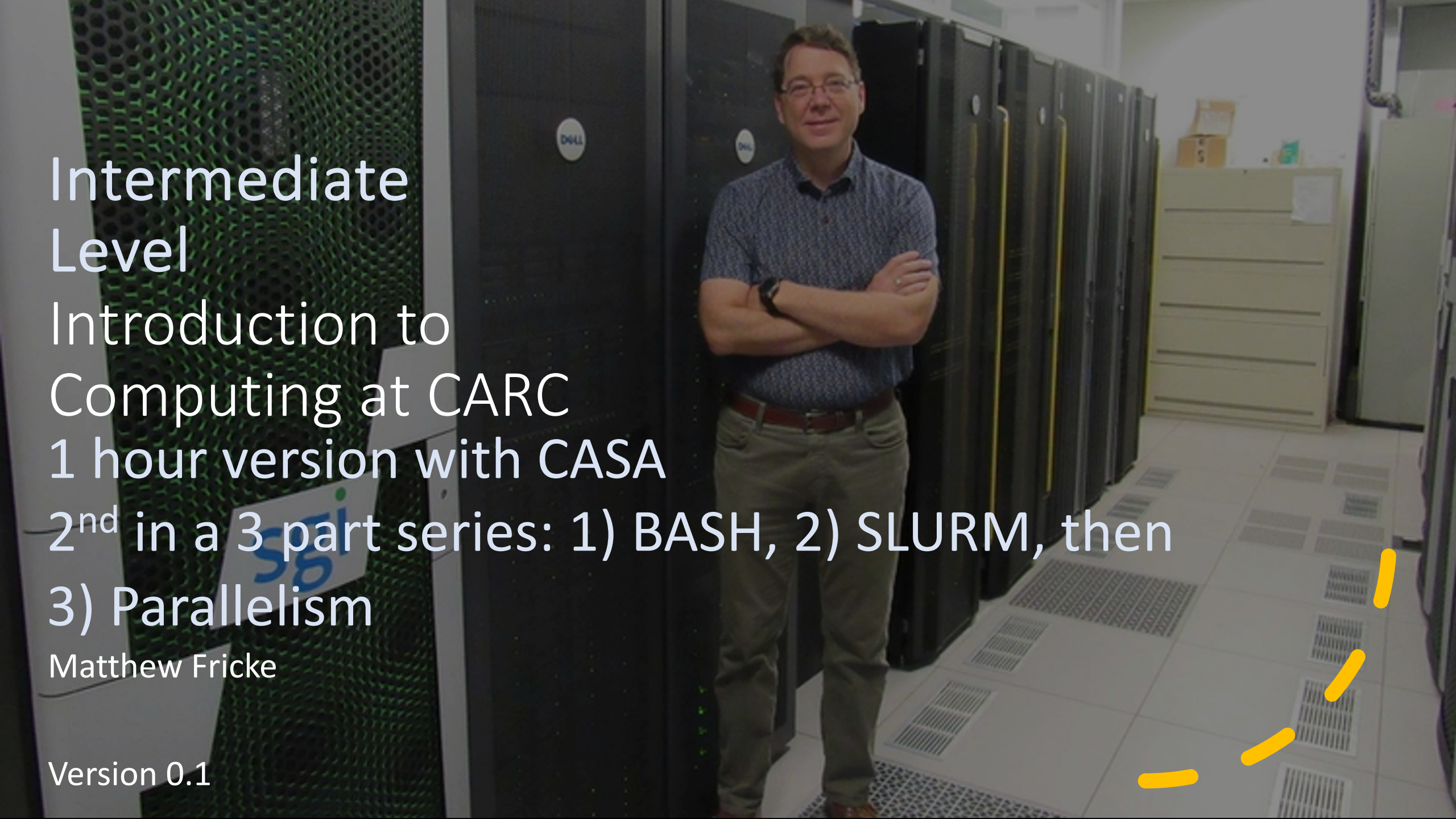


Intermediate Level Introduction to Computing at CARC

1 hour version with CASA

2nd in a 3 part series: 1) BASH, 2) SLURM, then 3) Parallelism

Matthew Fricke

Version 0.1

Goals

- 1) SLURM scheduler literacy
- 2) Running CASA interactively and in batch
- 2) JupyterHub and CASA

- We wont cover file transfer, storage systems, module system, conda, PBS. (These are all covered in depth in the video tutorials)

Logging into Hopper

First login to the Linux **workstation** in front of you. Your CARC username is on the sign in sheet.

If you have logged in before, use your existing password

Otherwise, your initial password is Welcome2CARC

This is an “important step” so don’t let me move on until you have logged in

Logging into Hopper


```
ssh vanilla@hopper.alliance.unm.edu
```

Should prompt you for a password…

Don’t let me move on until you are able to login.

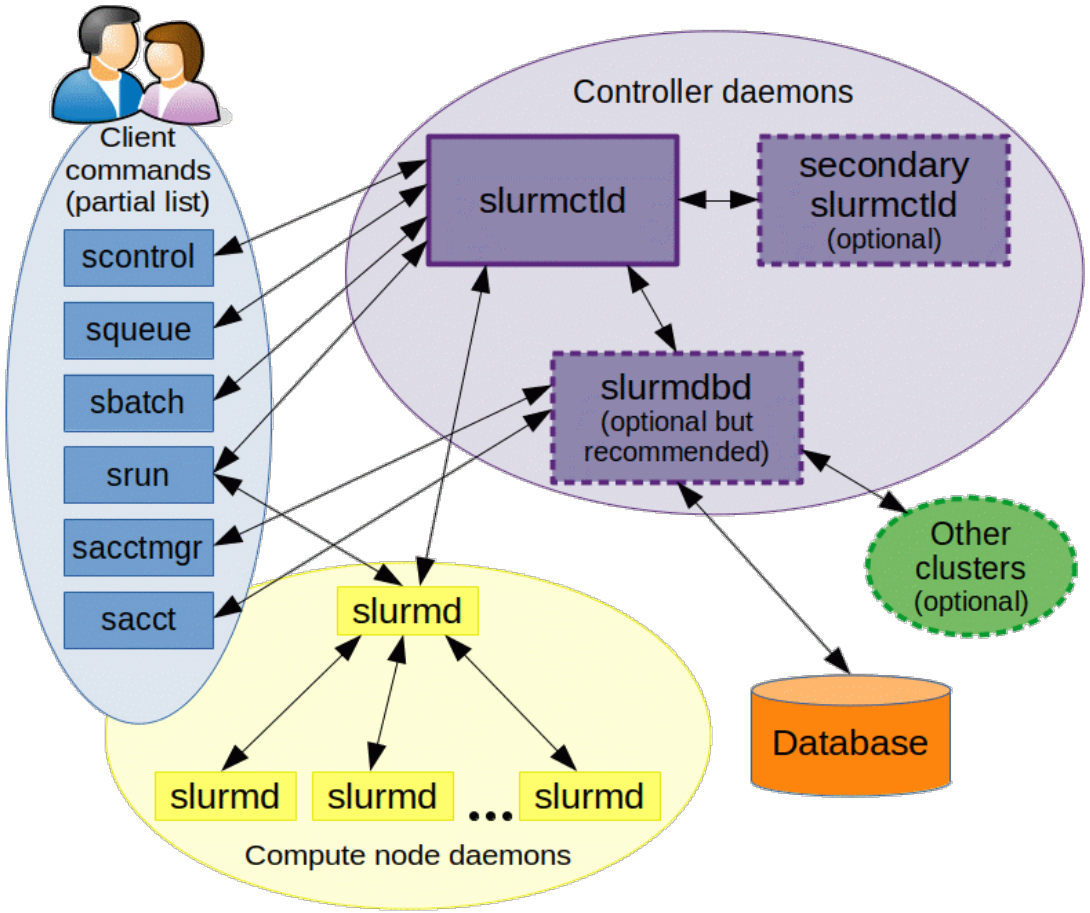
Replace vanilla with your name (unless your last name is Ice)

Logging into Hopper

```
Welcome to Hopper

  Be sure to review the "Acceptable Use" guidelines posted on the CARC website.

  For assistance using this system email help@carc.unm.edu.

  Tutorial videos can be accessed through the CARC website: Go to
    http://carc.unm.edu, select the "New Users" menu and then click
      "Introduction to Computing at CARC".

  Warning: By default home directories are world readable. Use the chmod command
    to restrict access.

  Don't forget to acknowledge CARC in publications, dissertations, theses and
    presentations that use CARC computational resources:

  "We would like to thank the UNM Center for Advanced Research Computing,
  supported in part by the National Science Foundation, for providing the
  research computing resources used in this work."

  Please send citations to publications@carc.unm.edu.

  There are three types of slurm partitions on Hopper:
  1) General - this partition is accessible by all CARC users.

  2) Condo - preemtable scavenger queue available to all condo users. Your job must use checkpointing to use this
queue or you will lose any work you have done if it is preempted by the partition's owner.

  3) Named partitions - these partitions are available to condo users working under the grant/lab/center that
purchased the associated hardware.

  Type "qgrok" to get the status of the partitions.
--------------------------------------------------------------------------------
Last login: Wed Jul 27 17:46:13 2022 from 129.24.246.68
vanilla@hopper:~ $
```

Simple Linux Utility for Resource Management

slurm
workload manager




ENJOY
Slurm
SODA
IT'S HIGHLY ADDICTIVE!
VOTED #1 SOFT DRINK OF THE 31ST CENTURY!

```
[vanilla@hopper ~]$ qgrok
```

queues	free	busy	offline	jobs	nodes	CPUs	GPUs	CPUs/node	GPUs/node	Memory/node	time_limit	CPU_limit
general	4	6	0	4	10	320	0	32	0	93G	2-00:00:00	64
debug	2	0	0	0	2	64	0	32	0	93G	4:00:00	8
condo	22	25	4	7	51	1632	28	32	2	93G-1.5T	2-00:00:00	192
bugs	2	0	0	0	2	64	0	32	0	93G	7-00:00:00	

```
vanilla@hopper:~ $ qgrok
queues      free   busy   offline  jobs   nodes  CPUs   GPUs
-----       -----  -----  -----    -----  -----  -----  -----
general     1      9      0        7      10     320    0
debug       2      0      0        0      2      64     0
condo       18     19     1        7      38     1216   8
bugs        0      2      0        0      2      64     0
pcnc        1      1
pathogen    1      0
tc          5      5
gold        2      0
fishgen     0      1
neuro-hsc   8      6
cup-ecs     0      2
tid         0      1
biocomp     0      1
chakra      1      0
pna         0      0
totals:     19     28
```

Open partitions for use by everyone with a CARC account.

Purchased by the Office for the Vice President for Research.

```
vanilla@hopper:~ $ qgrok
queues      free   busy   offline  jobs   nodes  CPUs   GPUs
-----       -----  -----  -----    -----  -----  -----  -----
general     1      9      0        7      10     320    0
debug       2      0      0        0      2      64     0
condo       18     19     1        7      38     1216   8
bugs        0      2      0        0      2      64     0
pcnc        1      1      0        0      2      64     0
pathogen    1      0      0        0      1      32     0
tc          5      5      0        3      10     320    0
gold        2      0      0        0      2      64     0
fishgen     0      1      0        0      1      32     0
neuro-hsc   8      6      0        0      14     448    0
cup-ecs     0      2      0        2      2      64     4
tid         0      1      0        0      1      32     2
biocomp     0      1      0        0      1      32     1
chakra      1      0      0        0      1      32     1
pna         0      0      1        0      1      32     0
totals:     19     28     1        14     48     1536   8
```

```
vanilla@hopper:~ $ qgrok
queues      free   busy   offline  jobs   nodes  CPUs   GPUs
-----       -----  -----  -----    -----  -----  -----  -----
general     1      9      0        7      10     320    0
debug       2      0      0        0      2      64     0
condo       18     19     1        7      38     1216   8
bugs        0      2      0        0      2      64     0
pcnc        1      1
pathogen    1      0
tc          5      5
gold        2      0
fishgen     0      1
neuro-hsc   8      6
cup-ecs     0      2
tid         0      1
biocomp     0      1
chakra      1      0
pna         0      0
totals:     19     28
```

Private partitions

- Reserved for use by the purchaser.
- Request access by emailing support@carc.unm.edu and CC the partition owner.

```
vanilla@hopper:~ $ qgrok
queues      free   busy   offline  jobs   nodes  CPUs   GPUs
-----       -----  -----  -----    -----  -----  -----  -----
general     1      9      0        7      10     320    0
debug       2      0      0        0      2      64     0
condo       18     19     1        7      38     1216   8
bugs        0      2      0        0      2      64     0
pcnc        1      1
pathogen    1      0
tc          5      5
gold        2      0
fishgen     0      1
neuro-hsc   8      6
cup-ecs     0      2
tid         0      1
biocomp     0      1
chakra      1      0
pna         0      0
totals:     19     28
```

Condo “scavenger” partition

- Allows you to use compute nodes purchased by another group that are currently idle.
- May be interrupted at any time if the owners start to use it.

```
[vanilla@hopper ~]$ quotas
Home Directory (/users/vanilla):
quota: Cannot resolve mountpoint path /root/.spack: Permission denied
Disk quotas for user vanilla (uid 659):
     Filesystem   space   quota   limit   grace   files   quota   limit   grace
chama:/home/homes
                 1527M    100G    200G           14913   4295m   4295m
----------------------------
Centerwide user scratch (/carc/scratch/users/vanilla)

Quota information for storage pool Default (ID: 1):

      user/group     ||           size          ||    chunk files
     name     |  id  ||    used    |    hard    ||  used   |  hard
--------------|------||------------|------------||---------|---------
       vanilla|  1512||  592.71 GiB | 1024.00 GiB ||    32784|unlimited

Centerwide scratch quota for project vanilla2016174 (/carc/scratch/projects/vanilla2016174)

Quota information for storage pool Default (ID: 1):
      user/group     ||           size          ||    chunk files
     name     |  id  ||    used    |    hard    ||  used   |  hard
--------------|------||------------|------------||---------|---------
vanilla2016174|2016142||  190.97 GiB| 1024.00 GiB||    23704|unlimited
```

```
[vanilla@hopper ~]$ sinfo --partition debug
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
debug        up    4:00:00      2   idle hopper[011-012]
```

sinfo reports information about partitions

```
[vanilla@hopper ~]$ sinfo --partition debug
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
debug        up    4:00:00      2   idle hopper[011-012]
```

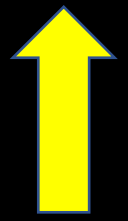
The debug queues are intended for testing your programs.

And for interactive jobs.

```
[vanilla@hopper ~]$ sinfo --partition debug
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
debug        up    4:00:00      2   idle hopper[011-012]
```

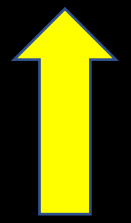
Name

```
[vanilla@hopper ~]$ sinfo --partition debug
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
debug        up    4:00:00      2   idle hopper[011-012]
```



You can run a “job” for up to 4 hrs.

```
[vanilla@hopper ~]$ sinfo --partition debug
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
debug        up    4:00:00      2   idle hopper[011-012]
```



There are two nodes in this partition.

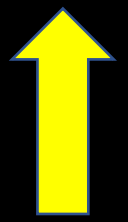
```
[vanilla@hopper ~]$ sinfo --partition debug
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
debug        up    4:00:00      2   idle hopper[011-012]
```

The names of the nodes in the partition

```
[vanilla@hopper ~]$ sinfo --partition debug
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
debug        up    4:00:00      2   idle hopper[011-012]
```

The names of the nodes in the partition

```
[vanilla@hopper ~]$ sinfo --partition general
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
general*     up 2-00:00:00      9  alloc hopper[001-009]
general*     up 2-00:00:00      1   idle hopper010
```



Hopper001 through 009 are running jobs.

Hopper010 is waiting to be used.

```
[vanilla@hopper ~]$ hostname
hopper
[vanilla@hopper ~]$
```

Running on the Head Node.

The head node’s name is “hopper”.

```
[vanilla@hopper ~]$ hostname
hopper
[vanilla@hopper ~]$ man hostname
```

```
[vanilla@hopper ~]$ hostname
hopper
[vanilla@hopper ~]$ man hostname
(‘q’ to quit)

[vanilla@hopper ~]$ man man
(‘q’ to quit)
```

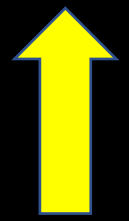
[vanilla@hopper ~]$ man sinfo

```
sinfo(1)                                    Slurm
Commands                                    sinfo(1)

NAME
       sinfo - View information about Slurm nodes and partitions.

SYNOPSIS
       sinfo [OPTIONS...]

DESCRIPTION
       sinfo is used to view partition and node information for a system running
Slurm

OPTIONS
       -a, --all
              Display  information  about  all partitions. This causes information
to be displayed about partitions that are configured as hidden and partitions
that are unavailable to the user's group.
```

[vanilla@hopper ~]$ sinfo --all

```
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
general*     up 2-00:00:00      9  alloc hopper[001-009]
general*     up 2-00:00:00      1   idle hopper010
debug        up    4:00:00      2   idle hopper[011-012]
condo        up 2-00:00:00      1  down* hopper045
condo        up 2-00:00:00      3    mix hopper[018-020]
condo        up 2-00:00:00     16  alloc hopper[013-015,028-036,049-052]
condo        up 2-00:00:00     18   idle hopper[016-017,021-027,037-044,053]
bugs         up 7-00:00:00      2  alloc hopper[013-014]
pcnc         up 7-00:00:00      1  alloc hopper015
pcnc         up 7-00:00:00      1   idle hopper016
pathogen     up 7-00:00:00      1   idle hopper017
tc           up 7-00:00:00      3    mix hopper[018-020]
tc           up 7-00:00:00      2  alloc hopper[029-030]
tc           up 7-00:00:00      5   idle hopper[021-025]
gold         up 7-00:00:00      2   idle hopper[026-027]
fishgen      up 7-00:00:00      1  alloc hopper028
neuro-hsc    up 7-00:00:00      6  alloc hopper[031-036]
neuro-hsc    up 7-00:00:00      8   idle hopper[037-044]
cup-ecs      up 7-00:00:00      2  alloc hopper[049-050]
tid          up 7-00:00:00      1  alloc hopper051
biocomp      up 7-00:00:00      1  alloc hopper052
chakra       up 7-00:00:00      1   idle hopper053
pna          up 7-00:00:00      1  down* hopper045
```

```
[vanilla@hopper ~]$ srun --partition debug hostname
```

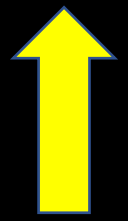


Tell slurm to run a program
on a compute node…

```
[vanilla@hopper ~]$ srun --partition debug hostname
```

Run the program on a compute node in the debug partition.

```
[vanilla@hopper ~]$ srun --partition debug hostname
```



The program to run.

```
[vanilla@hopper ~]$ srun --partition debug hostname
srun: Account not specified in script or
~/.default_slurm_account, using latest project
You have not been allocated GPUs. To request GPUs,
use the -G option in your submission script.
hopper011
```

```
[vanilla@hopper ~]$ squeue
```

```
[vanilla@hopper ~]$ squeue
 JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
 4314   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4315   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4317   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4318   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4319   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4320   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4321   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4322   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4323   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4324   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4325   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4326   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4328   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4329   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4330   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4331   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4332   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4333   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4334   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4335   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4336   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
 4337   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
```

```
[vanilla@hopper ~]$ squeue
JOBID PARTITION     NAME     USER ST       TIM
4314    general      PRE erowland PD      0:00
4315    general      PRE erowland PD      0:00
4317    general      PRE erowland PD      0:00
4318    general      PRE erowland PD      0:00
                                                   2 (QOSMaxCpuPerUserLimit)
                                                   2 (QOSMaxCpuPerUserLimit)
                                                   2 (QOSMaxCpuPerUserLimit)
                                                   2 (QOSMaxCpuPerUserLimit)
                                                   2 (QOSMaxCpuPerUserLimit)
                                                   2 (QOSMaxCpuPerUserLimit)
                                                   2 (QOSMaxCpuPerUserLimit)
                                                   2 (QOSMaxCpuPerUserLimit)
                                                   2 (QOSMaxCpuPerUserLimit)
                                                   2 (QOSMaxCpuPerUserLimit)
                                                   2 (QOSMaxCpuPerUserLimit)
                                                   2 (QOSMaxCpuPerUserLimit)
                                                   2 (QOSMaxCpuPerUserLimit)
                                                   2 (QOSMaxCpuPerUserLimit)
                                                   2 (QOSMaxCpuPerUserLimit)
                                                   2 (QOSMaxCpuPerUserLimit)
                                                   2 (QOSMaxCpuPerUserLimit)
                                                   2 (QOSMaxCpuPerUserLimit)
```

PD means programs that are waiting their turn.

Shows you what the slurm scheduler is doing right now.

Here we can see that user ’erowland’ has a lot of programs waiting to run.

```
[vanilla@hopper ~]$ squeue
 JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
             4314   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
             4315   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
             4317   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
             4318   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
                                                                 2 (QOSMaxCpuPerUserLimit)
             4337   general      PRE erowland PD       0:00      2 (QOSMaxCpuPerUserLimit)
```

The reason these jobs are not running is that ‘erowland’ is already using the maximum number of CPUs they are allowed.

```
[vanilla@hopper ~]$ squeue -t R --all
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
              4405     condo    2ndMA  vanilla  R 1-07:48:30      6 hopper[031-036]
              5208     condo       NN      kgu  R    5:48:49      1 hopper015
              5210     condo       NN      kgu  R    6:30:13      1 hopper014
              5209     condo       NN      kgu  R    6:31:13      1 hopper013
              5206     condo       NN      kgu  R    6:32:13      1 hopper051
              5207     condo       NN      kgu  R    6:32:13      1 hopper052
              5205     condo       NN      kgu  R    6:32:43      1 hopper028
              4595   cup-ecs golConfi aalasand  R 2-06:51:59      1 hopper050
              4594   cup-ecs golConfi aalasand  R 2-06:52:03      1 hopper049
              5120   general jupyterh   jacobm  R   11:45:47      1 hopper007
              4313   general      PRE erowland  R    1:17:29      2 hopper[003-004]
              5111   general    1stMA  vanilla  R   11:15:28      2 hopper[005-006]
              5025   general      c2n    jxzuo  R       1:50      1 hopper001
              5024   general      c2n    jxzuo  R      31:28      1 hopper002
              5203   general       NN      kgu  R    6:37:50      1 hopper009
              5201   general       NN      kgu  R    6:38:14      1 hopper008
              4390        tc UCsTpCyd lepluart  R 2-15:18:18      3 hopper[018-020]
              5198        tc       NN      kgu  R    6:40:19      1 hopper030
              5196        tc       NN      kgu  R    6:40:31      1 hopper029
```

```
[vanilla@hopper ~]$ srun --partition debug --ntasks 2 hostname
srun: Account not specified in script or
~/.default_slurm_account, using latest project
You have not been allocated GPUs. To request GPUs, use the -G
option in your submission script.
hopper011
hopper011
```

```
[vanilla@hopper ~]$ srun --partition debug --ntasks 2 hostname
srun: Account not specified in script or
~/.default_slurm_account, using latest project
You have not been allocated GPUs. To request GPUs, use the -G
option in your submission script.
hopper011
hopper011
```

You ran two copies of your program.

ntasks is the number of copies to run.

```
[vanilla@hopper ~]$ srun --partition debug --ntasks 2 --cpus-per-task 2 hostname
srun: Account not specified in script or ~/.default_slurm_account, using latest project
You have not been allocated GPUs. To request GPUs, use the -G option in your submission
script.
hopper011
hopper011
```

Here we are telling SLURM to run 2 copies of our program and let each copy of our program use 2 CPUs.

```
[vanilla@hopper ~]$ srun --partition debug --nodes 2 --ntasks-per-node 4 hostname
srun: Account not specified in script or ~/.default_slurm_account, using
latest project
hopper012
You have not been allocated GPUs. To request GPUs, use the -G option in
your submission script.
hopper012
hopper011
hopper011
hopper012
hopper012
hopper011
hopper011
```

Here we are telling SLURM to run 4 copies of our program on 2 different compute nodes.

This is useful when our programs need a bigger share of the compute node.

```
[vanilla@hopper ~]$ srun --partition debug --nodes 2
--ntasks-per-node 2 --cpus-per-task 2 hostname
srun: Account not specified in script or
~/.default_slurm_account, using latest project
hopper011
You have not been allocated GPUs. To request GPUs, use
the -G option in your submission script.
hopper011
hopper012
hopper012
```

And we can combine all three.

```
[vanilla@hopper ~]$ srun --partition debug --mem 4G
--nodes 2 --ntasks-per-node 2 --cpus-per-task 2
hostname
srun: Account not specified in script or
~/.default_slurm
hopper012
hopper012
You have not be                                    se
the -G option i
hopper011
Hopper011
```

And we can specify how much memory we want.

--mem 4G means give me 4 gigabytes of memory per node.

```
[vanilla@hopper ~]$ srun --partition debug --mem 4G
--nodes 2 --ntasks-per-node 2 --cpus-per-task 2
hostname
srun: Account not specified in script or
~/.default_slurm
hopper012
hopper012
You have not be                                    se
the -G option i
hopper011
Hopper011
```

Why does all this matter?

The purpose of SLURM is to provide you the hardware your programs need.

So you have to understand what those requirements are really well.

```
[vanilla@hopper ~]$ srun --partition debug --mem 4G
--nodes 2 --ntasks-per-node 2 --cpus-per-task 2 hostname
srun: Account not specified in script or
~/.default_slurm_account, using latest project
hopper012
hopper012
You have not be                                    se
the -G option i
hopper011
Hopper011
```

1) **Can my program use multiple CPUs?**
2) **How much memory does my program need?**
3) **Can my program use multiple compute nodes (MPI*, GNU Parallel*)?**
4) **Can my program use GPUs?**

```
[vanilla@hopper ~]$ srun --partition debug --mem 4G
--nodes 2 --ntasks-per-node 2 --cpus-per-task 2 hostname
srun: Account not specified in script or
~/.default_slurm_account, using latest project
hopper012
hopper012
You have not been                                    se
the -G option in
hopper011
Hopper011
```

This command is getting pretty long.

We can use shell scripts to automate all this in batch mode.

Interactive vs Batch Mode

Interactive Mode

- Everything so far has been interactive. You request hardware, run your program, and get the output on your screen right away.

Batch Mode

- Most programs at an HPC center are run in “batch” mode.
- Batch mode means we write a shell script that the SLURM scheduler runs for us. The script requests hardware just like we did with salloc and then runs the commands in the script.
- Whatever would have been written to the screen is saved to a file instead.

```
[vanilla@hopper ~]$ git clone https://lobogit.unm.edu/CARC/workshops.git
Cloning into 'workshops'...
remote: Enumerating objects: 132, done.
remote: Counting objects: 100% (75/75), done.
remote: Compressing objects: 100% (43/43), done.
remote: Total 132 (delta 33), reused 74 (delta 32), pack-reused 57
Receiving objects: 100% (132/132), 57.58 KiB | 3.60 MiB/s, done.
Resolving deltas: 100% (51/51), done.
```

Rather than make you write shell scripts lets just download some we wrote for this workshop…

```
[vanilla@hopper ~]$ tree workshops
workshops/
├── intro_workshop
│   ├── code
│   │   ├── calcPiMPI.py
│   │   ├── calcPiSerial.py
│   │   └── vecadd
│   │       ├── Makefile
│   │       ├── vecadd_gpu.cu
│   │       ├── vecadd_mpi_cpu
│   │       ├── vecadd_mpi_cpu.c
│   │       ├── vecaddmpi_cpu.sh
│   │       └── vecadd_mpi_gpu.c
│   ├── data
│   │   ├── H2O.gjf
│   │   └── step_sizes.txt
│   └── slurm
│       ├── calc_pi_array.sh
│       ├── calc_pi_mpi.sh
│       ├── calc_pi_parallel.sh
│       ├── calc_pi_serial.sh
│       ├── gaussian.sh
│       ├── hostname_mpi.sh
│       ├── vecadd_hopper.sh
│       ├── vecadd_xena.sh
│       ├── workshop_example2.sh
│       ├── workshop_example3.sh
│       └── workshop_example.sh
└── README.md
```

Run tree to see how the workshops directories are organized…

```
[vanilla@hopper ~]$ tree workshops
workshops/
├── intro_workshop
│   ├── code
│   │   ├── calcPiMPI.py
│   │   ├── calcPiSerial.py
│   │   └── vecadd
│   │       ├── Makefile
│   │       ├── vecadd_gpu.cu
│   │       ├── vecadd_mpi_cpu
│   │       ├── vecadd_mpi_cpu.c
│   │       ├── vecaddmpi_cpu.sh
│   │       └── vecadd_mpi_gpu.c
│   ├── data
│   │   ├── H2O.gjf
│   │   └── step_sizes.txt
│   └── slurm
│       ├── calc_pi_array.sh
│       ├── calc_pi_mpi.sh
│       ├── calc_pi_parallel.sh
│       ├── calc_pi_serial.sh
│       ├── gaussian.sh
│       ├── hostname_mpi.sh
│       ├── vecadd_hopper.sh
│       ├── vecadd_xena.sh
│       ├── workshop_example2.sh
│       ├── workshop_example3.sh
│       └── workshop_example.sh
└── README.md
```

Run tree to see how the workshops directories are organized…

The workshop files are divided into “code”, “slurm”, and “data” directories.

You will also see a radio_astronomy directory

```
[vanilla@hopper intro_workshop]$ pwd
/users/vanilla/workshops/intro_workshop
[vanilla@hopper intro_workshop]$ cat slurm/workshop_example.sh
#!/bin/bash
#SBATCH --partition debug
#SBATCH --ntasks 4
#SBATCH --time 00:05:00
#SBATCH --job-name ws_example
#SBATCH --mail-user your_username@unm.edu
#SBATCH --mail-type ALL

hostname
```

Let's take a look at the workshop_example.sh script in the slurm directory…

```
[vanilla@hopper intro_workshop]$ sbatch slurm/workshop_example.sh
sbatch: Account not specified in script or ~/.default_slurm_account,
using latest project
Submitted batch job 5252
[vanilla@hopper intro_workshop]$
```

We submit our slurm shell script with the sbatch command.

```
[vanilla@hopper intro_workshop]$ sbatch slurm/workshop_example.sh
sbatch: Account not specified in script or ~/.default_slurm_account,
using latest project
Submitted batch job 5252
[vanilla@hopper intro_workshop]$
```

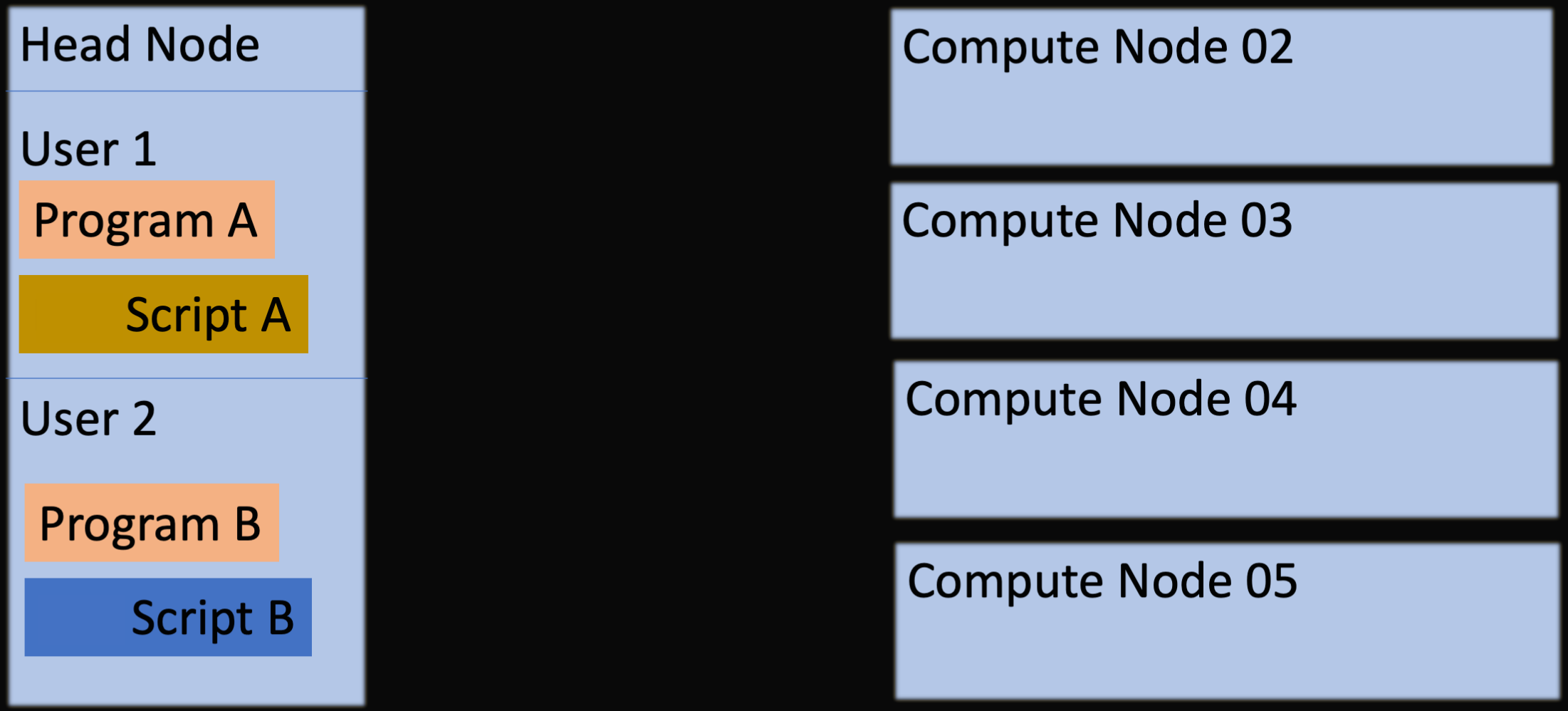
We submit our slurm shell script with the sbatch command.

Notice that the only output we get is a job id.

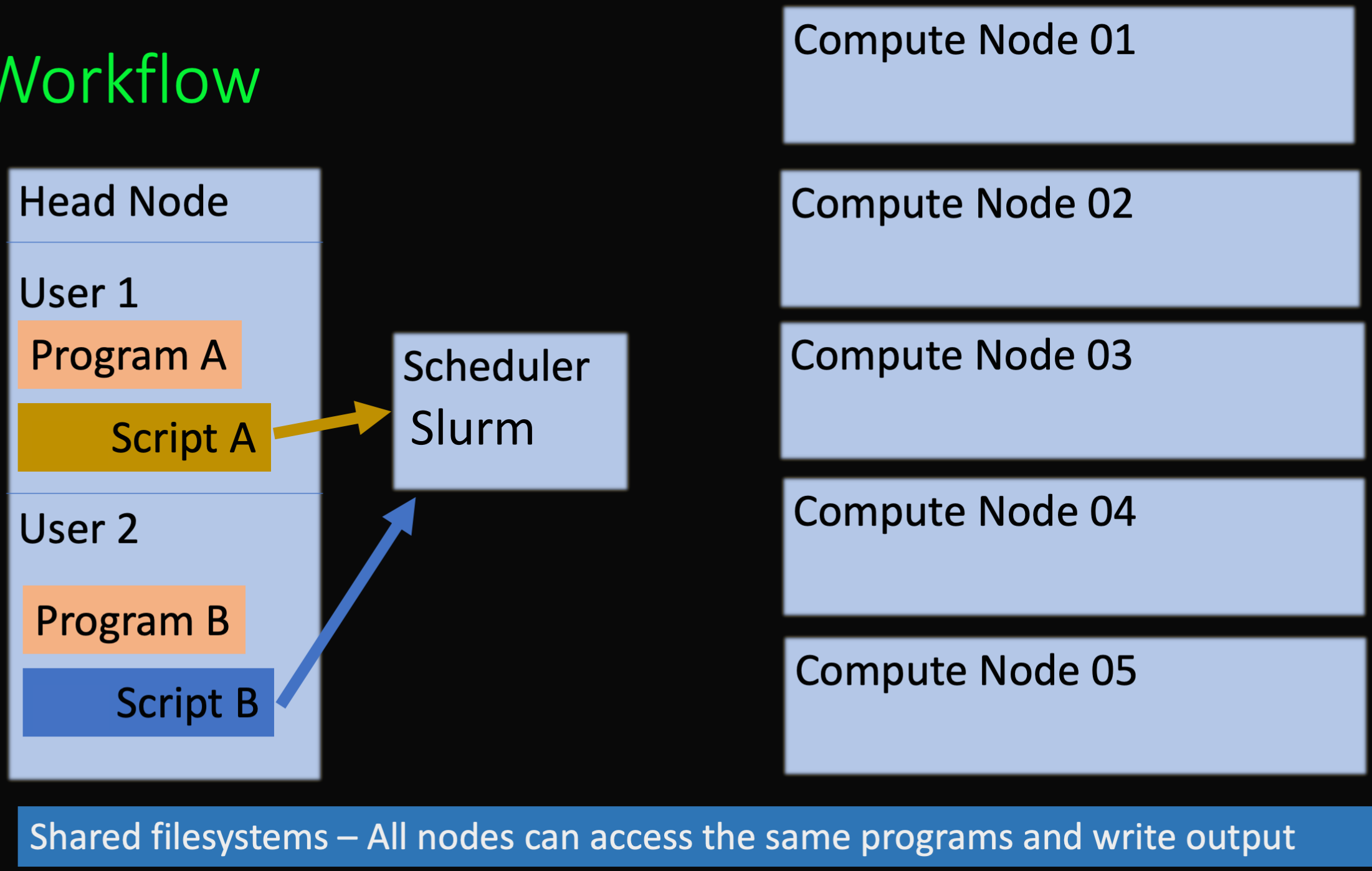
This indicates that the script was successfully sent to the scheduler.

The commands in the script will run as soon as the hardware requested is available.

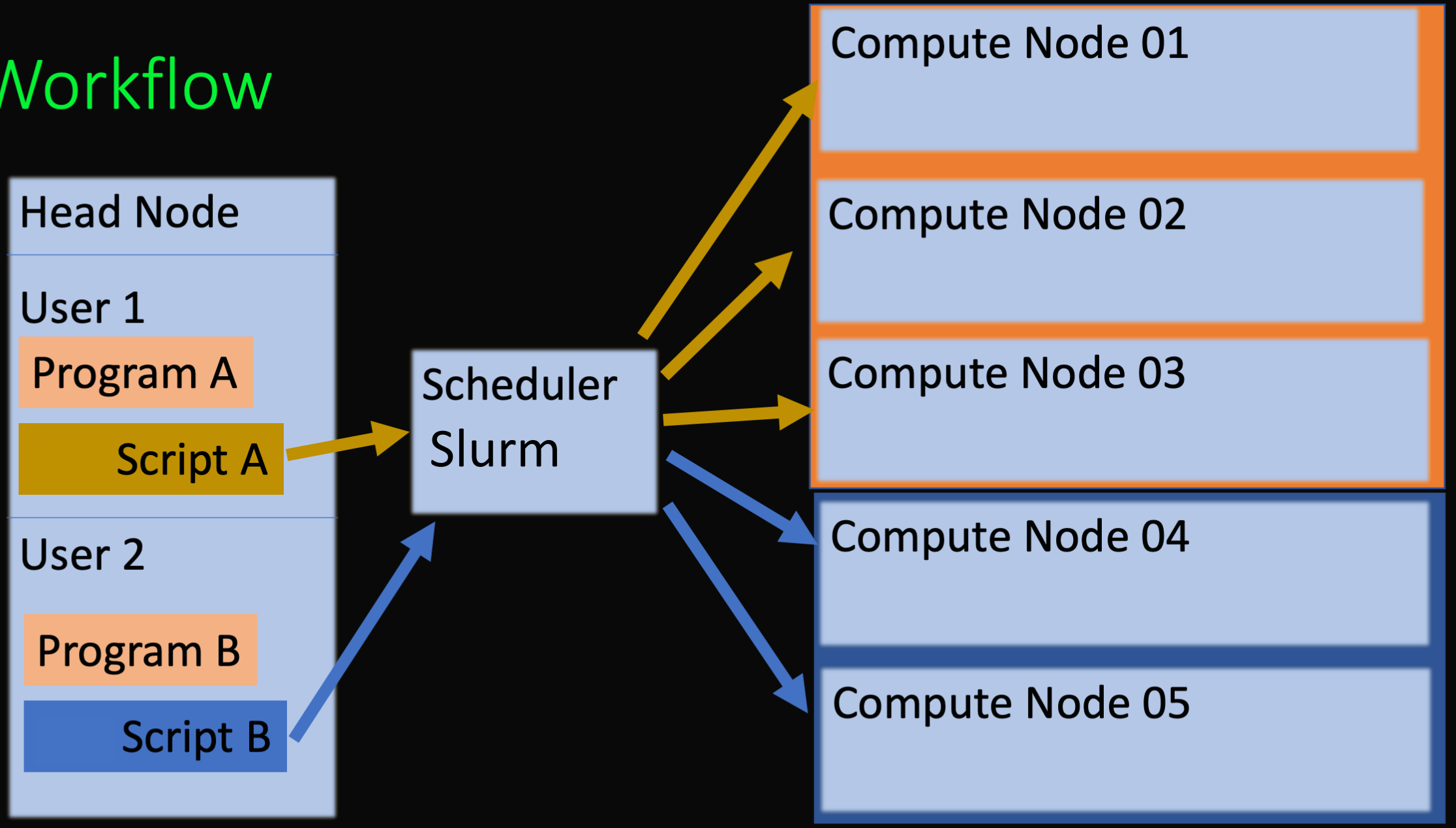
Workflow


Head Node
User 1
Program A
Script A
User 2
Program B
Script B
Compute Node 01
Compute Node 02
Compute Node 03
Compute Node 04
Compute Node 05
Shared filesystems – All nodes can access the same programs and write output

Workflow

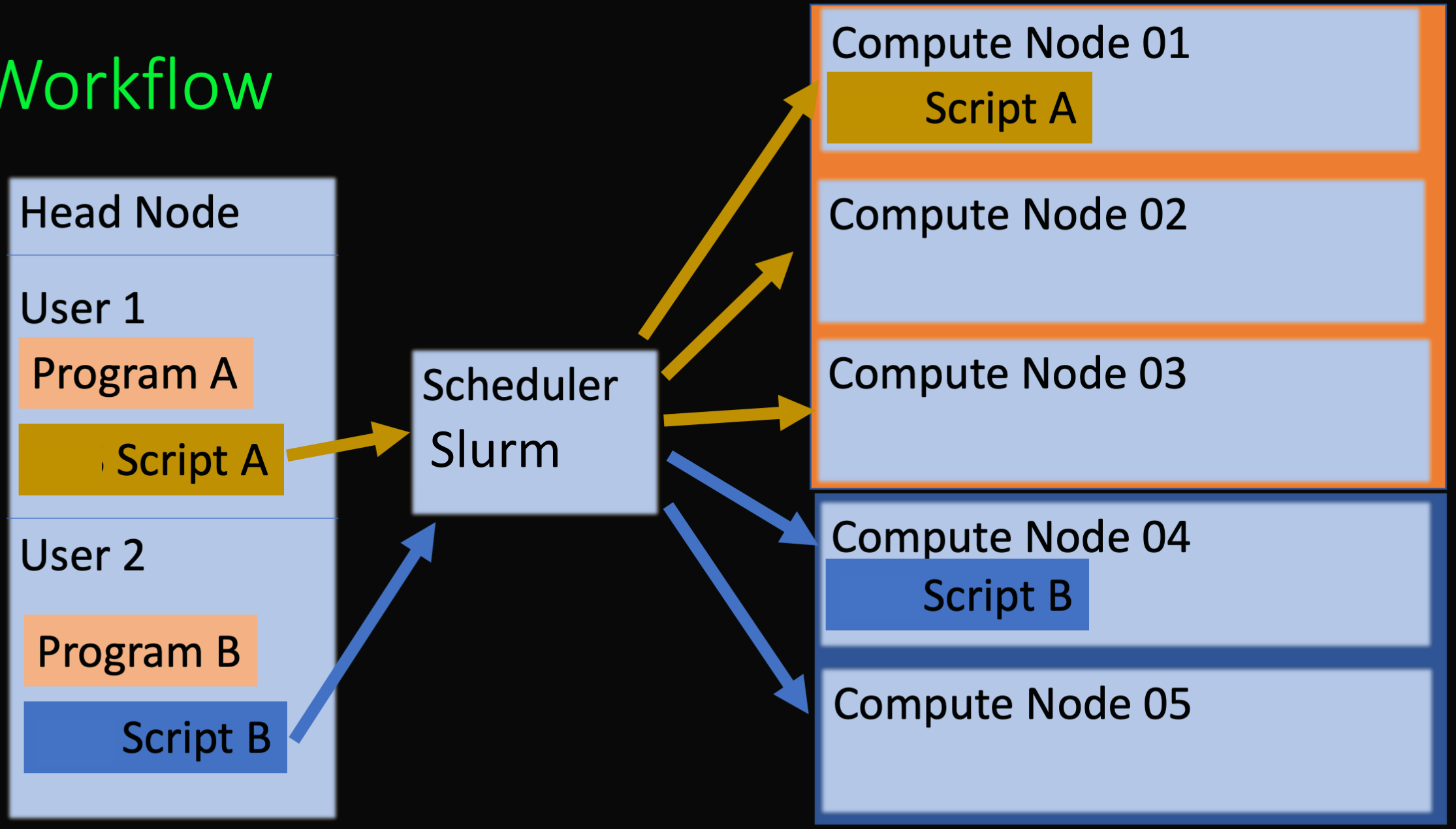

Head Node
User 1
Program A
Script A
User 2
Program B
Script B
Scheduler
Slurm
Compute Node 01
Compute Node 02
Compute Node 03
Compute Node 04
Compute Node 05
Shared filesystems – All nodes can access the same programs and write output

Workflow



Shared filesystems – All nodes can access the same programs and write output

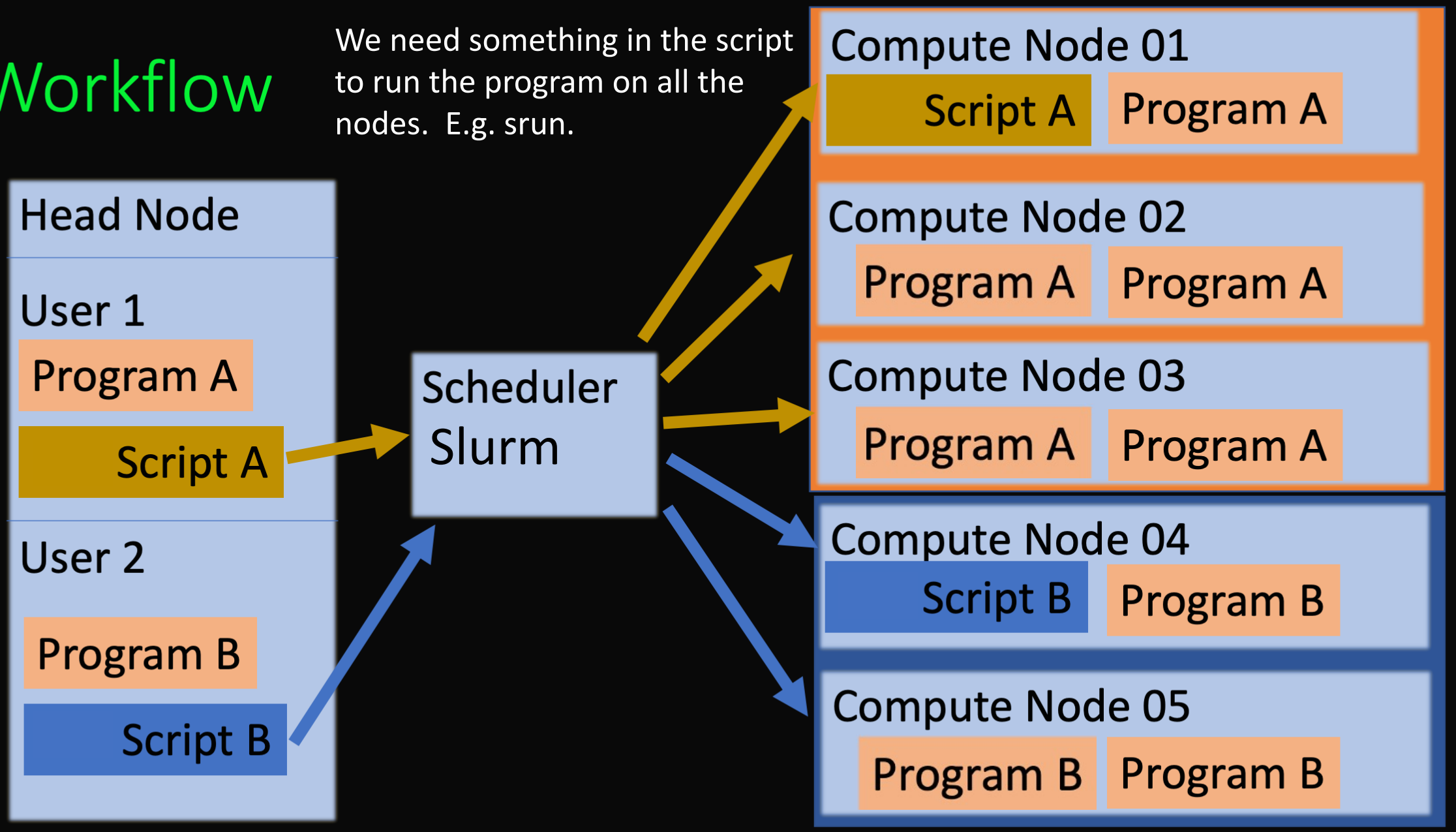
Workflow



Shared filesystems – All nodes can access the same programs and write output

Workflow

We need something in the script to run the program on all the nodes. E.g. srun.



Shared filesystems – All nodes can access the same programs and write output

```
[vanilla@hopper intro_workshop]$ ls
code  data  pbs  slurm  slurm-5252.out
```

The **hostname** command is very fast so everyone's job should finish in a few seconds.

When it is finished you will have a new file named slurm-{your job id}.out.

```
[vanilla@hopper intro_workshop]$ ls
code  data  pbs  slurm  slurm-5252.out
```

When it is finished you will have a new file named slurm-{your job id}.out.

```
[vanilla@hopper intro_workshop]$ cat slurm-5252.out
hopper011
```

```
[vanilla@hopper intro_workshop]$ ls
code  data  pbs  slurm  slurm-5252.out
```

When it is finished you will have a new file named slurm-{your job id}.out.

```
[vanilla@hopper intro_workshop]$ cat slurm-5252.out
hopper011
```

Why did it only run the program once instead of 4 times?

```
[vanilla@hopper intro_workshop]$ sbatch slurm/workshop_example1.sh
sbatch: Account not specified in script or ~/.default_slurm_account,
using latest project
Submitted batch job 5252
[vanilla@hopper intro_workshop]$
```

Take a look at the output file.

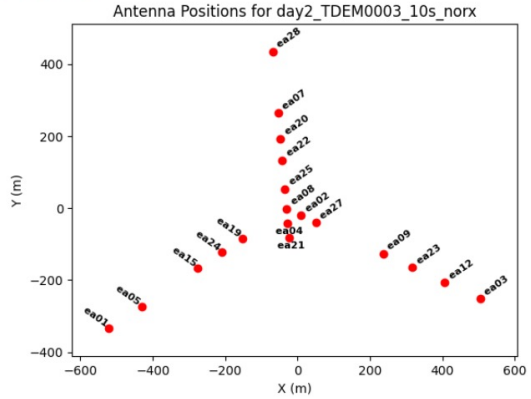
You should see 4 hostname output lines.

CASA

- Python based Common Astronomy Software Applications

```
matthew — IPython: users/mfricke — ssh hopper — 80×24
~ — -zsh                                  ~ — IPython: users/mfricke — ssh hopper
Last login: Sun Feb 19 15:12:04 2023 from 50.193.238.94
mfricke@hopper:~ $ srun --partition debug --pty bash
srun: Using account 2016199 from ~/.default_slurm_account
You have not been allocated GPUs. To request GPUs, use the -G option in your sub
mission script.
Job 1817180 running on hopper011
mfricke@hopper011:~ $ module load casa
mfricke@hopper011:~ $ casa --nogui

optional configuration file config.py not found, continuing CASA startup without
 it

IPython 7.15.0 -- An enhanced Interactive Python.

Using matplotlib backend: agg
Creating a new telemetry file
Telemetry initialized. Telemetry will send anonymized usage statistics to NRAO.
You can disable telemetry by adding the following line to the config.py file in
your rcdir (e.g. ~/.casa/config.py):
telemetry_enabled = False
--> CrashReporter initialized.
CASA 6.5.2.26 -- Common Astronomy Software Applications [6.5.2.26]

CASA <1>:
```

```
/users/mfricke/.local/lib/python3.8/site-packages/casatasks/private/task_plotants.py:356: UserWarning: Matplotlib is currently using module://matplotlib_inline.backend_inline, which is a non-GUI backend, so cannot show the figure.
  fig.show()
```



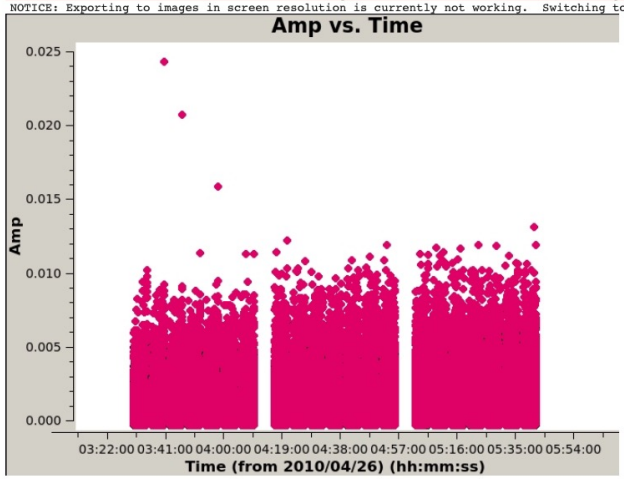
In [8]:
```
flagdata(vis=observation_path,
         mode='list',
         inpfile=["field='2,3' antenna='ea12' timerange='03:41:00~04:10:00'",
                  "field='2,3' antenna='ea07,ea08' timerange='03:21:40~04:10:00' spw='1'"])
```

Out[8]: {}

In [14]:
```
plotms(vis=observation_path,field='3',
       plotfile="temp1.jpg",
       xaxis='time',yaxis='amp',correlation='RR,LL',
       avgchannel='64',spw='0~1:4~60', coloraxis='spw')
c
```

```
2023-02-19 01:09:00     WARN    MSTransformManager::initDataSelectionParams     Number of selected channels 57 for SPW 0 is smaller than specified chanbin 64
2023-02-19 01:09:00     WARN    MSTransformManager::initDataSelectionParams+    Setting chanbin to 57 for SPW 0
2023-02-19 01:09:00     WARN    MSTransformManager::initDataSelectionParams     Number of selected channels 57 for SPW 1 is smaller than specified chanbin 64
2023-02-19 01:09:00     WARN    MSTransformManager::initDataSelectionParams+    Setting chanbin to 57 for SPW 1
NOTICE: Exporting to images in screen resolution is currently not working.  Switching to high resolution (which is slower, but works).
```

Out[14]:



In [15]:
```
plotms(vis=observation_path,field='3',plotfile="temp2.jpg",
       xaxis='uvdist',yaxis='amp',correlation='RR,LL',
       avgchannel='64',spw='0~1:4~60', coloraxis='spw')
Image(filename="temp2.jpg")
```

```
[vanilla@hopper ~]$ cd ~/workshops/radio_astronomy/
[vanilla@hopper radio_astronomy]$

$rsync -a --info=progress2 /projects/shared/workshops/radio_astronomy/casa/* .
  3,137,159,512 100%  169.43MB/s    0:00:17 (xfr#233, to-chk=0/271)
```

Copy example observation to your home storage …

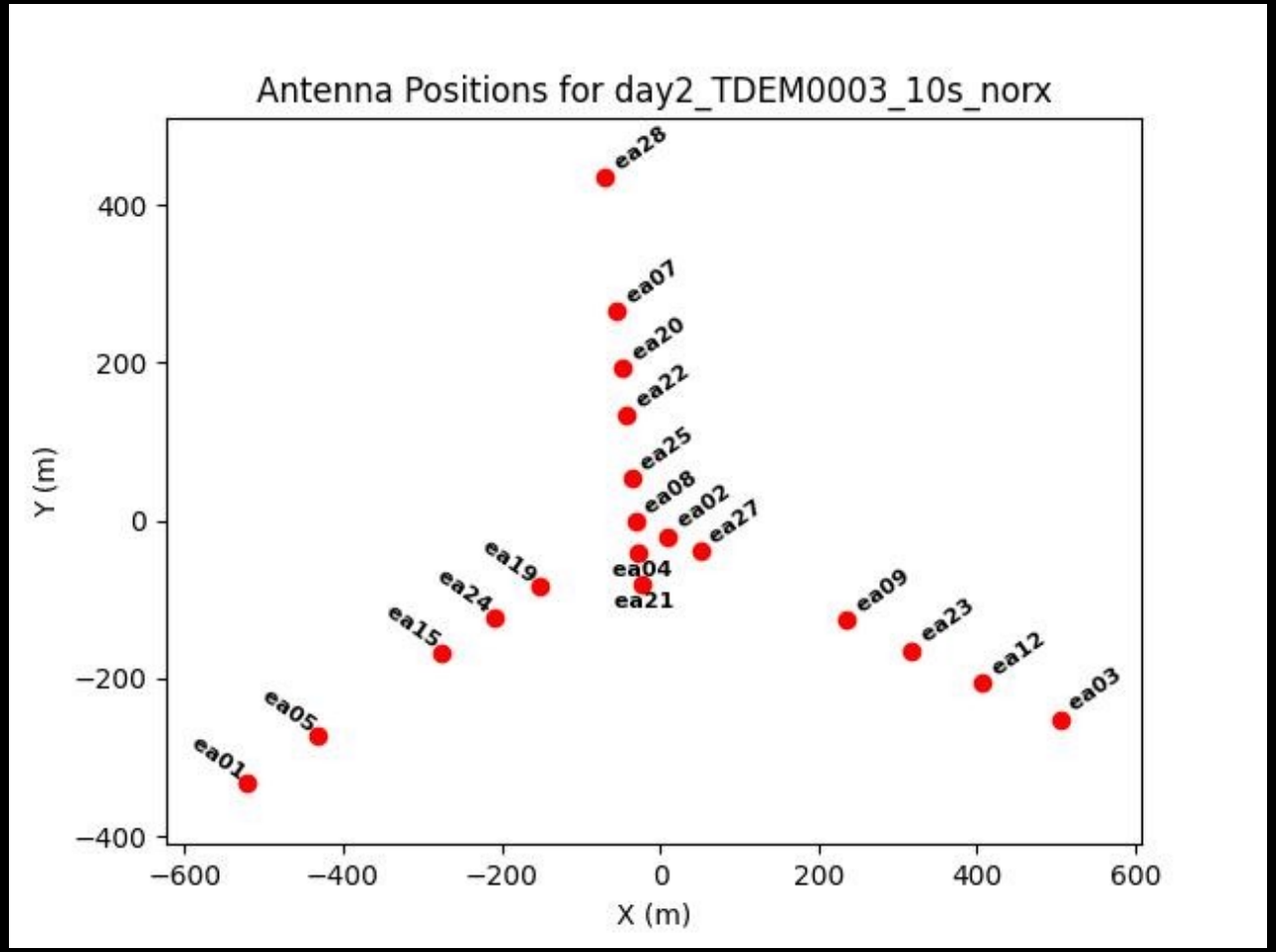
```
[vanilla@hopper radio_astronomy]$ srun --partition debug --pty bash
srun: Account not specified in script or ~/.default_slurm_account, using latest project
You have not been allocated GPUs. To request GPUs, use the -G option in your submission
script.
Job 1817350 running on hopper011
[vanilla@hopper011 radio_astronomy]$ module load casa
[vanilla@hopper011 radio_astronomy]$ casa --nogui


optional configuration file config.py not found, continuing CASA startup without it


IPython 7.15.0 -- An enhanced Interactive Python.


Using matplotlib backend: <object object at 0x1502a77a2590>
Telemetry initialized. Telemetry will send anonymized usage statistics to NRAO.
You can disable telemetry by adding the following line to the config.py file in your
rcdir (e.g. ~/.casa/config.py):
telemetry_enabled = False
--> CrashReporter initialized.
CASA 6.5.2.26 -- Common Astronomy Software Applications [6.5.2.26]


CASA <1>:
```

```
CASA 6.5.2.26 -- Common Astronomy Software Applications [6.5.2.26]


CASA <1>: default listobs

CASA <2>: inp
# listobs -- List the summary of a data set in the logger or in a file
vis             = ''                    # Name of input visibility file (MS)
selectdata      = True                  # Data selection parameters
   spw          = ''                    # Selection based on spectral-window/frequency/channel.
   field        = ''                    # Selection based on field names or field index numbers. Default is all.
   antenna      = ''                    # Selection based on antenna/baselines. Default is all.
   uvrange      = ''                    # Selection based on uv range. Default: entire range. Default units: meters.
   timerange    = ''                    # Selection based on time range. Default is entire range.
   correlation  = ''                    # Selection based on correlation. Default is all.
   scan         = ''                    # Selection based on scan numbers. Default is all.
   intent       = ''                    # Selection based on observation intent. Default is all.
   feed         = ''                    # Selection based on multi-feed numbers: Not yet implemented
   array        = ''                    # Selection based on (sub)array numbers. Default is all.
   observation  = ''                    # Selection based on observation ID. Default is all.
verbose         = True                  # Controls level of information detail reported. True reports more than False.
listfile        = ''                    # Name of disk file to write output. Default is none (output is written to
                                        # logger only).
listunfl        = False                 # List unflagged row counts? If true, it can have significant negative
                                        # performance impact.
cachesize       = 50.0                  # EXPERIMENTAL. Maximum size in megabytes of cache in which data structures can
                                        # be held.
```

```
CASA <5>: vis='day2_TDEM0003_10s_norx'

CASA <6>: go
Out[6]:
{'BeginTime': 55312.140173611115,
 'EndTime': 55312.256215277775,
 'IntegrationTime': 10026.0,
 'field_2': {'code': 'D',
  'direction': {'m0': {'unit': 'rad', 'value': 2.5959462699},
   'm1': {'unit': 'rad', 'value': 0.3093655364000001},
   'refer': 'J2000',
   'type': 'direction'},
  'name': 'J0954+1743'},
 'field_3': {'code': 'NONE',
  'direction': {'m0': {'unit': 'rad', 'value': 2.5654436141},
   'm1': {'unit': 'rad', 'value': 0.2317441393},
   'refer': 'J2000',
   'type': 'direction'},
  'name': 'IRC+10216'},
```

```
CASA <9>: default plotants

CASA <10>: vis='day2_TDEM0003_10s_norx'
CASA <11>: figfile='ant_locations.png'
CASA <12>: inp
# plotants -- Plot the antenna distribution in the local reference frame:
vis             = 'day2_TDEM0003_10s_norx'
                                       # Name of input visibility file (MS)
figfile         = 'ant_locations.png'  # Save the plotted figure to this file
antindex        = False                # Label antennas with name and antenna ID
logpos          = False                # Whether to plot logarithmic positions
exclude         = ''                   # Antenna name/id selection to exclude from
plot
checkbaselines = False                 # Whether to check baselines in the main
table.
title           = ''                   # Title for the plot
showgui         = True                 # Show plot on gui.

CASA <13>: go

CASA <14>: quit()
[vanilla@hopper011 radio_astronomy]$ exit
```

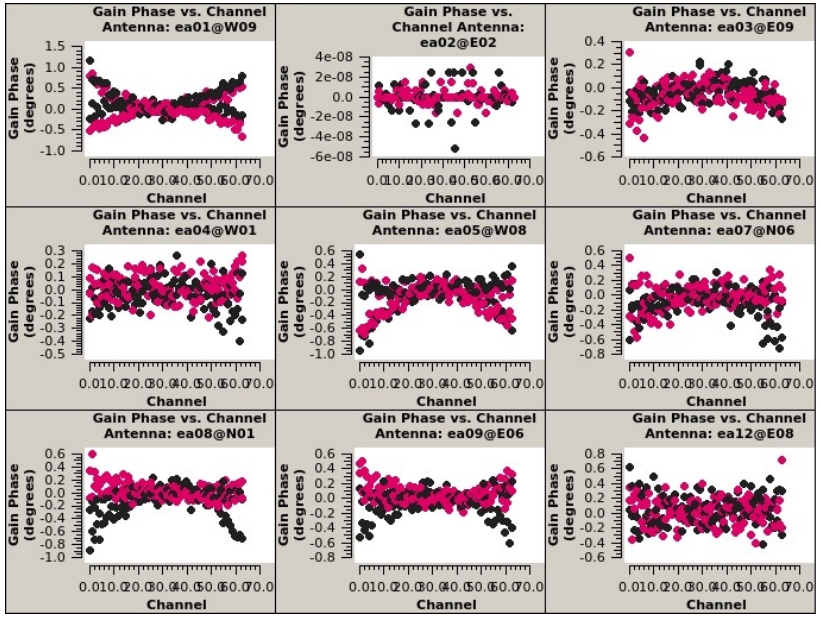
```
[vanilla@hopper radio_astronomy]$ file ant_locations.png
ant_locations.png: PNG image data, 640 x 480, 8-bit/color RGBA, non-
interlaced
```

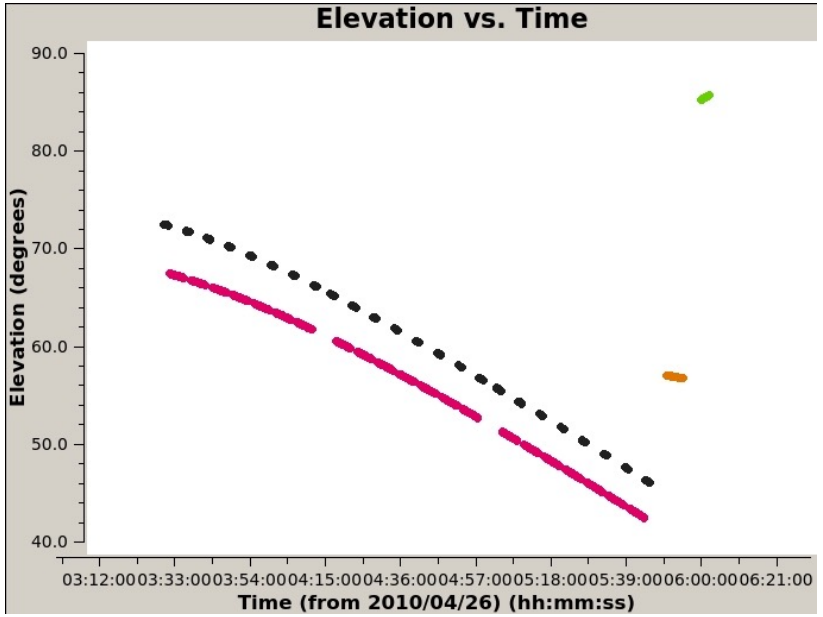


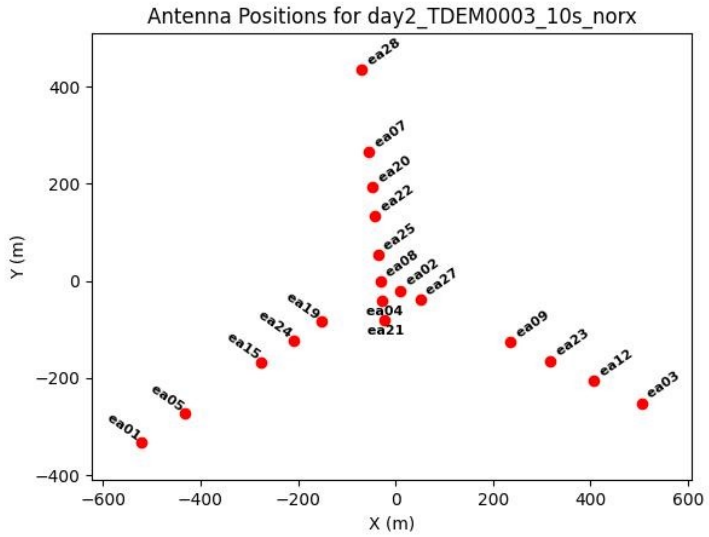
```
[vanilla@hopper radio_astronomy]$ cat casa_example1.sh
#!/bin/bash

#SBATCH --partition debug
#SBATCH --nodes 1
#SBATCH --ntasks-per-node 4
#SBATCH --time 00:10:00
#SBATCH --job-name CASAEX1
#SBATCH --mail-user your_username@unm.edu
#SBATCH --mail-type ALL

module load lsb_release
module load casa

mpicasa -n $SLURM_NTASKS xvfb-run -d casa -c casa_example1.py
```

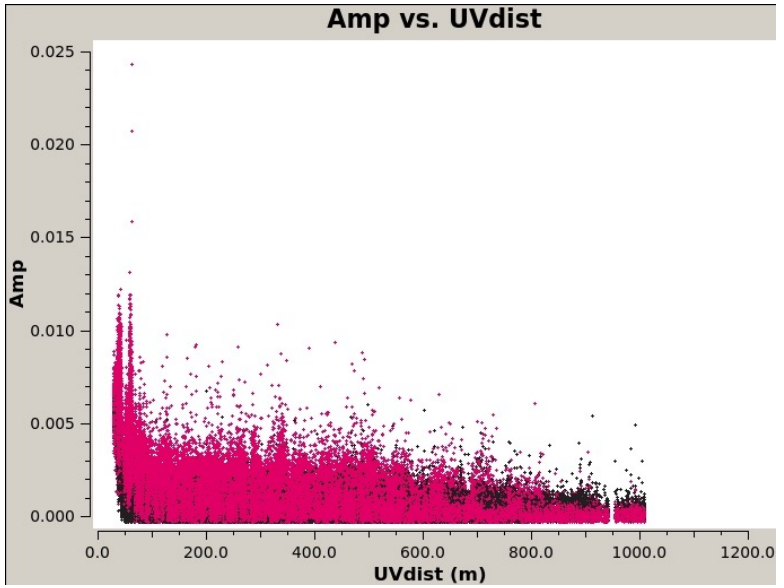
```
[vanilla@hopper radio_astronomy]$ cat casa_example1.py
visfile = 'day2_TDEM0003_10s_norx'

listobs(vis = visfile, listfile = 'casa_example1.list')

plotants(vis=visfile, figfile='example1_ant_locations.jpg')

plotms(vis=visfile,
       xaxis='time',yaxis='elevation',correlation='RR,LL',
       plotfile='time_elevation.jpg',
       avgchannel='64',spw='0:4~60', coloraxis='field')

plotms(vis=visfile,
       xaxis='time',yaxis='amp',correlation='RR,LL',
       plotfile='time_amp.jpg',
       avgchannel='64',spw='0:4~60', coloraxis='field’)

<snip>
```

```
[vanilla@hopper radio_astronomy]$ sbatch casa_example1.sh


[vanilla@hopper radio_astronomy]$ squeue --me
 JOBID PARTITION     NAME     USER    ST       TIME  NODES NODELIST(REASON)
 1817355     debug  CASAEX1  vanilla  R       0:14      1 hopper011
```


Gain Phase vs. Channel Antenna: ea01@W09
Gain Phase vs. Channel Antenna: ea02@E02
Gain Phase vs. Channel Antenna: ea03@E09
Gain Phase vs. Channel Antenna: ea04@W01
Gain Phase vs. Channel Antenna: ea05@W08
Gain Phase vs. Channel Antenna: ea07@N06
Gain Phase vs. Channel Antenna: ea08@N01
Gain Phase vs. Channel Antenna: ea09@E06
Gain Phase vs. Channel Antenna: ea12@E08
Gain Phase (degrees)
Channel


Elevation vs. Time
Elevation (degrees)
90.0
80.0
70.0
60.0
50.0
40.0
03:12:00
03:33:00
03:54:00
04:15:00
04:36:00
04:57:00
05:18:00
05:39:00
06:00:00
06:21:00
Time (from 2010/04/26) (hh:mm:ss)


Antenna Positions for day2_TDEM0003_10s_norx
ea28
ea07
ea20
ea22
ea25
ea08
ea02
ea27
ea04
ea21
ea19
ea24
ea15
ea05
ea01
ea09
ea23
ea12
ea03
Y (m)
X (m)


Amp vs. UVdist
Amp
UVdist (m)

```
[vanilla@hopper radio_astronomy]$ module load miniconda3

[vanilla@hopper radio_astronomy]$ conda env create -f
casa_env.yml
```

Install modular casa

```
[vanilla@hopper radio_astronomy]$ source activate casa

(casa) [vanilla@hopper radio_astronomy]$ python -m casatools --update-user-data
attempting to install/update runtime data in /users/vanilla/.casa/data
.....................................................................................
.....................................
```

Install casa user data…

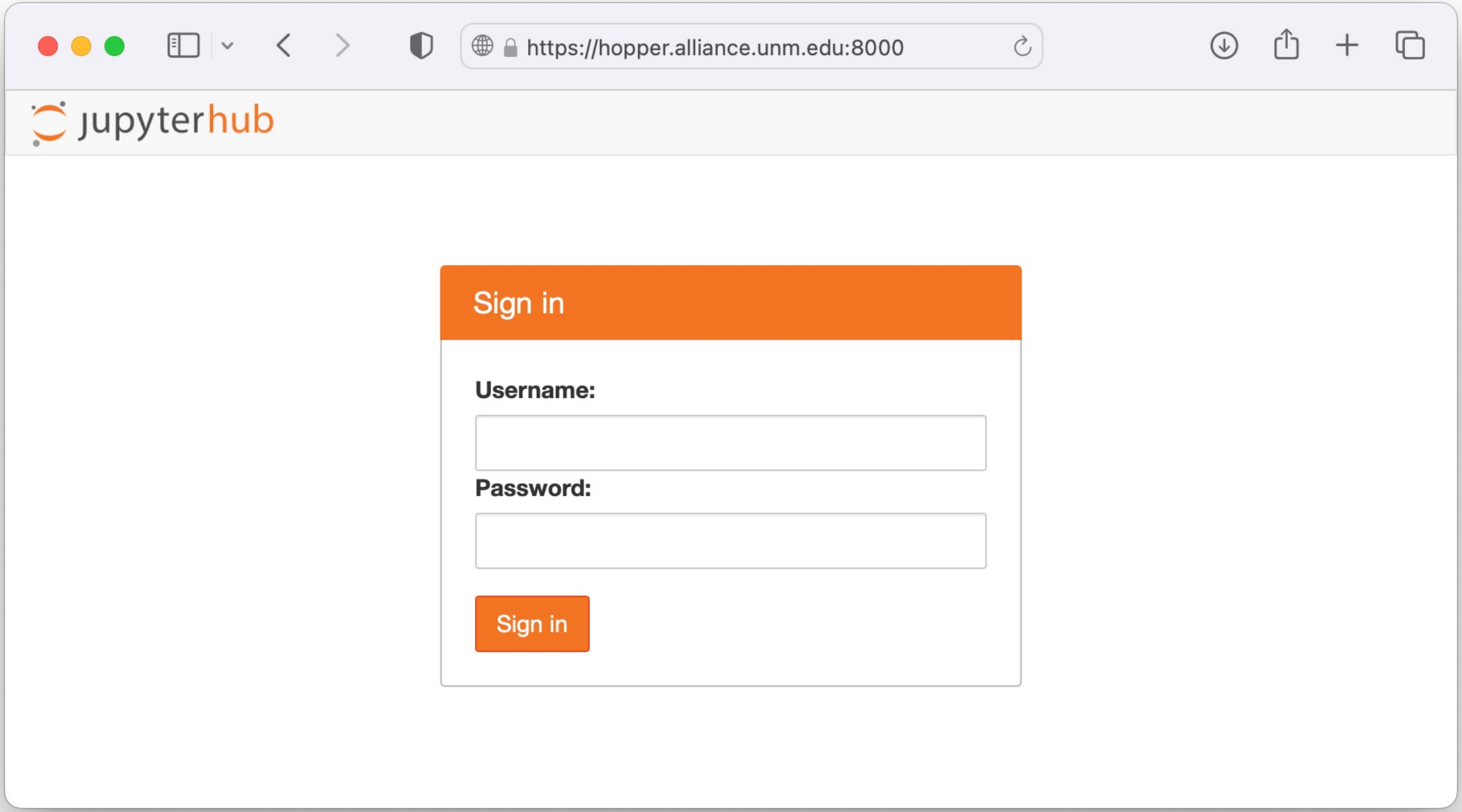
```
(casa) [vanilla@hopper radio_astronomy]$ pip install
pyvirtualdisplay

(casa) [vanilla@hopper radio_astronomy]$
```

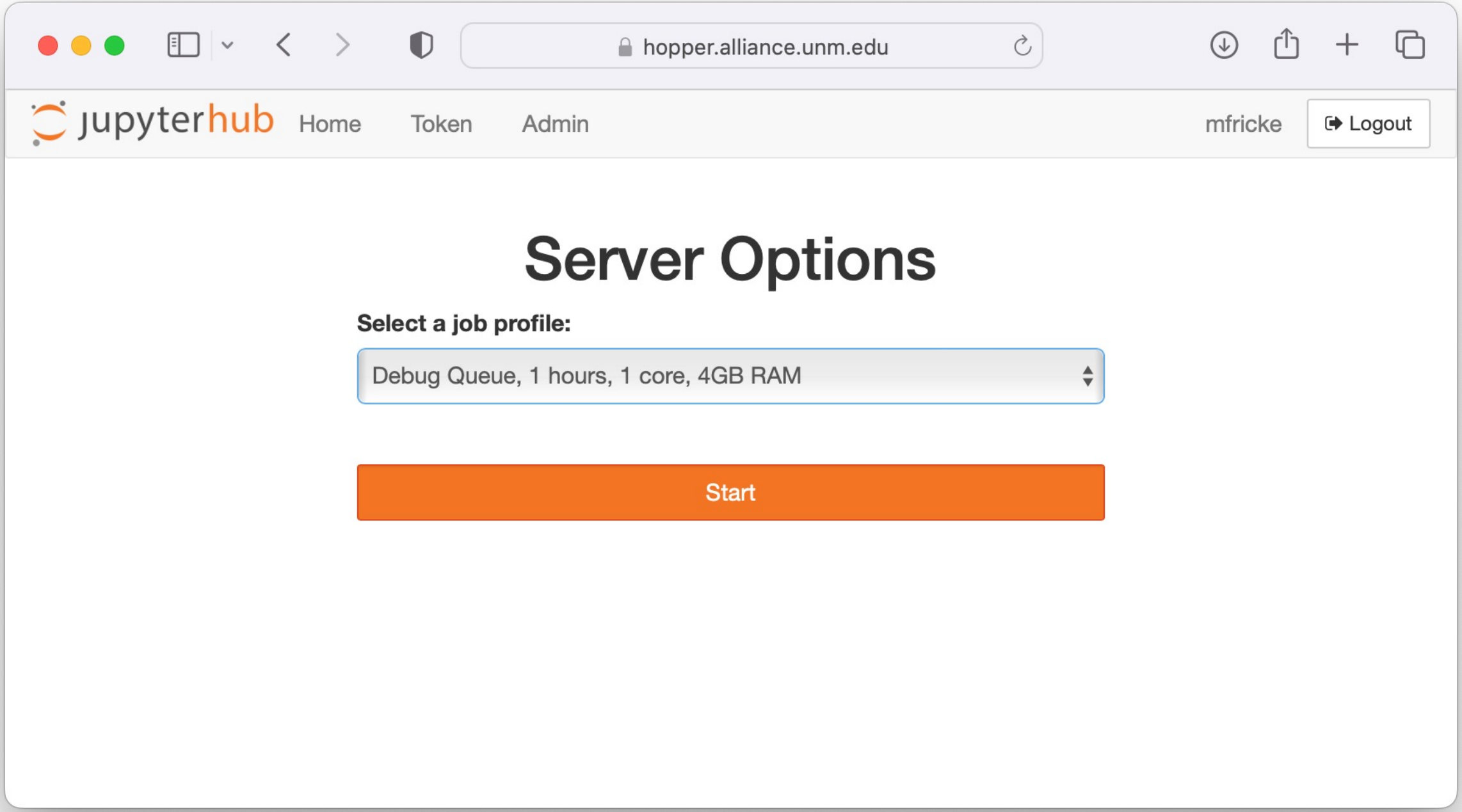
Install modular casa

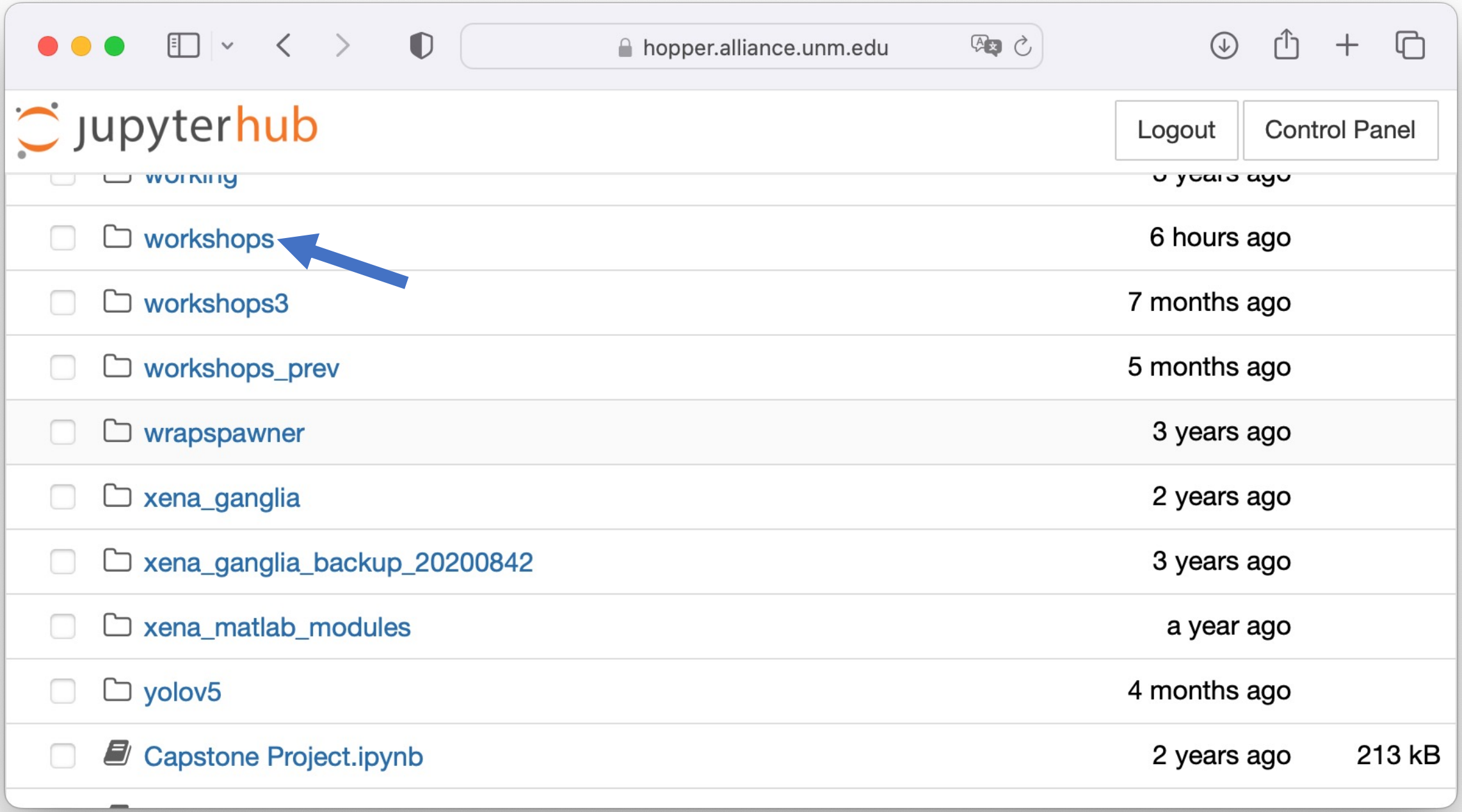
https://hopper.alliance.unm.edu:8000

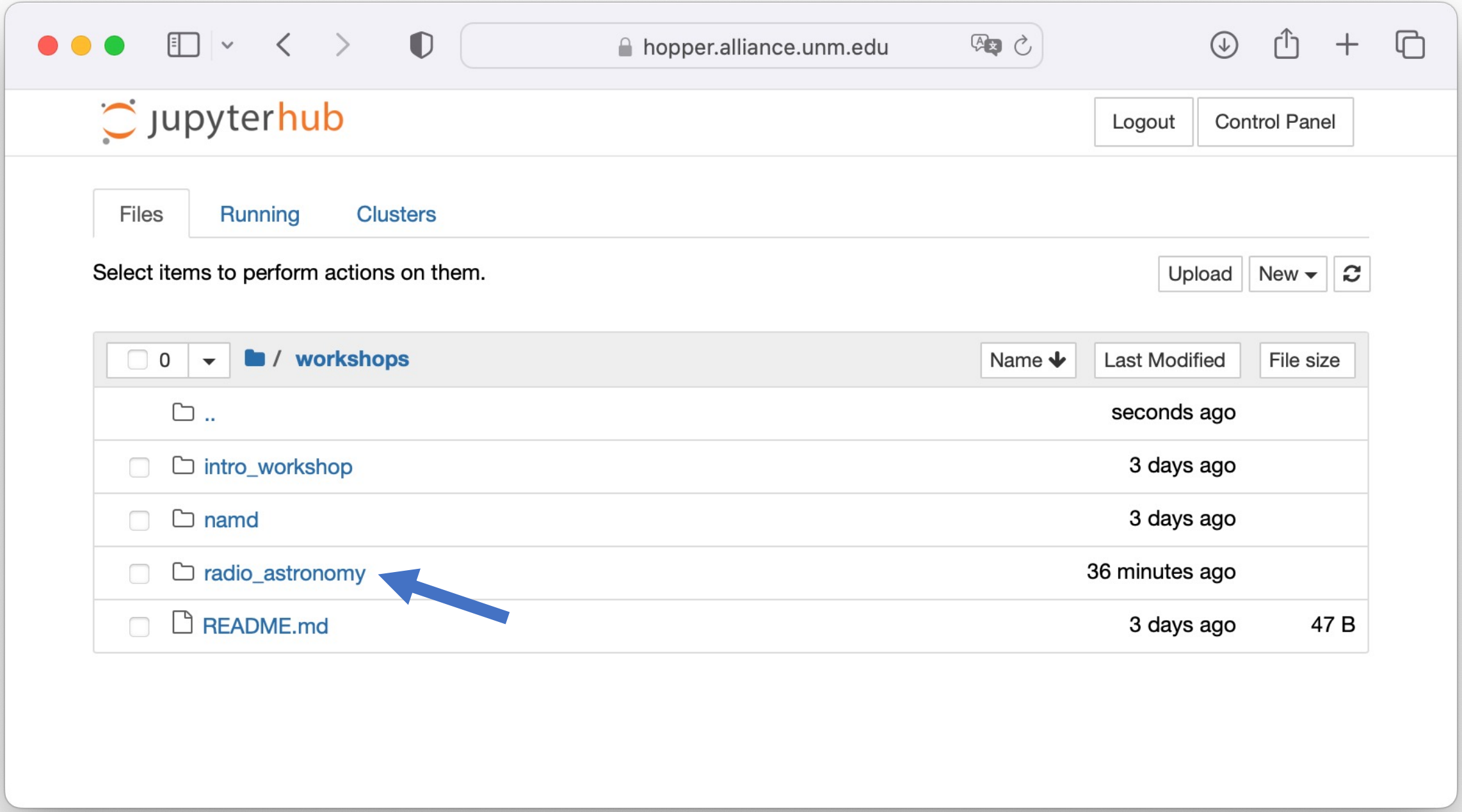
python™

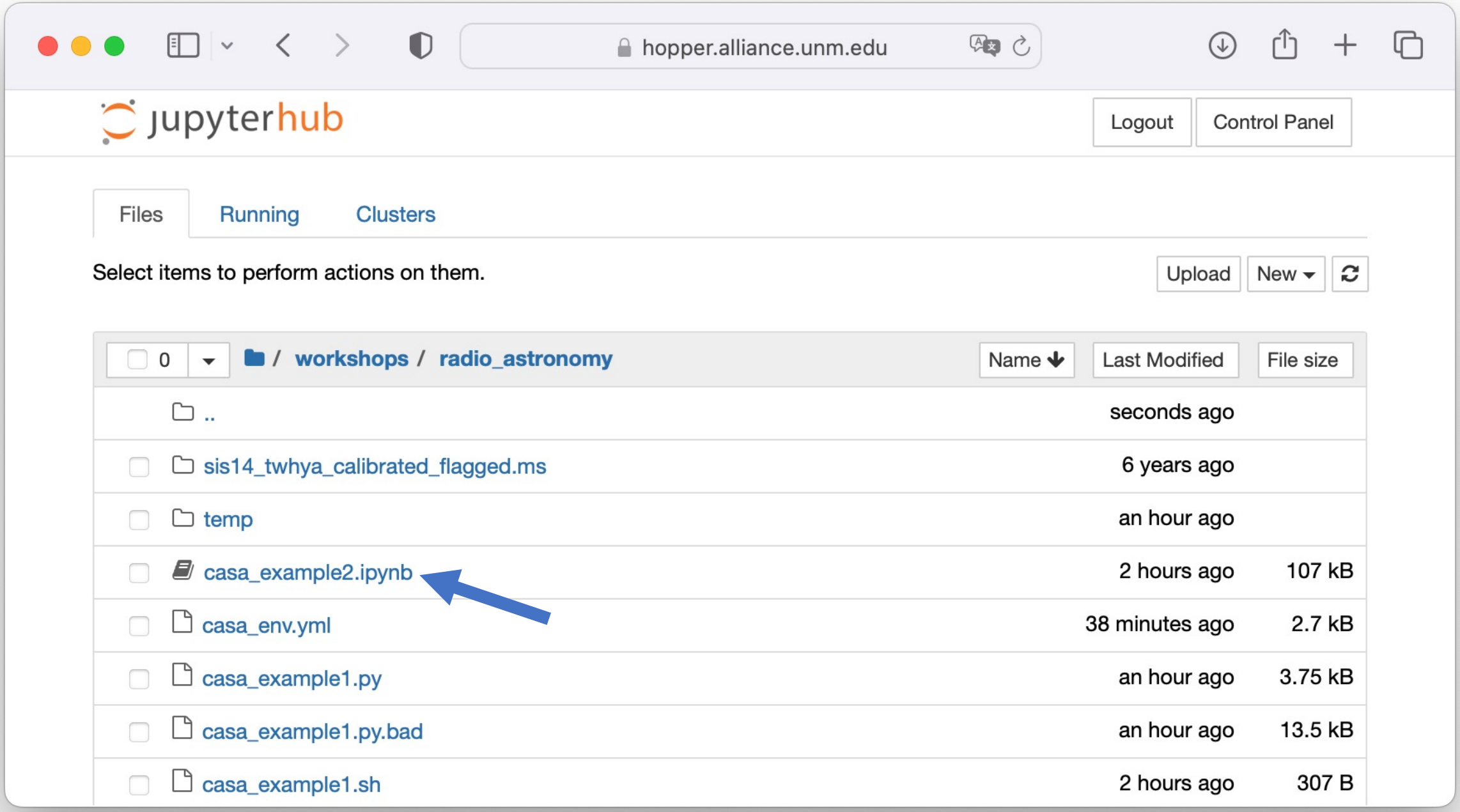
julia

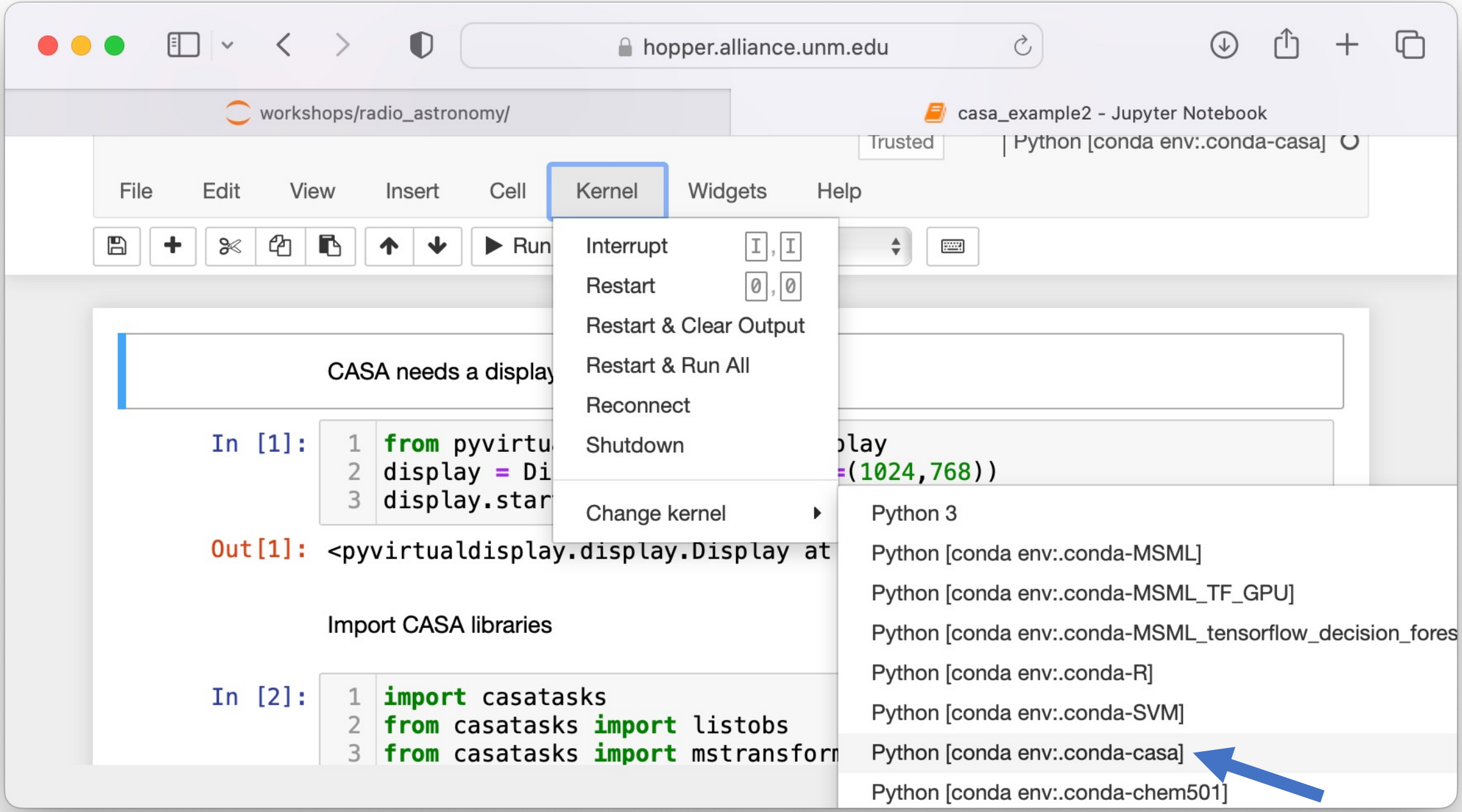

https://hopper.alliance.unm.edu:8000
jupyterhub
Sign in
Username:
Password:
Sign in

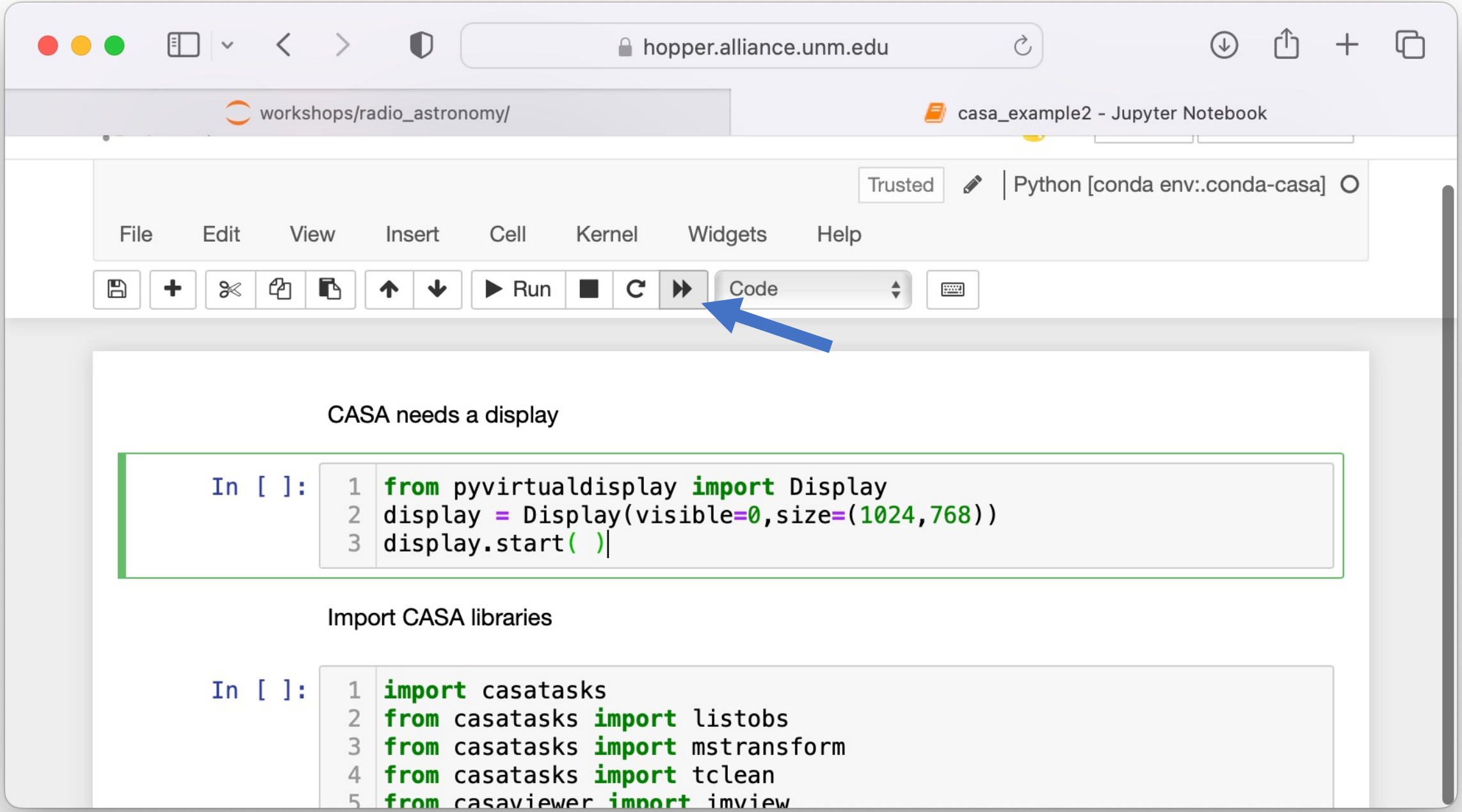

hopper.alliance.unm.edu
jupyterhub
Home
Token
Admin
mfricke
Logout
Server Options
Select a job profile:
Debug Queue, 1 hours, 1 core, 4GB RAM
Start

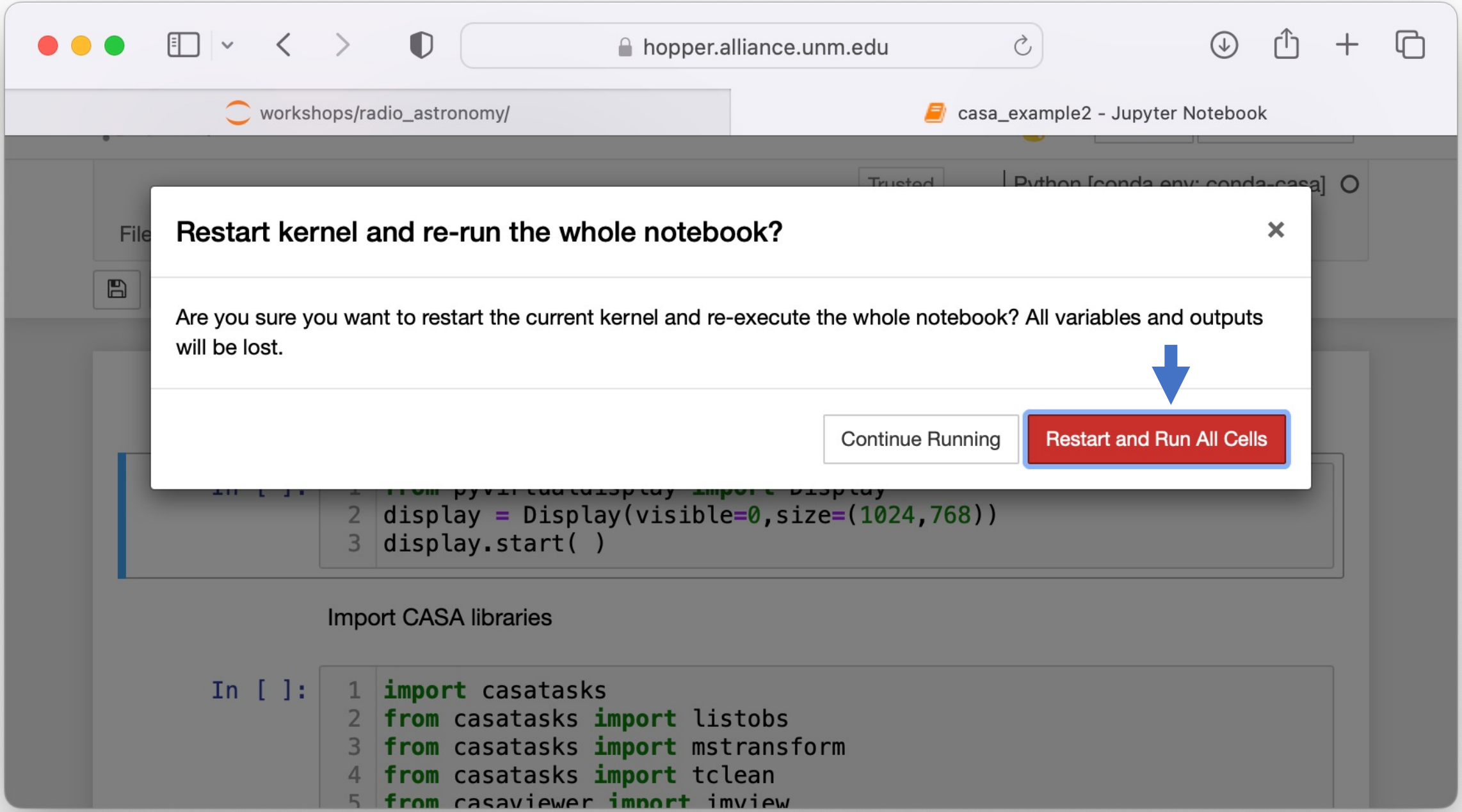

hopper.alliance.unm.edu
jupyterhub
Logout
Control Panel
workshops
6 hours ago
workshops3
7 months ago
workshops_prev
5 months ago
wrapspawner
3 years ago
xena_ganglia
2 years ago
xena_ganglia_backup_20200842
3 years ago
xena_matlab_modules
a year ago
yolov5
4 months ago
Capstone Project.ipynb
2 years ago
213 kB


hopper.alliance.unm.edu
jupyterhub
Logout
Control Panel
Files
Running
Clusters
Select items to perform actions on them.
Upload
New
0
/ workshops
Name
Last Modified
File size
..
seconds ago
intro_workshop
3 days ago
namd
3 days ago
radio_astronomy
36 minutes ago
README.md
3 days ago
47 B


hopper.alliance.unm.edu
jupyterhub
Logout
Control Panel
Files
Running
Clusters
Select items to perform actions on them.
Upload
New
0
/ workshops / radio_astronomy
Name
Last Modified
File size
..
seconds ago
sis14_twhya_calibrated_flagged.ms
6 years ago
temp
an hour ago
casa_example2.ipynb
2 hours ago
107 kB
casa_env.yml
38 minutes ago
2.7 kB
casa_example1.py
an hour ago
3.75 kB
casa_example1.py.bad
an hour ago
13.5 kB
casa_example1.sh
2 hours ago
307 B


hopper.alliance.unm.edu
workshops/radio_astronomy/
casa_example2 - Jupyter Notebook
Trusted
Python [conda env:.conda-casa]
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Run
Interrupt
Restart
Restart & Clear Output
Restart & Run All
Reconnect
Shutdown
Change kernel
Python 3
Python [conda env:.conda-MSML]
Python [conda env:.conda-MSML_TF_GPU]
Python [conda env:.conda-MSML_tensorflow_decision_fores
Python [conda env:.conda-R]
Python [conda env:.conda-SVM]
Python [conda env:.conda-casa]
Python [conda env:.conda-chem501]
CASA needs a displa
In [1]:
from pyvirtu
display = Di
display.star
lay
(1024,768))
Out[1]:
<pyvirtualdisplay.display.Display at
Import CASA libraries
In [2]:
import casatasks
from casatasks import listobs
from casatasks import mstransform


hopper.alliance.unm.edu
workshops/radio_astronomy/
casa_example2 - Jupyter Notebook
Trusted
Python [conda env:.conda-casa]
File Edit View Insert Cell Kernel Widgets Help
Run
Code
CASA needs a display
In [ ]:
1 from pyvirtualdisplay import Display
2 display = Display(visible=0,size=(1024,768))
3 display.start( )
Import CASA libraries
In [ ]:
1 import casatasks
2 from casatasks import listobs
3 from casatasks import mstransform
4 from casatasks import tclean
5 from casaviewer import imview


hopper.alliance.unm.edu
workshops/radio_astronomy/
casa_example2 - Jupyter Notebook
Restart kernel and re-run the whole notebook?
Are you sure you want to restart the current kernel and re-execute the whole notebook? All variables and outputs will be lost.
Continue Running
Restart and Run All Cells
display = Display(visible=0,size=(1024,768))
display.start( )
Import CASA libraries
In [ ]:
import casatasks
from casatasks import listobs
from casatasks import mstransform
from casatasks import tclean

hopper.alliance.unm.edu

workshops/radio_astronomy/

casa_example2 - Jupyter Notebook

Trusted | Python [conda env:.conda-casa]

File Edit View Insert Cell Kernel Widgets Help

Run Code

```
running tclean, may take a bit...

0%....10....20....30....40....50....60....70....80....90....100%

0%....10....20....30....40....50....60....70....80....90....100%
```

Convert the output image into a png file...

In [*]:
```
1 imview('first_image.image', out='test.png')
```

Display the image...

In [*]:
```
1 Image(filename="test.png")
```

workshops/radio_astronomy/ | casa_example2 - Jupyter Notebook

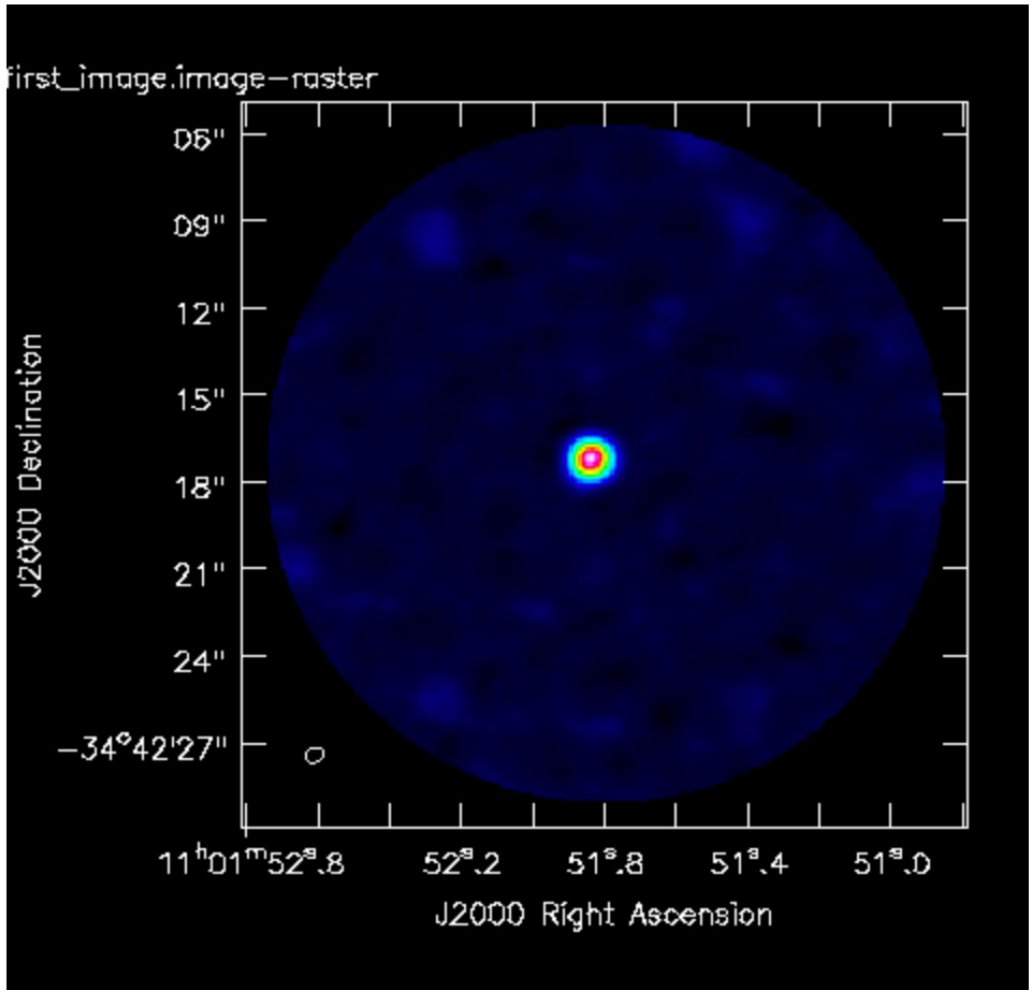
jupyterhub **casa_example2** Last Checkpoint: 2 hours ago (autosaved) Logout Control Panel

File Edit View Insert Cell Kernel Widgets Help Trusted Python [conda env:.conda-casa]

Run Code

In [10]:
```
1 Image(filename="test.png")
```

Out[10]:



You have learned

- how to run programs using the SLURM scheduler
- the difference between interactive and batch jobs
- how to check the status of your jobs
- how to ask for the hardware resources you need
- you ran CASA interactively in the terminal and in jupyterhub and in batch mode