UC **SANTA BARBARA**

CNSI MRL

CSC Center for Scientific Computing

ENTERPRISE TECHNOLOGY SERVICES

HPC Workshop

October 21 or 25, 2022

11:30 – 12:30 pm (followed by lunch)

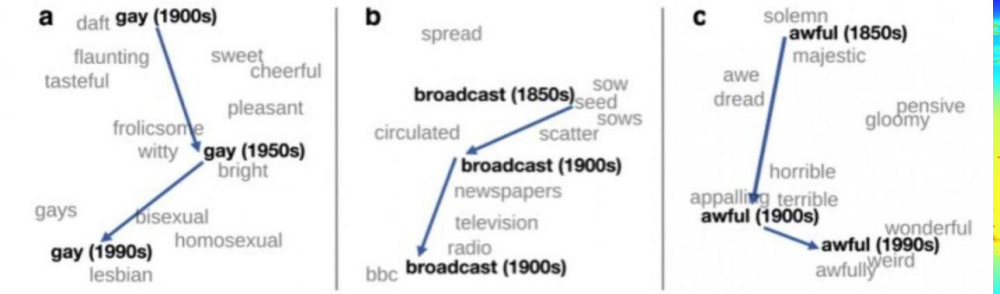
Location: Elings Hall 1601

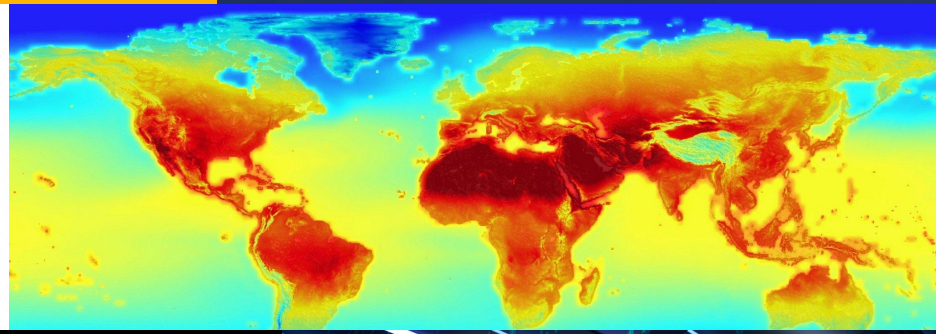
Register @ https://csc.cnsi.ucsb.edu

Quickly start using HPC resource at UCSB

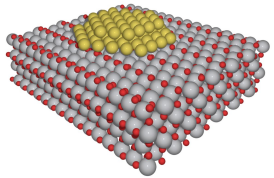
- What is HPC?
- Quickly get Started to Use HPC
- Basic Linux Commands
- Basic Slurm Commands
- National HPC & Commercial cloud computing Resources

Computational Linguistics





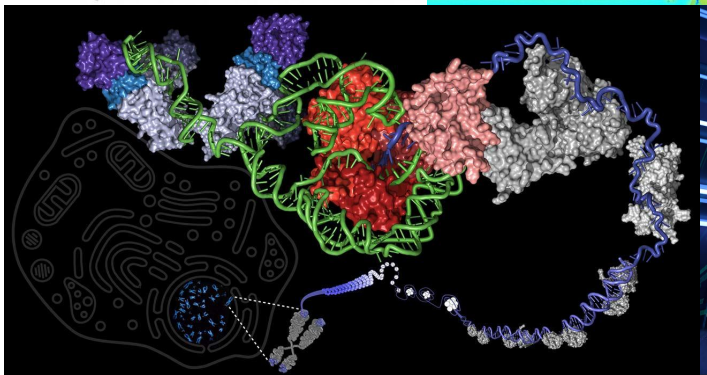
KS-DFT



The total energy in Kohn-Sham Density Functional Theory (KS-DFT) is expressed as

$$E_{total} = T_s + \int d\boldsymbol{r} V_{ext}(\boldsymbol{r})\rho(\boldsymbol{r}) + E_{xc}[\rho] + \frac{1}{2}\int\int d\boldsymbol{r}d\boldsymbol{r}' \frac{\rho(\boldsymbol{r})\rho(\boldsymbol{r}')}{|\boldsymbol{r}-\boldsymbol{r}'|}$$

Kohn-Sham kinetic energy | External energy | XC energy | Hartree energy



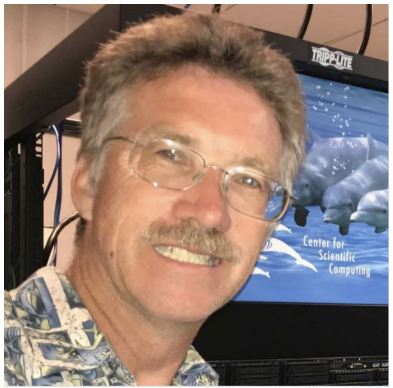
UC **SANTA BARBARA**

Introduction to High-Performance Computing (HPC) Resources and Linux

Paul Weakliem, Fuzzy Rogers, and Jay Chi

October 21, 2022

Speaker Introductions



Paul Weakliem, PhD
Co-Director
Center for Scientific Computing &
California Nanosystems Institute
Eling 3231
weakliem@cnsi.ucsb.edu

Fuzzy Rogers
Research Computing Administrator
Center for Scientific Computing &
Materials Research Laboratory
MRL 2066B
fuz@mrl.ucsb.edu

Yu-Chieh “Jay” Chi, PhD
Research Computing Consultant
Center for Scientific Computing &
Enterprise Technology Services
Elings 3229
jaychi@ucsb.edu

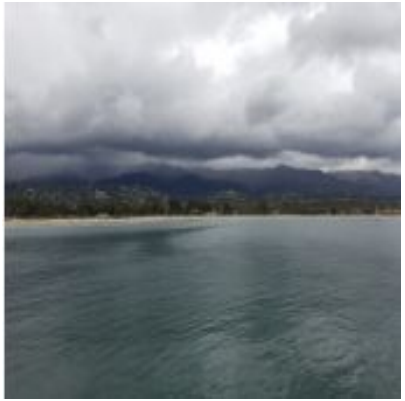
CSC Center for Scientific Computing

CNSI

MRL

ENTERPRISE TECHNOLOGY SERVICES

Our Research IT Partners



Mike Edwards
Director of Engineering Computing Infrastructure
3152A Harold Frank Hall
mcs@engineering.ucsb.edu

Michael Colee
Director of Earth Research Institute Computing (ERI)
6703 Ellison Hall
mtc@eri.ucsb.edu

Ted Cabeen
Director of Life Science Computing Group (LSCG)
2306 Life Science
ted.cabeeen@lscg.ucsb.edu

Letters & Science
INFORMATION TECHNOLOGY

Andreas Boschke
Director of Infor. Infrastructure at Letter & Science IT (LSIT)
2306 Life Science
andreas@lsit.ucsb.edu

Agenda

- What is the Center for Scientific Computing (CSC) at UCSB?
- Introduction to High-Performance Computing (HPC) at UCSB
- Goals of this workshop
  - Quickly get started to use cluster
  - Learn the basic of Linux Commands
  - Learn the basic of Slurm (Simple Linux Utility for Resource Management) commands to submit jobs to the cluster
  - File Transfer
  - Introduction to national and commercial HPC resources

What is Center for Scientific Computing (CSC)

What we are:

- A home for HPC and expertise with national supercomputing centers leveraging CNSI, MRL, and ETS resources to enable researchers to focus on the research project/education and not the infrastructure.

Support Capabilities

- We provide the computational infrastructure.
- We provide a large amount of data to store and/or process.
- We provide some expertise/assistance.
- ***We work with your local IT staff to provide help.***
- Regular working hours, realistically, 8:30 am - 5 pm Monday through Friday. But we try to make sure the clusters are running near 24/7 (I'd say 365, but it's UCSB and we're a small group)

Scenario (Distributed Computing)

Professor

Exam:
15 Questions
300 Students

Scenario

Teaching Assistants

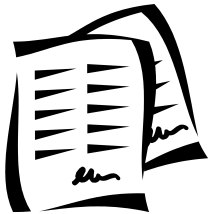
TA #1 TA #2 TA #3

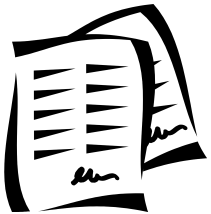
Data Parallelism

TA #1 TA #2 TA #3

100 Exams per TA

Task Parallelism

Question 1-5 

Question 6-10 

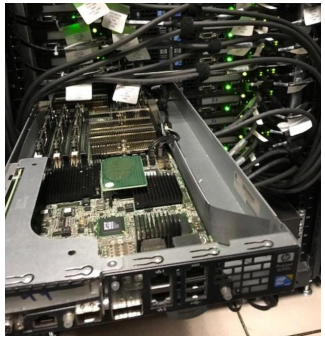
Question 10-15 

Overview

- Most research now involve some form of computing
  - Often you're solving equations, or analyzing data/doing statistics (‘data science’). Engineers often will model a device.
  - Some specific examples:
    - Protein Folding
    - Structure of crystal
    - Search for patterns in DNA
    - Predicting the spread of wildfire
    - Weather prediction
    - Natural Language Processing
- Like many parts of research, you often start a small, with a simple idea, but it grows beyond what you (or your computer) can do yourself!
- Solution:
  - Better Computer
  - High-Performance Computing (HPC)
  - Cloud (Can be both of above, with arbitrary size) - somebody else’s computer!

What is High-Performance Computing (HPC)?

- High-Performance Computing (HPC) allows scientists and engineers to solve complex science, engineering, and business problem using applications that require high bandwidth, enhanced networking, and very high compute capabilities. Ref: https://aws.amazon.com/hpc/
- Multiple computer nodes connected by a very fast interconnect.
- Each node contains many CPU cores (around 12-40 cores) and 4-6G RAM.
- Allows many users to run calculations simultaneously on nodes.
- Allows a single user to use many CPU caress incorporating multiple nodes.
- Often has high end (64 bit/high memory) GPUs

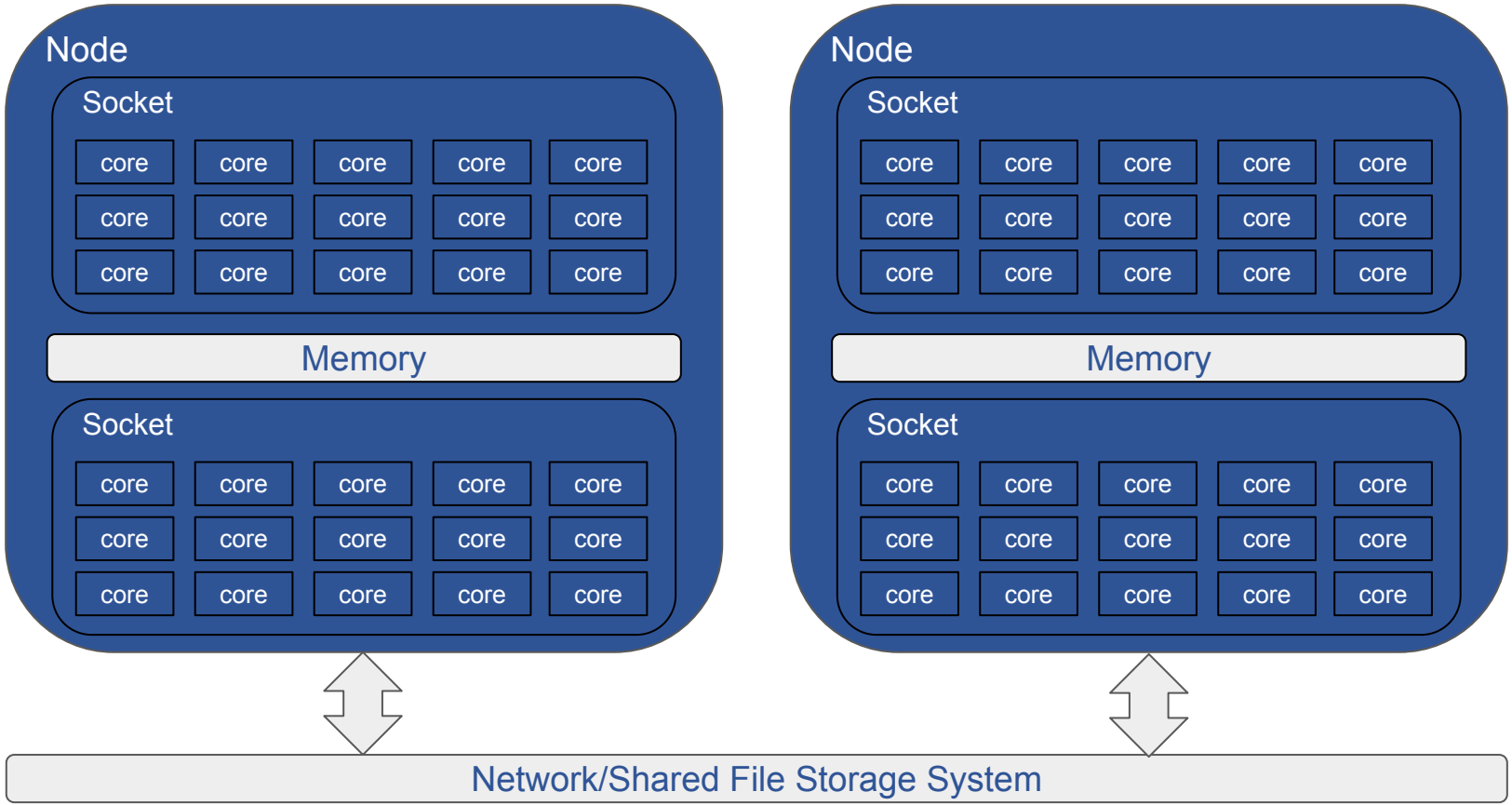


UCSB provides access and support for multiple HPC resources and educational/training/research support.

HPC is not always the only one solution!!!

- Sometimes you need a faster desktop workstation
- Sometimes 'Cloud' is the right solution (need 1000 nodes, but only once every 3 months)
- Sometimes you might even need your own cluster …….

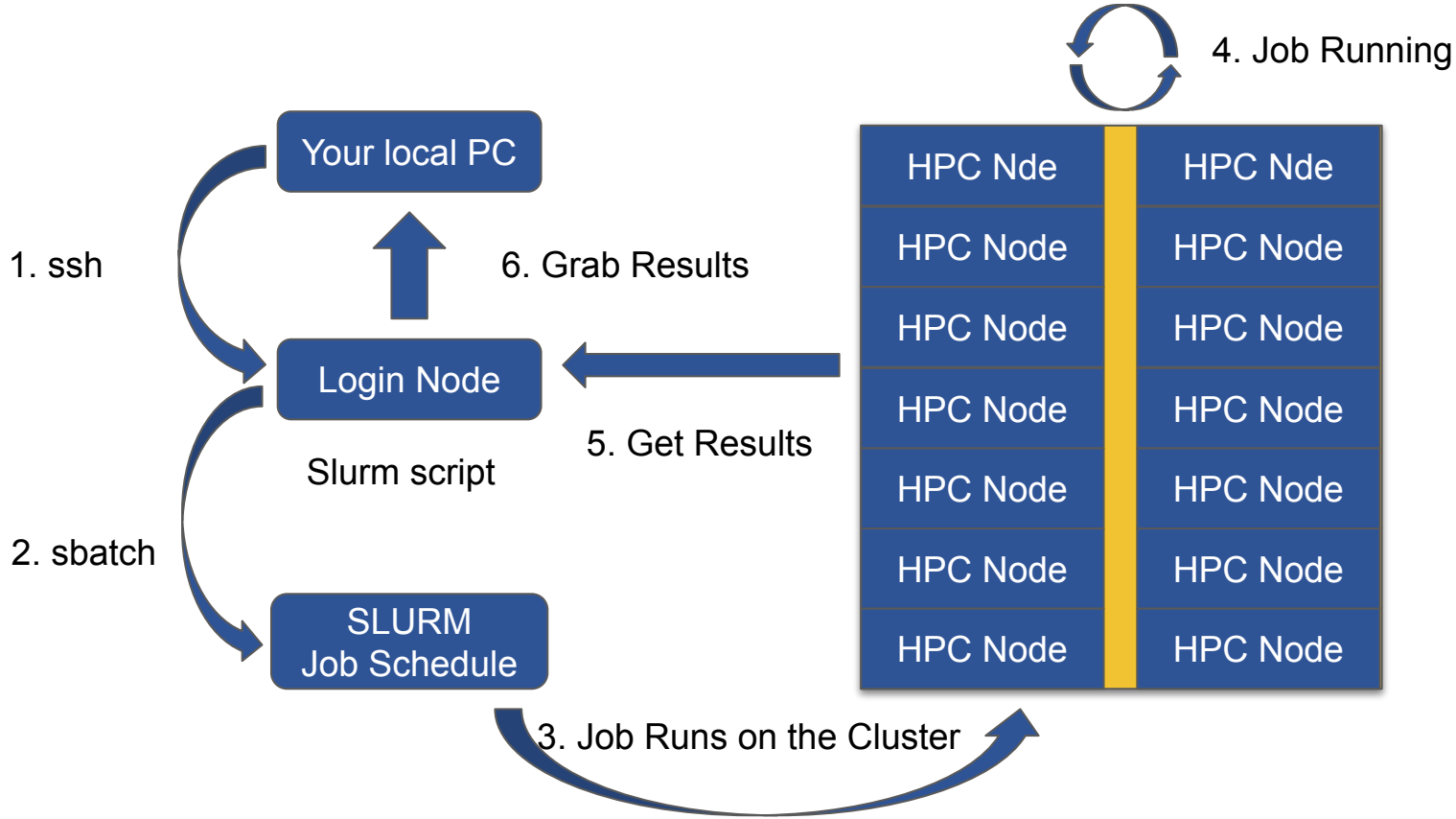
HPC Infrastructure



Terminologies Definitions

- **Core:** The smallest compute unit that can process logic and floating point (run a program).
- **CPU**: The chip that processes the basic instructions that drive a computer. The term *processor* is used interchangeably with the term **central processing unit** (**CPU**). CPUs have many cores.
- **Socket**: A physical processor which includes multiple cores with sharing memory. Most of our stuff has 2 sockets, for 2 CPUs.
- **Node**: An individual computer that includes one or more sockets, memory, storages, etc. The fast network connects other nodes.

General HPC Workflow



HPC system at CSC

- Campus available cluster Knot (CentOS/RH 7):
  - 110 nodes with ~ 1400 cores system
  - 4 “fat” nodes with 1TB memory RAM
  - GPU nodes (12 M2050’s) (too old now)
- Campus available cluster Pod (CentOS/RH7)
  - 70 nodes with ~ 2600 cores system
  - 4 “fat” nodes with 1TB memory RAM
  - 15 GPU nodes (Quad NVIDIA V100/32GB with NVLINK)
  - GPU Development node (P100, T4)
- Published papers should acknowledge CSC - https://csc.cnsi.ucsb.edu/publications

  Request access: https://csc.cnsi.ucsb.edu/forms/user-account

- Condo Clusters
  - Guild (70 nodes) EOL
  - Braid (120 nodes, also has GPUs) fairly old now
  - Braid2 (20 nodes with some GPUs)

PIs buy nodes in the clusters, CSC handles infrastructure.

What Computational Resources are available

- UCSB Center for Scientific Computing (CSC) HPC clusters
  - Access to all UCSB staff, Pod (free) and condo (PI) clusters.
- Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS)
  - National HPC resources funded by NSF. Free*
- NRP Nautilus Cluster (Consumer GPUs)
  - National cloud computing resource for accelerating machine learning on the GPUs. Free*
- Aristotle Cloud (LSIT)
  - UCSB local cloud resource, e.g. Jupyter hub
- Secure Compute Research Environment (SCRE)
- Other discipline specific UCSB resources
  - NCEAS, ERI, ECI, your local department
- Commercial Cloud Computing Resources:
  - AWS, Microsoft Azure, google Cloud Platform

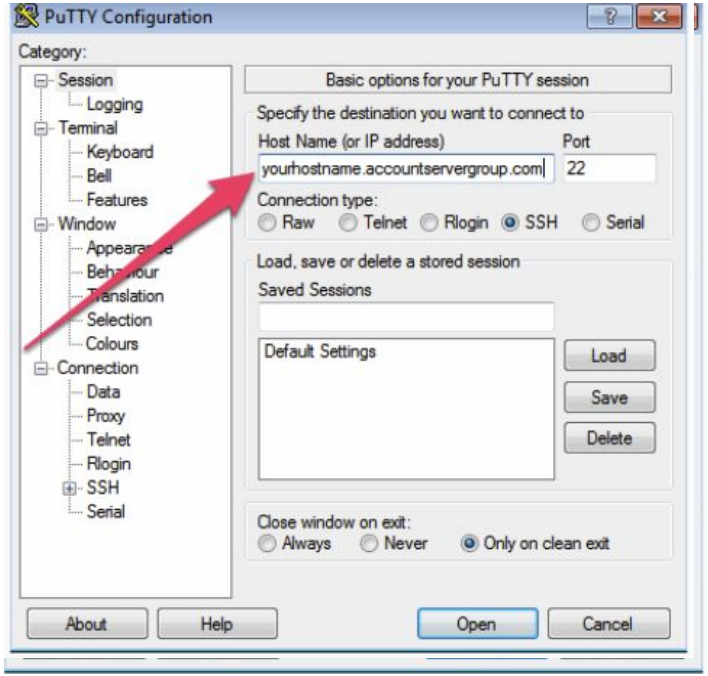
HPC Resources of Useful Information

- CSC Software Documentation
  - https://csc.cnsi.ucsb.edu/docs
- National HPC resources
  - ACCESS: https://access-ci.org/
  - San Diego Supercomputer Center: https://www.sdsc.edu/
  - NRP Nautilus: https://portal.nrp-nautilus.io/
- Transitioning from XSEDE to ACCESS by using Globus
  - https://www.globus.org/advance-to-access
- UCSB Aristotle Cloud (LSIT):
  - https://www.aristotle.ucsb.edu/ and https://help.lsit.ucsb.edu/hc/en-us/categories/360005255312-Jupyter
- UCSB Campus Cloud Information:
  - https://www.it.ucsb.edu/explore-services/ucsb-campus-cloud
  - https://docs.cloud.ucsb.edu/
- More information, go to https://csc.cnsi.ucsb.edu/resources

Connecting to the POD

- For the Windows system, you can use PuTTY ssh client
  - Download the PuTTY (https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html)

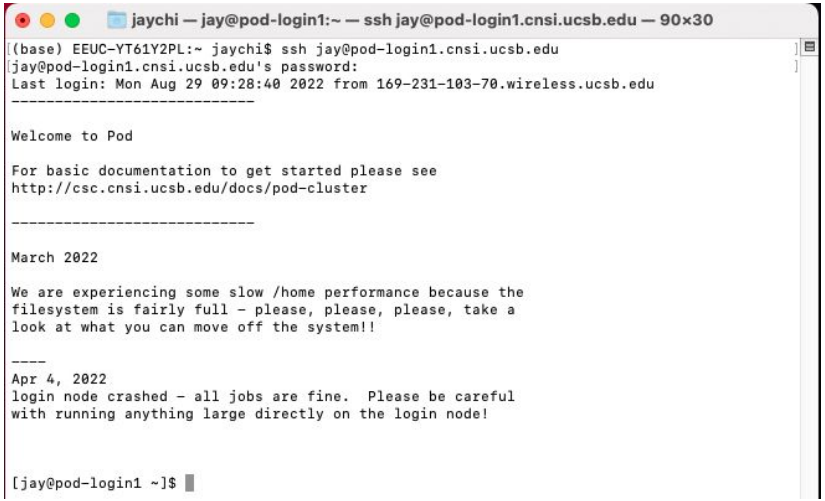
pod-login1.cnsi.ucsb.edu

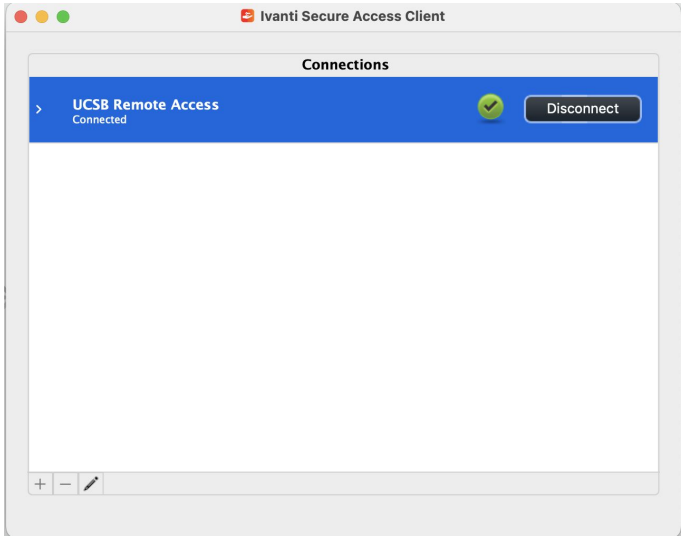


Connecting to the POD

- For the Mac or Linux system, you can open the terminal

ssh your_user_name@pod-login1.cnsi.ucsb.edu

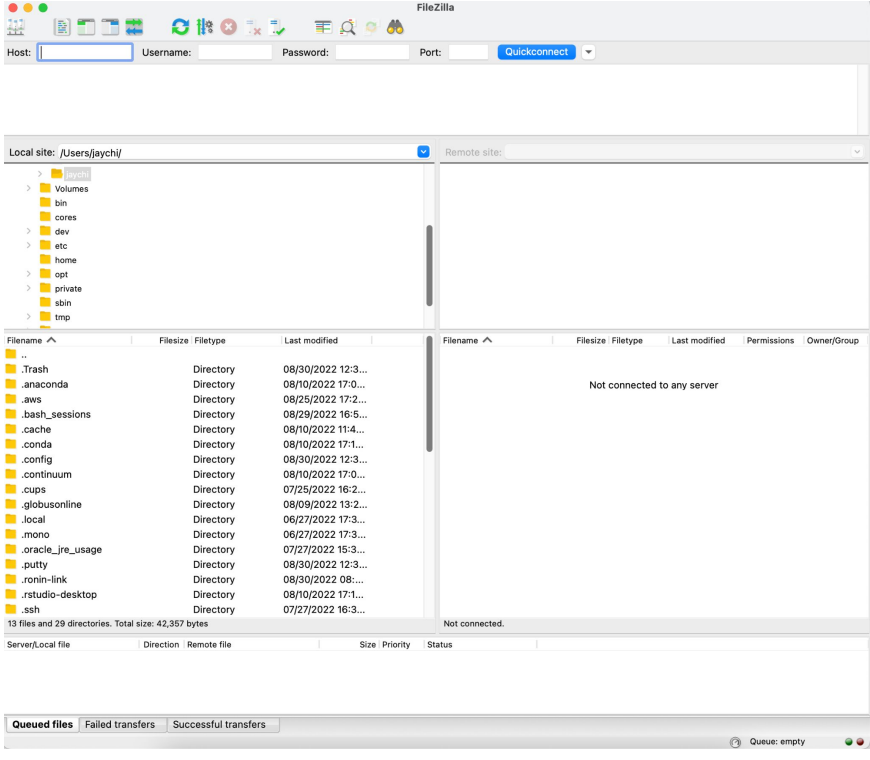




Important: Remote (non UCSB) login via VPN client:
https://www.ets.ucsb.edu/pulse-secure-campus-vpn/get-connected-campus-vpn

File Transfer

- How do I uploaded data & download my files?
  - Graphical User Interface (GUI)
    - Filezilla: https://filezilla-project.org/
    - Cyberduck: https://cyberduck.io/
  - Command-Line Interface (CLI)
    - “scp” command
- FileZilla
  - Host: pod-login1.cnsi.ucsb.edu
  - Username: your_user_name
  - Password: your_password
  - Port: 22
- Globus (for larger files transfers)
  - https://csc.cnsi.ucsb.edu/docs/globus-v5-new



Basic Linux Commands

- ls
- pwd
- cd
- mkdir
- cp
- mv
- rm
- scp
- nano, vim, or emacs to edit your file.

Basic Linux Commands (ls)

- The ls (list) command files and directories in a directory.
  - General syntax:

    ls [OPTIONS] [FILENAME]

  - OPTIONS include:

    -l long listing, includes file date and size

    -a displays all files

    -t show the newest files first

Basic Linux Commands (pwd & cd)

- pwd stands for print working directory.

  pwd

- The cd (change directory) command is used to change one directory to another.
  - General syntax:

    cd [DIRECTORY]

  - Change your present directory to the parent directory:

    $ cd ..

  - Change your present directory to the home directory:

    $ cd ~

Basic Linux Commands (mkdir & cp)

- The mkdir (make directory) command creates a new directory.
  - General syntax:

    mkdir [OPTIONS] Folder_name
- The touch command creates a new file.
  - General syntax:

    touch file_name
- The cp (copy) command is used to copy a file or directory.
  - General syntax:

    cp [OPTIONS] Source Destination
  - OPTIONS include:

    -r recursively copy a directory, all files and subdirectories inside it.

Basic Linux Commands (mv & rm)

- The mv (move) command is used to move or rename a file or directory.
  - General syntax:

    mv Source Destination

- The rm (remove) command is used to delete a file or directory.
  - General syntax:

    rm [OPTIONS] file_name

  - OPTIONS include:
    - -r recursively delete a directory, all files and subdirectories inside it.
  - ***Important***: After rm or rm -r command is executed, all files are gone and can't find in recycle bin.

File Transfer

- The scp (secure copy) command is used to transfer files between two locations.
  - General syntax:

    scp [OPTIONS] LOCAL REMOTE

    scp [OPTIONS] REMOTE LOCAL

    scp [OPTIONS] REMOTE REMOTE

  - OPTIONS include:

    -r recursively copy a directory, all files and subdirectories inside it.

More Linux Resource Information

- UCSB Software Carpentries
  - Introduction to the Unix Shell and Version Control with Git

    (https://ucsbcarpentry.github.io/2022-10-18-ucsb-bash-git/)

Modules

- Module system provides for the dynamics modification of a user's environment.
- Module commands allow the user to add applications and libraries to your environment.
- This allows us to simultaneously and safely provides several versions of the same softwares.
- All clusters have a default programming environment loaded for you when you login.
- There are some functional software are not modularized in /sw directory. Please take a look if you need.

Modules

1. List available modules
2. Search available modules for MatLab
3. Load the MatLab module
4. Unload the MatLab module
5. Purge all modules
6. List currently loaded modules

Modules

1. List available modules

   $ module avail
   …

2. Search available modules for MatLab

   $ module avail MatLab
   ---------- /sw/modulefiles ----------
   MatLab/R2016b MatLab/R2018a MatLab/R2018b MatLab/R2019a MatLab/R2019b MatLab/R2021b

3. Load the MatLab module

   $ module load MatLab/R2021b

Modules

4. Unload the MatLab module

```
$ module unload MatLab/R2021b
```

5. Purge all modules

```
$ module purge
```

6. List currently loaded modules

```
$ module list
Currently Loaded Modulefiles:
  1) autotools   2) prun/1.2     3) gnu/5.4.0   4) ohpc
```

Job Submission Script

- When you login to the Cluster, you are on the login node. This node is ***NOT*** for running calculations!
- All jobs must be submitted to the queue - it just allocate nodes.
- Submission to the queue requires a job script to be written.
- Job script need to specify the resource that you need. There are three basic units:
  - Number of Nodes
  - Number of Cores
  - Time (Optional)
- Other resource you might need to add such as: job name, memory, reminder email, etc.

Example Slurm Job Submission script

Slurm job script file: job.s

```
#!/bin/bash
#SBATCH -J 'testJob'                        ### Job Name
#SBATCH --nodes=1                           ### No. of Nodes
#SBATCH --ntasks=1                          ### No. of Tasks
#SBATCH -p short                            ### Submit the job to Partition (Optional)
#SBATCH -o outLog                           ### Output Log File (Optional)
#SBATCH -e errLog                           ### Error Log File (Optional but suggest to have it)
#SBATCH -t 00:10:00                         ### Job Execution Time
#SBATCH --mail-user=usernam@ucsb.edu        ### Mail to you (Optional)
#SBATCH --mail-type ALL                     ### Mail send you when the job starts and end (Optional)

module purge
module load openmpi/2.1.0-opt

cd $SLURM_SUBMIT_DIR/                       ### Absolute path of the current working directory when you submit the job

mpirun ./hello
```

How to Submit and Monitor Your Job

- Once you have a job script, you may submit this script to SLURM using the sbatch command. SLURM will find an available compute node or set of compute nodes and run your job there, or leave your job in a queue until some resources become available.

```
$ sbatch job.s
Submitted batch job 1234567
```

- List all current jobs from the user.

```
$ squeue -u your_user_name
$ showq your_user_name
```

- Stop and delete the Job

```
$ scancel 1234567
```

How to Submit and Monitor Your Job

- List all partitions on the cluster

  $ sinfo

```
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
batch*       up 32-00:00:0      1  down* node1
batch*       up 32-00:00:0      1   drng node20
batch*       up 32-00:00:0      1  drain node4
batch*       up 32-00:00:0     30    mix node[6,8-9,11,13-16,19,22-23,26-28,32,36-39,42-44,49,52-56,58,60]
batch*       up 32-00:00:0     29  alloc node[3,5,7,10,12,17-18,21,24-25,29-31,33-35,40-41,45-48,50-51,57,59,61-63]
batch*       up 32-00:00:0      1   idle node2
short        up    2:00:00      1    mix node64
largemem     up 37-12:00:0      4    mix node[101-104]
gpu          up 7-00:00:00      1  down* node117
gpu          up 7-00:00:00     13    mix node[111-113,115-116,118-125]
```

- List the partition who are using

  $ squeue -p short

- Report the job expected start time

  $ squeue –start -job job_ID

Running Jobs on Pod (Slurm)

- Start a job: $ sbatch job.s
- Check status of the running jobs: $ squeue -u user_name

  $ showq user_name
- Delete a running job: $ scancel job_id

- Available partition:
  - Short partition: running under 2 hrs
    - #SBATCH -p short
  - Large memory partition: running the longest 37 days
    - #SBATCH -p largemem
  - GPU partition: running the longest 7 days
    - #SBATCH -p gpu

Other Computing Options

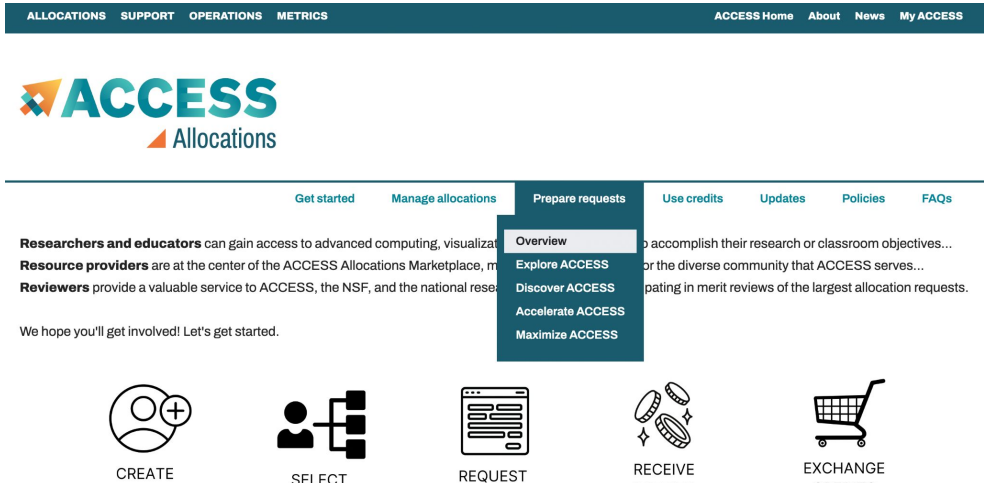
- National HPC Resource: ACCESS

- Cloud Computing: Amazon Web Services

National HPC Resources: ACCESS

Four Allocation Opportunities to suit a variety of needs (credit thresholds):

- ***Explore (400,000)***
  - Best-suited for endeavors with light resource requirements
    - Grad students can be PIs
- ***Discover (1,500,000)***
  - Minimal effort to start production research activities
    - Potential best-fit for Campus Champion Allocations
- ***Accelerate (3,000,000)***
  - More substantial resource requirements
    - Multi-grand research, Gateways, etc.
- ***Maximize (No upper limit)***
  - For large-scale research project with extreme resource needs
    - Will largely resemble XRAC process



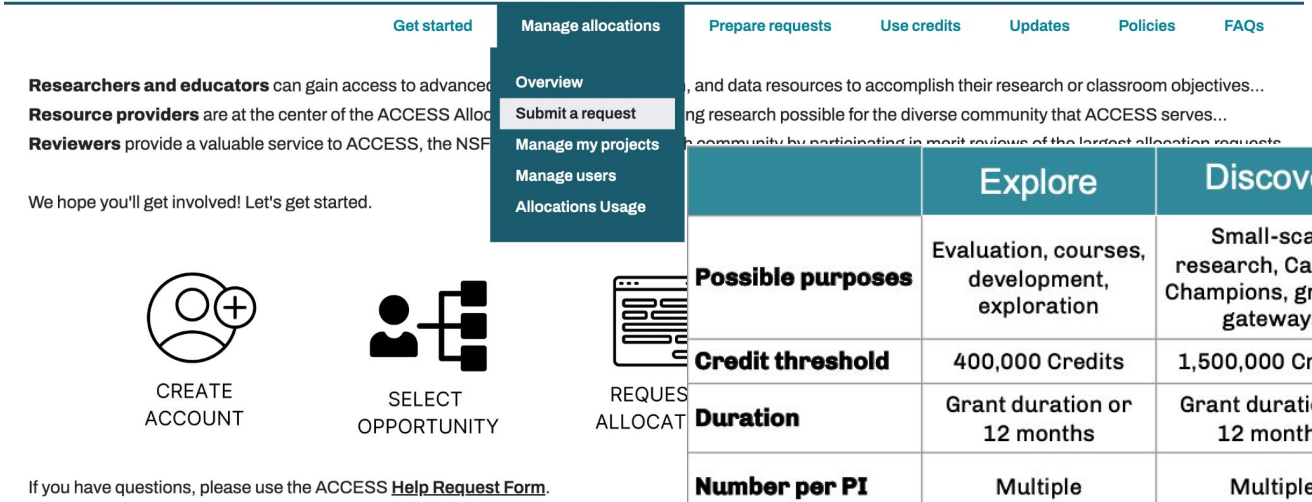
ACCESS Credits and Thresholds

Researchers have opportunities to request ACCESS allocations at four levels, which are described at the links in the table.

Allocation	Credit Threshold
Explore ACCESS	400,000
Discover ACCESS	1,500,000
Accelerate ACCESS	3,000,000
Maximize ACCESS	Not awarded in credits.

National HPC Resources: ACCESS

ACCESS Allocations



	Explore	Discover	Accelerate	Maximize
Possible purposes	Evaluation, courses, development, exploration	Small-scale research, Campus Champions, growing gateways	Mid-scale needs, consolidating related grants, collaborations	Largest-scale research activities
Credit threshold	400,000 Credits	1,500,000 Credits	3,000,000 Credits	No upper limit
Duration	Grant duration or 12 months	Grant duration or 12 months	Grant duration or 12 months	12 months
Number per PI	Multiple	Multiple	One (some exceptions)	One (limited exceptions)
Accepted	Continuously	Continuously	Continuously	Semi-annually
Proposal length	Abstract	1 page	3 pages	10 pages
Review	Eligibility	Advisory review requestable	Rolling panel review	Panel review

National HPC Resources: ACCESS

ACCESS Allocations

Purdue Anvil CPU

Purdue Anvil GPU

SDSC Expanse CPU

SDSC Expanse GPU

Resource Type:	Compute
Resource Description:	Expanse GPU will be a Dell integrated cluster, NVIDIA V100 GPUs with NVLINK, interconnected with Mellanox HDR InfiniBand in a hybrid fat-tree topology. There are a total of 52 nodes with four V100 SMX2 GPUs per node (with NVLINK connectivity). There are two 20-core Xeon 6248 CPUs per node. Full bisection bandwidth will be available at rack level (52 CPU nodes, 4 GPU nodes) with HDR100 connectivity to each node. HDR200 switches are used at the rack level and there will be 3:1 oversubscription cross-rack. In addition, Expanse also has four 2 TB large memory nodes. The system will also feature 12PB of Lustre based performance storage (140GB/s aggregate), and 7PB of Ceph based object storage.
Recommended Use:	GPUs are a specialized resource that performs well for certain classes of algorithms and applications. Recommend to be used for accelerating simulation codes optimized to take advantage of GPUs (using CUDA, OpenACC). There is a large and growing base of community codes that have been optimized for GPUs including those in molecular dynamics, and machine learning. GPU-enabled applications on Expanse will include: AMBER, Gromacs, BEAST, OpenMM, NAMD, TensorFlow, and PyTorch.
Organization:	San Diego Supercomputer Center
Units:	GPU Hours
Description:	

SDSC Expanse Projects Storage

allocations | Prepare requests | Use credits | Updates | Policies | FAQs

- Overview
- Available resources
- Exchange calculator

g, visualization, and data res… r research or classroom objectives…

ketplace, making research p… mmunity that ACCESS serves…

tional research community … views of the largest allocation requests.

REQUEST ALLOCATION

Exchange Calculator

Number of units on this resource:

10,000	ACCESS Credits

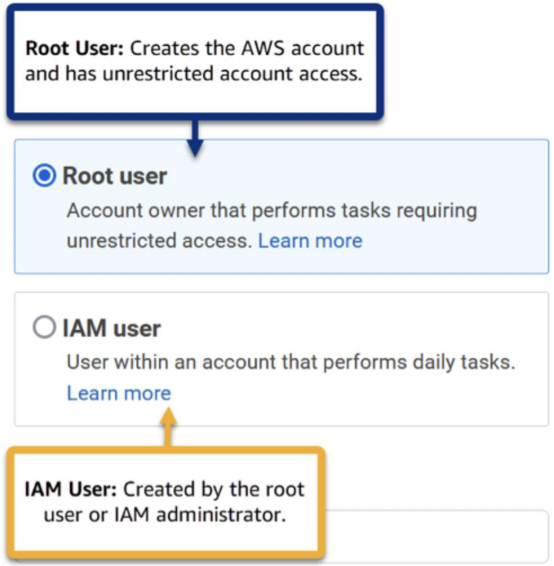
Equals this many units on this resource:

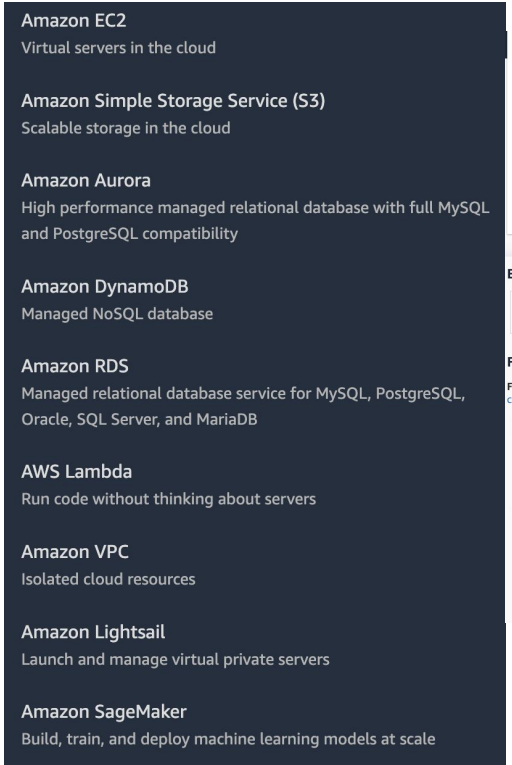
186	SDSC Expanse GPU

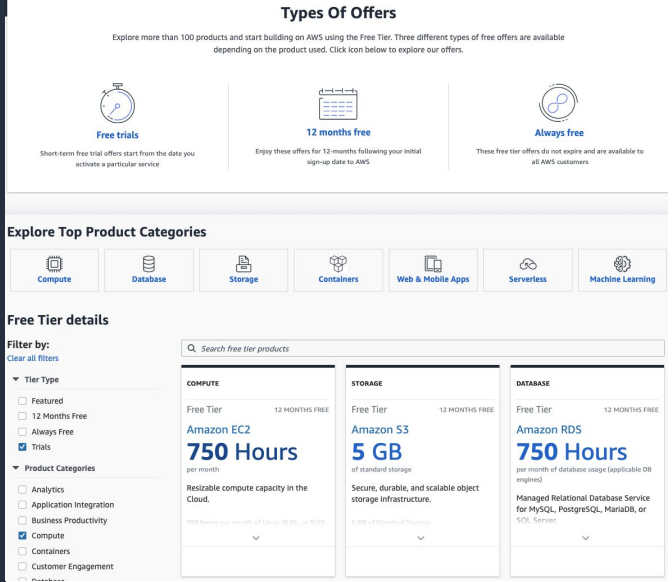
RESET

Cloud Computing: Amazon Web Services (AWS) (will delete)

- Sign in to AWS Management Console





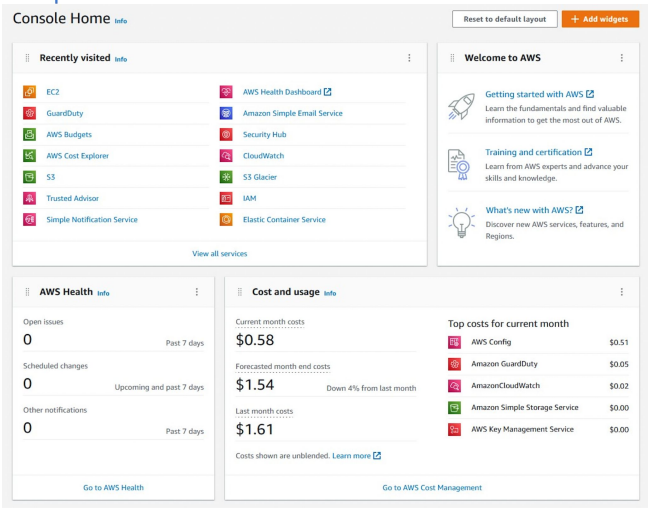


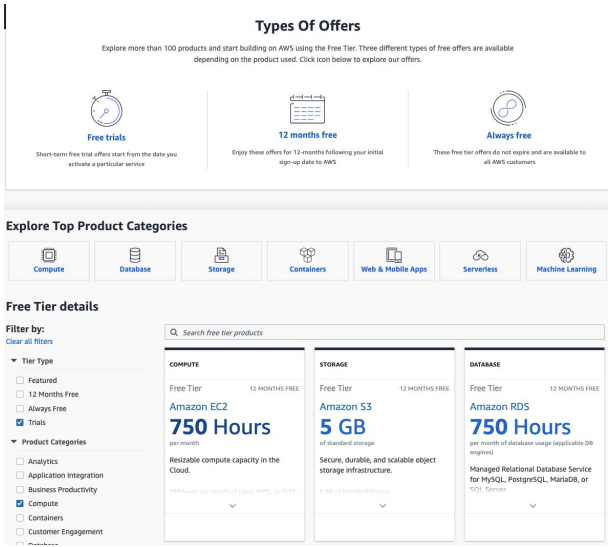
Cloud Computing: Amazon Web Services (AWS)

- If you choose to use AWS, it is recommended to take advantage of the Campus Cloud Landing Zone (LZ) for AWS. A UCSB purchases order is required to request an Campus Cloud account (https://ucsb.github.io/campus-cloud-docs/getting-started/#procurement).

Campus Single Sign On for AWS: https://aws.cloud.ucsb.edu

- Supported Campus Cloud Regions:
  - **US-West-2 (Oregon)**
  - **US-East-1 (N.Virginia)**
- We recommend starting in *US-West-2*

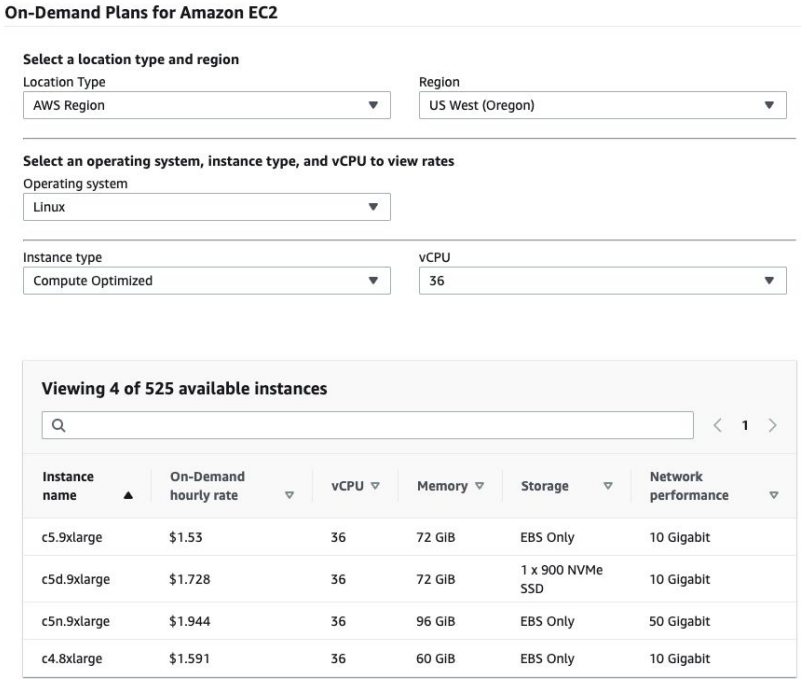




Important: You may need the help of a PI or Department Purchasing person to create a Purchase Order which is necessary to request an account in the Campus Cloud.

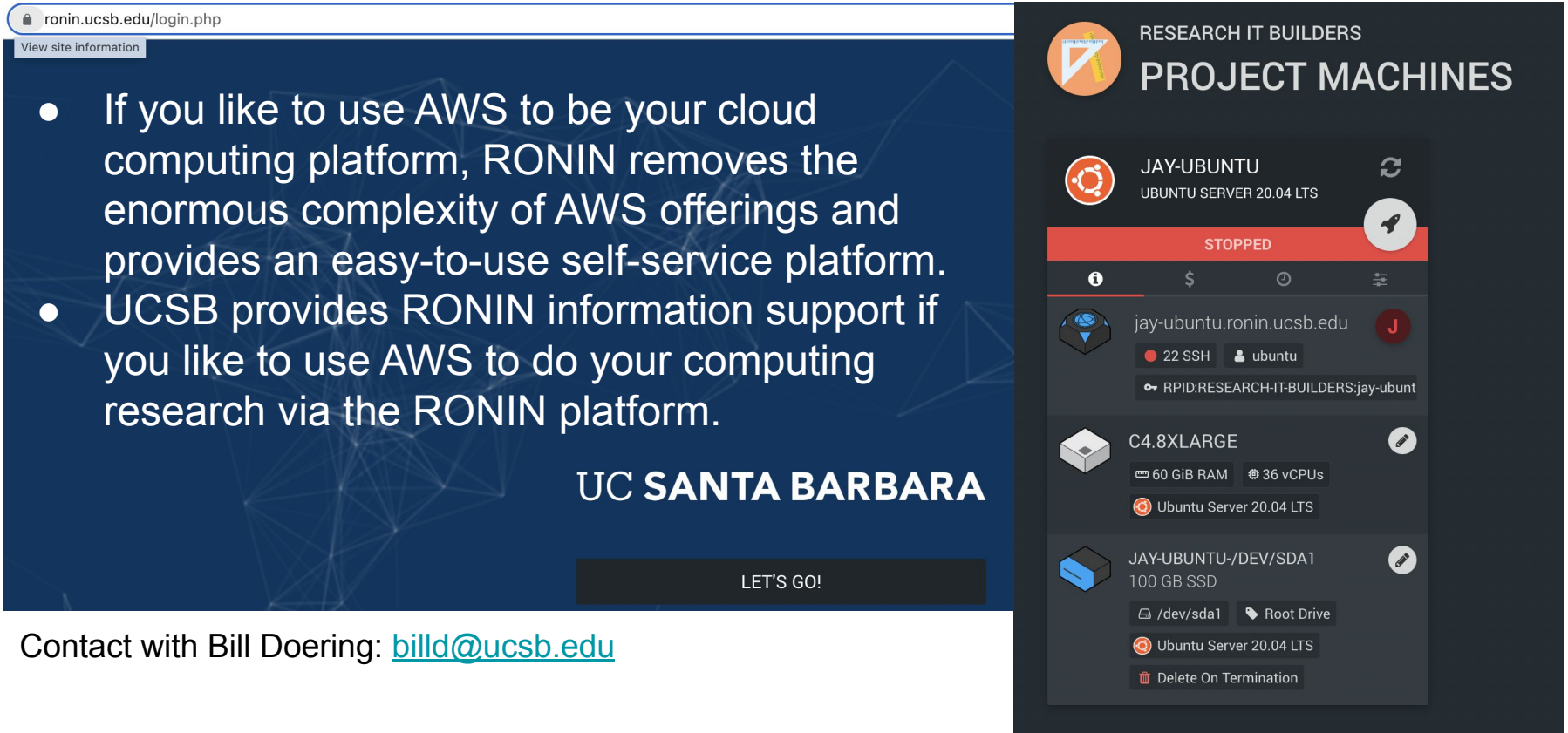
Amazon Elastic Compute Cloud (Amazon EC2)

- Use Case:
  - Run cloud-native and enterprise applications
  - Scale for HPC applications
  - Train and deploy ML applications
- EC2 Instance Types
  - General Purpose
  - Compute Optimized
  - Memory Optimization
  - Accelerated Computing
  - Storage Optimized
- More Information
  - Amazon EC2: https://aws.amazon.com/ec2/
  - Amazon EC2 Pricing Estimation: https://aws.amazon.com/ec2/pricing/on-demand/



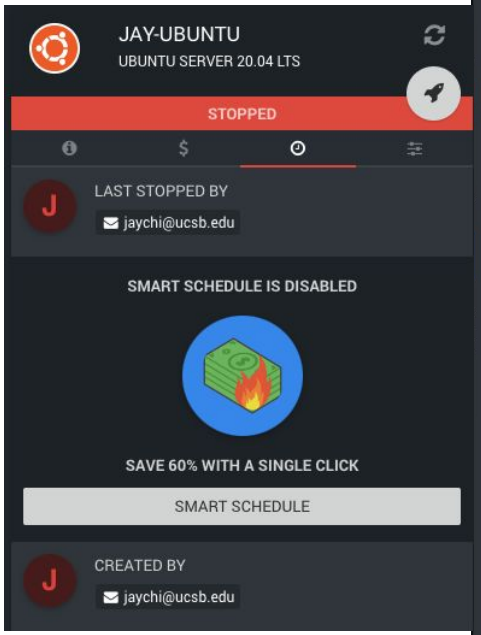
https://instances.vantage.sh/

Ronin Platform RONIN



Contact with Bill Doering: billd@ucsb.edu

Ronin Platform: Control Your AWS Cost



Questions and Thought

- What else content should we cover?
- Other ideas for a workshop?
  - Running Parallel Python / Matlab / R on the Cluster, Mathematica, Lumerical, Singularity/Docker Container, etc.

- More Information:

  https://csc.cnsi.ucsb.edu/