SDSC HPC/CI User Training 2022

HPC/CI Basic Concepts: Accessing Expanse

EXPANSE

COMPUTING WITHOUT BOUNDARIES

Presented by: Mary Thomas

01-27-2022

SAN DIEGO SUPERCOMPUTER CENTER

NSF Award 1928224

Outline

- Expanse overview
- Overview of Basic Skills Needed to Use HPC/CI Systems
- Expanse overview
- Logging on to Expanse
- Modules
- Account Management

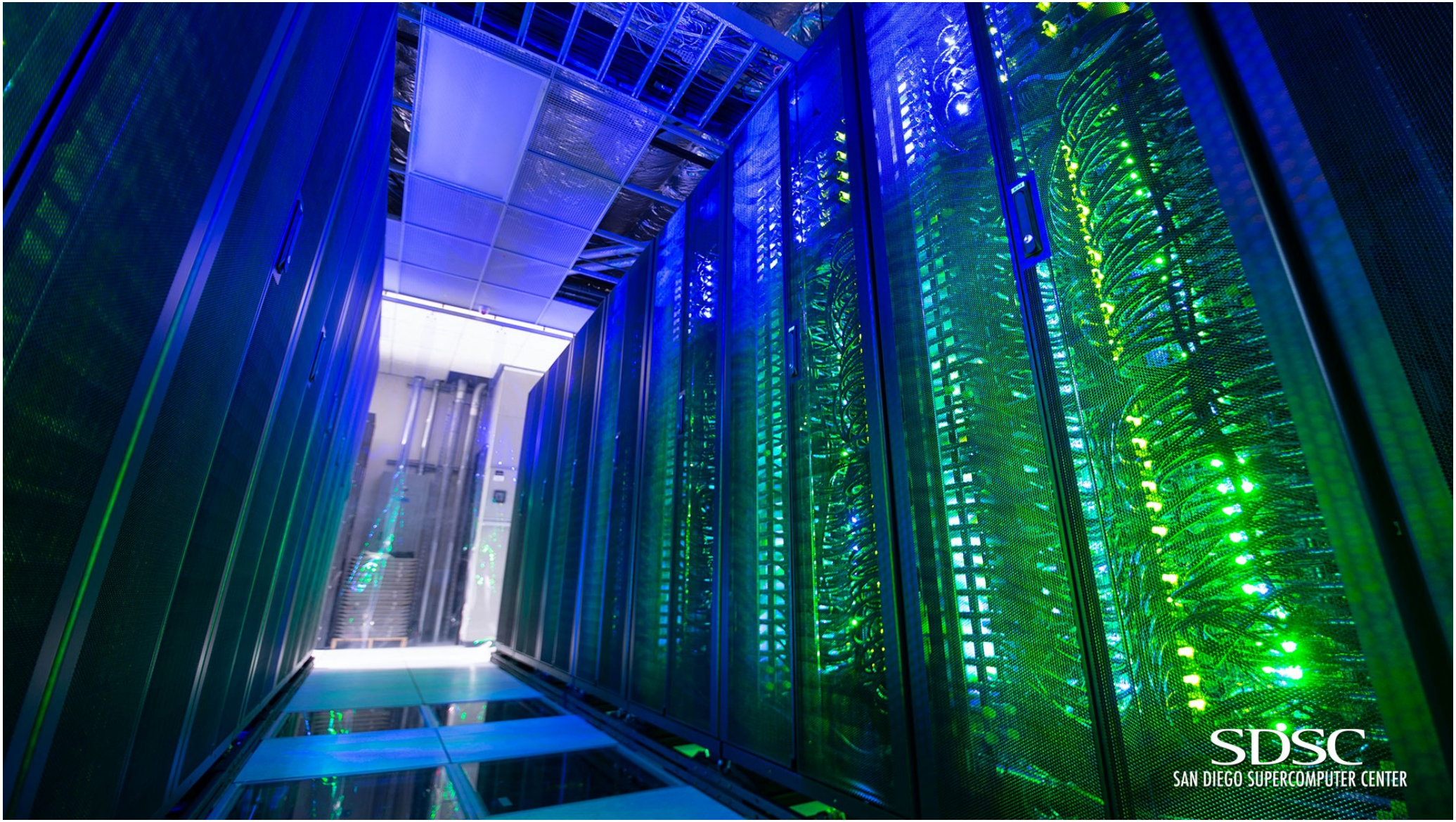
Useful Links

- Expanse User Guide:
  - https://www.sdsc.edu/support/user_guides/expanse.html
- Expanse Overview:
  - https://education.sdsc.edu/training/interactive/202201_parallel_computing_concepts/index.html
- You need to have an Expanse account in order to access the system. There are a few ways to do this:
  - Submit a proposal through the XSEDE Allocation Request System
  - A PI on an active allocation can add you to their allocation (if you are collaborators working on the same project).
  - Request a trial account: instructions @ https://portal.xsede.org/allocations/startup.
- Online repo and information:
  - https://github.com/sdsc-hpc-training-org/expanse-101
  - https://hpc-training.sdsc.edu/expanse-101/

Outline

- Expanse overview
- Overview of Basic Skills Needed to Use HPC/CI Systems
- Expanse overview
- Logging on to Expanse
- Modules
- Account Management

Expanse

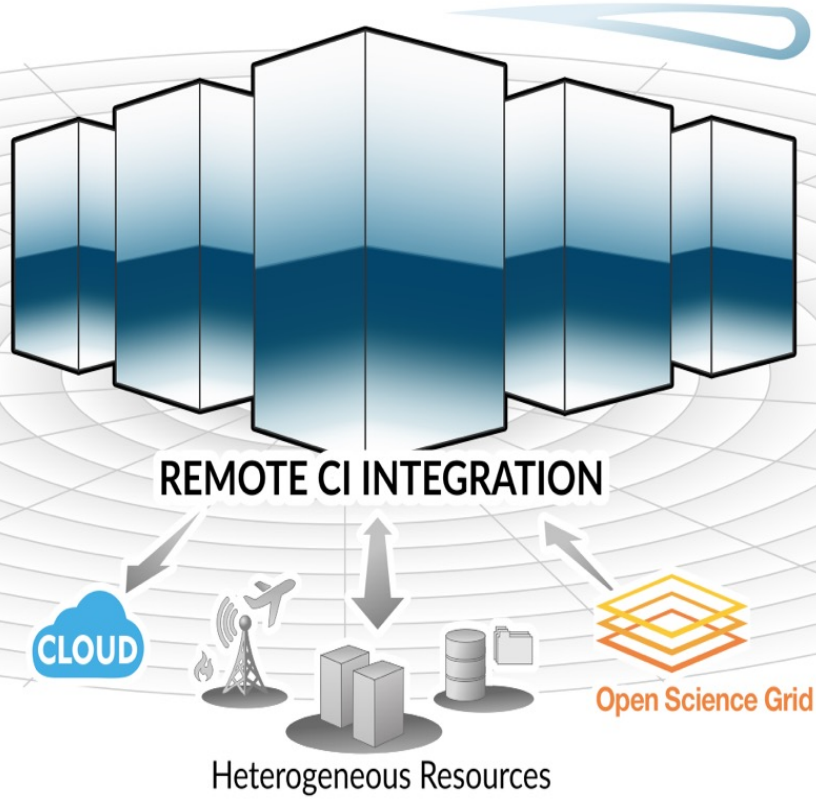


EXPANSE

COMPUTING WITHOUT BOUNDARIES
5 PETAFLOP/S HPC and DATA RESOURCE

HPC RESOURCE
13 Scalable Compute Units
728 Standard Compute Nodes
52 GPU Nodes: 208 GPUs
4 Large Memory Nodes

DATA CENTRIC ARCHITECTURE
12PB Perf. Storage: 140GB/s, 200k IOPS
Fast I/O Node-Local NVMe Storage
7PB Ceph Object Storage
High-Performance R&E Networking



LONG-TAIL SCIENCE
Multi-Messenger Astronomy
Genomics
Earth Science
Social Science

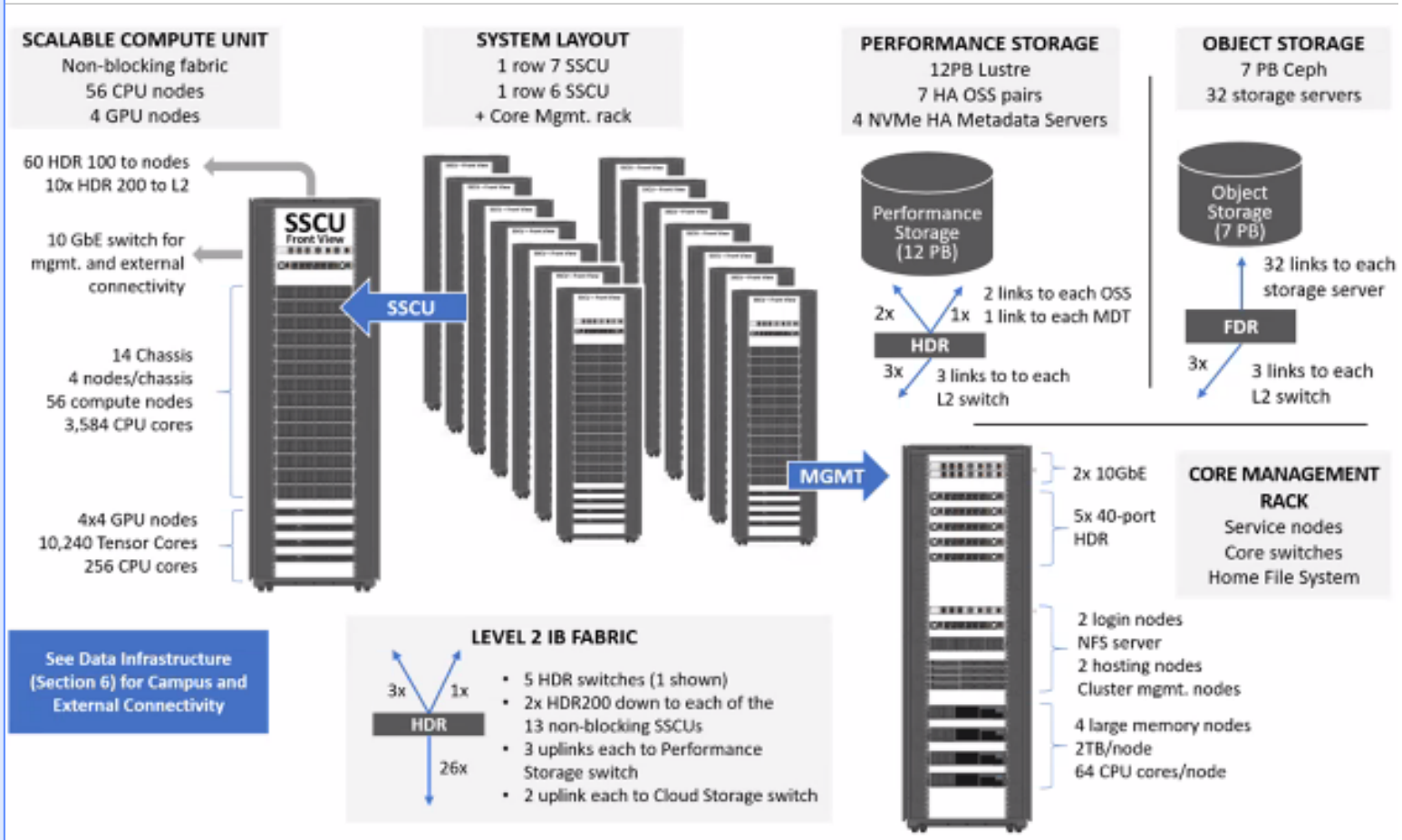
INNOVATIVE OPERATIONS
Composable Systems
High-Throughput Computing
Science Gateways
Interactive Computing
Containerized Computing
Cloud Bursting

For more details see the Expanse user guide @ https://www.sdsc.edu/support/user_guides/expanse.html
and the "Introduction to Expanse" webinar @ https://www.sdsc.edu/event_items/202006_Introduction_to_Expanse.html

Expanse Heterogeneous Architecture

System Summary

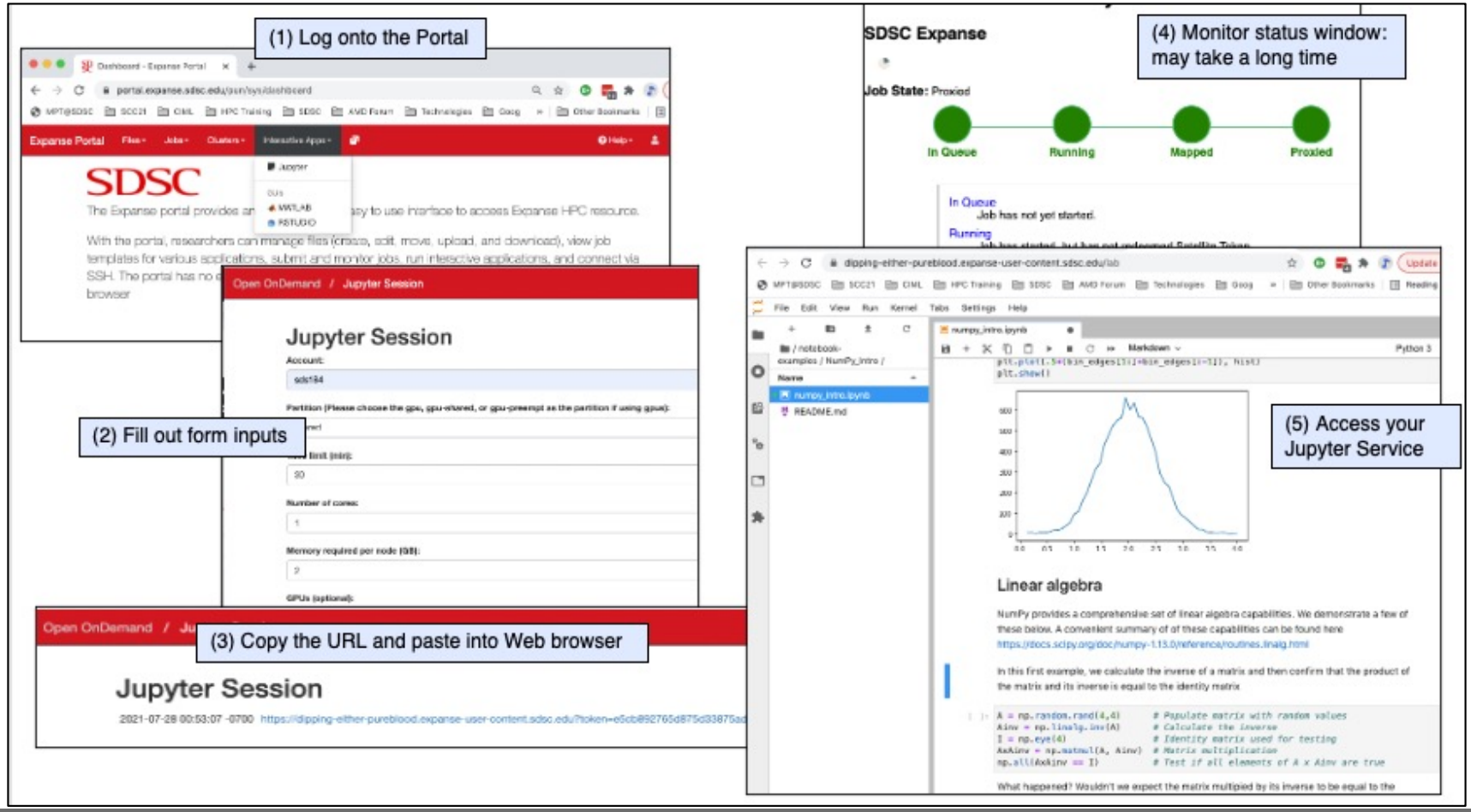
- *13 SDSC Scalable Compute Units (SSCU)*
- *728 x 2s Standard Compute Nodes*
- *93,184 Compute Cores*
- *200 TB DDR4 Memory*
- *52x 4-way GPU Nodes w/NVLINK*
- *208 V100s*
- *4x 2TB Large Memory Nodes*
- *HDR 100 non-blocking Fabric*
- *12 PB Lustre High Performance*
- *Storage*
- *7 PB Ceph Object Storage*
- *1.2 PB on-node NVMe*
- *Dell EMC PowerEdge*
- *Direct Liquid Cooled*



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



Outline

- Expanse overview
- Overview of Basic Skills Needed to Use HPC/CI Systems
- Logging on to Expanse
- Modules
- Account Management

Basic Skills Needed to Use HPC/CI Systems

https://github.com/sdsc-hpc-training-org/basic_skills

- Overview of the HPC system you want to work on: e.g. The Expanse cluster:
  - https://drive.google.com/file/d/1i1zwOFOz438y0SWcjkDp2zd6FJXjVyY_/view?usp=sharing
- Security and Authentication
- Using Github
- Unix/Linux
- Understand HPC system environment (accounts, modules, etc)

Outline

- Expanse overview
- Overview of Basic Skills Needed to Use HPC/CI Systems
- Logging on to Expanse
- Modules
- Account Management

Logging onto Expanse

- Expanse supports Single Sign-On through the XSEDE User Portal
- From the command line using an XSEDE-wide password,
- From the Expanse User Portal (https://portal.expanse.sdsc.edu).
- Note that CPU and GPU resources are allocated separately, but the login nodes are the same.
- To log in to Expanse from the command line
  - hostname: login.expanse.sdsc.edu
  - Secure shell (SSH) command examples:

```
ssh <user_name>@login.expanse.sdsc.edu
ssh -l < user_name > login.expanse.sdsc.edu
```

- When you log onto *login.expanse.sdsc.edu*, you will be assigned one of the two login nodes login0[1-2]-expanse.sdsc.edu. Both systems are identical.

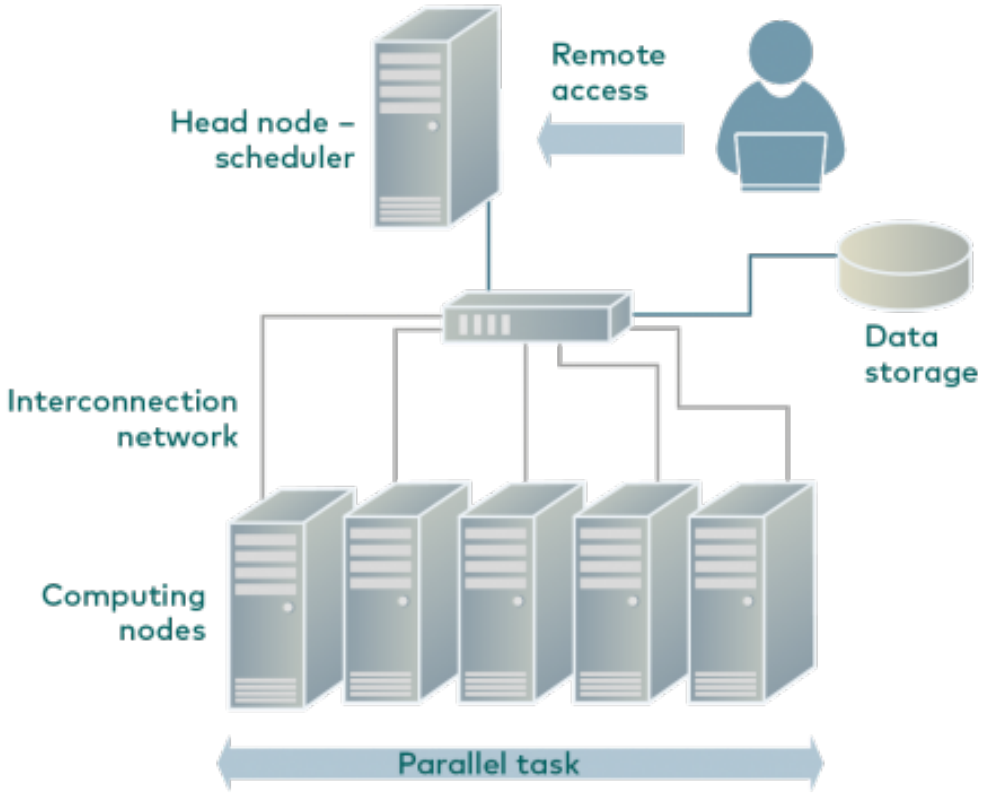
Using SSH Keys

- You can append your public key (e.g.from your laptop) to your ~/.ssh/authorized_keys file to enable access from authorized hosts without having to enter your password.
- RSA, ECDSA and ed25519 keys are accepted.
- Make sure you have a strong passphrase on the private key on your local machine.
- You can use *ssh-agent* forwarding to avoid repeatedly typing the private key password:
  - https://github.com/sdsc-hpc-training-org/hpc-security/blob/master/ssh_methods/connect-using-ssh-agent.md
- Hosts which try to connect to SSH more frequently than ten times per minute may get blocked for a short period of time
- See the SDSC webinar: Indispensable Security: Tips to Use SDSC's HPC Resources Securely:
  - 2021 Webinar: https://www.sdsc.edu/event_items/202007_CometWebinar.html
  - 2022 Webinar scheduled for April, 2022. Watch the training pages:
    - https://www.sdsc.edu/education_and_training/training_hpc.html

System Access: Clients

- Linux/Mac –
  - use terminal + installed ssh app
- Windows:
  - Win10 terminal app + installed ssh app
  - Older Windows OS's: ssh clients apps Putty, Cygwin

- Login hostname for SDSC Expanse:
  - login.expanse.sdsc.edu
  - 198.202.113.252



Source: https://hpc.rtu.lv/hpc/introduction-to-hpc/?lang=en

For more on SDSC security, see: https://github.com/sdsc-hpc-training-org/hpc-security

Example of a terminal connection:

```
Warning: No xauth data; using fake authentication data for X11 forwarding.
Welcome to Bright release         9.0

                                                Based on CentOS Linux 8
                                                        ID: #000002
--------------------------------------------------------------------------------

                              WELCOME TO

                 _______  __ ____  ___    _  ________ _______
                / ____/ |/ // __ \/   |  / |/ / ____// ____/
               / __/  |   // /_/ / /| | /  |/ /\__ \/ __/
              / /___ /   |/ ____/ ___ |/ /|  /___/ / /___
             /_____//_/|_/_/   /_/  |_/_/ |_//____/_____/

--------------------------------------------------------------------------------

Use the following commands to adjust your environment:

'module avail'            - show available modules
'module add <module>'     - adds a module to your environment for this session
'module initadd <module>' - configure module to be loaded at every login


--------------------------------------------------------------------------------
Last login: Fri Jan 28 12:00:38 2022 from 12.345.678.90
[username@login02 ~]$
```

The logon message you see is called the MOTD (message of the day, located in /etc/motd).

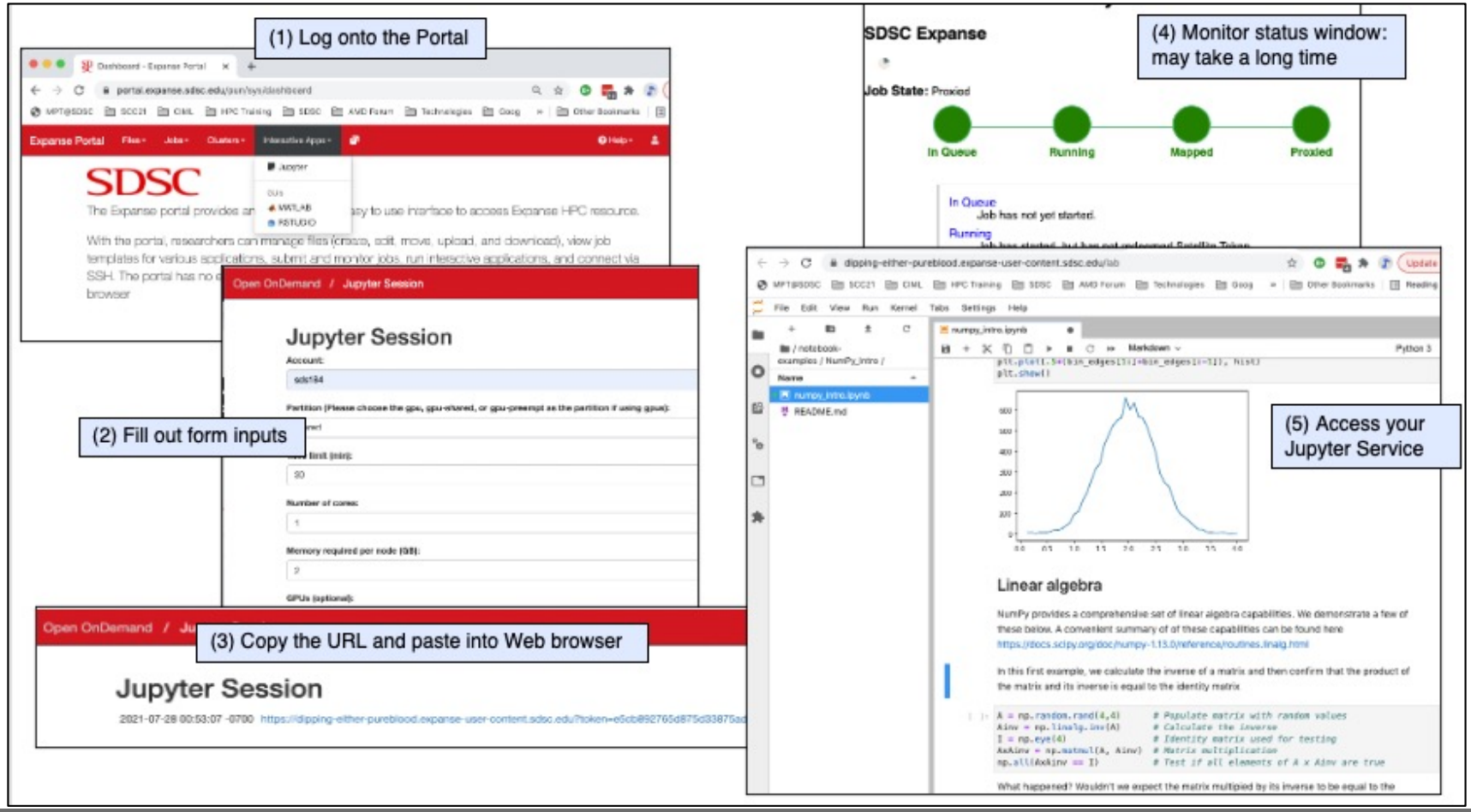
Using Login Nodes Properly

- The login nodes are meant for file editing, simple data analysis, & tasks that use minimal compute resources.
- All computationally demanding jobs should be submitted and run through the batch queuing system.
- Do not use the login nodes for:
  - computationally intensive processes,
  - as hosts for running workflow management tools
  - as primary data transfer nodes for large or numerous data transfers
  - as servers providing other services accessible to the Internet.
  - running Jupyter notebooks
- Login nodes are not the same as the batch nodes.
  - Users should request an interactive sessions to compile ;arge programs.

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



Outline

- Expanse overview
- Overview of Basic Skills Needed to Use HPC/CI Systems
- Logging on to Expanse
- Modules
- Account Management

Expanse Environment Modules

- Expanse uses *Lmod*, a *Lua* based module system.
  - https://lmod.readthedocs.io/en/latest/010_user.html
- Users setup custom environments by loading available modules into the shell environment, *including needed compilers and libraries* and the batch scheduler.
- What modules let you do:
  - Dynamic modification of your shell environment
  - User can set, change, or delete environment variables
  - User chooses between different versions of the same software or different combinations of related codes.

Modules on Expanse

- Users will need to load the scheduler (e.g. slurm)
- Users will *not* see all available modules when they run command "module available" *without loading a compiler*.
- Use the command "module spider" option to see if a particular package exists and can be loaded, run command
  - *module spider <package>*
  - *module keywords <term>*
- For additional details, and to identify module dependencies modules, use the command: *module spider <application_name>*
- The module paths are different for the CPU and GPU nodes. Users can enable the paths by loading the following modules:
  - `module load cpu  (for cpu nodes)`
  - `module load gpu  (for gpu nodes)`
  - `avoid loading both modules`

Modules: Popular commands

Command	Description
module list	List the modules that are currently loaded
module avail	List the modules that are available in environment
module spider	List of the modules and extensions currently available
module display <module_name>	Show the environment variables used by <module name> and how they are affected
module unload <module name>	Remove <module name> from the environment
module load <module name>	Load <module name> into the environment
module swap <module one> <module two>	Replace <module one> with <module two> in the environment
module help	get a list of all the commands that module knows about do:
Shorthand notation: ml foo ml -bar	"ml" == module load foo "ml -bar" == module unload bar

SDSC Guidance: add module calls to your environment and batch scripts

Module Command Examples

```
[username @login02 ~]$ module reset
Resetting modules to system default. Resetting $MODULEPATH back to system default. All extra directories will
be removed from $MODULEPATH.
[username@login02 ~]$ module list
Currently Loaded Modules:
  1) cpu/0.15.4   2) slurm/expanse/20.02.3   3) intel/19.1.1.217

username@login02 ~]$ module avail
---------------------- /cm/shared/apps/spack/cpu/lmod/linux-centos8-x86_64/intel/19.1.1.217 ----------------------
   bamtools/2.5.1      grace/5.1.25                libpng/1.6.37                openmpi/4.0.4 (D)
   bedtools2/2.27.1    gsl/2.5                     libtirpc/1.2.6               papi/6.0.0.1
     [SNIP]
   eigen/3.3.7         jasper/2.0.16               openmpi/3.1.6

---------------------------- /cm/shared/apps/spack/cpu/lmod/linux-centos8-x86_64/Core -----------------------------
   abaqus/2018                      emboss/6.6.0             gmp/6.1.2               parallel/20200822
   anaconda3/2020.11                freesurfer/7.1.1         go/1.15.1               pciutils/3.7.0
  [SNIP]
   cmake/3.18.2                     gcc/9.2.0                mpfr/4.0.2              zstd/1.4.5
   curl/7.72.0                      gcc/10.2.0        (D)    nbo/7.0-openblas
   doxygen/1.8.17                   gh/1.13.1                openjdk/11.0.2

------------------------------------------- /cm/local/modulefiles --------------------------------------------
   boost/1.71.0    cmjob    lua/5.3.5    shared    singularitypro/3.5    slurm/expanse/20.02.3 (L)
------------------------------------- /cm/shared/apps/xsede/modulefiles --------------------------------------
   cue-login-env    xdinfo/1.5-1    xdusage/2.1-1
----------------------------------------- /usr/share/modulefiles -----------------------------------------
   DefaultModules    cpu/0.15.4 (L)    gct/6.2    globus/6.0    gpu/0.15.4    nostack/0.15.4
------------------------------------------- /cm/shared/modulefiles -------------------------------------------
   AMDuProf/3.4.475    default-environment    sdsc/1.0

  Where:
   L:  Module is loaded
   D:  Default Module
```

List Current environment

Show available modules

Module Command Examples

Use “module show” to find out what a particular module will change in the environment

```
[username@login02 ~]$ module show cmake
-------------------------------------------------------------------------------------------------------------
   /cm/shared/apps/spack/cpu/lmod/linux-centos8-x86_64/Core/cmake/3.18.2.lua:
-------------------------------------------------------------------------------------------------------------
whatis("Name : cmake")
whatis("Version : 3.18.2")
whatis("Target : zen")
whatis("Short description : A cross-platform, open-source build system. CMake is a family of tools designed to
build, test and package software. ")
help([[A cross-platform, open-source build system. CMake is a family of tools
designed to build, test and package software.]])
prepend_path("PATH","/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen/gcc-8.3.1/cmake-3.18.2-
rfzatdti4qlsrf2zezwad75fnccy4f7d/bin")
prepend_path("ACLOCAL_PATH","/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen/gcc-8.3.1/cmake-3.18.2-
rfzatdti4qlsrf2zezwad75fnccy4f7d/share/aclocal")
prepend_path("CMAKE_PREFIX_PATH","/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen/gcc-8.3.1/cmake-3.18.2-
rfzatdti4qlsrf2zezwad75fnccy4f7d/")
setenv("CMAKEHOME","/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen/gcc-8.3.1/cmake-3.18.2-
rfzatdti4qlsrf2zezwad75fnccy4f7d")


[username@login02 ~]$
```

Module: check Environment

Once you have loaded the modules, you can check the system variables that are available for you to use.

```
[username@login01 ~]$ module load  gcc/9.2.0
[username@login01 ~]$ module show  gcc/9.2.0
-------------------------------------------------------------------
   /cm/local/modulefiles/gcc/9.2.0:
-------------------------------------------------------------------
whatis("adds GNU Cross Compilers to your environment variables ")
prepend_path("PATH","/cm/local/apps/gcc/9.2.0/bin")
prepend_path("LD_LIBRARY_PATH","/cm/local/apps/gcc/9.2.0/lib:/cm/local/apps/gcc/9.2.0/lib6
4")
help([[ Adds GNU Cross Compilers to your environment variables,
]])


[username@login01 ~]$ which gcc
/cm/local/apps/gcc/9.2.0/bin/gcc
[username@login01 ~]$
```

Module: command not found

- Sometimes happens when switching from one shell to another or attempting to run the module command from within a shell script or batch job.
- Module command may not be inherited to the shell
- To keep this from happening, execute the following:
  - From the command line (interactive shells)
    - source /etc/profile.d/modules.sh
  - OR add to your shell script (including Slurm batch scripts)

Outline

- Expanse overview
- Overview of Basic Skills Needed to Use HPC/CI Systems
- Logging on to Expanse
- Modules
- Account Management

Multiple Allocations

- Many users will have access to multiple accounts and hence *projects*:
  - an allocation for a research project and a separate allocation for classroom or educational use
- Users should verify that the correct *project* is designated for all batch jobs.
- Awards are granted for a specific purposes and should not be used for other *projects*.
- To charge your job to one of your *projects*, replace << project >> with one from your list and put this PBS directive in your job script:
  - #SBATCH -A << project >>
  - *srun –account*=<< *project* >>

Allocation Information

module load sdsc
expanse-client user
expanse-client user -r expanse_gpu

```
[username@login02 ~]$ module load sdsc
[username@login02 ~]$ expanse-client user

 Resource  expanse

┌─────┬──────────┬─────────┬──────────────┬──────┬───────────┬─────────────────┐
│     │ NAME     │ PROJECT │ TG PROJECT   │ USED │ AVAILABLE │ USED BY PROJECT │
├─────┼──────────┼─────────┼──────────────┼──────┼───────────┼─────────────────┤
│ 1   │ username │ ddp363  │              │   65 │     50000 │            2223 │
│ 2   │ username │ sds173  │ TG-CCR190013 │    6 │      1000 │            4750 │
│ 3   │ username │ sds184  │ TG-TRA210003 │  121 │    100000 │           56336 │
│ 4   │ username │ abc123  │              │ 1596 │   5050000 │         3267835 │
└─────┴──────────┴─────────┴──────────────┴──────┴───────────┴─────────────────┘
[username@login02 ~]$ expanse-client user -r expanse_gpu

 Resource  expanse_gpu
┌─────┬──────────┬─────────┬──────────────┬──────┬───────────┬─────────────────┐
│     │ NAME     │ PROJECT │ TG PROJECT   │ USED │ AVAILABLE │ USED BY PROJECT │
├─────┼──────────┼─────────┼──────────────┼──────┼───────────┼─────────────────┤
│ 1   │ username │ ddp363  │              │    0 │      2500 │              58 │
│ 2   │ username │ sds173  │ TG-CCR190013 │    0 │       100 │              47 │
│ 3   │ username │ sds184  │ TG-TRA210003 │    0 │      5000 │            1172 │
│ 4   │ username │ abc123  │              │   42 │    269000 │           62058 │
└─────┴──────────┴─────────┴──────────────┴──────┴───────────┴─────────────────┘
```

Charging

- Charge unit for all SDSC machines, including Expanse, is the Service Unit (SU).
  - 1 CPU core using <= 2G of data for 1 hour
  - 1 GPU using < 96G of data for 1 hour
  - 'shared' partitions: based on either # of cores or fraction of memory requested, whichever is larger
- Charges based on resources used by job regardless of use.
- Charges are based on either # of cores or fraction of the memory requested, whichever is larger.
- Minimum charge for any job is 1 SU:
  - *Can quickly use up SUs if you run a lot of very short jobs.*
- More details in Expanse user guide:

  https://www.sdsc.edu/support/user_guides/expanse.html#charging

Thank You

Resources

- Expanse User Guide & Tutorial
  - https://www.sdsc.edu/support/user_guides/expanse.html
  - https://hpc-training.sdsc.edu/expanse-101/
- Clone code examples for this tutorial:
  - https://github.com/sdsc-hpc-training-org/expanse-101
- SDSC Training Resources
  - https://www.sdsc.edu/education_and_training/training_hpc.html
- XSEDE Training Resources
  - https://www.xsede.org/for-users/training
  - https://cvw.cac.cornell.edu/expanse/