# Interactive High-performance (HPC) Computing

COMPLECS Team

https://bit.ly/COMPLECS

https://github.com/sdsc-complecs/interactive-computing/

SDSC
SAN DIEGO SUPERCOMPUTER CENTER

UC San Diego

# About COMPLECS

COMPLECS (COMPrehensive Learning for end-users to Effectively utilize CyberinfraStructure) is a new SDSC program where training will cover non-programming skills needed to effectively use supercomputers. Topics include parallel computing concepts, Linux tools and bash scripting, security, batch computing, how to get help, data management and interactive computing.

**COMPLECS is supported by NSF award 2320934.**

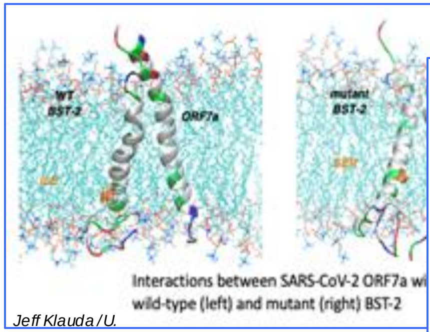
# Outline

- Defining Interactive HPC
  - High-performance computing (HPC)
  - HPC batch computing
  - Interactive computing
- Accessing Interactive HPC Nodes
  - Launching nodes
  - Running GUIs using X11 forwarding
- Interactive Application Examples
  - Viewing Data: unix file ops (grep, awk, cat), gnuplot, NetCDF
  - Programming & Visualization Platforms: Matlab, R, Jupyter Notebooks
  - Gateways & Portals: simplify access to interactive apps
- Q&A

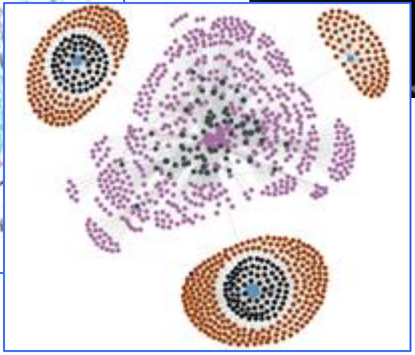
# You can run amazing jobs on supercomputers!

## Typically parallel or very large memory

*Pelagic fish communities Shapes (Jerome Guiet, UCLA)*

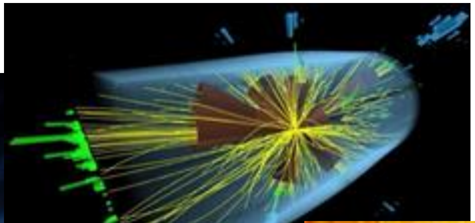

WT BST-2
ORF7a
mutant BST-2
Interactions between SARS-CoV-2 ORF7a wi
wild-type (left) and mutant (right) BST-2

*Jeff Klauda / U. Maryland*



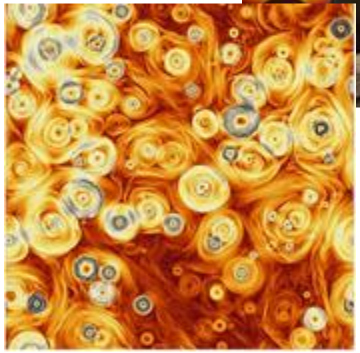
*Sample of Internet structure from CAIDA data (Mark Burgess, 12/16/21)*

*Cooper Downs, Predictive Science Inc*



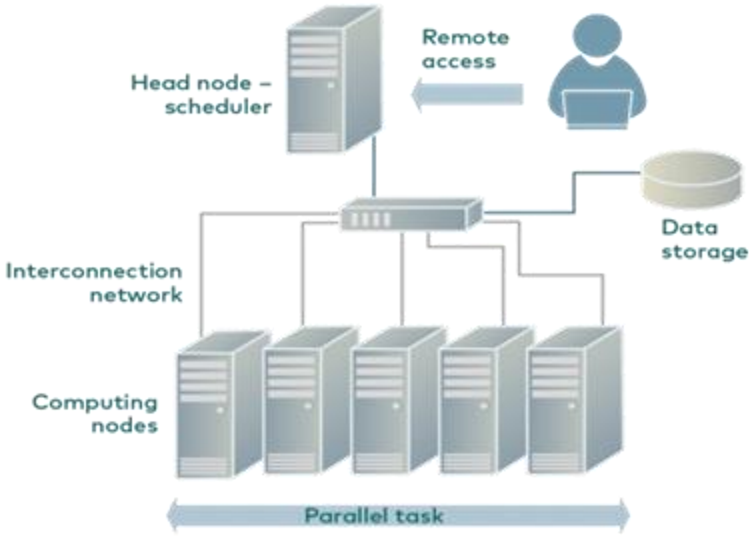
Compact Muon Solenoid (**CMS**)experiment at the LHC, CERN. *Image courtesy of CMS Collaboration; Mc Cauley, Thomas*

*Model of Both Inner and Outer Solar System*
*M. S.Clement (Carnegie Institution for Science)*



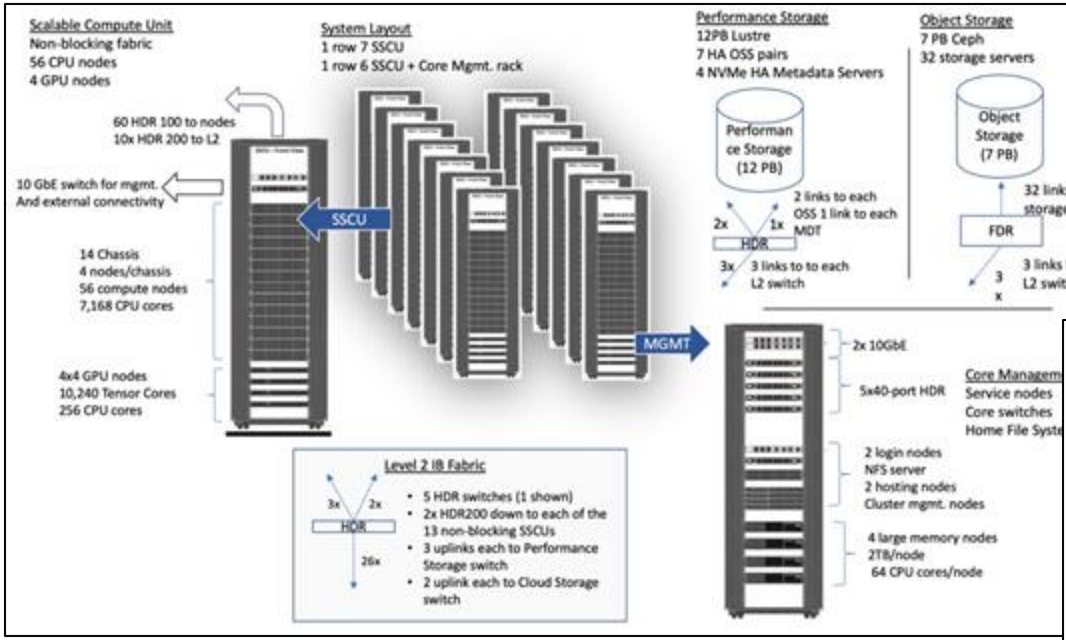
*Modeling the sun’s corona, Alfred Mallet (UC Berkeley)*

# However, you have to run your jobs on HPC clusters



- Systems powerful but complex
- Jobs can be run from:
  - Command line interface (CLI)
  - The batch queue system
  - Application clients (PyTorch)
- Job scaling:
  - Parallel: 1 core to 1000’s of nodes
- For large jobs, schedulers are needed to coordinate tasks

# HPC System Architecture: Expanse @ SDSC



Scalable Compute Unit
Non-blocking fabric
56 CPU nodes
4 GPU nodes

System Layout
1 row 7 SSCU
1 row 6 SSCU + Core Mgmt. rack

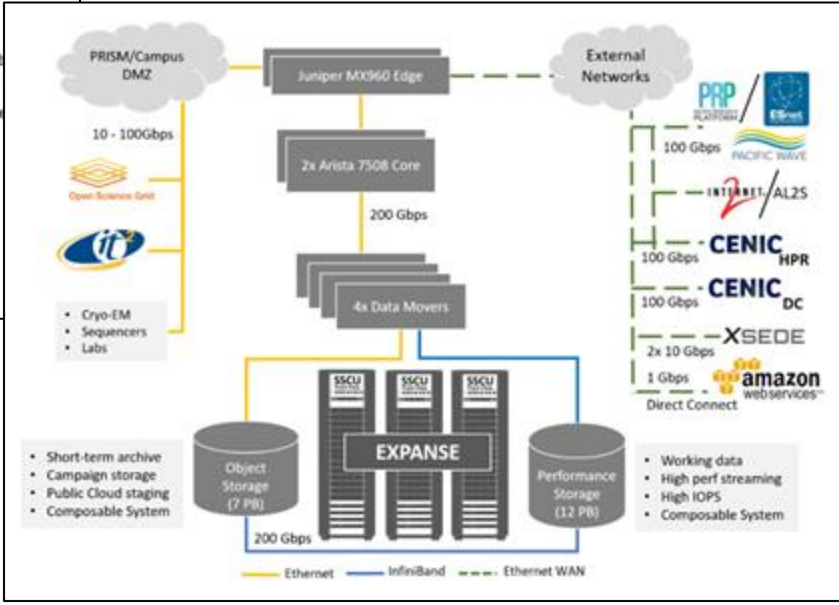
Performance Storage
12PB Lustre
7 HA OSS pairs
4 NVMe HA Metadata Servers

Object Storage
7 PB Ceph
32 storage servers

Core Managem
Service nodes
Core switches
Home File Syste

Level 2 IB Fabric

- 5 HDR switches (1 shown)
- 2x HDR200 down to each of the 13 non-blocking SSCUs
- 3 uplinks each to Performance Storage switch
- 2 uplink each to Cloud Storage switch



- Cryo-EM
- Sequencers
- Labs

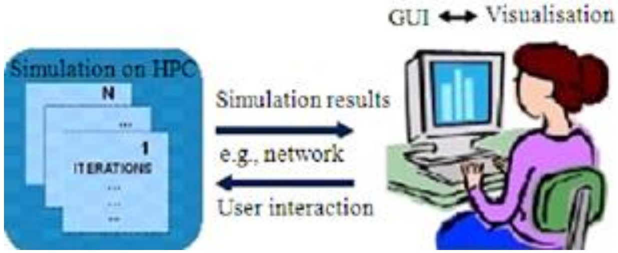
- Short-term archive
- Campaign storage
- Public Cloud staging
- Composable System

- Working data
- High perf streaming
- High IOPS
- Composable System

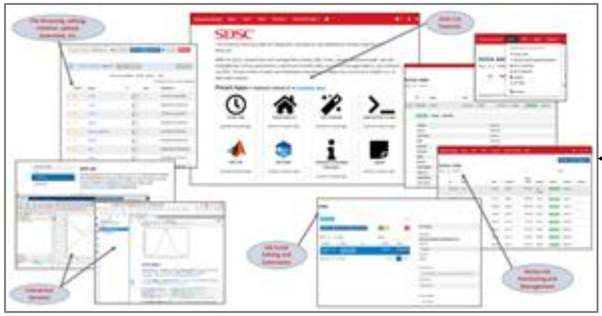
https://expanse.sdsc.edu
https://www.youtube.com/watch?v=uNZyg6X_t3s
https://access-ci.org

# HPC Jobs can be run as batch jobs (background) or interactively (real time)

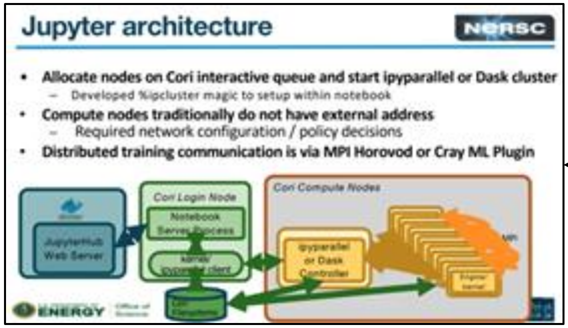
Terminal shell – submit batch



https://portal.expanse.edu



Interactive Distributed Computing with Jupyter

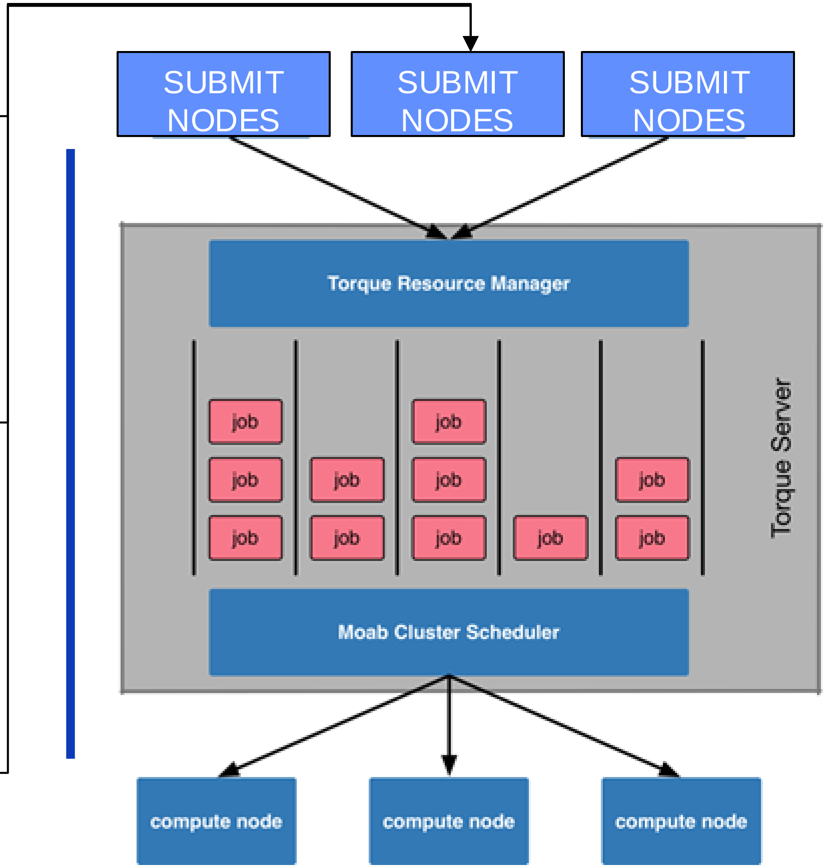


## Jupyter architecture

NERSC

- Allocate nodes on Cori interactive queue and start ipyparallel or Dask cluster
  - Developed %ipcluster magic to setup within notebook
- Compute nodes traditionally do not have external address
  - Required network configuration / policy decisions
- Distributed training communication is via MPI Horovod or Cray ML Plugin

JupyterHub Web Server
Cori Login Node
Notebook Server Process
kernel
ipyparallel client
Cori Compute Nodes
ipyparallel or Dask Controller
MPI
Engine Server
ENERGY
Office of Science



Src: https://hpc.dccn.nl/_images/torque_moab_arch.png

# Outline

- Defining Interactive HPC
  - High-performance computing (HPC)
  - HPC batch computing
  - Interactive computing
- Accessing Interactive HPC Nodes
  - Launching nodes
  - Running GUIs using X11 forwarding
- Interactive Application Examples
  - Viewing Data: unix file ops (grep, awk, cat), gnuplot, NetCDF
  - Programming & Visualization Platforms: Matlab, R, Jupyter Notebooks
  - Gateways & Portals: simplify access to interactive apps
- Q&A

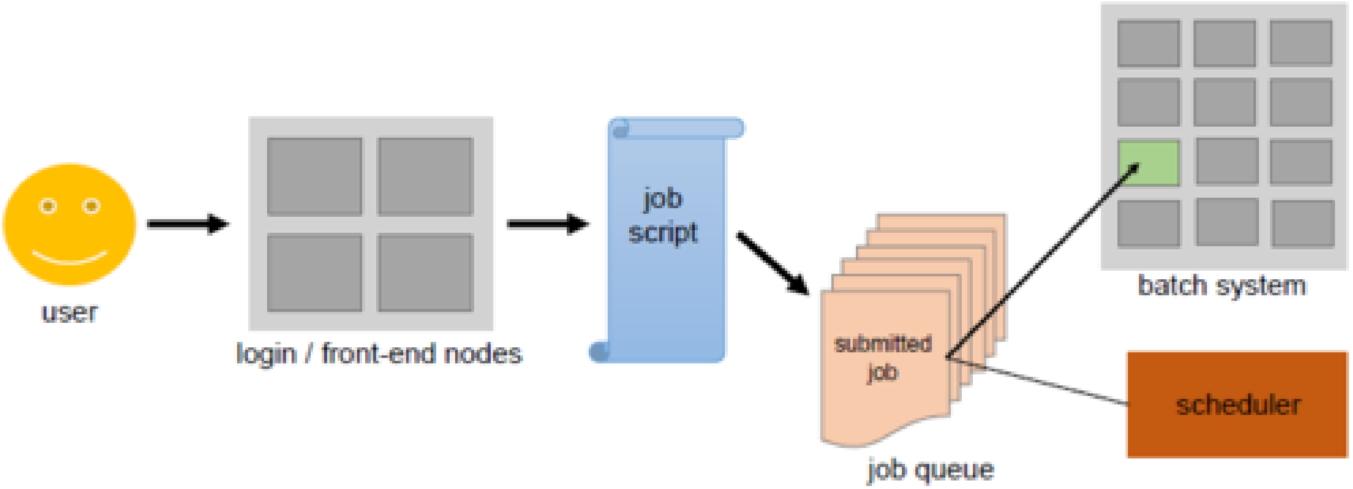
# What is a batch job scheduler & how are they used?

- Any HPC (or HTC) system — usually a cluster of machines — needs a means of sharing computational resources fairly between users; without this, there would be anarchy.
- Batch-queueing systems — usually abbreviated to simply *batch systems* — are intended to do this.
- All batch systems have at least these features:
    - *a scheduler* for allocating resources (CPUs!) to jobs and for prioritizing jobs;
    - *one or more queues* to which jobs are submitted
    - Job  partitions (or job queues): queue can be configured for a particular type of job (serial or parallel jobs; long or short jobs; or those requiring particularly high memory); members of a group or project).
- You need a basic understanding of how batch jobs work in order to run interactive jobs on HPC Clusters (ironic?)

# Batch Scheduler: main goals

- Minimize time between job submission and completion:
  - No job should stay in queue for extensive periods of time.
- Optimize CPU utilization:
  - Algorithms focus on minimizing CPU idle times.
- Maximize job throughput:
  - Manage as many jobs per time unit as possible.
- Support running jobs automatically in the background
- Finalize jobs: ensure job data and results are recorded & stored where you want them.

# Simple “Batch Scheduler” Architecture



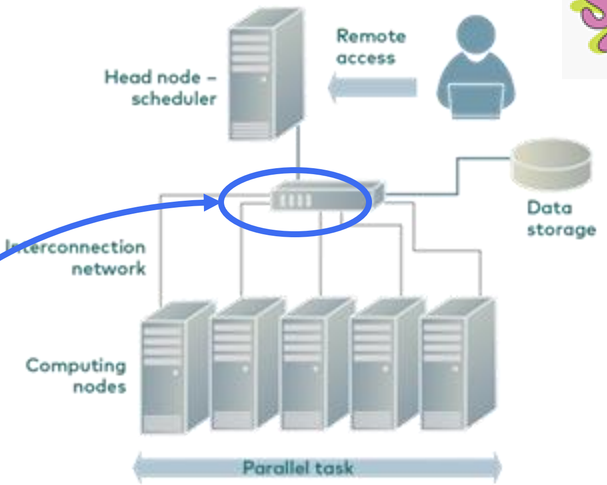
- Batch scheduler: software that implements a *batch system* on a cluster.
- Users do not run calculations interactively -- instead they submit *non-interactive batch jobs* to the *scheduler*.
- All work about the same: some are open source; some cost money; some are very expensive.

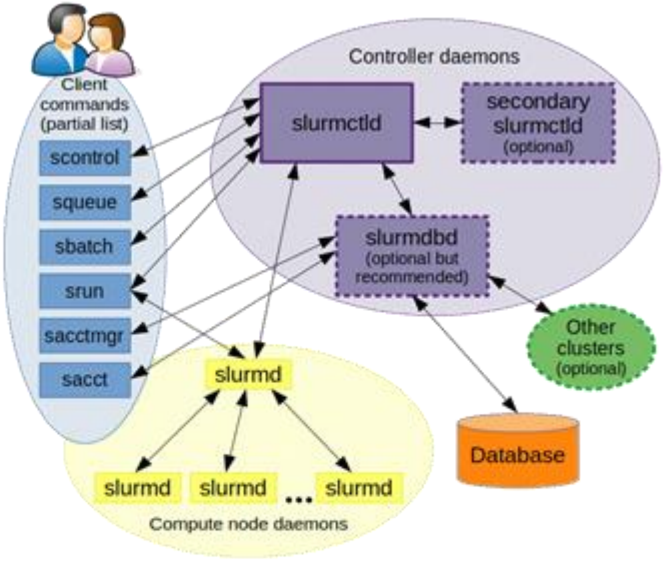
https://hpc-wiki.info/hpc/Scheduling_Basics

# Batch jobs: Slurm resource manager

Simple Linux Utility for Resource Management

- Open Source, runs on many systems
- “Glue” for parallel computer to schedule and execute jobs
- Role: Allocate resources within a cluster
  - Nodes (unique IP address)
  - Interconnect/switches
  - Generic resources (e.g. GPUs)
  - Launch and otherwise manage jobs




Client commands (partial list)
scontrol
squeue
sbatch
srun
sacctmgr
sacct
Controller daemons
slurmctld
secondary slurmctld (optional)
slurmdbd (optional but recommended)
Other clusters (optional)
Database
slurmd
slurmd
slurmd
...
slurmd
Compute node daemons

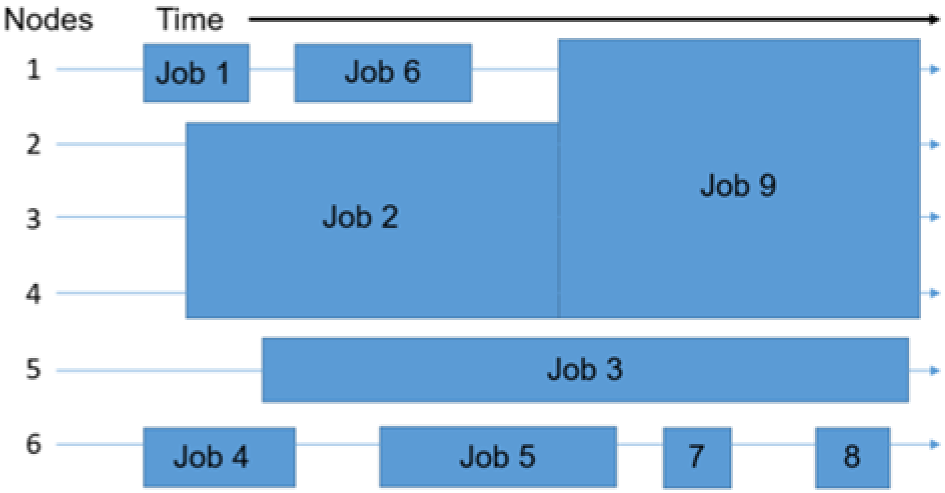
Functionality:

- Prioritize queue(s) of jobs;
- decide when and where to start jobs;
- terminate job when done;
- Appropriate resources;
- manage accounts for jobs

https://slurm.schedmd.com/sbatch.html

# How a scheduler schedules jobs

- Simple example:
  - 6-node system
  - Users submit 9 jobs.
  - Jobs 2 and 9 are multi-node
- Scheduler places the jobs in the queue and then onto the available nodes as they open up.
- Many parameters affect scheduling:
  - number of jobs submitted
  - required runtime
  - required number of cores
  - Required number of nodes
  - required main memory
  - CPU vs GPU
  - libraries, etc.



**Scheduler needs to play kind of "multidimensional tetris" to fill the cluster's nodes evenly and efficiently.**

https://hpc-wiki.info/hpc/Scheduling_Basics

# Outline

- Defining Interactive HPC
  - High-performance computing (HPC)
  - HPC batch computing
  - Interactive computing
- Accessing Interactive HPC Nodes
  - Launching nodes
  - Running GUIs using X11 forwarding
- Interactive Application Examples
  - Viewing Data: unix file ops (grep, awk, cat), gnuplot, NetCDF
  - Programming & Visualization Platforms: Matlab, R, Jupyter Notebooks
  - Gateways & Portals: simplify access to interactive apps
- Q&A

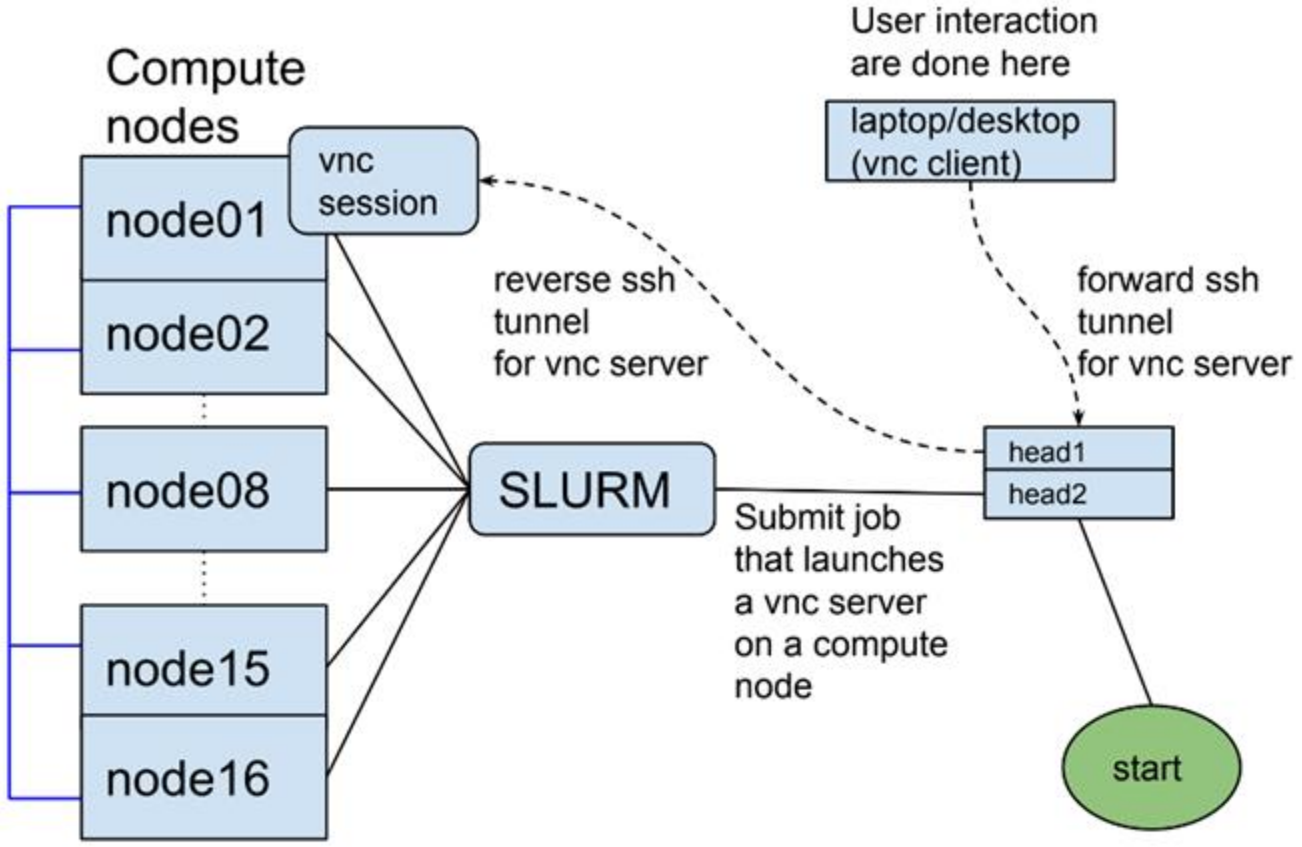
# What is Interactive HPC Computing

- In **computer** science, **interactive computing** refers to software which accepts input from the user as it runs.
  - **Interactive** software includes commonly used programs, such as word processors or spreadsheet applications.
- **Interactive HPC computing** involves *real-time* user inputs to perform tasks on a set of compute node(s) including:
  - Code development, real-time data exploration, and visualizations.
  - Used when applications have large data sets or are too large to download to local device, software is difficult install, etc.
  - User inputs come via command line interface or application GUI (Jupyter Notebooks, Matlab, R-studio).
  - Actions performed on remote compute nodes as a result of user input or program out.

# Interactive HPC Computing- Motivation

- Need more memory: Your jobs no longer fit onto the CPU/system you have been using:
  - That is why Expanse has 768 nodes (128 cores per node), with 256 GB DDR memory on each node
  - My MacBook Pro: 1 node, 8 cores, 16 GB DDR
- Too much data: your application needs more room:
  - Expanse has 1TB NVME/node, and 12 PB file system.
  - My MBP: 500 GB, no NVME
- Your network is too slow:
  - Expanse has connections to ~ 150GB/sec (or faster) networks
  - My MBP: 300 Mbps download; 11 Mbps upload
  - Bioinfo lab, running analysis on PC: FastQ dataset ~ 500 MB: would need 360+ seconds to upload 1 run.

# Real-time User Interactions



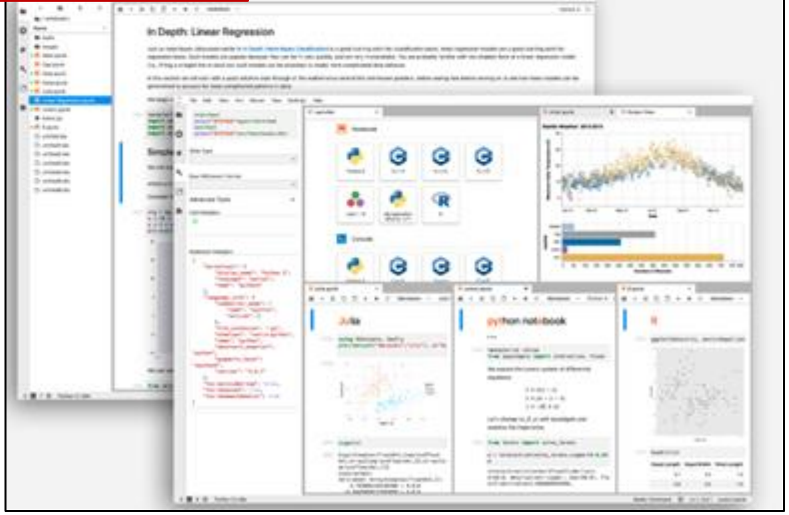
- *Real-time:* user inputs and performs tasks on a set of compute nodes
- Examples:  Juptyer notebooks; ParaView visualization tools

# Interactive HPC Methods & Applications

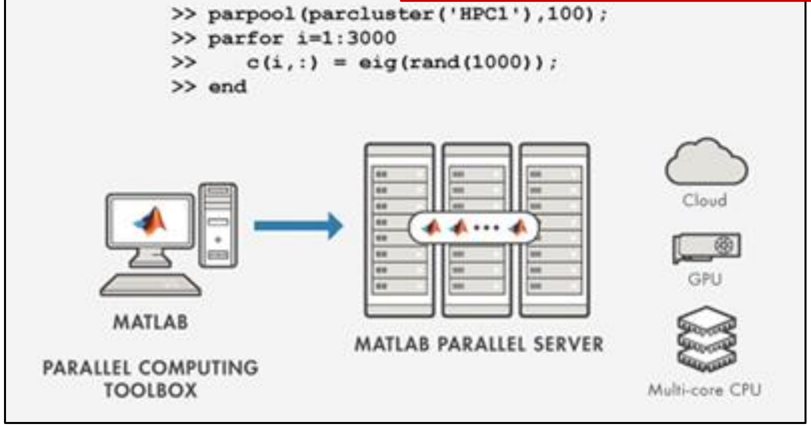
- Interact with data after job is done:
  - Unix: query file info, location, output, grep, awk, sed
  - Cat the file contents from batch job or raw data
  - NetCDF data browser
- Plot results:
  - From within the code/model using libraries
  - Command-line driven graphing utility : Gnuplot
- Data visualization apps
  - NetCDF, HPF, TAU, ParaView
  - other
- Data Analysis Platforms: Matlab, R

# Interactive HPC Computing

https://jupyter.org/



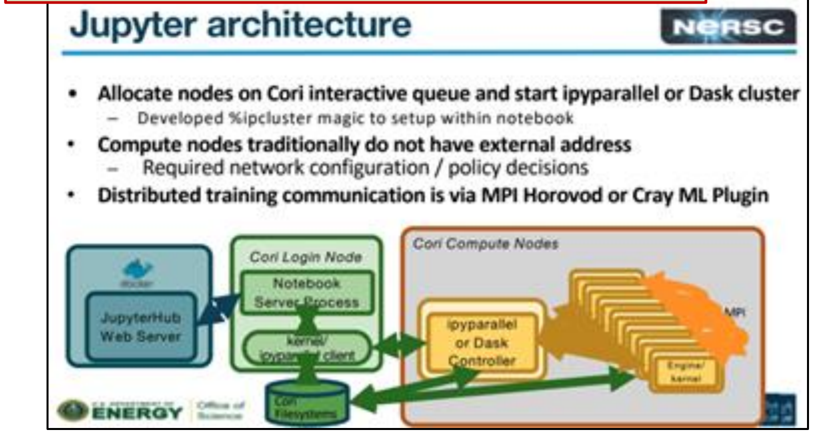
Parallel Matlab (AWS)



```
>> parpool(parcluster('HPC1'),100);
>> parfor i=1:3000
>>     c(i,:) = eig(rand(1000));
>> end
```

https://azuremarketplace.microsoft.com/en-us/marketplace/apps/mathworks-inc.matlab-parallel-server-listing?tab=Overview

Interactive Distributed Computing with Jupyter (NERSC)

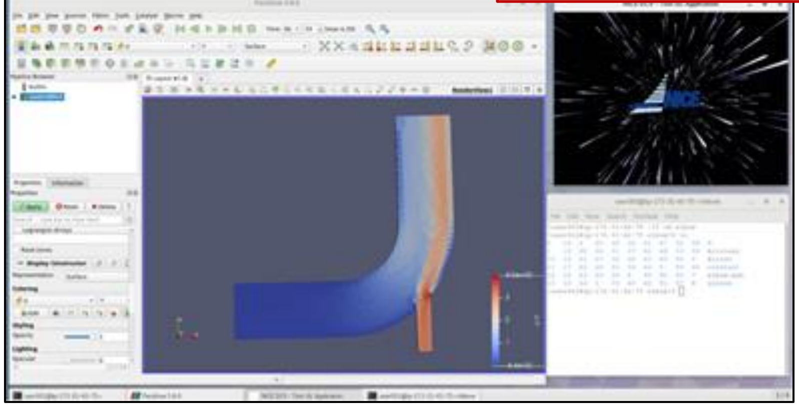


## Jupyter architecture

- Allocate nodes on Cori interactive queue and start ipyparallel or Dask cluster
  - Developed %ipcluster magic to setup within notebook
- Compute nodes traditionally do not have external address
  - Required network configuration / policy decisions
- Distributed training communication is via MPI Horovod or Cray ML Plugin

https://drive.google.com/file/d/1-OFjrk1q3L1d3uakr2xkozrPn2c2VZpZ/view

Paraview (AWS)

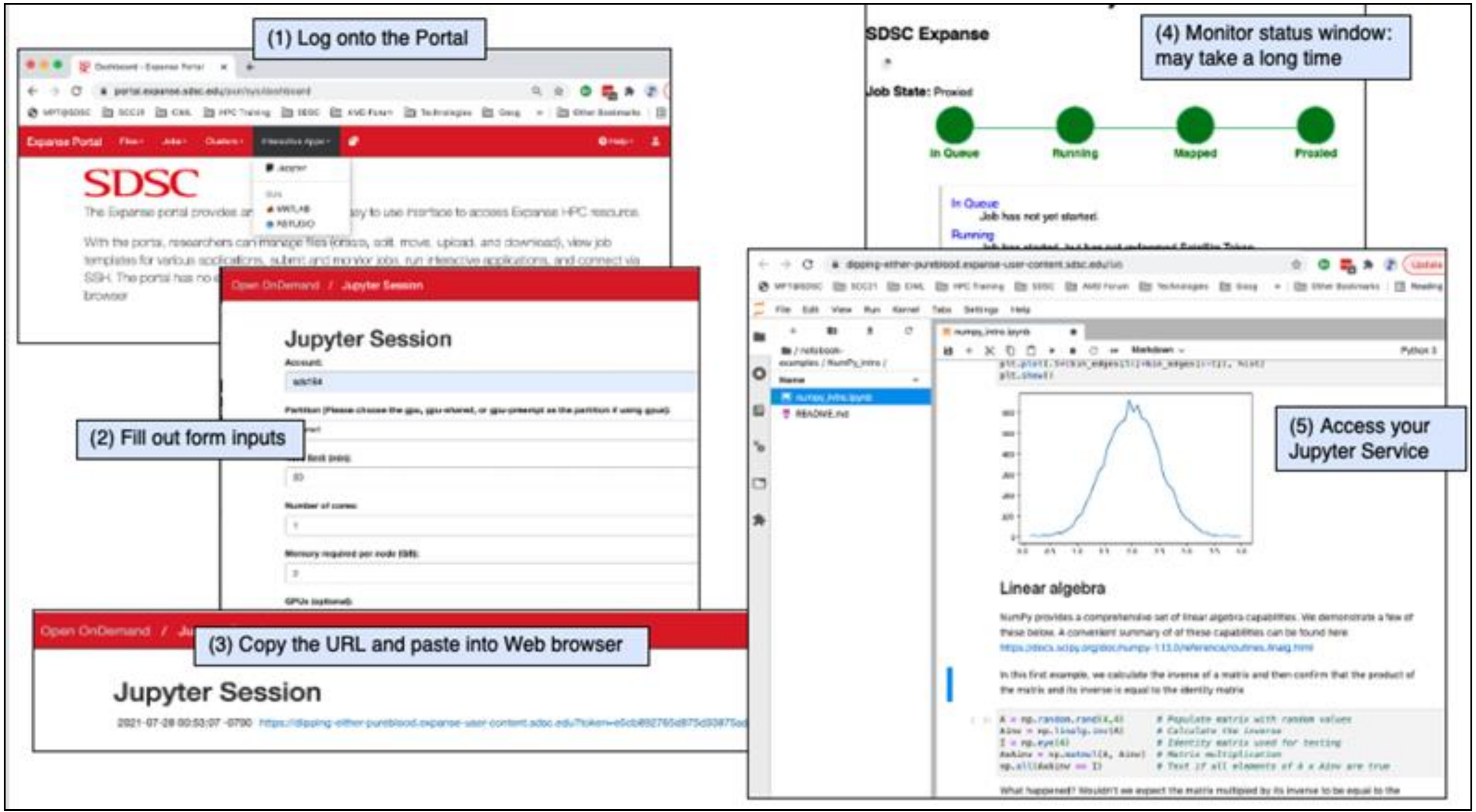


https://aws.amazon.com/blogs/compute/how-to-run-3d-interactive-applications-with-nice-dcv-in-aws-batch/

# Expanse User Portal

https://portal.expanse.sdsc.edu

- Provides Web based access to interactive applications including Jupyter Notebooks & Jupyter Lab, Matlab, Rstudio.
- Access using XSEDE portal account



(1) Log onto the Portal

SDSC

The Expanse portal provides an easy to use interface to access Expanse HPC resources.

With the portal, researchers can manage files (create, edit, move, upload, and download), view job templates for various applications, submit and monitor jobs, run interactive applications, and connect via SSH. The portal has no ... browser

(2) Fill out form inputs

Open OnDemand / Jupyter Session

Jupyter Session

Account:

Partition (Please choose the gpu, gpu-shared, or gpu-preempt as the partition if using gpus):

Number of cores:

Memory required per node (GB):

GPUs (optional):

(3) Copy the URL and paste into Web browser

Open OnDemand / Jupyter Session

Jupyter Session

(4) Monitor status window: may take a long time

SDSC Expanse

Job State: Proxied

In Queue
Job has not yet started.

Running

(5) Access your Jupyter Service

Linear algebra

NumPy provides a comprehensive set of linear algebra capabilities. We demonstrate a few of these below. A convenient summary of of these capabilities can be found here

In this first example, we calculate the inverse of a matrix and then confirm that the product of the matrix and its inverse is equal to the identity matrix

```
A = np.random.rand(4,4)      # Populate matrix with random values
Ainv = np.linalg.inv(A)      # Calculate the inverse
I = np.eye(4)                # Identity matrix used for testing
AxAinv = np.matmul(A, Ainv)  # Matrix multiplication
np.all(AxAinv == I)          # Test if all elements of A x Ainv are true
```

What happened? Wouldn't we expect the matrix multiplied by its inverse to be equal to the

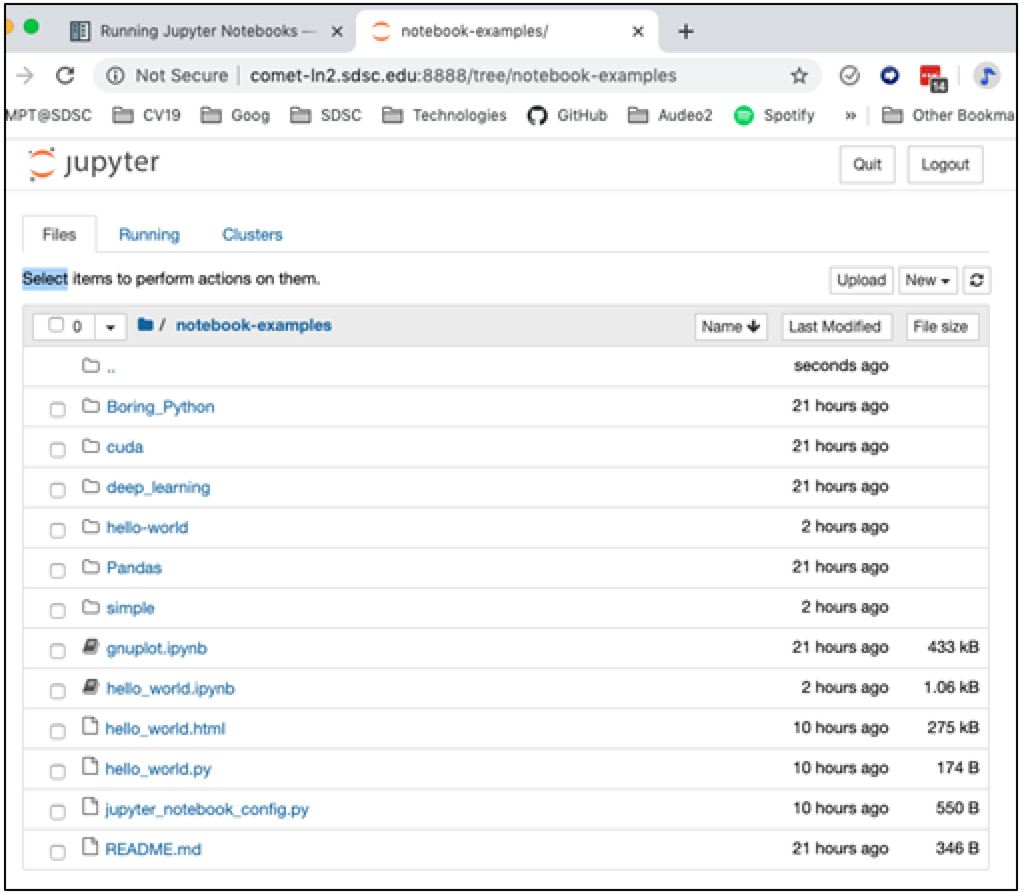
# Note: Interactive Services have a Key Vulnerability: They Provide Access to HPC File Systems, often over HTTP

SDSC Interactive Services Policy:

- Portals, JupyterHub, and other services cannot be mounted directly to disk (must be on VM or external):
    - Many use root in vulnerable ways
    - If a user launches Jupyter Lab or Notebooks, the jobs will be killed.
- Applications cannot run on login nodes.

SDSC recommendation:

- Use secure connections: when you choose insecure connections your account is vulnerable to hacking.
- Use portal.expanse.sdsc.edu



Select items to perform actions on them.

/ notebook-examples

| Name | Last Modified | File size |
|---|---|---|
| .. | seconds ago | |
| Boring_Python | 21 hours ago | |
| cuda | 21 hours ago | |
| deep_learning | 21 hours ago | |
| hello-world | 2 hours ago | |
| Pandas | 21 hours ago | |
| simple | 2 hours ago | |
| gnuplot.ipynb | 21 hours ago | 433 kB |
| hello_world.ipynb | 2 hours ago | 1.06 kB |
| hello_world.html | 10 hours ago | 275 kB |
| hello_world.py | 10 hours ago | 174 B |
| jupyter_notebook_config.py | 10 hours ago | 550 B |
| README.md | 21 hours ago | 346 B |

# Outline

- Defining Interactive HPC
  - High-performance computing (HPC)
  - HPC batch computing
  - Interactive computing
- Accessing Interactive HPC Nodes
  - Launching nodes
  - Running GUIs using X11 forwarding
- Interactive Application Examples
  - Viewing Data: unix file ops (grep, awk, cat), gnuplot, NetCDF
  - Programming & Visualization Platforms: Matlab, R, Jupyter Notebooks
  - Gateways & Portals: simplify access to interactive apps
- Q&A

# When would you need to use an Interactive node?

- The application is interactive (Matlab, Notebooks)
- Need to stage large amounts of data
- AI applications (Pytorch, Tensorflow)
- Need to compile a very large, complex application that may take a long time.
  - Java code, 1 million lines, > 30 minutes on DELL 5511 laptop

# Accessing Interactive Compute Nodes on Expanse

- Connect to HPC system (e.g. Expanse) via terminal using SSH □ secure connections
- Use the *srun* command to obtain nodes for ‘live,’ command line interactive access:

| | |
|---|---|
| **CPU** | srun --partition=gpu-debug --pty --account=use300 --ntasks-per-node=10 --nodes=1 --mem=96G --gpus=1 -t 00:30:00 --wait=0 --export=ALL /bin/bash |
| **GPU** | srun --partition=gpu-debug --pty --account=use300 --ntasks-per-node=10 --nodes=1 --mem=96G --gpus=1 -t 00:30:00 --wait=0 --export=ALL /bin/bash |

(Tested 04/17/2024)

# Using An Interactive CPU node

```
[mthomas@login01 calc-prime]$ srun --partition=gpu-debug --pty --account=use300 --
ntasks-per-node=16 --nodes=1 --mem=96G --gpus=1 -t 00:30:00 --wait=0 --
export=ALL /bin/bash

srun: job 24457429 has been allocated resources
[mthomas@exp-9-55 calc-prime]$ module purge
[mthomas@exp-9-55 calc-prime]$ module load slurm
[mthomas@exp-9-55 calc-prime]$ module load cpu
[mthomas@exp-9-55 calc-prime]$ module load gcc/10.2.0
[mthomas@exp-9-55 calc-prime]$ module load openmpi/4.1.1
[mthomas@exp-9-55 calc-prime]$ mpirun -n 16 ./mpi_prime
06 August 2023 11:10:26 PM
PRIME_MPI    n_hi= 5000000     C/MPI version
An MPI example program to count the number of primes: # processes is 64
        N        Pi         Time
        1        0        0.013258
        2        1        0.001058
        4        2        0.000101
        8        4        0.000101
    [SNIP]
   131072    12251       0.110848
   262144    23000       0.410792
   524288    43390       1.527210
  1048576     82025       5.733612
  2097152    155611      21.725862
PRIME_MPI - Master process: Normal end of execution.
06 August 2023 11:12:26 PM
```

Request an interactive node for 30 minutes

- Exit interactive session when your work is done or you will be charged more CPU time.
- Beware of oversubscribing your job: don't ask for more cores than you have requested.
- Intel compiler allows this, but your performance will be degraded.

# Using Interactive GPU nodes

Request an interactive node for 30 minutes

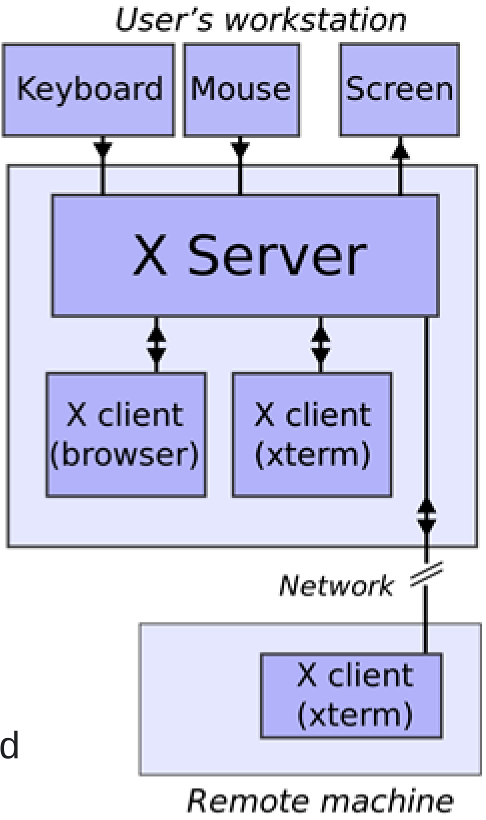
```
[snip]
Last login: Fri Feb 18 12:58:32 2022 from 76.176.117.51
[username @login02 ~]$
[username @login02 ~]$ srun --partition=gpu-debug --pty --account=use300 --ntasks-per-node=10 --nodes=1 --
mem=96G --gpus=1 -t 00:30:00 --wait=0 --export=ALL /bin/bash
srun: job 9794018 queued and waiting for resources
srun: job 9794018 has been allocated resources
[mthomas@exp-14-57 ~]$
[mthomas@exp-14-57 ~]$ nvidia-smi
Fri Feb 18 13:04:19 2022
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 460.32.03    Driver Version: 460.32.03    CUDA Version: 11.2     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================
|   0  Tesla V100-SXM2...  On   | 00000000:86:00.0 Off |                    0 |
| N/A   34C    P0    41W / 300W |      0MiB / 32510MiB |      0%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|  No running processes found
+-----------------------------------------------------------------------------+
[username @exp-14-57 ~]$ exit
```

Verify you are on a GPU node

Exit when tasks are done

# Visualizating Apps: Use X11 Forwarding

- X Window System (X11): windowing system for bitmap displays, common on Unix-like operating systems.
  - MacOS uses Quartz
- X is an architecture-independent system for remote graphical user interfaces and input device capabilities



- X server receives input from a local keyboard and mouse and displays to a screen.
- Web browser and a terminal emulator run on the user's workstation
- A terminal emulator runs on a remote computer but is controlled and monitored from the user's machine

SRC: https://en.wikipedia.org/wiki/X_Window_System

# X11 Forwarding: connecting to Expanse

- On MacOS: Install X11 forwarding software:
  - MacOS: Xquartz
- Use the connection command:

ssh -Y mthomas@login.expanse.sdsc.edu'

```
[mthomas@home]$ brew install --cask xquartz
Updating Homebrew...
Updated 2 taps (homebrew/core and homebrew/cask).
==> New Formulae
[SNIP]
installer: The upgrade was successful.
🍺  xquartz was successfully installed!
quantum:~ mthomas$ which xquartz
/opt/X11/bin/xquartz
```

Note: MacOS/Ventura has an X11 problem that is being debugged.

https://en.wikipedia.org/wiki/X_Window_System

# X11 Forwarding to Expanse Interactive Node

- Use Xquartz windows
- Create two Expanse X11 login connections
  ssh -Y mthomas@login.expanse.sdsc.edu
- Window/Connection #1: request interactive node

```
[localhost]. ssh -Y mthomas@login.expanse.sdsc.edu
[mthomas@login01 ~] srun --partition=debug  --pty --account=use300 --nodes=1
--ntasks-per-node=4  --mem=8G -t 00:30:00 --wait=0 --export=ALL /bin/bash
srun: job 24457429 has been allocated resources
[mthomas@exp-9-55]$
```

- Window/Connection #2: connect  to the interactive node; set up module environment and launch app

```
[localhost] ssh -Y mthomas@login.expanse.sdsc.edu
[mthomas@login01 ~] ssh -Y exp-9-55
Last login: Wed Dec  6 19:21:56 2023 from 10.21.0.19
[mthomas@exp-9-55 ~]$ module load  gpu/0.17.3b
[mthomas@exp-9-55 ~]$ module load matlab/2022b/nmbr5dd
[mthomas@exp-9-55 ~]$ matlab
```

- Note: MacOS X11 problem that is being debugged. Use the Expanse Portal

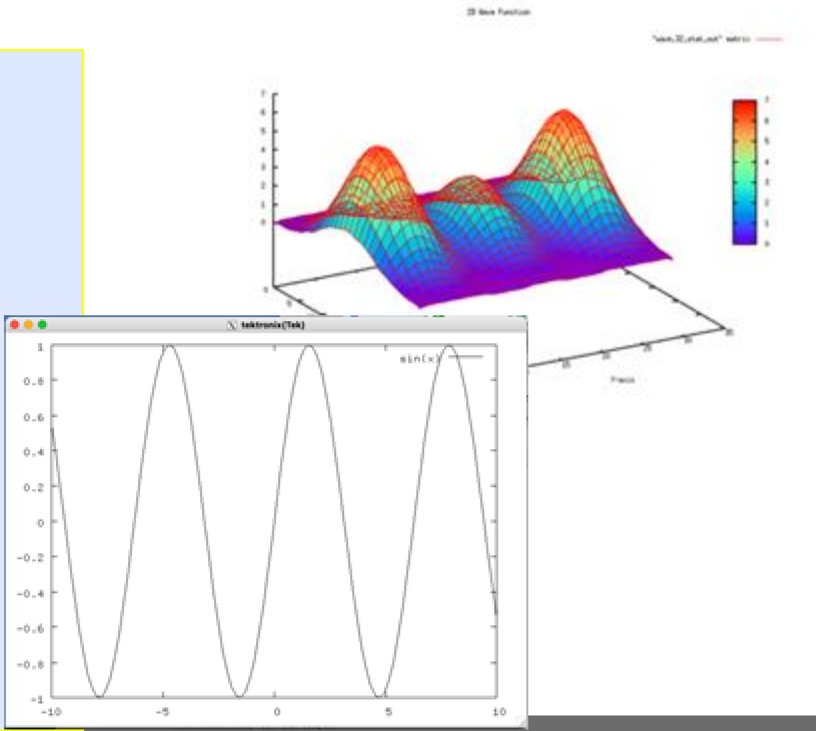
https://en.wikipedia.org/wiki/X_Window_System

# gnuplot

- *Depends on X11, and your local host (MacOS XQuartz not working fully)*
- *gnuplot* is a command-driven interactive function plotting program. It can be used to plot functions and data points in both two- and three- dimensional plots in many different formats.
- Designed primarily for the visual display of scientific data.
- Can be run from the gnuplot app or from within your program so you can save visualizations of results.

cpu/0.15.4 gcc/10.2.0

```
[mthomas@exp-9-55]$ module load cpu/0.17.3b  gcc/10.2.0/npcyll4
gnuplot/5.4.2/mfinpvw
[mthomas@exp-9-55]$ which gnuplot
/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen2/gcc-
10.2.0/gnuplot-5.2.8-uwugzxg4dgxaiciheiepgol67cw7m6yg/bin/gnuplot
[mthomas@login01 ~]$ gnuplot
        G N U P L O T
        Version 5.4 patchlevel 2    last modified 2021-06-01
        Copyright (C) 1986-1993, 1998, 2004, 2007-2021
        Thomas Williams, Colin Kelley and many others
        gnuplot home:     http://www.gnuplot.info
        faq, bugs, etc:   type "help FAQ"
        immediate help:   type "help"  (plot window: hit 'h')
Terminal type is now 'unknown'
gnuplot> set terminal xterm
Terminal type is now 'xterm'
gnuplot> plot sin(x)
```



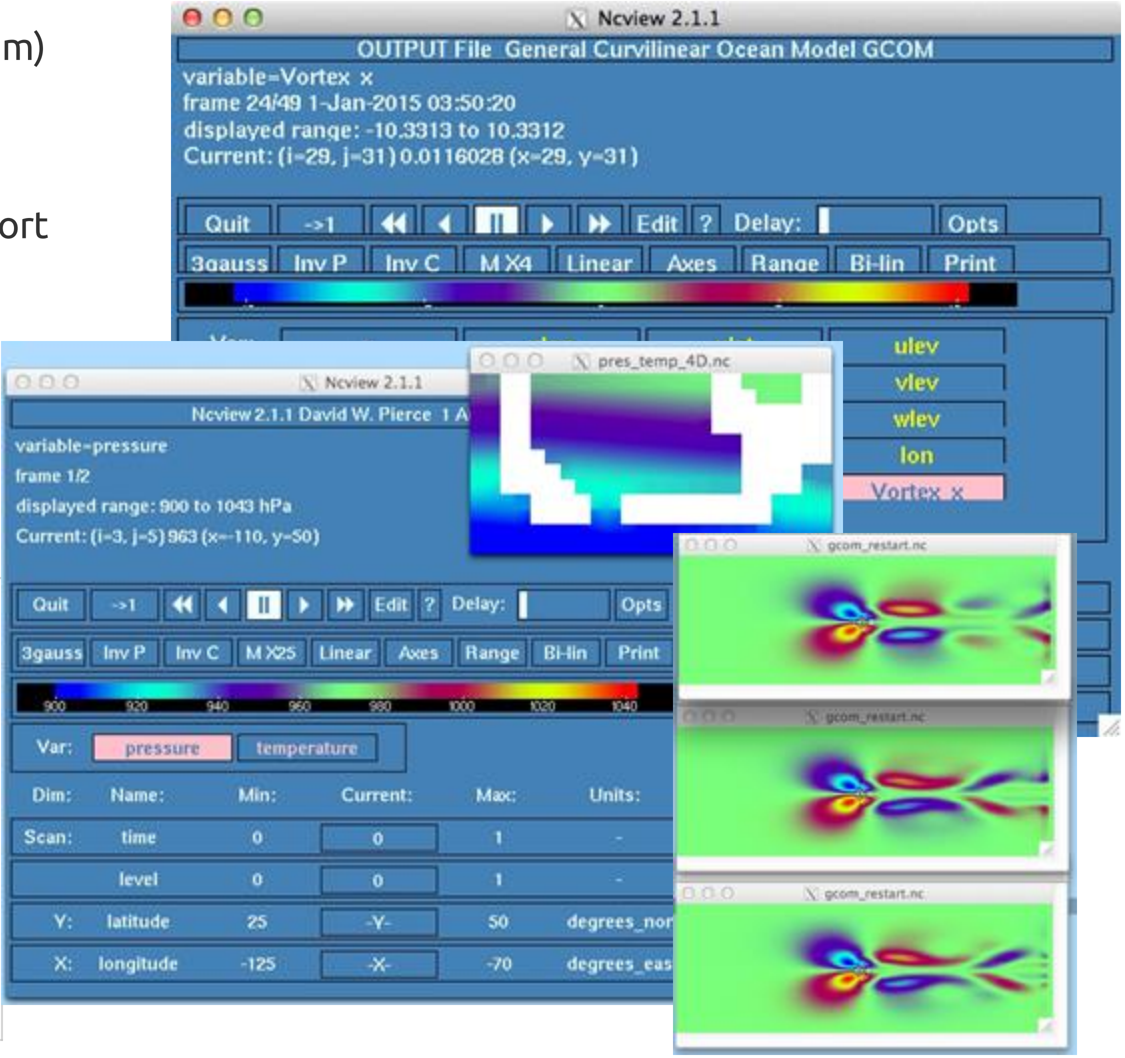
https://alvinalexander.com/technology/gnuplot-charts-graphs-examples/

# Outline

- Defining Interactive HPC
  - High-performance computing (HPC)
  - HPC batch computing
  - Interactive computing
- Accessing Interactive HPC Nodes
  - Launching nodes
  - Running GUIs using X11 forwarding
- Interactive Application Examples
  - Viewing Data: Unix file ops (grep, awk, cat), gnuplot, NetCDF
  - Programming & Visualization Platforms: Matlab, R, Jupyter Notebooks
  - Gateways & Portals: *simplify access* to interactive apps
- Q&A

# NetCDF Data Files

- NetCDF (Network Common Data Form) files are binary files containing both data and metadata about the data.
- API: software libraries and machine-independent data formats that support the creation, access, and sharing of array-oriented scientific data
- Similar to high-density format (HDF)
- NetCDF clients (*ncview*, *ncdump*) can be used to query and plot data in real-time

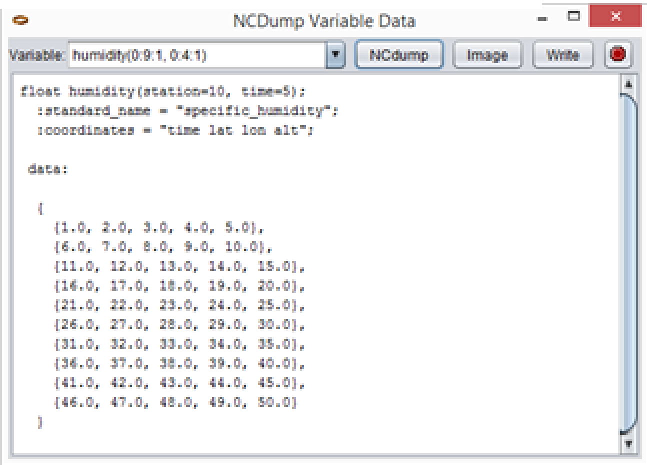


Ncview 2.1.1

OUTPUT File General Curvilinear Ocean Model GCOM
variable=Vortex x
frame 24/49 1-Jan-2015 03:50:20
displayed range: -10.3313 to 10.3312
Current: (i=29, j=31) 0.0116028 (x=29, y=31)

Ncview 2.1.1 David W. Pierce 1 A
variable=pressure
frame 1/2
displayed range: 900 to 1043 hPa
Current: (i=3, j=5) 963 (x=-110, y=50)

| Dim: | Name: | Min: | Current: | Max: | Units: |
|---|---|---|---|---|---|
| Scan: | time | 0 | 0 | 1 | - |
| | level | 0 | 0 | 1 | - |
| Y: | latitude | 25 | -Y- | 50 | degrees_nor |
| X: | longitude | -125 | -X- | -70 | degrees_eas |



NCDump Variable Data

Variable: humidity(0:9:1, 0:4:1)

```
float humidity(station=10, time=5);
  :standard_name = "specific_humidity";
  :coordinates = "time lat lon alt";

 data:

  {
    {1.0, 2.0, 3.0, 4.0, 5.0},
    {6.0, 7.0, 8.0, 9.0, 10.0},
    {11.0, 12.0, 13.0, 14.0, 15.0},
    {16.0, 17.0, 18.0, 19.0, 20.0},
    {21.0, 22.0, 23.0, 24.0, 25.0},
    {26.0, 27.0, 28.0, 29.0, 30.0},
    {31.0, 32.0, 33.0, 34.0, 35.0},
    {36.0, 37.0, 38.0, 39.0, 40.0},
    {41.0, 42.0, 43.0, 44.0, 45.0},
    {46.0, 47.0, 48.0, 49.0, 50.0}
  }
```

https://unidata.ucar.edu/netcdf

# Configuring NetCDF on Expanse

- Example below is for very simple, serial example provided by SciNet (Canada): see
  - Select the right module environment: it’s a big software system
- Set up modules
- Put SciNet ENV vars into bash scripts
- Build the compile command (Makefile not easy to adapt)
- Example will generate a data.nc file
- Browse & query file contents

# NetCDF: Configure Shell Environment

- Add environment variables to your shell env:

```
[mthomas@login02 netcdf]$ cat ~/.bash_profile
[SNIP]
# netcdf library and include paths
SCINET_NETCDF_INC=/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen2/intel-19.1.1.217/netcdf-c-4.7.4-62xm3fwd6soo2zqqbd4t7jza4ro5riyv/include
SCINET_NETCDF_LIB=/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen2/intel-19.1.1.217/netcdf-c-4.7.4-62xm3fwd6soo2zqqbd4t7jza4ro5riyv/lib

#HDF5 library and include paths
SCINET_HDF5_INC=/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen2/intel-19.1.1.217/hdf5-1.10.7-jsee7cinki6pcunimgfe3ovv6lz7gwnt/include
SCINET_HDF5_LIB=/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen2/intel-19.1.1.217/hdf5-1.10.7-jsee7cinki6pcunimgfe3ovv6lz7gwnt/lib

#update path to binaries
PATH=$PATH:/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen2/intel-19.1.1.217/netcdf-c-4.7.4-62xm3fwd6soo2zqqbd4t7jza4ro5riyv/bin:/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen2/intel-19.1.1.217/hdf5-1.10.7-jsee7cinki6pcunimgfe3ovv6lz7gwnt/bin:/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen2/intel-19.1.1.217/ncview-2.1.8-bm3tockq6ljgtxefovxia5hujnqmpdpn/bin
[SNIP]
```

# NetCDF: Configure Modules

- Setup the right modules for the version you want to use:

```
[mthomas@login02 netcdf]$
[mthomas@login02 netcdf]$ module load cpu/0.15.4  intel/19.1.1.217  mvapich2/2.3.4 module load netcdf-c/4.7.4
[mthomas@login02 netcdf]$
[mthomas@login02 netcdf]$
[mthomas@login02 netcdf]$ which ncdump
/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen2/intel-19.1.1.217/netcdf-c-4.7.4-
62xm3fwd6soo2zqqbd4t7jza4ro5riyv/bin/ncdump
[mthomas@login02 netcdf]$ which ncview
/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen2/intel-19.1.1.217/ncview-2.1.8-
bm3tockq6ljgtxefovxia5hujnqmpdpn/bin/ncview
```

# Query NetCDF Data File Contents: ncdump

```
[mthomas@login02 netcdf]$ pwd
[mthomas@login02 netcdf]$ pwd
/home/mthomas/hpctr-examples/netcdf/parIO/netcdf
[mthomas@login02 netcdf]$ ls *.c
-rw-r--r-- 1 mthomas use300   8874 Oct  5  2010 2darray.c
-rwxr-xr-x 1 mthomas use300 304648 Dec  6 01:27 2darray-simple.c
-rw-r--r-- 1 mthomas use300  15903 Oct  6  2010 parallel2darray.c
-rw-r--r-- 1 mthomas use300   5865 Oct  5  2010 read2darray-simple.c
-rw-r--r-- 1 mthomas use300 240176mpicc -I/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen2/intel-
19.1.1.217/netcdf-c-4.7.4-62xm3fwd6soo2zqqbd4t7jza4ro5riyv/include -L/cm/shared/apps/spack/cpu/opt/spack/linux-
centos8-zen2/intel-19.1.1.217/netcdf-c-4.7.4-62xm3fwd6soo2zqqbd4t7jza4ro5riyv/lib -lnetcdf -o 2darray 2darray.c
 Dec  6 01:27 data-simple.nc
[mthomas@login02 netcdf]$ ./2darray
[mthomas@login02 netcdf]$ ll *.nc
-rw-r--r-- 1 mthomas use300 241164 Dec  6 01:25 data.2darray.nc
-rw-r--r-- 1 mthomas use300 241164 Dec  6 21:27 data.nc
```

# Query NetCDF Data File Contents: ncdump

Tool for viewing output data file contents

```
[mthomas@login02 netcdf]$ ncdump -h
ncdump [-c|-h] [-v ...] [[-b|-f] [c|f]] [-l len] [-n name] [-p n[,n]] [-k] [-x] [-s] [-t|-i] [-g ...] [-w] [-Ln] file
  [-c]             Coordinate variable data and header information
  [-h]             Header information only, no data
  [-v var1[,...]]  Data for variable(s) <var1>,... only
  [-b [c|f]]       Brief annotations for C or Fortran indices in data
  [-f [c|f]]       Full annotations for C or Fortran indices in data
  [-l len]         Line length maximum in data section (default 80)
  [-n name]        Name for netCDF (default derived from file name)
  [-p n[,n]]       Display floating-point values with less precision
  [-k]             Output kind of netCDF file
  [-s]             Output special (virtual) attributes
  [-t]             Output time data as date-time strings
  [-i]             Output time data as date-time strings with ISO-8601 'T' separator
  [-g grp1[,...]]  Data and metadata for group(s) <grp1>,... only
  [-w]             With client-side caching of variables for DAP URLs
  [-x]             Output XML (NcML) instead of CDL
  [-Xp]            Unconditionally suppress output of the properties attribute
  [-Ln]            Set log level to n (>= 0); ignore if logging not enabled.
  file             Name of netCDF file (or URL if DAP access enabled)
```

# Read NetCDF Data File Contents: ncdump

***ncdump***: Tool for viewing output data file contents

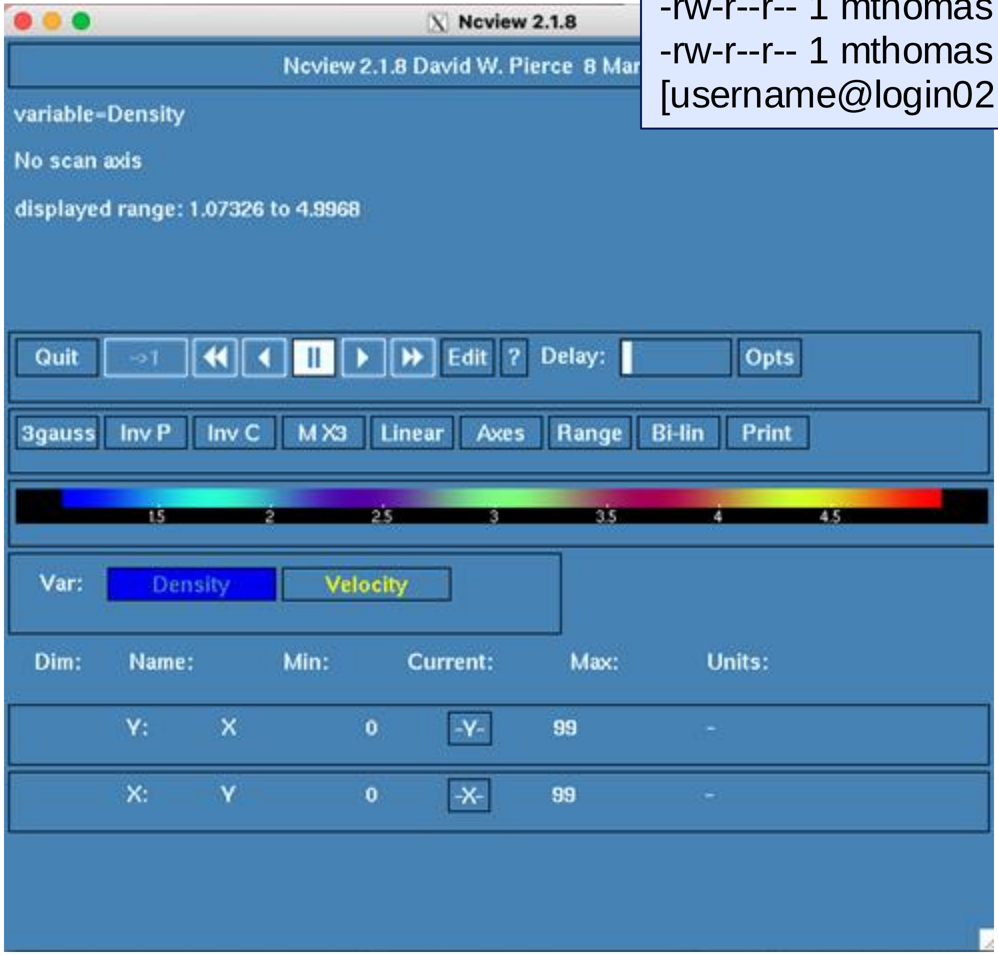
```
[mthomas@login02 netcdf]$ ll *.nc
-rw-r--r-- 1 mthomas use300 241164 Dec  6 01:25 data.2darray.nc
-rw-r--r-- 1 mthomas use300 240176 Dec  6 01:27 data-simple.nc
[mthomas@login02 netcdf]$ ncdump -h data-simple.nc
netcdf data-simple {
dimensions:
X = 100 ;
Y = 100 ;
velocity\ component = 2 ;
variables:
double Density(X, Y) ;
double Velocity(velocity\ component, X, Y) ;
}
[mthomas@login02 netcdf]$ ncdump -x data-simple.nc
<?xml version="1.0" encoding="UTF-8"?>
<netcdf xmlns="http://www.unidata.ucar.edu/namespaces/netcdf/ncml-2.2" location="data-simple.nc">
  <dimension name="X" length="100" />
  <dimension name="Y" length="100" />
  <dimension name="velocity component" length="2" />
  <variable name="Density" shape="X Y" type="double">
  </variable>
  <variable name="Velocity" shape="velocity component X Y" type="double">
  </variable>
</netcdf>
```

# Read NetCDF Data File Contents: ncdump

```
[mthomas@login02 netcdf]$ ncdump -t  data-simple.nc | more
netcdf data-simple {
dimensions:
X = 100 ;
Y = 100 ;
velocity\ component = 2 ;
variables:
double Density(X, Y) ;
double Velocity(velocity\ component, X, Y) ;
data:
 Density =
  1.07326255555494, 1.07930091286413, 1.08569972700294, 1.09246679805167,
    1.09960871294202, 1.10713070597055, 1.11503651964736, 1.12332826729031,
    1.13200629886235, 1.14106907162359, 1.1505130272287, 1.16033247694148,
    1.17051949666066, 1.18106383345281, 1.19195282526666, 1.20317133545959,
    1.21470170369791, 1.22652371469942, 1.23861458616838, 1.25094897713016,
    1.26349901770561, 1.27623436117604, 1.28912225897801, 1.30212765903751,
    1.31521332760545, 1.3283399944956, 1.34146652135377, 1.35455009230782,
    1.3675464260648, 1.38041000823913, 1.39309434241745, 1.40555221819654,
[SNIP]
    -2.8468704354508, -2.69398173507181, -2.54522816466233,
    -2.40084389414749, -2.2610296301804, -2.12595326718743,
    -1.99575075552494, -1.87052716603845, -1.75035792940546,
    -1.63529022805598, -1.52534451818797, -1.42051615941371,
    -1.32077712986823, -1.22607780516019, -1.13634878032523, -1.05150271492394 ;
```

# Visualize NetCDF Data: ncview

```
[mthomas@login02 netcdf]$ ll *.nc
-rw-r--r-- 1 mthomas use300 241164 Dec  6 01:25 data.2darray.nc
-rw-r--r-- 1 mthomas use300 240176 Dec  6 01:27 data-simple.nc
[username@login02 netcdf] ncview data-simple.nc
```



Ncview 2.1.8 David W. Pierce 8 Mar

variable=Density

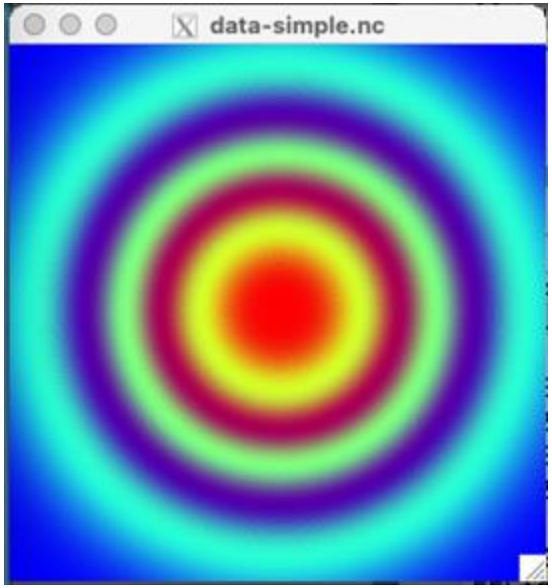
No scan axis

displayed range: 1.07326 to 4.9968

Var: Density Velocity

```
Dim:   Name:   Min:   Current:   Max:   Units:
Y:     X       0      -Y-        99     -
X:     Y       0      -X-        99     -
```

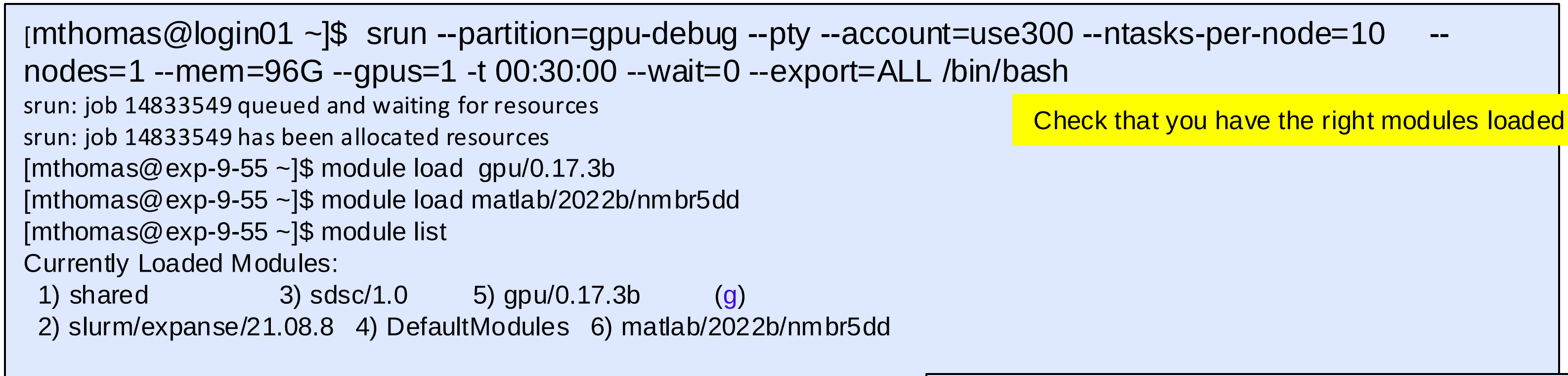
Viewing 2D density plot



# Outline

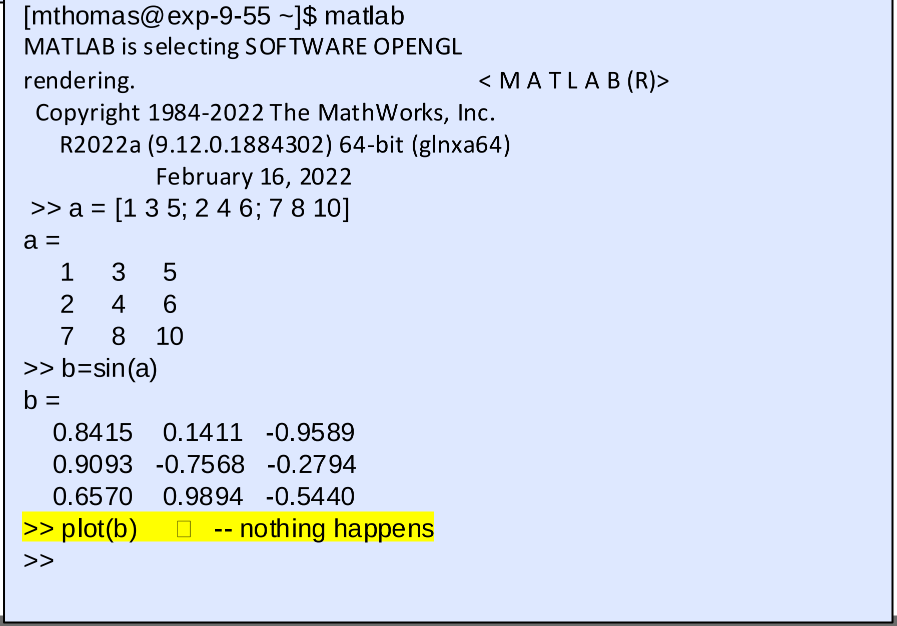
- What is High-Performance Computing (HPC)?
- What is HPC batch computing
- Defining Interactive HPC Computing
- Accessing Interactive HPC Nodes
- Interactive Application Examples
  - Viewing Data: unix file operators (grep, awk, cat), gnuplot, NetCDF
  - Programming & Visualization Platforms: Matlab, R, Jupyter Notebooks (and NetCDF)
  - Gateways & Portals: simplify access to interactive apps

# Expanse Interactive Jobs: Running Matlab from Command Line



```
[mthomas@login01 ~]$  srun --partition=gpu-debug --pty --account=use300 --ntasks-per-node=10     --
nodes=1 --mem=96G --gpus=1 -t 00:30:00 --wait=0 --export=ALL /bin/bash
srun: job 14833549 queued and waiting for resources
srun: job 14833549 has been allocated resources
[mthomas@exp-9-55 ~]$ module load  gpu/0.17.3b
[mthomas@exp-9-55 ~]$ module load matlab/2022b/nmbr5dd
[mthomas@exp-9-55 ~]$ module list
Currently Loaded Modules:
  1) shared              3) sdsc/1.0        5) gpu/0.17.3b        (g)
  2) slurm/expanse/21.08.8   4) DefaultModules   6) matlab/2022b/nmbr5dd
```

Check that you have the right modules loaded



```
[mthomas@exp-9-55 ~]$ matlab
MATLAB is selecting SOFTWARE OPENGL
rendering.                                        < M A T L A B (R)>
  Copyright 1984-2022 The MathWorks, Inc.
      R2022a (9.12.0.1884302) 64-bit (glnxa64)
                 February 16, 2022
 >> a = [1 3 5; 2 4 6; 7 8 10]
a =
     1     3     5
     2     4     6
     7     8    10
>> b=sin(a)
b =
    0.8415    0.1411   -0.9589
    0.9093   -0.7568   -0.2794
    0.6570    0.9894   -0.5440
>> plot(b)     □   -- nothing happens
>>
```

# Expanse Interactive Jobs: Running Matlab with GUI

To use a Graphical User Interface (GUI) as part of your interactive job, you will need to set up Xforwarding. Example below is for using XQuartz on a MAC. For examples for Windows, see: http://systems.eecs.tufts.edu/x11-forwarding/

***Step 1:** Log on to Expanse, using **–Y** option (trusted); create two connections; W2 should be the XQuartz terminal*

W1

```
[mthomas@home]$ ssh –Y mthomas@login.expanse.sdsc.edu
```

W2

```
[mthomas@home]$ ssh –Y mthomas@login.expanse.sdsc.edu
```

***Step 2: Window 1:** Request an interactive node, using **srun command***

*This window will not be used for anything else but holding the interactive node connection*

W1

```
[mthomas@login01 ~]$ srun --partition=debug  --pty --account=use300 --nodes=1 --ntasks-per-node=4  --mem=8G -t 00:30:00 --wait=0 --export=ALL /bin/bash
srun: job 26814162 queued and waiting for resources
srun: job 26814162 has been allocated resources
[mthomas@exp-9-55 ~]$
```

***Step 3: Window 2:***

***Connect to the interactive node:** atlab*

W2

```
[mthomas@login02 ~]$ ssh -Y exp-9-55
Last login: Wed Dec  6 19:23:47 2023 from 10.21.0.19
```

***Step 4: Window 2:***

*Setup Matlab module environment*

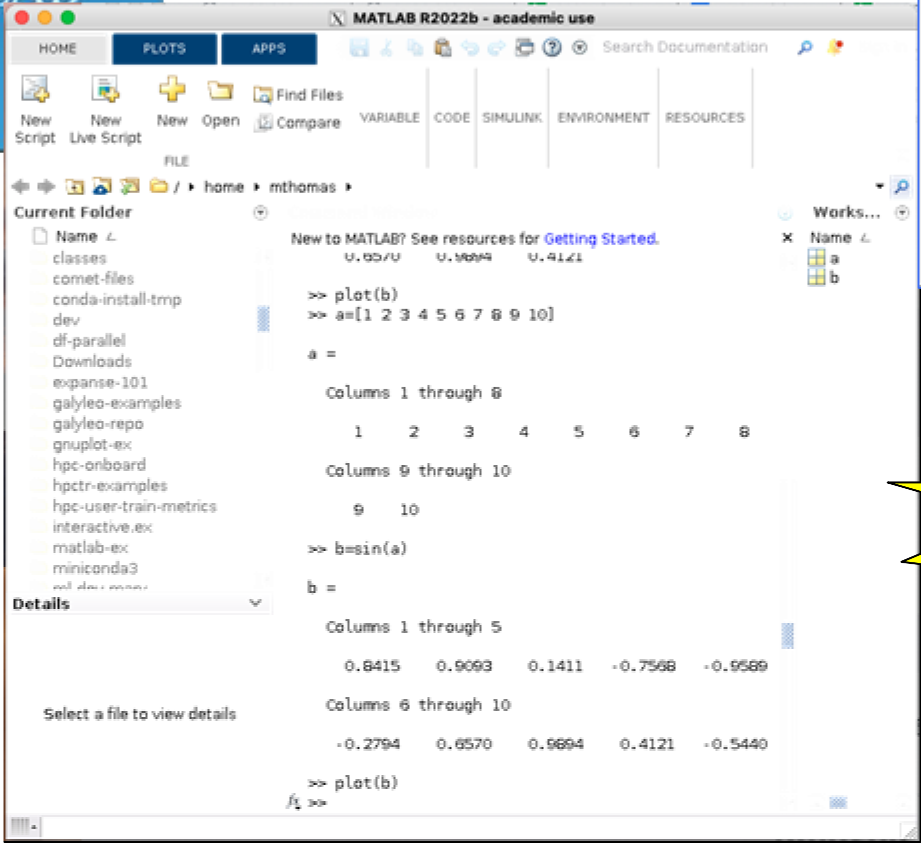
*Launch Matlab*

W2

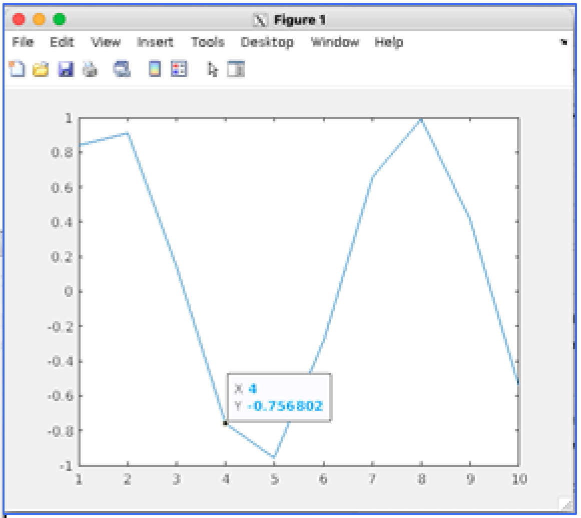
```
[mthomas@exp-9-55 ~]$ module load  gpu/0.17.3b
[mthomas@exp-9-55 ~]$ module load matlab/2022b/nmbr5dd
[mthomas@exp-9-55 ~]$ module list
Currently Loaded Modules:
  1) shared     3) sdsc/1.0    5) gpu/0.17.3b  (g)
  2) slurm/expanse/21.08.8   4) DefaultModules   6) matlab/2022b/nmbr5dd
[mthomas@exp-9-55 ~]$ matlab
MATLAB is selecting SOFTWARE OPENGL rendering.
```

**GUI : USE EXPANSE PORTAL**

# HPC Interactive Jobs: Running Matlab with GUI



```
New to MATLAB? See resources for Getting Started.
    0.6570    0.9894    0.4121

>> plot(b)
>> a=[1 2 3 4 5 6 7 8 9 10]

a =

  Columns 1 through 8

     1     2     3     4     5     6     7     8

  Columns 9 through 10

     9    10

>> b=sin(a)

b =

  Columns 1 through 5

    0.8415    0.9093    0.1411   -0.7568   -0.9589

  Columns 6 through 10

   -0.2794    0.6570    0.9894    0.4121   -0.5440

>> plot(b)
fx >>
```



FOR GUIs: USE EXPANSE PORTAL!!

# Interactive Jobs: Running R (TSCC only)
# Expanse: use portal.expanse.sdsc.edu

***Step 3:** Request an interactive node, using **–X** option*

```
[mthomas@tscc-login2 ~]$ qsub -I -q glean -l nodes=1:ppn=1
qsub: waiting for job 26844488.tscc-mgr7.local to start
qsub: job 26844488.tscc-mgr7.local ready
[mthomas@tscc-4-46 ~]$
```

***Step 4:** Setup your module environment and run Matlab*

```
[mthomas@tscc-4-46 ~]$ module load R
Unloading compiler-dependent module openmpi_ib/3.1.4
[mthomas@tscc-4-46 ~]$ R
R version 4.0.2 (2020-06-22) -- "Taking Off Again"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
[SNIP]
R is a collaborative project with many contributors.
[SNIP]
> myString <- "Hello, World!"
> print ( myString )
[1] "Hello, World!"
>
```

**For GUI: USE EXPANSE PORTAL**

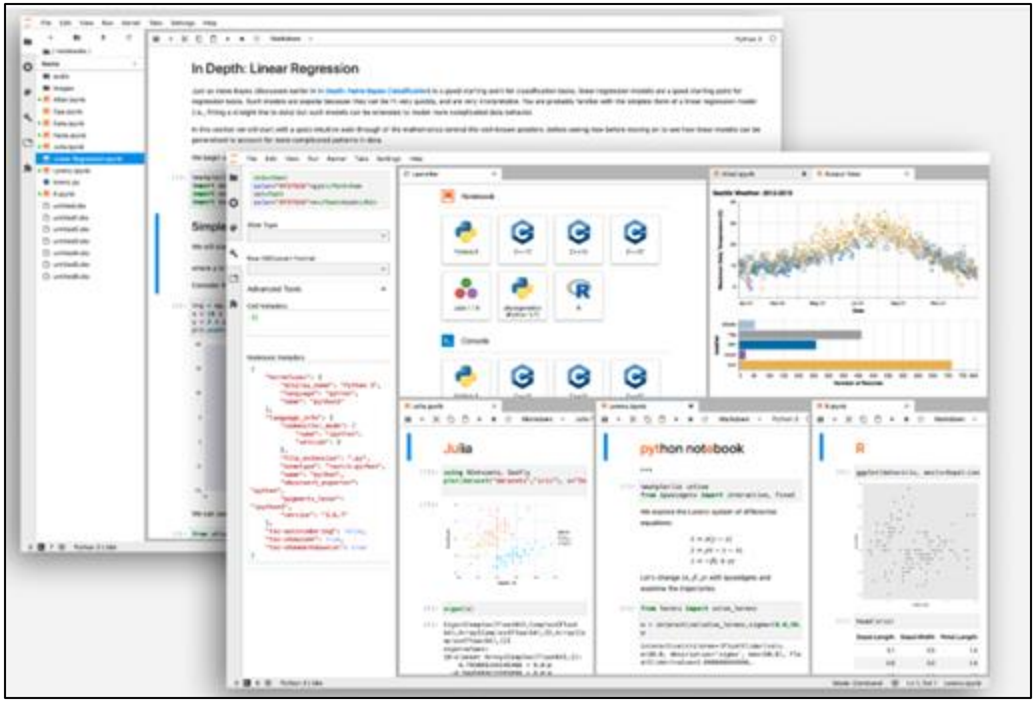
# Juptyer Notebooks

What is Jupyter?

> ***Jupyter*** *is a free, open-source, interactive web tool known as a computational notebook, which researchers can use to combine software code, computational output, explanatory text and multimedia resources in a single document.* (J. Perkel, *https://www.nature.com/articles/d41586-018-07196-1)*

## Common Jupyter Services:

- Jupyter Notebooks (single user)
- JupyterLab: advanced version of notebook
- JupyterHub: multiuser Jupyter service



# Accessing and Running Secure Notebooks on SDSC HPC Systems

- Install notebook application:
  - Locally: install Anaconda on your laptop
  - Remotely:
    - Install Anaconda/conda on the remote machine (default is HTTP) – not recommended
- Running remotely:
  - Connect over HTTP (default, insecure)
  - Connect over HTTP + SSH tunneling (secure, but inconvenient)
  - Connect over HTTPS + using the *Satellite Reverse Proxy Service* (SRPS) and *galyleo* client (secure, convenient)
- ➢ You can launch Jupyter services on SDSC:
  - Launch securely (HTTPS) using SRPS/*galyleo* -- *recommended*
  - CPU and GPUs
  - Interactive nodes: command line *or* Slurm batch script
- Treat the Notebook URL like a Password!

# SDSC Satellite Reverse Proxy Service

- SRPS: prototype system that allows users to launch secure standard Jupyter Notebooks on on any Expanse compute node using a reverse proxy server.
  - Notebooks will be hosted on the internal cluster network as an HTTP service using standard Jupyter commands.
  - Service available to the user outside of the cluster firewall over HTTPS connection between the external users web browser and the reverse proxy server.
- Goal: minimize software changes for users, improve security of user notebooks running on SDSC systems.
- SRPS can run on any HPC system capable of supporting Apache on internal network.

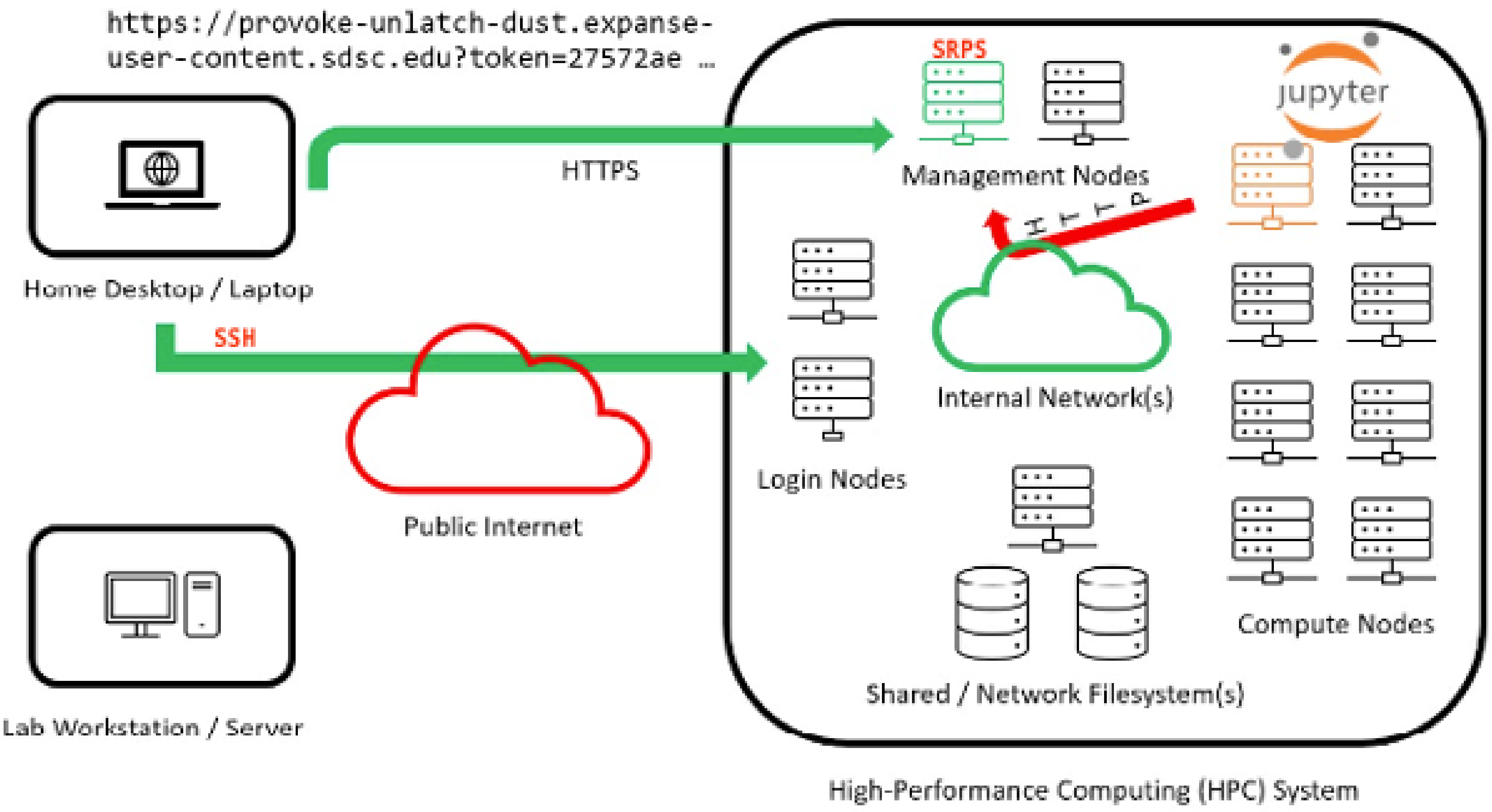
# galyleo

- 2nd generation shell utility developed to orchestrate a user’s interaction with both Satellite and Slurm to start a Jupyter session within a batch job.
- Developed while reviewing start-jupyter (prototype client) codebase to sort out how best to support Expanse (OOD) Portal and HPC User Services Group long-term; integrated into an existing SSH tunneling orchestration utility to use Satellite proxy service instead

- Key features in design:
  - HTTPS URL
  - Supports containers (Singularity on Expanse) □ **GPU environment**
  - No need to install conda environment or update packages
  - Increases flexibility for users to configure software environment; but also try to makes it simpler for them to do this themselves
  - Batch job script is generated completely on-the-fly.
  - Command-line argument driven.
  - Quiet mode for OOD portal

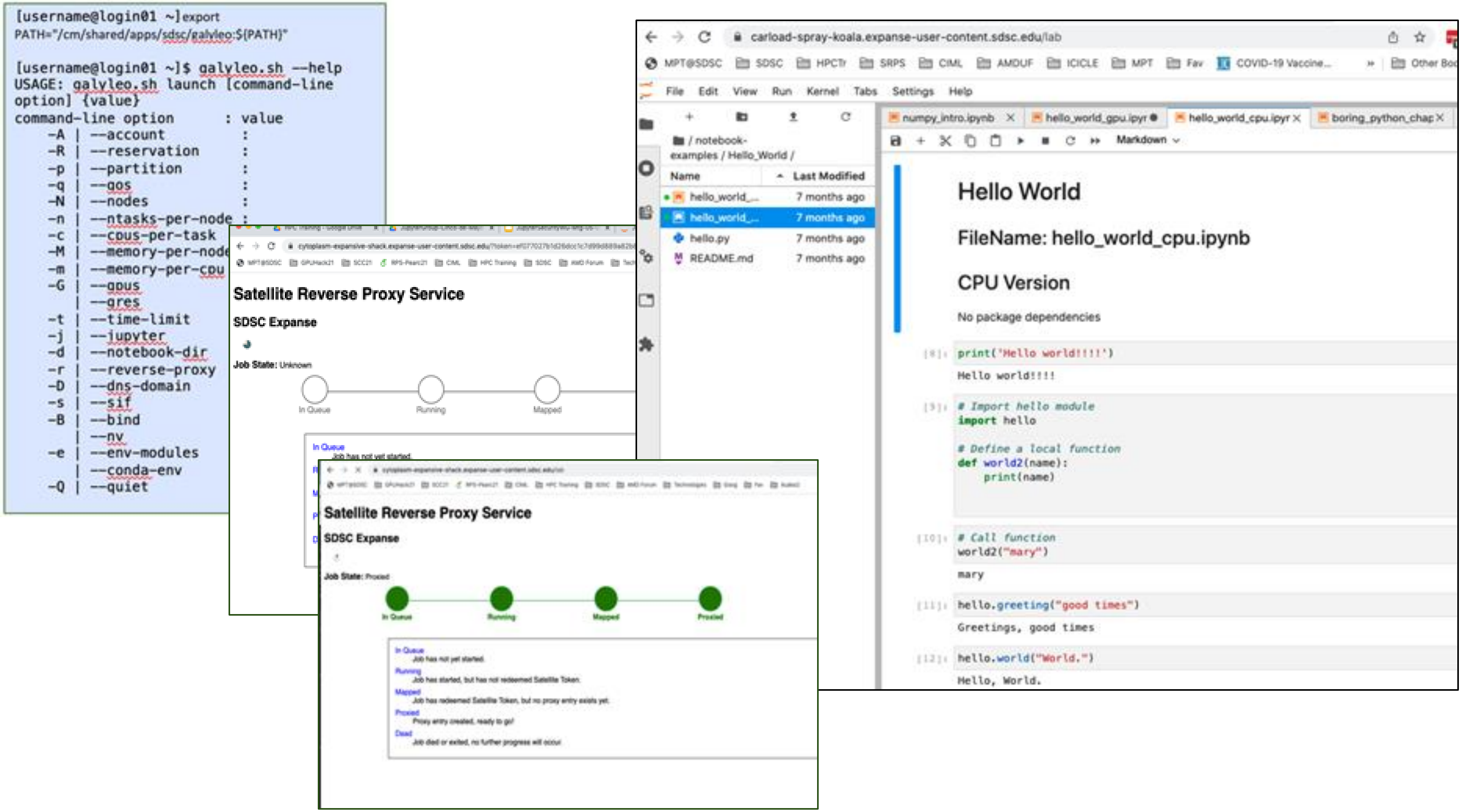
https://github.com/mkandes/galyleo

# Running Notebooks Remotely & Securely Using SRPS


https://provoke-unlatch-dust.expanse-user-content.sdsc.edu?token=27572ae …
Home Desktop / Laptop
HTTPS
SSH
Public Internet
Lab Workstation / Server
SRPS
Management Nodes
jupyter
HTTP
Login Nodes
Internal Network(s)
Compute Nodes
Shared / Network Filesystem(s)
High-Performance Computing (HPC) System

Src: M Kandes: https://education.sdsc.edu/training/interactive/202112_running_jupyter_notebooks_on_expanse/

# Satellite-Galyleo System



```
[username@login01 ~]export
PATH="/cm/shared/apps/sdsc/galvleo:${PATH}"

[username@login01 ~]$ galyleo.sh --help
USAGE: galyleo.sh launch [command-line
option] {value}
command-line option        : value
    -A | --account          :
    -R | --reservation      :
    -p | --partition        :
    -q | --qos              :
    -N | --nodes            :
    -n | --ntasks-per-node  :
    -c | --cpus-per-task
    -M | --memory-per-node
    -m | --memory-per-cpu
    -G | --gpus
       | --gres
    -t | --time-limit
    -j | --jupyter
    -d | --notebook-dir
    -r | --reverse-proxy
    -D | --dns-domain
    -s | --sif
    -B | --bind
       | --nv
    -e | --env-modules
       | --conda-env
    -Q | --quiet
```

**Satellite Reverse Proxy Service**

**SDSC Expanse**

**Job State:** Unknown

In Queue
Job has not yet started.

**Satellite Reverse Proxy Service**

**SDSC Expanse**

**Job State:** Proxied

In Queue
  Job has not yet started.
Running
  Job has started, but has not redeemed Satellite Token.
Mapped
  Job has redeemed Satellite Token, but no proxy entry exists yet.
Proxied
  Proxy entry created, ready to go!
Dead
  Job died or exited, no further progress will occur.

carload-spray-koala.expanse-user-content.sdsc.edu/lab
File Edit View Run Kernel Tabs Settings Help
numpy_intro.ipynb
hello_world_gpu.ipyr
hello_world_cpu.ipyr
boring_python_chap
Markdown

/ notebook-examples / Hello_World /

| Name | Last Modified |
|---|---|
| hello_world_... | 7 months ago |
| hello_world_... | 7 months ago |
| hello.py | 7 months ago |
| README.md | 7 months ago |

## Hello World

### FileName: hello_world_cpu.ipynb

### CPU Version

No package dependencies

```
[8]: print('Hello world!!!!')
Hello world!!!!

[9]: # Import hello module
     import hello

     # Define a local function
     def world2(name):
         print(name)

[10]: # Call function
      world2("mary")
mary

[11]: hello.greeting("good times")
Greetings, good times

[12]: hello.world("World.")
Hello, World.
```

# Satellite Client: galyleo

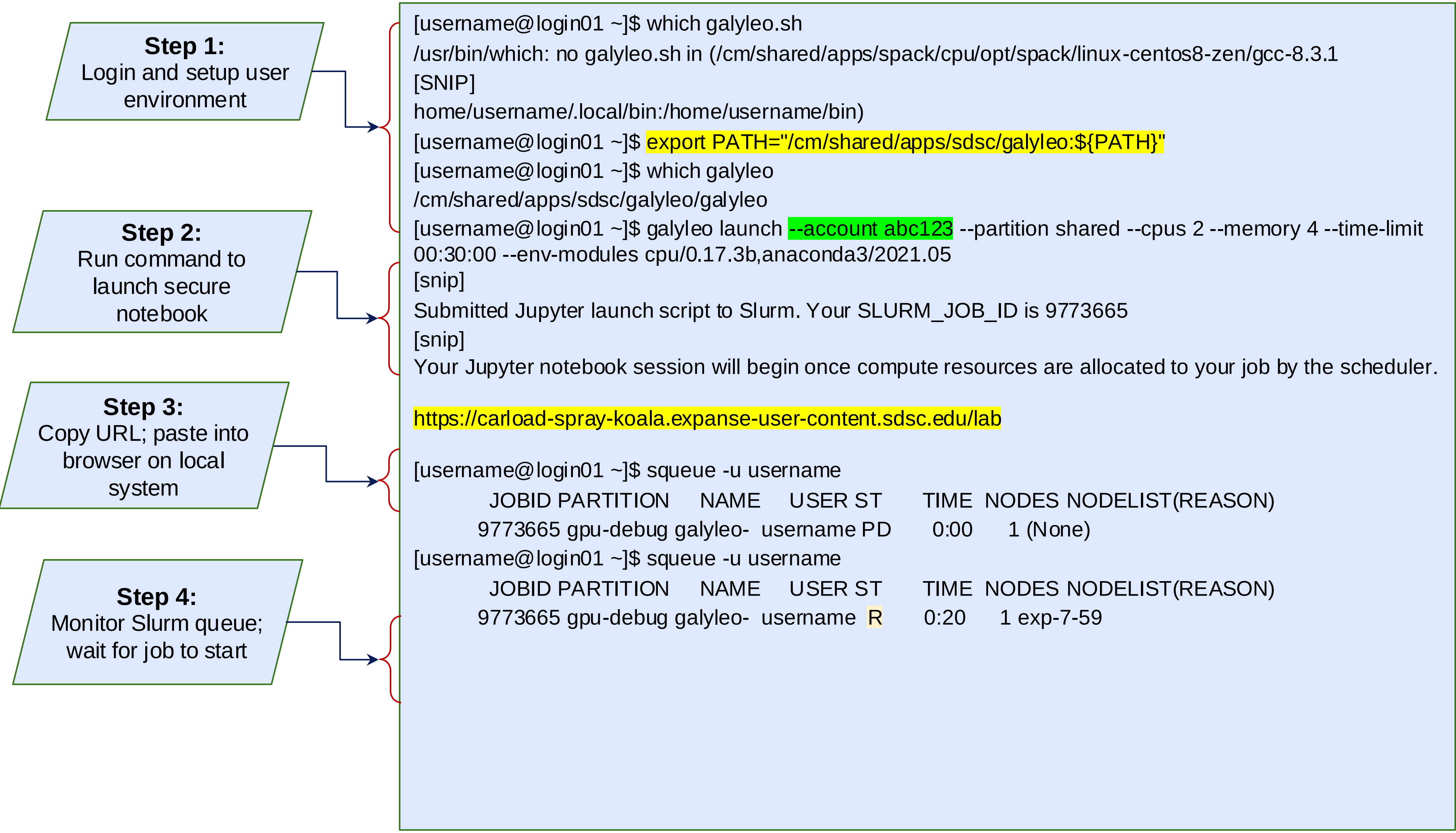
Key features in design:

- User calls galyleo.sh launch script, which requests token from Satellite, passes token to batch job script and submits the job to Slurm; token redeemed from batch job once it runs
- Increase flexibility for users to configure software environment; but also try to make it simpler for them to do themselves
- Batch job script is generated completely on-the-fly.
- Command-line argument driven.
- Quiet mode for OOD portal

```
[username@login01 ~]export PATH="/cm/shared/apps/sdsc/galyleo:${PATH}"

[username@login01 ~]$ galyleo.sh --help
USAGE: galyleo.sh launch [command-line option] {value}
command-line option      : value
   -A | --account          :
   -R | --reservation      :
   -p | --partition        :
   -q | --qos              :
   -N | --nodes            :
   -n | --ntasks-per-node :
   -c | --cpus-per-task   :
   -M | --memory-per-node :  GB
   -m | --memory-per-cpu  :  GB
   -G | --gpus             :
      | --gres             :
   -t | --time-limit       :
   -j | --jupyter          :
   -d | --notebook-dir     :
   -r | --reverse-proxy    :
   -D | --dns-domain       :
   -s | --sif              :
   -B | --bind             :
      | --nv               :
   -e | --env-modules      :
      | --conda-env        :
   -Q | --quiet            : 1
```

https://github.com/mkandes/galyleo

# Launching CPU notebooks using galyleo



**Step 1:**
Login and setup user environment

**Step 2:**
Run command to launch secure notebook

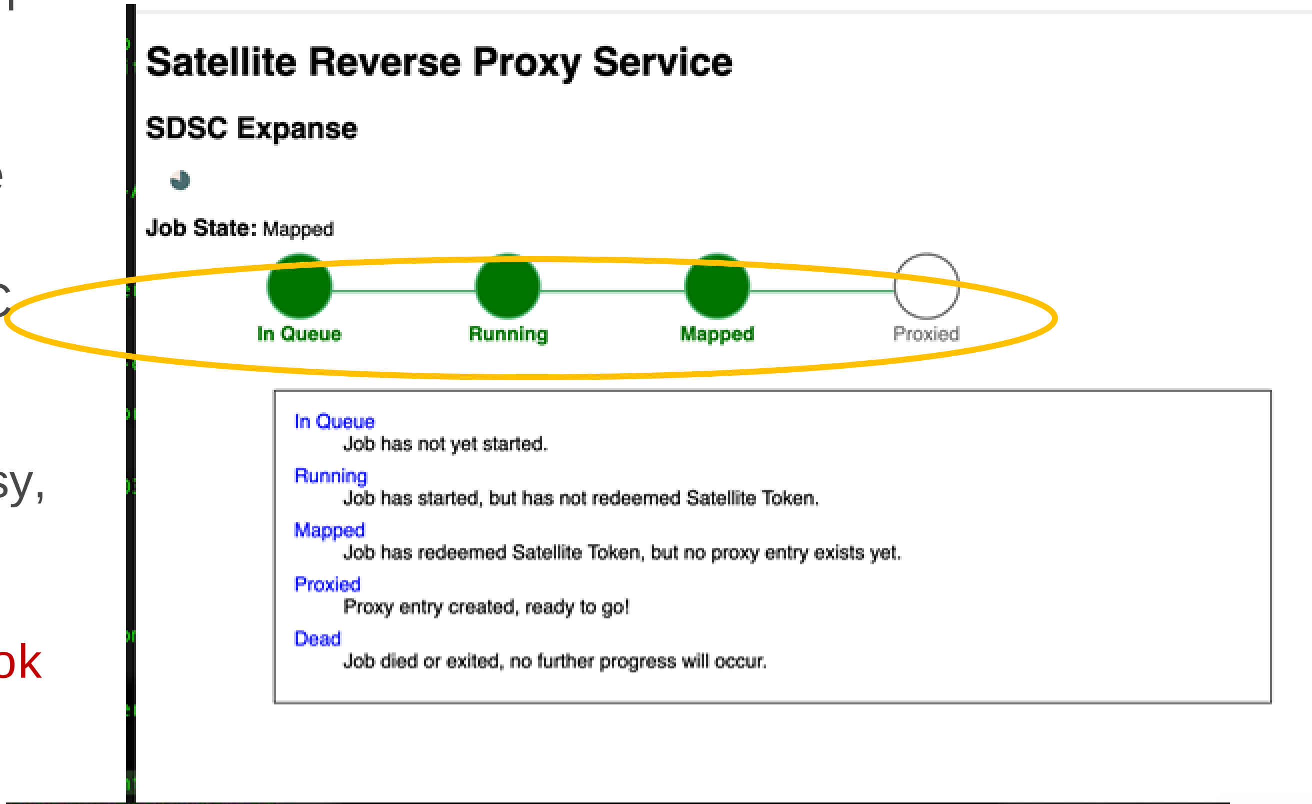
**Step 3:**
Copy URL; paste into browser on local system

**Step 4:**
Monitor Slurm queue; wait for job to start

```
[username@login01 ~]$ which galyleo.sh
/usr/bin/which: no galyleo.sh in (/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen/gcc-8.3.1
[SNIP]
home/username/.local/bin:/home/username/bin)
[username@login01 ~]$ export PATH="/cm/shared/apps/sdsc/galyleo:${PATH}"
[username@login01 ~]$ which galyleo
/cm/shared/apps/sdsc/galyleo/galyleo
[username@login01 ~]$ galyleo launch --account abc123 --partition shared --cpus 2 --memory 4 --time-limit
00:30:00 --env-modules cpu/0.17.3b,anaconda3/2021.05
[snip]
Submitted Jupyter launch script to Slurm. Your SLURM_JOB_ID is 9773665
[snip]
Your Jupyter notebook session will begin once compute resources are allocated to your job by the scheduler.

https://carload-spray-koala.expanse-user-content.sdsc.edu/lab

[username@login01 ~]$ squeue -u username
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
           9773665 gpu-debug galyleo-  username PD       0:00      1 (None)
[username@login01 ~]$ squeue -u username
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
           9773665 gpu-debug galyleo-  username  R       0:20      1 exp-7-59
```

# Satellite Server Status/Pending Page

- Load notebook URL in browser; wait for it to launch
- Monitor pending page
- Run the “squeue” command on the HPC system to check job status
- If the job queue is busy, it may take a while to launch the notebook
- Treat Jupyter Notebook URL as a password!



## Satellite Reverse Proxy Service

### SDSC Expanse

**Job State:** Mapped

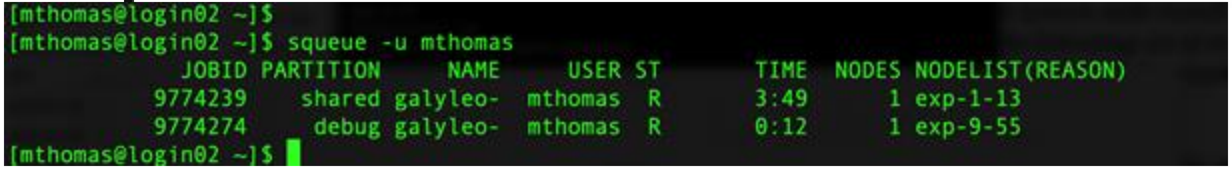
In Queue
    Job has not yet started.

Running
    Job has started, but has not redeemed Satellite Token.

Mapped
    Job has redeemed Satellite Token, but no proxy entry exists yet.

Proxied
    Proxy entry created, ready to go!

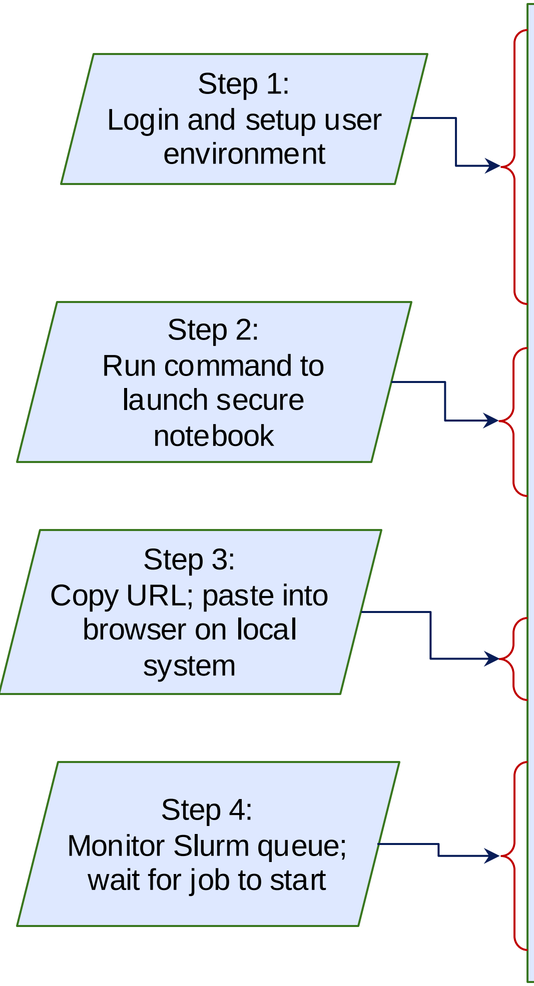
Dead
    Job died or exited, no further progress will occur.



```
[mthomas@login02 ~]$
[mthomas@login02 ~]$ squeue -u mthomas
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
           9774239    shared galyleo-  mthomas  R       3:49      1 exp-1-13
           9774274     debug galyleo-  mthomas  R       0:12      1 exp-9-55
[mthomas@login02 ~]$
```

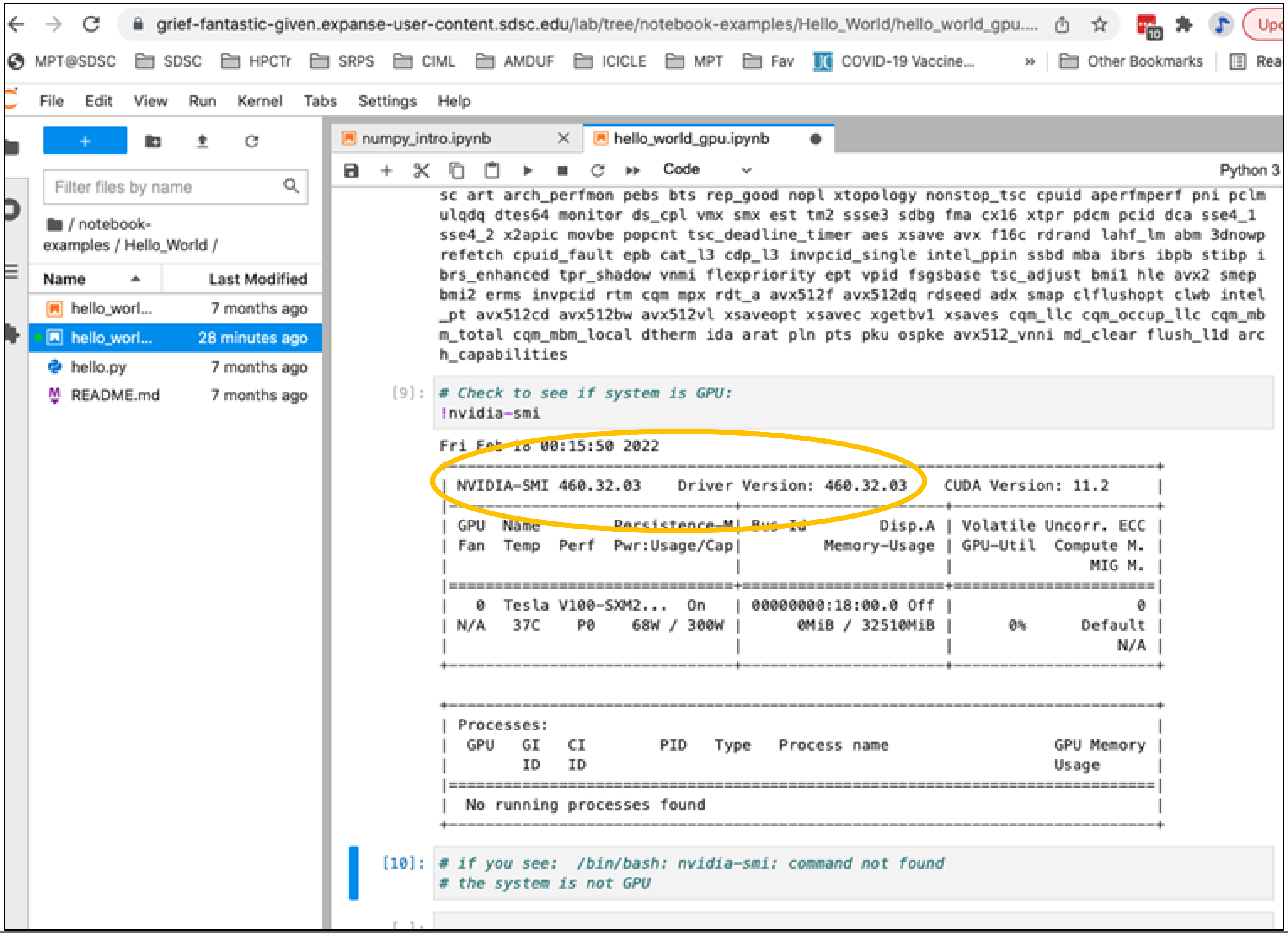
# Launching GPU notebooks using galyleo

- GPU Notebooks run better when using containers. SDSC maintains several containers on Expanse
- See: /cm/shared/apps/containers/singularity


Step 1:
Login and setup user environment
Step 2:
Run command to launch secure notebook
Step 3:
Copy URL; paste into browser on local system
Step 4:
Monitor Slurm queue; wait for job to start

```
[username@login01 ~]$ which galyleo.sh
/usr/bin/which: no galyleo.sh in (/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen/gcc-8.3.1
[SNIP]
home/username/.local/bin:/home/username/bin)
[username@login01 ~]$ export PATH="/cm/shared/apps/sdsc/galyleo:${PATH}"
[username@login01 ~]$ which galyleo
/cm/shared/apps/sdsc/galyleo/galyleo
[username@login01 ~]$ galyleo launch --account use300 --partition gpu-debug --cpus 1 --memory 93 --gpus
1 --time-limit 00:5:00 --env-modules singularitypro --sif
/cm/shared/apps/containers/singularity/pytorch/pytorch-latest.sif --bind /expanse,/scratch --nv
[snip]
Submitted Jupyter launch script to Slurm. Your SLURM_JOB_ID is 9773912
[snip]
Your Jupyter notebook session will begin once compute resources are allocated to your job by the scheduler.
https://grief-fantastic-given.expanse-user-content.sdsc.edu?token=5097acb6f32ab82dd51b4524c267d2fd

[username@login01 ~]$ squeue -u username
          JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
        9773912 gpu-debug galyleo-  username PD       0:00      1 (None)
[username@login01 ~]$ squeue -u username
          JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
        9773912 gpu-debug galyleo-  username  R       0:20      1 exp-7-59
```

# Verify notebook launched on GPU device



/ notebook-examples / Hello_World /

| Name | Last Modified |
|---|---|
| hello_worl... | 7 months ago |
| hello_worl... | 28 minutes ago |
| hello.py | 7 months ago |
| README.md | 7 months ago |

```
sc art arch_perfmon pebs bts rep_good nopl xtopology nonstop_tsc cpuid aperfmperf pni pclm
ulqdq dtes64 monitor ds_cpl vmx smx est tm2 ssse3 sdbg fma cx16 xtpr pdcm pcid dca sse4_1
sse4_2 x2apic movbe popcnt tsc_deadline_timer aes xsave avx f16c rdrand lahf_lm abm 3dnowp
refetch cpuid_fault epb cat_l3 cdp_l3 invpcid_single intel_ppin ssbd mba ibrs ibpb stibp i
brs_enhanced tpr_shadow vnmi flexpriority ept vpid fsgsbase tsc_adjust bmi1 hle avx2 smep
bmi2 erms invpcid rtm cqm mpx rdt_a avx512f avx512dq rdseed adx smap clflushopt clwb intel
_pt avx512cd avx512bw avx512vl xsaveopt xsavec xgetbv1 xsaves cqm_llc cqm_occup_llc cqm_mb
m_total cqm_mbm_local dtherm ida arat pln pts pku ospke avx512_vnni md_clear flush_l1d arc
h_capabilities
```

```
[9]:  # Check to see if system is GPU:
      !nvidia-smi
```

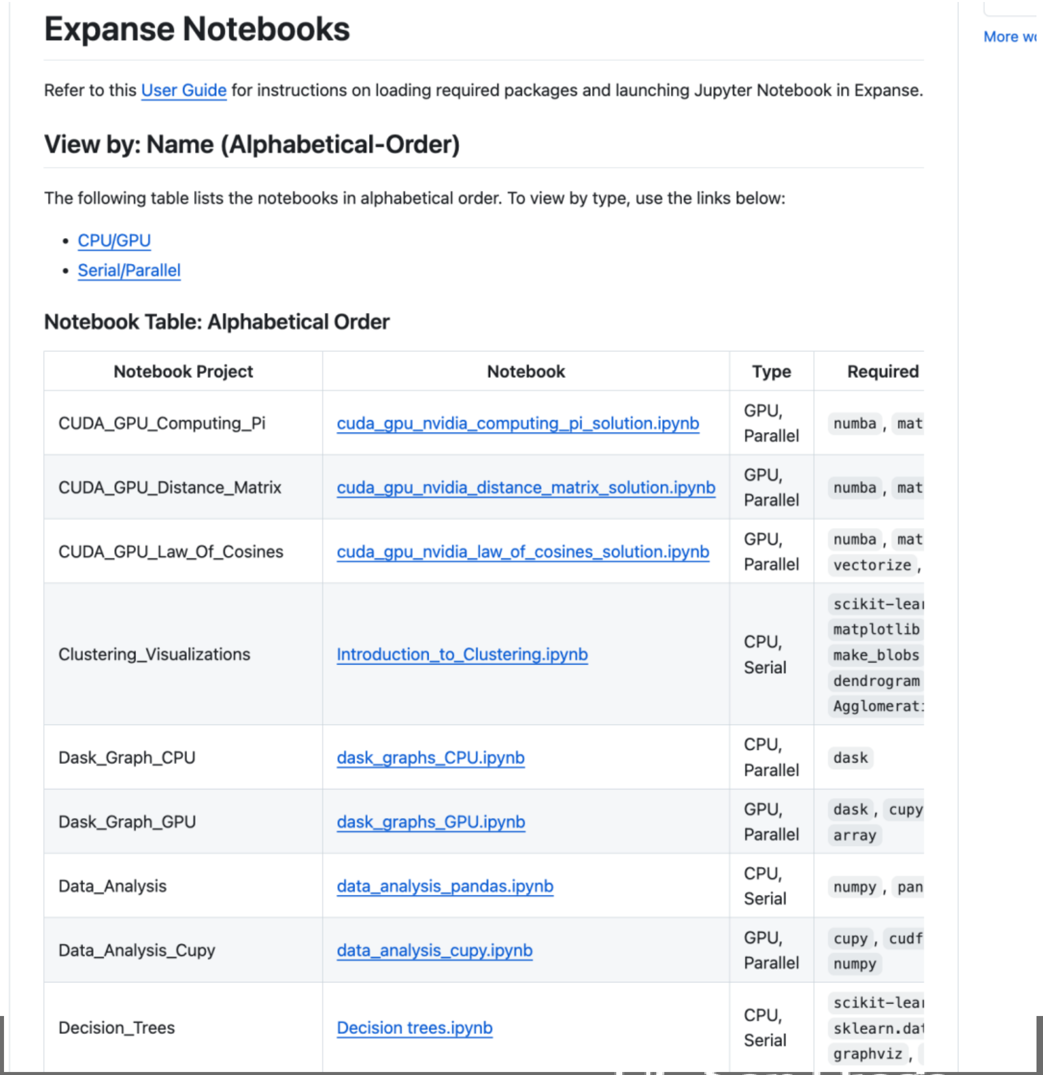
```
Fri Feb 18 00:15:50 2022
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 460.32.03    Driver Version: 460.32.03    CUDA Version: 11.2     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  Tesla V100-SXM2...  On   | 00000000:18:00.0 Off |                    0 |
| N/A   37C    P0    68W / 300W |      0MiB / 32510MiB |      0%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|  No running processes found                                                 |
+-----------------------------------------------------------------------------+
```

```
[10]: # if you see:  /bin/bash: nvidia-smi: command not found
      # the system is not GPU
```

# Notebook Examples

https://github.com/sdsc-hpc-training-org/Expanse-Notebooks

- Collection of notebooks tested on Expanse and other SDSC HPC systems
- Includes range of materials from “hello world” to ML notebooks.
- View by name, CPU/GPU, or Serial/Parallel



## Expanse Notebooks

Refer to this User Guide for instructions on loading required packages and launching Jupyter Notebook in Expanse.

### View by: Name (Alphabetical-Order)

The following table lists the notebooks in alphabetical order. To view by type, use the links below:

- CPU/GPU
- Serial/Parallel

**Notebook Table: Alphabetical Order**

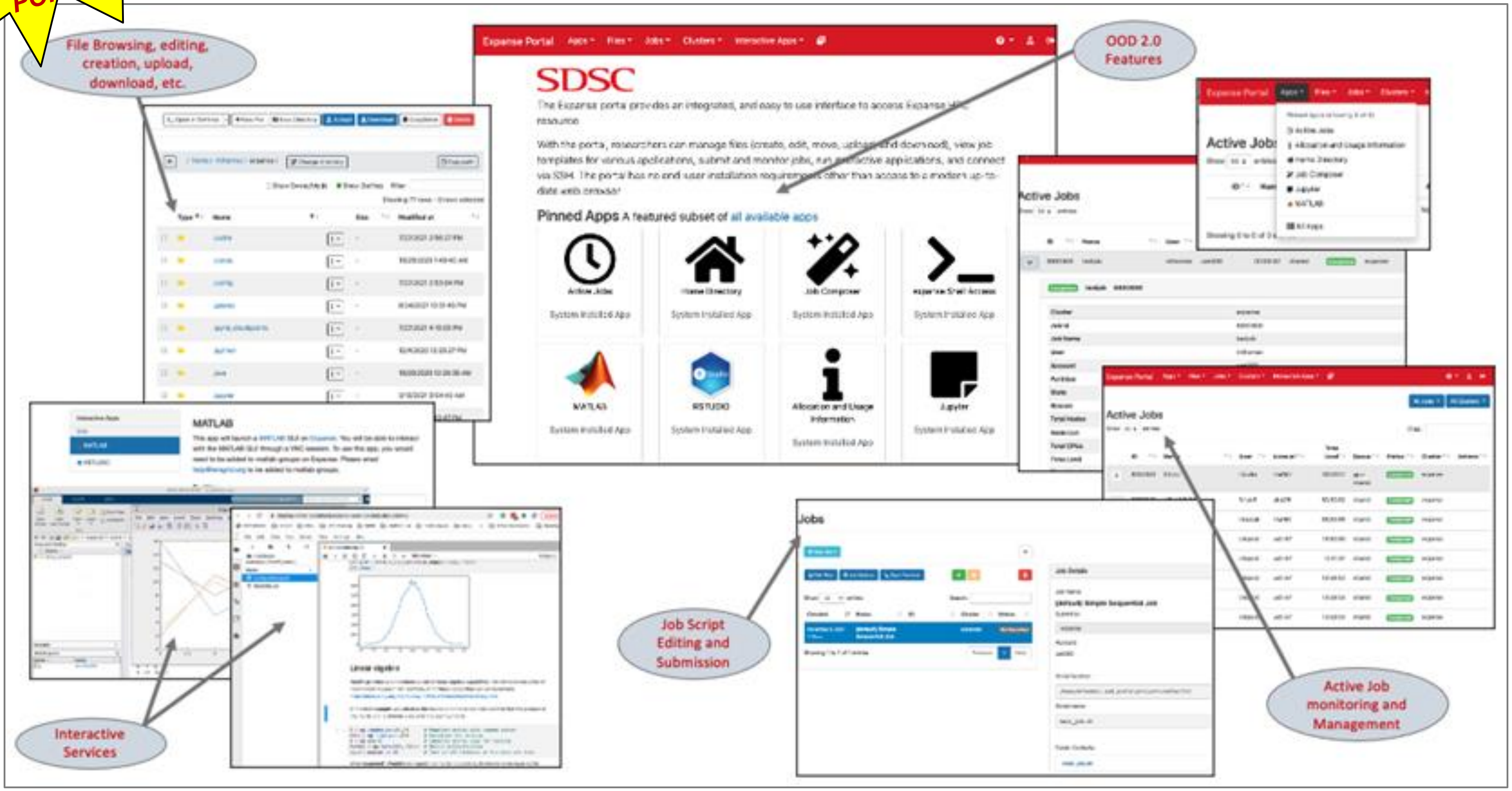
| Notebook Project | Notebook | Type | Required |
|---|---|---|---|
| CUDA_GPU_Computing_Pi | cuda_gpu_nvidia_computing_pi_solution.ipynb | GPU, Parallel | numba , mat |
| CUDA_GPU_Distance_Matrix | cuda_gpu_nvidia_distance_matrix_solution.ipynb | GPU, Parallel | numba , mat |
| CUDA_GPU_Law_Of_Cosines | cuda_gpu_nvidia_law_of_cosines_solution.ipynb | GPU, Parallel | numba , mat<br>vectorize , |
| Clustering_Visualizations | Introduction_to_Clustering.ipynb | CPU, Serial | scikit-lea<br>matplotlib<br>make_blobs<br>dendrogram<br>Agglomerat |
| Dask_Graph_CPU | dask_graphs_CPU.ipynb | CPU, Parallel | dask |
| Dask_Graph_GPU | dask_graphs_GPU.ipynb | GPU, Parallel | dask , cupy<br>array |
| Data_Analysis | data_analysis_pandas.ipynb | CPU, Serial | numpy , pan |
| Data_Analysis_Cupy | data_analysis_cupy.ipynb | GPU, Parallel | cupy , cudf<br>numpy |
| Decision_Trees | Decision trees.ipynb | CPU, Serial | scikit-lea<br>sklearn.da<br>graphviz , |

# Outline

- Defining Interactive HPC
  - High-performance computing (HPC)
  - HPC batch computing
  - Interactive computing
- Accessing Interactive HPC Nodes
  - Launching nodes
  - Running GUIs using X11 forwarding
- Interactive Application Examples
  - Viewing Data: unix file ops (grep, awk, cat), gnuplot, NetCDF
  - Programming & Visualization Platforms: Matlab, R, Jupyter Notebooks
  - Gateways & Portals: simplify access to interactive apps
- Q&A

SIMPLIFY! USE THE EXPANSE PORTAL

# Expanse User Portal



The Expanse portal provides an integrated, and easy to use interface to access Expanse HPC resources.

With the portal, researchers can manage files (create, edit, move, upload and download), view job templates for various applications, submit and monitor jobs, run interactive applications, and connect via SSH. The portal has no end user installation requirements other than access to a modern up-to-date web browser.

Pinned Apps A featured subset of all available apps

- https://portal.expanse.sdsc.edu; authenticate using ACCESS credentials
- Securely hosts batch job submission & monitoring, and interactive applications
- Simplifies launching supported interactive applications  manages software dependencies

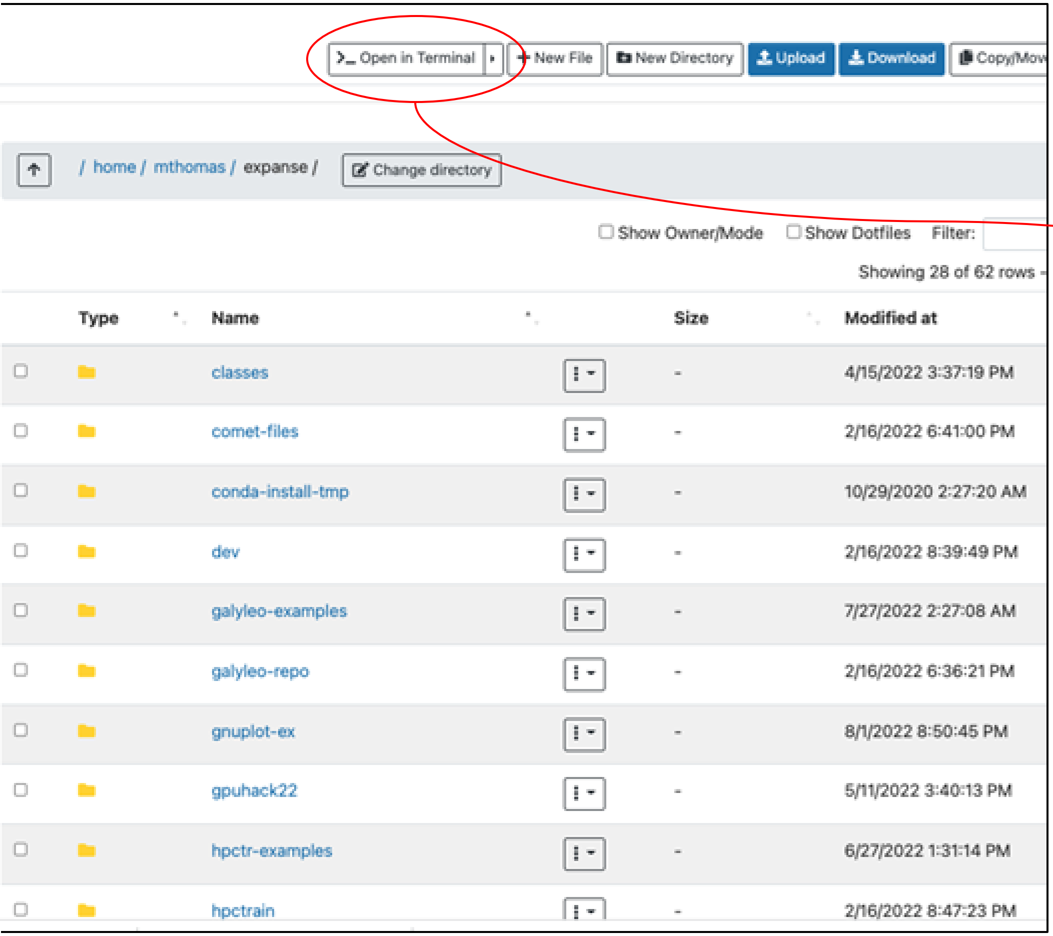
SDSC SAN DIEGO SUPERCOMPUTER CENTER

Sivagnanam, S., Irving, C., Kandes, M., Mishin, D., Sakai, S., Strande, S., Tatineni, M., Thomas, M., Norman, M. (2021). Experiences in building a user portal for Expanse supercomputer. *PEARC21: Practice and Experience in Advanced Research Computing*, Accepted: ACM. https://doi.org/10.1145/3437359.3465590

*COMPLECS NSF award #2320934*

UC San Diego

# Expanse Portal: File Management

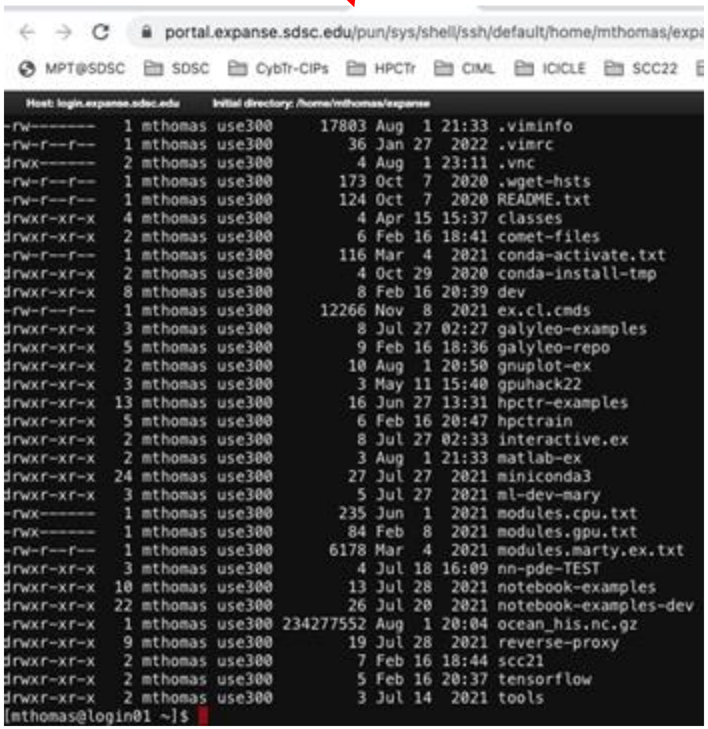


/ home / mthomas / expanse /

□ Show Owner/Mode □ Show Dotfiles Filter:

Showing 28 of 62 rows -

| | Type | Name | | Size | Modified at |
|---|---|---|---|---|---|
| □ | 📁 | classes | ⋮ ▾ | - | 4/15/2022 3:37:19 PM |
| □ | 📁 | comet-files | ⋮ ▾ | - | 2/16/2022 6:41:00 PM |
| □ | 📁 | conda-install-tmp | ⋮ ▾ | - | 10/29/2020 2:27:20 AM |
| □ | 📁 | dev | ⋮ ▾ | - | 2/16/2022 8:39:49 PM |
| □ | 📁 | galyleo-examples | ⋮ ▾ | - | 7/27/2022 2:27:08 AM |
| □ | 📁 | galyleo-repo | ⋮ ▾ | - | 2/16/2022 6:36:21 PM |
| □ | 📁 | gnuplot-ex | ⋮ ▾ | - | 8/1/2022 8:50:45 PM |
| □ | 📁 | gpuhack22 | ⋮ ▾ | - | 5/11/2022 3:40:13 PM |
| □ | 📁 | hpctr-examples | ⋮ ▾ | - | 6/27/2022 1:31:14 PM |
| □ | 📁 | hpctrain | ⋮ ▾ | - | 2/16/2022 8:47:23 PM |

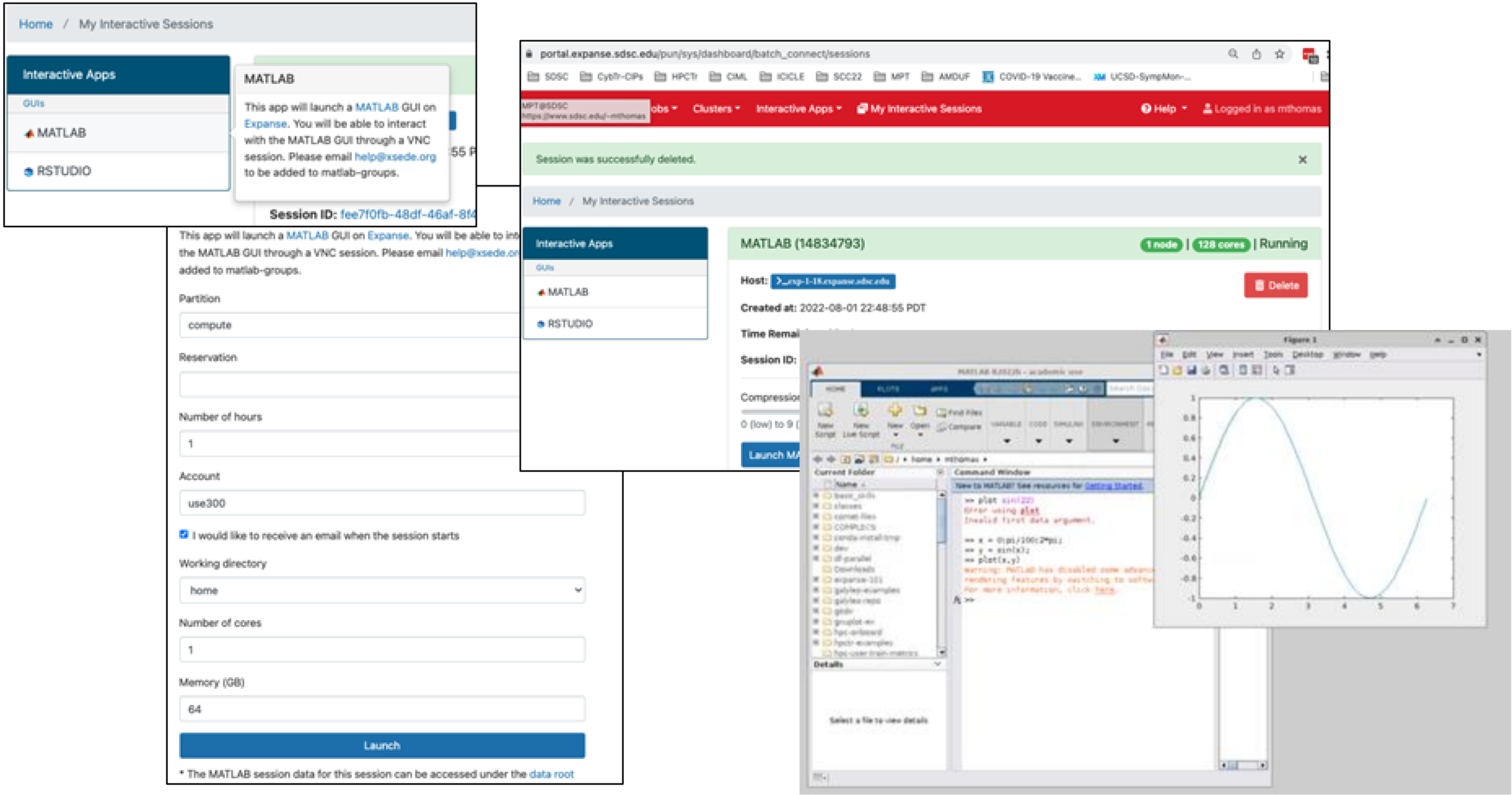


portal.expanse.sdsc.edu/pun/sys/shell/ssh/default/home/mthomas/expa
MPT@SDSC
SDSC
CybTr-CIPs
HPCTr
CIML
ICICLE
SCC22

Host: login.expanse.sdsc.edu    Initial directory: /home/mthomas/expanse

```
-rw-------   1 mthomas use300     17803 Aug  1 21:33 .viminfo
-rw-r--r--   1 mthomas use300        36 Jan 27  2022 .vimrc
drwx------   2 mthomas use300         4 Aug  1 23:11 .vnc
-rw-r--r--   1 mthomas use300       173 Oct  7  2020 .wget-hsts
-rw-r--r--   1 mthomas use300       124 Oct  7  2020 README.txt
drwxr-xr-x   4 mthomas use300         4 Apr 15 15:37 classes
drwxr-xr-x   2 mthomas use300         6 Feb 16 18:41 comet-files
-rw-r--r--   1 mthomas use300       116 Mar  4  2021 conda-activate.txt
drwxr-xr-x   2 mthomas use300         4 Oct 29  2020 conda-install-tmp
drwxr-xr-x   8 mthomas use300         8 Feb 16 20:39 dev
-rw-r--r--   1 mthomas use300     12266 Nov  8  2021 ex.cl.cmds
drwxr-xr-x   3 mthomas use300         8 Jul 27 02:27 galyleo-examples
drwxr-xr-x   5 mthomas use300         9 Feb 16 18:36 galyleo-repo
drwxr-xr-x   2 mthomas use300        10 Aug  1 20:50 gnuplot-ex
drwxr-xr-x   3 mthomas use300         3 May 11 15:40 gpuhack22
drwxr-xr-x  13 mthomas use300        16 Jun 27 13:31 hpctr-examples
drwxr-xr-x   5 mthomas use300         6 Feb 16 20:47 hpctrain
drwxr-xr-x   2 mthomas use300         8 Jul 27 02:33 interactive.ex
drwxr-xr-x   2 mthomas use300         3 Aug  1 21:33 matlab-ex
drwxr-xr-x  24 mthomas use300        27 Jul 27  2021 miniconda3
drwxr-xr-x   3 mthomas use300         5 Jul 27  2021 ml-dev-mary
-rwx------   1 mthomas use300       235 Jun  1  2021 modules.cpu.txt
-rwx------   1 mthomas use300        84 Feb  8  2021 modules.gpu.txt
-rw-r--r--   1 mthomas use300      6178 Mar  4  2021 modules.marty.ex.txt
drwxr-xr-x   3 mthomas use300         4 Jul 18 16:09 nn-pde-TEST
drwxr-xr-x  10 mthomas use300        13 Jul 28  2021 notebook-examples
drwxr-xr-x  22 mthomas use300        26 Jul 20  2021 notebook-examples-dev
-rwxr-xr-x   1 mthomas use300 234277552 Aug  1 20:04 ocean_his.nc.gz
drwxr-xr-x   9 mthomas use300        19 Jul 28  2021 reverse-proxy
drwxr-xr-x   2 mthomas use300         7 Feb 16 18:44 scc21
drwxr-xr-x   2 mthomas use300         5 Feb 16 20:37 tensorflow
drwxr-xr-x   2 mthomas use300         3 Jul 14  2021 tools
[mthomas@login01 ~]$ 
```

# Expanse Portal: Running Matlab



MATLAB

This app will launch a MATLAB GUI on Expanse. You will be able to interact with the MATLAB GUI through a VNC session. Please email help@xsede.org to be added to matlab-groups.

Session ID: fee7f0fb-48df-46af-8f4

This app will launch a MATLAB GUI on Expanse. You will be able to int the MATLAB GUI through a VNC session. Please email help@xsede.or added to matlab-groups.

Partition: compute

Reservation:

Number of hours: 1

Account: use300

☑ I would like to receive an email when the session starts

Working directory: home

Number of cores: 1

Memory (GB): 64

Launch

* The MATLAB session data for this session can be accessed under the data root

portal.expanse.sdsc.edu/pun/sys/dashboard/batch_connect/sessions

SDSC CybT-CIPs HPCTr CIML ICICLE SCC22 MPT AMDUF COVID-19 Vaccine... UCSD-SympMon-...

MPT@SDSC https://www.sdsc.edu/~mthomas

Session was successfully deleted.

MATLAB (14834793) | 1 node | 128 cores | Running

Host: >_exp-1-18.expanse.sdsc.edu

Created at: 2022-08-01 22:48:55 PDT

Time Remai

Session ID:

Compressio

0 (low) to 9 (

MATLAB R2022b - academic use

/ ▸ home ▸ mthomas ▸

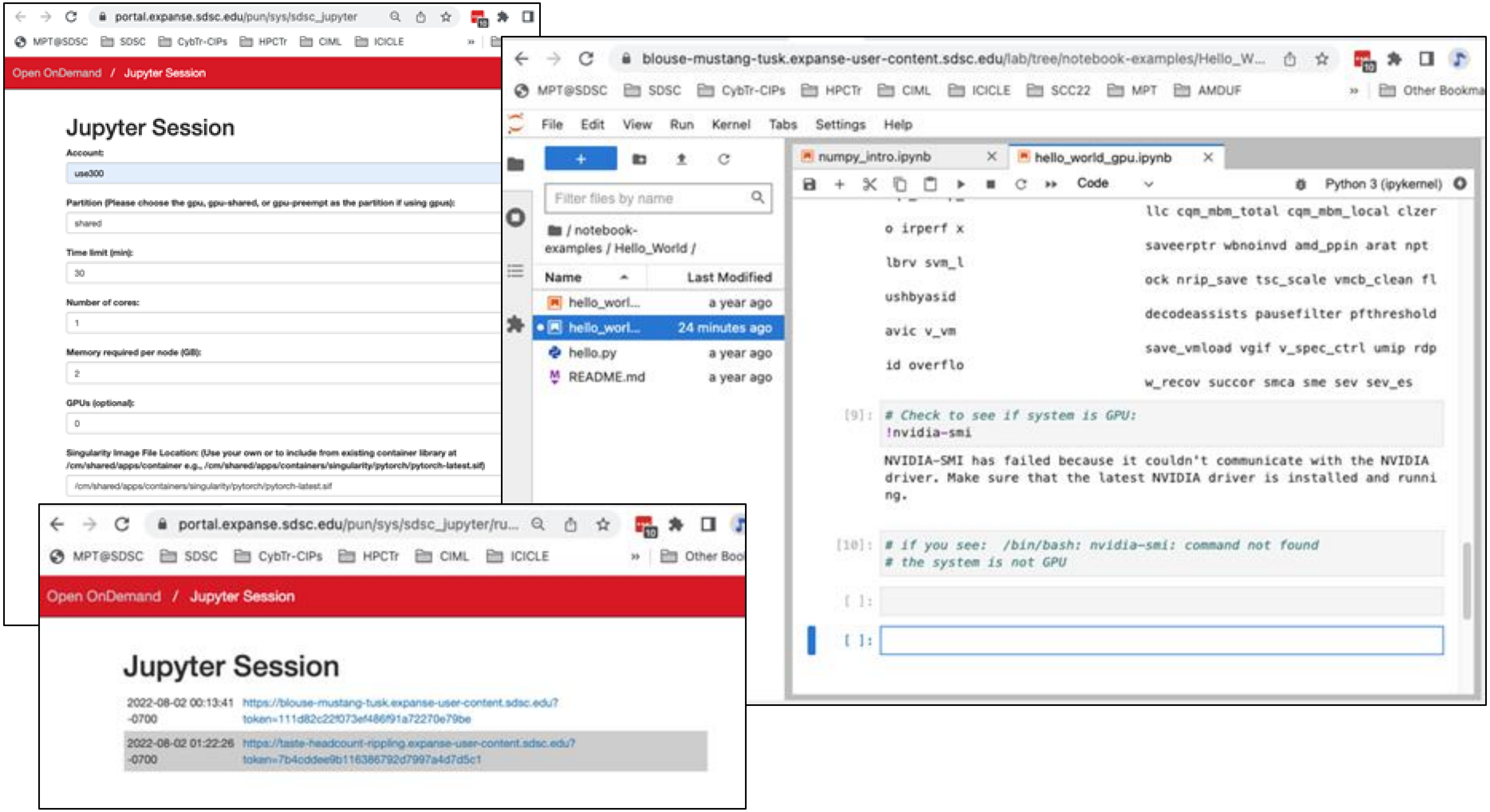
base_skills, classes, cornet-files, COMPLECS, conda-install-tmp, dev, df-parallel, Downloads, expanse-101, gatyles-examples, galyles-repo, gitdir, gnuplot-ex, hpc-onboard, hpctr-examples, hpc-user-train-metrics

Select a file to view details

New to MATLAB? See resources for Getting Started.

```
>> plot sin(22)
Error using plot
Invalid first data argument.

>> x = 0:pi/100:2*pi;
>> y = sin(x);
>> plot(x,y)
Warning: MATLAB has disabled some advanc
rendering features by switching to softw
For more information, click here.
>>
```

# Expanse Portal: Launching Notebooks


portal.expanse.sdsc.edu/pun/sys/sdsc_jupyter
MPT@SDSC
SDSC
CybTr-CIPs
HPCTr
CIML
ICICLE
Open OnDemand / Jupyter Session

## Jupyter Session

Account:
use300

Partition (Please choose the gpu, gpu-shared, or gpu-preempt as the partition if using gpus):
shared

Time limit (min):
30

Number of cores:
1

Memory required per node (GB):
2

GPUs (optional):
0

Singularity Image File Location: (Use your own or to include from existing container library at /cm/shared/apps/container e.g., /cm/shared/apps/containers/singularity/pytorch/pytorch-latest.sif)
/cm/shared/apps/containers/singularity/pytorch/pytorch-latest.sif

portal.expanse.sdsc.edu/pun/sys/sdsc_jupyter/ru...
MPT@SDSC
SDSC
CybTr-CIPs
HPCTr
CIML
ICICLE
Other Boo
Open OnDemand / Jupyter Session

## Jupyter Session

| | |
|---|---|
| 2022-08-02 00:13:41 -0700 | https://blouse-mustang-tusk.expanse-user-content.sdsc.edu?token=111d82c22f073ef486f91a72270e79be |
| 2022-08-02 01:22:26 -0700 | https://taste-headcount-rippling.expanse-user-content.sdsc.edu?token=7b4cddee9b116386792d7997a4d7d5c1 |

| Name | Last Modified |
|---|---|
| hello_worl... | a year ago |
| hello_worl... | 24 minutes ago |
| hello.py | a year ago |
| README.md | a year ago |

```
                                       llc cqm_mbm_total cqm_mbm_local clzer
o irperf x
                                       saveerptr wbnoinvd amd_ppin arat npt
lbrv svm_l
                                       ock nrip_save tsc_scale vmcb_clean fl
ushbyasid
                                       decodeassists pausefilter pfthreshold
avic v_vm
                                       save_vmload vgif v_spec_ctrl umip rdp
id overflo
                                       w_recov succor smca sme sev sev_es

[9]:  # Check to see if system is GPU:
      !nvidia-smi

      NVIDIA-SMI has failed because it couldn't communicate with the NVIDIA
      driver. Make sure that the latest NVIDIA driver is installed and runni
      ng.

[10]: # if you see:  /bin/bash: nvidia-smi: command not found
      # the system is not GPU

[ ]:

[ ]:
```

# Expanse Portal: Input data example for Jupyter Notebook

**Account:** use300

**Partition:** *(Please choose the gpu, gpu-shared, or gpu-preempt as the partition if using gpus): debug*

*Time limit (min)*: 30

**Number of cores:** 1

**Memory required per node (GB):** 2

**GPUs (optional):** 0

**Singularity Image File Location:** *(Use your own or to include from existing container library at /cm/shared/apps/container e.g., /cm/shared/apps/containers/singularity/pytorch/pytorch-latest.sif)*
/cm/shared/apps/containers/singularity/pytorch/pytorch-latest.sif

**Environment modules to be loaded** *(E.g., to use latest version of system Anaconda3 include cpu,gcc,anaconda3):* singularitypro

**Conda Environment** (Enter your own conda environment if any):

**Conda Init** (Provide path to conda initialization scripts):

**Conda Yaml** *(Upload a yaml file to build the conda environment at runtime) No file chosen:*

***Turn on use of mamba for speeding up conda-yml installs:***

***Enable use of new caching mechanism that will store and reuse conda-yml created environments using conda-pack !????***

**Reservation:**

**QoS:**

**Working directory:** HOME

**Type:** JupyterLab

# Outline

- Defining Interactive HPC
  - High-performance computing (HPC)
  - HPC batch computing
  - Interactive computing
- Accessing Interactive HPC Nodes
  - Launching nodes
  - Running GUIs using X11 forwarding
- Interactive Application Examples
  - Viewing Data: unix file ops (grep, awk, cat), gnuplot, NetCDF
  - Programming & Visualization Platforms: Matlab, R, Jupyter Notebooks
  - Gateways & Portals: simplify access to interactive apps
- Q&A

# Thank You!

# Q&A

If you have problems, please contact **consult@sdsc.edu**

https://github.com/sdsc-complecs/interactive-computing/

# Resources

- GitHub Repo for this presentation:
  - https://github.com/sdsc-complecs/interactive-computing/
- SDSC Training Resources
  - https://www.sdsc.edu/education_and_training/training
  - Code: https://github.com/sdsc-hpc-training-org/hpctr-examples
  - Running notebooks
    - using *galyleo:* https://github.com/mkandes/galyleo
    - Examples: https://github.com/sdsc-hpc-training-org/notebook-examples
- Expanse :
  - Landing page: expanse.sdsc.edu
  - User Guide: https://expanse.sdsc.edu/support/user_guides/expanse.html
  - Training: https://www.sdsc.edu/education_and_training/training_hpc.html
- Problems? Contact consult@sdsc.edu