Accessing Public Experimental Data

Dr. Princess Rodriguez

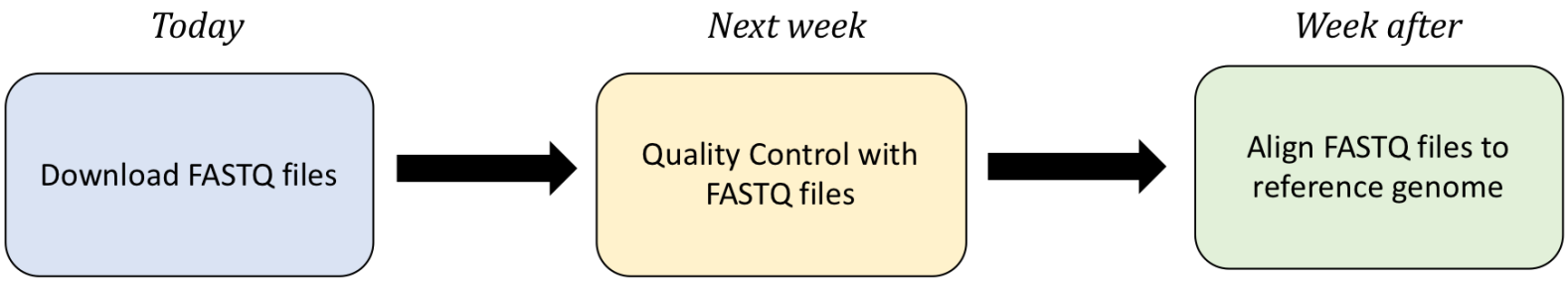
2025-01-29

Learning Objectives

We spent the last few weeks introducing UNIX, navigating the file system, and working on a high performance cluster. Now we will proceed with:

- Understand the types of data that are accessible from Gene Expression Omnibus (GEO)
- Learning how to use SRA-toolkit to retrieve data from the Sequence Reads Archive
  - Download data from the SRA with `fastq-dump`
  - split files into forward and reverse reads
  - Download part, not all, the data

Where are we heading?



- Where do I download from? GEO
- What bioinformatic tool do I use to perform the download? sratoolkit
- How can I use sratoolkit? A) Environmental modules B) Job submission

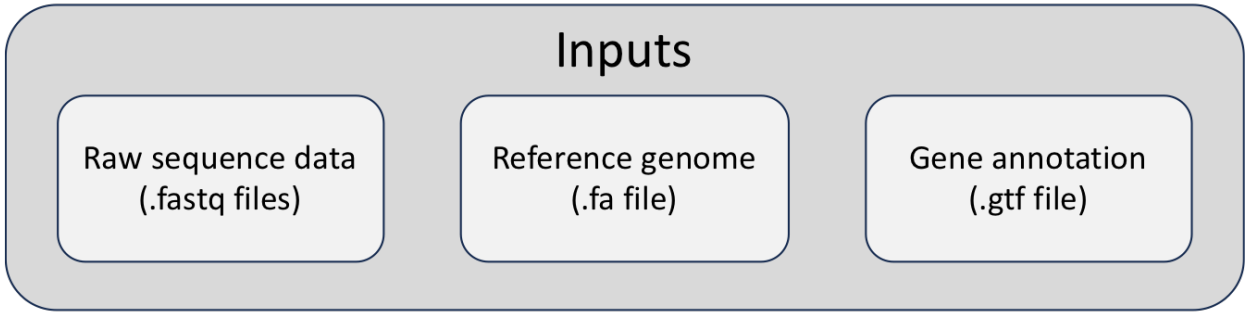
I want to stress to "learn" these fundamentals in data processing we are using RNA-Seq as the example. But this outline can be applied to most big data analysis. Its just about identifying the proper bioinformatic tool along the way!

Figure 1: Overview

Sequence file formats

Below is a cartoon displaying the (3) file types required to perform an RNA-Seq analysis.

- FASTQ files will contain the raw sequence reads
- The reference genome will be in the form of a FASTA file
- Gene annotations will be in the form of a GTF file



If you are not performing RNA-Seq analysis, what are your inputs and where would you find them?

Figure 2: Required File Inputs

FASTA file

During an NGS experiment, the nucleotide sequences stored inside the raw FASTQ files, or "sequence reads", need to be mapped or aligned to the reference genome to determine from where these sequences originated. Therefore, we need a reference genome (in FASTA format) in which to align our sequences.

```
>NC_000006.12:151654148-152129619 Homo sapiens chromosome 6, GRCh38.p13 Primary
Assembly
TATTGATTTTTGTGTAAACATGTGTTTGTATATATCTATAACGAGAACTCAAGTCATACTGTAATCCTAT
TTTGTAAACTGACTTTTTTCCTTTATCAGTATATCAAGATTATTTTCCCACATCATTTGACATTTTTTCT
ACAGTGTAATTTAATGGCTACATTGTTTTCTATCCTATGAATATATCAAACCTATTTCTTAAAAACCCTA
CTCAGGGATTTTAAAAAATAAAAACGATGTTTTAATATTATAAAGATTCAGTGAGGTATATTCTTATACG
TACACATTTCTAAGGTTTGAGTTCTTACAAGATGCTGAACTAGCTAAGACTACTGGTTCTCATCTGTCAC
ATAGGGAAAAATTATAGAAGGAAAACATCAAGATTTGGAAAAATCTGTGAGAATTGTTTTGCATTAGTGT
GTAGGTGTGTGTGTTGGGGTGGTGGCTGCAGCTTGGGGCAGAGGCCTCAGGTGTGGCTGTGGAGTGATCA
GATAGAGTTTTTGGAGTTCGGCTTTTGCCCCAGGACACTTGGTGCCTGCCCCCAGAGCTGCAGCCCAGAA
GGCCGTTCTCAGAGGTGAAGTCCAGGCAGTGAGGAGCTGTCTGCCAGTAGGCAGTTGAAGAAAAAAAATG
AGCTAGAGGAAAAAAACAAAAAAACAAAATCTCCTTCTAATGCTGCCAGGCTGCCGGGAGCTGGAAATGA
AGCACTGACAGGAGTGGGTATTTCATGGTGAAGGGAATAATCAACTGGTTTTTTTGGTACCCAAGACTTT
CCACCTTCACACACACACATGAGATGCTTTGAAATAAAGATAGTCACTTGACTTAGTAAAGTTTGTTGAC
ATAAAAATATGAGAAATACCAAAGAATACAAAAAGGAAAACTTCGTTAATATTATTCAGACTTAAAATTC
CAGATTGTATCAACATTAAGGGGGTTGATGAAAACATGGGAGAAAGCCAAGGGACGTGAGATCGGGCTCA
ATTCTTGACTTGCTGGGGGAAGGTATCAACACAGAACTTTTAAGAATTAGAAGGCATTAAAAAGAAATAG
AAATCCTGAATCAAATTGAAACAGTAAAATAAAATAGTCCAAAGATGTGTAAATATATCACTATCACAAT
```

HEADER

SEQUENCE

GTF file

In addition, many NGS methods require knowing where known genes or exons are located on the genome in order to quantify the number of reads aligning to different genome features, such as exons, introns, transcription start sites, etc. These analyses require reference data containing specific information about genomic coordinates of various genomic "features", such as gene annotation files (in GTF, GFF, etc.).

Col 1	Col 2	Col 3	Col 4	Col 5	Col 6	Col 7	Col 8	Col 9
chr21	HAVANA	transcript	10862622	10863067	.	+	.	gene_id "ENSG00000169..
chr21	HAVANA	exon	10862622	10862667	.	+	.	gene_id "ENSG00000169..
chr21	HAVANA	CDS	10862622	10862667	.	+	0	gene_id "ENSG00000169..
chr21	HAVANA	start_codon	10862622	10862624	.	+	0	gene_id "ENSG00000169..
chr21	HAVANA	exon	10862751	10863067	.	+	.	gene_id "ENSG00000169..
chr21	HAVANA	CDS	10862751	10863064	.	+	2	gene_id "ENSG00000169..
chr21	HAVANA	stop_codon	10863065	10863067	.	+	0	gene_id "ENSG00000169..
chr21	HAVANA	UTR	10863065	10863067	.	+	.	gene_id "ENSG00000169..

Figure 4: GTF format

FASTQ file

These are the extension of FASTA files which contain quality scores and are output from the NGS technologies.

Downloading file formats

To find and download NGS experimental data and associated reference data we will explore a few key repositories.

- For **finding reference data**, we will navigate the Ensembl database.
- For **accessing experimental data**, we will explore the Gene Expression Omnibus and the Sequence Read Archive repositories.

General vs organism-specific databases

- **General biological databases:** Ensembl, NCBI, and UCSC
- **Organism-specific biological databases:** Wormbase, Flybase, Cryptodb, etc. (often updated more frequently, so may be more comprehensive)
  - Sometime's you will need to pay annual membership fee to access files and use organism specific tools.

Human Reference Genome

The **current genome build** is GRCh38/hg38 for the human, which was released in 2013 and is maintained by the Genome Reference Consortium (GRC).

Figure 5: GRC logo

Differences from Biological Databases

Genome databases incorporate these genomes and generate the gene annotations with the following **similarities/differences**:

- **Ensembl, NCBI, and UCSC** all use the **same genome assemblies or builds** provided by the GRC
  - GRCh38 = hg38; GRCh37 = hg19
- Each biological database **independently determines the gene annotations**; therefore, gene annotations between these databases can differ, even though the genome assembly is the same. Naming conventions are also different (chr1=1) between databases.
- **Always use the same biological database for all reference data!**

Ensembl

Ensembl provides a website that acts as a **single point of access to annotated genomes** for vertebrate species.

For all other organisms there are additional Ensembl databases available through Ensembl Genomes; however, they do not include viruses (NCBI does).

Ensembl annotations updates

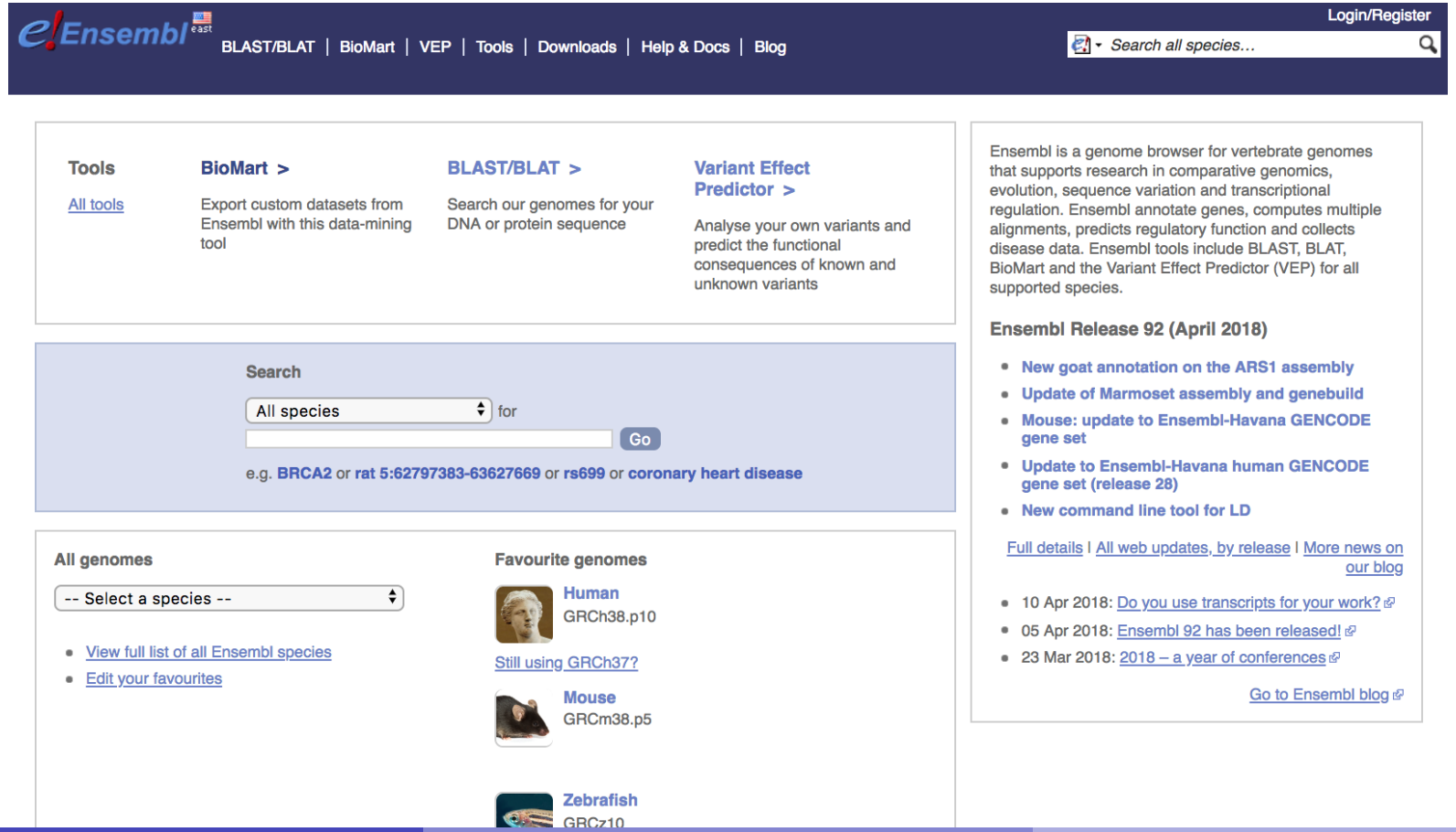
- **Genome assemblies/builds (reference genomes)**
  - New genome builds are released every few years or more depending on the species
  - Genome assemblies are typically updated every two years to include patches, but sometimes less often depending on the species

Ensembl annotations updates continued

- **Gene annotations**
  - Gene annotations are created or updated using a variety of sources (ENA, UniProtKB, NCBI RefSeq, RFAM, miRBase, and tRNAscan-SE databases)
  - Automatic annotation is performed for all species using identified proteins and transcripts
  - Manual curation by the HAVANA group is performed for human, mouse, zebrafish, and rat species, providing better confidence of transcript annotations
  - Directly imports annotations from FlyBase, WormBase and SGD

Ensembl database

Navigate to the Ensembl website to view the interface. The homepage for Ensembl has a lot to offer, with the a lot of information and access to a range of functionality and tools.



Ensembl identifiers

- **Ensembl identifiers**: When using Ensembl, note that it uses the following format for biological identifiers:
  - **ENSG##########:** Ensembl Gene ID
  - **ENST##########:** Ensembl Transcript ID
  - **ENSP##########:** Ensembl Peptide ID
  - **ENSE##########:** Ensembl Exon ID

  For non-human species a suffix is added:
  - **ENSMUSG###:** MUS (Mus musculus) for mouse
  - **ENSDARG###:** DAR (Danio rerio) for zebrafish

Downloading reference data from Ensembl

- Go to Downloads, then click FTP Download on the left side bar.

Show 10 entries Show/hide columns

★	Species	DNA (FASTA)	cDNA (FASTA)	CDS (FASTA)	ncRNA (FASTA)	Protein sequence (FASTA)	Annotated sequence (EMBL)	Annotated sequence (GenBank)	Gene sets	Other annotations	Whole databases	Variation (GVF)	Variat (VCF
Y	Human *Homo sapiens*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	TSV JSON	MySQL	GVF	VCF
Y	Mouse *Mus musculus*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	TSV JSON	MySQL	GVF	VCF
Y	Zebrafish *Danio rerio*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	TSV JSON	MySQL	GVF	VCF
	Abingdon island giant tortoise *Chelonoidis abingdonii*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	TSV JSON	MySQL	-	-
	African ostrich *Struthio camelus australis*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	TSV JSON	MySQL	-	-
	Agassiz's desert tortoise *Gopherus agassizii*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	TSV JSON	MySQL	-	-
	Algerian mouse *Mus spretus*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	TSV JSON	MySQL	-	-
	Alpaca *Vicugna pacos*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	TSV JSON	MySQL	-	-
	Alpine marmot *Marmota marmota marmota*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	TSV JSON	MySQL	-	-
	Amazon molly *Poecilia formosa*	FASTA	FASTA	FASTA	FASTA	FASTA	EMBL	GenBank	GTF GFF3	TSV JSON	MySQL	-	-

Showing 1 to 10 of 317 entries

Class Exercise

Amanda is a graduate student studying optimal breeding practices for cattle. They are interested in investigating transcriptional differences in cattle raised in tropical versus temperate conditions.
Amanda needs to download the **Bos taurus FASTA file** to set up their pipeline on the VACC.
Amanda comes to you for help.
How would you download the Bos taurus FASTA file from Ensembl?
Once downloaded perform `head` to view the file.

Gene Expression Omnibus (GEO)

- GEO is a database for curated functional genomics data, including gene expression datasets from microarrays, RNA-Seq, and other transcriptomic studies.
- It stores processed and analyzed data, such as gene expression matrices and differential expression results.
- This database provides access to data for tens of thousands of studies as it is a requirement for publication.
- For datasets containing sequencing data, GEO often links to the Sequence Read Archive (SRA) (also maintained by NCBI).
- Users can access the SRA database to download raw sequencing data files in the FASTQ format.

To download FASTQ(s) from GEO, you need the following:

1) A list of accession numbers (SRRXXXXXX format) for the files you wish to download. Use **Run Selector** to acquire this list.
2) Knowledge on how to access and use `fastq-dump`
3) An understanding of how to submit a script using **SLURM** batch system

Finding GEO data for a particular publication

The publication will provide the GEO accession number. Let's find the data associated with the paper, "MOV10 and FRMP regulate AGO2 association with microRNA recognition elements".

1. Search for PMC4268400
2. Seach for GEO Accession # in the article.

Step 1: Identify the article

Figure 8: Kenny et al. 2014 dataset

Step 2: Find the "GEO" Accession

Good search terms include "RNA-Seq", "Gene Expression Omnibus", "Supplementary Data"

Article

Switch to Standard View

MOV10 and FMRP Regulate AGO2 Association with MicroRNA Recognition Elements

Phillip J. Kenny[6], Hongjun Zhou[6], Miri Kim[6], Geena Skariah, Radhika S. Khetani, Jenny Drnevich, Mary Luz Arcila, Kenneth S. Kosik, Stephanie Ceman

▶ Author Contributions

▶ Acknowledgments

▼ Accession Numbers

All iCLIP data files and RNA-seq files are available from the NCBI Gene Expression Omnibus (http://www.ncbi.nlm.nih.gov/geo/) under the accession numbers GSE51443 (MOV10 iCLIP-SEQ) and GSE50499 (RNA-seq).

▶ Supplemental Information

Step 3: Open GEO page for experiment

Please Note: Many paper have multiple GEO accession numbers. Each will correspond to a specific dataset

NCBI

Gene Expression Omnibus

HOME | SEARCH | SITE MAP | GEO Publications | FAQ | MIAME | Email GEO

NCBI > GEO > **Accession Display** | Not logged in | Login

Scope: Self | Format: HTML | Amount: Quick | GEO accession: GSE51443 | GO

Series GSE51443 | Query DataSets for GSE51443

Status	Public on Nov 20, 2014
Title	Identification of the cellular RNAs bound by MOV10
Organism	Homo sapiens
Experiment type	Expression profiling by high throughput sequencing
Summary	Using the iCLIP protocol we have identified the cellular RNA entities that are bound by MOV10. We report the location and sequence of the MOV10 binding region on each RNA entity.
Overall design	To identify the RNAs that bound MOV10, we UV-cross-linked HEK293F cells and immunoprecipitated with an irrelevant antibody (ir or "control") followed by a MOV10-specific antibody (MOV10) to isolate associated RNAs after stringent washing.
Contributor(s)	Kim M, Kenny PJ, Khetani RS, Arcila ML, Kosik KS, Ceman S

GEO page Anatomy

The GEO page contains information about the experiment, including:

- an experimental summary: gives you an understanding of *how* the experiment was performed.
- literature citation
- contact information
- links to the Sample GEO pages: each sample will have its own page with additional information regarding how the sample was generated and analyzed
- link to the SRA project containing the raw FASTQ files

GEO page also contains processed data

In addition, if we were interested in **downloading the raw counts matrix (`GSE50499_GEO_Ceman_counts.txt.gz`)** we could scroll down to **supplementary data** at the bottom of the page. This provides the , the number of reads/sequences aligning to each gene.

Download family	Format
SOFT formatted family file(s)	SOFT
MINiML formatted family file(s)	MINiML
Series Matrix File(s)	TXT

Supplementary file	Size	Download	File type/resource
GSE50499_GEO_Ceman_counts.txt.gz	320.2 Kb	(ftp)(http)	TXT

Raw data are available in SRA

Processed data is available on Series record

Figure 11: Raw Counts Download

Step 4: Click on the SRP Accession

- Towards the bottom of the GEO page you will find a link for **SRA** under the heading **Relations**.
- The Sequence Read Archive (SRA) is an archive for high throughput sequencing data, publicly accessible, for the purpose of enhancing reproducibility in the scientific community.

Platforms (1)	GPL11154 Illumina HiSeq 2000 (Homo sapiens)
Samples (8)	GSM1220262 MOV10 knockdown 2
Less...	GSM1220263 MOV10 knockdown 3
	GSM1220264 MOV10 overexpression 1
	GSM1220265 MOV10 overexpression 2
	GSM1220266 MOV10 overexpression 3
	GSM1220267 irrelevant siRNA 1
	GSM1220268 irrelevant siRNA 2
	GSM1220269 irrelevant siRNA 3
Relations	
BioProject	PRJNA217781
SRA	SRP029367

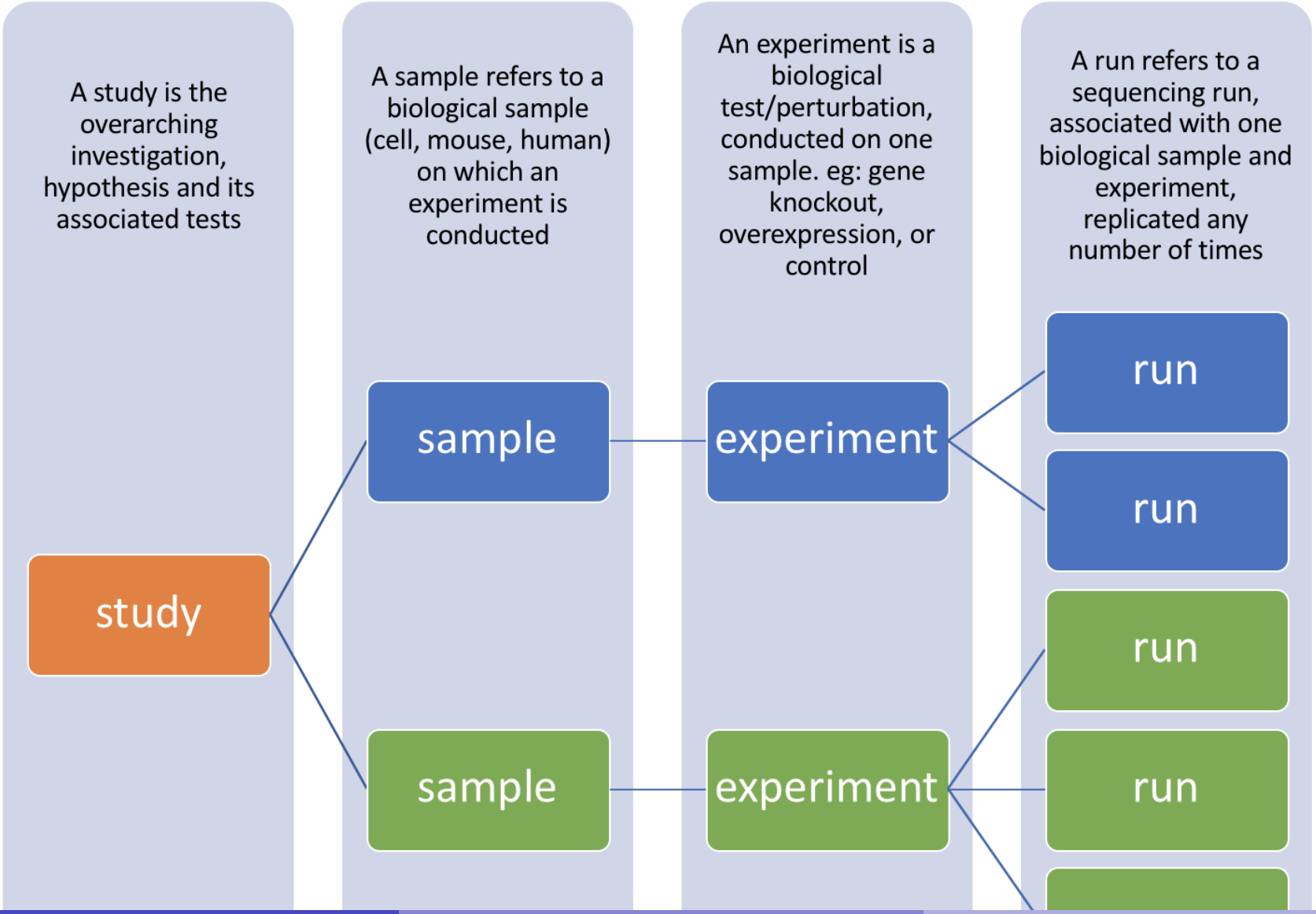
Figure 12: SRA Platforms, Samples, and Relations displayed

SRA Hierarchy & Accessions

There are four hierarchical levels of SRA entities and their accessions:

1. **STUDY** with accessions in the form of SRP, ERP, or DRP
2. **SAMPLE** with accessions in the form of SRS, ERS, or DRS
3. **EXPERIMENT** with accessions in the form of SRX, ERX, or DRX
4. **RUN** with accessions in the form of SRR, ERR, or DRR

SRA Hierarchy & Accessions continued



Step 5: Send to Run Selector

- We will use **Run Selector** to obtain a comprehensive list for multiple samples and their replicates

Click on the samples you wish to process, then **Send to**, select the radio button for **Run Selector**, and then press **Go**.

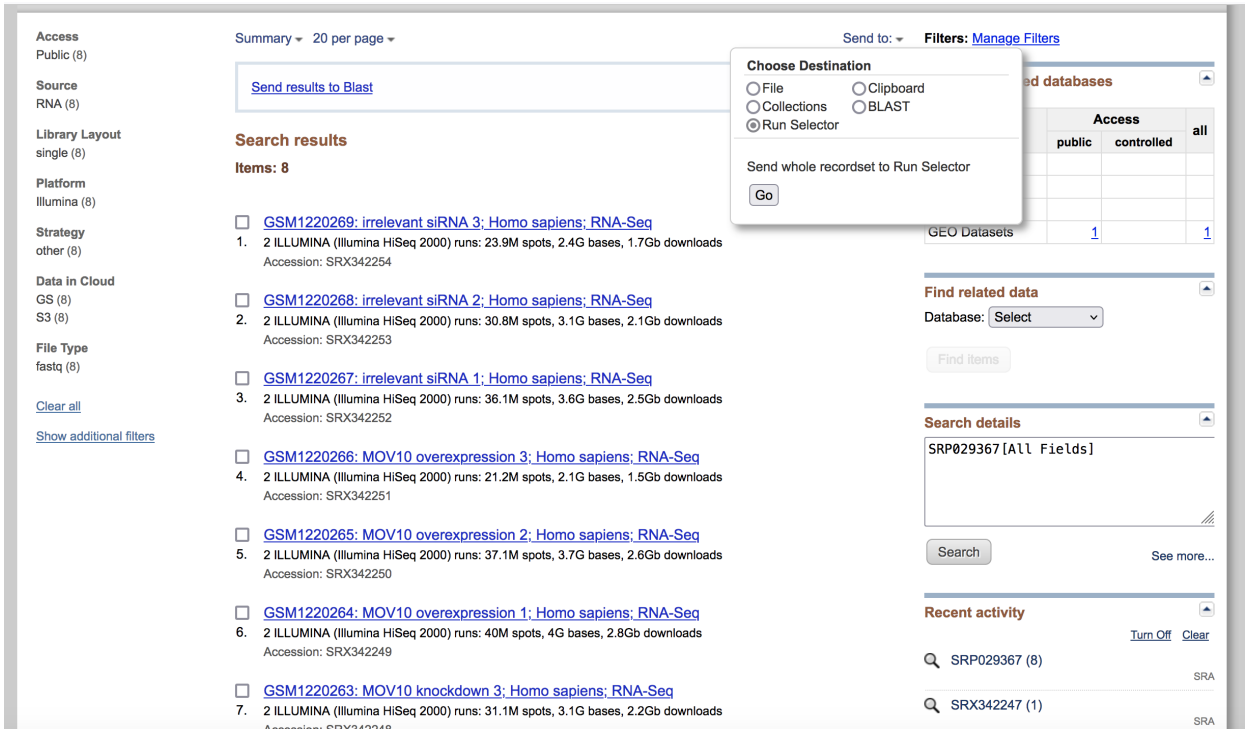


Figure 14: Run Selector

Run Selector overview

Run Selector will aggregate all the information for the study samples, giving information on:

- Library Layout - whether the reads were sequenced using single or paired end sequencing
- Platform - which sequencing technology was used
- Organism
- Instrument
- Cell type/ tissue type . . . and other useful information that should be noted for downstream analysis.

Common Fields

BioProject	PRJNA217781
Consent	PUBLIC
Assay Type	RNA-Seq
AvgSpotLen	100
Cell_Line	HEK293F
Cell_type	Human Embryonic Kidney cells
Center Name	GEO
DATASTORE filetype	FASTQ, SRA
DATASTORE provider	S3

Run Selector Summary

- Below this there is a Summary detailing the total number of runs in the study.
- We selected (5) samples. Why are there 10 listed?

Found 16 Items

	Run	BioSample	Bases	Bytes	Experiment	GEO_Accession	mov_expression	create_date
1	SRR960455	SAMN02340011	2.74 G	1.90 Gb	SRX342247	GSM1220262	low	2013-08-30 13:30:00Z
2	SRR960456	SAMN02340011	2.53 G	1.74 Gb	SRX342247	GSM1220262	low	2013-08-30 13:29:00Z
3	SRR960457	SAMN02340009	1.62 G	1.12 Gb	SRX342248	GSM1220263	low	2013-08-30 13:38:00Z
4	SRR960458	SAMN02340009	1.49 G	1.03 Gb	SRX342248	GSM1220263	low	2013-08-30 13:30:00Z
5	SRR960459	SAMN02340010	2.08 G	1.44 Gb	SRX342249	GSM1220264	high	2013-08-30 13:32:00Z
6	SRR960460	SAMN02340010	1.92 G	1.32 Gb	SRX342249	GSM1220264	high	2013-08-30 13:28:00Z
7	SRR960461	SAMN02340016	1.93 G	1.34 Gb	SRX342250	GSM1220265	high	2013-08-30 13:32:00Z
8	SRR960462	SAMN02340016	1.78 G	1.23 Gb	SRX342250	GSM1220265	high	2013-08-30 13:30:00Z
9	SRR960463	SAMN02340013	1.10 G	778.10 Mb	SRX342251	GSM1220266	high	2013-08-30 13:26:00Z
10	SRR960464	SAMN02340013	1.02 G	721.91 Mb	SRX342251	GSM1220266	high	2013-08-30 13:27:00Z
11	SRR960465	SAMN02340014	1.88 G	1.30 Gb	SRX342252	GSM1220267	normal	2013-08-30 13:26:00Z
12	SRR960466	SAMN02340014	1.73 G	1.20 Gb	SRX342252	GSM1220267	normal	2013-08-30 13:26:00Z

Interpreting Run Selector Summary

- Going back to the summary page for a sample: https://www.ncbi.nlm.nih.gov/sra?term=SRX342247 we can find more information.
- These samples were submitted for sequencing either twice or on two separate lanes.

Interpreting Run Selector Summary continued

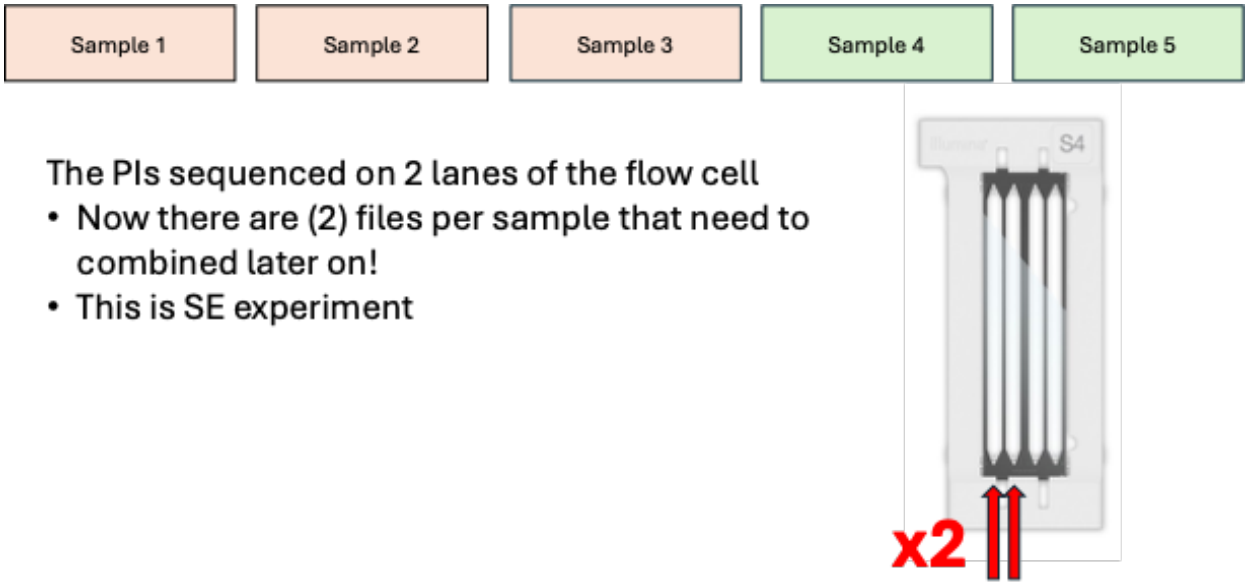
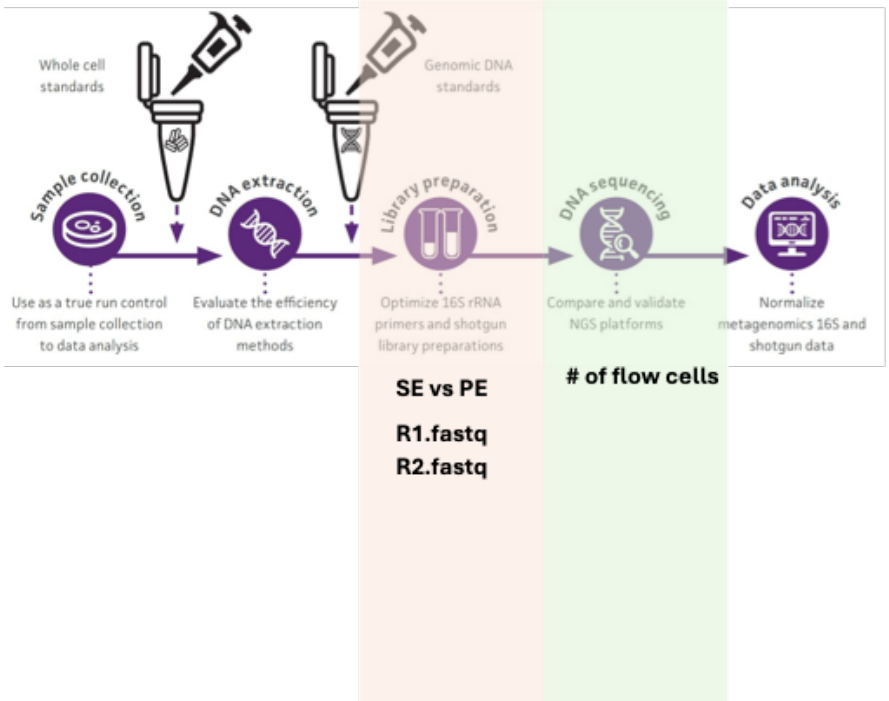


Figure 17: Understanding Run Selector

Interpreting Run Selector Summary continued

Therefore, for a single sample, there will be double the amount of sequencing files to process.



What does this mean for us?

Based on the decision made early in the workflow, this can result in more files/data to process

Figure 18: Run Selector Take-Away

Step 6: Download the Accession List in a text format

It is on this page that we can download the **Metadata** and **Accession List** in text format.

- The **Metadata** is a very useful text summary of all metadata for all runs in the study
- The **Accession List** is a list of all the SRR accession numbers for the study. We will need this list to download the FASTQ files using the script below.

SRA-Toolkit

- The SRA Toolkit is a set of utilities developed by NCBI for accessing data in the SRA.
- The toolkit provides command-line tools for downloading, manipulating, and converting sequencing data stored in the SRA format, making it easier for researchers to work with large-scale genomic data. It's widely used in bioinformatics and genomics research for tasks such as sequence alignment, quality control, and data analysis.

`fastq-dump`

`fastq-dump` is a command-line tool included in the SRA Toolkit. It's used to extract data from the SRA and convert it into the FASTQ format, which is a standard file format used to store biological sequences and their corresponding quality scores from high-throughput sequencing experiments.

Using fastq-dump

We would like to run the program fastq-dump to download fastq files. Let's type the following command:

```
fastq-dump --help
```

Does it work?

Using `fastq-dump` with the Environmental Module System

- If this does not work it means that this program `fastq-dump` is not available in your current environment.

However, a great work-around to downloading and configuring programs yourself is to first check if they are available as library packages through the Environmental Module System found within the VACC.

- Environmental Modules provide a convenient way for VACC users to load and unload packages. These packages are maintained and updated by the VACC.

Environmental Module System Specific Commands

The following commands are necessary to work with modules:

Module commands	description
`module avail`	List all available software modules
`module load`	Loads the named software module
`module list`	Lists all the currently loaded modules
`module unload`	Unload a specific module
`module purge`	Unload all loaded modules
`module help`	Displays general help/information about modules

Note: Before using software, we have to load the software. You will have to load the software every time you would like to use it.

Module load

`module load` command modifies your environment so that the path and other variables are set so that you can use a program such as gcc, matlab, or sratoolkit!

Type the following:

```
module load gcc/13.3.0-xp3epyt
module load sratoolkit/3.0.0-y2rspiu
```

Module List

Once a module for a tool is loaded, you have essentially made it directly available to you like any other basic shell command. We can check to see if its loaded using:

```
module list
```

```
Currently Loaded Modules:
  1) gmp/6.2.1-ip3t4a7    3) mpc/1.3.1-dv3gprk       5) z
  2) mpfr/4.2.1-344sqki   4) zlib-ng/2.1.6-ibq6yfi   6) g
```

Using fastq-dump

The SRAToolKit contains the program called fastq-dump. Now try:

```
fastq-dump  --help
```

```
Usage:
  fastq-dump [options] <path> [<path>...]
  fastq-dump [options] <accession>

INPUT
  -A|--accession <accession>        Replaces accession der
                                    filename(s) and deflin
                                    table dump)
  --table <table-name>              Table name within cSRA
                                    "SEQUENCE"
```

Overview so far

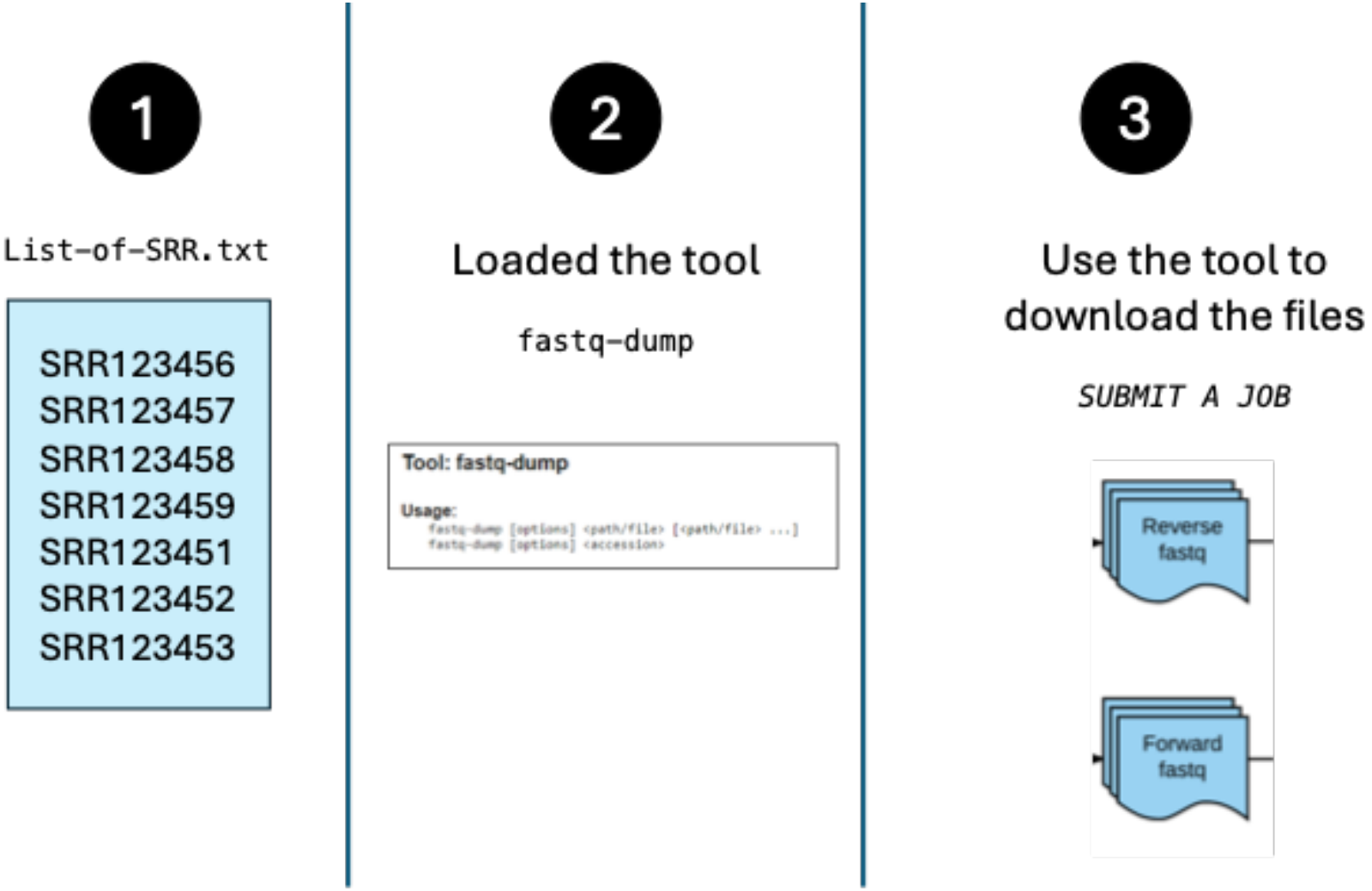


Figure 19: Overview

Why Submit a job?

- To download multiple **SRR FASTQ** files sometimes can take hours.
- Lucky for us, we have a script we can run.
- Lets discuss how to submit a script to be run using the SLURM batch system.

Two kinds of jobs

1) An **interactive job** entails directly entering a compute node and running programs from there. *This is what we have been doing.* This is useful for initial exploration but not serious computation as you will lose connection with the cluster.
2) A **batch job** is a computing job that you "send to the cluster". Instead of issuing commands manually, a batch job entails a sequence of commands specified in a submission script. **Once you submit a batch job, you do not need to continue being logged into the cluster for the job to finish running.**

Submitting a batch job using SLURM

- Submitting a job to an HPC machine is done using a workload manager called SLURM (Simple Linux Utility for Resource Management).
- SLURM handles job scheduling, resource allocation (nodes, processors, memory, GPUs), and job monitoring.
- Jobs can be put in queue and then run as resources become available.

The basic steps you will follow include:

1. Log into VACC
2. Write job script
3. Submit batch job
4. Monitor job and wait for it to run
5. Retrieve your output!

SLURM Directives

- At the top of the job script will always be several lines that start with `#SBATCH`.
- The SLURM directives provide the job setup information used by SLURM, including resources to request.
- This information is then followed by the commands to be executed in the script.

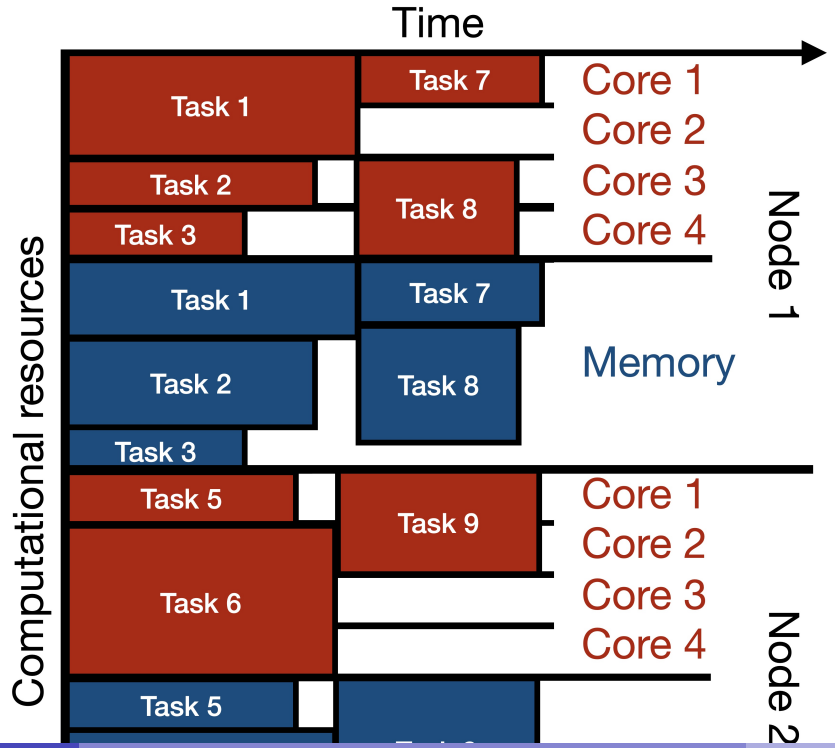
Resource requests

You need to request resources from the scheduler. Below are the kinds of resources that are typically requested.

- Time: The amount of time you expect your job (or each constituent task) to run, specified in hours, minutes, and seconds.
- Memory: The amount of memory you expect your job (or each constituent task) to use, often specified in gigabytes (GB).
- Number of nodes/tasks: A node is a single computational unit within a cluster. Nodes contain CPUs. An HPC can consist of a few hundred or thousands of nodes!

Job Scheduler

- The scheduler manages the allocation of resources to each job.
- It does so by maintaining a queue of jobs waiting to be executed.
- The scheduler allocates resources to each job in the queue based on their resource requests and the order in which they are submitted.



Class Exercise

Please download the `SRR_download` folder found in this location:

```
/gpfs1/cl/mmg3320/course_materials/SRR_download
```

Once downloaded open the file called `inner_script.sh`

Partition

A partition refers to a group of nodes which are characterized by their hardware. Specifying a partition is optional and if not specified the default partition is `general`. We will *mostly* specify `general`:

```
#SBATCH --partition=general
```

Other Partitions

Partition	Intended Use	Max Runtime
general	General computing – default partition	30 hours
short	General computing with short runtime	3 hours
week	General computing with longer runtime	7 days
nvgpu	NVIDIA GPU partition	48 hours

You can check partition usage using the following command:

```
sinfo -p partition_name
sinfo -p general
```

Walltime

Walltime is the maximum amount of time your job will run.
Your job may run for less time than you request, but it will not run for more time than you request.
Walltime is requested with #SBATCH –time=, where "dd" refers to day(s), "hh" to hour(s), "mm" to minute(s), and "ss" to second(s). You will replace each of these units with a two-digit numeral. Other acceptable formats are: mm, mm:ss, hh:mm:ss, dd-hh, dd-hh:mm, dd-hh:mm:ss.

```
# requesting 30 hours of walltime (hh:mm:ss)
#SBATCH --time=30:00:00
```

Nodes, Tasks, and Cores (CPUs)

The nodes, tasks, and core (CPU) resources you request depend on the type of job you are running.

- Node: A "node" is a server in the cluster. Each node is configured with a certain number of cores (CPUs).
- Task: A "task" is a process sent to a core. By default, 1 core is assigned per 1 task.
- Core/CPU: The terms "core" and "cpu" are used interchangeably in high-performance computing.

VACC recommend's that you begin with 1 node and 2 processes. As we move forward, we will change the number of nodes required for "bigger" jobs.

Estimated Memory Requirements for `fastq-dump`

Data Type	Memory(Approx.)
SE small dataset (~1GB)	4-8 GB
SE large dataset (~10GB)	8-16 GB
PE small dataset (~10-20GB)	8-16 GB
PE large dataset (~50GB)	16-32 GB+

Mail Type

In order to receive emails, you must set what types of emails you would like to receive, using the flag –mail-type. The options include: BEGIN (when your job begins), END (when your job ends), FAIL (if your job fails), ALL. For example:

```
#SBATCH --mail-type=END
```

JOB NAME

- Job name is used as part of the name of the job log files. It also appears in lists of queued and running jobs.
- Specifying a job name is not required. If you don't supply a job name, the job ID (supplied by Slurm) is used.

However, if you do wish to specify a job name, use the –job-name flag. For example, where your job name is "myjob":

```
# replace "myjob" with YOUR chosen job name
#SBATCH --job-name=myjob
```

Job Submission

Once your job script is written, you can submit it. To submit your job, use the `sbatch` command with your filename. For example, where the filename is "myfilename":

```
# replace "myfilename" with YOUR filename
sbatch myfilename
```

Job Submission continued

When you submit your job, Slurm will respond with the job ID. For example, where the job ID Slurm assigns is "123456," Slurm will respond:

```
Submitted batch job 123456
```

Its good to note your job ID!

Some commands used to interact with SLURM

Command	What It Does
sbatch	Submits a job, e.g., sbatch myjob
squeue	Checks status of all jobs in scheduling queue
squeue -u	Checks status of all jobs belonging to the named user, e.g., squeue -u usr1234
scancel	Deletes/cancels a particular job, e.g., scancel 123456

Script description in SRR_download/

The first script is a loop that will go through your list_of_SRR.txt, and calls a second script at each iteration, passing the `fastq-dump --gzip` command for each SRR number on the list.

sra_fqdump.sh

```
#while there are lines in the list of SRRs file
while read p
do
#call the bash script that does the fastq dump, passing
sbatch inner_script.sh $p
done <list_of_SRRs.txt
```

inner_script.sh: specific for SE data

```
#for single end reads only
fastq-dump --gzip $1
```

Paired end files

Paired end files need to be split at the download step. SRA toolkit has an option for this called "–split-files". By using this, one single SRR file will download as SRRxxx_1.fastq and SRRxxx_2.fastq.

Furthermore, there is a very helpful improvement on this function called "–split-3" which splits your SRR into 3 files: one for read 1, one for read 2, and one for any orphan reads (ie: reads that aren't present in both files). This is important for downstream analysis, as some aligners require your paired reads to be in sync (ie: present in each file at the same line number) and orphan reads can throw this order off. Change the inner_script.sh as follows if your reads are paired end:

```
fastq-dump --split-3  $1
```

Final Class Exercise

1. Run the `sra_fqdump.sh` script
2. If this ran successfully, you should see two new fastq files and emails in your inbox.

```
SRR25462427.fastq.gz
SRR25462429.fastq.gz
```

3. Check their sizes to see that SRR25462396 is 3.2MB and SRR25462427 is 4.6MB.

Bypassing storage issues with /scratch

- When downloading large datasets to the server, it's important to consider storage limits. If you download files to your home directory, the maximum storage allowed is ~100GB. This can become an issue when handling tens or hundreds of FASTQ files because SRA-Toolkit does not download FASTQ files directly.
- To avoid this, use the scratch space on the VACC (/scratch). This location has a much larger storage limit (12TB) and is better suited for handling large downloads.

Citation

This lesson was developed using materials from the Vermont Advanced Computing Center. These materials are freely available.