NERSC Users Group
Monthly Meeting
April 27, 2017
1. Announcements and Outages
   a. May 16-17 Quarterly Maintenance
2. Cori update (Doug Jacobsen)
3. Cori KNL Node Charging
4. Backup your files! (Lisa Gerhardt)
5. Burst Buffer update (Wahid Bhimji)
6. Using Machine Learning on Cori (Evan Racah)
NERSC systems will be unavailable for a quarterly maintenance on May 16 beginning at 6 am on Tuesday, May 16. The work being performed includes system patches, a filesystem check, data-transfer node hardware work, and SGN patches. All systems except Cori will be returned to users at the end of the day.

All Cori compute nodes will be reserved for a 24-hour full-system test after the maintenance is completed. The full-system reservation will conclude at the end of the day on Wednesday, May 17. During this test period Cori scratch (CSCRATCH) will be accessible from other systems.
Cori Status Update

Doug Jacobsen
Cori System Status

Since last NUG (3/23):

● Added new haswell nodes (384)
  ○ Cori now has all planned hardware components integrated
● Operating system updated (CLE 6.0up3)
● KNL firmware updated then reverted
  ○ Update correlated to decreased stability of KNL nodes
● Slurm updated (17.02.2)
● Numerous other patches in shorter scheduled maintenances
● Completed the bulk of the planned system-dedicated testing
Cori System Status - Upcoming

Within the next month:

- At least two minor maintenances (< 8 hours) to install patches, reboot system
- One larger maintenance for cscratch1, new KNL BIOS, and system-dedicated benchmarking time (roughly 24 hours)

In June:

- OS upgrade to CLE6.0up04
Cori KNL Node Charging
Cori KNL Charging Begins July 1

- DOE program managers will distribute 2.4B additional NERSC Hours in June for use on KNL beginning July 1
- No new ERCAP request needed (but not prohibited)
- DOE will make decisions based on science priorities and code readiness
  - NERSC will send to DOE KNL usage report and list of codes known to run well on KNL
  - There may be a short questionnaire for you to request hours (more info to come)
Backup your Files!
Cori and Edison Scratch is Purged

• File are automatically purged if they are not accessed
  – In 8 weeks on Edison scratch
  – In 12 weeks on Cori scratch

• Please back up your important files
  – To project
  – To HPSS
  – To another site
Tips for Optimal Transfers to HPSS

• Use recommended striping settings for files on scratch

http://www.nersc.gov/users/storage-and-file-systems/i-o-resources-for-scientific-applications/optimizing-io-performance-for-lustre/
Tips Con’t

• Optimal bundle size is several hundreds of GBs
  – Bundle files together with htar
  – Or regular tar

• Use the xfer queue to parallelize transfers
  – Can run up to 15 simultaneous transfers
    #SBATCH –M escori
    #SBATCH –p xfer
    #SBATCH –t 10:00:00
Burst Buffer Update

Wahid Bhimji
Burst Buffer Overview

• Burst Buffer is a layer of NVRAM that sits inside the Aries high-speed interconnect
• Currently 1.8 PB on 288 Burst Buffer Nodes
• Allows apps to accelerate I/O significantly

### Peak Bandwidth

<table>
<thead>
<tr>
<th></th>
<th>Read</th>
<th>Write</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.7 TB/s</td>
<td>1.6 TB/s</td>
</tr>
</tbody>
</table>

### Peak IOPS

<table>
<thead>
<tr>
<th></th>
<th>Read</th>
<th>Write</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>28 M</td>
<td>13 M</td>
</tr>
</tbody>
</table>
NUG Talk with lots of background on Burst Buffer:

Tutorials and example batch scripts
http://www.nersc.gov/users/computational-systems/cori/burst-buffer/example-batch-scripts/
Usage (most basic example)

- Open to all users - default allocation ~50T
- Basic usage is very easy - e.g.
  - In batch script: 
    #DW jobdw capacity=200GB access_mode=striped type=scratch
  - Filesystem mounted on your compute nodes - pointed to by $DW_JOB_STRIPED: write your output there
- Filesystem destroyed at end of job. Copy output either with 'cp' or stage_out (later works even if exe hits time limit)

  #DW stage_out source=$DW_JOB_STRIPED/dirname
destination=/global/cscratch1/sd/username/path/to/dirname type=directory

- stage_in for inputs and persistent res for chaining jobs
Current status

- Basic operation is easy - and many more examples on web including ‘persistent’ reservations, and API
- Most people see IO improvement over Lustre
- Features such as ‘transparent caching’ still in progress
- It has quirks - for example right now for ‘stage-in’ must specify output filename not just a directory - ie.:

  [ ... ] destination=$DW_JOB_STRIPED/file.txt type=file NOT  [...] destination=$DW_JOB_STRIPED type=file

- Contact us if you have problems. Keep an eye on [http://www.nersc.gov/users/computational-systems/cori/burst-buffer/known-issues/](http://www.nersc.gov/users/computational-systems/cori/burst-buffer/known-issues/)
Using Machine Learning on Cori

Evan Racah
Available Tools -> General Machine Learning

- **learn**
  - great for non-image based machine learning
  - easy to use
  - support for wide range of algorithms

- **Spark**
  - Multinode
  - great for data parallel operations
  - relatively easy to use
  - support for only a subset of ML algorithms

- **XGBoost**
  - Great for gradient-boosted decision tree type algorithms like those used in Kaggle
Available Tools -> Deep Learning

- **TensorFlow**
  - Python interface -> ease of use and flexibility
  - requires a lot of coding
  - large, growing community
  - some *multi-node support*

- **Caffe**
  - Config file-based, but some python bindings
  - High performance, multinode IntelCaffe version available
  - Harder to use for non-standard data formats/algorithms

- **Keras**
  - wrapper around TensorFlow for ease of use

- **Others:** PyTorch, Theano, Lasagne
How Do I Use These Tools? -> Command Line

Deep Learning Module

- Python-based tools available
- Just one module load call
- Includes access to:
  - TensorFlow, Keras, PyTorch, Theano, XGBoost, Lasagne, scikit-learn

Scikit-Learn - Also available in standard python module

Caffe Module

Spark Module

How Do I Use These Tools? -> Notebook

Steps to use:

1. Enter “module load deeplearning” in a Cori shell and logout if you have never done so
2. Go to jupyter.nersc.gov or jupyter-dev.nersc.gov
3. Open your desired notebook
4. Select Kernel->Change kernel->deeplearning
5. Enjoy!