Q & A New User Training and Updated Best Practices on Perlmutter: February 15-16, 2024

Tinyurl link: http://tinyurl.com/bdh8z74x

Please post your questions for each talk below. NERSC staff will help to monitor the questions, read them to the presenters to answer them, or answer them here in the GDoc.

Day 1 User Survey is at: https://forms.gle/kKKrrbPa5bEh4AhX9 (updated)
Day 2 User Survey is at: https://forms.gle/EZwm554pLUckKhd69

The training event page is at
https://www.nersc.gov/users/training/events/2024/new-user-training-february2024/
Slides will be posted, video links will come soon.

(Apologies: zoom link gives “page not found” from training page link above?)

==========

Day 1

9:00 am - Welcome and Introduction to NERSC

Q (name): What does quota mean?
A (Lipi): Each user gets a “quota” which is the amount of space you get by default. You can check your quota by typing `showquota` on Perlmutter

Q ([redacted]): Are there plans to make $PSCRATCH accessible from the DTNs?
A (Helen): There are some technical difficulties in doing so, so there is no timeline when and whether it could be mounted on DTNs

Q (name): What is a “DTN”?
A (Lipi): The data transfer nodes are NERSC servers dedicated to performing transfers between NERSC data storage resources such as HPSS and the NERSC Global File System (NGF), and storage resources at other sites. These nodes are being managed (and monitored for performance) as part of a collaborative effort between ESnet and NERSC to enable high performance data movement over the high-bandwidth 100Gb ESnet wide-area network (WAN).

Q (name): I have many folders in $PSCRATCH however I am not given anywhere the option to download them, not even in Globus. What is the best and optimal way to backup my $PSCRATCH folders so that they do not get purged?
A (Lipi): You can move files that you need to CFS (Community File System) which is permanent. These will not get purged. Scratch is meant for keeping actively used files (outputs from computation), so when you are done using those files, moving them to CFS or archiving in HPSS is the best option. Once those files are on CFS, you can use Globus to move them off of NERSC.
A(Helen): All above are good. Just to add a quick way to archive to HPSS via command line option is to use htar directly from a login node on PSCRATCH.
A(Nick): You can access Perlmutter scratch from the NERSC Perlmutter collection on Globus. The NERSC DTN collection currently only has access to CFS and home directories.

Q ([redacted]): Are there any plans to support Rucio (https://rucio.cern.ch/) for data transfer?
A (Lipi): Helen may have more information about this but I am not sure about this.
A (Helen): ~~checking with Data and AI Services support staff, will report back~~. Rucio is a data management system which connects data transfer services together. Usually large collaborations like those in HEP (ATLAS, CMS, etc) use rucio to manage their data. If your data is already managed by rucio then it would be managed by an experiment. If you are part of an experiment that want to setup your own rucio setup at NERSC, please submit a ticket so we can talk with you more.

Q ([redacted]: Which category of appointment would you select for support on compiling a new program?
A (Lipi): The best route for this is to submit a user ticket or, you can even ask NERSC to install software for you using our software request form. We will discuss this briefly during the "Navigating NERSC" portion.
A (Helen): Also feel free to choose "NERSC 101" for the user appointment or if it is for running on GPU, then "GPU Basics".

Q ([redacted]): What's the better practice for simulation data output. Should it be output on PSCRATCH or CFS?
A (Helen): It is best to run out of PSCRATCH for best IO performance for scaling. The simulation data output if important, please be sure to back up in CFS or HPSS. Data on PSCATCH for more than 8 weeks with no access are subject to be purged.

9:30 am - Accounts & Allocations:

No Questions Submitted During Live Presentation

9:50 am - Navigating NERSC

Q (name): How do we access NERSC through SSH? Do you have any NERSC documentation links to look at?
A (Helen): https://docs.nersc.gov/connect/

Q (name): Is it possible to set up a way to bypass needing the MFA code every time so that one can automate, e.g., file transfers to external servers?
A (name): No, We enhance security by requiring multiple forms of verification before granting access. By passing this security measure could introduce vulnerabilities and increase the risk of unauthorized access to sensitive data or systems.
-sorry, I didn't word that very well. SSHProxy sounds like a very useful tool!

Q (name): I am familiar with Authy and have used it on a Ubuntu laptop - I just visited their page: https://authy.com/download/ and see that the desktop app is end of life in March 2024.
A (Helen): We are researching alternatives for an desktop app and you will hear more from us, starting from an email to all users first.
Meanwhile, I recommend you to download the Authy app if you have an iOS or Android phone. Or to use the Google Authenticator app.

Q ([redacted]): Is there a way to view the annual node hour usage for our account? I could not find it on Iris.
A (Helen): In Iris, there is a left side menu bar, that you could choose "report", and choose "usage tracking", or the direct link is https://iris.nersc.gov/reports/usage_track, you could choose cpu or gpu, current year or a specific previous year in the drop down box.
Also, when you login to Iris, go to the "cpu" menu, to see a list of your projects, and click on a project, then you will see the summary of current allocation, and hours used/remaining details on the top of the page

Q ([redacted]): Is there any documentation on the subtle difference between running a terminal from Jupyter versus actually logging into a perlmutter login node? I've had users attempt to set up their environment using the jupyter terminal and sometimes this causes confusion.
A (Helen): With Jupyter, after you login, you could choose the "terminal" kernel, then it is the same as ssh into Perlmutter. With Jupyter, you could also set up your own kernel from a conda environment for example. Here are some documentations:
https://docs.nersc.gov/services/jupyter/how-to-guides/#how-to-use-a-conda-environment-as-a-python-kernel
The entire Jupyter how-to guide is also very helpful:
https://docs.nersc.gov/services/jupyter/how-to-guides

Q (name): What is the advantage of accessing NERSC via SHH or via JupyterHub?

A (name): Good Question! It basically comes down to preference. Many of our researchers are not necessarily advanced computer scientists and prefer a more graphical based approach. JupyterHub provides a more UI-friendly interface.

Q (name): Is it recommended to use the recent windows terminal in recent windows 11 build to connect to NERSC? Additionally, what's the difference between ssh -X user@perlmutter@nersc.gov and ssh user@perlmutter@nersc.gov?
A (Charles Lively):Yes, using the latest terminal with windows 11 is fine With if that is your preference on connecting.

With regards to your other questions:

ssh -X user@perlmutter@nersc.gov` : The `-X` option is used to enable X11 forwarding. X11 forwarding allows you to securely forward graphical applications from the remote server (in this case, perlmutter@nersc.gov) to your local machine. This is useful if you need to run graphical applications remotely and display them on your local desktop.

`ssh user@perlmutter@nersc.gov`: This command simply connects to the remote server perlmutter@nersc.gov without enabling X11 forwarding. It establishes a secure shell (SSH) connection to the remote server, allowing you to interact with the command line interface of the remote system. This is useful for tasks such as running commands, editing files, and managing the remote server.

Q ([redacted]: Can I use any option to Log in? For example; CPU, GPU, etc.
A (Helen): You always login and land on to a compute node via SSH. Then you ask for a CPU or GPU compute node via salloc (interactive batch) or sbatch (batch job).
If you use Jupyter to access Perlmutter, you can choose from login node, CPU compute node, or GPU compute node, and land onto one of those directly.

Q (name): How do we access the Community File System (CFS) from Globus?
A (Helen): Once login to Globus, you can choose an end point, you can choose either NERSC Perlmutter or NERSC DTN, and access CFS from there.

10:40 am - Programming Environment & Compilation With Best Practices

Q ([redacted]): Any thoughts on "module purge"? I have users who seem to prefer completely removing all the default loaded modules, this continues to make me a little uneasy :)
A (Helen): module purge is not encouraged. There are a lot of lower level modules (besides the usual compiler, mpi, math libraries, etc.) that are required as you can see from the default loaded modules at login.

Q ([redacted]): If you load a programming environment, and then also set optimization flags or paths in a Makefile, will the Makefile override programming environment options? I don't know much about compilation, but for example I have used the PrgEnv-intel module and had bugs using MKL calls. The output is numerically incorrect. I will open a help ticket, but just curious if Makefiles clash with programming environments?
A (Helen): Loading a programming environment such as PrgEnv-intel or PrgEnv-gnu, it only makes the corresponding compiler as the underlying compiler when using the compiler wrappers such as cc, CC, and ftn. The optimization flags or libraries to be linked that are specified in your Makefile will be used for the compiler, and if no specific optimization flags are set, then the default compiler flags will be used.
Erik made a good point live that you probably should not provide own MPI libraries and such since the compiler wrapper would automatically include that.

Q (Alfred Tang) Docker has some security issues. Is it still true?
A (name): We use Shifter and Podman-hpc at NERSC, since Docker requires beyond-user level access on NERSC systems.

Q (name):
A (name):

11:20 am - Running Jobs and Best Practices

Q ([redacted]): The regular queue limits use to ½ of a node. Is this based on the number of processors, RAM, or a combination? If you request more than ½ of a node in SLURM, will the job fail or will it simply not allocate beyond the allowed limit?
A (Helen): The regular queue can use the entire node. The shared queue limits to ½ of a node. If you need more than ½ a node, please submit to regular queue.

Q (name): Is there any video tutorial for the following documentation and will it be covered in this training? Is this checkpoint-restart regularly used by NERSC users?
https://docs.nersc.gov/development/checkpoint-restart/
A (Helen): We have had some checkpoint restart training for Cori, but not Perlmutter yet. DMTCP works, MANA not yet for Perlmutter.
https://www.nersc.gov/users/training/past-training-events/2021/user-training-on-checkpointing-and-restarting-vasp-jobs-using-mana-on-may-25-2021/
https://www.nersc.gov/users/training/past-training-events/2020/online-hands-on-user-training-on-variable-time-jobs-on-thursday-may-21-2020/

Q (name): Is there any benefit to using Dask to complete a job vs just submitting a job with SLURM?

A (name): Dask can use some of the same benefits of using any other workflow tool. It can help to schedule tasks on multiple nodes instead of having slurm schedule all your tasks. If your work is already working in Dask then you can refer to these docs on some tips for using it at NERSC.

Q (name): Why is it recommended to use only 1 GPU per task?
A (name): Sorry if that was confusing. The recommendation was more to know there are ways to map GPUs to tasks and that can be important for your application.

Q (name):
A (name):

12: 00 pm - Day 1 Wrap-Up & Q&A

Q (name):
A (name):

Day 2

9:00 am - Recap of Day 1 and Follow Up Questions

Q ([redacted]): Is there a command to query available allocation on Perlmutter?
Something similar to ALCF's sbank --
https://docs.alcf.anl.gov/account-project-management/allocation-management/sbank-allocation-accounting-system/
A (Helen): yes, just do “iris” in command line. Use ‘iris -h” to find out more options.

Q (name):How to get the current ip address of the current server that the Python script is running in?
A (Helen): Try this command: “ip address” from a login or a compute node in a terminal session.

Q (name):Can we deploy a job to a specific server given the ip address?
A (name): Could you please clarify? Jobs are run on compute nodes (you submit jobs to a batch queue without specifying server IP address).

Q (name):Let us assume that a job (Job A) ends in 24 hours, can we automatically send another job (Job B) as soon as Job A ends or do we have to do this manually? What is the best way to achieve this and reduce the regular queue time?

A (Helen): You can use the dependency jobs feature as described at:
https://docs.nersc.gov/jobs/examples/#dependencies

Q (name):
A (name):

9:10 am - Using Python at NERSC and Best Practices

Q (name): What's the difference between conda environment and virtualenv environment?
A (Helen): Python virtual environments are the light version of the conda environments. Here is a good article:
https://notes.aquiles.me/difference_conda_environment_and_virtual_environment_in_pyhon

Q ([redacted]): Are there any concerns about using the conda installed cudatoolkit (in those cases where a conda install results in pulling in cudatoolkit) and unloading NERSC's cudatoolkit module? Or should we really be trying to give preference to the cudatoolkit module?
A (Helen): I posted this question internally at NERSC, and here are the comments from our experts:
- Daniel Margala: There is some discussion of that here:
https://docs.nersc.gov/development/languages/python/using-python-perlmutter/#cudatoolkit-dependency
- Lisa Gerhardt: From a file system perspective, users need to put their conda installs on global common or in a container. The system one is automatically positioned to be optimal from an I/O POV
- Daniel Margala: Related problem is that cudatoolkit is quite large, so you can same some space by reusing the cudatoolkit module. The main downside is that you are breaking the boundary of the conda environment, you are starting to depend on things that are not in your conda environment. You may be stuck if you need some other package in your conda environment that has a cudatoolkit dependency. when you install that package, conda will also install cudatoolkit.

Q ([redacted]: what does -no-cache-dir do in pip?
A (Helen): Using "--no-cache-dir flag in pip install" makes sure dowloaded packages by pip don't cached on system. This is a best practise which make sure to fetch ftom repo instead of using local cached one.

Q ([redacted]): Why shouldn't we use system python?
A (Helen): The system default python is version 2.7.18, very old, and no official support. NERSC provided default python is of version 3.11 currently. The Python we provide at NERSC is Anaconda Python. The NERSC python module also provides a python environment with several commonly used python packages pre-installed. Lots of NERSC Python documentations:

https://docs.nersc.gov/development/languages/python/
https://docs.nersc.gov/development/languages/python/python-intro/
(DanielM) you are welcome to use the system python if that meets your needs but we do not support it.

Q ([redacted]): Is conda-forge the default?
A (Helen): you can install packages from conda-forge as well as with pip at NERSC.
https://docs.nersc.gov/development/languages/python/using-python-perlmutter/#installing-from-conda-forge
https://docs.nersc.gov/development/languages/python/using-python-perlmutter/#installing-with-pip
https://docs.nersc.gov/development/languages/python/nersc-python/#overview-of-conda
([redacted]) what I meant is: if I just type conda install something, is the default channel conda-forge?
(DanielM) yes, conda-forge is the default channel if you are using the NERSC provided conda / python modules. the "main" channel provided by anaconda is also available by default but has lower priority than conda-forge.

9:35 am - Using Juypter at NERSC and Best Practices

Q ([redacted]) is jupyter notebook better for development work? For the simple reason that you don't want to open a notebook and wait for it to run for days.
A (Kelly): For rapid interactive development, it can be very useful, just like developing code in an interactive job via salloc. Notebooks can also be used on login nodes for lightweight development; it's generally the case that Jupyter servers start quickly (within seconds to minutes) for both compute jobs and on login nodes.

Q (name):
A (name):

10:00 am - Best Practices for Performance Tools, and Debugging and Best Practices

Q ([redacted]): Is there a profiler tool that has functionality similar to GNU gprof?
I am interested in getting function/subroutine timings primarily, then
line-by-line timings if possible from a CPU-only application. Thanks,
https://ftp.gnu.org/old-gnu/Manuals/gprof-2.9.1/html_node/gprof_17.html#SEC17
A (name): Valgrind can do this with the HPC additions. Additionally you can get this with craypat, Forge MAP, etc. It depends on the parallel programming model support of the tool and what your application is using.

Q (name):
A (name):

10: 45 am - IO, Data Storage & Sharing Best Practices

Q ([redacted]): Our /global/common/software/<project name> is a shared directory by the whole institution. I am not sure if I have permission to use it. When installing conda environments, could I use $CFS/<project name>/<new group directory> instead?
A (Helen): $CFS stands for /global/cfs/cdirs, so you do need add a <project name> to it to use your own project space.
You can create subdirectories under /global/common/software/<project name>, although ACLs are discouraged to control permission. You can create a ticket to request for a new file group instead.
(Steve) It's best **not** to put conda environments in $CFS though, use /global/common/software/<my_project> instead

Q (name): How can we download entire folders/directories?
A (Helen): you could use tar (or htar to archive the directory to HPSS)
A(Lipi): tar to compress the folder/directory and then globus to move it off of NERSC

11:05 am - Interactive Discussion and Q&A + Closing Remarks

Q (name):
A (name):