Introduction to Minerva

Minerva Scientific Computing Environment

https://labs.icahn.mssm.edu/minervalab

Patricia Kovatch
Eugene Fluder, PhD
Hyung Min Cho, PhD
Lili Gai*, PhD
Dansha Jiang, PhD
Wayne Westerhold, MS

September 16, 2020

Mount Sinai

Outlines

- Minerva compute and storage resources
- Minerva account and logging in
- User software environment
- Services on file transfer, data archive, web server and Rstudio-Connect server
- Preview on job scheduler LSF (Load Sharing Facility)

Minerva cluster @ Mount Sinai

Chimera Computes:

- 3x **login nodes** - Intel 8168 24C, 2.7GHz - **384 GB** memory
- 274 **compute nodes** - Intel 8168 24C, 2.7GHz
  - 13,152 cores (48 per node) - **192 GB**/node)
- 4x **high memory nodes** - Intel 8168 24C, 2.7GHz - **1.5 TB** mem
- Adding 30 **high memory** nodes - Intel 8268, 2.9 GHz- 1.5 TB mem
- 48 **V100 GPUs** in 12 nodes -
  Intel 6142, 2.6GHz - 384 GB memory - 4x V100-**16 GB** GPU
- Adding 32 **A100 GPUs** in 8 nodes -
  Intel 8268, 2.9 GHz - 384 GB memory - 4x A100- **40 GB** GPU

BODE2 Computes:

- $2M S10 BODE2 awarded by NIH (Kovatch PI)
- 78 compute nodes - Intel 8268, **2.9 GHz**
  - 3,744 cores (48 cores per node - **192 GB**/node)
  - **Open to all NIH funded projects**

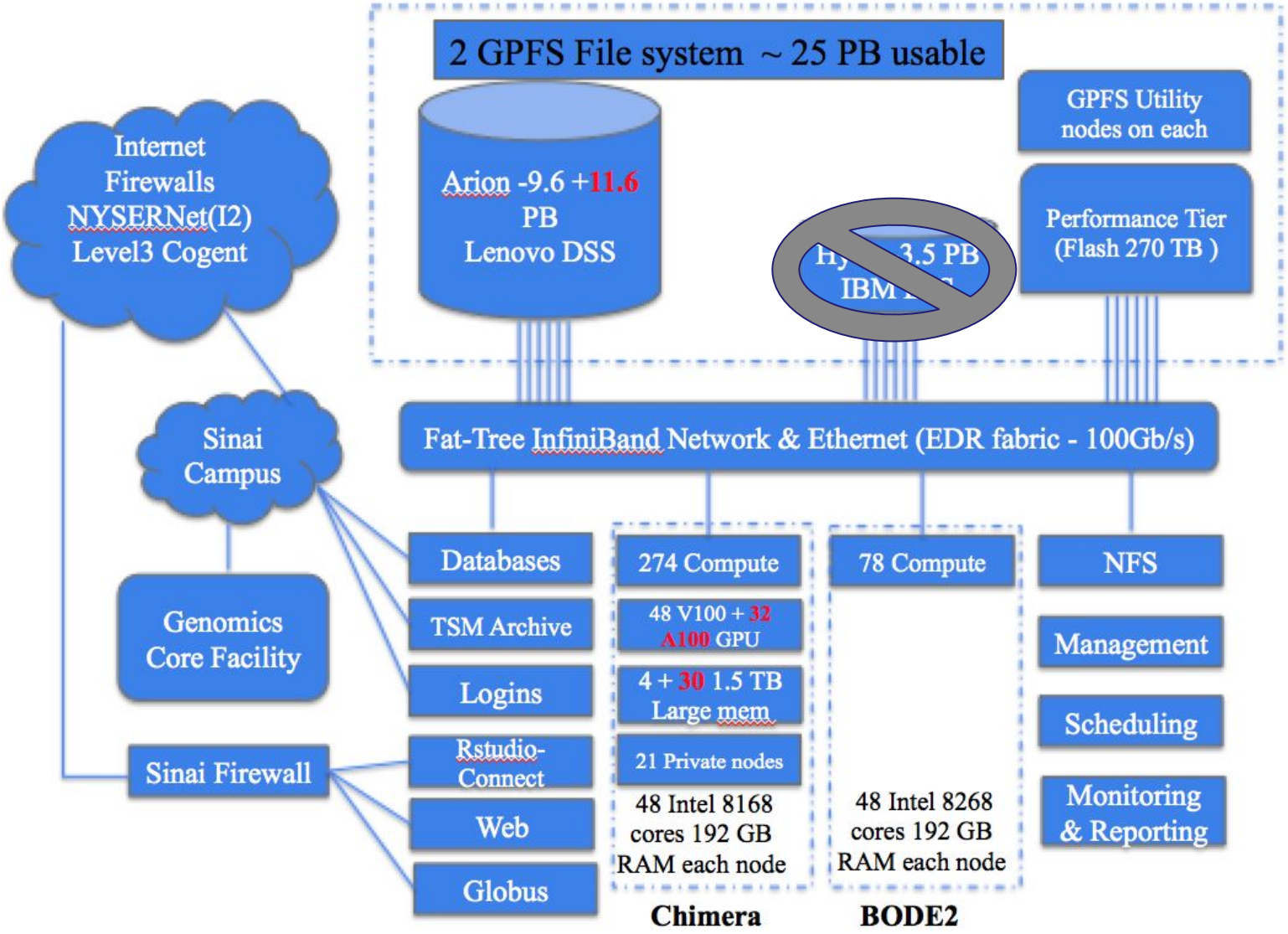
Storage:

Total amount of storage space: ~13+ **12** PB usable in total

- **/sc/arion** : primary storage
  - Use the system path environment variable in scripts
- **/sc/hydra** will be out of warantty end of 2020 - Being migrated to /sc/arion

Minerva cluster @ Mount Sinai

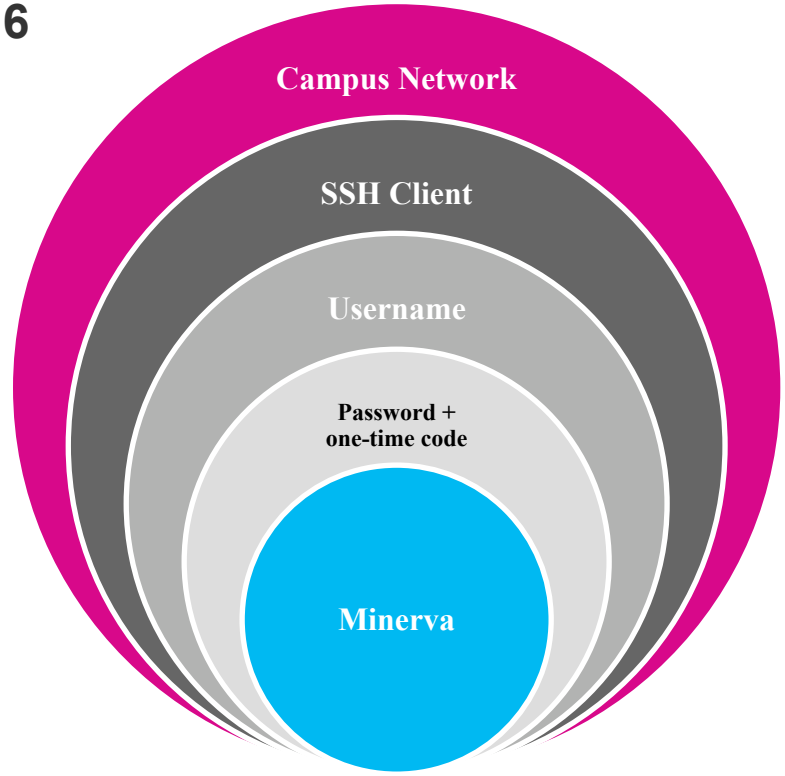
- Minerva nodes and infrastructure fully refreshed 2019 and expanded 2020
- ~20,000 compute cores & ~25PB of high speed storage



Logging in - General

Minerva is a Linux machine with Centos 7.6

- Linux is command line based, not GUI
- Logging in requires campus network,
  SSH client installed on your machine,
  username,
  memorized password,
  and **one-time use code**
  obtained from a physical/software VIP token



Logging in - Detailed procedures

Detailed procedures:

- SSH client: terminal (Mac), Putty or MobaXterm (Windows)
- Campus network (School VPN needed if out of campus)
- Apply for an account at https://acctreq.hpc.mssm.edu/
  - Apply account for external users following here
- Register your token at the Self Service Portal **school site** (https://register4vip.**mssm**.edu/vipssp/)
- Logging info at https://labs.icahn.mssm.edu/minervalab/logging-in/

Note: Minerva is school resource, so use your school password and school portal for register

Logging in - Linux / Mac

Connect to Minerva via ssh:

- ssh *your_userID*@minerva.hpc.mssm.edu
- To display graphics remotely on your screen, pass the “-X” or “-Y” flag:
  - ssh -X *your_userID*@minerva.hpc.mssm.edu
  - Mac: Install XQuartz on your mac first
- Open a terminal window on your workstation(Linux/Mac)
  - Landed on one of the login nodes, and at your home directory
    - Never run jobs on login nodes
    - For file management, coding, compilation, etc., purposes only

```
imac:~ gail01$ ssh -X gail01@minerva.hpc.mssm.edu
Please input your password and two factor token:
Password:
Last login: Fri Sep 11 16:19:23 2020 from li03c03.chimera.hpc.mssm.edu
================================================================
2020 Fall Training Sessions:
Introduction to Minerva
Wednesday, 16 Sep. 2020 2:00-3:00PM
Zoom link: https://mssm.zoom.us/j/93191131895
LSF job scheduler
Wednesday, 23 Sep. 2020 2:00-3:00PM
Zoom link: https://mssm.zoom.us/j/92723688624
   === Send ticket to hpchelp@hpc.mssm.edu ===
================================================================
gail01@li03c04: ~ $ pwd
/hpc/users/gail01
gail01@li03c04: ~ $
```

Logging in - Windows

- **Install MobaXterm from** https://mobaxterm.mobatek.net/
  - Enhanced terminal for Windows with **X11 server**, tabbed SSH client, network tools and much more

OR

- **Install PuTTY from www.putty.org**
  - Google it. It will be the first hit.
  - https://www.youtube.com/watch?v=ma6Ln30iP08

- **If you are going to be using GUI's**
  - In Putty: Connection > SSH > X11
    - Ensure "Enable X11 forwarding" is selected
  - On Windows box install Xming
    - Google; Download; Follow bouncing ball
  - Test by logging into Minerva and run the command: xclock
    - Should see a clock

Logging in - login nodes

3 login nodes: minerva[12-14], which points to the login node **li03c[02-04]**

- only available within campus-network

Users	Login method	Login servers	Password Components
Sinai users	userID	**@minerva.hpc.mssm.edu or specific nodes: @minerva12.hpc.mssm.edu @minerva13.hpc.mssm.edu @minerva14.hpc.mssm.edu**	Sinai Password + 6 Digit Symantec VIP token code
External users*	userID		Sinai Password + 6 Digit Symantec VIP token code

*the old way of authenticating for external users with yldap and yubikey will be retired in future

Note: Load balancer **Round-robin** is configured for **minerva.hpc.mssm.edu.** It will distribute client connections to the nearest across a group of login nodes.

Logging in - Config file

- **~/.ssh/config at your local workstation**
  - Set ControlMaster to reuse ssh connection for all hosts
  - Enable X11 forwarding
  - Set alias for hostname, so just type ssh minerva for login.

```
$ cat ~/.ssh/config
Host *
        ControlMaster auto
        ControlPath /tmp/ssh_mux_%h_%p_%r
        ControlPersist 24h
        PreferredAuthentications        keyboard-interactive
        ServerAliveInterval 240
        ServerAliveCountMax 2
        ForwardX11 yes
        ForwardX11Timeout     12h

Host minerva
        Hostname minerva.hpc.mssm.edu
        User gail01
```

Minerva Storage

- Storage is in folders and subfolders. In linux, subfolders are separated by “/”
- 4-ish folders you can have (Possibly multiple project folders)

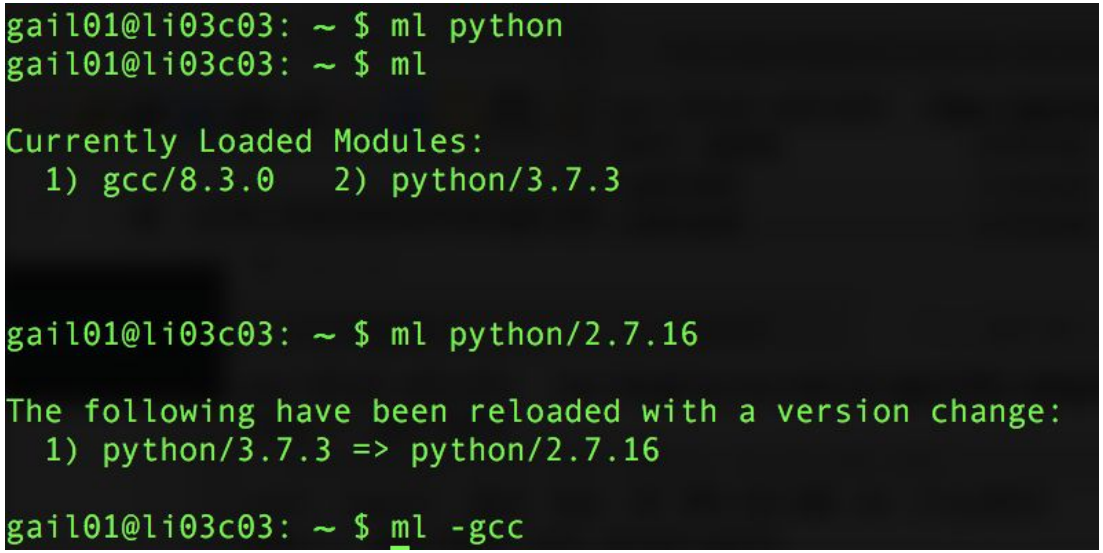
Home	/hpc/users/<userid> $ **quota -s**	• 20GB quota. • Slow. Use for “config” files, executables…NOT DATA • NOT purged and is backed up
Work	/sc/arion/work/<userid> **$df -h** /sc/arion/work<userid>	• 100GB quota • Fast, keep your personal data here • NOT purged but is NOT backed up
Scratch	/sc/arion/scratch/<userid> **$df -h** /sc/arion/scratch/<userid>	• Free for all, shared by all • Current size is about 100TB • Purge every 14 days and limit per user is 10TB
Project	/sc/arion/projects/<projectid> **$df -h** /sc/arion/projects/<projectid>	• PI’s can request project storage • Need to submit an allocation request and get approval from allocation committee https://labs.icahn.mssm.edu/minervalab/event/allocation-renewal-period-20-21/ • Not backed up • Incurs charges **$100**/TiB/yr

User Software Environment: Lmod

> 1000 modules, and different versions are supported on Minerva

Lmod Software Environment Module system implemented:

- Search for module: **$module avail** or **$ module spider**
  Check all available R versions *$ ml spider R*

  …….R/3.3.1, R/3.4.0-beta, R/3.4.0, R/3.4.1, R/3.4.3_p, R/3.4.3, R/3.5.0, R/3.5.1_p, R/3.5.1, R/3.5.2, R/3.5.3
- To check the detailed PATH setting in module files: **$ml show R**
- Load module: **$ml python ; unload module $ml -gcc**
- 

- Unload all modules: **$ml purge**
- Autocompletion with tab

User Software Environment: Lmod

Example

> ml python bedtools gnuplot fftw

> ml R python/2.7.16 -fftw

> ml R/4.0.2

> ml

Currently Loaded Modules:

1) zlib/1.2.8 4) gcc/8.3.0 7) gsl/2.5 10) R/4.0.2

2) bedtools/2.29.0 5) intel/parallel_studio_xe_2019 8) libpng/12 11) python/2.7.16

3) gnuplot/5.2.6 6) hdf5/1.10.5 9) java/1.8.0_211

> ml save myenv1

ml save: Lmod provides a simple way to store the currently loaded modules and restore them later through named collections

> **ml purge**

> ml

No modules loaded

> ml restore myenv1

> ml savelist

Named collection list :

1) default 2) myenv1

> ml disable myenv1

More at Lmod user guide

User Software Environment - Major packages

OS: **Centos 7.6** with **glibc-2.17(GNU C library) available**

GCC: system default /usr/bin/gcc is gcc 4.8.5

$ module load gcc (default is 8.3.0) or *$ ml gcc*

$ ml python Python: default version 3.7.3

(it will load python and all available python packages)

Note: python2 or python3

$ ml R R: default version 3.5.3

(it will load R and all available R packages)

$ml CPAN Collection of system Perl software: default system version 5.16.3

$ml anaconda3 Anaconda3: default version 2018-12

$ml java java: default version 1.8.0_211

SAS access: *$ml SAS*

- The cost for the license is **$150.00** per activation, and request form at https://mountsinai.formstack.com/forms/sas_software_distribution_form_1

Matlab access: *$ml matlab*

  - The cost for the license is **$100.00** per activation, and request form at https://mountsinai.formstack.com/forms/mathworksacademiclicense

User Software Environment - Anaconda Distribution

Anaconda3:

- Support minimal conda environments (such as tensorflow, pytorch, qiime)

  e.g., tensorflow (both in CPU and GPU)

  To avoid incompatibilities with other python, clear your environment with module purge before loading Anaconda

  $ml purge

  $ml anaconda3 (or anaconda2)

  $ml cuda

  $source activate tfGPU

- User should install their own envs locally,
  - ➔ Use option -p PATH, --prefix PATH Full path to environment location (i.e. prefix).

    $conda create python=3.x -p /sc/arion/work/gail01/conda/envs/myenv
  - ➔ Set envs_dirs and pkgs_dirs in .condarc file, specify directories in which environments and packages are located

    $conda create -n myenv python=3.x

- Set conda base auto-activation false

  conda config --set auto_activate_base false

```
$ cat ~/.condarc file
envs_dirs:
- /sc/arion/work/gail01/conda/envs
pkgs_dirs:
- /sc/arion/work/gail01/conda/pkgs
conda config --set auto_activate_base false
```

More at Conda config guide

User Software - Singularity Container

Singularity tool is supported, instead of docker (Security concern)

- Docker gives superuser privilege, thus is better at applications on VM or cloud infrastructure

To use singularity: *$ module load singularity*

To pull a singularity image:

$ singularity pull --name hello.simg shub://vsoch/hello-world

$singularity build --sandbox lolcow/ shub://GodloveD/lolcow (create container within a writable directory)

To pull a docker image:

$singularity pull docker://ubuntu:latest

To run a singularity image:

$ singularity run hello.simg *# or,* *$ ./hello.simg*

Note: /tmp and user home directory is automatically mounted into the singularity image. If you would like to **get a shell with arion mounted** in the image, use command:

$ singularity run -B /sc/arion/project/xxx hello.simg

To build a new image from recipe files: use Singularity Hub or your local workstation

- Singularity build is not fully supported due to the sudo privileges for users
- After registering an account on Singularity Hub, you can pull or upload your recipe, trigger the singularity build and download the image after built.
- Convert docker recipe files to singularity recipe files:

$ml python

$spython recipe Dockerfile Singularity

User Software Environment - some config

- You can load modules in your **.bashrc** script to load them on startup
- You can create your own modules and modify MODULEPATH so they can be found by

  module use /hpc/users/fludee01/mymodules

  or

  export MODULEPATH=/hpc/users/fludee01/mymodules:$MODULEPATH

- You can set PATH or PYTHONPATH by

  export PATH=/hpc/users/gail01/.local/bin:$PATH

  export PYTHONPATH=/hpc/users/gail01/.local/lib/python3.7/site-packages:$PYTHONPATH

File Transfer - Globus (Preferred)

- Globus is developed/maintained at the University of Chicago and used extensively at supercomputer centers
- Globus makes it easy to move/sync/share large amounts of data.
- Globus will retry failures, recover from faults automatically when possible, and report the status of your data transfer. Globus website 

Globus on Minerva

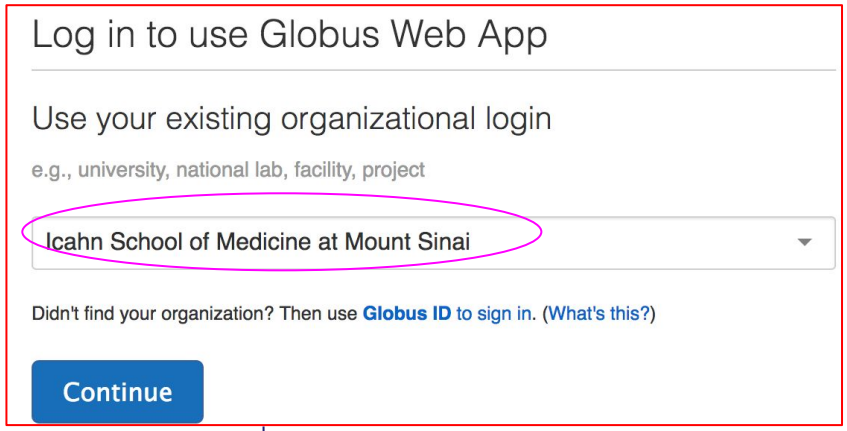
- Globus login bypasses the VPN (minerva11.hpc.mssm.edu)
- Two endpoints needed
  - **Minerva Endpoint: mssm#minerva**
  - **Globus Connect Personal to make your laptop an endpoint**
  - More information at https://docs.globus.org/how-to/get-started/ and https://docs.globus.org/how-to/share-files/

- *Minerva will be under HIPAA+BAA subscription, making sharing more easily*
  - *Be able to share data with anyone using their identity or their email address, even if they don't have accounts on the endpoint where the data is stored*
  - *Can upgrade your Globus account to Plus*
    - *You can share data from a Globus Connect Personal endpoint*
    - *two Globus Connect Personal endpoints*

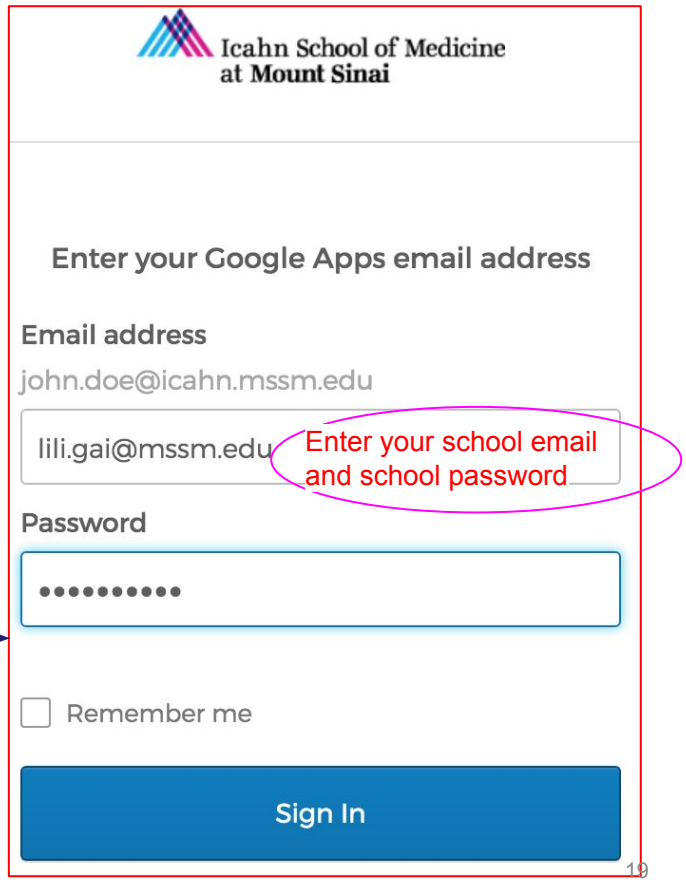
Globus - Logging

1. If you have email **@mountsinai.org**, please click "**use Globus ID to sign in"----> "Sign Up"** or **continue with your google account**

2. If you have email **@mssm.edu/@icahn.mssm.edu**, you can use this school email address and corresponding password to access Globus. Go to https://www.globus.org/ and click on Log In



You will be taken to a familiar-looking page for sinai login.



Globus - Install/use Globus Connect Personal on your desktop

- Follow these instructions to download Globus Connect Personal and set up an endpoint and collection on your own Mac, Linux, or Windows system.
- Navigate to the File Manager page and request a transfer between your new collection and **mssm#minerva**.
- Navigate to the Activity page and review your transfers.
- Transfers to and from your system won't fail if you shut your system down or disconnect from the network. Instead, transfers will be suspended and they'll automatically resume when your system comes back online.

File Transfer - Con't

- **SCP, SFTP**
  - Good for relatively small files, not hundreds of TB's
  - *Some scp apps for Windows/Mac use cached password. This feature must be turned off.*
  - *Note when you use VPN, data transfer between Minerva and your local computer may be pretty slow because the bandwidth is limited by school IT*

- **On Minerva**
  - After login to Minerva, ssh *li03c01* for data transfer, no time limit
  - minerva12 no time limit, minerva13/14 (33h) or interactive nodes (12h).
  - Please use a screen session so that you can return to your work after the drop of the connection.

Archiving Data: IBM Tivoli Storage Management (TSM)

- Keep for 6 years with two copies
- Can be accessed via either a GUI or the command line

```
$ module load java
$ dsmj -se=userid
```

or

```
$ dsmc -se= userid
```

- Large transfers can take a while. Use a *screen* session and disconnect to prevent time-outs
- Full more details at https://labs.icahn.mssm.edu/minervalab/archiving-data/
- Collaboration account:
  - If your group is in need of a collaboration account for group related tasks like archiving a project directory or managing group website, contact us at hpchelp@mssm.edu. For more info. see https://labs.icahn.mssm.edu/minervalab/collaboration-account/

Web server https://users.hpc.mssm.edu/

Find the link to your website at "**https://users.hpc.mssm.edu**" or go directly to **https://userid.u.hpc.mssm.edu**

Step 1: If this folder does not exist in your home directory, you should create it. $ mkdir ~/www

Step 2: 1) Place content in the www folder. $ cat > ~/www/index.html <<EOF

Hello World from my website.

EOF

2) put files or create symlink (from arion) under the ~/www

The *indexes* option is turned off by default for security reasons. You will an see error message "Forbidden, You don't have permission to access this resource." if you don't have an **index.html** or **index.php** file under the folder.

You can enable this option in the **htaccess file** in order to list your files, for example:

```
[gail01@li03c03 ~]# cat /hpc/users/gail01/www/.htaccess
Options +Indexes
```

Authentication

If you use your website for file sharing, we strongly recommend you to set up password protection for your files. Please refer to the "Authentication" part of the instructions, located here: https://labs.icahn.mssm.edu/minervalab/web-services/

WARNING WARNING WARNING: Be careful! Content, executables, scripts, symlinks, applications, etc. within the www/ folder may be (or are) publicly accessible. Scripts and applications launched via Apache in that folder run as your user! They can access any data (including your groups' /project data), delete data, archive data, submit jobs, cancel jobs, email people, etc., as your user. **You are responsible for any actions taken on your behalf!**

Web server https://users.hpc.mssm.edu/

Some demos on setting up your first python flask and dash app

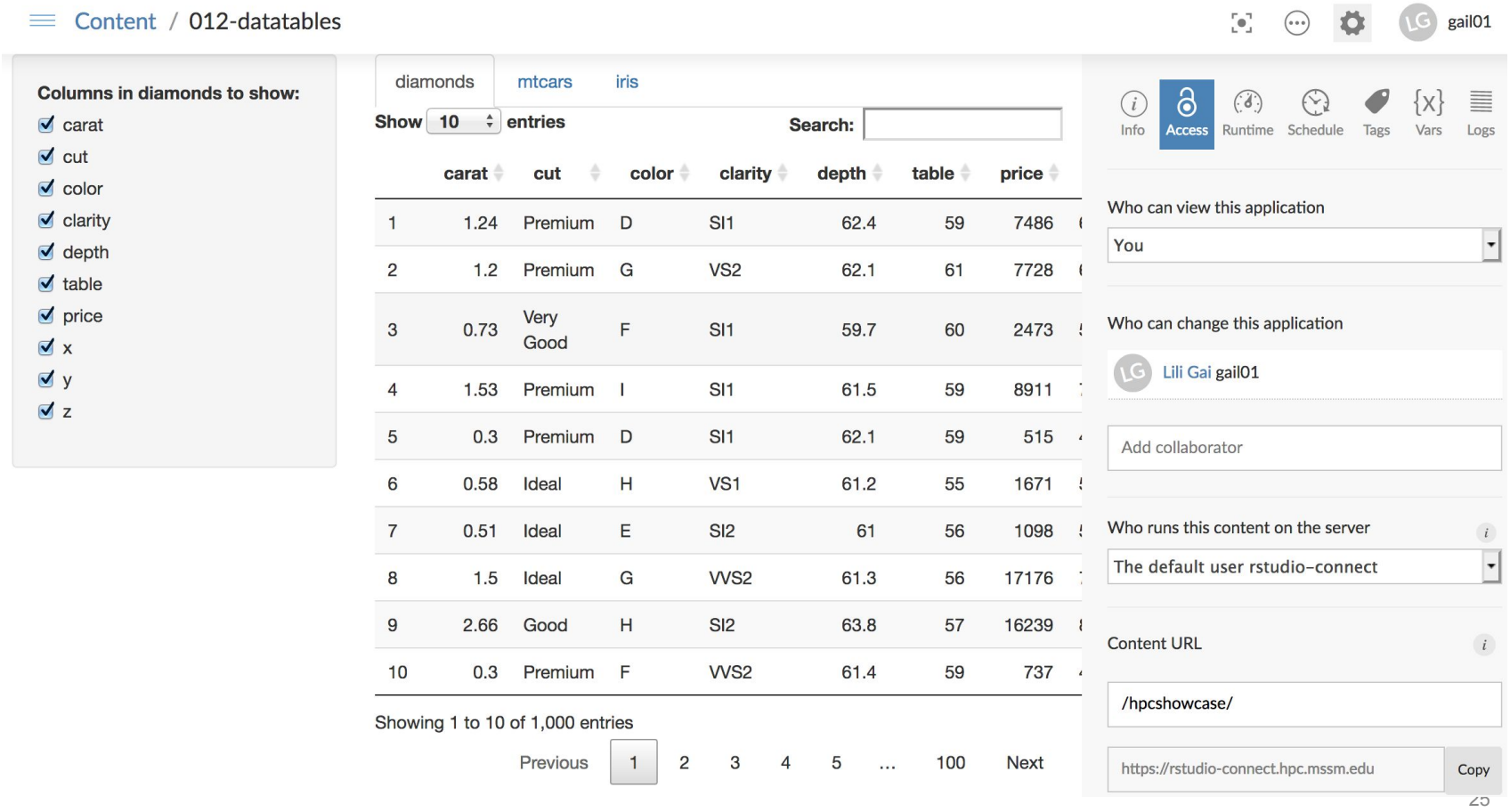
https://gail01.u.hpc.mssm.edu/flask_demo/

https://gail01.u.hpc.mssm.edu/dash_demo/

Code is at https://gail01.u.hpc.mssm.edu/code/

Rstudio Connect server https://rstudio-connect.hpc.mssm.edu

- You can publish Shiny, R Markdown for collaborators or others (More details at https://rstudio.com/products/connect/)
- If interested in publishing on Rstudio-connect, contact us at hpchelp@hpc.mssm.edu



HIPAA

- Minerva will be HIPAA compliant on October 1st, 2020, i.e., Protected Health Information (PHI) data will be allowed to be stored and processed on Minerva.

- All users have to read the HIPAA policy and complete Minerva HIPAA Agreement Form at https://labs.icahn.mssm.edu/minervalab/hipaa/ by October 1, 2020.

- Any user who has not signed the agreement by September 30, 2020, will have their accounts locked until the agreement is signed.

Load Sharing Facility(LSF)

A Distributed Resource Management System

LSF: batch job submission examples with bsub

Interactive session:

 # interactive session

$ bsub -P acc_hpcstaff -q interactive -n 1 -W 00:10 -Is /bin/bash

 # interactive GPU nodes, flag "-R v100" is required

$ bsub -P acc_hpcstaff -q interactive -n 1 -R v100 -R rusage[ngpus_excl_p=1] -W 01:00 -Is /bin/bash

Batch jobs submission:

 # simple standard job submission

$ bsub -P acc_hpcstaff -q premium-n 1 -W 00:10 echo "Hello World"

 # GPU job submission if you don't mind the GPU card model

$ bsub -P acc_hpcstaff -q gpu -n 1 -R rusage[ngpus_excl_p=1] -W 00:10 echo "Hello World"

*# himem job submission, flag "**-R himem**" is required*

*$ bsub -P acc_hpcstaff -q **premium** -n 1 -R himem -W 00:10 echo "Hello World"*

LSF: batch job submission using a script

$ cat star.lsf

```
#!/bin/bash
#BSUB -J mySTARjob                  # Job name
#BSUB -P acc_hpcstaff               # allocation account
#BSUB -q express                    # queue
#BSUB -n 8                          # number of compute cores
#BSUB -W 4:00                       # walltime in HH:MM
#BSUB -R rusage[mem=4000]           # 32 GB of memory (4 GB per core)
#BSUB -R span[hosts=1]              # all cores from the same node
#BSUB -o %J.stdout                  # output log (%J : JobID)
#BSUB -eo %J.stderr                 # error log
#BSUB -L /bin/bash                  # Initialize the execution environment

module load star
WRKDIR=/sc/hydra/projects/hpcstaff/benchmark_star
STAR --genomeDir $WRKDIR/star-genome  --readFilesIn Experiment1.fastq  --runThreadN 8
--outFileNamePrefix Experiment1Star
```

$ bsub < star.lsf

Job <2937037> is submitted to queue <express>.

Last but not Least

Got a problem? Need a program installed? Send an email to:

hpchelp@hpc.mssm.edu