Module 6: Monitoring Resource Use

Website: www.rc.colorado.edu

Documentation: https://curc.readthedocs.io

Helpdesk: rc-help@colorado.edu

Meet the User Support Team

Layla Freeborn

Brandon Reyes

Andy Monaghan

Michael Schneider

John Reiland

Dylan Gottlieb

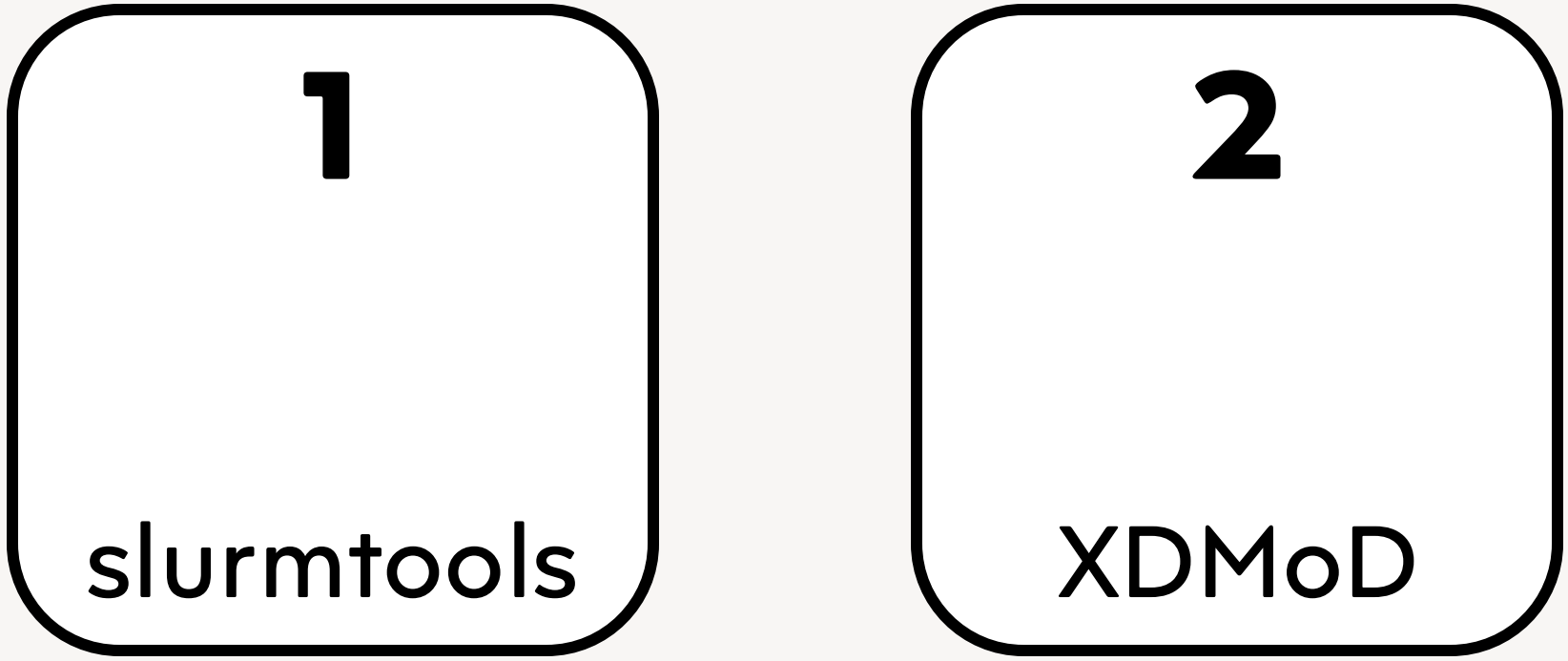
Mohal Khandelwal

Ragan Lee

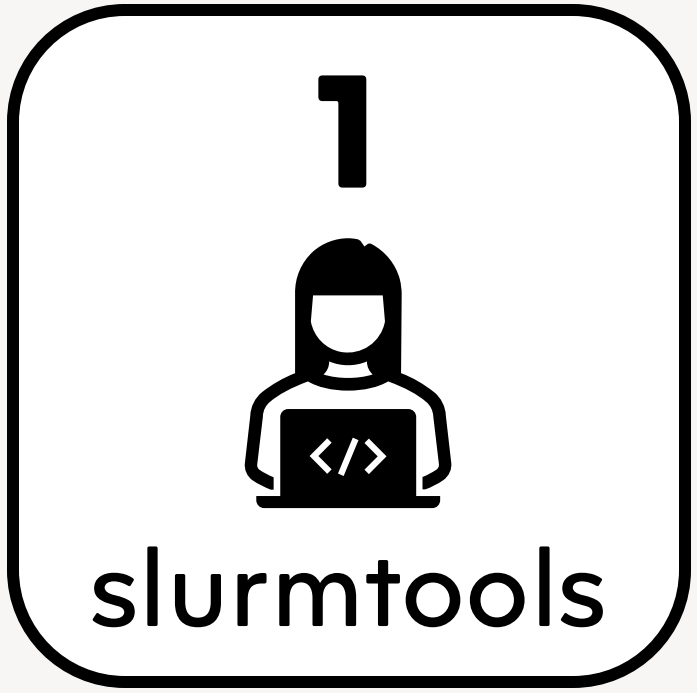
Learning Objectives

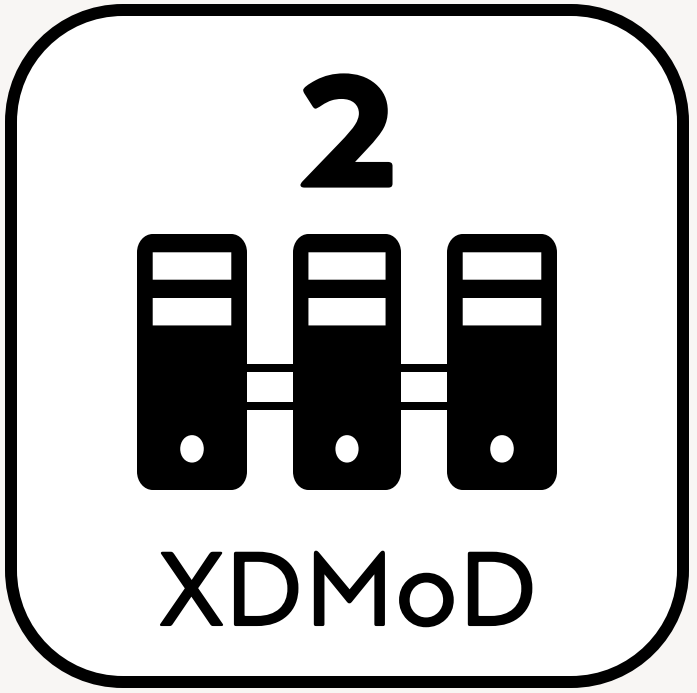
- Calculate your resource usage/consumption
- View average wait times in the CURC queues
- Find your relative "priority”
- Check the efficiency of your research workflows

Overview



Overview



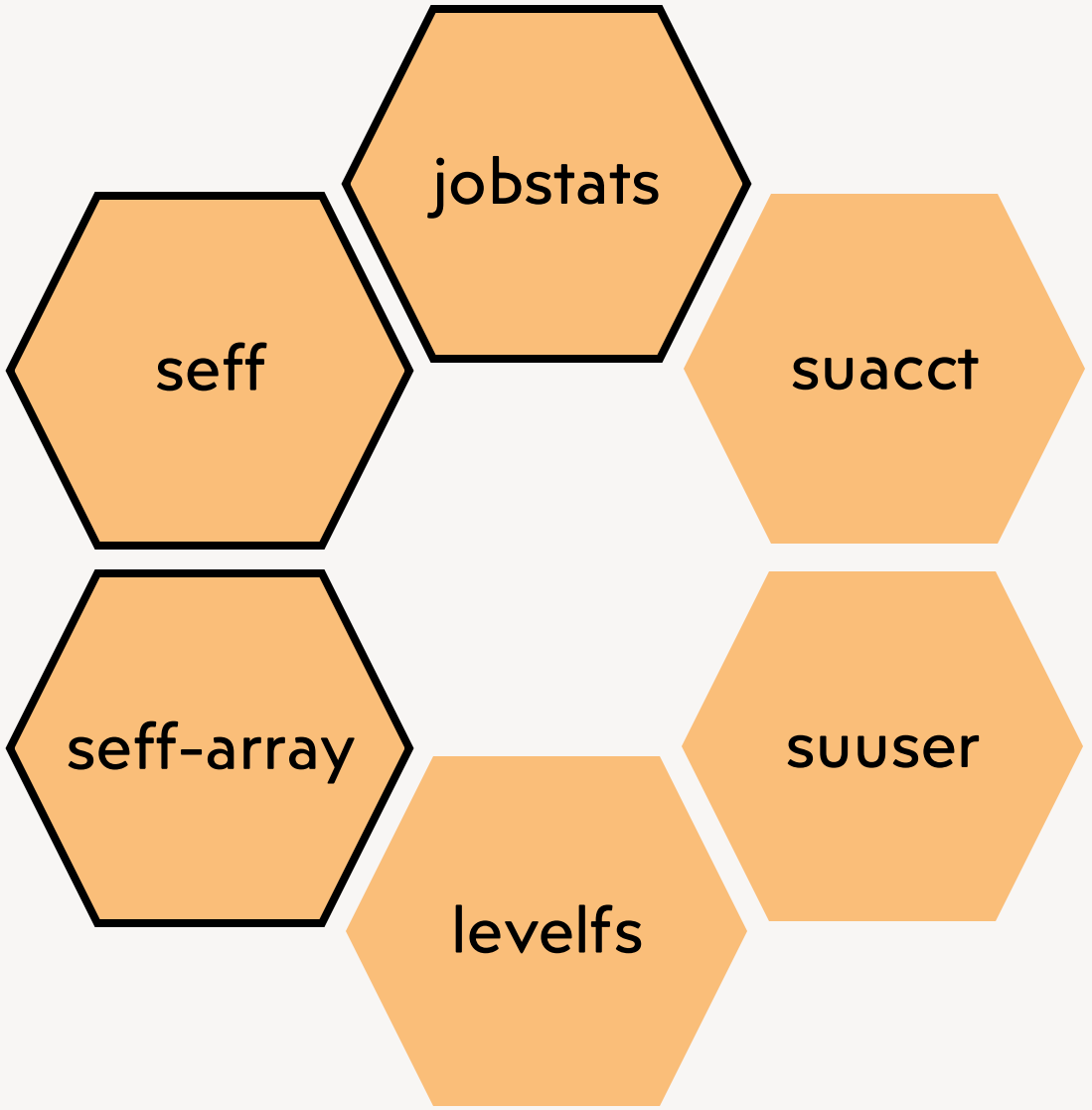


The slurmtools Module

A module that loads a collection of functions to assess recent usage statistics

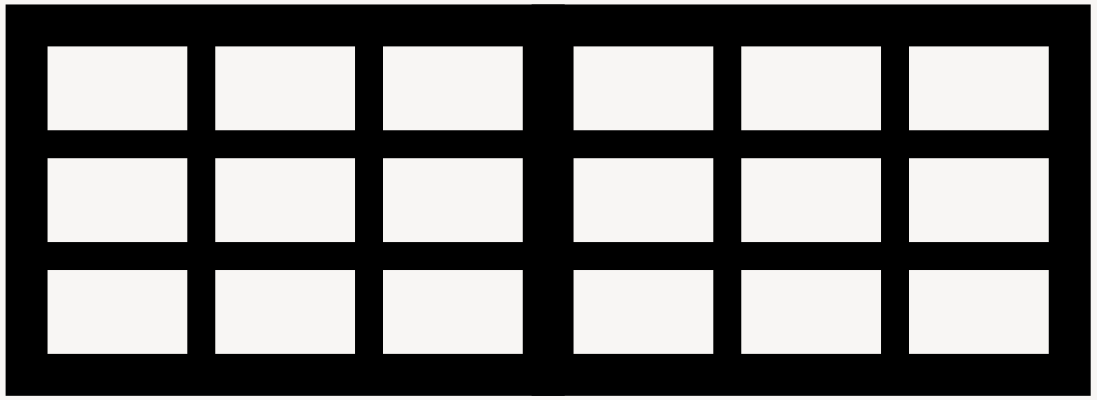
```
$ module load slurmtools
```

Job Metrics



jobstats

```
$ jobstats $USER <# Days>
```



- jobid
- jobname
- partition
- qos
- account
- cpus
- state
- start-date-time
- elapsed
- wait

seff

```
$ seff <jobid>
```

seff

```
$ seff <jobid>
```

Job ID: 12345

Cluster: alpine

User/Group: ralphie/ralphiepgrp

State: COMPLETED (exit code 0)

Nodes: 1

Cores per node: 4

CPU Utilized: 01:00:07

CPU Efficiency: 25.00% of 04:00:28 core-walltime

Job Wall-clock time: 01:00:07

Memory Utilized: 21.14 MB (estimated maximum)

Memory Efficiency: 0.15% of 13.59 GB (3.40 GB/core)

seff

```
$ seff <jobid>
```

Job ID: 12345

Cluster: alpine

User/Group: ralphie/ralphiepgrp

State: COMPLETED (exit code 0)

Nodes: 1

Cores per node: 4

CPU Utilized: 01:00:07

CPU Efficiency: 25.00% of 04:00:28 core-walltime

Job Wall-clock time: 01:00:07

Memory Utilized: 21.14 MB (estimated maximum)

Memory Efficiency: 0.15% of 13.59 GB (3.40 GB/core)

Total CPU Time

seff

```
$ seff <jobid>
```

Job ID: 12345

Cluster: alpine

User/Group: ralphie/ralphiepgrp

State: COMPLETED (exit code 0)

Nodes: 1

Cores per node: 4

CPU Utilized: 01:00:07

CPU Efficiency: 25.00% of 04:00:28 core-walltime

Job Wall-clock time: 01:00:07

Memory Utilized: 21.14 MB (estimated maximum)

Memory Efficiency: 0.15% of 13.59 GB (3.40 GB/core)

Max Memory Used

seff-array

```
$ seff-array <jobid>
```

seff-array

Allocated Resources Per Job

```
$ seff-array <jobid>
```

```
Job Information
ID: 12345678
Name: job_array.sh
Cluster: alpine
User/Group: ralphie/ralphiepgrp
Requested CPUs: 1 cores on 1 node(s)
Requested Memory: 1.88G
Requested Time: 00:10:00
```

seff-array

```
$ seff-array <jobid>
```

```
Job Status
COMPLETED: 3
---------------------------------
Finished Job Statistics
Average CPU Efficiency 0.10%
Average Memory Usage 0.00G
Average Run-time 42.00s
```

Allocated Resources Per Job

Average Job Efficiency

seff-array

```
$ seff-array <jobid>
```

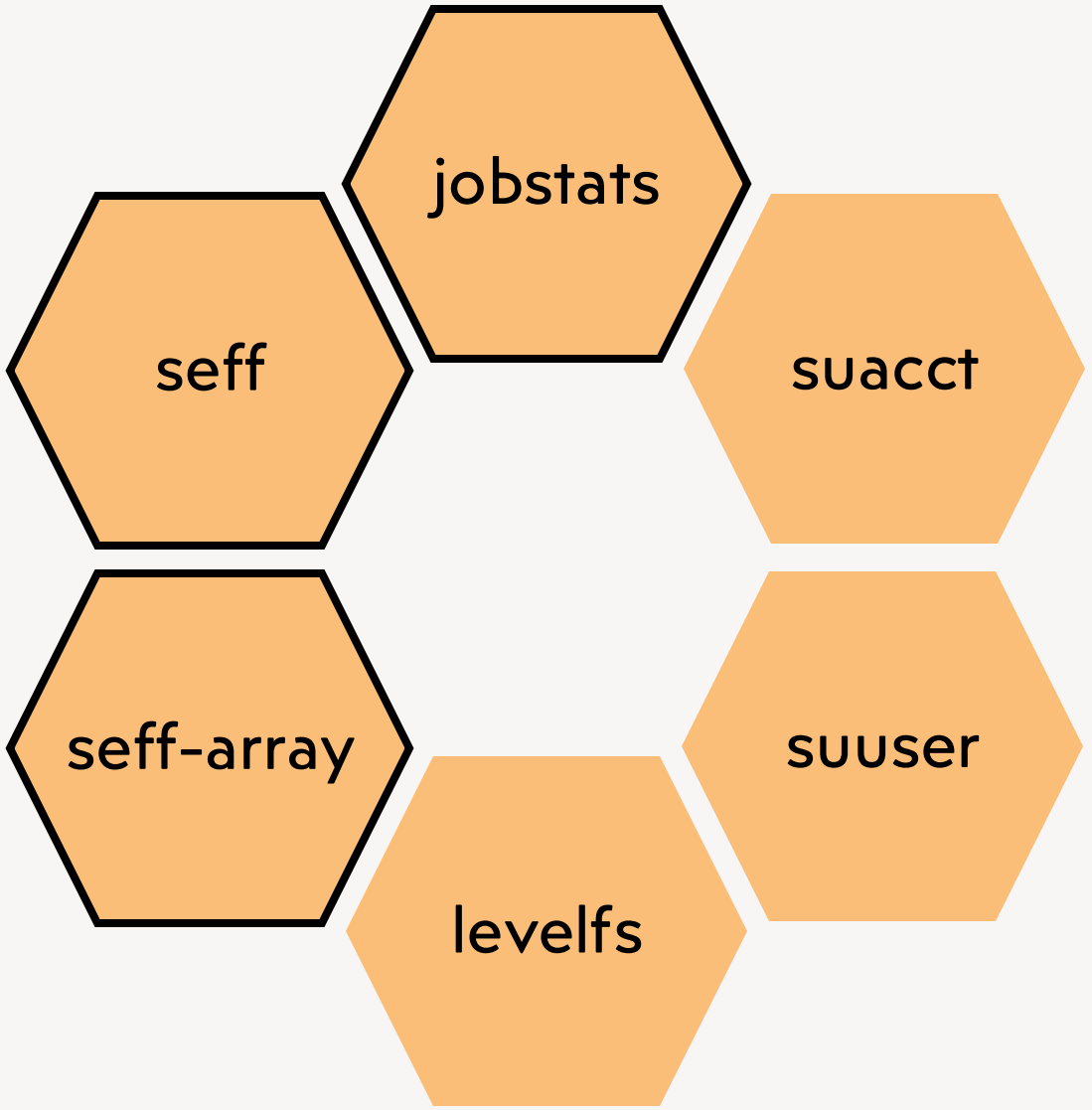
```
CPU Efficiency (%)
--------------------
+0.00e+00 - +1.00e+01   [2] ■■■■■■■■■■
+1.00e+01 - +2.00e+01   [1] ■■■
+2.00e+01 - +3.00e+01   [0]
[…]
+8.00e+01 - +9.00e+01   [0]
+9.00e+01 - +1.00e+02   [0]
```

Allocated Resources Per Job

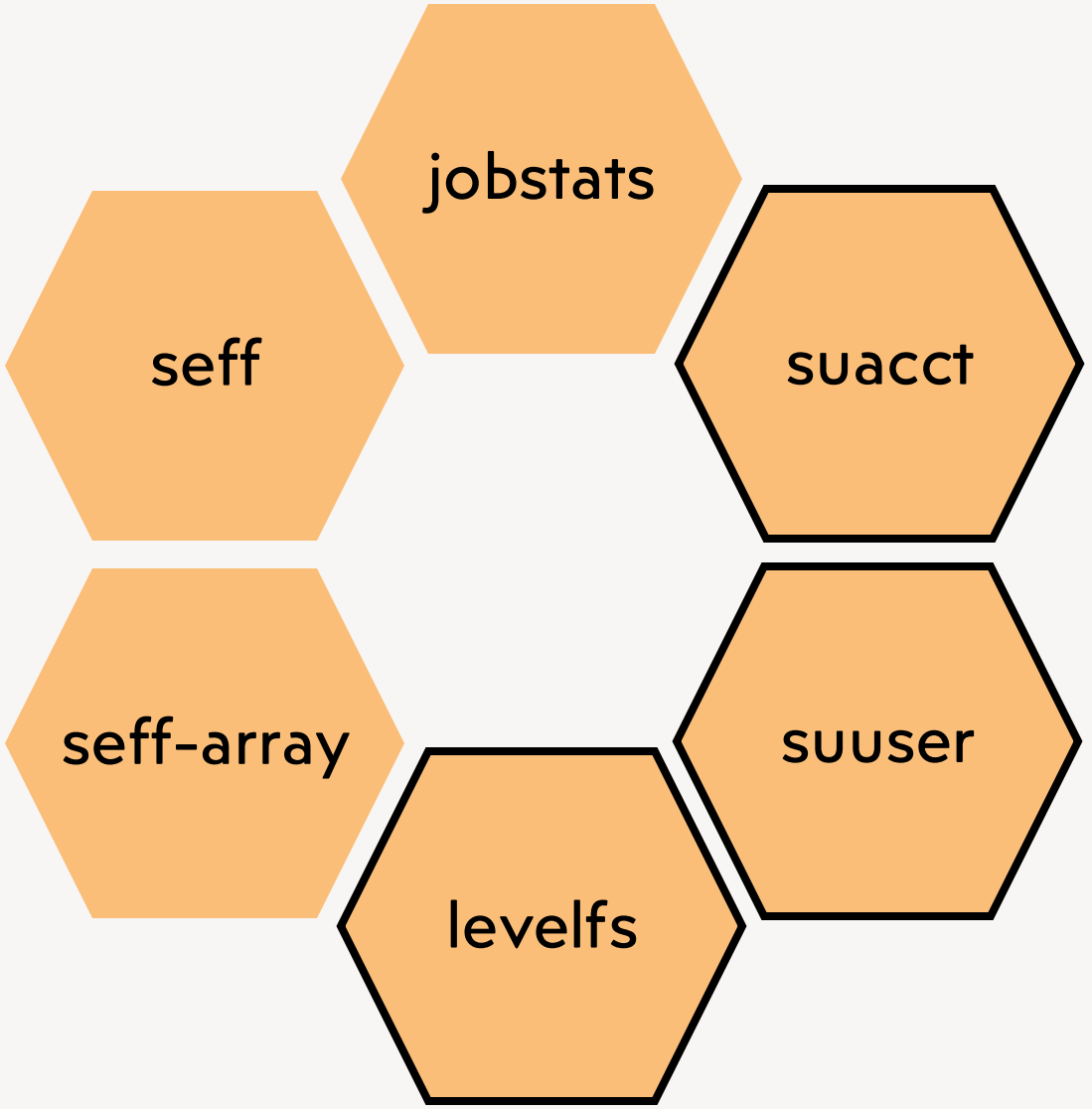
Average Job Efficiency

Average Job Efficiency

Job Metrics



Job Metrics



Job Priority

Priority

- A job's priority determines its position in the queue.
- A job's priority is based on multiple factors, including:
  - ❑ FairShare score
  - ❑ Age
  - ❑ Resources requested
  - ❑ Job size
  - ❑ QOS

levelfs

```
$ levelfs <username>
```

```
LevelFS for user ralphie and institution ucb:

Account              LevelFS_User         LevelFS_Inst
-------------------------------------------------------
ucb-general          2.015951             0.943096
```

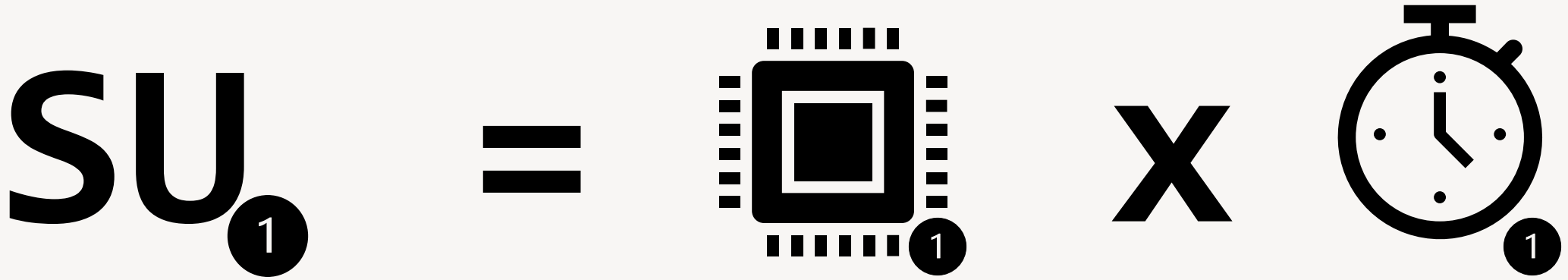
> 1

High Priority

< 1

Low Priority

Service Units



Service units (SUs), sometimes called “core hours”, reflect the processing that a core performs in one hour modified by some scaling factor

suuser

How many Service Units (SUs) have I used?

```
$ suuser <username>
```

```
SU used by user ralphie in the last 30 days:
Cluster |Account     |Login   |Proper Name|TRES Name|Used|
alpine  |ucb-general|ralphie|R. Buffalo |billing  |283 |
```

suacct

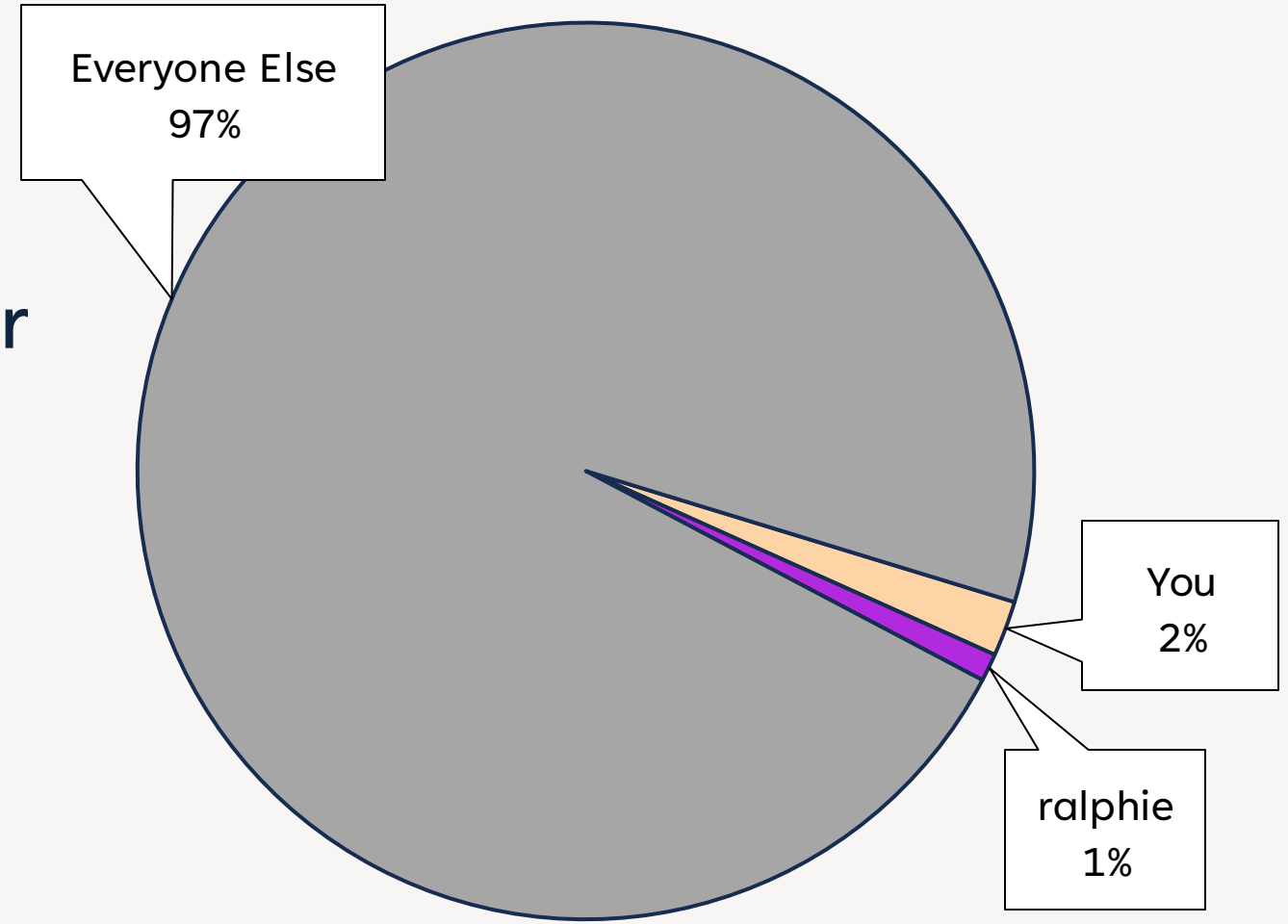
Who is using all the SUs on my group's account?

```
$ suacct <account / allocation>
```

```
SU used by account ucb123_asc1 in the last 30 days:
Cluster|Account    |Login  |Proper Name|TRES Name|Used|
alpine |ucb123_asc1|       |           |billing  |5802|
alpine |ucb123_asc1|ralphie|R. Buffalo |billing  |5802|
```

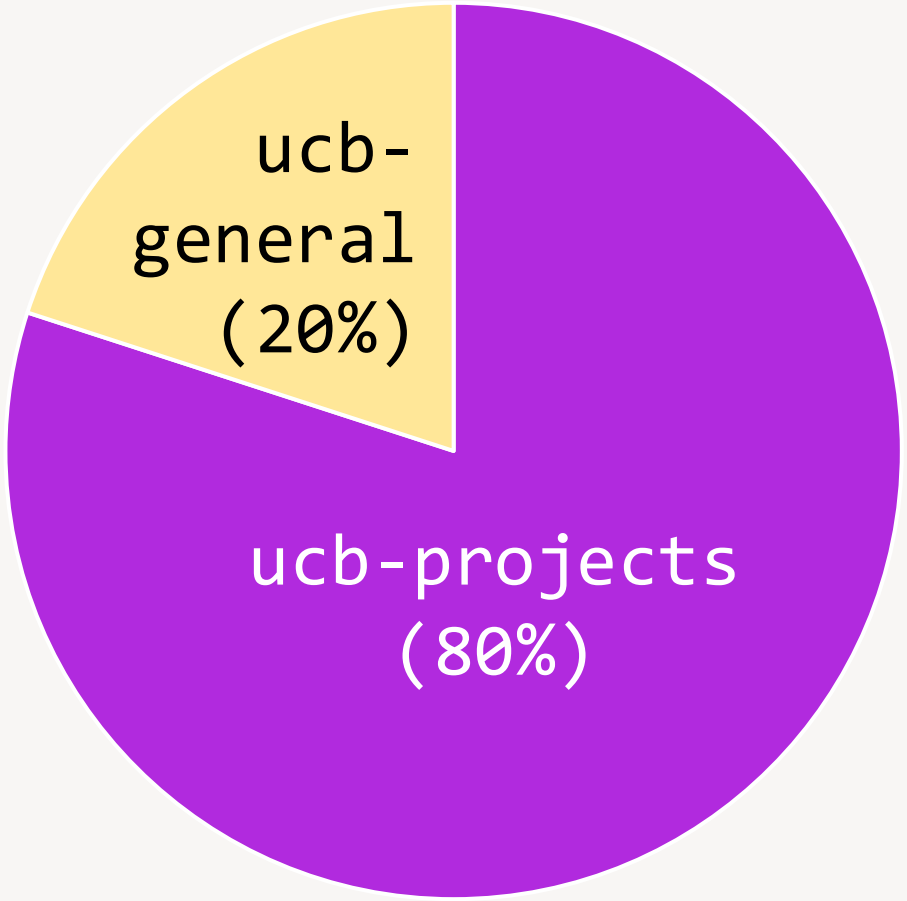
Allocations

- An allocation is a way for us to specify your cut of Alpine's computational resources.



general vs projects

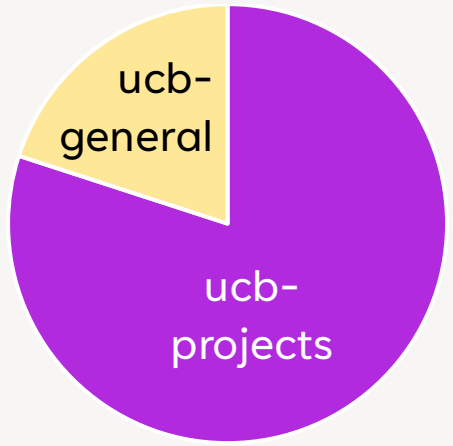
The ‘general’ and ‘project’ accounts are structured such that your jobs are likely to have a **higher** priority if they are running in the ‘**project**’ account.



Alpine Allocation Tiers (CU Boulder)

Trailhead

- Automatic with a CURC account.
- ~couple thousand SUs in **ucb-general**
- Does not expire



Ascent

Peak

Ascent vs Peak Allocations

Ascent

- 350,000* SUs over 12 months
- Quick approval process
- Easy to renew each year

*** Max 1.6M per Research Group**

Peak

- Up to 6,000,000 SUs over 12 months
- Detailed review of past jobs
- Renewal must justify size

Ascent Allocations (RMACC)

Ascent

- 100,000* SUs over 12 months
- Quick approval process
- Easy to renew each year

*** Max 400K per Research Group**

Using Your Allocation

Allocations are referred to as accounts in Slurm's documentation and are indicated by the --account directive:

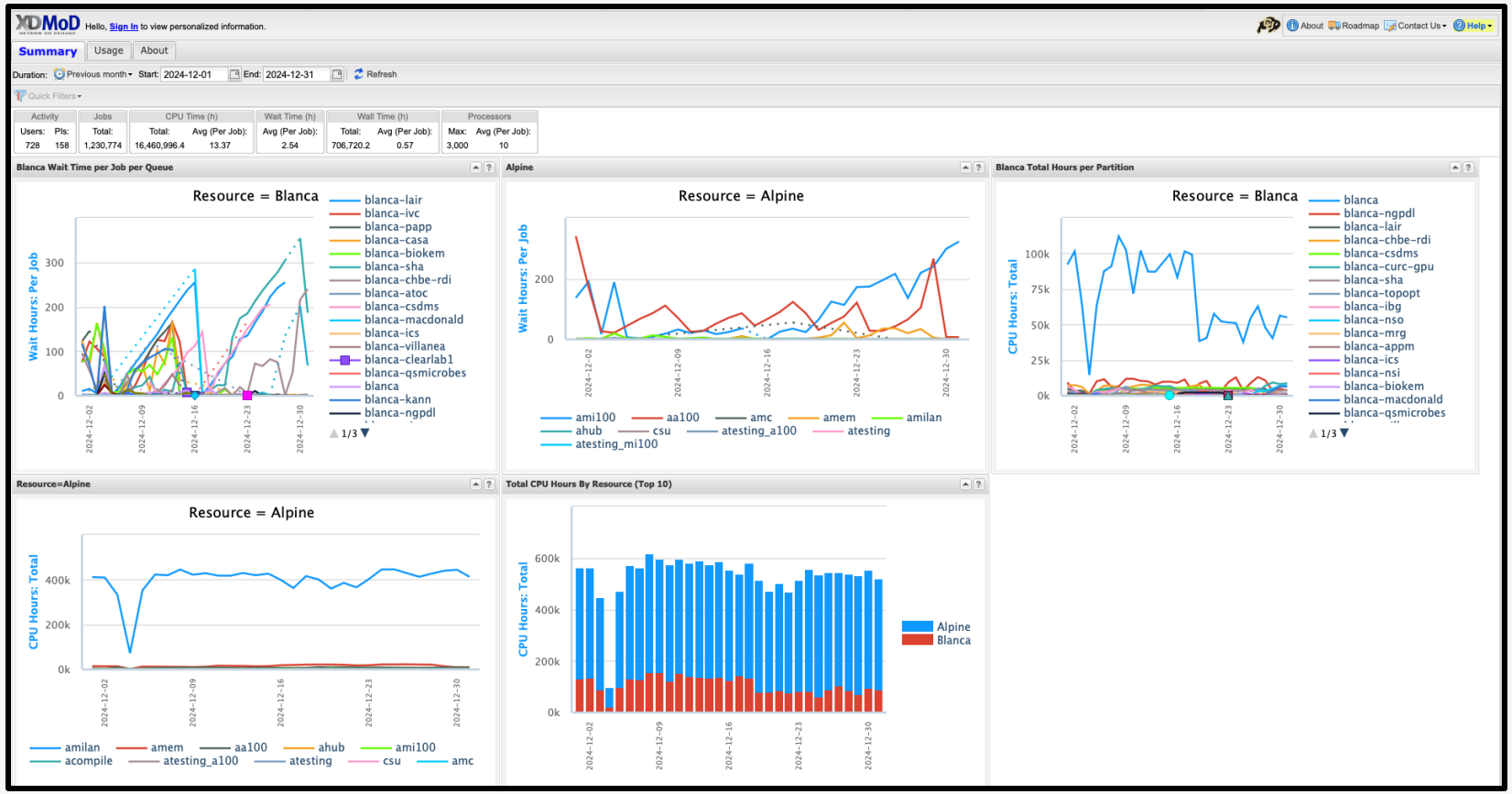
```
#SBATCH --account=<your allocation name>
```

or

```
sbatch --account <your allocation name> yourjob.sh
```

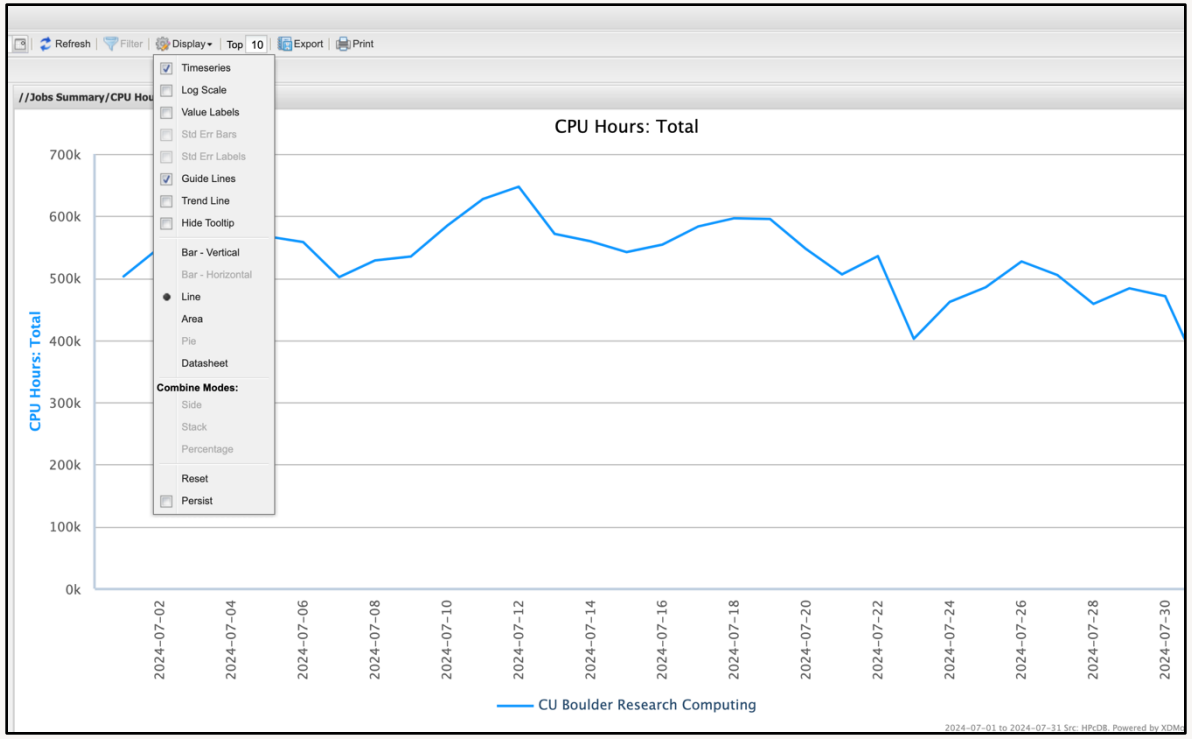

NSF
XDMoD
METRICS ON DEMAND

XDMoD

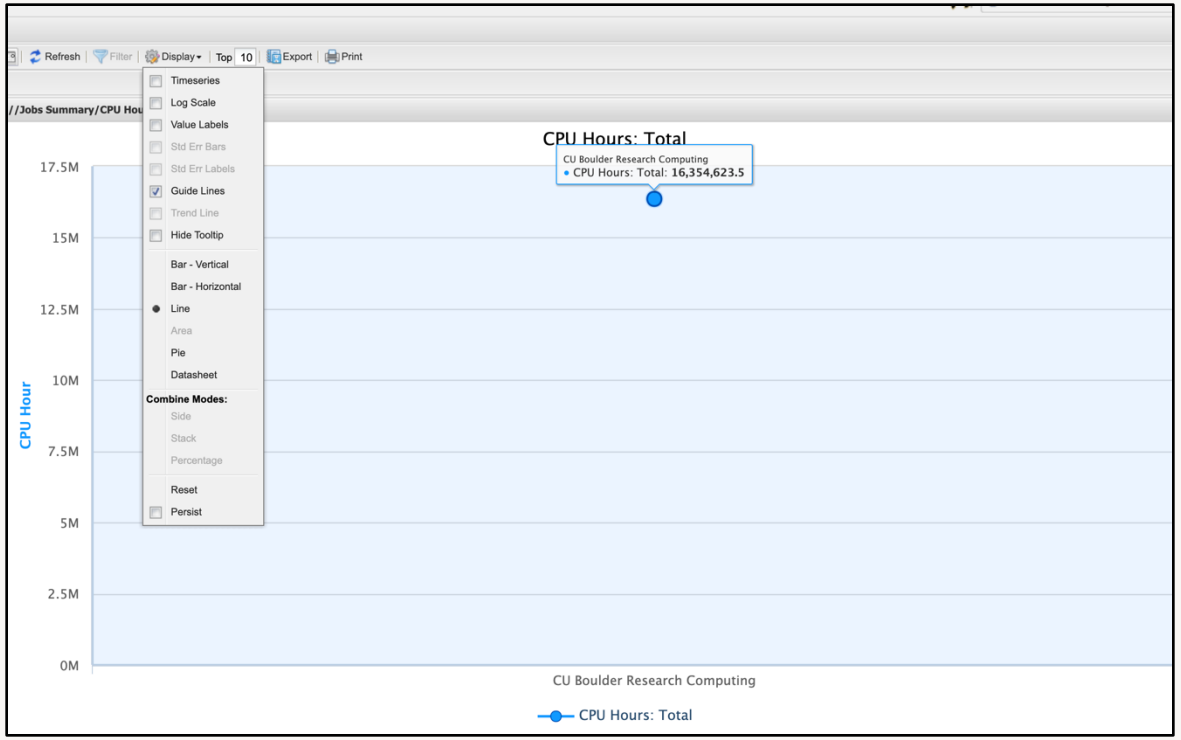


https://xdmod.rc.colorado.edu/

XDMoD Display Menu Defaults

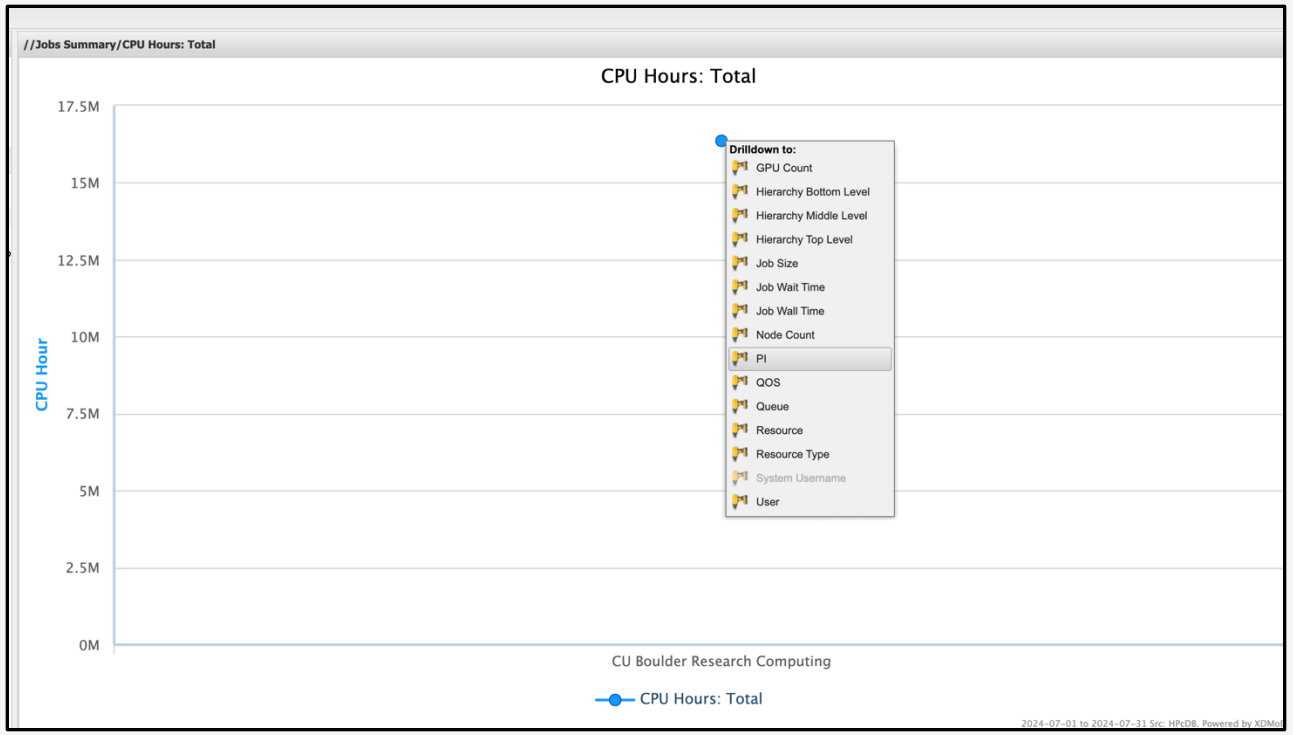


Display Menu: timeseries checked (default)

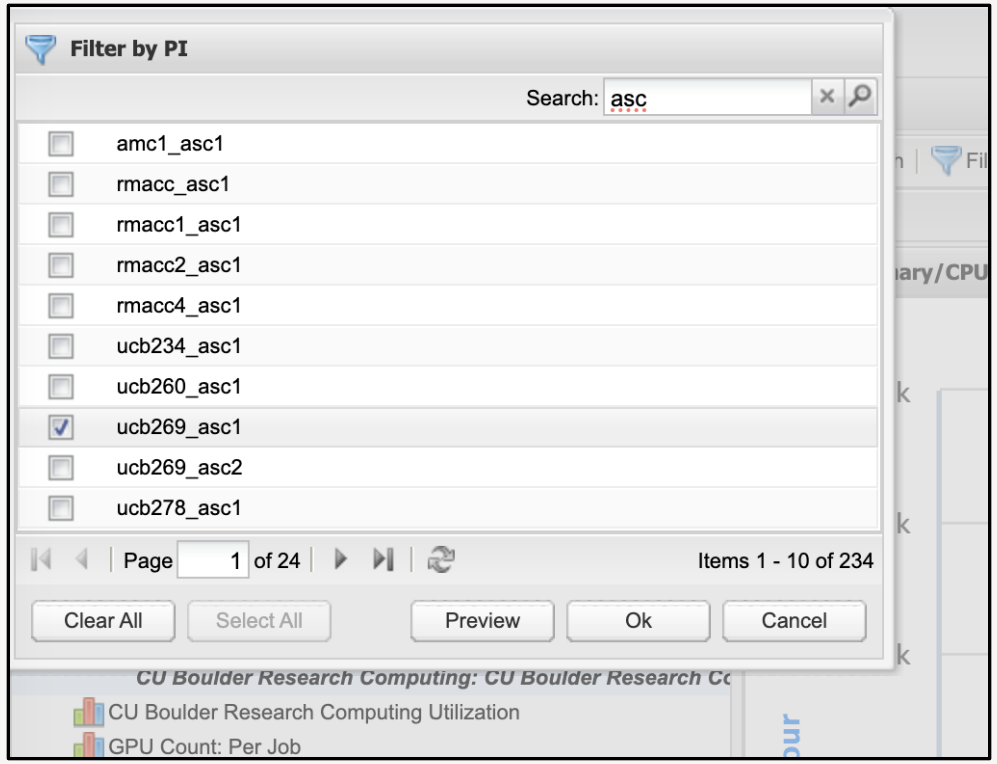


Display Menu: timeseries unchecked

XDMoD Drilldown and Filter



Click on a data point to see Drilldown menu
(PI = Slurm allocation)



Filter and search

Feedback Survey

http://tinyurl.com/curc-survey18