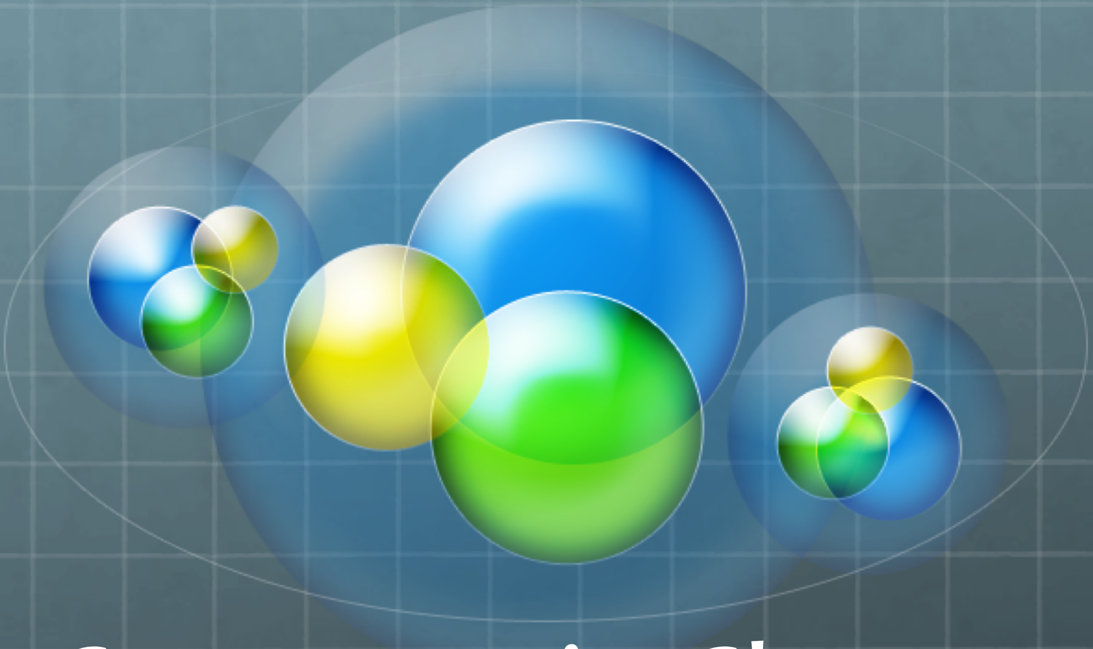


Concepts in Cluster Computing – Introduction to Biocomputing

HITS Research High Performance Computing

August, 2019

Roadmap

- Great Lakes overview
- Logging in to Great Lakes
- Slurm script creation
- Job submission
- Job monitoring
- Advanced topics

Great Lakes

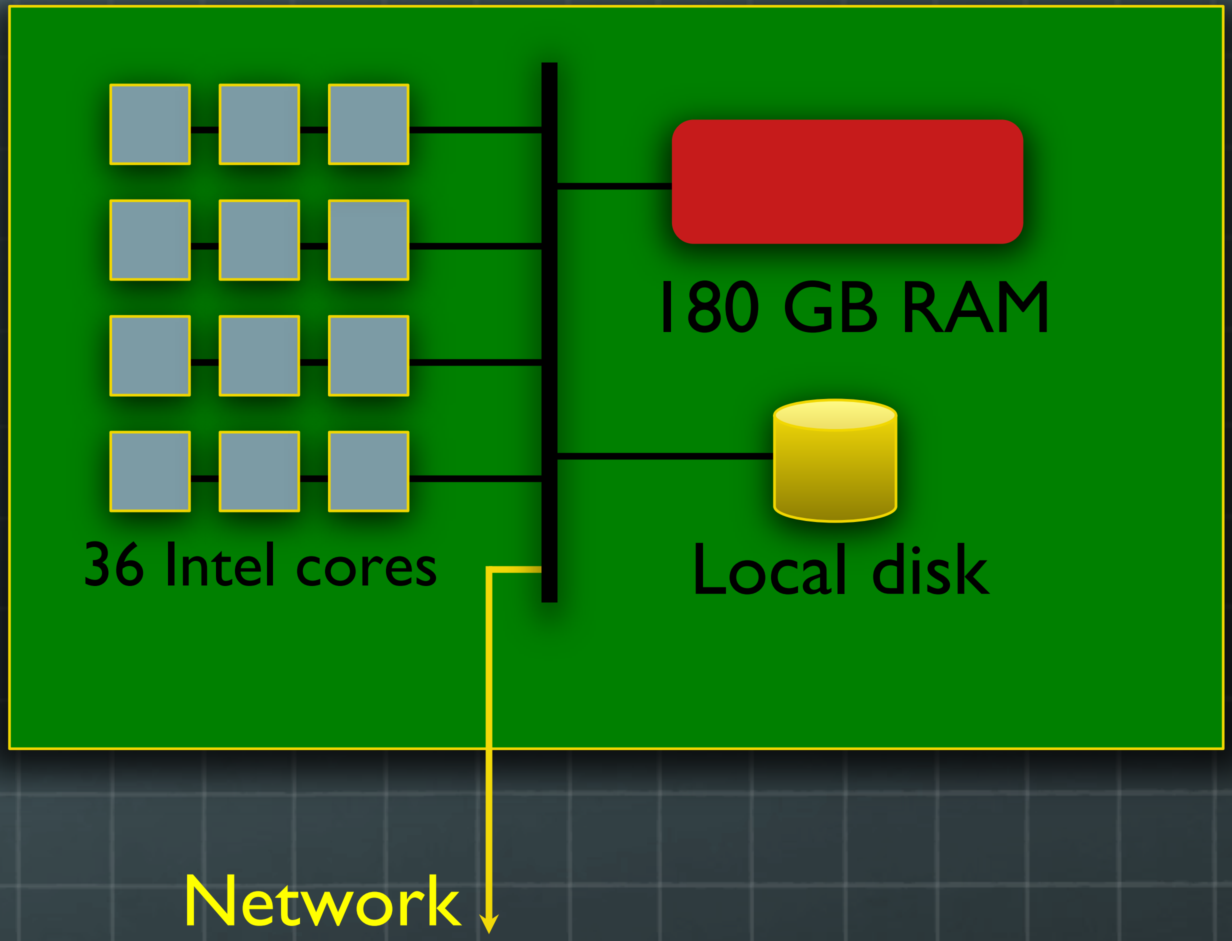
Great Lakes

Great Lakes is a university-wide shared computational discovery / high-performance computing service.

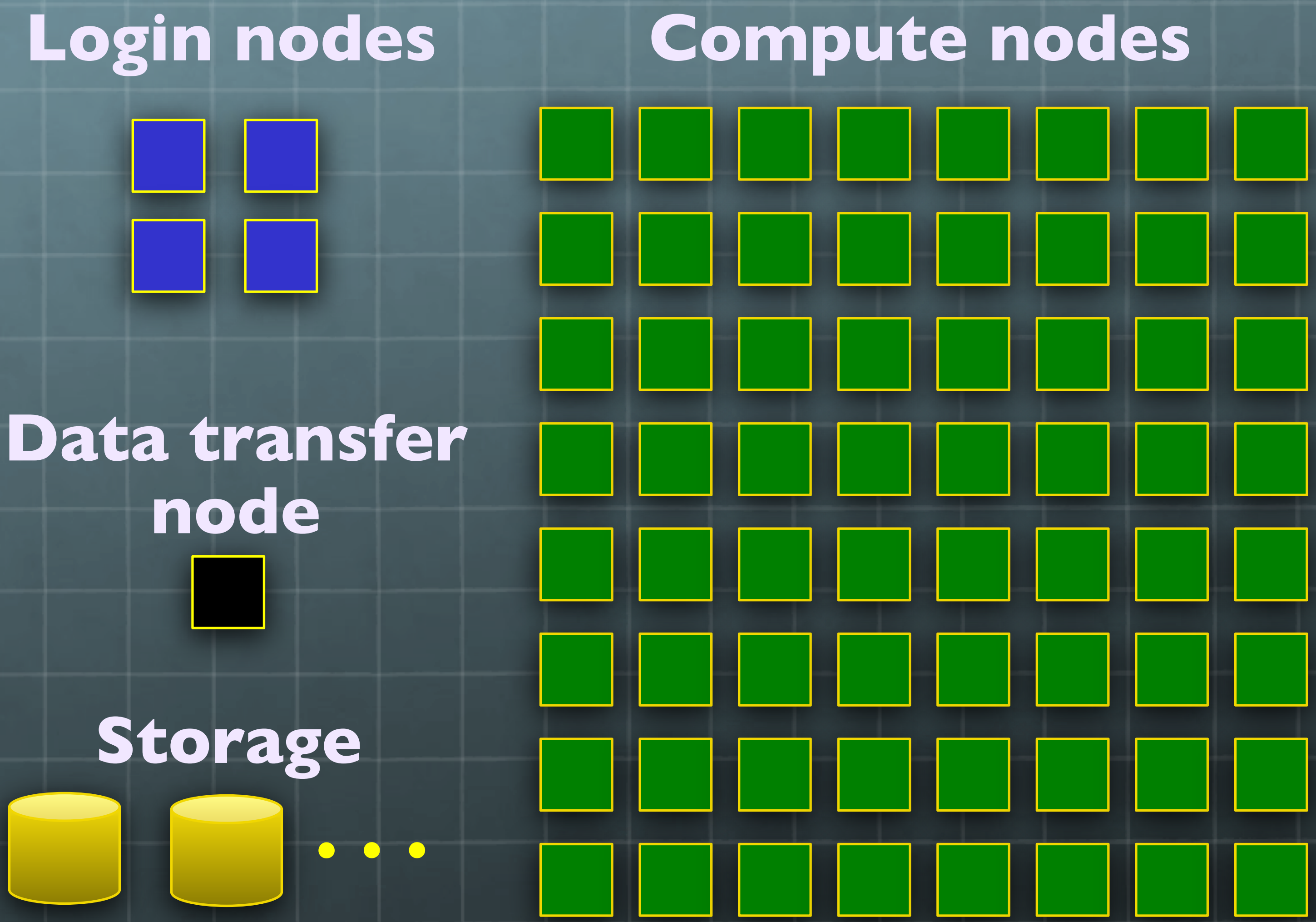
- Provided by Advanced Research Computing at U-M
- Operated by ARC Technology Services
- Procurement, licensing, billing by U-M ITS
- Successor to Flux, operating 2010-2019

https://arc-ts.umich.edu/greatlakes/

A Great Lakes Node



The Great Lakes Cluster



Login nodes

Compute nodes

Data transfer node

Storage

Logging In To Great Lakes

Connecting via ssh

- Terminal emulators
  - Linux and Mac OS X
    - Start Terminal
    - Use `ssh` command
  - Windows
    - MobaXterm
      http://mobaxterm.mobatek.net/
    - U-M PuTTY/WinSCP (Compute at the U)
      http://its.umich.edu/computing/computers-software/compute/
    - PuTTY
      http://www.chiark.greenend.org.uk/~sgtatham/putty/

Logging in to Great Lakes

Great Lakes user login, Duo and Level 1 password are required

- For Mac or other Linux workstation, from a terminal window type:
  ssh uniqname@greatlakes.arc-ts.umich.edu

- On a PC, start MobaXterm. If you have not attached to Great Lakes and saved your session, click on the "Session" button. In the Session settings box, select SSH.

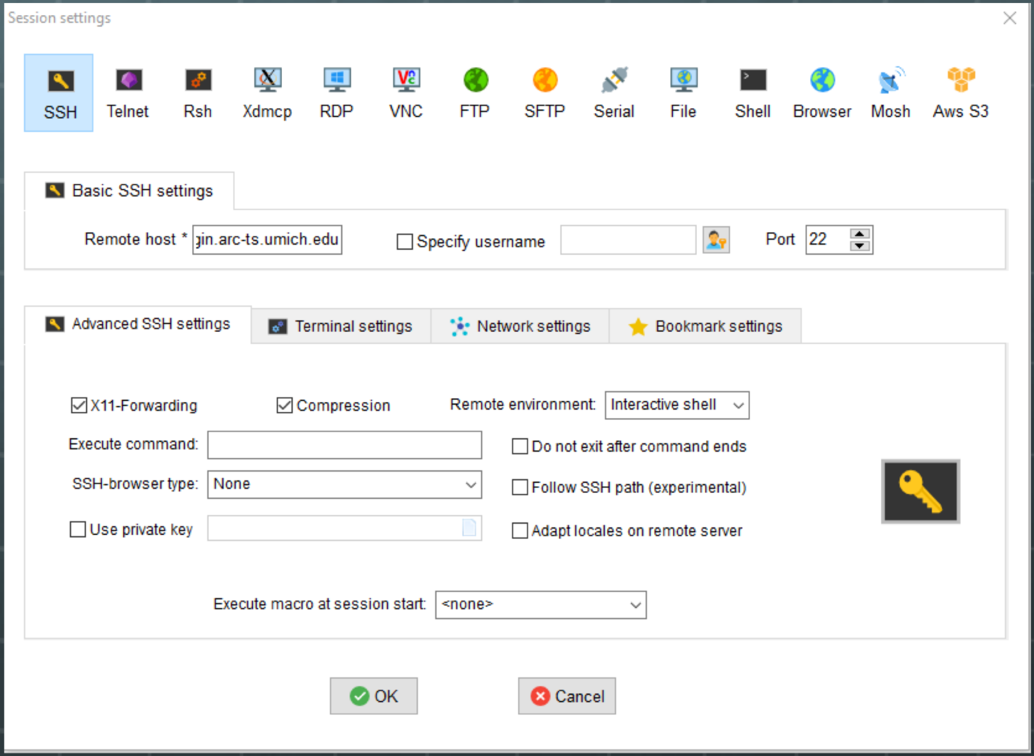
Remote Host:
greatlakes.arc-ts.umich.edu

In the Advanced SSH settings tab
Check:
X11-Forwarding, Compression

SSH-browser type:
None

Remote environment:
Interactive shell

Click OK. A terminal window will open with the prompt 'Login as :'
Type your uniqname and press enter.


Session settings

Basic SSH settings

Remote host * gin.arc-ts.umich.edu ☐ Specify username Port 22

Advanced SSH settings | Terminal settings | Network settings | Bookmark settings

☑ X11-Forwarding ☑ Compression Remote environment: Interactive shell

Execute command:

☐ Do not exit after command ends

SSH-browser type: None

☐ Follow SSH path (experimental)

☐ Use private key

☐ Adapt locales on remote server

Execute macro at session start: <none>

Logging in to Great Lakes

- You will be prompted, “Password:”

  Enter your Level 1 password and press enter.

- You will be prompted: “Enter a passcode or select one of the following options:”

  Select your Duo method of two factor authentication and press enter.

- You will be randomly connected to a Great Lakes login node

A shell on a login node

- Now you have a shell on the login node
- Your shell prompt has changed to something like `[uniqname@gl-login1 ~]$`
- Try these simple commands here!
  - ~$ date   ~$ id   ~$ ps   ~$ df -kh
  - ~$ who   ~$ top # type Control-c or q to exit
- Great Lakes is for HPC work
  - The login nodes are reserved for copying data, editing and compiling programs and scripts, and short test runs
  - For HPC computation, you should run on the compute nodes
    - A batch job will always run on the compute nodes
    - An *interactive* batch job will get you a shell on a compute node

Why command line?

1. Linux was designed for the command line
2. You can create new Linux commands using the command line, without programming
3. Many systems provide only the command line, or poorly support a GUI interface
    - Such as most HPC systems
4. Many things can be accomplished only through the command line
    - Much systems administration & troubleshooting
5. You want to be cool

Access to Software

- Software modules have been installed on Great Lakes to make it easy to use the software which you need. Use the module command to setup the software for your session.

  module avail

  module list

  module load *software_package*

  module unload *software_package*

  module purge

  module spider *some_text* (searches if software is available)

  if the software is available as part of a collection, module spider will tell you which collection module you need to load before being able to access the software you want

- You can create your own modules in your /home directory if you wish to install software yourself.
- Useful documentation about how software is managed on the clusters and what is available:
  - https://arc-ts.umich.edu/greatlakes/software/lmod/
  - https://arc-ts.umich.edu/greatlakes/software/

Slurm Script Creation

Slurm Script Creation

- Slurm Workload Manager (formerly known as Simple Linux Utility for Resource Management)
- With so many users on the system and a finite number of resources, the scheduler takes job requests and allocates resources, assigns which compute node(s) the job will run on and maintains ‘fair sharing’ of the cluster.
- The scheduler also maintains job logs which are useful for:
  - selecting the proper amount of resources to request
  - troubleshooting issues with a user’s workflow
  - identifying hardware issues with the compute nodes

A Sample Slurm Script

```
#!/bin/bash
#SBATCH --job-name=Rcoinflip
#SBATCH --account=biobootcamp
#SBATCH --partition=standard
#SBATCH --nodes=1
#SBATCH --ntasks=1
#SBATCH --cpus-per-task=1
#SBATCH --mem=8G
#SBATCH --time=30:00
#SBATCH --mail-user=kgweiss@umich.edu
#SBATCH --mail-type=FAIL,END
#SBATCH --output=./%x-%j

#  Include these lines to print to your output file to verify script is working
if [[ $SLURM_JOB_NODELIST ]] ; then
   echo "Running on"
   scontrol show hostnames $SLURM_JOB_NODELIST
fi

# Load your modules
module load R
```

Scheduler Directives

```
#  Put your job commands after this line
date
R CMD BATCH --no-save --no-restore coinflip.R
date
```

Your workflow goes here

/scratch

- What is /scratch?

  /scratch is a high performance file system connected to fast networking which allows for high throughput reads/writes

- Why should I use /scratch?

  In a word… performance

- What should I do with my files on /scratch

  Once your program is done running, copy your results files to your /home directory or other storage you have access to

- Your /scratch space for this class is:

  /scratch/biobootcamp_root/biobootcamp/*uniqname*

- WARNING: /scratch is NOT backed up. Make sure that files contained on /scratch are not your only copy.

Job Submission

Job Submission

- For a job to run on a cluster, you must submit the job to the scheduler.
- The command to send jobs to the scheduler is:

```
sbatch Slurm_script_name
```

- To get an estimate of when your job will run (optional):

```
sbatch --test-only Slurm_script_name
```

- Useful commands for working with queued jobs:

scancel <job ID #> - removes the job from the queue (will work also if the job is running)

scontrol hold <job ID #> - puts the job on hold (pause)

scontrol release <job ID #> - releases a held job (resume)

Job Monitoring

Job Monitoring

- Often when you have very long jobs and/or very many jobs, you wish to know the status of your job(s). There are a variety of ways you can monitor your jobs.

squeue -u <uniqname> - print a list of jobs for a user

scontrol show job <job ID #> - list job details for a given job

Job Submission Example

- First, let’s get the files into your /scratch directory.

  type: cd /scratch/biobootcamp_root/biobootcamp/*uniqname*

  cp ../kgweiss/coinflip.* .

- Next, modify coinflip.sbat to send email to your uniqname.

  type: nano coinflip.sbat

  modify: #SBATCH --mail-user=*uniqname*@umich.edu

  save and exit

- Let’s see how long we should expect to wait for the job to start

  type: sbatch --test-only coinflip.sbat

- Finally, submit the job

  type: sbatch coinflip.sbat

- Now, let’s check the status of our job

  type: squeue –u <uniqname>

Advanced Topics

Advanced Topics

- Job dependencies – Start a job after some condition

  sbatch --dependency=afterok:<job1_ID_num> job2.sbat

  https://arc-ts.umich.edu/greatlakes/slurm-user-guide/

- Job arrays

  #SBATCH --array=1-10

  matlab < test-${SLURM_ARRAY_TASK_ID}.m (command to execute)

  https://arc-ts.umich.edu/greatlakes/slurm-user-guide/

- Interactive jobs

  ```
  srun --account=<acct_name> --time=02:00:00 --nodes=1 --ntasks=1 --cpus-per-task=2 --mem=3GB --pty /bin/bash
  ```

  https://arc-ts.umich.edu/greatlakes/slurm-user-guide/

- Globus - another, very useful way to copy files

  See https://www.globus.org/ for details

Job Arrays

Slurm job arrays provide a convenient way to submit a large number of independent processing jobs. For example, Slurm job arrays can be useful for applying the same or similar computation to a collection of data sets. When a job array script is submitted, a specified number of “array tasks” are created based on the “master” sbatch script.

Sample code in:
/scratch/biobootcamp_root/biobootcamp/jkenyon/job_array

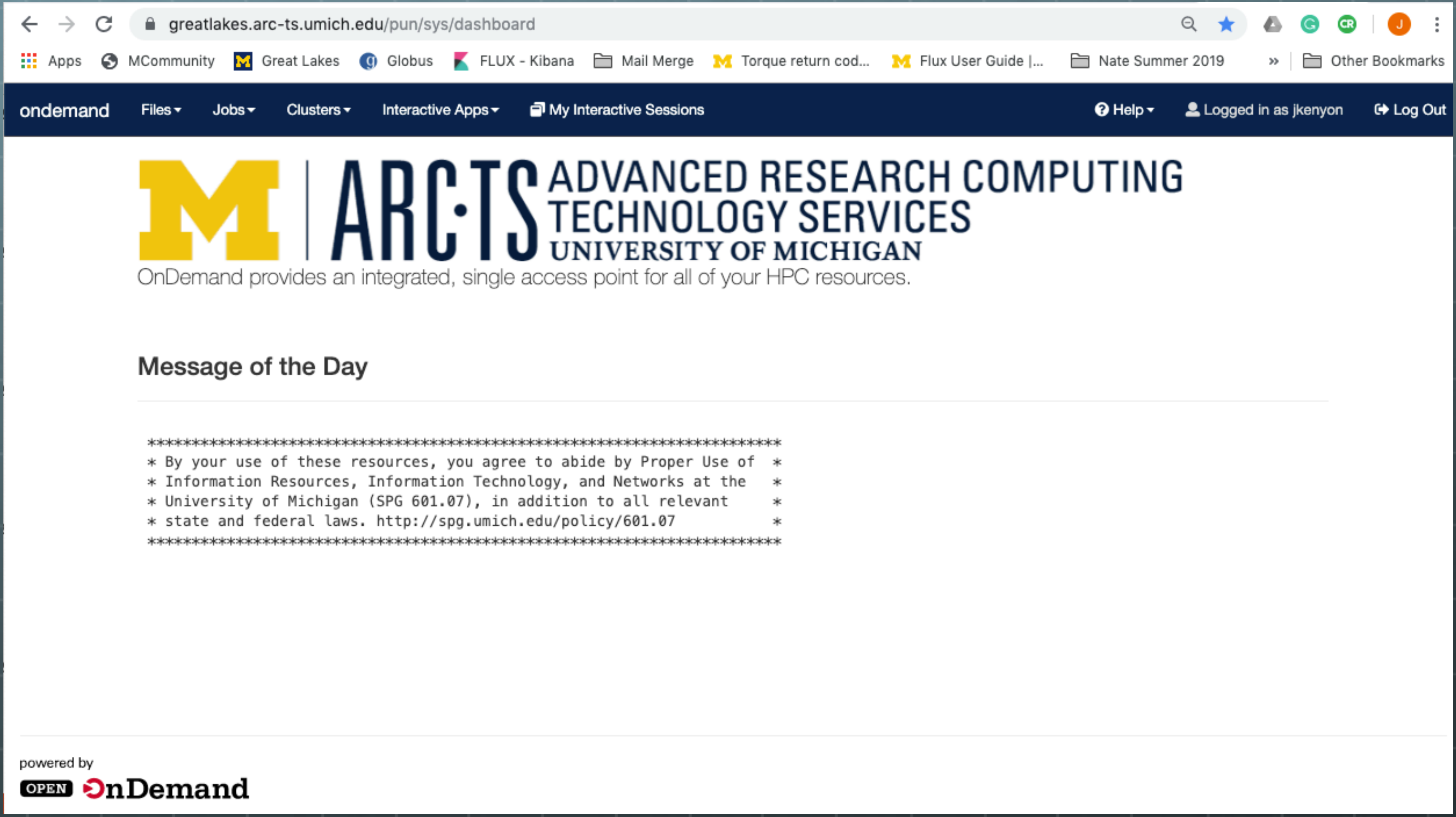
Interactive Jobs

- Command line

  ```
  srun --account=biobootcamp --time=02:00:00 --nodes=1 --ntasks=1 --cpus-per-task=2 --mem=3GB --pty /bin/bash
  ```

- Graphical
  - Jupyter Notebook
  - X desktop
  - Accessed through https://greatlakes.arc-ts.umich.edu

OPEN OnDemand

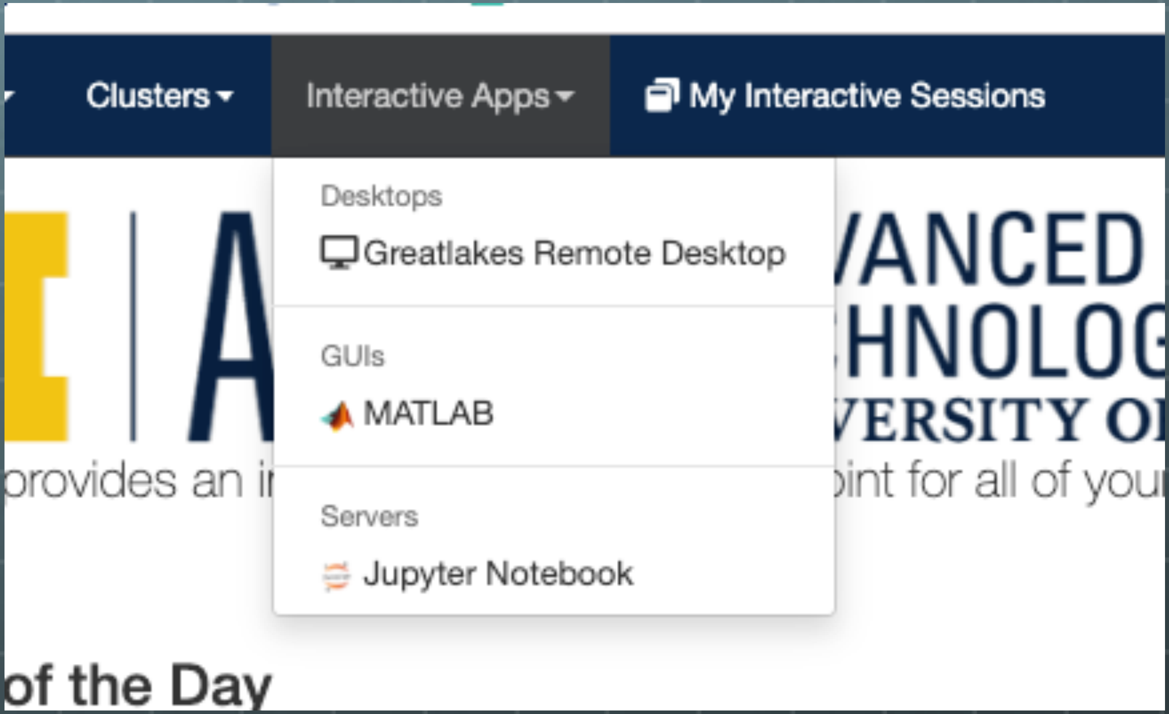


OnDemand provides an integrated, single access point for all of your HPC resources.

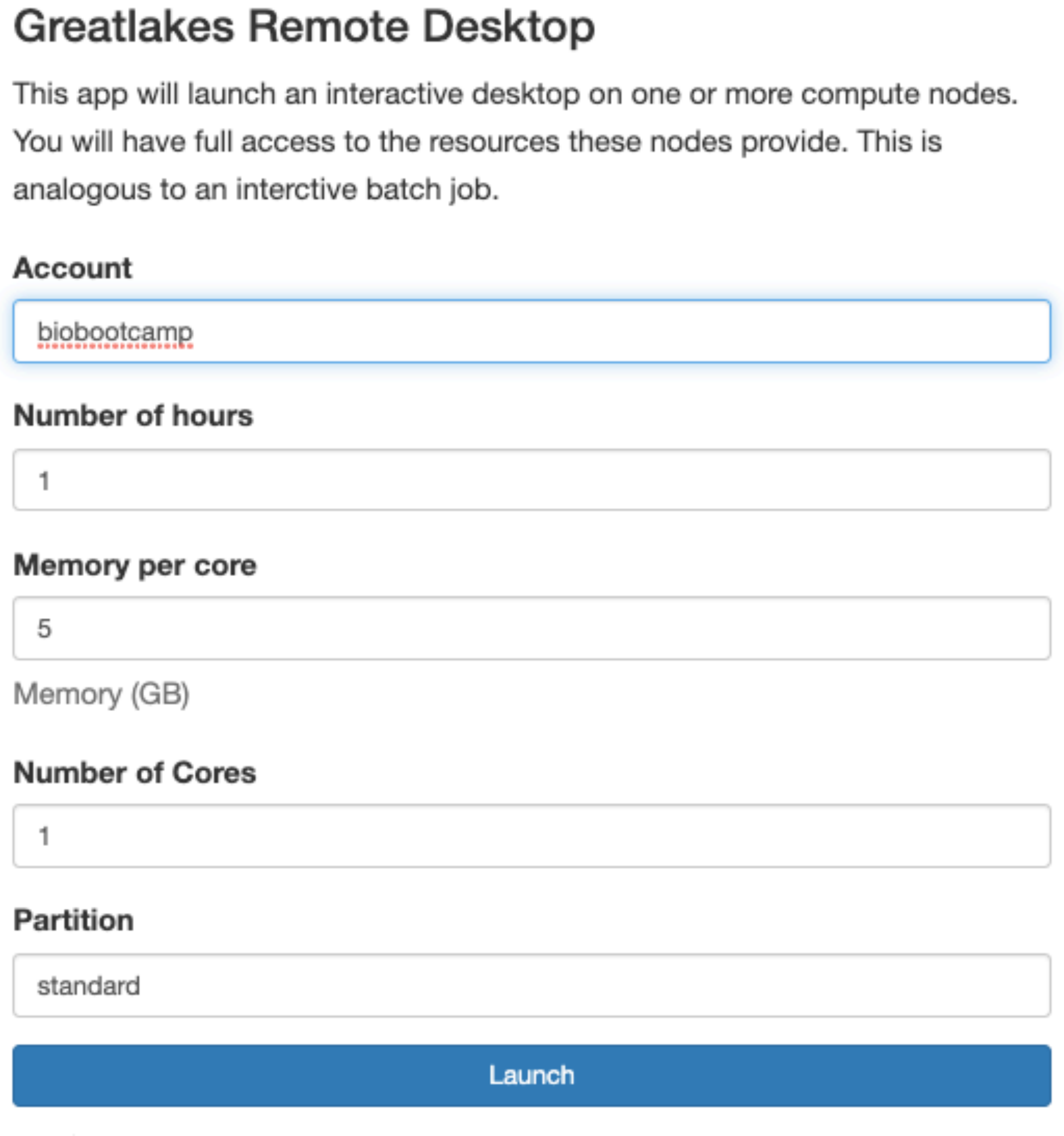
Message of the Day

```
************************************************************************
* By your use of these resources, you agree to abide by Proper Use of  *
* Information Resources, Information Technology, and Networks at the   *
* University of Michigan (SPG 601.07), in addition to all relevant     *
* state and federal laws. http://spg.umich.edu/policy/601.07           *
************************************************************************
```

OPEN OnDemand



OPEN OnDemand



Greatlakes Remote Desktop

This app will launch an interactive desktop on one or more compute nodes. You will have full access to the resources these nodes provide. This is analogous to an interctive batch job.

Account

biobootcamp

Number of hours

1

Memory per core

5

Memory (GB)

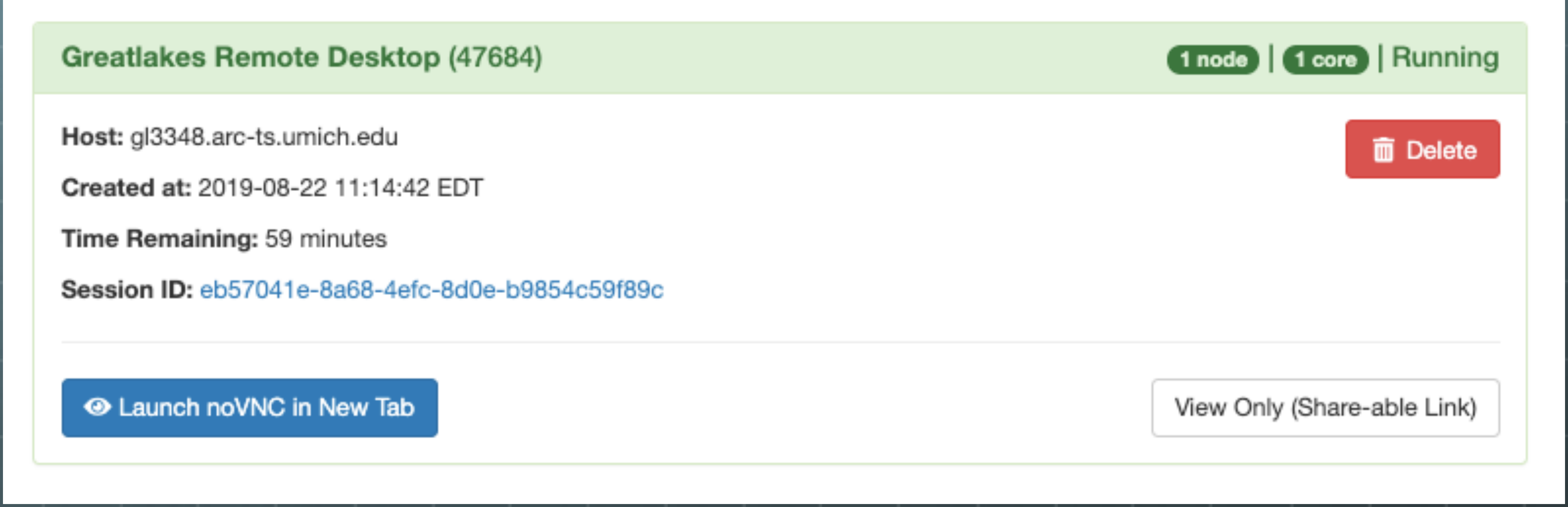
Number of Cores

1

Partition

standard

OPEN OnDemand



Greatlakes Remote Desktop (47684)

Host: gl3348.arc-ts.umich.edu

Created at: 2019-08-22 11:14:42 EDT

Time Remaining: 59 minutes

Session ID: eb57041e-8a68-4efc-8d0e-b9854c59f89c

Useful Links

- https://www.schedmd.com/
- http://arc-ts.umich.edu/
- http://arc-ts.umich.edu/resources/
- https://arc-ts.umich.edu/greatlakes/software/
- http://fluxhpc.blogspot.com/
- https://twitter.com/ARCTS_UM/
- http://its.umich.edu/accounts-access/uniqnames-passwords/two-factor-authentication /
- https://www.globus.org/

Contact Info

- hpc-support@umich.edu
- Slack – DANG! umich-dang.slack.com
- Personal Contact Information
    - kgweiss@umich.edu
    - jkenyon@umich.edu