Before we get started...

1. Please go to ondemand.bil.psc.edu

2. Log in with your BIL account

3. Comment in the chat if you **<u>cannot</u>** log in

Data Exploration for Spatial Transcriptomics

Using BIL resources for data analysis and collaborative projects

This workshop will be recorded and shared internally.

Topic	Length (mins)	Presenter	Schedule (ET)
Storage ● Intro to BIL Analysis Ecosystem ● File Spaces and Storage ● Project Space ● Data Transfer ● *Hands-on Activity: Deposit Training Data/Models* ● *Discussion: Data Organization*	45	Mariah/Luke	1:00-1:45
Computational Resources ● BIL GPUs ● Bridges2 ● Neocortex ● Software/Modules ● Containers ● *Discussion: Desired Software*	15	Ivan	1:45-2:00
Break	*15*		*2:00-2:15*
Analysis Resources ● SLURM, Interact ● Workflow Systems ● Open OnDemand ● Jupyter Lab, RStudio ● *Hands-on Activity: Running Cellpose*	60	Ivan	2:15-3:15
Questions/Discussion	15		3:15-3:30

Github with all the slides and materials for the workshop can be found on the BIL GitHub:

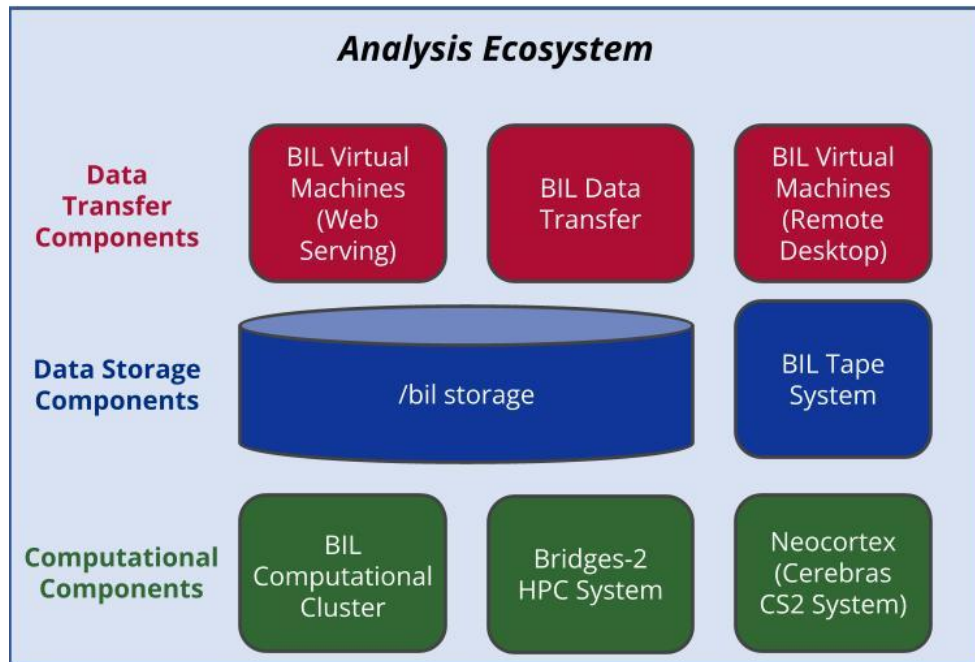https://github.com/brain-image-library/workshops/tree/master/2024/january

Overview of the BIL Analysis Ecosystem

Computational Resources

- BIL Analysis Ecosystem
  - Flexible resource that is made up of several large memory machines equipped with modern GPUs
- Bridges-2
  - NSF funded supercomputer located at the Pittsburgh Supercomputing Center.
- Neocortex
  - Neocortex is a resource that targets AI-powered scientific discovery and provides hardware for the development of efficient algorithms for artificial intelligence and graph analytics.

Tools available to access some of these resources:

- Open OnDemand
  - Open OnDemand is an open-source portal that enables web-based access to HPC services
   - Jupyter Lab
    - RStudio
    - File System Interface

Analysis Ecosystem

Data Transfer Components: BIL Virtual Machines (Web Serving), BIL Data Transfer, BIL Virtual Machines (Remote Desktop)

Data Storage Components: /bil storage, BIL Tape System

Computational Components: BIL Computational Cluster, Bridges-2 HPC System, Neocortex (Cerebras CS2 System)

BIL File System File Spaces and Storage

Directory	Description	Storage and Deletion Policy
/bil/users/username	User home directory. Location for any scripts, source code, or parameter files. Your home directory is visible on all compute nodes.	RAID redundant against drive failures. Not intended for long-term storage. Files are not backed up to tape.
/bil/lz/username	Landing zone where new submission data files are stored until they are ready for validation.	RAID redundant against drive failures. Not intended for long-term storage. Files are not backed up to tape.
/bil/data	Public dataset files. Files are organized by collection number. For example, submission id: 1234abcd would be found at /bil/data/12/34/1234abcd.	RAID redundant against drive failures. Permanent storage for public data. Backed up to tape. Users cannot modify files in /bil/data.
/bil/proj	Custom project directories that are created by special request.	RAID redundant against drive failures. Intended to enable data sharing and project work, not long-term storage. Files are not backed up to tape.
/scratch	Scratch storage for jobs via slurm on BIL Analysis Ecosystem. Visible on all compute nodes. Not intended to be long-term storage.	50 TB of storage shared across all users. *Files will be deleted after 14 days.*
/local	Space on each node for a specific job and can be used as a working space for a job while it is running.	7TB of storage shared across all users and cleared after each job has finished running

Group Project Space:
/bil/proj/bicanstwg

- Shared space for multiple (predefined) users to access and share data - not public
- Permissions and access-control lists (ACLs) for files are are important to keep track of
- Data is not backed up in the project space.
- Not intended for long term storage
- Accessible to all BIL computational resources
- Data organization within the project space is not monitored by the BIL team
    - *Coming Later: Discussion on how to organize your groups data*
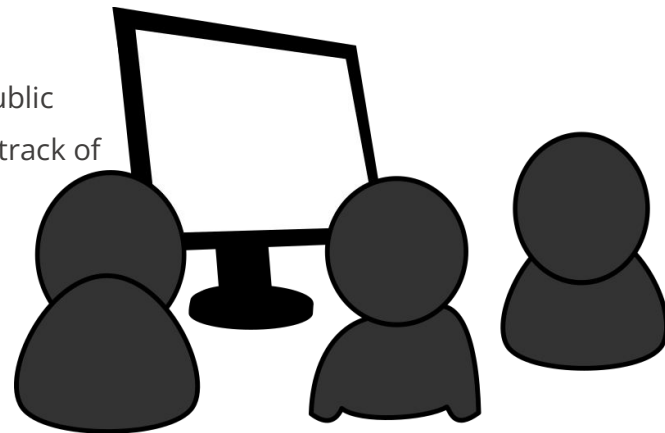
Data Transfer Overview

Several options are available transferring files depending on your operating system

Windows?

1. SFTP/SCP
    - WinSCP, FileZilla (User Interface applications)
2. Globus

MacOS X?

1. SFTP/SCP
    - FileZilla, Fetch
2. Rsync (Command Line)
3. Globus

Linux?

1. SFTP/SCP (Interfaces available depending on Linux Distribution)
2. Rsync (Command Line)
3. Globus

Node Details

Node	Host	VM	Description
login	login.brainimagelibrary.org	vm004	The BIL login node provides command line (ssh terminal) access to BIL data. Everyone who requests a BIL account has access to the login node using their BIL Username and Password. The login node is not intended for heavy computation or visualization, but rather to provide convenient shell access to BIL data and enable interactive and batch use of the BIL Computational Cluster. The BIL computational cluster provides a suitable resource for both computation and visualization.
data transfer	upload.brainimagelibrary.org	vm018	Due to size, image data can not be uploaded through the submission portal (submit.brainimagelibrary.org). It must be uploaded separately through the BIL data transfer nodes, which are available at the virtual host: upload.brainimagelibrary.org.

Node Policy

There should be no computation run on the login node (login.brainimagelibrary.org) or data transfer node (upload.brainimagelibrary.org). Any jobs run on these nodes will be terminated.

Data Transfer

Create a directory for yourself in the project space

1. Log into the BIL machine
   a. **ssh username@login.brainimagelibrary.org**
2. Navigate to the project directory
   a. **cd /bil/proj/bicanstwg**
3. Create a directory to point your data upload to
   a. **mkdir *your username***

Initiate data transfer

1. Use **rsync** to transfer your folder up to BIL
   a. rsync -lrtpDvP mouse1 username@upload.brainimagelibrary.org:/bil/proj/bicanstwg/**username**/
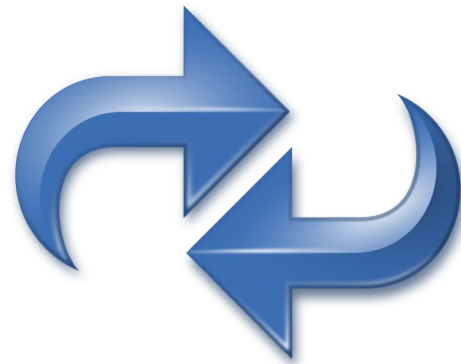
Setting Permissions and ACLs

Files and directories on the BIL file systems are protected with UNIX Permissions. You can see the permissions on a file with the `ls -l` command. For example:

```
$ ls -l
total 1
-rw-r--r-- 1 ropelews pscstaff 3 Jan  5 17:24 file.txt
```

The permission string is shown in the first column, the owner of the file in the third column and the group in the fourth column.

The permissions for any file/directory will look something like: **drwxrwxrwx**
- The purple triplet controls the permissions for the owner/user (u) of the file.
- The orange triplet controls the permissions for the group (g) of the file
- The green triplet controls the permissions for others (o) which is any user on the system.
- If permission is granted to a file, you will see a code, otherwise a dash
  - "r" = read, "w" = write, "x" = execute, '-' = no permission.
- A leading "d" means that the file is a directory. Note that directories must have execute permission to look up the contents.

Use the chmod command to change permissions of files. The "+" sign grants permission, the '-' sign removes permission. For example to give the group read and execute permissions on the file above, one would use the command:

```
$ chmod g+rx file.txt
```

All members of this workshop are in the bicanstwg group. The chgrp command can change the file group to bicanstwg:

```
$ chgrp bicanstwg file.txt
```

Use the `man chmod` command and `man chgrp` to find out more information about these commands.

Setting Permissions and ACLs

Access Control Lists (ACLs) provide finer grain access to files and directories than UNIX permissions. You can see if an ACL has been set on a file or directory with the `ls -l` command. For example:

```
$ ls -ld /bil/lz/ropelews
drwxrwx---+ 46 bil bil 15872 Aug  8 12:52 /bil/lz/ropelews
```

Note that the permission string now contains a "+" at the end, to indicate that a ACL has been set on the directory.

To find out the ACLs that have been set on a file or directory use the getfacl command. For example to see if an acl has been set on the file file.txt:

```
$ getfacl file.txt
```

To set an ACL use the setfacl command. The general format for an ACL entry to grant a specific user permission is u:username:permissions. For example, to grant the specific user awatson read and execute permission on the file file.txt one would use the following command:

```
$ setfacl -m u:awatson:rx file.txt
```

Use the `man getfacl` command and `man setfacl` to find out more information about these commands.

Organizing Data

- You can organize data in /bil/proj in any way that works for you
- This does not need to be the same organization that you will use for data submission/publication
- During previous BICAN STWG Meetings the following data organization structures were proposed:
  - Organizing by species
  - Organizing by acquisition technique

Discussion: How to organize your data for model testing

Reach out with questions!
bil-support@psc.edu

If you ever have a question about the BIL Analysis Ecosystem please reach out!

Bridges-2

https://www.psc.edu/resources/bridges-2/

Bridges-2 is available at no cost for research and education, and at cost-recovery rates for other purposes.

Regular memory nodes: 256GB - 512GB RAM (488 RM nodes have 256GB of RAM, and 16 have 512GB of RAM)

Extreme memory nodes: 4TB RAM (4 nodes)

GPU nodes: 512GB of RAM (Eight NVIDIA Tesla V100-32GB SXM2 GPUs) (24 GPU nodes)

The BIL file system is mounted on Bridges 2 so all data is directly accessible on these resources.

Bridges-2 is allocated through ACCESS. More information on ACCESS allocations and information on how to get started is available here: https://allocations.access-ci.org/

BIL Analysis Ecosystem Computational Hardware

Large Memory Compute Nodes (L nodes)

- Number of nodes: 8 (l001 - l008)
- Processors: HPE ProLiant DL580 Gen9
- Cores per node: 80 - 20 per CPU
- RAM: 3TB
- CPU per node: 4
- CPU model: Xeon E7 8870

GPU

- Number of GPUs: 8
- Model: NVIDIA A100
- Memory per GPU: 80GB
- The GPU node is meant only for GPU computing
- Access to the GPU is controlled. To gain access, send an email to bil-support@psc.edu

Node Policy

There should be no computation run on the login node (login.brainimagelibrary.org) or data transfer node (upload.brainimagelibrary.org). Any jobs run on these nodes will be terminated.

Wall Time Limits

- Compute L nodes - 2 day limit (2:00:00:00)
- GPU - 4 day limit (4:00:00:00)