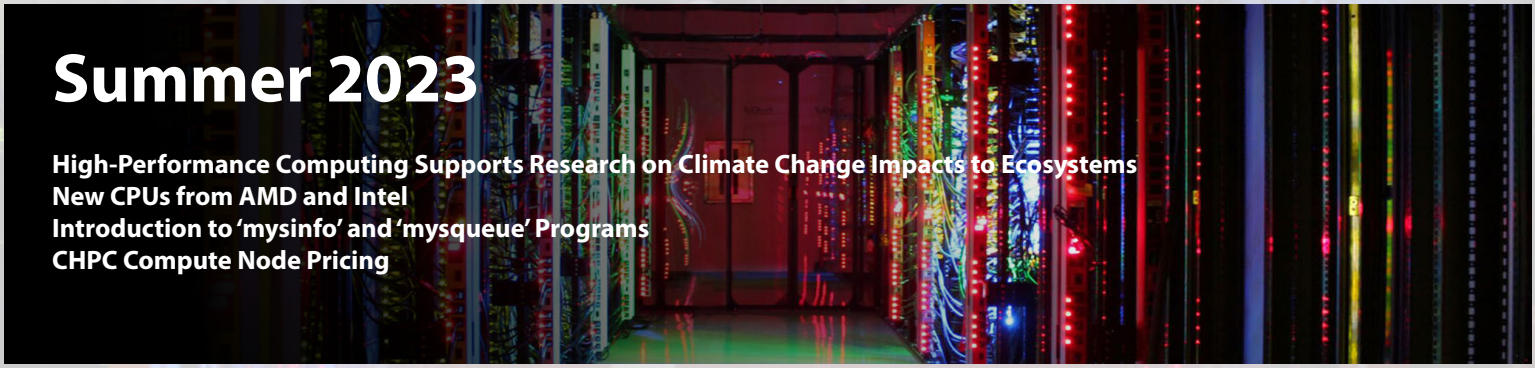
Center for
HIGH PERFORMANCE COMPUTING
THE UNIVERSITY OF UTAH

Newsletter



Summer 2023

High-Performance Computing Supports Research on Climate Change Impacts to Ecosystems
New CPUs from AMD and Intel
Introduction to 'mysinfo' and 'mysqueue' Programs
CHPC Compute Node Pricing

High-Performance Computing Supports Research on Climate Change Impacts to Ecosystems

Professor William Anderegg - School of Biological Sciences, University of Utah

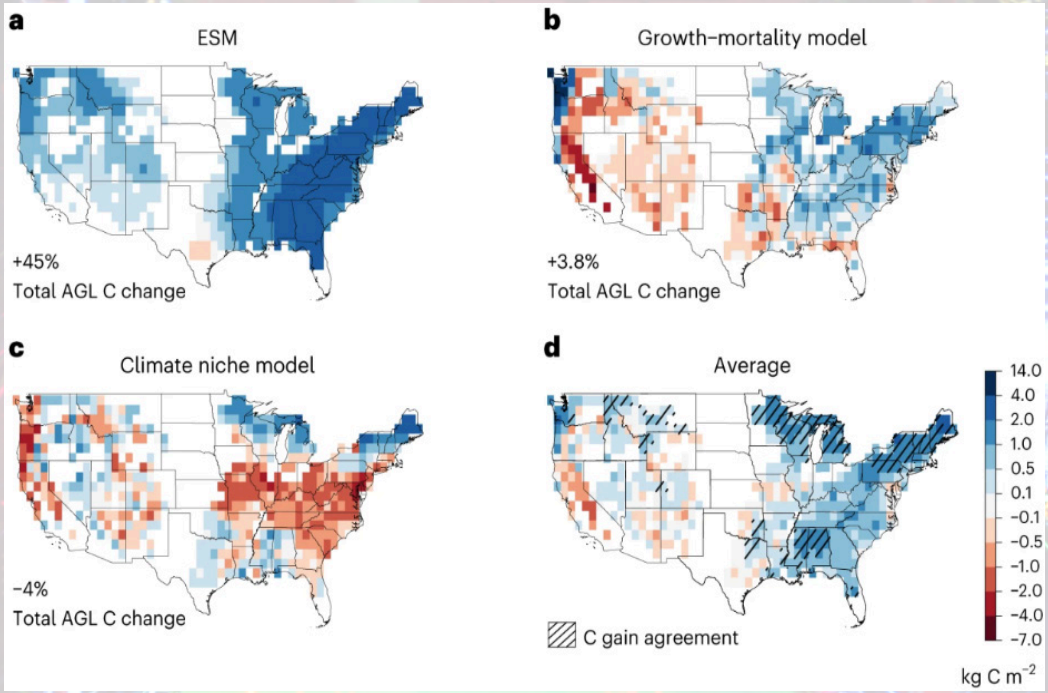


Figure 1: Projections for end-of-century C storage potential in CONUS forests diverge depending on scientific approach (Wu at al., 2023, Nature Geoscience)

Computational modeling of terrestrial ecosystems

Terrestrial ecosystems may have considerable potential to mitigate anthropogenic climate change through carbon sequestration, yet the future of ecosystems in a rapidly changing climate is uncertain. On one hand, rising atmospheric CO_2 concentrations stimulate plant photosynthesis and carbon accumulation. However, on the other hand, climate change also increases the occurrence of climate risks, such as drought and wildfires that may undermine the carbon storage of terrestrial ecosystems. In addition, human activities, for example, land use change and harvest also impact land carbon balance. Therefore, the future of terrestrial carbon storage remains unclear and how the future of terrestrial carbon will change is a long-standing ecological question. Scientists use several approaches to tackle these questions, including the mechanistic land surface models, such as those incorporated in Earth system models (ESMs) in the Coupled Model Intercomparison Project Phase 6 (CMIP6), which simulate forest carbon dynamics through complex ecological, physiological, and phenological processes and ecosystem interactions with the atmosphere. Because of the complexity and high computation demand of these models, high-performance computers are usually used for running long-term large-scale Earth system modeling projects. The Anderegg group at the University of

Utah focuses on forest ecosystem modeling and has developed a method to examine the carbon permanence of US forests in response to climate change risks with collaborators using the computing and analysis resources at CHPC. Our initial results suggest diverging signs and magnitudes of projected future forest aboveground carbon storage potential across three broad modeling approaches, i.e., Earth system models, a growth-mortality model, and a climate niche model (Fig. 1).

Each approach used here has inherent strengths and limitations. The mechanistic land surface models within ESMs are built to examine long-term vegetation dynamics, stocks and fluxes, including the benefits of rising CO_2 levels, but they still require further process development to skillfully represent the disturbance-driven mortality mechanisms. The other two approaches do not have the same mechanistic underpinning in response to climate and atmospheric CO_2, but provide more reliable vegetation–climate relationships and disturbance representation than the current capability of ESMs.

However, compared to the other two approaches, the projection from ESMs unreasonably overestimated the forest carbon storage potential. One of the reasons might be that these CMIP6 ESM simulations are originally used for global-scale analyses at low spatial resolution (e.g., 2-3 degrees). Therefore, to improve our understanding of the future terrestrial carbon storage in the US, a finer-scale simulation that should include detailed information about spatial heterogeneity in soil and vegetation types is urgently needed.

Our group is currently running a project on CHPC about modeling the historical and future terrestrial carbon storage and fluxes in the US using four state-of-art mechanistic vegetation models at a high spatial resolution (8km).

The simulation will cover the periods from 1950 to 2014 in the historical and from 2015 to 2100 under three future climate scenarios which cover future lower and upper boundaries of the climate and land use risks. We believe this new project will significantly add to our new understanding of the potential and risks of future terrestrial carbon stocks in the US and provide a critical foundation to guide ecosystem conservation, restoration, and nature-based climate solutions.

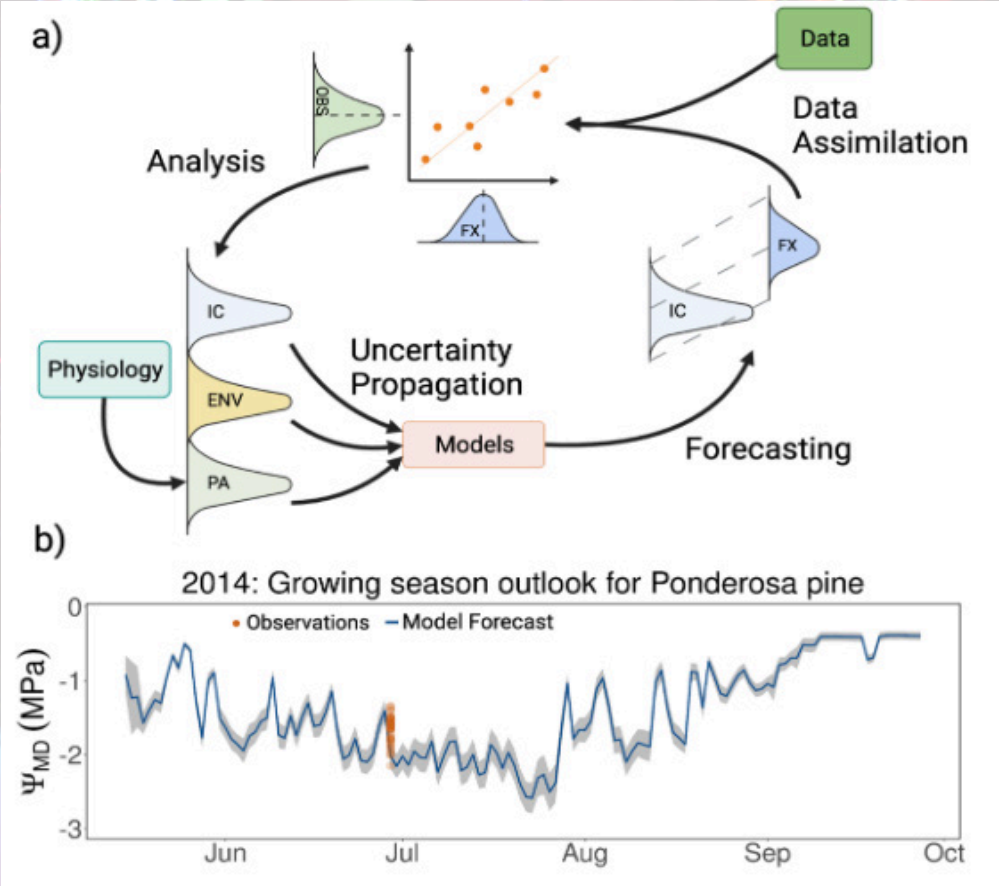


Figure 2: a) The ecological forecasting cycle of vegetation responses to drought. In this cycle, prior distributions of the initial conditions of vegetation (IC), environmental data (ENV), and physiological parameters (PA) define the uncertainty of the parameters used in vegetation models. Model performance is assessed by comparing data assimilation of observed forest dynamics responses to drought with responses from the forecasted posterior distribution. Diagram created in www.biorender.com. b) Forecasted seasonal midday water potentials (Ψ_{MD}), a metric of plant water stress, for Ponderosa pine as a metric of drought stress, where values below -2 MPa indicate acute water stress.

Forecasting climate risks to western forests

Over the last two decades, climate change has brought about extreme weather conditions in the Inter-Mountain West (IMW) region of the United States. The IMW has experienced decreasing precipitation that has resulted in the death of many trees, significant financial losses, and concerns about the perpetuity of forest ecosystem services like carbon dioxide uptake, erosion control, and water cycle regulation. To plan effectively for climate change, it's crucial to understand how ecosystems will respond on a seasonal to decadal scale. This is where the rapidly growing field of 'ecological forecasting' comes in. Similar to weather forecasts that we're familiar with in our day-to-day lives, ecological forecasts aim to predict changes in ecosystems in response to environmental drivers such as climate variability or extreme weather events using an iterative computational framework and mechanistic model. By predicting the impacts of drought in the near term, stakeholders can gain a better understanding of the potential consequences and uncertainties for forest ecosystems. The major challenge when developing reliable forecasts of vegetation responses to drought is the limited data available on the water status of ecosystems. Ecosystem models partially address this challenge by using mathematical equations based on first principles of physiology and ecology (e.g. photosynthesis) to simulate how plants and ecosystems respond to environmental conditions. When used within an iterative forecasting framework (Fig. 2), Ecosystem models become valuable tools providing high temporal resolution information on plant water status that otherwise would have been financially or logistically impossible to measure. Ecosystem models require extensive datasets that include detailed weather information, vegetation, and soil characteristics. To account for model uncertainty and generate timely and reliable ecological forecasts, these models must be run multiple times across different locations within a region. CHPC at the University of Utah provides the efficient computing infrastructure required to achieve such a task. With the CHPC resources and the invaluable help of their staff, we are working to characterize data uncertainty and propagate it to ecosystem models to provide reliable forecasts of vegetation drought stress in the IMW.

Quantifying nature's complexity represents a challenge, and our work intends to improve the way we do so. We use long-term forest monitoring data from the San Juan National Forest in Southwestern Colorado as a case study to develop ecological forecasts of drought stress in two important tree species: ponderosa pine and trembling aspen. We collect data on forests, soils, and vegetation physiological characteristics along an elevation gradient on the hillslopes of Mt. Hesperus. The computing infrastructure found within the CHPC allows us to efficiently use the field data to run ecosystem models that estimate plant water status for 30 sites and two distinct model scenarios at each site totaling around 17 million timestep simulations that span environmental conditions from 1990 to 2023 in those 30 forest stands.

In the summer of 2023, we will collect more data in the field that will tell us the water status of various ponderosa pine and trembling aspen stands. We'll then compare this data with our forecasted model to assess the accuracy of our simulations. This project represents one of the first attempts to produce near-term ecological forecasts of plant water status and hopefully the beginning of a line of work aimed at predicting the fate of Inter-Mountain West forests considering climate change.

References

Wu, Chao, Shane R. Coffield, Michael L. Goulden, James T. Randerson, Anna T. Trugman, and William RL Anderegg. "Uncertainty in US forest carbon storage potential due to climate risks." Nature Geoscience 16, no. 5 (2023): 422-429.

New CPUs from AMD and Intel

Martin Cuma, CHPC Scientific Consultant

In this article we look at the performance of fourth generation AMD EPYC CPUs (code named Genoa), and at Intel fourth generation Xeon Scalable processor (code named Sapphire Rapids). The AMD CPUs were released at the end of 2022, and the Intel CPUs in early 2023. A *white paper* provides more results and details. Both AMD and Intel release a wide range of CPU models with different clock speed, core counts and other features, which results in a wide range of performance. We focus on mid range CPU models, which are of most interest for CHPC with regards to the price/performance ratio, and are related to the previous CPU generations, that we have been buying. There was only a minor difference in the OS version between the test machines, running Red Hat Enterprise Linux (RHEL) 8.6, and the CHPC machines, running Rocky Linux 8.5, a clone of RHEL. On both sites, the Spack package manager was used to build the applications. Using Spack may not allow for the fine tuned performance, but, it mimics the way we build applications on our systems and as such gives appropriate comparisons to performance on existing CHPC machines and expectations for the new CPUs performance on our systems.

A server vendor gave us access to one Intel and three

AMD CPU models in their lab. The new and previous generation CPUs we look at are summarized in Table 1.

High Performance Linpack (HPL)

HPL solves a dense system of linear equations in double precision and is a base for the TOP500 list of the fastest computers in the world. It gives a good estimate of the raw double precision CPU performance. It uses the Basic Linear Algebra Subprograms (BLAS), which we have provided either with Intel's Math Kernel Library (MKL), or with OpenBLAS.

The MKL uses a CPU check at runtime, which decreases its performance for the AMD CPUs. We see about 20% performance reduction as compared to when this CPU check is disabled on the AMD 9334 CPU.

OpenBLAS performance is about 5% worse than MKL on the same CPU, which is why we only report MKL results. The raw performance increased impressively from the previous to current generation of both AMD and Intel CPUs, almost 90% for the AMD and 50% for the Intel. The theoretical peak performance and the percentage of theoretical peak achieved by the CPUs is discussed in the *white paper*.

High Performance Computing Challenge (HPCC)

HPCC is a synthetic benchmark suite designed to assess HPC performance from different angles. It consists of seven main benchmarks, that stress various computer subsystems, such as raw performance, memory access and communication. For detailed description of the benchmark see *http://icl.cs.utk.edu/hpcc/*.

The HPL result is about 20% less, than with the specific HPL runs above, that's likely due to the parameters of the run. Nevertheless, the trends of impressive gains with the new CPUs are preserved.

The SingleDGEMM result is in Intel's favor, highlighting the more performant vector units per Intel CPU core. The rest of the benchmarks are more memory bound and favor the AMD CPUs. The AMD has all core memory bandwidth up to ~350 GB/s, while Intel has ~250 GB/s using the DDR5-4800 memory, thus the AMD has an advantage. The 6430 Intel CPU also only supports up to DDR5-4400, not 4800, which will make the Intel peak even lower.

LAMMPS

LAMMPS is a popular molecular dynamics simulation program developed at Sandia National Laboratory. It is a good representative for multi-body like simulations, that use internally coded computational kernels, not relying so much on vendor accelerated libraries.

We have run three LAMMPS benchmarks from *http://lammps.sandia.gov/bench.html* (see *white paper* for details). In Figure 1 we compare the runtime on a whole node between the select CPUs. Intel has improved

CPU	AMD Genoa 9334	AMD Genoa 9554P	AMD Genoa 9654P	Intel Sapphire Rapids 6430	AMD Milan 7713P	Intel Ice Lake 6330
Core Count	2x32	64	96	2x32	64	2x28
Base Clock Speed (GHz)	2.70	3.10	2.40	2.10	2.00	2.00
Boost Clock Speed (GHz)	3.90	3.75	3.70	3.40	3.68	3.10
Base TDP (Watts)	210	360	360	270	225	205
HPL (TFLOP/s)	3.11	2.94	3.26	3.23	1.38	1.87
SingleDGEMM (GFLOP/s)	59.9	58.5	50.7	71.1	31.8	60.3
PTRANS (GB/s)	29.85	24.62	29.13	18.2	15.34	22.04
MIPRandomAccess GPUs	0.445	0.472	0.579	0.337	0.475	0.309
SingleStreamTriad	43.2	41.1	33.9	13.3	24.7	14.3
MPIFFT (GFLOP/s)	89.58	65.44	68.11	57.73	22.69	24.01
HPL Score	3.527	3.319	3.811	3.435	1.749	2.346
Max Theoretical HPL	2.765	3.174	3.686	4.301	2.048	3.584
% Max Theoretical HPL	128%	105%	103%	80%	85%	65%
Processor List Price	$5,980	$7,104	$10,625	$4,256	$5,010	$3,788

Table 1: New and previous generation performance comparison

markedly from Skylake to Sapphire Rapids, but, so did the AMD Genoa from the Milan and Rome CPUs, for the Ice Lake and Milan, part of it may be use of the Nehalem optimized binary.

Unless the price of the 96 core AMD 1P node is more than 20% higher than of the 64 core 1P node, the 96 core node is a better choice for LAMMPS.

GROMACS

GROMACS is another molecular dynamics program, similar to LAMMPS, but, it does its internal assembly optimizations for various CPU microarchitectures, which makes it important to build it for that particular microarchitecture. We ran two different benchmarks, benchMEM, 82k atoms, protein in membrane surrounded by water, 2 fs time step, and benchRIB, 2M atoms, ribosome in water, 4 fs time step, obtained from the Max Planck Institute for multidisciplinary sciences.

In Table 2 we present the GROMACS performance on the whole node. The AMD 1P 64 core node has 12% to 16% better performance than the Intel node. The 96 core 1P AMD node has 10-20% advantage over the 64 core 1P node. And, the 2P vs 1P 64 core node has similar performance. GROMACS shows similar trends to LAMMPS, with the 96 core 1P AMD node being the best choice, as long as the price is not more than 10-20% higher than that of the 64 core 1P node.

NWCHEM

NWCHEM is a quantum chemistry simulation program which depends heavily on dense linear algebra provided by BLAS and LAPACK, which performance should indicate that of other quantum chemistry simulations like VASP or Gaussian. The advantage of NWCHEM is that it is open source and buildable by Spack, although on the CHPC systems, we have used a prior build of NWCHEM 7.0.2 with Intel MPI and MKL. We did not succeed to build 6.8.1 on CHPC, and 7.0.2 on the new test system, due to dependency issues that were difficult to resolve. Therefore the comparison to the older CPUs is not ideal.

We look at the C_{240} buckyball benchmark, which is fairly widely used and published.

Table 2 shows the runtime for the C_{240} benchmark run on the whole node. There is a moderate improvement from Milan to Genoa, and no improvement from Ice Lake to Sapphire Lake. This may be due to the speed improvements done in the version 7.0.2 as compared to

CPU	AMD Genoa 9334	AMD Genoa 9554P	AMD Genoa 9654P	Intel Sapphire Rapids 6430	AMD Milan 7713P	Intel Ice Lake 6330
GROMACS bench-MEM	145.9	150.1	164.9	134.5	84.0	112.8
GROMACS bench-RIB	13.1	12.8	15.4	11.1	5.9	7.9
NWCHEM C_{240}	971	968	961	1,354	1029	1358

Table 2: Whole node GROMACS performance (ns/day, higher is better) and NWCHEM C_{240} runtime (seconds, lower is better).

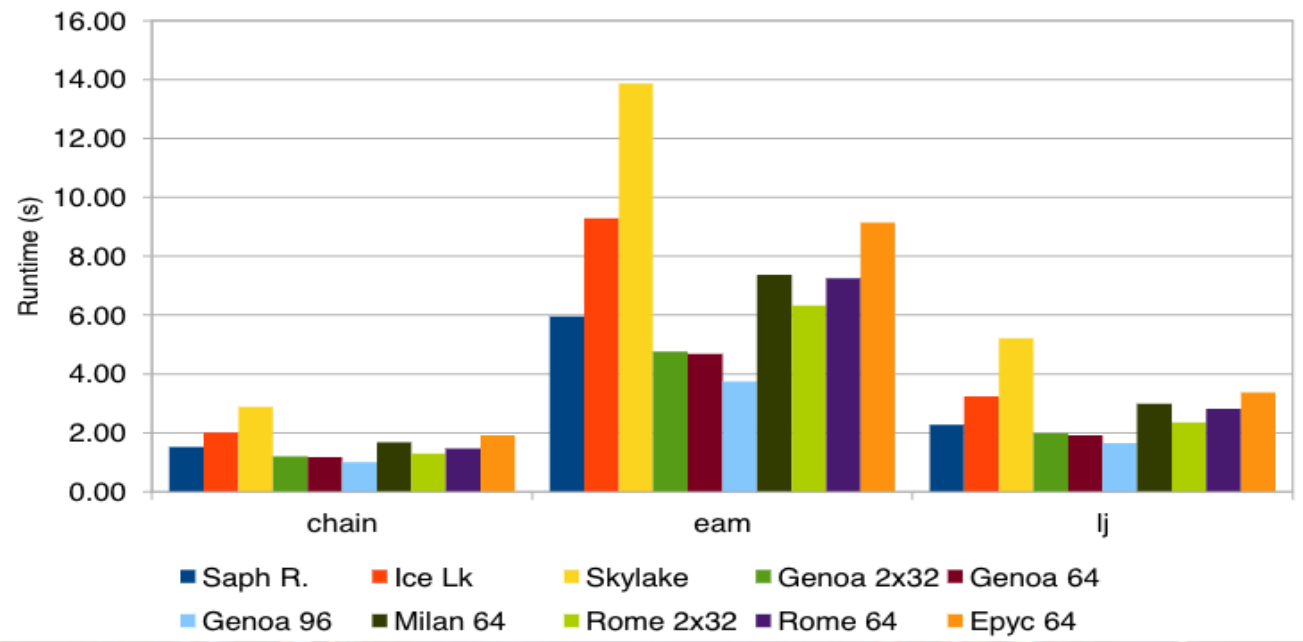


Figure 1. Whole node performance of the three LAMMPS benchmarks, in seconds, lower is better.

6.8.2 and warrants revisiting once we get nodes with the new CPUs in the CHPC environment.

There's also not too much difference between the three AMD CPUs. Furthermore, the runtime is about the same for MKL (shown in the table) and OpenBLAS, and disabling the CPU ID check in MKL only yields about 1% speed up, as compared to about 20% in the HPL, which suggests that this NWCHEM run is not bound by the BLAS/LAPACK.

Conclusions

Both the Intel Sapphire Rapids and AMD Genoa generation CPUs provide a significant performance boost for most applications, anywhere from 20-50%. The AMD CPUs are more performant than those of Intel, and the 96 core 1P AMD CPU is a better choice, if it's priced no more than 20% more than the 64 core 1P node for CPU bound applications like molecular dynamics, or dense linear algebra.

Introduction to 'mysinfo' and 'mysqueue' Programs

Adri Kingston, CHPC Computing Technician

Two commonly used SLURM commands are sinfo and squeue. The sinfo command shows the partition and node information of a system, whereas squeue shows the information about the jobs of the system. For the CHPC clusters, with many different partitions often having hundreds of jobs at any given time, this information can be overwhelming, and it can be difficult to find ways to parse the data to get to the information that pertains to a given user. mysinfo and mysqueue are scripts that address these problems. Similar to the myallocation script, these scripts aim to simplify the tasks of querying states of nodes that a user can access as well as seeing what jobs are running or are in the same queues that might be waiting for resources.

The mysinfo script displays the partitions you can access, and the account used to access them. It merges output for partitions that have overlapping node sets and appends the account that gives you access to those partitions. For example, partitions mygroup-np and mygroup-shared-np are two partitions, one set up for shared jobs and one set up for exclusive jobs. The rest of the output is standard sinfo output.

The mysqueue script provides user catered information about what jobs are running, who is running them, and for how long. To use the mysinfo script, simply type `mysinfo` into any terminal on a CHPC cluster node that has the chpc/1.0 module loaded. That's it. With no additional parameters, the default behavior for mysinfo will show you all the partitions you have access to in a streamlined list. Now, from here if you type `mysinfo -h` you will be met with a prompt detailing all the flags as well as examples of how to use them. In the case you're an avid reader and a textual learner, I will cover these flags below as well. To look at someone else's mysinfo or mysqueue results, simply pass their username as an argument after the command like so: `mysinfo someUser`.

Note, you can also achieve this same behavior with the user -u flag. However, if you choose to simultaneously pass a user using the -u flag as well as pass another user as the base argument, the script will drop the -u flag as well as the user defined by it. Example: `mysinfo -u someUser someOtherUser` . In this example, the results would be from 'someOtherUser' and 'someUser' would just be forgotten.

The -f flag changes the format of the output giving you more details on the partitions. For mysinfo the -f flag adds the number of nodes, their current state, their Sockets, Cores, and Threads (S:C:T), memory, and local storage. For mysqueue, the -f flag adds the full name of the process running on the node if there is one, as well as the state of the node, the number of nodes used, the time limit on the job as well as the time left on the job.

Neither of the scripts pass additional arguments to the vanilla sinfo or squeue command, and attempting this will generate an error. Both scripts will default to showing you all the accounts you have access to. To isolate accounts use the -A flag.

The -A flag for both scripts behaves identically. It will filter your results based on an account. Here is an example of how to run this flag: `mysinfo -A chpc` . This will only show me what partitions I have access to from the chpc account. If, in the previous example you had access to no partitions at CHPC, the command will fail with an error message.

The -c flag, or colorless flag, disables the colors in the output. This is useful if using the scripts for other scripts or using a terminal that doesn't support colors.

The -M flag is used to specify the cluster you want to look in. Here is an example of how to use it: `mysqueue -M someCluster` . This will show you your mysqueue results on someCluster rather than the default which is the cluster you are on.

The -h flag is the most important flag. This flag will show you all the flags, describe what they do, and some will provide examples.

Now that our textual learners are satisfied, I will provide some examples for the visual learners like myself.

In the output below, the first panel displays the output of the original sinfo command, showing partitions somepartition-sc and somepartition-shared-sc which have the same node list, while the second panel shows an example of how mysinfo combines both outputs and merges them.

```
[someUser@someCluster1:~]$ sinfo -p somepartition-sc,somepartition-shared-sc
PARTITION           AVAIL  TIMELIMIT  NODES  STATE NODELIST
somepartition-sc          up 14-00:00:0     1    mix node316
somepartition-sc          up 14-00:00:0     5  alloc node[312-315,317]
somepartition-shared-sc   up 14-00:00:0     1    mix node316
somepartition-shared-sc   up 14-00:00:0     5  alloc node[312-315,317]
```

```
[someUser@someCluster1:~]$ mysinfo

mysinfo: Below are the partitions and node states that you have access to on someCluster

PARTITIONS: somepartition-sc somepartition-shared-sc listed below. Accessible through account: somegroup-sc somegroup-shared-sc
PARTITION    AVAIL  TIMELIMIT   NODES  STATE  NODELIST
<see_above>  up     14-00:00:0  1      mix    notch316
<see_above>  up     14-00:00:0  5      alloc  notch[312-315,317]
```

As you can see in the following figures, if you were looking for 'somepartition-sc' or 'somepartition-shared-sc' you absolutely wouldn't have found it in the first 18 lines of the vanilla squeue output, and it's worth mentioning the hundreds of lines that would have followed would probably have been just as useless to you, and maybe you would do what I do and just learn to filter the output using the grep command and some key words. Doing that though is inefficient and in some cases unreliable depending on how you filter the output. sinfo and squeue are still useful commands, however, when you have specific requirements and wish to use more complex parameters. Our goal was to address common use cases with mysinfo and mysqueue to help simplify what can be an overly complex set of output. If you would like to suggest additional use cases that we should consider for mysqueue or mysinfo please send in your suggestion to *helpdesk@chpc.utah.edu*.

```
[someUser@someCluster1:~]$ squeue |head -18
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
           8042757 notYours-    Nope2 notYouID  R 3-03:40:26      1 node363
           8042755 notYours-    Nope1 notYouID  R 3-03:44:29      1 node362
           8040470 notYours-  wrong41 notYouID  R 4-05:07:15      1 node361
           8040456 notYours-  wrong71 notYouID  R 4-05:08:44      1 node364
           8076547 stlnotit- misleads stillNoo  R       3:55      1 node065
           8075439 nope-gpu- big_mstk k33pLook  R    6:20:53      1 node405
           8075430 nope-gpu- big_mstk k33pLook  R    6:33:53      1 node405
           8066741 others-sc  048cont ingForIt  R   23:25:30      1 node120
           8066296 others-sc  032cont ingForIt  R   23:25:45      1 node123
           8076489 notminee-   badcmd wrngUser  R      22:09      1 node072
   8076003_[42-48] someOther confused eyeSores PD       0:00      1 (JobArrayTaskLimit)
        8076003_35 someOther confused eyeSores  R      51:40      1 node399
        8076003_36 someOther confused eyeSores  R      50:47      1 node399
        8076003_39 someOther confused eyeSores  R      38:29      1 node399
        8076003_38 someOther confused eyeSores  R      38:51      1 node399
        8076003_37 someOther confused eyeSores  R      40:38      1 node399
        8076003_40 someOther confused eyeSores  R      31:37      1 node399
```

```
[someUser@nsomeCluster1:~]$ mysqueue

mysqueue: Below are the partitions and node states that you have access to on someCluster

PARTITIONS: somepartition-sc somepartition-shared-sc listed below. Accessible through account: somegroup-sc somegroup-shared-sc
JOBID    PARTITION    NAME      USER        ST  TIME         NODES  NODELIST(REASON)
8062665  <see_above>  job_some  otherUser1  PD  0:00         2      (Dependency)
8062664  <see_above>  job_some  otherUser1  PD  0:00         2      (Dependency)
8062663  <see_above>  job_some  otherUser1  PD  0:00         2      (Dependency)
8062662  <see_above>  job_some  otherUser1  PD  0:00         2      (Dependency)
8062660  <see_above>  job_some  otherUser1  PD  0:00         2      (Dependency)
8062659  <see_above>  job_some  otherUser1  PD  0:00         2      (Dependency)
8062658  <see_above>  job_some  otherUser1  PD  0:00         2      (Dependency)
8062661  <see_above>  job_some  otherUser1  R   1-05:48:57   2      node[314-315]
8062657  <see_above>  job_some  otherUser1  R   1-05:49:02   2      node[312-313]
```

CHPC Compute Node Pricing

Anita Orendt, CHPC Scientific Consultant

With the release of the new generation CPUs discussed in Martin's article in this newsletter, we have obtained pricing for servers with these new processor lines. At this time, we can get servers with either the new processor lines or with the AMD Milan and Intel Ice Lake processors. The prices given are for servers configured to be added to the current CHPC clusters of notchpeak in the general environment and redwood in the protected environment. The AMD Milan as well as the Intel based servers have a 7 year warranty included, whereas the AMD Genoa systems have only a 5 year warranty. Note that with the newer generation AMD servers, due to a change in the number of memory channels the memory footprints have changed.

The costs in the table are the cost from the vendor, which are subject to change. To this cost, CHPC adds on $1200/node to cover the per node cost of the cluster infrastructure. If you are interested in another configuration, whether a different CPU only node or a node with GPUs, please reach out to CHPC via *helpdesk@chpc.utah.edu* and we will work with you to get quotes.

	AMD Genoa 9334	AMD Genoa 9454	AMD Genoa 9554P	AMD Genoa 9654P	Intel Sapphire Rapids 6430	AMD Milan 7713P	Intel Ice Lake 6330
Core Count	2x32	2x48	1x64	1x96	2x32	1x64	2x28
256 GB memory					$8,963	$6,597.45	$7,385.58
512 GB memory					$9,465	$8,122.96	$8,783.26
1 TB memory					$11,033	$11,938.73	$12,198.57
384 GB memory	*	*	$9,596	$10,447			
768 GB memory	*	*	$10,936	$11,787			
1536 GB memory	$14,847	$16,336	**	**			

** Costs more than single CPU processor with same number of cores ** Costs more than two CPU processor with same number of cores*

The CHPC Fall 2023 Presentation Series will start on Monday September 28, 2023. For the entire schedule see URL: *https://www.chpc.utah.edu/presentations/fall2023chpcpresentationschedule.php*

Please acknowledge the use of CHPC Resources

If you use CHPC computer time or staff resources, we request that you acknowledge this in technical reports, publications, and dissertations. An example of what we ask you to include in your acknowledgment is:

"A grant of computer time from the Center for High Performance Computing is gratefully acknowledged."

If you make use of the CHPC Protected Environment, please also acknowledge the NIH shared instrumentation grant:

"The computational resources used were partially funded by the NIH Shared Instrumentation Grant 1S10OD021644-01A1."

Please submit copies or citations of dissertations, reports, pre-prints, and reprints in which CHPC is acknowledged by sending to *helpdesk@chpc.utah.edu.*

The University of Utah
University Information Technology
Center for High Performance Computing
155 South 1452 East, Room 405
SALT LAKE CITY, UT 84112–0190

Center for HIGH PERFORMANCE COMPUTING
THE UNIVERSITY OF UTAH