<<<<;<<<<7;:<<<6;<<<<<<<<<<7<<<< \
      MF:i:18 RG:Z:L2
```

The alignment section consists of multiple TAB-delimited lines with each line describing an alignment. Each line is:

```
<QNAME> <FLAG> <RNAME> <POS> <MAPQ> <CIGAR> <MRNM> <MPOS> <ISIZE> <SEQ> <QUAL> \
  [<TAG>:<VTYPE>:<VALUE> [...]]
```

Field	Regular expression	Range	Description
QNAME	[^ \t\n\r]+		Query pair NAME if paired; or Query NAME if unpaired [2]
FLAG	[0-9]+	[0,2^{16}-1]	bitwise FLAG (Section 2.2.2)
RNAME	[^ \t\n\r@=]+		Reference sequence NAME [3]
POS	[0-9]+	[0,2^{29}-1]	1-based leftmost POSition/coordinate of the clipped sequence
MAPQ	[0-9]+	[0,2^{8}-1]	MAPping Quality (phred-scaled posterior probability that the mapping position of this read is incorrect) [4]
CIGAR	([0-9]+[MIDNSHP])+|*		extended CIGAR string
MRNM	[^ \t\n\r@]+		Mate Reference sequence NaMe; "=" if the same as <RNAME> [3]
MPOS	[0-9]+	[0,2^{29}-1]	1-based leftmost Mate POSition of the clipped sequence
ISIZE	-?[0-9]+	[-2^{29},2^{29}]	inferred Insert SIZE [5]
SEQ	[acgtnACGTN.=]+|*		query SEQuence; "=" for a match to the reference; n/N/. for ambiguity; cases are not maintained [6,7]
QUAL	[!-~]+|*	[0,93]	query QUALity; ASCII-33 gives the Phred base quality [6,7]
TAG	[A-Z][A-Z0-9]		TAG
VTYPE	[AifZH]		Value TYPE
VALUE	[^\t\n\r]+		match <VTYPE> (space allowed)

SAM/BAM

```
A00182:618:HL7TTDSX2:1:1101:1325:1000    83      13      34692686        60
151M    =       34692426        -411    GGATCAAATACGAGCTGGACTTGTGACTTACGCCATAGCT
ATGGCAACACTGGATCTTTAACCTACCGCCGTGCCAGGCTGGGGATTGAATCTGTGTCCCTGCTGCTAAAGAGACCATGG
ATACTGCTGAGCCACAGCAGGAACTCCAGGA :FFF:FF,:FFF:FFFFF:FF,FFFFFF:FFFFFFFFFFF,,FFFF:F
:FFF::FF:FFF:FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF:FFFFFFF::FFF:FFFFFF:FFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFF NM:i:0  MD:Z:151        MC:Z:151M       AS:i:151
```

Field	Regular expression	Range	Description
QNAME	[^ \t\n\r]+		Query pair NAME if paired; or Query NAME if unpaired [2]
FLAG	[0-9]+	[0,2^{16}-1]	bitwise FLAG (Section 2.2.2)
RNAME	[^ \t\n\r@=]+		Reference sequence NAME [3]
POS	[0-9]+	[0,2^{29}-1]	1-based leftmost POSition/coordinate of the clipped sequence
MAPQ	[0-9]+	[0,2^{8}-1]	MAPping Quality (phred-scaled posterior probability that the mapping position of this read is incorrect) [4]
CIGAR	([0-9]+[MIDNSHP])+|*		extended CIGAR string
MRNM	[^ \t\n\r@]+		Mate Reference sequence NaMe; "=" if the same as <RNAME> [3]
MPOS	[0-9]+	[0,2^{29}-1]	1-based leftmost Mate POSition of the clipped sequence
ISIZE	-?[0-9]+	[-2^{29},2^{29}]	inferred Insert SIZE [5]
SEQ	[acgtnACGTN.=]+|*		query SEQuence; "=" for a match to the reference; n/N/. for ambiguity; cases are not maintained [6,7]
QUAL	[!-~]+|*	[0,93]	query QUALity; ASCII-33 gives the Phred base quality [6,7]
TAG	[A-Z][A-Z0-9]		TAG
VTYPE	[AifZH]		Value TYPE
VALUE	[^\t\n\r]+		match <VTYPE> (space allowed)

SAM/BAM

Picard

build passing

A set of command line tools (in Java) for manipulating high-throughput sequencing (HTS) data and formats such as SAM/BAM/CRAM and VCF.

Latest Jar Release | Source Code ZIP File | Source Code TAR Ball | View On GitHub

Decoding SAM flags

This utility makes it easy to identify what are the properties of a read based on its SAM flag value, or conversely, to find what the SAM Flag value would be for a given combination of properties.

To decode a given SAM flag value, just enter the number in the field below. The encoded properties will be listed under Summary below, to the right.

SAM Flag: 83 Explain

Switch to mate Toggle first in pair / second in pair

Find SAM flag by property:

To find out what the SAM flag value would be for a given combination of properties, tick the boxes for those that you'd like to include. The flag value will be shown in the SAM Flag field above.

- [x] read paired
- [x] read mapped in proper pair
- [ ] read unmapped
- [ ] mate unmapped
- [x] read reverse strand
- [ ] mate reverse strand
- [x] first in pair
- [ ] second in pair
- [ ] not primary alignment
- [ ] read fails platform/vendor quality checks
- [ ] read is PCR or optical duplicate
- [ ] supplementary alignment

Summary:

read paired (0x1)
read mapped in proper pair (0x2)
read reverse strand (0x10)
first in pair (0x40)

```
A00182:618:HL7TTDSX2:1:1101:1325:1000    83      13      34692686        60
151M    =       34692426        -411    GGATCAAATACGAGCTGGACTTGTGACTTACGCCATAGCT
ATGGCAACACTGGATCTTTAACCTACCGCCGTGCCAGGCTGGGGATTGAATCTGTGTCCCTGCTGCTAAAGAGACCATGG
ATACTGCTGAGCCACAGCAGGAACTCCAGGA :FFF:FF,:FFF:FFFFF:FF,FFFFFF:FFFFFFFFFFF,,FFFF:F
:FFF::FF:FFF:FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF:FFFFFFF::FFF:FFFFFF:FFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFF NM:i:0  MD:Z:151        MC:Z:151M       AS:i:151
```

Field	Regular expression	Range	Description
QNAME	[^ \t\n\r]+		Query pair NAME if paired; or Query NAME if unpaired [2]
FLAG	[0-9]+	[0,2^{16}-1]	bitwise FLAG (Section 2.2.2)
RNAME	[^ \t\n\r@=]+		Reference sequence NAME [3]
POS	[0-9]+	[0,2^{29}-1]	1-based leftmost POSition/coordinate of the clipped sequence
MAPQ	[0-9]+	[0,2^{8}-1]	MAPping Quality (phred-scaled posterior probability that the mapping position of this read is incorrect) [4]
CIGAR	([0-9]+[MIDNSHP])+|*		extended CIGAR string
MRNM	[^ \t\n\r@]+		Mate Reference sequence NaMe; "=" if the same as <RNAME> [3]
MPOS	[0-9]+	[0,2^{29}-1]	1-based leftmost Mate POSition of the clipped sequence
ISIZE	-?[0-9]+	[-2^{29},2^{29}]	inferred Insert SIZE [5]
SEQ	[acgtnACGTN.=]+|*		query SEQuence; "=" for a match to the reference; n/N/. for ambiguity; cases are not maintained [6,7]
QUAL	[!-~]+|*	[0,93]	query QUALity; ASCII-33 gives the Phred base quality [6,7]
TAG	[A-Z][A-Z0-9]		TAG
VTYPE	[AifZH]		Value TYPE
VALUE	[^\t\n\r]+		match <VTYPE> (space allowed)

SAM/BAM

Nie wszystkie odczyty muszą przyrównać się do genomu referencyjnego jak również jeden odczyt może zostać przyrównany do kilku miejsc w genomie.

CIGAR - PRZYKŁAD

```
samtools view -F 4 S2709Nr12.merged.bam | grep A00788:260:HV32JDMXX:2:1271:13313:3286
```

Nie wszystkie odczyty muszą przyrównać się do genomu referencyjnego jak również jeden odczyt może zostać przyrównany do kilku miejsc w genomie.c

CIGAR - PRZYKŁAD

samtools view -F 4 S2709Nr12.merged.bam | grep A00788:260:HV32JDMXX:2:1271:13313:3286

A00788:260:HV32JDMXX:2:1271:13313:3286 177 1 145 60 151M = 5438 5260
CTGGACATAAAAGAGGAATTAGCATGTGCACTCTTCAGATATAAATGCCATCAGTATTTTCCTATTAAAATGAAGCTTGTTTTCATCTCAGTGGAAATCTGTGGCTAAAGTACAACAATAGTAAT
GATAATGGTGAGGCTGTTGTACTTCA FFFFFFFFF,:FFF:FFFFFFFFFFFF:FFFFFFFFFF:F:FF,FF:FFFFFF:FFFFFFFFF:FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFF:FFFFFFFFFFFFFFFFFFFF NM:i:0 MD:Z:151 AS:i:151 XS:i:20 MQ:i:60 MC:Z:33S118M ms:i:5563
RG:Z:S2709Nr12.trimm.rg2.fixmate.srt.markdup
A00788:260:HV32JDMXX:2:1271:13313:3286 2145 1 1 60
116H35M = 145 295 GCTTAATTTTTGTCATTTCTCACCCCTGCTCTTGA FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF NM:i:0 MD:Z:35
MC:Z:151M AS:i:35XS:i:19 SA:Z:1,5438 ,33S118M,60,0; RG:Z:S2709Nr12.trimm.rg2.fixmate.srt.markdup
A00788:260:HV32JDMXX:2:1271:13313:3286 113 1 5438 60 33S118M = 145 -
5260 TCAAGAGCAGGGGTGAGAAATGACAAAAATTAAGCACATAATCACGCTCAACTATATGGAAATCAGATGATAACACACATCCTTACTCATGCCTGATTTTAAGAAAATGAAGTTACT
GCTGTCTGCAGTGATGAAAATCAACCACCTTAAC FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF:FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF:F NM:i:0 MD:Z:118 AS:i:118 XS:i:20 SA:Z:1,1,+,116S35M,60,0; MQ:i:60
MC:Z:151M ms:i:5405 RG:Z:S2709Nr12.trimm.rg2.fixmate.srt.markdup

Nie wszystkie odczyty muszą przyrównać się do genomu referencyjnego jak również jeden odczyt może zostać przyrównany do kilku miejsc w genomie.c

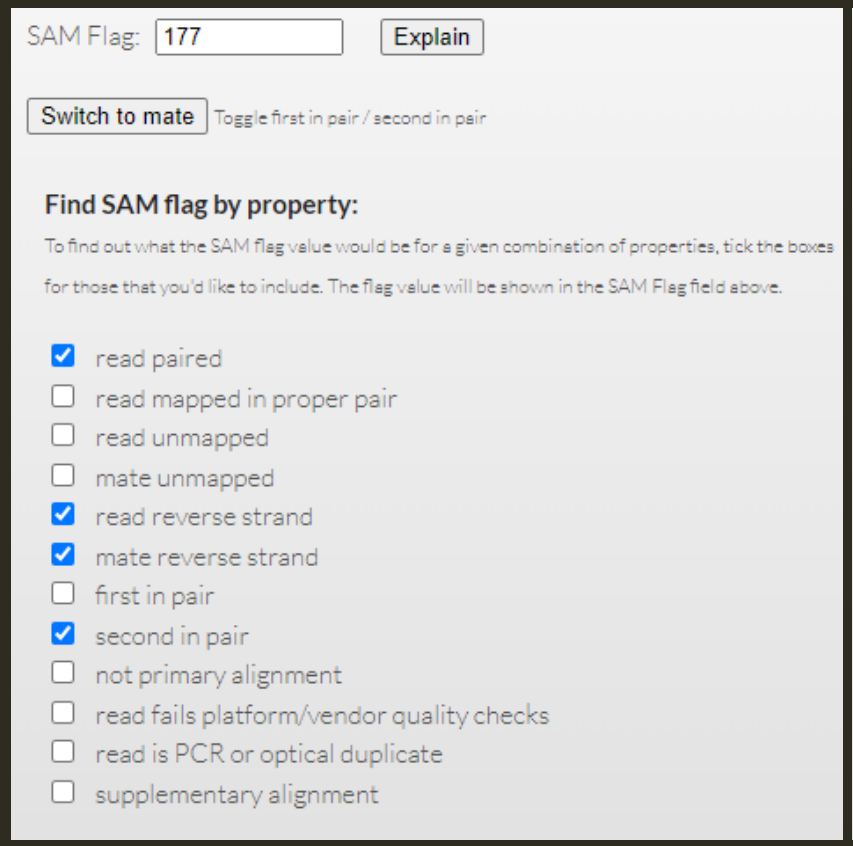
CIGAR - PRZYKŁAD

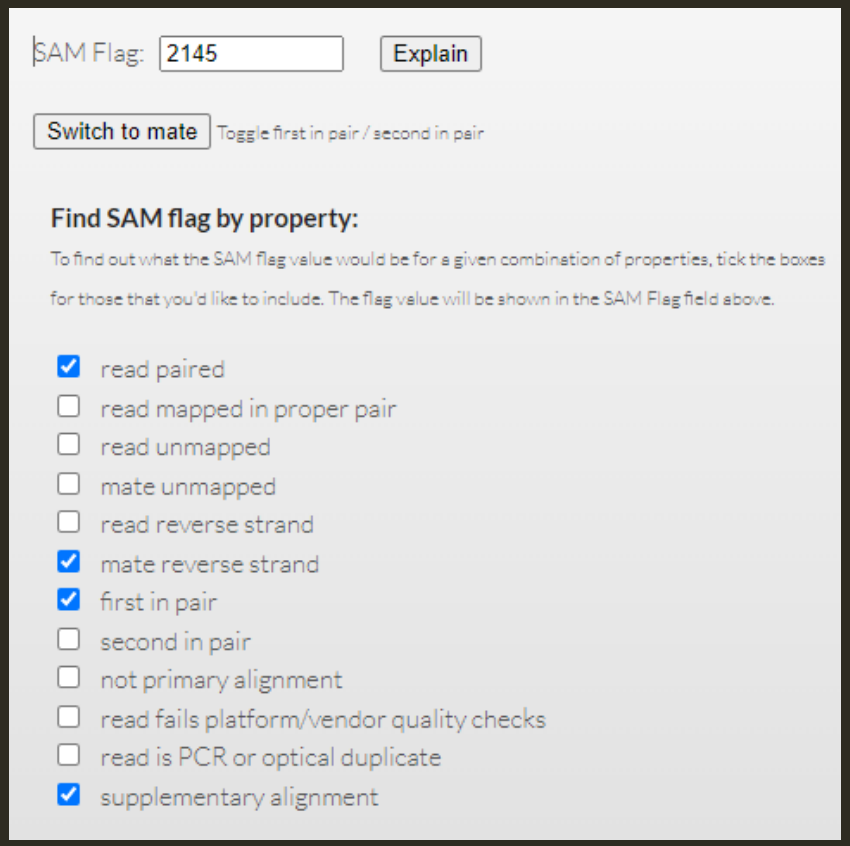
samtools view -F 4 S2709Nr12.merged.bam | grep A00788:260:HV32JDMXX:2:1271:13313:3286

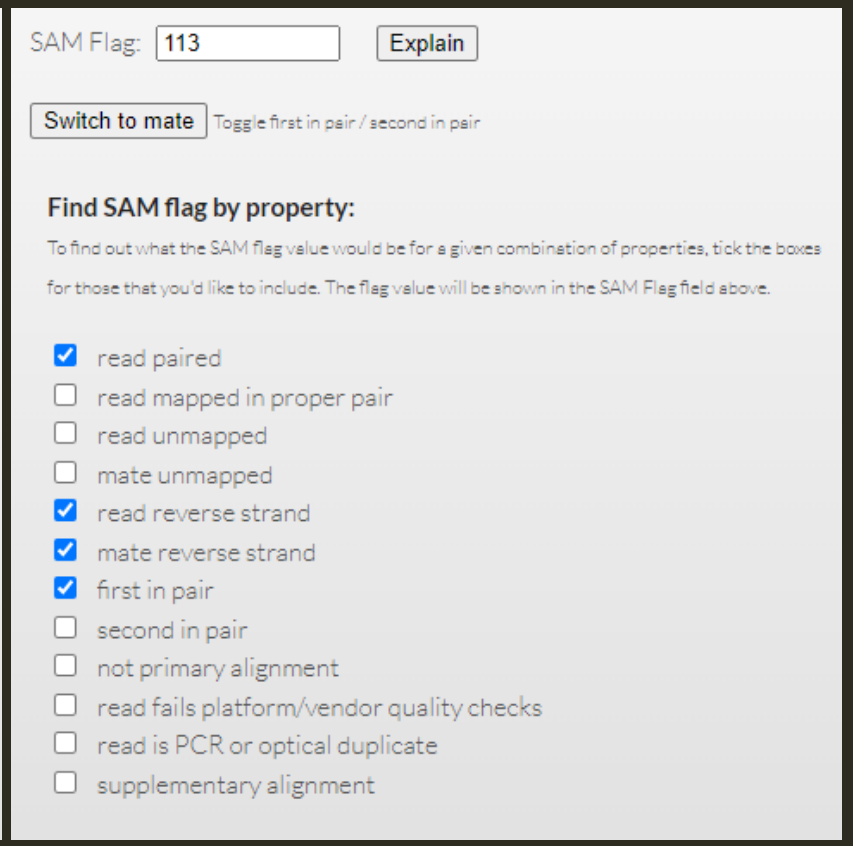
A00788:260:HV32JDMXX:2:1271:13313:3286 177 1 145 60 151M = 5438 5260
CTGGACATAAAAGAGGAATTAGCATGTGCACTCTTCAGATATAAATGCCATCAGTATTTTCCTATTAAAATGAAGCTTGTTTTCATCTCAGTGGAAATCTGTGGCTAAAGTACAACAATAGTAAT
GATAATGGTGAGGCTGTTGTACTTCA FFFFFFFFF;:FFF:FFFFFFFFFFFF:FFFFFFFFFF:F:FF,FF:FFFFFF:FFFFFFFF:FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFF:FFFFFFFFFFFFFFFFFFFF NM:i:0 MD:Z:151 AS:i:151 XS:i:20 MQ:i:60 MC:Z:33S118M ms:i:5563
RG:Z:S2709Nr12.trimm.rg2.fixmate.srt.markdup
A00788:260:HV32JDMXX:2:1271:13313:3286 2145 1 1 60
116H35M = 145 295 GCTTAATTTTTGTCATTTCTCACCCCTGCTCTTGA FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF NM:i:0 MD:Z:35
MC:Z:151M AS:i:35XS:i:19 SA:Z:1,5438 ,33S118M,60,0; RG:Z:S2709Nr12.trimm.rg2.fixmate.srt.markdup
A00788:260:HV32JDMXX:2:1271:13313:3286 113 1 5438 60 33S118M = 145 -
5260 TCAAGAGCAGGGGTGAGAAATGACAAAAATTAAGCACATAATCACGCTCAACTATATGGAAATCAGATGATAACACACATCCTTACTCATGCCTGATTTTAAGAAAATGAAGTTACT
GCTGTCTGCAGTGATGAAAATCAACCACCTTAAC FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF:FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF:F NM:i:0 MD:Z:118 AS:i:118 XS:i:20 SA:Z:1,1,+,116S35M,60,0; MQ:i:60
MC:Z:151M ms:i:5405 RG:Z:S2709Nr12.trimm.rg2.fixmate.srt.markdup

S	4	soft clipping (clipped sequences present in SEQ)
H	5	hard clipping (clipped sequences NOT present in SEQ)

FLAG - WYJAŚNIENIE

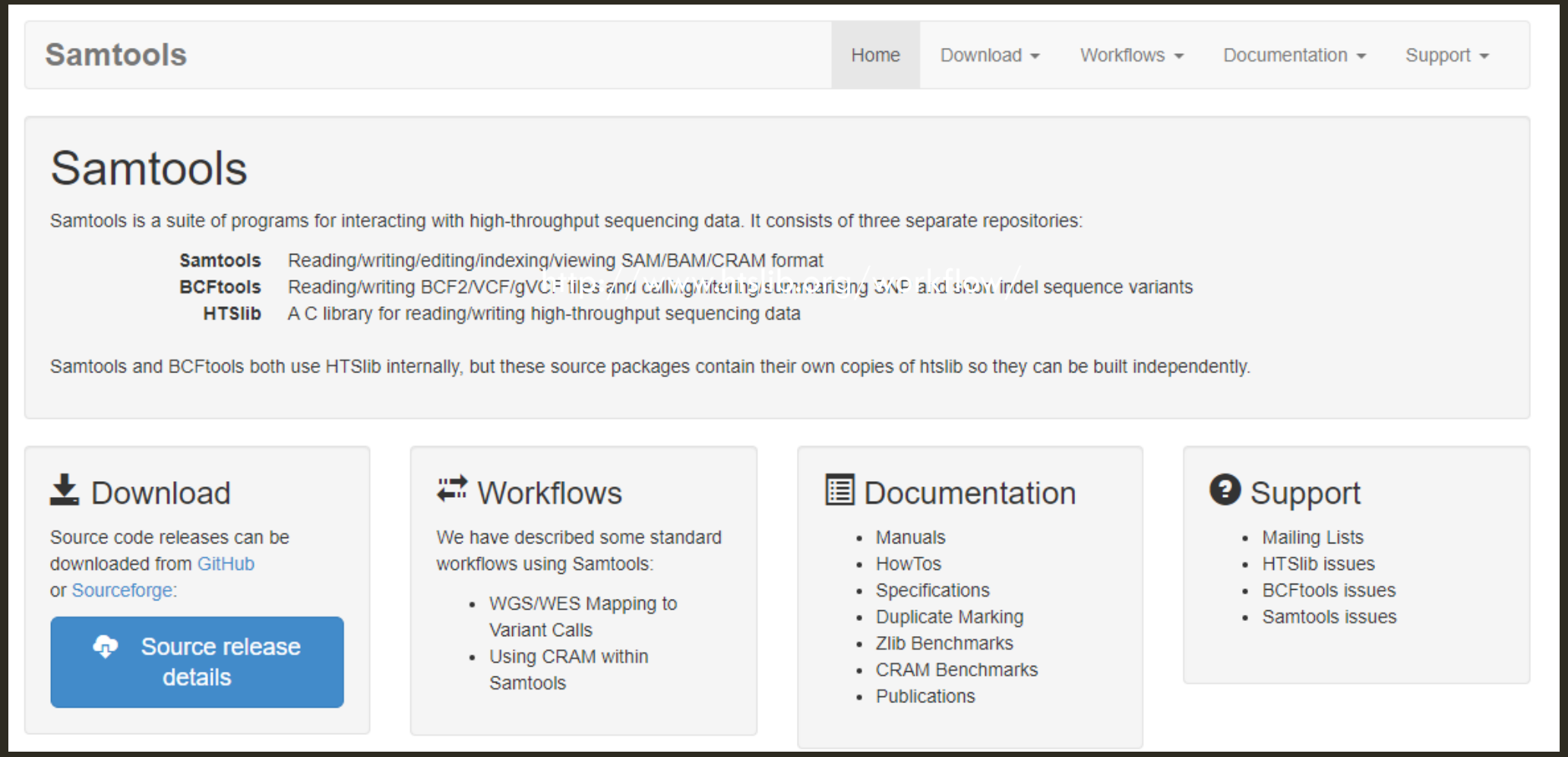






http://www.htslib.org/

POST-ALIGNMENT PROCESSING



POST-ALIGNMENT PROCESSING

Manual page from samtools-1.10
released on 6 December 2019

NAME

samtools – Utilities for the Sequence Alignment/Map (SAM) format

SYNOPSIS

samtools view -bt ref_list.txt -o aln.bam aln.sam.gz

samtools sort -T /tmp/aln.sorted -o aln.sorted.bam aln.bam

samtools index aln.sorted.bam

samtools idxstats aln.sorted.bam

samtools flagstat aln.sorted.bam

samtools stats aln.sorted.bam

samtools bedcov aln.sorted.bam

samtools depth aln.sorted.bam

samtools view aln.sorted.bam chr2:20,100,000-20,200,000

samtools merge out.bam in1.bam in2.bam in3.bam

samtools faidx ref.fasta

samtools fqidx ref.fastq

POST-ALIGNMENT PROCESSING - PRZYKŁAD

samtools view

samtools merge
samtools fixmate
samtools sort
samtools markdup
samtools index
samtools flagstat
samtools depth

POST-ALIGNMENT PROCESSING - PRZYKŁAD

Manual page from samtools-1.15
released on 21 February 2022

NAME

samtools view – views and converts SAM/BAM/CRAM files

SYNOPSIS

samtools view [*options*] *in.sam*|*in.bam*|*in.cram* [*region*...]

DESCRIPTION

With no options or regions specified, prints all alignments in the specified input alignment file (in SAM, BAM, or CRAM format) to standard output in SAM format (with no header).

You may specify one or more space-separated region specifications after the input filename to restrict output to only those alignments which overlap the specified region(s). Use of region specifications requires a coordinate-sorted and indexed input file (in BAM or CRAM format).

The **-b**, **-C**, **-1**, **-u**, **-h**, **-H**, and **-c** options change the output format from the default of headerless SAM, and the **-o** and **-U** options set the output file name(s).

The **-t** and **-T** options provide additional reference data. One of these two options is required when SAM input does not contain @SQ headers, and the **-T** option is required whenever writing CRAM output.

FORMAT BAM

Binary Alignment/Map Format:
- binarny odpowiednik formatu SAM
- zajmuje mniej pamięci dysku (stanowi ok. 27% oryginalnego pliku w formacie SAM)

Unikamy zapisywania plików w formacie SAM! Tworzymy od razu BAM!

Uwaga!
- „For example, mapping 8.5 TB of FASTQ files would require about 140 TB of storage for temporary files, while the final mapped files measure only 9.9 TB.” (https://doi.org/10.1093/bioinformatics/btad028)

Obsługa BAM poprzez narzędzie samtools view!
- samtools view file.bam | head

POST-ALIGNMENT PROCESSING - PRZYKŁAD

Manual page from samtools-1.15
released on 21 February 2022

NAME

samtools merge – merges multiple sorted files into a single file

SYNOPSIS

samtools merge [*options*] **-o** *out.bam* [*options*] *in1.bam ... inN.bam*

samtools merge [*options*] *out.bam in1.bam ... inN.bam*

DESCRIPTION

Merge multiple sorted alignment files, producing a single sorted output file that contains all the input records and maintains the existing sort order.

The output file can be specified via **-o** as shown in the first synopsis. Otherwise the first non-option filename argument is taken to be *out.bam* rather than an input file, as in the second synopsis. There is no default; to write to standard output (or to a pipe), use either "**-o -**" or the equivalent using "**-**" as the first filename argument.

If **-h** is specified the @SQ headers of input files will be merged into the specified header, otherwise they will be merged into a composite header created from the input headers. If in the process of merging @SQ lines for coordinate sorted input files, a conflict arises as to the order (for example input1.bam has @SQ for a,b,c and input2.bam has b,a,c) then the resulting output file will need to be re-sorted back into coordinate order.

POST-ALIGNMENT PROCESSING - PRZYKŁAD

Samtools fixmate is a tool that can fill in information (insert size, cigar, mapq) about paired end reads onto the corresponding other read. Also has options to remove secondary/unmapped alignments and recalculate whether reads are proper pairs.

SYNOPSIS

samtools fixmate [**-rpcmu**] [**-O** *format*] *in.nameSrt.bam out.bam*

DESCRIPTION

Fill in mate coordinates, ISIZE and mate related flags from a name-sorted or name-collated alignment.

OPTIONS

-r	Remove secondary and unmapped reads.
-p	Disable FR proper pair check.
-c	Add template cigar ct tag.
-m	Add ms (mate score) tags. These are used by **markdup** to select the best reads to keep.

POST-ALIGNMENT PROCESSING - PRZYKŁAD

Manual page from samtools-1.15
released on 21 February 2022

NAME

samtools sort – sorts SAM/BAM/CRAM files

SYNOPSIS

samtools sort [**-l** *level*] [**-u**] [**-m** *maxMem*] [**-o** *out.bam*] [**-O** *format*] [**-M**] [**-K** *kmerLen*] [**-n**] [**-t** *tag*] [**-T** *tmpprefix*] [**-@** *threads*] [*in.sam*|*in.bam*|*in.cram*]

DESCRIPTION

Sort alignments by leftmost coordinates, or by read name when **-n** is used. An appropriate **@HD-SO** sort order header tag will be added or an existing one updated if necessary.

The sorted output is written to standard output by default, or to the specified file (*out.bam*) when **-o** is used. This command will also create temporary files *tmpprefix.%d*.**bam** as needed when the entire alignment data cannot fit into memory (as controlled via the **-m** option).

Consider using **samtools collate** instead if you need name collated data without a full lexicographical sort.

Note that if the sorted output file is to be indexed with **samtools index**, the default coordinate sort must be used. Thus the **-n** and **-t** options are incompatible with **samtools index**.

POST-ALIGNMENT PROCESSING - PRZYKŁAD

Manual page from samtools-1.15
released on 21 February 2022

NAME

samtools markdup – mark duplicate alignments in a coordinate sorted file

SYNOPSIS

samtools markdup [**-l** *length*] [**-r**] [**-s**] [**-T**] [**-S**] [**-f** *file*] [**-d** *distance*] [**-c**] [**-t**] [**-m**] [**--mode**] [**--include-fails**] [**--no-PG**] [**-u**] [**--no-multi-dup**] [**--read-coords**] [**--coords-order**] *in.algsort.bam out.bam*

DESCRIPTION

Mark duplicate alignments from a coordinate sorted file that has been run through **samtools fixmate** with the **-m** option. This program relies on the MC and ms tags that fixmate provides.

OPTIONS

-l *INT* Expected maximum read length of *INT* bases. [300]

-r Remove duplicate reads.

POST-ALIGNMENT PROCESSING - PRZYKŁAD

Manual page from samtools-1.15
released on 21 February 2022

NAME

samtools index – indexes SAM/BAM/CRAM files

SYNOPSIS

samtools index [**-bc**] [**-m** *INT*] *aln.sam*|*aln.bam*|*aln.cram* [*out.index*]

DESCRIPTION

Index a coordinate-sorted BGZIP-compressed SAM, BAM or CRAM file for fast random access. Note for SAM this only works if the file has been BGZF compressed first.

This index is needed when *region* arguments are used to limit **samtools view** and similar commands to particular regions of interest.

If an output filename is given, the index file will be written to *out.index*. Otherwise, for a CRAM file *aln.cram*, index file *aln.cram***.crai** will be created; for a BAM file *aln.bam*, either *aln.bam***.bai** or *aln.bam***.csi** will be created; and for a compressed SAM file *aln.sam.gz*, either *aln.sam.gz***.bai** or *aln.sam.gz***.csi** will be created, depending on the index format selected.

The BAI index format can handle individual chromosomes up to 512 Mbp (2^29 bases) in length. If your input file might contain reads mapped to positions greater than that, you will need to use a CSI index.

POST-ALIGNMENT PROCESSING - PRZYKŁAD

Manual page from samtools-1.15
released on 21 February 2022

NAME

samtools flagstat – counts the number of alignments for each FLAG type

SYNOPSIS

samtools flagstat *in.sam|in.bam|in.cram*

DESCRIPTION

Does a full pass through the input file to calculate and print statistics to stdout.

Provides counts for each of 13 categories based primarily on bit flags in the FLAG field. Information on the meaning of the flags is given in the SAM specification document <https://samtools.github.io/hts-specs/SAMv1.pdf>.

Each category in the output is broken down into QC pass and QC fail. In the default output format, these are presented as "#PASS + #FAIL" followed by a description of the category.

The first row of output gives the total number of reads that are QC pass and fail (according to flag bit 0x200). For example:

122 + 28 in total (QC-passed reads + QC-failed reads)

Which would indicate that there are a total of 150 reads in the input file, 122 of which are marked as QC pass and 28 of which are marked as "not passing quality controls"

POST-ALIGNMENT PROCESSING - PRZYKŁAD

```
968022158 + 0 in total (QC-passed reads + QC-failed reads)
0 + 0 secondary
3562322 + 0 supplementary
0 + 0 duplicates
967737165 + 0 mapped (99.97% : N/A)
964459836 + 0 paired in sequencing
482228158 + 0 read1
482231678 + 0 read2
954809642 + 0 properly paired (99.00% : N/A)
964125647 + 0 with itself and mate mapped
49196 + 0 singletons (0.01% : N/A)
5949584 + 0 with mate mapped to a different chr
2249044 + 0 with mate mapped to a different chr (mapQ>=5)
```

POST-ALIGNMENT PROCESSING - PRZYKŁAD

```
968022158 + 0 in total (QC-passed reads + QC-failed reads)
0 + 0 secondary
3562322 + 0 supplementary
0 + 0 duplicates
967737165 + 0 mapped (99.97% : N/A)
964459836 + 0 paired in sequencing
482228158 + 0 read1
482231678 + 0 read2
954809642 + 0 properly paired (99.00% :
964125647 + 0 with itself and mate mapp
49196 + 0 singletons (0.01% : N/A)
5949584 + 0 with mate mapped to a diffe
2249044 + 0 with mate mapped to a diffe
```

Program	Time (h)	Conf (%)	Paired (%)
Bowtie	5.2	84.4	96.3
BWA	4.0	88.9	98.8
MAQ	94.9	86.1	98.7
SOAP2	3.4	88.3	97.5

The 12.2 million read pairs were mapped to the human genome. CPU time in hours on a single core of a 2.5 GHz Xeon E5420 processor (Time), percent confidently mapped reads (Conf) and percent confident mappings with the mates mapped in the correct orientation and within 300 bp (Paired), are shown in the table.

POST-ALIGNMENT PROCESSING - PRZYKŁAD

Manual page from samtools-1.15
released on 21 February 2022

NAME

samtools depth – computes the read depth at each position or region

SYNOPSIS

samtools depth [*options*] [*in1.sam|in1.bam|in1.cram* [*in2.sam|in2.bam|in2.cram*] [...]]

DESCRIPTION

Computes the depth at each position or region.

OPTIONS

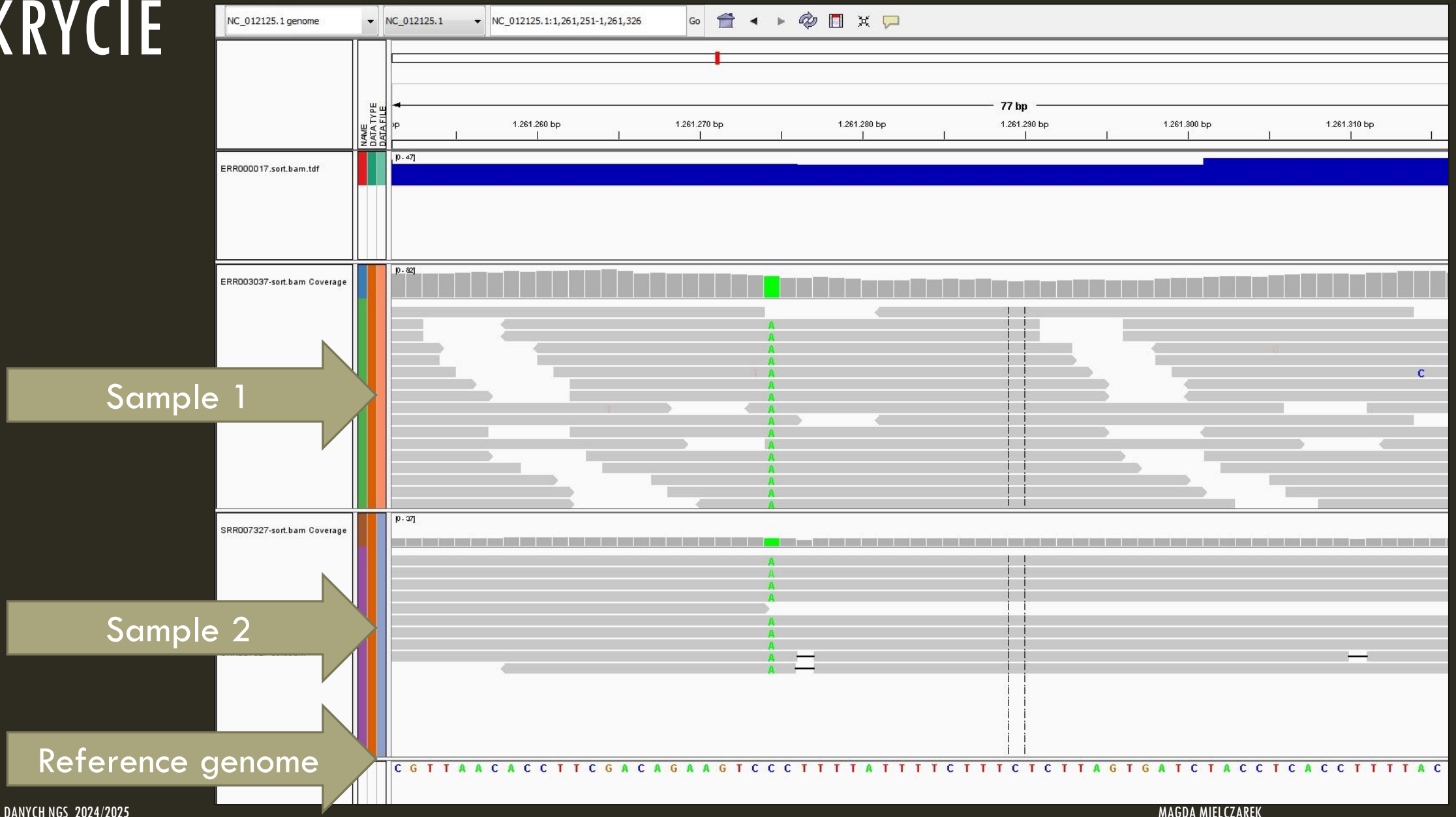
-a Output all positions (including those with zero depth)

-a -a, -aa Output absolutely all positions, including unused reference sequences. Note that when used in conjunction with a BED file the -a option may sometimes operate as if -aa was specified if the reference sequence has coverage outside of the region specified in the BED file.

-b *FILE* Compute depth at list of positions or regions in specified BED *FILE*. []

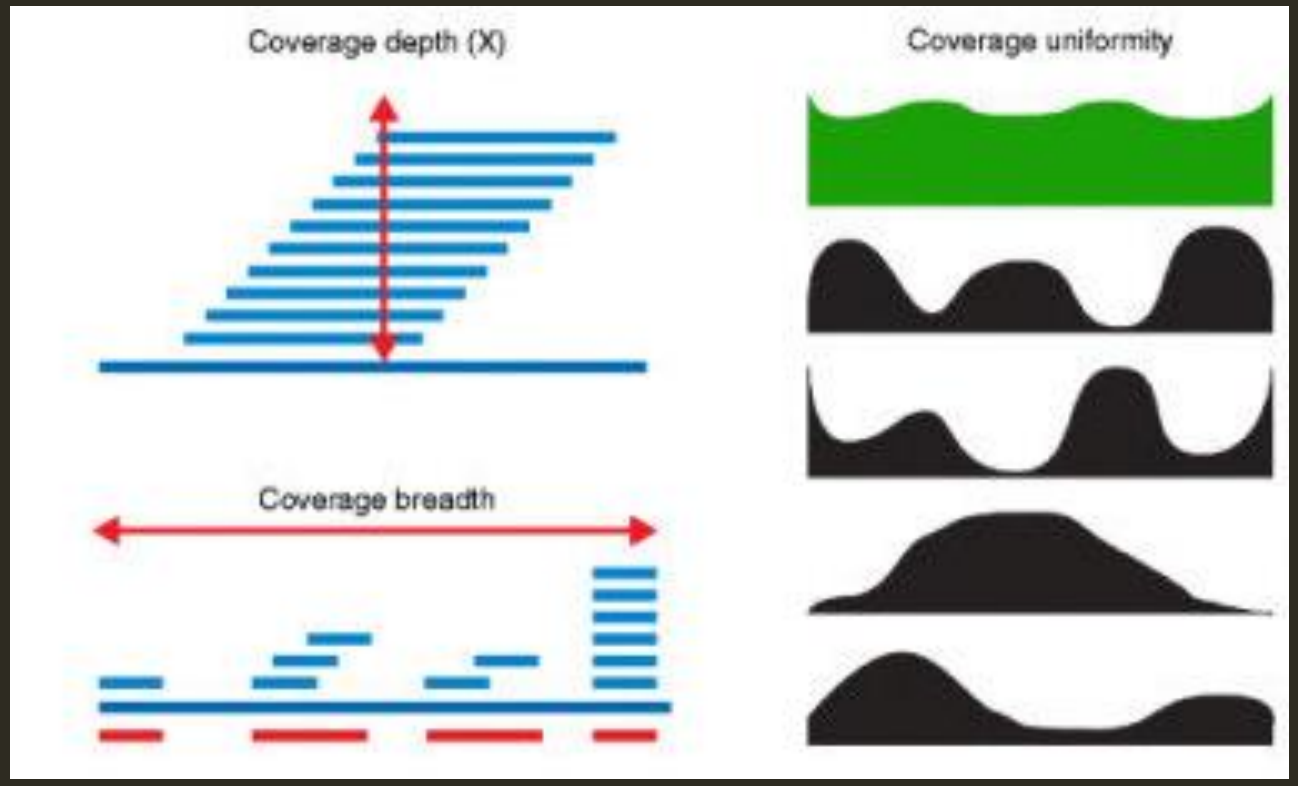
http://software.broadinstitute.org/software/igv/

POKRYCIE



POKRYCIE

- Technical errors and coverage in sequencing data



Navin, N.E. Cancer genomics: one cell at a time. Genome Biol 15, 452 (2014). https://doi.org/10.1186/s13059-014-0452-9

- PCR duplicates

POKRYCIE

Obliczenie pokrycia (po przyrównaniu)

- Skrypt własny
  - coverage = (read count * read length) / total genome size.
- Samtools depth
  - computes the read depth at each position or region
  - samtools depth [options] [in1.sam|in1.bam|in1.cram [in2.sam|in2.bam|in2.cram] [...]]
- Bedtools
  - bedtools coverage tool computes both the depth and breadth of coverage of features
  - bedtools coverage / coverageBed

Średnie pokrycie genomu