Introduction to High Performance Computing (HPC)

Alex Townsend
Research Computing Center
September 9 2026

Agenda

- What is High Performance Computing (HPC)
  - HPC Definitions
- Accessing the HPC
- Support from RCC Team

What is High Performance Computing?

Specialized Hardware

- Contains more processors than the average home computer
- Has several generations of both AMD and Intel Processors
- Can house custom machines (via a research grant) with specialized hardware like special CPUs, DSPs, SSDs, GPUs and otherwise.

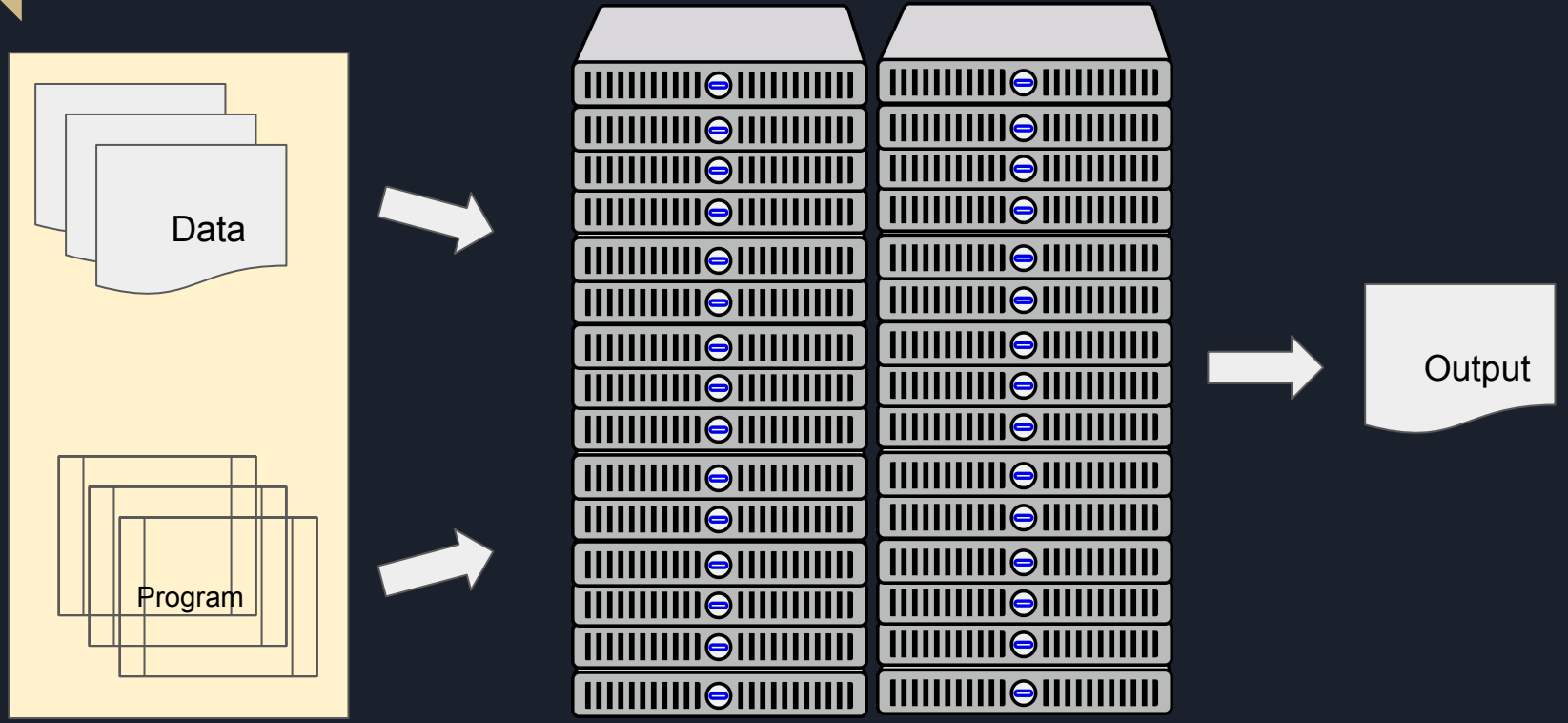
Offload Your Work

- Takes pressure off your regularly used desktop, laptop or other devices
- Scales computational processes not possible on home machines
  - parallel processes
- Runs tasks that need to be repeated many times over a long period of time

RCC System Metrics

649 Nodes	114 GPUs	20,144 Processors	958.64 TeraFLOPS	29,170,841 jobs
Beowulf Cluster AlmaLinux 9.7	NVIDIA: - H100 (2) - A4000 - A4500 Ada Gen - GTX 1080 Ti	AMD, Intel	Very Fast Maximum Speeds	Since 2008

HPC Job



HPC Definitions

How HPC Works

FSU's HPC is a shared resource for all FSU researchers

Job Scheduler: SLURM

- Directs jobs so they can be processed quickly and effectively
- Allocates resources to jobs so the cluster does not freeze up

User Accounts

- Logins and Home Directories

Slurm Accounts

- Queues / Partitions



Job Scheduler: SLURM



How to Access the HPC

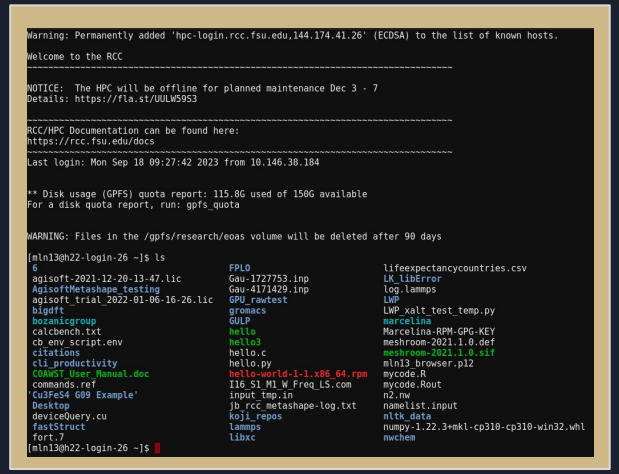
Ways to Use HPC

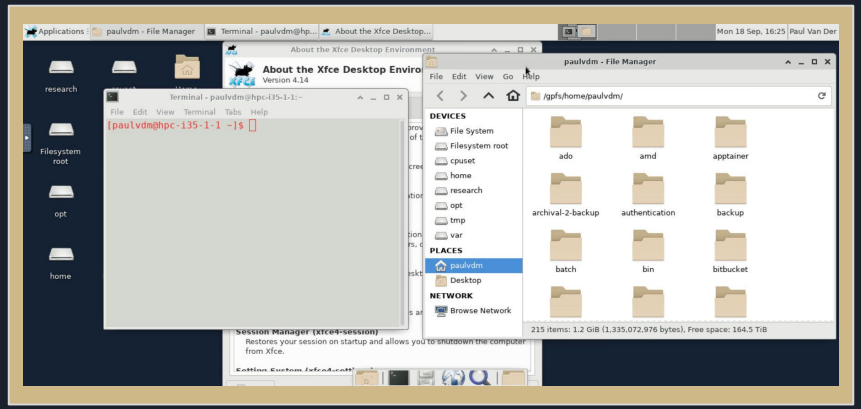
Command-Line Interface

- Any SSH-Supporting CLI Terminal
  - Windows PowerShell
  - Cygwin Emulator
  - MobaXTerm Emulator
  - PuTTY
  - Linux Terminal
  - Mac Terminal

Graphical Interface

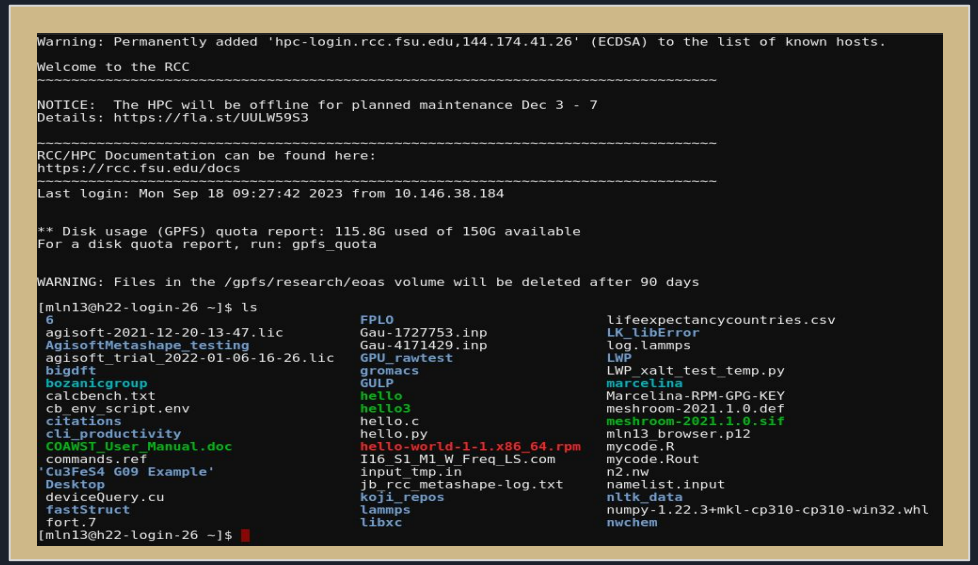
- Open OnDemand





Command Line Interface

- RCC uses Linux on all of our compute nodes and login nodes
  - RCC offers an Intro to Linux Workshop
- Connect using SSH (Secure Shell)



Open OnDemand Applications

GUI Applications

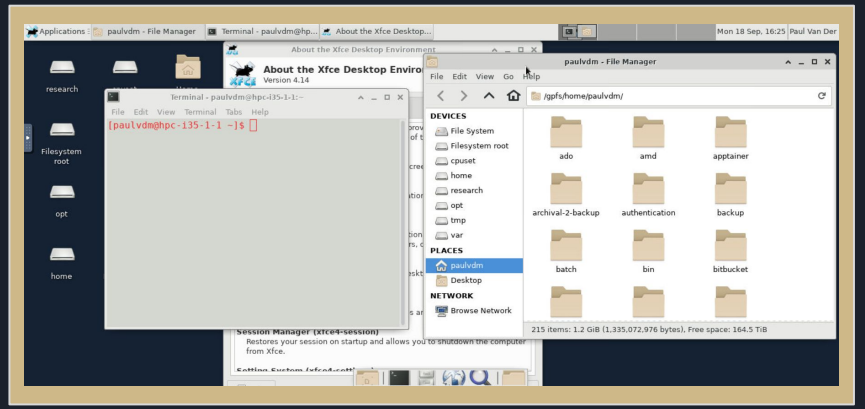
- Desktop
- MATLAB
- STATA
- VisIt

Servers

- Jupyter Notebooks
- RStudio Server

Visualizations

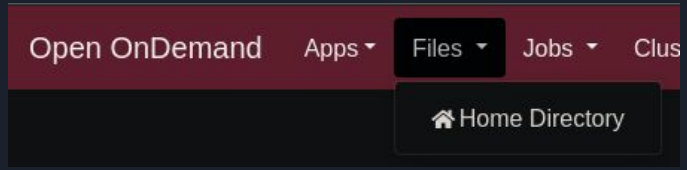
- PyMOL
- VMD

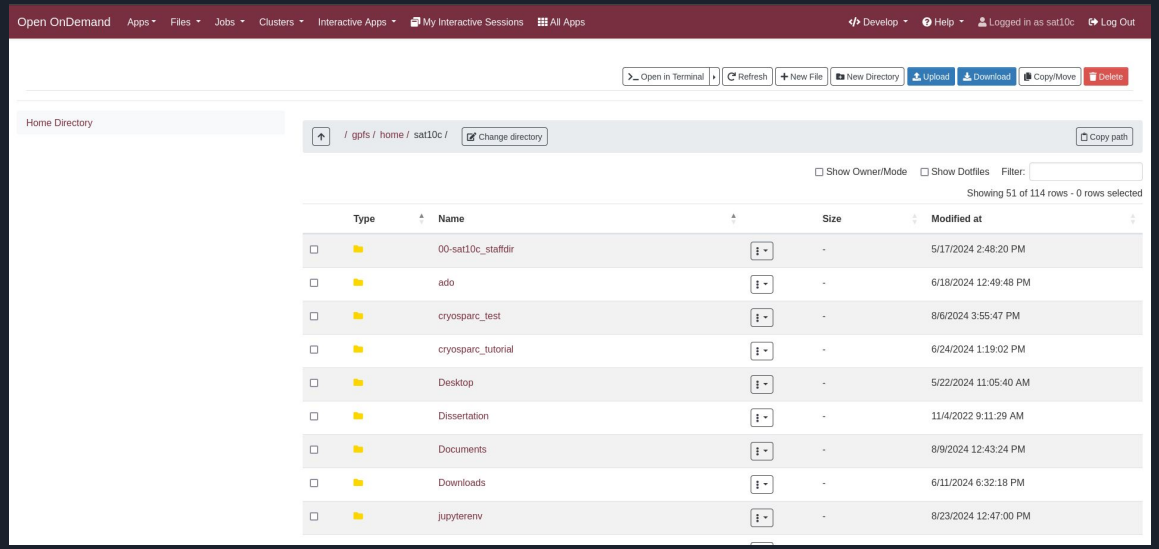


ood.rcc.fsu.edu

https://docs.rcc.fsu.edu/hpc/ood/

Open OnDemand File Manager

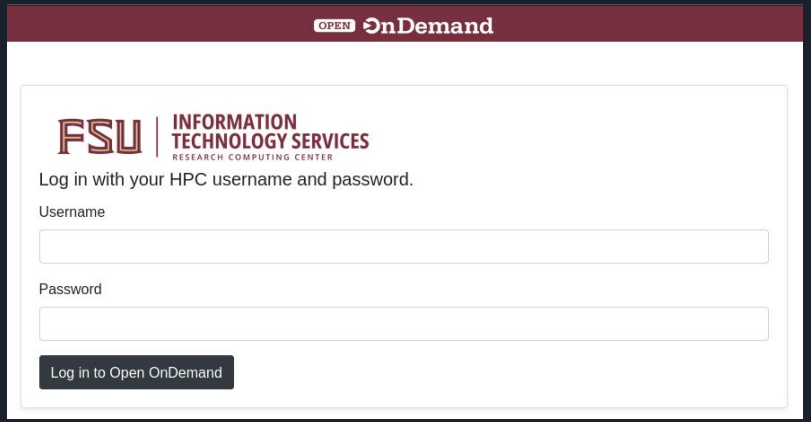




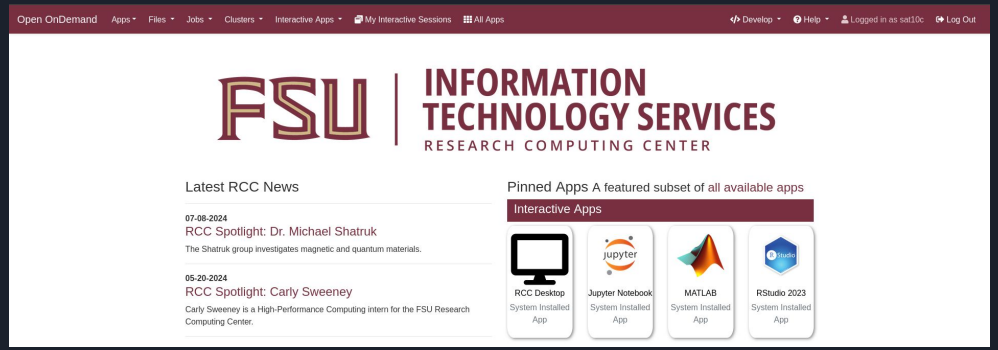
- Browse files on the HPC
- Upload small files (code or small datasets < 10 GB)
- Download small files (code or small datasets < 10 GB)
- Open and look at files.
- Edit files.
- Delete and move files.

Submitting Jobs to the HPC System

Open OnDemand Interactive Jobs

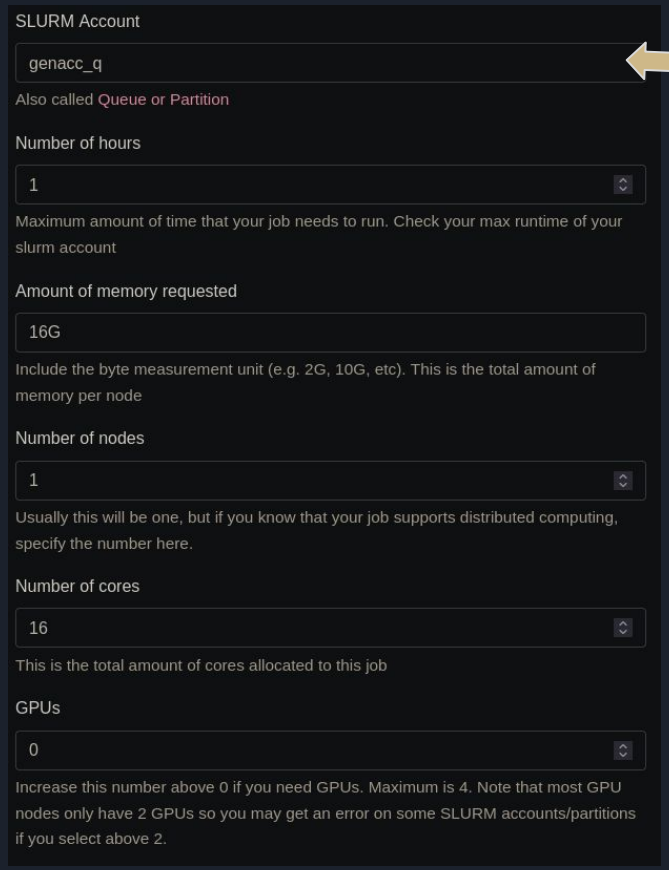


- ood.rcc.fsu.edu
- Login credentials
  - *Your RCC Account details*



- Open OnDemand landing page.
- Interactive Apps
- Job Creator
- Terminal Access
- File Browser

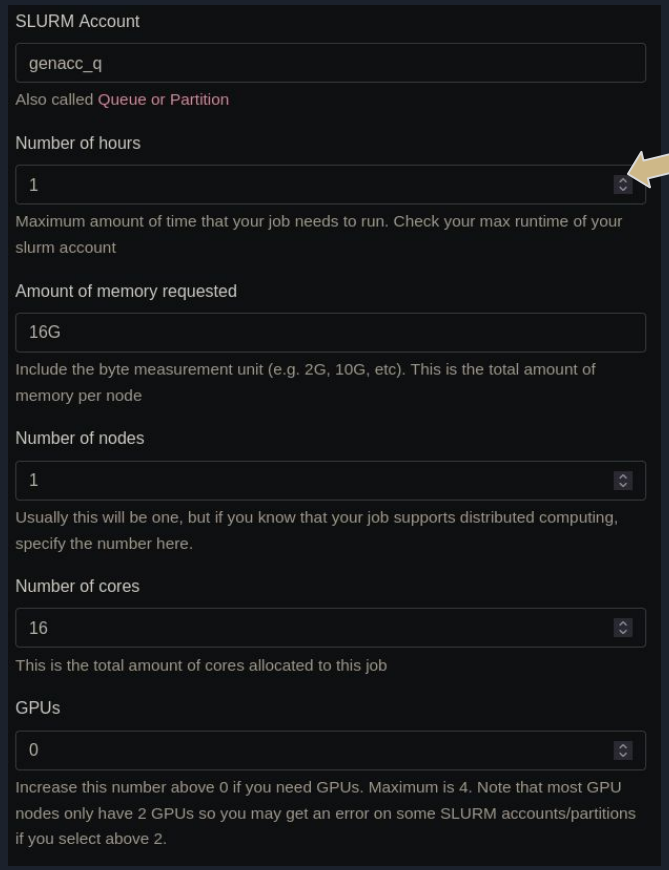
Resource Allocation (OOD)



Queue/SLURM Account

- *The set of computers you want to use*
- *If you leave this blank, it will default to the **genacc_q** SLURM Account.*

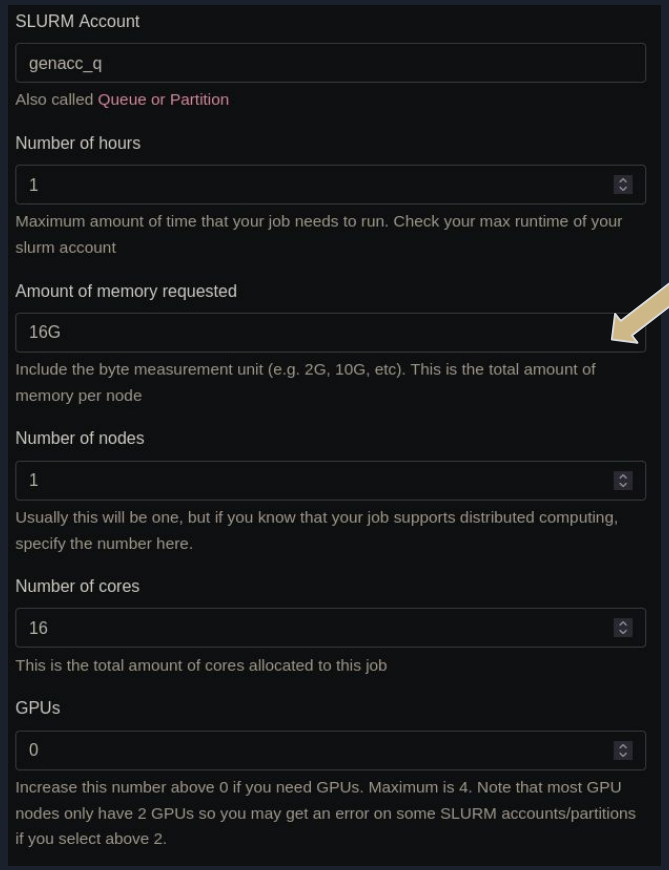
Resource Allocation (OOD)



Number of Hours

- *Maximum amount of time you expect to need for your work*

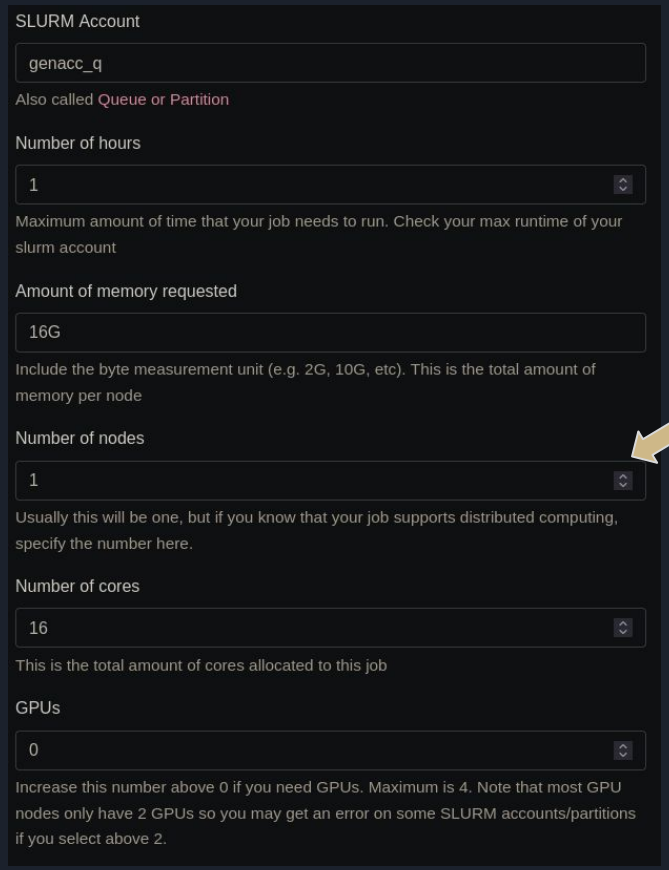
Resource Allocation (OOD)



Amount of Memory Requested

- *The total amount of memory you expect to need per node for your job.*

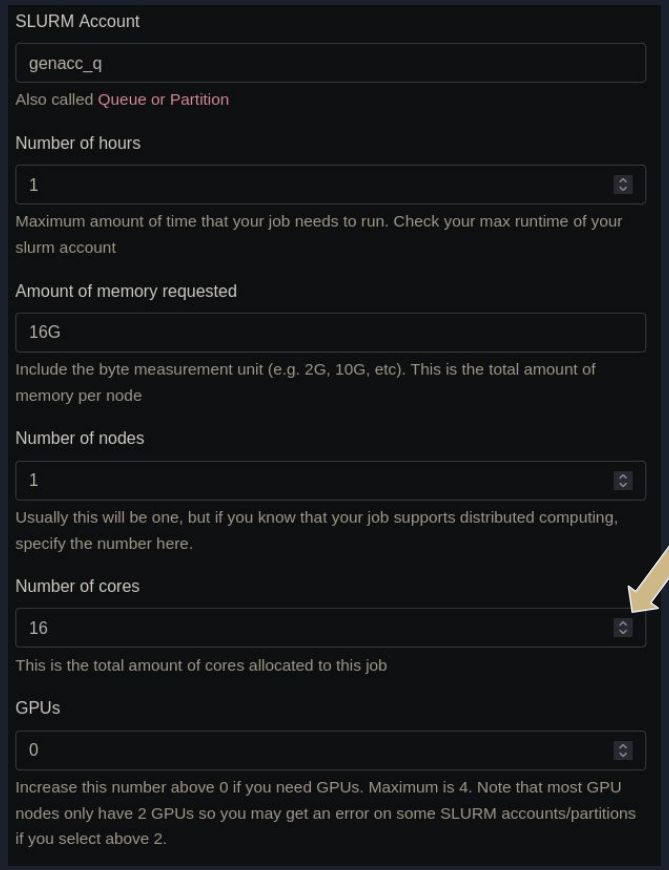
Resource Allocation (OOD)



Number of Nodes

- *The number of individual computer machines you want to use. (Not all OOD apps will have this option available!)*

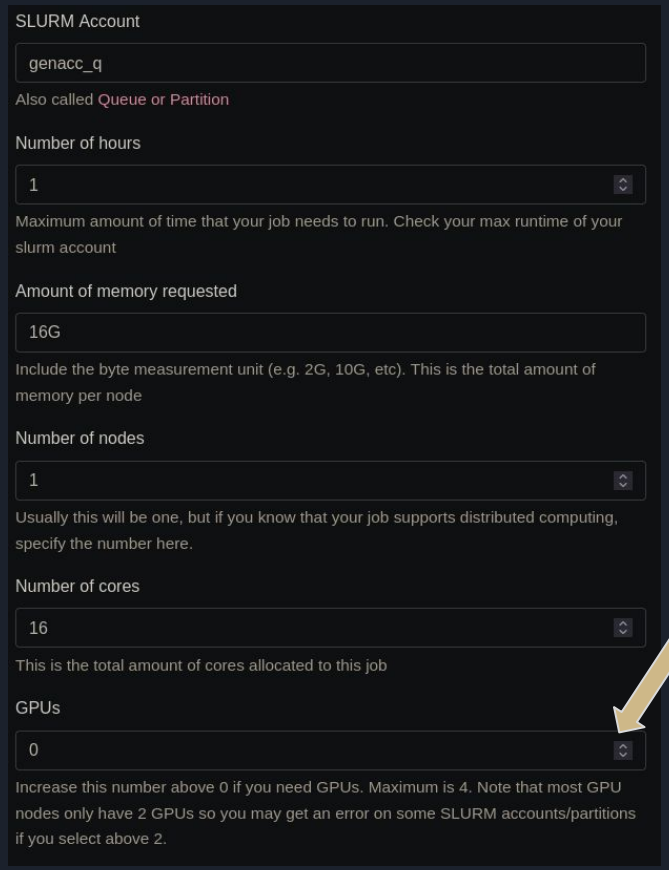
Resource Allocation (OOD)



Number of Cores

- *The total number of cores you want to use for your job.*

Resource Allocation (OOD)



GPUs

- *The number of GPU cards you want to use for your job (can leave at 0 if none needed).*

Command-Line Batch Jobs

```
#!/bin/bash
#SBATCH -J "MyJob"
#SBATCH -A backfill2
#SBATCH -t 4:00:00
#SBATCH --mem=16G
#SBATCH -n 16
#SBATCH -N 1
#SBATCH --gres=gpu:1

module load gnu openmpi
srun example.x
```

Job Name

- *The job name parameter is simply an identifier for your job. This can be helpful for keeping track of what each job is doing.*

Command-Line Batch Jobs

```
#!/bin/bash
#SBATCH -J "MyJob"
#SBATCH -A backfill2
#SBATCH -t 4:00:00
#SBATCH --mem=16G
#SBATCH -n 16
#SBATCH -N 1
#SBATCH --gres=gpu:1

module load gnu openmpi
srun example.x
```

Queue/SLURM Account

- *If this is not specified, it will default to the **genacc_q** SLURM Account.*

Command-Line Batch Jobs

```
#!/bin/bash
#SBATCH -J "MyJob"
#SBATCH -A backfill2
#SBATCH -t 4:00:00
#SBATCH --mem=16G
#SBATCH -n 16
#SBATCH -N 1
#SBATCH --gres=gpu:1

module load gnu openmpi
srun example.x
```

Number of Hours

- *Time requests are more flexible in batch jobs. SLURM's time allocation format is:*
- ***Days-Hours:Minutes:Seconds***
- *Example: 14-00:00:00 = 14 Days*
- *Example: 34:25:16 = 34 Hours, 25 Minutes and 16 Seconds.*
- *If this is not specified, it will default to the listed* ***Default Wall Time*** *listed in the documentation.*

Command-Line Batch Jobs

```
#!/bin/bash
#SBATCH -J "MyJob"
#SBATCH -A backfill2
#SBATCH -t 4:00:00
#SBATCH --mem=16G
#SBATCH -n 16
#SBATCH -N 1
#SBATCH --gres=gpu:1

module load gnu openmpi
srun example.x
```

Amount of Memory Requested

- *If you don't specify this, you'll be automatically allocated:*
  - *Number of Cores * Default Memory per Core (see the documentation) as this number depends on the SLURM Account)*

Command-Line Batch Jobs

```
#!/bin/bash
#SBATCH -J "MyJob"
#SBATCH -A backfill2
#SBATCH -t 4:00:00
#SBATCH --mem=16G
#SBATCH -n 16
#SBATCH -N 1
#SBATCH --gres=gpu:1

module load gnu openmpi
srun example.x
```

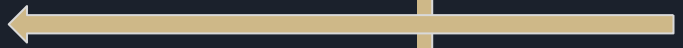
Number of Cores

- *This is the number of physical CPU processors you want to use.*
- *Most of the benefits of HPC come from parallel processing.*
- *Check your software's documentation to see if it supports this!*

Command-Line Batch Jobs

```
#!/bin/bash
#SBATCH -J "MyJob"
#SBATCH -A backfill2
#SBATCH -t 4:00:00
#SBATCH --mem=16G
#SBATCH -n 16
#SBATCH -N 1
#SBATCH --gres=gpu:1

module load gnu openmpi
srun example.x
```



Number of Nodes

- *This is the number of full computers (think desktop/workstation towers for reference) that you want to use.*
- *This is for programs that support Distributed Computing.*
- *If this parameter is left out the job scheduler will decide this on its own.*

Command-Line Batch Jobs

```
#!/bin/bash
#SBATCH -J "MyJob"
#SBATCH -A backfill2
#SBATCH -t 4:00:00
#SBATCH --mem=16G
#SBATCH -n 16
#SBATCH -N 1
#SBATCH --gres=gpu:1

module load gnu openmpi
srun example.x
```



GPUs

- *If you don't want GPUs, you can omit the entire #SBATCH --gres=gpu: line.*

Command-Line Batch Jobs

```
#!/bin/bash
#SBATCH -J "MyJob"
#SBATCH -A backfill2
#SBATCH -t 4:00:00
#SBATCH --mem=16G
#SBATCH -n 16
#SBATCH -N 1
#SBATCH --gres=gpu:1

module load gnu openmpi
srun example.x
```



Environment Modules

- *Open OnDemand Apps will often do this for you. These load software and applications into your terminal environment for you.*

Command-Line Batch Jobs

```
#!/bin/bash
#SBATCH -J "MyJob"
#SBATCH -A backfill2
#SBATCH -t 4:00:00
#SBATCH --mem=16G
#SBATCH -n 16
#SBATCH -N 1
#SBATCH --gres=gpu:1

module load gnu openmpi
srun example.x
```



Command to Run your Program

- *In non-interactive jobs, you have to tell the computer what you want it to do in your job ahead of time.*

Command-Line Batch Jobs

```
#!/bin/bash
#SBATCH -J "MyJob"
#SBATCH -A backfill2
#SBATCH -t 4:00:00
#SBATCH --mem=16G
#SBATCH -n 16
#SBATCH -N 1
#SBATCH --gres=gpu:1

module load gnu openmpi
srun example.x
```

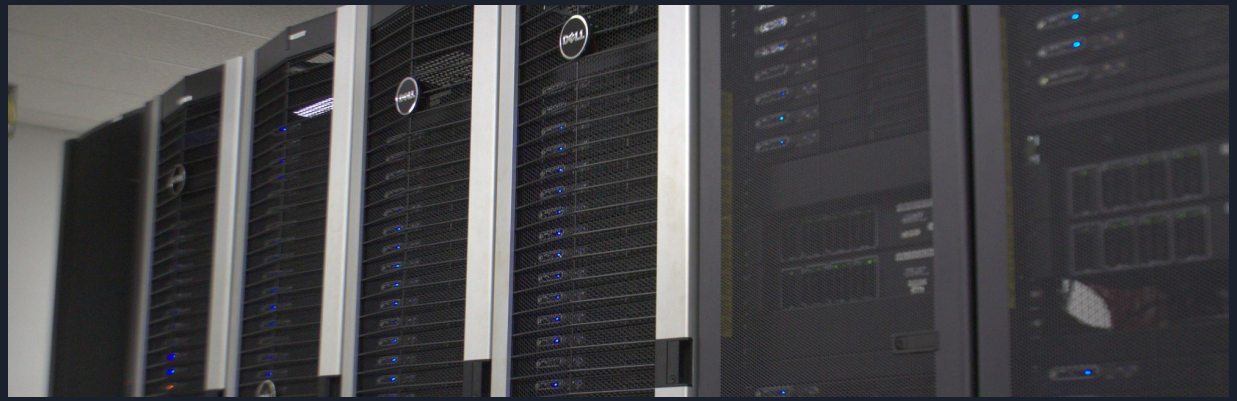
Translating the Job Script

- *Run it in the backfill2 SLURM Account*
- *Give me at most 4 hours to run my job*
- *I need 16 GB of memory per node for this.*
- *I also need 16 cores for this.*
- *I also need all 16 cores to be localized onto 1 physical computer node (don't spread the cores over multiple nodes).*
- *I also need 1 GPU card of any type.*
- *My program needs the OpenMPI built with the default GNU compilers.*
- *Please run my code called example.x*

Support from RCC Team

What is the FSU Research Computing Center (RCC)

- A unit of the FSU Information Technology Services department
- Originally a division of Scientific Computing back before 2013
- The team that hosts and administers the main supercomputing resource at FSU: the High Performance Computing system (HPC)



Register For an Account

Faculty Requirements

- FSU ID and password

Student Requirements

- FSU ID and password
- Faculty Sponsor

Guest (non-FSU)

- Guest FSU ID
- Faculty Sponsor



www.rcc.fsu.edu/manage

Infrastructure

Physical Location

- Cluster held in the Sliger Data Center in Innovation Park. Tours available!

Standard RCC accounts

- 1 Home directory per user
- 150GB of Parallel Storage
- Temporary Scratch Space (Space Limit is variable)
- Access to the General Access and Backfill Queues
- Accessible via Open On Demand and CLI

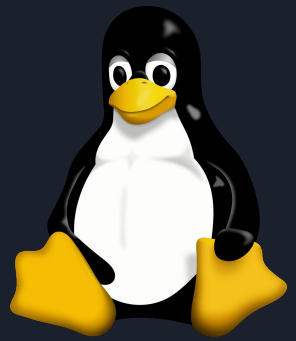


Infrastructure

Additional Features for Paid RCC Accounts

- Access to highly scalable Parallel and Archival storage systems
- Priority access to dedicated computing resources in your own queue
- Custom and specialized hardware and infrastructure (as available)





General Access Slurm Accounts

https://acct.rcc.fsu.edu/manage/hpc_partitions

Slurm Account	Default Runtime	Max Runtime	Max CPU Cores/Job	Max Jobs/User	Max Running Jobs/User
genacc_q	14 days	14 days	400	100	100
backfill	4 hours	4 hours	512	100	100
backfill2	4 hours	4 hours	512	100	100
condor	14 days	90 days	8	100	100
quicktest	10 minutes	10 minutes	8	2	1

Support Services

Basic Software Support

- Basic Software installation, configuration, and maintenance
- Technical support for installed applications and software
- Workflow process support and improvement assistance
- Assistance with HPC commands and utilities

Basic Hardware Support

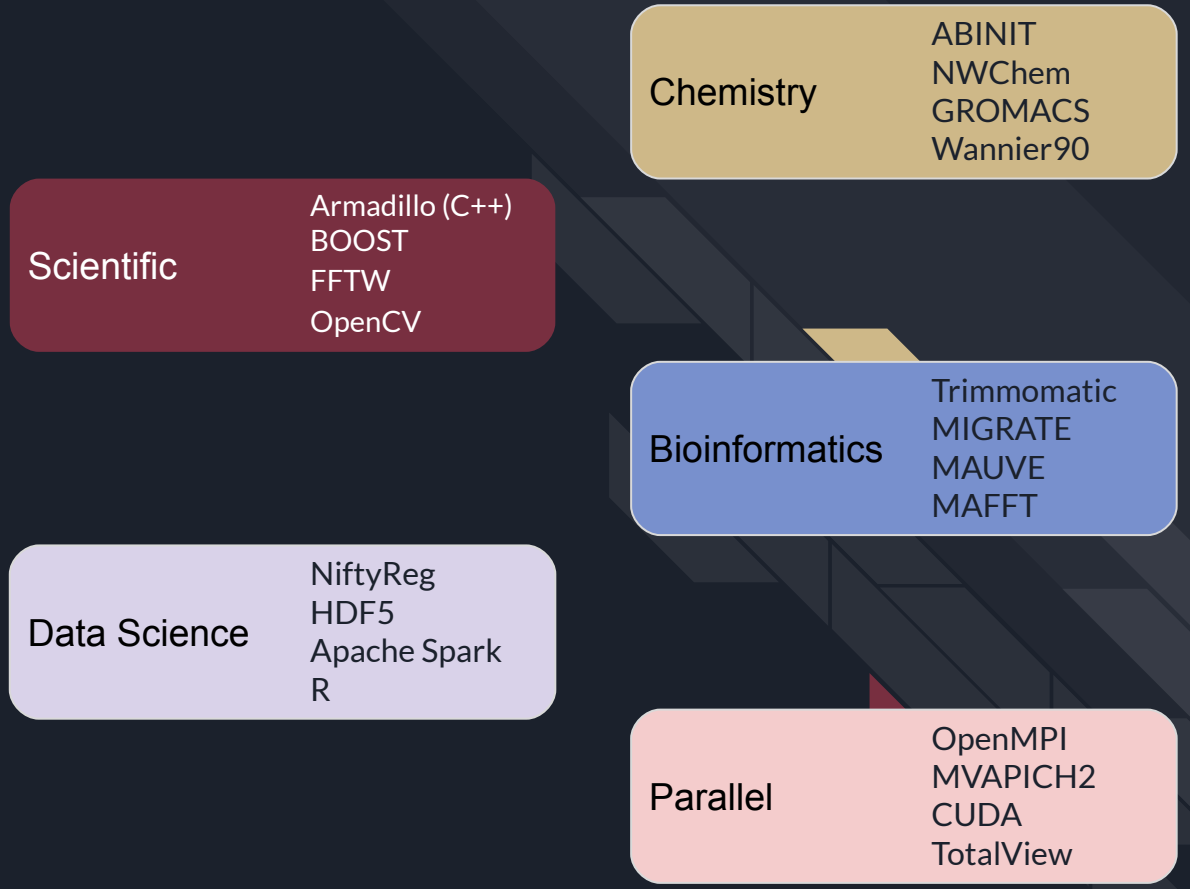
- Data Center Colocation

Support Services

Additional Paid Support (more Information)

- Software Development Consulting
- Complex or Highly Customized Software Installations
- In-depth HPC and software focused consulting for research projects

Software



https://rcc.fsu.edu/software

RCC Service Metrics



Average Jobs per Year

Research Software Tools Managed and Supported

Support Cases Serviced since 2022

Software Team Services

Email us at support@rcc.fsu.edu for any questions or concerns

- Regular Office Hours 8AM - 5PM Monday through Friday
  - We work a hybrid schedule and are available by Zoom or email
  - Dirac Science Library 151

- HPC Drivers Ed - Introductory HPC materials
- docs.rcc.fsu.edu - RCC Official Documentation

Questions?

Post-Workshop Exercise

1. Navigate to https://ood.rcc.fsu.edu
2. Log in with your RCC Credentials
3. Navigate to the RCC Desktop
4. Set up a RCC Desktop job with the following resources:
    a. *4 Hours*
    b. *Using the **workshop** SLURM Account*
    c. *4 Cores*
    d. *16 GB of Memory*
    e. *No GPUs*
    f. *Leave everything else default*
5. Launch the job
6. Open the RCC Desktop window in OOD and look around a bit!

Post-Workshop Exercise

1. Navigate to https://ood.rcc.fsu.edu
2. Log in with your RCC Credentials
3. Navigate to the MATLAB Interactive App
4. Set up a MATLAB Job with the following resources:
    a. *4 Hours*
    b. *Using the **workshop** SLURM Account*
    c. *4 Cores*
    d. *16 GB of Memory*
    e. *No GPUs*
    f. *MATLAB 2022b Version*
    g. *Leave everything else default*
5. Launch the job
6. Open the MATLAB window in OOD and look around a bit!

Post-Workshop Exercise

Run these commands

```
module load gnu openmpi
cp /gpfs/research/software/examples/trap* .
mpicc trap.c -o trap.x -lm
```

This will not return anything but if you list the files on your home directory, you will see a new file.

```
ls
```

The above command will show several files including:

`trap.c trap.sh trap.x`

Post-Workshop Exercise

Now run these commands

```
module load gnu openmpi
sbatch trap.sh
```

This will return a job ID number (number will change each job)

```
Submitted batch job 123456
```

When the job is complete, a new file fill be present called:

- ***Slurm-123456.out*** *(again, the number will change from job to job)*
- *The results from your job will be in there.*

Survey

https://fsu.qualtrics.com/jfe/form/SV_3arW54ZInjjRg1g

