NUG Webinar 20/10/2016

Announcements

Agenda

Cori status update (Tina/Jay)

Announcements:

- Science gateways - cutover to new hardware (Annette)
- NESAP for Data proposals (Annette)
- Upcoming outages (Steve)
- Upcoming training (Steve)

Edison queue wait times (Richard/Steve)

Preparing for Cori's return:

- New affinity options (Helen)
- Procedure for getting early access to Cori phase 2 nodes (Thorsten)

Science Gateways

Cutover to new hardware on Tuesday Oct 25

Users should have tested their gateways on the new hardware

https://www.nersc.gov/users/announcements/featured-announcements/science-gateways-moving-to-berkeley-hardware

Report issues to help@nersc.gov

NESAP for Data

Applications due Tuesday, November 1, 2016

https://www.nersc.gov/users/announcements/featured-announcements/call-for-proposals-nesap-for-data/

NESAP: NERSC partners with code teams and library and tool developers

To prepare for Cori KNL architecture.

The teams gain access to resources at NERSC, Cray, and Intel

NESAP for Data: extension of NESAP targeting data-intensive science applications

For processing/analysis of massive datasets acquired from experimental and observational sources

To take full advantage of Cori KNL architecture

Upcoming Outages

Saturday 22 Oct, 5am-3pm

Power work in CRT - only Mendel (JGI, NP, HEP, Materials, Carl) is affected

Tuesday 8 Nov, 8am-11am

Quarterly maintenance

Minor disruptions only, queues paused, users can still submit jobs

Upcoming Training

Thurs 3 Nov 9am-4pm

Cori KNL training https://www.nersc.gov/users/training/events/cori-knl-training

- How to get access
- KNL architecture
- Performance tuning tools

Weds 9 - Fri 11 Nov

NERSC VASP workshop

http://www.nersc.gov/users/training/events/3-day-vasp-workshop/

VASP Developer Dr. Martijn Marsman from U Vienna

Suitable for beginners to advanced users

Edison Queue Wait times

We know the wait times on Edison are very long!

We are working (very) hard to have Cori Haswell nodes ready for users on Nov. 1

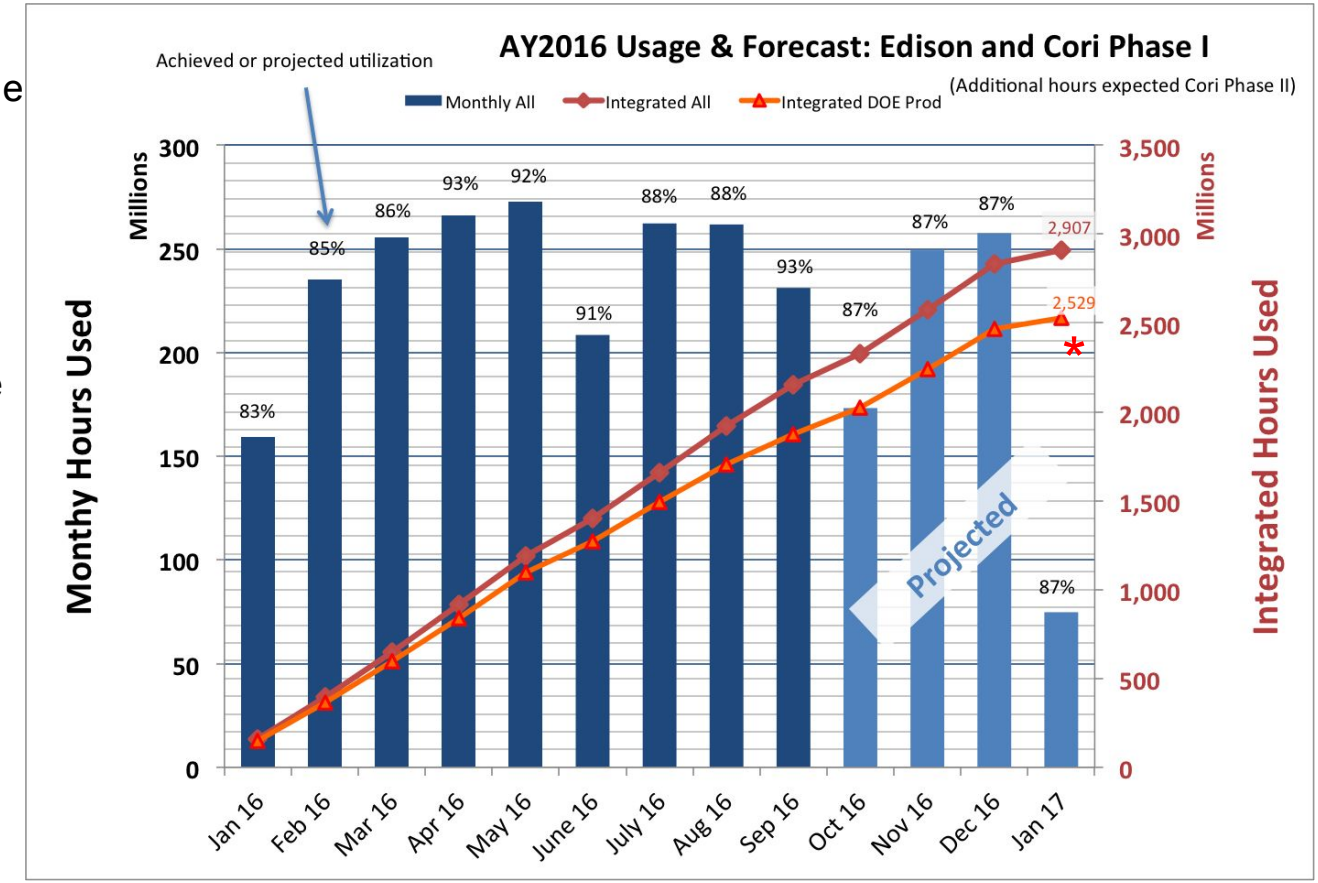
Sep. 1-17 <Wait>: 9 hours

Oct. 1-17 <Wait>: 161 hours

We're on Track to Deliver Hours

We planned for this outage, so NERSC will still deliver the 2.4 billion hours allocated to DOE * and the hours allocated to ALCC on Cori Haswell nodes and Edison.

Cori KNL nodes will become available to NESAP teams and other early users before AY16 ends, providing additional hours to science.



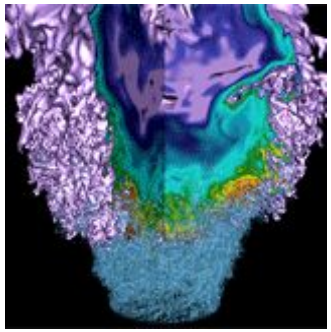
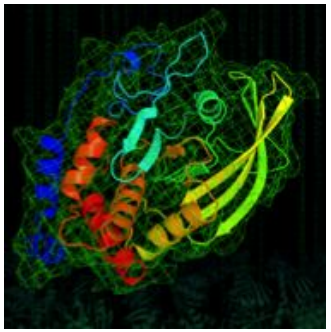
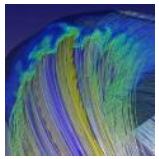
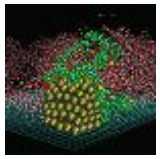
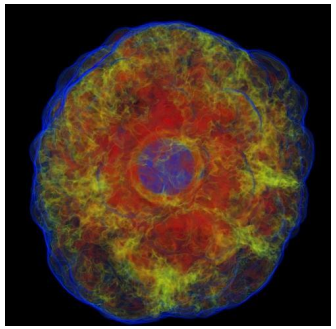
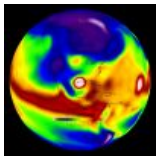
We Appreciate Your Patience

If you have low priority work that can wait until Nov. 1, other users would appreciate your consideration.

We're working to make sure Edison achieves maximum utilization and the scheduler is treating all jobs fairly.

Edison job queue backlog

Upcoming Changes for Running Jobs on Cori Haswell Partition


NERSC

Helen He, Zhengji Zhao, Steve Leak

October 20, 2016

U.S. DEPARTMENT OF ENERGY | Office of Science

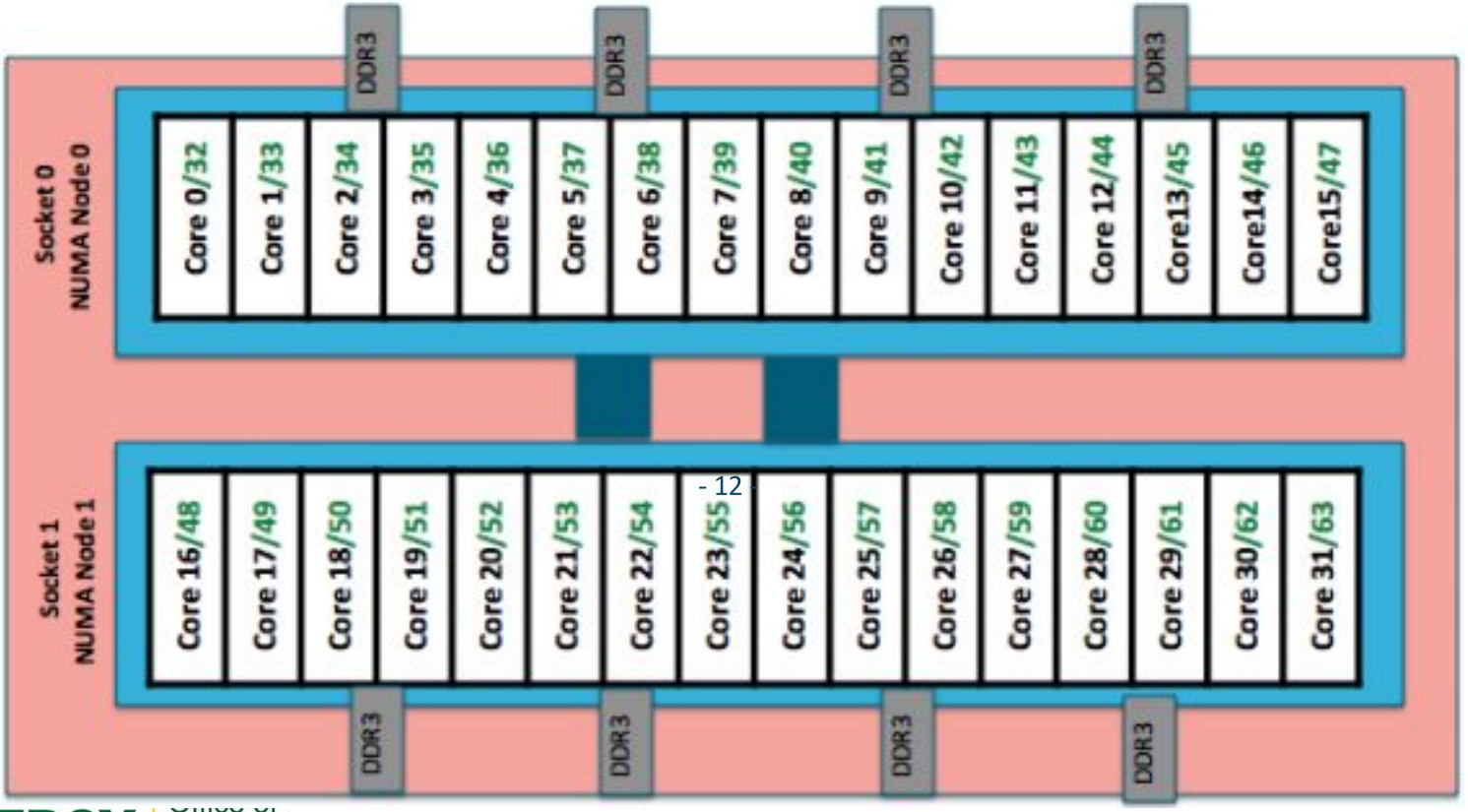
BERKELEY LAB
Lawrence Berkeley National Laboratory

Process and Thread Affinity

- **Process affinity (or CPU pinning): binds MPI process to a CPU or a range of CPUs on the node**
- **Important to spread MPI ranks evenly onto different NUMA nodes**
- **Thread affinity: forces each process or thread to run on a specific subset of processors, to take advantage of local process state**
- **Correct process and thread affinity is the basis for getting optimal performance**

Cori Haswell Compute Node

2 sockets per node,
16 physical cores (32 logical CPUs) per socket
(in SLURM: each hyperthread is a logical CPU)



Original Configuration

- **In SLURM: TaskPlugin = task/cgroup,task/cray**
- **We choose this for simplicity of usage for Cori Phase 1**
  - Only need to specify number of nodes via #SBATCH
  - Request number of MPI tasks and threads per task via srun -n, -c
  - Set OMP_NUM_THREADS via run time env
  - No other srun flags needed
  - Tasks and threads are distributed among nodes and sockets, with correct thread affinity
  - SLURM decides to use hyperthreads or not automatically
  - Meets most users need
  - However not easy to customize and no memory binding

Original Way: Hybrid MPI/OpenMP Example

2 nodes, 8 MPI tasks, 4 per node, 2 per socket,
2 OpenMP threads per task

```
yunhe> salloc -N 2 -p debug -t 30:00
yunhe> export OMP_NUM_THREADS=2
yunhe> srun -n 8 -c 2 ./xthi | sort -k4n,6n
Hello from rank 0, thread 0, on nid00102. (core affinity = 0,1,32,33)
Hello from rank 0, thread 1, on nid00102. (core affinity = 0,1,32,33)
Hello from rank 1, thread 0, on nid00102. (core affinity = 16,17,48,49)
Hello from rank 1, thread 1, on nid00102. (core affinity = 16,17,48,49)
Hello from rank 2, thread 0, on nid00102. (core affinity = 2,3,34,35)
Hello from rank 2, thread 1, on nid00102. (core affinity = 2,3,34,35)
Hello from rank 3, thread 0, on nid00102. (core affinity = 18,19,50,51)
Hello from rank 3, thread 1, on nid00102. (core affinity = 18,19,50,51)
Hello from rank 4, thread 0, on nid00114. (core affinity = 0,1,32,33)
Hello from rank 4, thread 1, on nid00114. (core affinity = 0,1,32,33)
…
Hello from rank 7, thread 0, on nid00114. (core affinity = 18,19,50,51)
Hello from rank 7, thread 1, on nid00114. (core affinity = 18,19,50,51)
```

New Configuration

- **In SLURM: TaskPlugin = task/affinity,task/cgroup,task/cray**
  - Task/affinity enables more srun flags, such as --cpu_bind, --mem_bind, and cpu allocation over sockets
  - The job launching mechanism is now more complicated than with the "original" mechanism to achieve optimal affinity binding
- **New configuration is required for KNL**
  - For job launching scalability
  - Helps to use high bandwidth memory (HBM) with memory binding
- **Will be adopted for Cori Haswell and KNL with unified SLURM configuration when the merged Cori is back to users (also later on Edison)**

New Way: Usage

- **Basically, you will need to:**
  - Biggest change from the original way: Use the "-c" flag for srun and set it to # logical cores per MPI task on each node.
    - Total number of logical cores per Haswell node is 64
    - So the -c value is usually set to 64/#MPI_per_node
  - Also use "--cpu_bind=cores" *when #MPI_per_node is not a divisor of 64.*
    - the -c value should be set to 2 * int [32/#MPI_per_node] in this case
  - Not due to new SLURM configuration, but we also recommend to use OpenMP4 settings (OMP_PROC_BIND and OMP_PLACES) to fine tune thread affinity
- **Final recommendations will be ready when Cori is back**

New Way: Hybrid MPI/OpenMP Example

2 nodes, 8 MPI tasks, 4 per node, 2 per socket,
2 OpenMP threads per task

```
yunhe> salloc -N 2 -p debug -t 30:00
yunhe> export OMP_NUM_THREADS=2
# -c is set to  64/#MPI_per_node: each MPI task has 16 logical cores (8 physical cores)
yunhe> srun -n 8 -c 16 ./xthi |sort -k4n,6n
Hello from rank 0, thread 0, on nid00021. (core affinity = 0-7,32-39)
Hello from rank 0, thread 1, on nid00021. (core affinity = 0-7,32-39)
Hello from rank 1, thread 0, on nid00021. (core affinity = 16-23,48-55)
Hello from rank 1, thread 1, on nid00021. (core affinity = 16-23,48-55)
Hello from rank 2, thread 0, on nid00021. (core affinity = 8-15,40-47)
Hello from rank 2, thread 1, on nid00021. (core affinity = 8-15,40-47)
…
Hello from rank 7, thread 0, on nid00022. (core affinity = 24-31,56-63)
Hello from rank 7, thread 1, on nid00022. (core affinity = 24-31,56-63)
```

New Way: Hybrid MPI/OpenMP Example (cont'd)

Continue session from last slide …

```
yunhe> export OMP_PROC_BIND=spread
yunhe> export OMP_PLACES=threads
yunhe> > srun -n 8 -c 16 ./xthi |sort -k4n,6n
Hello from rank 0, thread 0, on nid00021. (core affinity = 0)
Hello from rank 0, thread 1, on nid00021. (core affinity = 4)
Hello from rank 1, thread 0, on nid00021. (core affinity = 16)
Hello from rank 1, thread 1, on nid00021. (core affinity = 20)
Hello from rank 2, thread 0, on nid00021. (core affinity = 8)
Hello from rank 2, thread 1, on nid00021. (core affinity = 12)
…
Hello from rank 5, thread 0, on nid00022. (core affinity = 16)
Hello from rank 5, thread 1, on nid00022. (core affinity = 20)
Hello from rank 6, thread 0, on nid00022. (core affinity = 8)
Hello from rank 6, thread 1, on nid00022. (core affinity = 12)
Hello from rank 7, thread 0, on nid00022. (core affinity = 24)
Hello from rank 7, thread 1, on nid00022. (core affinity = 28)
```

Available Methods to Check Affinity

- **We will provide pre-built binaries of a small test code (xthi.c) for users to check affinity.**
- **Use srun flag: --cpu_bind=verbose**
- **At run time**
  - Intel compiler: export KMP_AFFINITY=verbose
  - CCE compiler: export CRAY_OMP_CHECK_AFFINITY=TRUE

Key Messages

- **Users will need to start specifying affinity explicitly in job scripts on Cori**
- **There will be differences in usage between Edison and Cori initially**
  - Edison will NOT switch to new mechanism until OS upgrade to CLE6 in early 2017
- **We will provide a "thread binding advisor" web interface to generate sample batch scripts for Edison, Cori Haswell, and KNL**
- **Web pages with more details will be ready before Cori is back**