

Ptolemy: High Performance Computing Cluster

Getting Started Guide

Anup Zope

Assistant Research Professor

Michael Navicky

Director

High Performance Computing Collaboratory
Mississippi State University

September 26, 2024

Overview

- 1 Introduction to Ptolemy
- 2 Access and Login Management
- 3 Connection to Ptolemy
- 4 Data Management
- 5 Computations on Ptolemy

Outline

- 1 Introduction to Ptolemy
- 2 Access and Login Management
- 3 Connection to Ptolemy
- 4 Data Management
- 5 Computations on Ptolemy

Introduction to Ptolemy

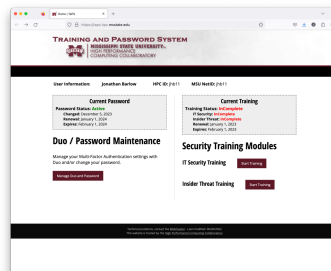
- Ptolemy is a supercomputer broadly available to MSU students and researchers.
- The cluster consists of
 - 1024 processor cores, 8 Terabytes (TB) of RAM
 - 64 NVIDIA A100 GPGPUs (80 GB RAM per GPU)
 - NVIDIA/Mellanox HDR200 InfiniBand Network (200Gb/s)
- Funded by
 - Offices of the President, Provost and Executive Vice President
 - Office of Research and Economic Development
 - The Division of Agriculture, Forestry and Veterinary Medicine
 - The Division of Finance and Administration

Outline

- 1 Introduction to Ptolemy
- 2 Access and Login Management**
- 3 Connection to Ptolemy
- 4 Data Management
- 5 Computations on Ptolemy

How to Get Access to Ptolemy?

- Email HelpDesk (help@hpc.msstate.edu) with a description of the purpose and compute requirements.
- When you receive an invitation email, click on the link in the email to fill out requested information, complete training.
- Wait to hear that your account has been created.
- Follow the instruction in the email and setup Ptolemy credentials with 2FA.



- Class instructors who plan to use the cluster for class projects need to setup class accounts by submitting helpdesk ticket.

Outline

- 1 Introduction to Ptolemy
- 2 Access and Login Management
- 3 Connection to Ptolemy**
- 4 Data Management
- 5 Computations on Ptolemy

How to Access Ptolemy?

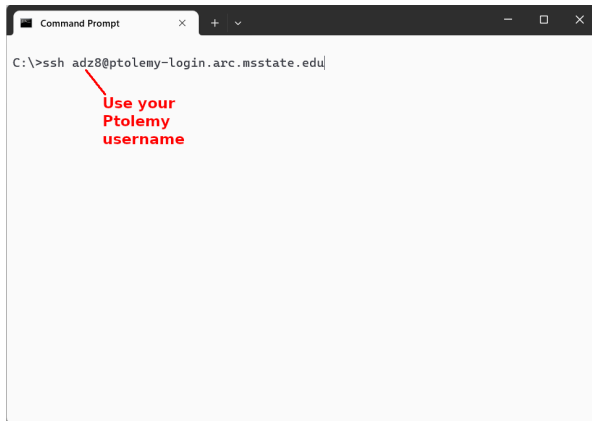
- Note: Most up to date documentation on Ptolemy can be found at <https://www.hpc.msstate.edu/computing/ptolemy>.
- What is required?
 - Active Ptolemy Account
 - Smartphone with activated DuoMobile 2FA
 - Laptop/desktop (a.k.a. client) with Windows/Linux/MacOS
 - Internet connection
- What are ways to connect to Ptolemy:
 - via Secure Shell a.k.a. SSH
 - via Open OnDemand

Connect to Ptolemy via Secure Shell (SSH)

- What is SSH?
 - A network protocol that allows users to securely access remote computers such as the Ptolemy cluster.
- This method provides a command line interface to Ptolemy. You need to be familiar with Linux bash commands to effectively use this method.
- You need an SSH client installed on your device.
 - On Windows, several options are available:
 - Windows Terminal (included in the OS)
 - Windows Subsystem for Linux (WSL)
 - Putty (<https://www.putty.org/>)
 - Cygwin (<https://www.cygwin.com/>)
 - MSYS2 (<https://www.msys2.org/>)
 - On Linux and MacOS: Terminal app included in the OS

Connect to Ptolemy via Secure Shell (SSH): Terminal

- Enter command `ssh`
`<ptolemy-user-name>@ptolemy-login.arc.msstate.edu` and hit Enter.



A screenshot of a Windows Command Prompt window. The title bar reads "Command Prompt". The command prompt shows the command `C:\>ssh adz8@ptolemy-login.arc.msstate.edu|`. A red arrow points from the text "Use your Ptolemy username" to the username "adz8" in the command.

```
C:\>ssh adz8@ptolemy-login.arc.msstate.edu|
```

Use your
Ptolemy
username

Connect to Ptolemy via Secure Shell (SSH): Terminal

- Enter password and hit Enter.

```

Command Prompt - ssh adz8

University (MSU). It is for authorized use only. By using this system, all
users acknowledge notice of and agree to comply with all MSU and High
Performance Computing Collaboratory (HPC2) policies governing use of
information systems.

Any use of this system and all files on this system may be intercepted,
monitored, recorded, copied, audited, inspected, and disclosed to authorized
university and law enforcement personnel, as well as authorized individuals
of
other organizations. By using this system, the user consents to such
interception, monitoring, recording, and disclosure at the discretion of au
disclosure at the discretion of au
Unauthorized, improper or negligent use of this system may result in
administrative disciplinary action, civil
charges, criminal penalties, and/or
ble
law, MSU policies, HPC2 policies, and Federal agencies.
***** N O T I C E *****
(adz8@ptolemy-login.arc.msstate.edu) Password: |
  
```

Enter your password and hit Enter. There is no visible sign that you are entering the password.

Connect to Ptolemy via Secure Shell (SSH): Terminal

- Enter Duo passcode and hit Enter or enter 1 for Duo push notification.

```

Command Prompt - ssh adz8 x + v
Any use of this system and all files on this system may be intercepted,
monitored, recorded, copied, audited, inspected, and disclosed to authorized
university and law enforcement personnel, as well as authorized individuals
of
other organizations. By using this system, the user consents to such
interception, monitoring, recording, copying, auditing, inspection and
disclosure at the discretion of authorized university personnel.

Unauthorized, improper or negligent use of this system may result in
administrative disciplinary action, up to and including termination, civil
charges, criminal penalties, and/or other sanctions as determined by applica
ble
law, MSU policies, HPC2 policies, law enforcement or other authorized State
and Federal agencies.

***** N O T I C E *****

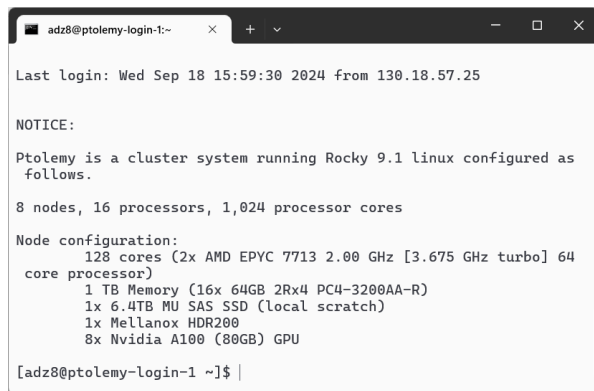
(adz8@ptolemy-login.arc.msstate.edu) P
(adz8@ptolemy-login.arc.msstate.edu) E adz8

Enter a passcode or select one of the
1. Duo Push to XXX-XXX-9524
Passcode or option (1-1): |
  
```

Enter Duo Passcode
or enter 1 and hit
Enter for a Duo
push.

Connect to Ptolemy via Secure Shell (SSH): Terminal

- After successful login, you will see following screen.
- Use Linux bash shell commands to create/copy/move files, submit and monitor compute jobs etc.



```

adz8@ptolemy-login-1:~
Last login: Wed Sep 18 15:59:30 2024 from 130.18.57.25

NOTICE:

Ptolemy is a cluster system running Rocky 9.1 linux configured as
follows.

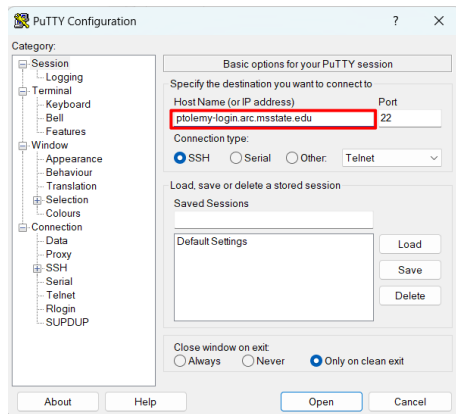
8 nodes, 16 processors, 1,024 processor cores

Node configuration:
  128 cores (2x AMD EPYC 7713 2.00 GHz [3.675 GHz turbo] 64
  core processor)
  1 TB Memory (16x 64GB 2Rx4 PC4-3200AA-R)
  1x 6.4TB MU SAS SSD (local scratch)
  1x Mellanox HDR200
  8x Nvidia A100 (80GB) GPU

[adz8@ptolemy-login-1 ~]$
  
```

Connect to Ptolemy via Secure Shell (SSH): PuTTY

- Download and install PuTTY.
- Start PuTTY client.



Connect to Ptolemy via Secure Shell (SSH): PuTTY

- Enter username, password, and Duo passcode or 1 for Duo push.

```

ptolemy-login.arc.mstate.edu - PuTTY
login as: adxs
Pre-authentication banner message from server:
|
| ***** NOTICE *****
|
| This system is under the control of and/or the property of Mississippi State
| University (MSU). It is for authorized use only. By using this system, all
| users acknowledge notice of and agree to comply with all MSU and High
| Performance Computing Collaboratory (HPC2) policies governing use of
| information systems.
|
| Any use of this system and all files on this system may be intercepted,
| monitored, recorded, copied, audited, inspected, and disclosed to authorized
| university and law enforcement personnel, as well as authorized individuals o
| f
| other organizations. By using this system, the user consents to such
| interception, monitoring, recording, copying, auditing, inspection and
| disclosure at the discretion of authorized university personnel.
|
| Unauthorized, improper or negligent use of this system may result in
| administrative disciplinary action, up to and including termination, civil
| charges, criminal penalties, and/or other sanctions as determined by applicab
| le
| law, MSU policies, HPC2 policies, law enforcement or other authorized State
| and Federal agencies.
|
| ***** NOTICE *****
|
| End of banner message from server
| Keyboard-interactive authentication prompts from server:
| Password:
| Duo two-factor login for adxs
|
| Enter a passcode or select one of the following options:
|
| 1. Duo Push to XXX-XXX-9524
|
| Passcode or option (1-1): 1
  
```

Connect to Ptolemy via Secure Shell (SSH): PuTTY

- After successful login:

```

adzs@ptolemy-login-1:~
|
| End of banner message from server
| Keyboard-interactive authentication prompts from server:
| Password:
| Duo two-factor login for adzs
|
| Enter a passcode or select one of the following options:
|
| 1. Duo Push to XXX-XXX-9524
|
| Passcode or option (1-1): 1
| End of keyboard-interactive prompts from server
| Pre-authentication banner message from server:
| Success. Logging you in...
| End of banner message from server
Success. Logging you in...

NOTICE:

Ptolemy is a cluster system running Rocky 9.1 linux configured as follows.

8 nodes, 16 processors, 1,024 processor cores

Node configuration:
  128 cores (2x AMD EPYC 7713 2.00 GHz [3.675 GHz turbo] 64 core processor)
  1 TB Memory (16x 64GB 2Rx4 PC4-3200AA-R)
  1x 6.4TB M0 SAS SSD (local scratch)
  1x Mellanox HDR200
  8x Nvidia A100 (80GB) GPU

Last login: Wed Sep 25 18:04:19 2024 from 172.59.90.56

NOTICE:

Ptolemy is a cluster system running Rocky 9.1 linux configured as follows.

8 nodes, 16 processors, 1,024 processor cores

Node configuration:
  128 cores (2x AMD EPYC 7713 2.00 GHz [3.675 GHz turbo] 64 core processor)
  1 TB Memory (16x 64GB 2Rx4 PC4-3200AA-R)
  1x 6.4TB M0 SAS SSD (local scratch)
  1x Mellanox HDR200
  8x Nvidia A100 (80GB) GPU

adzs@ptolemy-login-1:~$
  
```


Some Linux Bash Commands

- `pwd`: Prints present working directory path.
- `ls <path-to-directory>`: List directory contents.
- `mkdir <directory-path>`: Create directory.
- `rmdir <directory-path>`: Deletes directory.
- `cd <directory-path>`: Change to the specified directory.
- `touch <file-path>`: Creates a file with specified path.
- `mv <src> <dst>`: Moves/renames files/folders.
- `cp <src> <dst>`: Copies files/folders.
- `head <file-path>`: Displays first few lines in the specified file.
- `tail <file-path>`: Displays last few lines in the specified file.
- `chmod`: Change permissions.
- `exit`: Logout and exit the terminal window.
- `top`: Command-line Task Manager.

Some Utility Commands and Tricks

● Utility Commands

- `vi`, `emacs`, `nano`: Command line text file editors.
- `awk`: Command line programmable manipulation of text-based data files, e.g. Excel-like calculations on columnar data.
- `sed`: Stream editor. text-file transformations, e.g. find-replace.
- `grep`: Search for files/folders with specified pattern in the name or in the content.

● Tricks

- Use `<TAB>` for autocompletion.
- Use `<Up Arrow>` and `<Down Arrow>` keys to browse through command history.
- Use `man <command-name>` or `info <command-name>` or `<command-name> --help` to seek help on the specified command.

Software Stack on Ptolemy

- The software stack on Ptolemy can be queried from command line using module avail.
- Use module load <software-name> to load the software for use.

- These software products can be used via their command line interface (CLI). See the respective product documentation.
- Many compilers and libraries are available which you can use to compile your code or open source software.

```

freebayes/1.3.6      npich/4.1.1          singularity/3.8.7
gcc/12.2.0           muscle/3.8.1551      sparsehash/2.0.4
gdb/12.1             nand/2.13            su2/7.3.1
gdb/12.1             nccmp/1.9.0.1        subread/2.0.2
git-lfs/3.1.2        ncl/6.0.2            subversion/1.14.1
glx/1.4              nco/5.0.1            tassal/5.2.86
gmt/6.2.0            ncurses/6.3          texlive/20220321
grace/5.1.25         netcdf-c/4.9.0       ucx/1.13.1
grads/2.2.1          netcdf-cxx/4.2       vmd/1.9.3
graphviz/2.49.0      netcdf-fortran/4.6.0 w3mco/2.4.1
gs/12.7.1            nmap/7.92            wgr1b2/3.1.1
hdf5/1.12.2          nvhpc/22.9           zlib/1.2.13

-----
contrib/0.1          julia/1.9.3           /apps/other/modulefiles
freerdp/2.9.0        lammps/82AUG2023      ont-guppy/6.5.7
                                quantum-espresso-gpu/develop

-----
ansys/2022.2         fieldview/21           /apps/licensed/modulefiles
console/6.0          gaussian/16-C.02       naple/2023.1
converge/3.1.7       harris/5.6.2           mathematics/13.1.0  pointwise/22.1      tecplot/2022r1
cubit/2022.11        hyperworks/2022.1     matlab/2022b         pointwise/22.2 (D)  thermocalc/2022b
                                netashape/1.8.4      qchem/6.0
                                simulia/2023

Where:
0: Default Module

[ad28@ptolemy-login-1 ~]$

```

Connect to Ptolemy via Open OnDemand

- What is Open OnDemand? A browser-based High Performance Computing portal.
- URL: `https://ptolemy-ood.arc.msstate.edu`
- Use ARC credentials to login.

MISSISSIPPI STATE UNIVERSITY
HIGH PERFORMANCE
COMPUTING COLLABORATORY

Open OnDemand

Enter HPC2 Username & Password

Username*

Password*

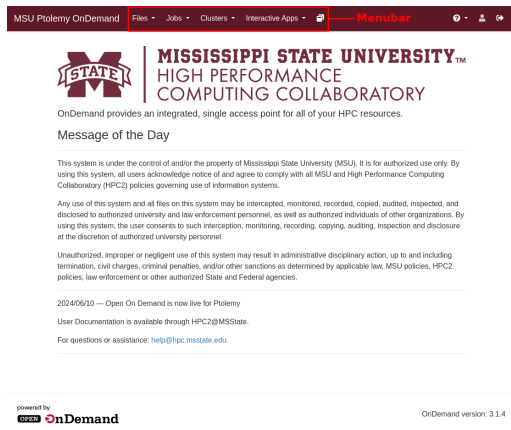
LOGIN

For security reasons, please [log out](#) and exit your web browser when you are done accessing services that require authentication.


High Performance Computing Collaboratory HPC2

Connect to Ptolemy via Open OnDemand

- Main dashboard.



MSU Ptolemy OnDemand | Files ▾ Jobs ▾ Clusters ▾ Interactive Apps ▾ | Menubar ⓘ 👤 ⚙

 **MISSISSIPPI STATE UNIVERSITY™**
HIGH PERFORMANCE
COMPUTING COLLABORATORY

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

Message of the Day

This system is under the control of and/or the property of Mississippi State University (MSU). It is for authorized use only. By using this system, all users acknowledge notice of and agree to comply with all MSU and High Performance Computing Collaboratory (HPC2) policies governing use of information systems.


Any use of this system and all files on this system may be intercepted, monitored, recorded, copied, audited, inspected, and disclosed to authorized university and law enforcement personnel, as well as authorized individuals of other organizations. By using this system, the user consents to such interception, monitoring, recording, copying, auditing, inspection and disclosure at the discretion of authorized university personnel.

Unauthorized, improper or negligent use of this system may result in administrative disciplinary action, up to and including termination, civil charges, criminal penalties, and/or other sanctions as determined by applicable law, MSU policies, HPC2 policies, law enforcement or other authorized State and Federal agencies.

2024/06/10 — Open On Demand is now live for Ptolemy

User Documentation is available through HPC2@MSState.

For questions or assistance: help@hpc.msstate.edu

powered by  OnDemand

OnDemand version: 3.1.4

Connect to Ptolemy via Open OnDemand

- Can be used to browse folders, download/upload/copy/move data.
- Only for data **less than 1 GB in size**.

The screenshot displays the MSU Ptolemy OnDemand web interface. The top navigation bar includes links for Files, Jobs, Clusters, and Interactive Apps. A dropdown menu is open, showing the Home Directory structure: /work, /scratch, and /reference. The main content area shows the current directory as /home / ad28 /. Below the directory path, there are checkboxes for 'Show Owner/Mode' and 'Show Dattiles', and a 'Filter' input field. A table lists 14 files and directories, showing their Type, Name, Size, and Modified date. The files listed are: cpu, Desktop, MiGAllocationTest, ondemand, PtolemyTests, raytracinginoneweekend, raytracinginoneweekendincuda, sample, temp, and test-skirm-allocation.

Type	Name	Size	Modified at
Folder	cpu	-	5/3/2024 1:58:26 PM
Folder	Desktop	-	9/17/2024 2:24:20 PM
Folder	MiGAllocationTest	-	7/31/2024 11:12:22 AM
Folder	ondemand	-	9/17/2024 2:22:58 PM
Folder	PtolemyTests	-	7/31/2024 11:12:22 AM
Folder	raytracinginoneweekend	-	6/6/2024 2:18:05 PM
Folder	raytracinginoneweekendincuda	-	5/3/2024 3:52:08 PM
Folder	sample	-	5/3/2024 4:10:17 PM
Folder	temp	-	5/8/2024 9:26:52 AM
Folder	test-skirm-allocation	-	5/3/2024 2:34:23 PM

Connect to Ptolemy via Open OnDemand

- Can be used to monitor cluster jobs.

The screenshot displays the MSU Ptolemy OnDemand web interface. At the top, there is a navigation bar with links for Files, Jobs, Clusters, and Interactive Apps. A dropdown menu for 'Jobs' is open, showing 'Active Jobs' and 'Job Composer'. Below the navigation bar, the 'Active Jobs' section is visible, featuring a 'Show 50 entries' dropdown and a 'Filter' input field. A table lists four active jobs, each with columns for ID, Name, User, Account, Time Used, Queue, Status, Cluster, and Actions. The jobs are all in a 'Running' status. At the bottom of the table, it says 'Showing 1 to 4 of 4 entries' with 'Previous' and 'Next' navigation links. The footer of the interface includes the 'powered by' logo for Open OnDemand and the version number '3.1.4'.

ID	Name	User	Account	Time Used	Queue	Status	Cluster	Actions
27556	sys/dashboard/sys/jupyter	adz8	test	00:10:23	ptolemy	Running	Ptolemy-login-1	[Stop]
27556	sys/dashboard/sys/jupyter	adz8	test	00:10:23	ptolemy	Running	Ptolemy-login-2	[Stop]
27556	sys/dashboard/sys/jupyter	adz8	test	00:10:23	ptolemy	Running	Ptolemy-login-3	[Stop]
27556	sys/dashboard/sys/jupyter	adz8	test	00:10:23	ptolemy	Running	Ptolemy	[Stop]

Connect to Ptolemy via Open OnDemand

- Can be used to obtain shell access.

The screenshot displays the MSU Ptolemy OnDemand web interface. At the top, there is a navigation bar with links for Files, Jobs, Clusters, and Interactive Apps. Below this, a dropdown menu is open, showing options for shell access: >_ Ptolemy Shell Access, >_ Ptolemy-login-1 Shell Access, >_ Ptolemy-login-2 Shell Access, and >_ Ptolemy-login-3 Shell Access. The main content area is titled 'Active Jobs' and shows a table with columns for ID, Name, User, Account, Time Used, Queue, Status, Cluster, and Actions. The table is currently empty, displaying 'No data available in table'. Below the table, it says 'Showing 0 to 0 of 0 entries'. To the right of the table, there are buttons for 'Your Jobs' and 'All Clusters', and a search filter input. At the bottom of the interface, there is a terminal window showing a shell prompt and system information. The terminal output includes a notice about the Ptolemy cluster system and its configuration: 8 nodes, 16 processors, 1,024 processor cores. Node configuration: 128 cores (2x AMD EPYC 7713 2.00 GHz [3.675 GHz turbo] 64 core processor), 1 TB Memory (16x 64GB 2Rx4 PC4-3200AA-R), 1x 6.4TB HLU SAS SSD (local scratch), 1x Mellanox HDR28B, 8x Nvidia A100 (80GB) GPU. The terminal prompt is [adz8@ptolemy-login-1 ~]\$.

MSU Ptolemy OnDemand Files Jobs Clusters Interactive Apps

Active Jobs

Show 50 entries

Filter:

Your Jobs All Clusters

ID Name User Account Time Used Queue Status Cluster Actions

No data available in table

Showing 0 to 0 of 0 entries

Previous Next

Terminal Window:

```

Host: ptolemy-login-1.arc.mstate.edu
Theme: Default

NOTICE:
Ptolemy is a cluster system running Rocky 9.1 linux configured as follows.
8 nodes, 16 processors, 1,024 processor cores

Node configuration:
128 cores (2x AMD EPYC 7713 2.00 GHz [3.675 GHz turbo] 64 core processor)
1 TB Memory (16x 64GB 2Rx4 PC4-3200AA-R)
1x 6.4TB HLU SAS SSD (local scratch)
1x Mellanox HDR28B
8x Nvidia A100 (80GB) GPU

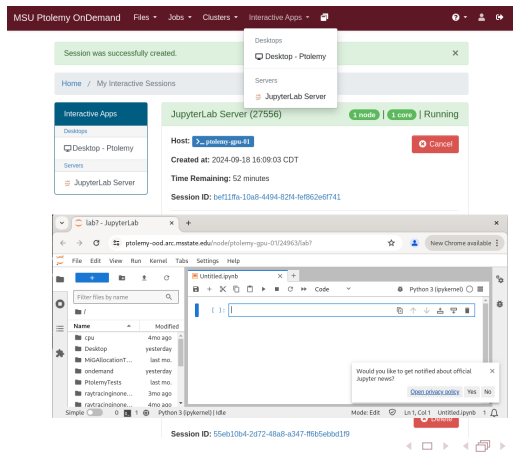
[adz8@ptolemy-login-1 ~]$
    
```

powered by OPEN OnDemand

OnDemand version: 3.1.4

Connect to Ptolemy via Open OnDemand

- Can be used to launch interactive applications.
- To request additional software, submit HelpDesk ticket at help@hpc.msstate.edu



Outline

- 1 Introduction to Ptolemy
- 2 Access and Login Management
- 3 Connection to Ptolemy
- 4 Data Management**
- 5 Computations on Ptolemy

Storage Space

- Home Folder
 - Each user has a home folder `/home/<user-name>` (5GB quota).
- Scratch and Work Folder
 - Due to limited `$HOME` space, it is not suitable for parallel computations that read or write large data.
 - Instead, use
 - `/scratch/ptolemy/users/<user-name>` for temporary transient data.
 - `/work/ptolemy/users/<user-name>` for user specific data
 - `/work/ptolemy/projects/<project-name>` for project specific data
 - `/work/ptolemy/reference` for reference data sets
 - See **`/work/README`** and **`/scratch/README`** for detailed info.
 - `/scratch` is large, but older content is purged.
 - `/work` is not purged, but has limited disk quota. Submit a HelpDesk ticket for getting allocation here.
 - None of this data is backed up. So always remember to backup important files on your device.

Transferring Data Between Your Device and Ptolemy

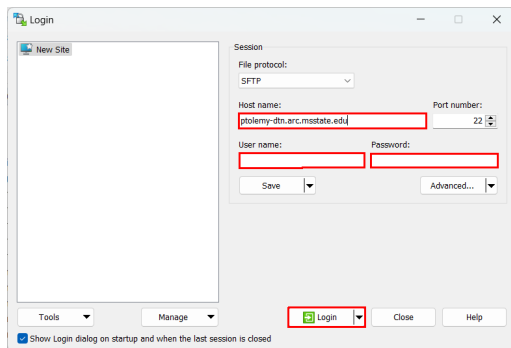
- Using Ptolemy Open OnDemand
- Use this method **only for data less than 1 GB.**

The screenshot shows the MSU Ptolemy Open OnDemand web interface. The top navigation bar includes links for Files, Jobs, Clusters, and Interactive Apps. A dropdown menu is open, showing the Home Directory structure: /work, /scratch, and /reference. The main content area displays a file listing for the directory /home/ad28/. The listing shows various files and directories, including cpu, Desktop, MiGAllocationTest, ondemand, PtolemyTests, raytracinginoneweekend, raytracinginoneweekendincuda, sample, temp, and test-skirm-allocation. The interface also includes a search bar, a filter dropdown, and a table of file details.

Type	Name	Size	Modified at
Folder	cpu	-	5/3/2024 1:58:26 PM
Folder	Desktop	-	9/17/2024 2:24:20 PM
Folder	MiGAllocationTest	-	7/31/2024 11:12:22 AM
Folder	ondemand	-	9/17/2024 2:22:58 PM
Folder	PtolemyTests	-	7/31/2024 11:12:22 AM
Folder	raytracinginoneweekend	-	6/6/2024 2:18:05 PM
Folder	raytracinginoneweekendincuda	-	5/3/2024 3:52:08 PM
Folder	sample	-	5/3/2024 4:10:17 PM
Folder	temp	-	5/8/2024 9:26:52 AM
Folder	test-skirm-allocation	-	5/3/2024 2:34:23 PM

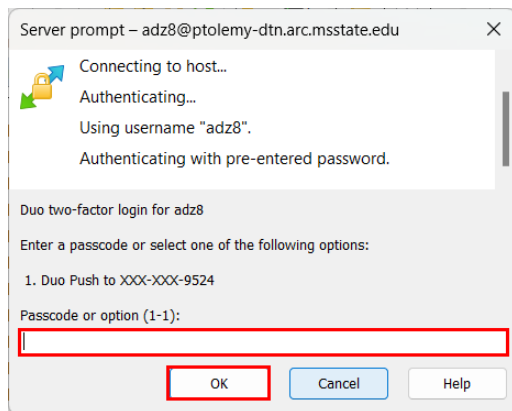
Transferring Data Between Your Device and Ptolemy

- Using SCP client such as WinSCP on Windows.



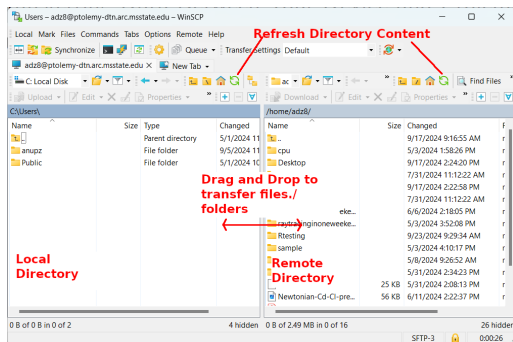
Transferring Data Between Your Device and Ptolemy

- Using SCP client such as WinSCP on Windows.



Transferring Data Between Your Device and Ptolemy

- Using SCP client such as WinSCP on Windows.



Transferring Data Between Your Device and Ptolemy

Using Globus Online

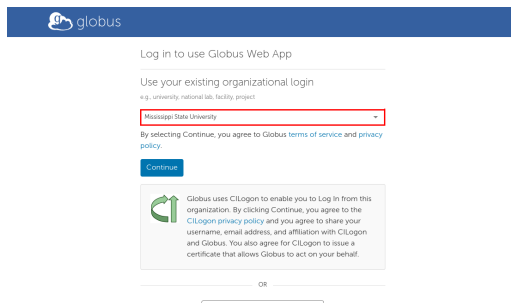
- URL: <https://www.globus.org/>



Transferring Data Between Your Device and Ptolemy

Using Globus Online

- Login using MSU NetID and password (Central Auth Service credentials), and 2FA.



globus

Log in to use Globus Web App

Use your existing organizational login
e.g., university, national lab, facility, project

Mississippi State University

By selecting Continue, you agree to Globus [terms of service](#) and [privacy policy](#).

Continue

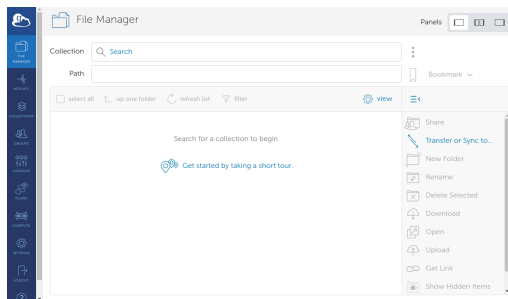
Globus uses CILogon to enable you to Log in from this organization. By clicking Continue, you agree to the [CILogon privacy policy](#) and you agree to share your username, email address, and affiliation with CILogon and Globus. You also agree for CILogon to issue a certificate that allows Globus to act on your behalf.

OR

Transferring Data Between Your Device and Ptolemy

Using Globus Online

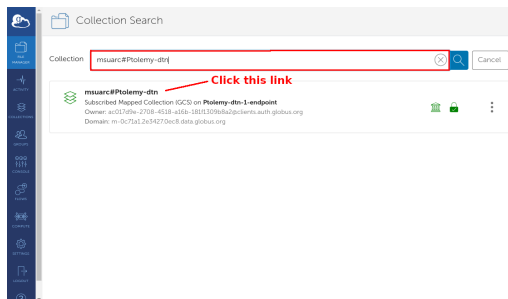
- File Manager Dashboard:



Transferring Data Between Your Device and Ptolemy

Using Globus Online

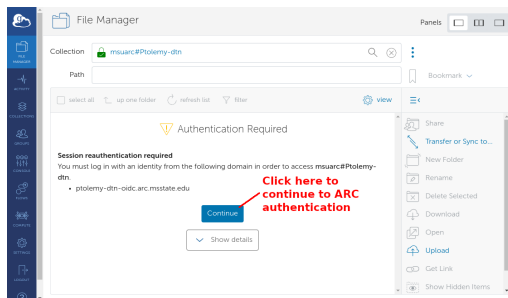
- Search for collection (a.k.a. end-point): msuarc#Ptolemy-dtn
- Click the link to the collection



Transferring Data Between Your Device and Ptolemy

Using Globus Online

- It asks for authentication to access the end-point.



Transferring Data Between Your Device and Ptolemy

Using Globus Online

- Use ARC credentials.



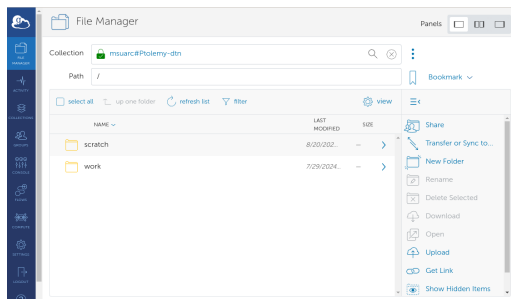
**Use Ptolemy
username and
password.**

[Login Issues? Contact Administrator](#)

Transferring Data Between Your Device and Ptolemy

Using Globus Online

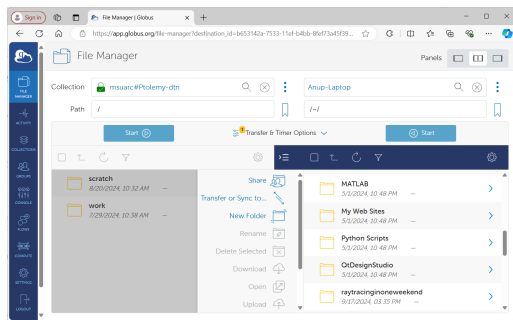
- Once connected to the end-point, folders can be browsed, data can be uploaded/downloaded.



Transferring Data Between Your Device and Ptolemy

Using Globus Online

- You can setup data-sync between Ptolemy and your device.
- Needs installation and configuration of Globus Personal Connect on your device: <https://www.globus.org/globus-connect-personal>.



Transferring Data Between Your Device and Ptolemy

Using SSH bash command line interface.

- Use scp or rsync
- Linux/MacOS: Use Terminal
- Windows: Use WSL/Cygwin/MSYS2 Terminal
- Syntax:
 - `scp -r <source> <destination>`
 - `rsync -avE <source> <destination>`
- Transfer from your workstation to Ptolemy
 - `<source>` is your workstation folder
 - `<destination>` is `<uname>@ptolemy-dtn.arc.msstate.edu:<scratch-folder>`
- Transfer from Ptolemy to your workstation
 - `<source>` is `<uname>@ptolemy-dtn.arc.msstate.edu:<scratch-folder>`
 - `<destination>` is your workstation folder

Notice that the role of `<source>` and `<destination>` has changed.

Outline

1 Introduction to Ptolemy

2 Access and Login Management

3 Connection to Ptolemy

4 Data Management

5 Computations on Ptolemy

Types of Nodes on Ptolemy

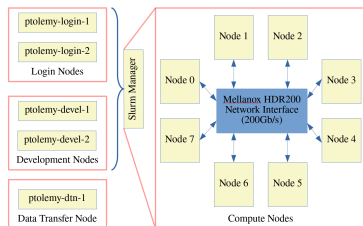


Figure: Types of nodes on Ptolemy

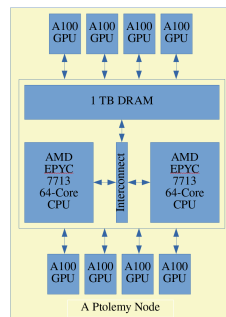


Figure: Schematic of a Compute Node

- **NEVER** run jobs on the login nodes - this is **PARAMOUNT**. Users that abuse this rule will have access to the cluster removed.

So, How to Run a Software on Ptolemy?

- Use Slurm to submit a compute job to the cluster.
- Slurm manages and allocates compute resources required for a compute job.
- There are two primary methods to run a software on Ptolemy are:
 - via Open OnDemand
 - via command line interface using Slurm
- Special cases:
 - Users can run SHORT DEVELOPMENT jobs on the devel nodes. But they must not monopolize the node's resources, and runs should be kept under 30 mins as a general rule.
 - Jobs are allowed on data transfer nodes via Slurm 'service' partition. Still requires Slurm, but allows automated data transfer.

Using Open OnDemand

Choose Slurm parameters:

The screenshot shows the 'JupyterLab Server' configuration page. It includes fields for Account, Partition, QOS, Number of hours, Number of nodes, and Number of tasks. There is also a section for 'Additional Slurm Parameters' and a 'Launch' button.

Account (-account=) test

Partition (-partition=) ptolemy

QOS (-qos=) normal

Number of hours (-time=) 1

Number of nodes (-nodes=) 1

Number of tasks (-ntasks=) 1

Additional Slurm Parameters

Parameters such as --exclude, --reservation, --time, or --reservation can be specified here.

☐ I would like to receive an email when my job is near the time limit.

Working Directory

\$HOME

(default is \$HOME)

Launch

*The JupyterLab Server session data for this session can be accessed under the [data root directory](#).

Launch:

The screenshot shows the Open OnDemand interface after a successful session launch. A notification 'Session was successfully created.' is displayed. The 'JupyterLab Server (27556)' session is shown as 'Running' with 1 node and 1 core. The 'Host' is 'ptolemy-gpu-41', 'Created at' is '2024-09-18 16:09:03 CDT', 'Time Remaining' is '52 minutes', and 'Session ID' is 'baf11f7a-10a8-4494-8294-b6862e6f741'.

Below the notification, the 'Interactive Apps' section shows 'Desktops' and 'Servers'. The 'Desktops' section includes 'Desktop - Ptolemy' and 'JupyterLab Server'.

The 'JupyterLab' window is open, showing a file explorer with a list of files and folders. A notification 'Would you like to get notified about official Jupyter news?' is displayed. The 'Session ID' is '55eb10b4-2d72-48a8-a347-8b65ebbd19'.

Using Command Line Slurm Submission

Two steps:

- Write a bash script for job submission

```
#!/bin/bash

#SBATCH --job-name=Test           # Name of the parallel job
#SBATCH --nodes=1                 # Number of nodes
#SBATCH --ntasks-per-node=1       # Number of cores per node
#SBATCH --partition=gpu-a100      # Name of the partition
#SBATCH --mem=10G                 # Amount of CPU memory required per node
#SBATCH --account=test            # Account number
#SBATCH --gres=gpu:a100_1g.10gb:1 # Requests one Nvidia A100 GPU with 10 GB of memory.
#SBATCH --time=1:00:00            # Maximum wall time
#SBATCH --output=stdout-%j.out    # Output file (stdout), %j is job allocation number
#SBATCH --error=stderr-%j.out     # Error file (stderr), %j is job allocation number
#SBATCH --mail-type=ALL           # All types all emails (BEGIN, END, FAIL, ALL)
#SBATCH --mail-user=<email-address> # Where to send the email

# Get a clean BASH environment
module purge

# Setup required software and BASH environment e.g.:
module load contrib cuda cuda-libs

# Command to start parallel program run, parallel job using
# nodes*ntasks-per-node CPU cores. e.g.:
srun ./convolutionFFT2D
```

- Use sbatch command to submit the job to Slurm

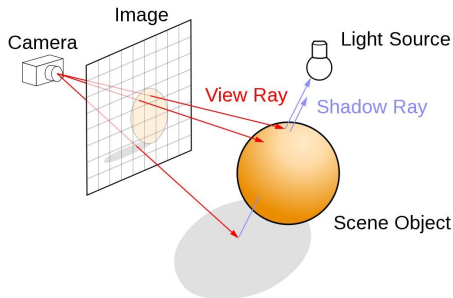
Slurm: Additional Commands

- To obtain the information on Slurm account (`-account` option) and QOS (`-qos` option):

```
sacctmgr show association where user=<user-name> \
format=cluster,account\%30,user,qos,maxjobs
```

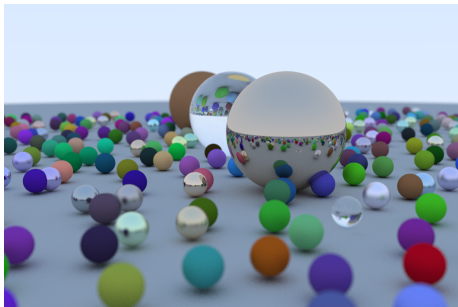
- Obtain information of partitions and nodes managed by Slurm.
`sinfo`
- To submit the parallel job, use the `sbatch` command
`sbatch <sbatch-script-file-name.sh>`
- To query the status of submitted job, use `squeue` command
`squeue -u <user-name>`
- To cancel a submitted job
`scancel <job-id>`

Ray Tracing Example - CPU vs GPGPU



- 1 Ray tracing is a technique to produce 2D images of 3D object model using the light transport mechanisms: absorption, reflection, refraction etc.
- 2 Calculation of the color of each pixel is an independent computation \implies embarrassingly parallel computation.

Ray Tracing Example - CPU vs GPGPU



```

└─ raytracinginoneweekend
    └─ cpu
        ├── camera.h
        ├── hitable.h
        ├── hitable_list.h
        ├── main.cc
        ├── Makefile
        ├── material.h
        ├── ray.h
        ├── README.md
        ├── sphere.h
        ├── submit.sh
        └── vec3.h
    └─ cuda
        ├── camera.h
        ├── hitable.h
        ├── hitable_list.h
        ├── main.cu
        ├── Makefile
        ├── material.h
        ├── ray.h
        ├── README.md
        ├── sphere.h
        ├── stderr.txt
        ├── stdout.txt
        ├── submit.sh
        └── vec3.h
  
```

¹Code taken from <https://github.com/rogerallen/raytracinginoneweekendincuda.git>

²Copy the code from `/scratch/ptolemy/users/adz8/raytracinginoneweekend`

Ray Tracing Example - CPU vs GPGPU

- Copy source code:

```
DIR=/scratch/ptolemy/users/<user-name>  
cp -R /scratch/ptolemy/users/adz8/raytracinginoneweekend ${DIR}
```

- Load software required to compile the code:

```
module load gcc cuda
```

- Compile CPU code and submit to cluster:

```
cd ${DIR}/raytracinginoneweekend/cpu  
make  
sbatch submit.sh
```

- Compile GPU code and submit to cluster:

```
cd ${DIR}/raytracinginoneweekend/cuda  
make  
sbatch submit.sh
```

Question: How much faster is the CUDA version?

Using OCI Containers

- OCI containers are standalone, executable package of a software that includes all the library dependencies needed to run the software.
- Where to find OCI containers?
 - Use following commands in SSH terminal to print the directory location where containers are stored:

```
module load apptainer  
echo $CONTAINERS
```
 - If not available in the \$CONTAINERS directory, submit a HelpDesk ticket (help@hpc.msstate.edu) to request the required container.
 - You can build your own container. Submit a HelpDesk ticket to get assistance.

Using OCI Containers

- Write batch submission file:

```
#!/bin/bash
```

```
# The usual SBATCH options
```

```
# Get a clean environment
```

```
module purge
```

```
# Load apptainer
```

```
module load apptainer
```

```
# Launch the container
```

```
srn apptainer exec ${CONTAINERS}/<container-file-name.sif> <exec-command>
```

- Use sbatch to submit

Questions?