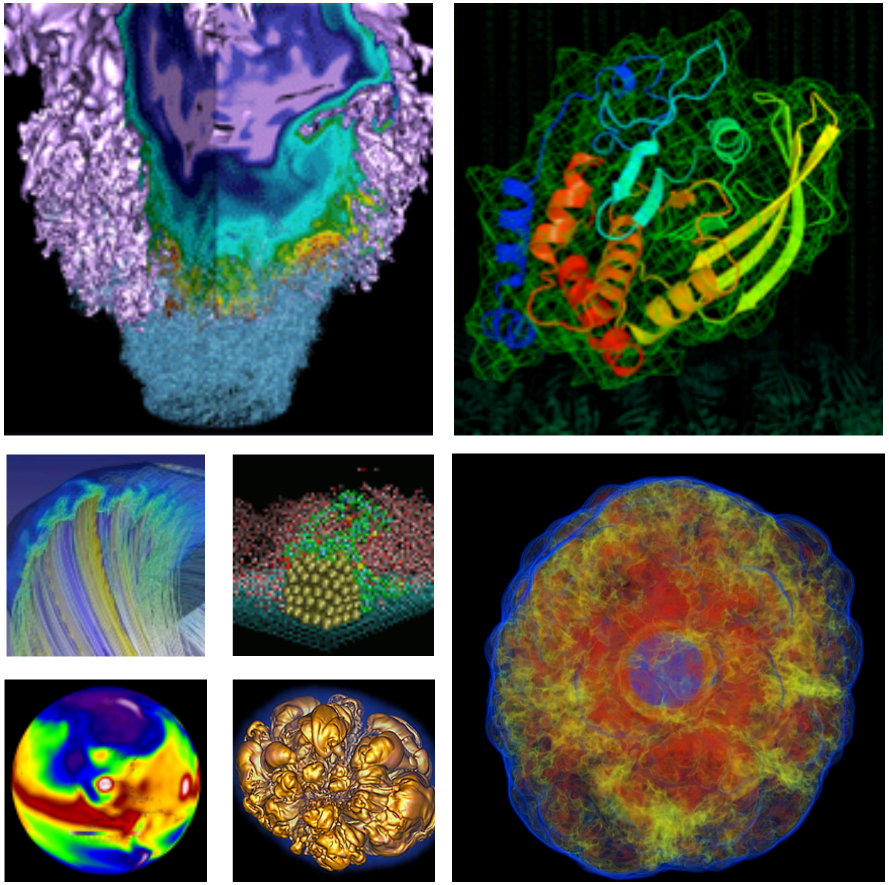
Running HEP Workloads on the NERSC HPC Systems



T. Quan, J. Botts, L. Gerhardt, D. Jacobsen, D. Paul, S. Canon, W. Bhimji, D. Bard, T. Declerck

October 21, 2016

HEP has different requirements than traditional HPC environments

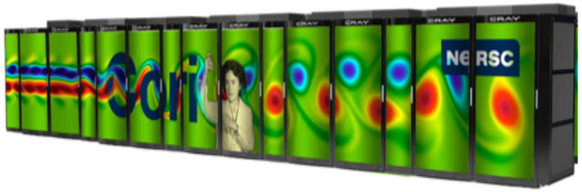
- **Stable, static execution environment**
  - NERSC Shifter allows docker images to be used on HPC clusters
- **Very large, very challenging I/O**
  - Cori Burst Buffer provides NVRAM for intermediate storage later within cluster HSN
- **Flexible, high performance networking between external HEP instruments and compute nodes**
  - NERSC is configuring Software Defined Networking for on-demand network performance

HPC Computing at NERSC

- **Phase 1 Cori (completed) is aimed at data intensive computing**
  - HPC system from Cray: 1630 Haswell nodes, each w/ 32 cores and 128 GB memory
  - Lustre File system
    - 28 PB capacity, >700 GB/sec peak performance
  - NVRAM "Burst Buffer" for I/O acceleration
    - ~1.5PB capacity, ~1TB/s (half with Phase 1)
  - Outbound connections allowed from compute nodes
  - Queue structure friendly to real-time data ingestion/ analysis and long-running and data-intensive workloads
- **Phase 2 Cori: NERSC-8, Cori, Cray XC40 is being installed now**
  - 9,300 Knights Landing Compute nodes (72 cores each) Global GPFS file system for long term file retention and sharing
- **Cray Aries high-speed "dragonfly" topology interconnect**

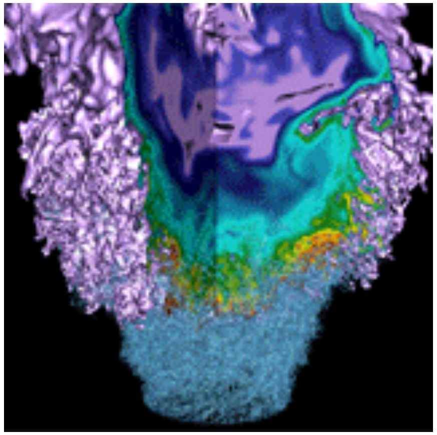
Cori Phase I Data Features

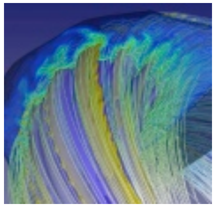
- **Cori Phase 1 has many features designed to support data-intensive computing**
- User-defined images/Shifter
- Burst Buffer for high bandwidth, low latency I/O
- Software Defined Networking for high bandwidth transfers in and out of the compute node with Large number of login/interactive nodes to support applications with advanced workflows
  - Used for Spark, JupyterHub and experiment specific workflows (e.g. the ATLAS LHC experiment)

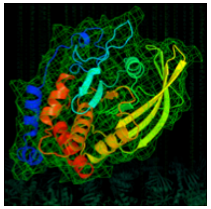


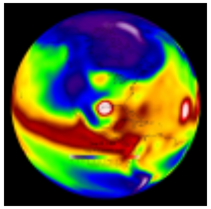
- Flexible queues with SLURM
  - Immediate access (realtime) queues for jobs requiring real-time data ingestion or analysis
  - High throughput and serial (shared) queues can handle a large number of jobs
- Improved outbound Internet connections to communicate with the outside world. (e.g. to access a database in another center.) via RSIP
- High-performance Lustre Filesystem
- Large amount of memory per node (128 GB/node) as well as high-memory nodes (775GB/node) accessed via a separate queue.

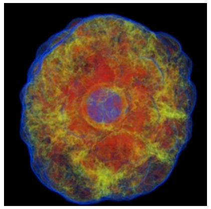
Providing a static execution environment with Shifter

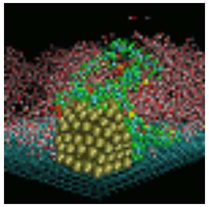












See also Cray Users Group Paper for more use-cases beyond HEP

HPC Computing at NERSC

- **HEP workflow and reproducibility requires a static software environment.**
  - Cray compute node has stripped down SLES 12 environment
- **HEP software stacks are often complicated**
  - Many dependencies and difficult to compile on many different systems

Deploying containers to HPC

- **Outside of HPC (in the cloud), Docker provides consistent, portable execution environments**
  - Wraps up software and filesystem into a package that will run the same, wherever it runs
  - Done via chroot
- **NERSC is enabling Docker-like container technology on its systems through a new software package known as Shifter**

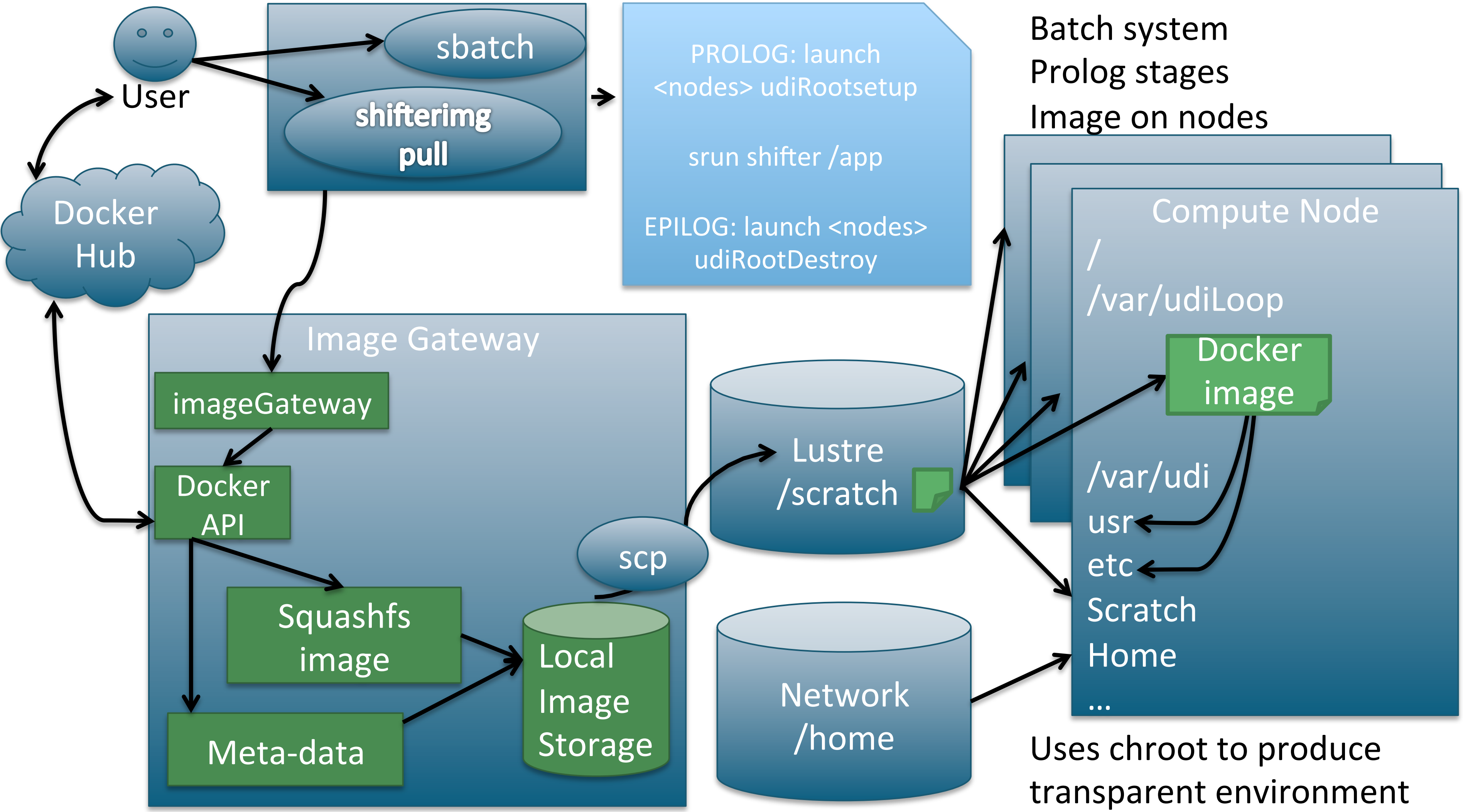
Shifter at NERSC

Work of Doug Jacobsen at NERSC

- **Secure and scalable way to deliver containers to HPC**
- **Deployed on Edison and on Cori**
- **Supports Docker images and other images (vmware, ext4, squashfs, etc.)**
- **Basic Idea**
  - Convert from native image format to common format
  - Chroot using common image on compute nodes

https://www.nersc.gov/research-and-development/user-defined-images/

How Shifter Works



PROLOG: launch <nodes> udiRootsetup

srun shifter /app

EPILOG: launch <nodes> udiRootDestroy

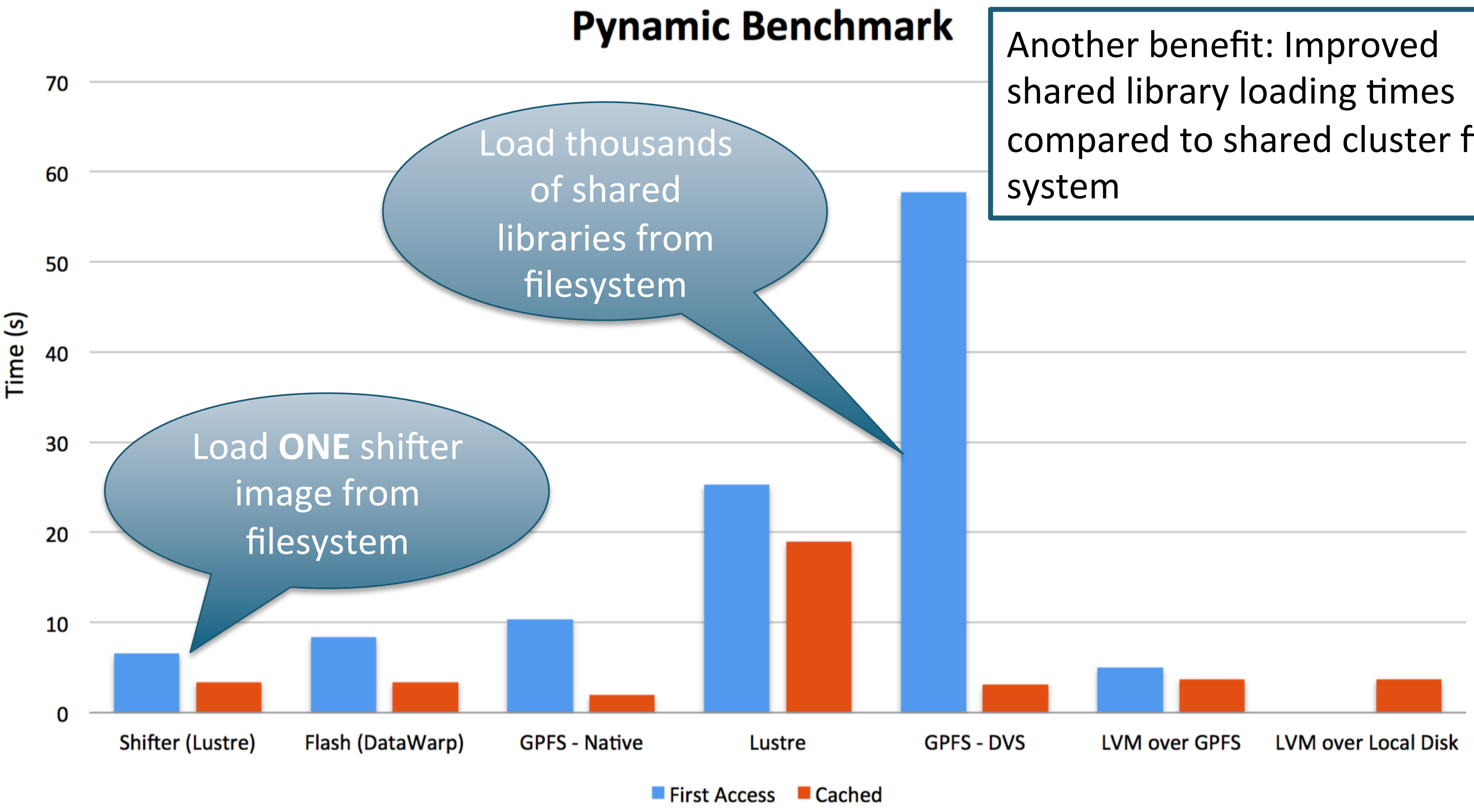
Batch system
Prolog stages
Image on nodes

Uses chroot to produce transparent environment for user's job

Thanks Shane C. and Doug J. (LBNL)

Shifter is Fast

Another benefit: Improved shared library loading times compared to shared cluster file system



Pynamic Benchmark

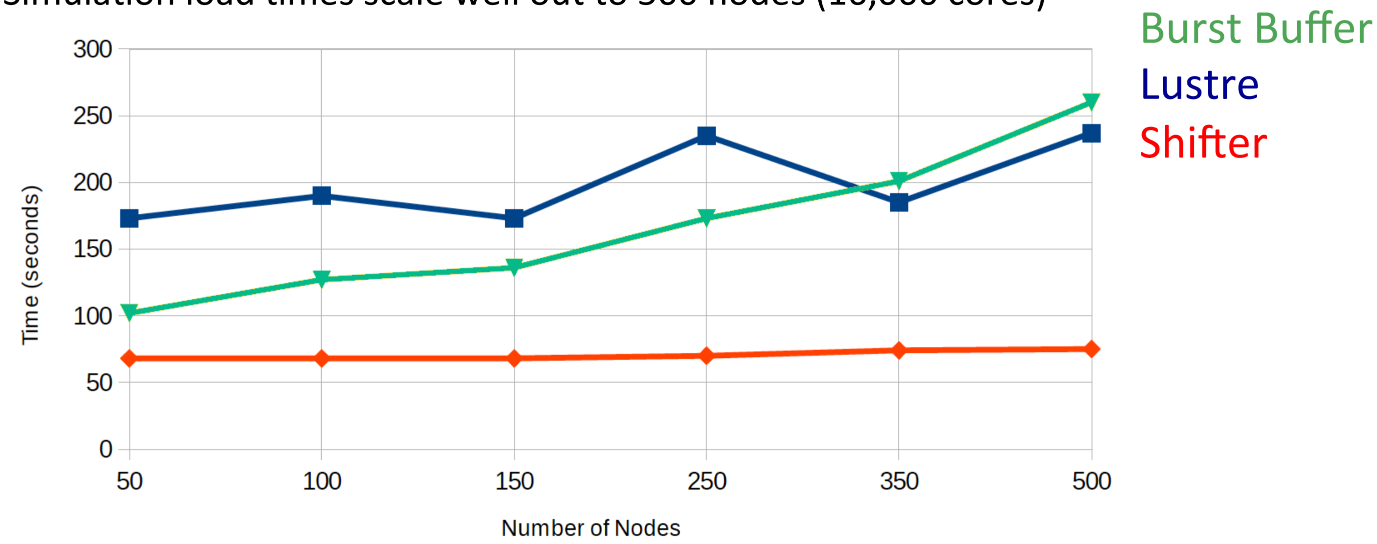
Load thousands of shared libraries from filesystem

Load **ONE** shifter image from filesystem

Thanks Rollin Thomas (LBNL)

Running a Large Shifter Image

- **Faster than running from filesystem, independent of node count**
- **As proof of concept created “Mega” CVMFS shifter image**
  - Full CVMFS stack pulled down and deduped with uncvmfs software stack. 1 – 3 TB ext4 file uncompressed, 300 GB compressed w/squashfs
- **Use Shifter to load job**
  - Add a single flag to batch script “--image=<image name>”
  - ATLAS cvmfs repository is found at /cvmfs/atlas.cern.ch like normal
- **Tested with ATLAS G4 simulations and Analysis Software (QuickAna)**
  - Simulation load times scale well out to 500 nodes (16,000 cores)

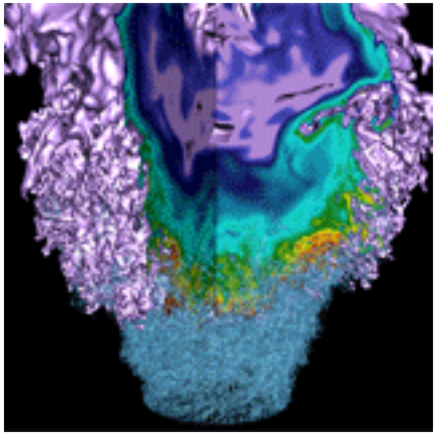


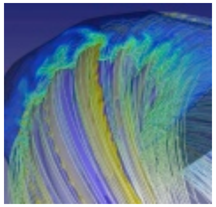
Thanks Vakho Tsulaia et al. (LBNL)

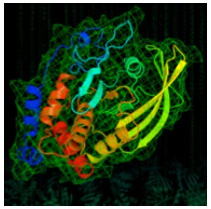
Shifter General Perspectives

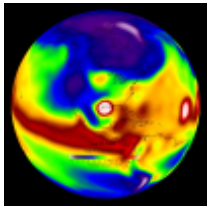
- **Shifter has been approved to be released as open source through a BSD license**
  - The intent is that others can download it and use it at their centers
- **Cray made a product out of Shifter to provide mainstream capability for Cray systems**
- **Contact Doug Jacobsen or Shane Canon (the author of Shifter) if you are interested in collaborating on this project**
- **Contact Lisa Gerhardt if you are interested in running a Shifter CVMFS image at NERSC**
- **Shifter-hpc google group for those interested in installing the framework on their systems**

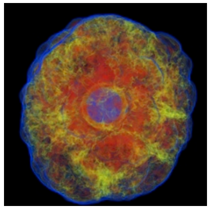
Extreme I/O on HPC for HEP using the Burst Buffer at NERSC

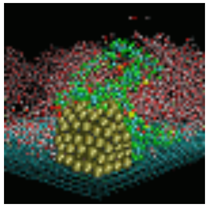










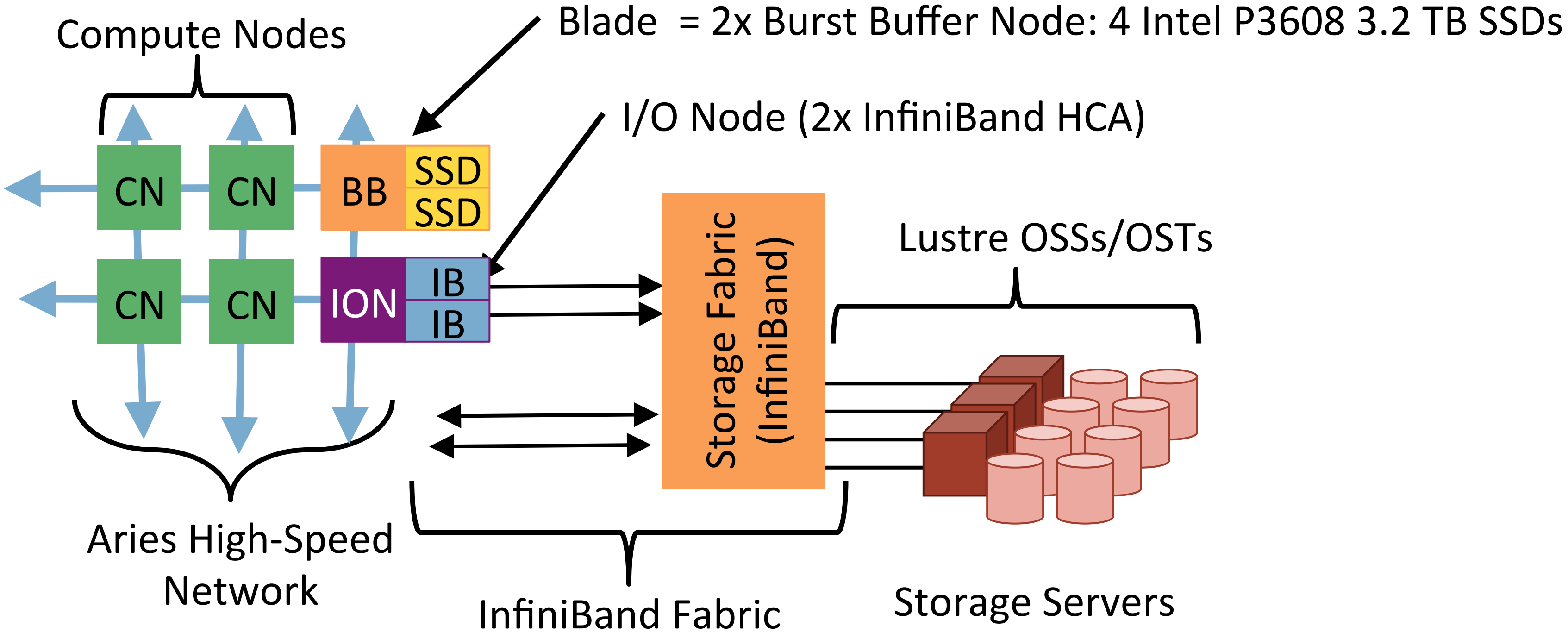


See also Cray Users Group Paper for more use-cases beyond HEP

Burst Buffer Concepts

- **HDD performance not increasing sufficiently**
  - HPC centers buying large capacity parallel filesystems to get bandwidth
  - Huge POSIX filesystems don't scale
  - Actual bandwidth demands comes in 'bursts'
  - For bandwidth SSD is cheaper than HDD
- **Some applications (including experimental HEP) have I/O patterns that better match SSD than disk**
- **Use NVRAM-based 'Burst Buffer' (BB) as intermediate layer**
  - Handle I/O bandwidth spikes without needing a huge PFS
  - Underlying media supports challenging I/O
  - Software for filesystems- on-demand - scales better than large POSIX PFS
  - Staging to PFS asynchronously
- **Cori Burst Buffer (Phase 1) 920TB on 144 BB nodes**
  - Now being doubled for Phase 2

Nersc Burst Buffer Architecture



Blade = 2x Burst Buffer Node: 4 Intel P3608 3.2 TB SSDs

I/O Node (2x InfiniBand HCA)

Compute Nodes

CN CN CN CN

BB SSD SSD

ION IB IB

Storage Fabric (InfiniBand)

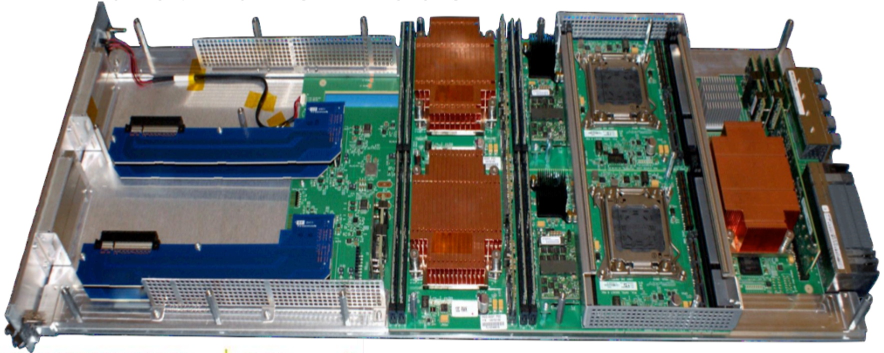
Lustre OSSs/OSTs

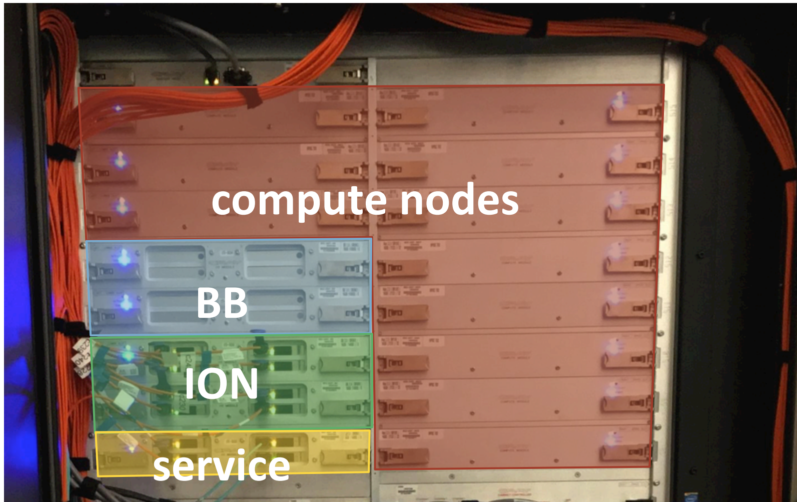
Aries High-Speed Network

InfiniBand Fabric

Storage Servers

Burst Buffer Blade:

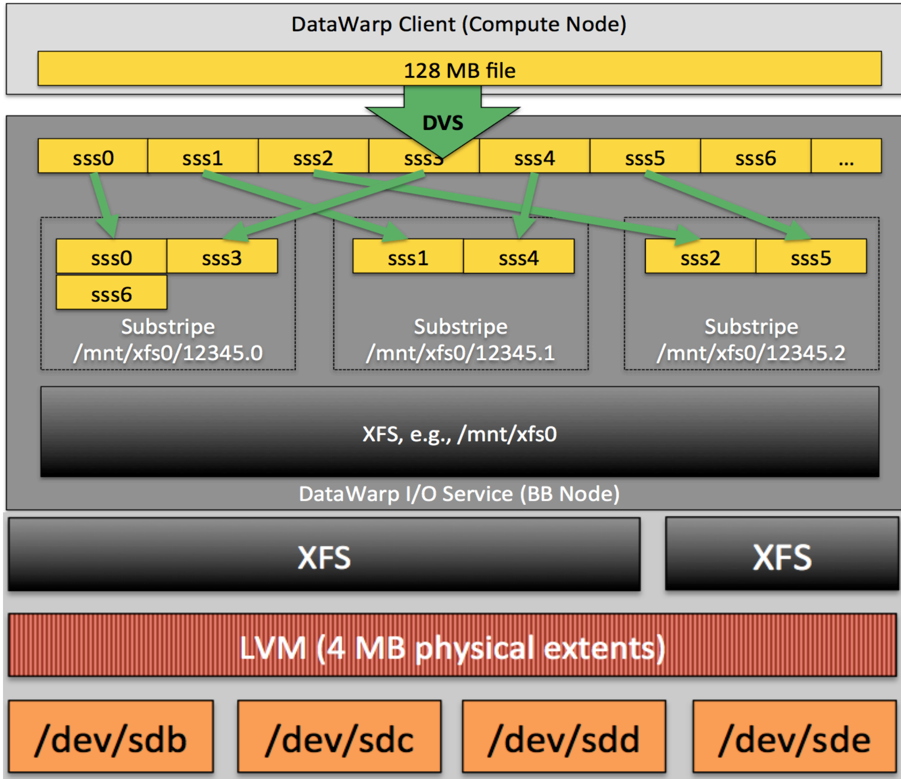




How it works

- Cray DataWarp Software:
  - Works with Slurm
  - Users add directives to job script
  - BB space reserved, files staged in, while job still in queue
  - Presents complex layers of hardware to the user as a normal POSIX filesystem

```
#!/bin/bash
#SBATCH —p regular —N 10 —t 00:10:00
#DW jobdw capacity=1000GB access_mode=striped
                type=scratch
#DW stage_in source=/lustre/input.dat
 destination=$DW_JOB_STRIPED/inputs type=file
#DW stage_out source=$DW_JOB_STRIPED/outputs
   destination=/lustre/outputs type=directory
srun my.x --infile=$DW_JOB_STRIPED/input.dat
  --outdir=$DW_JOB_STRIPED/outputs
```



ATLAS and ALICE

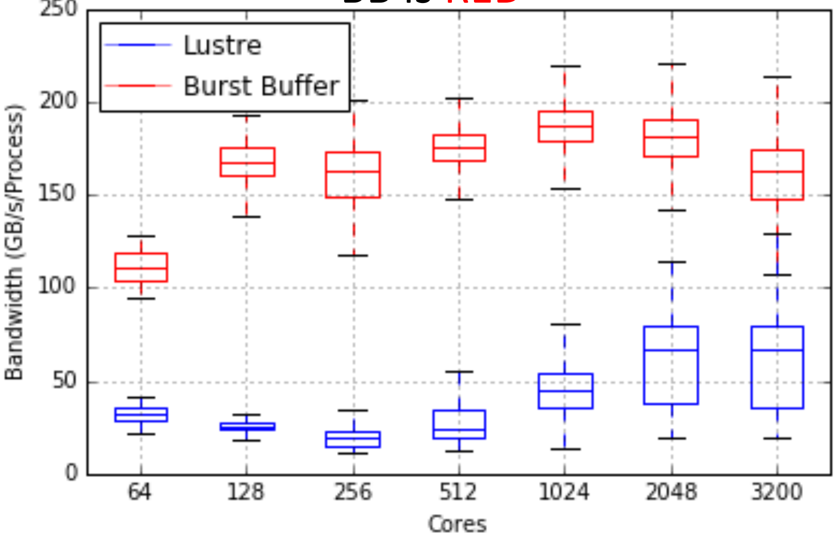
Markus Fasel, Jeff Porter

- Performance vs core count for ATLAS and ALICE
  - **Higher is better**

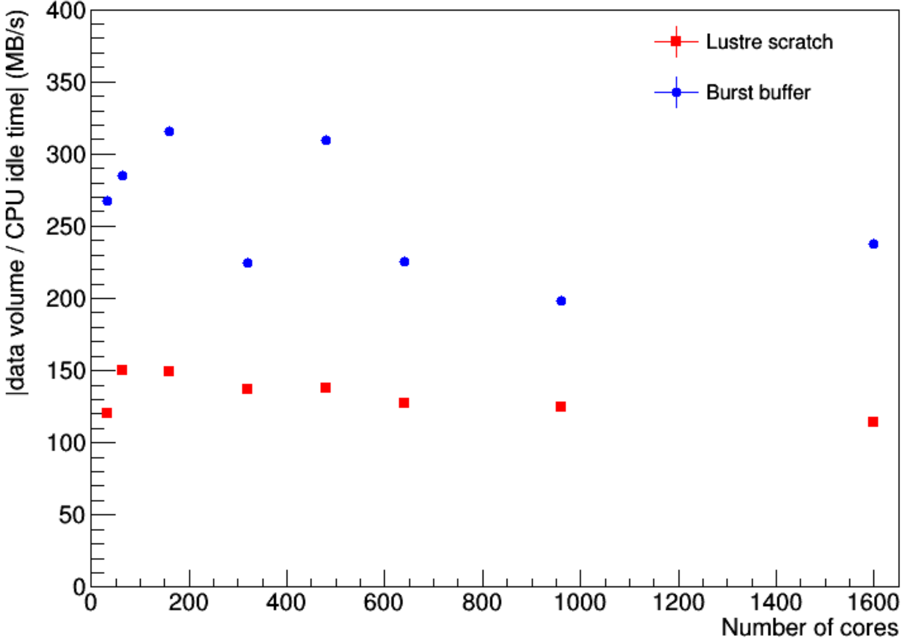
ATLAS

ALICE

BB is RED



BB is BLUE



https://indico.cern.ch/event/505613/contributions/2227423/

Coming performance improvements

1. DVS client-side caching (faster re-reads)

- Lustre has client-side caching, currently DVS (used for BB) does not
- Essential for small sequential Read/Write transfers and re-reads
- Expected later this year

2. Smaller granularity (more flexible)

- Amount of space allocated on each BB node
  - Previously couldn't be less that 200G
  - Users had to request larger space to get striped performance
  - Now have the ability to reduce this – testing lower values

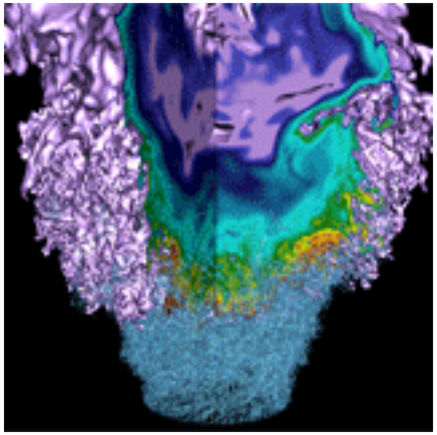
3. Transparent caching (BB as invisible cache layer for Lustre)

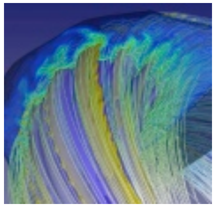
- Allows user to specify directory in Lustre and blocks are cached as used
- Software now available - but undergoing testing

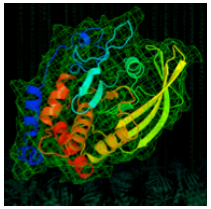
4. Twice as much Burst Buffer! (and therefore ~2x bandwidth!)

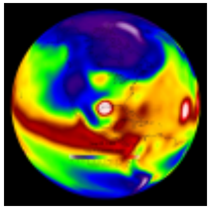
We're working with Cray to improve BB performance out-of-the-box and for all use cases

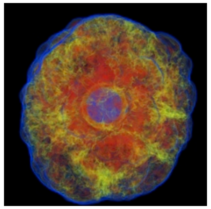
Cori Gateway Nodes (a.k.a SDN)

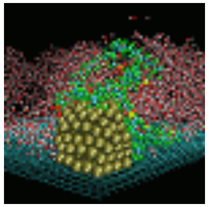












See also Cray Users Group Paper for more use-cases beyond HEP

The Long Term Vision

- A scientist ***somewhere***
- Runs a job on compute nodes at NERSC
- To analyze the output from an instrument ***somewhere***
  - As if they were ***all on the same network***

- We use the SLURM batch system and Software Defined Networking to route network traffic smoothly between an external site and specific compute nodes
  - On a job-by-job basis

Why?

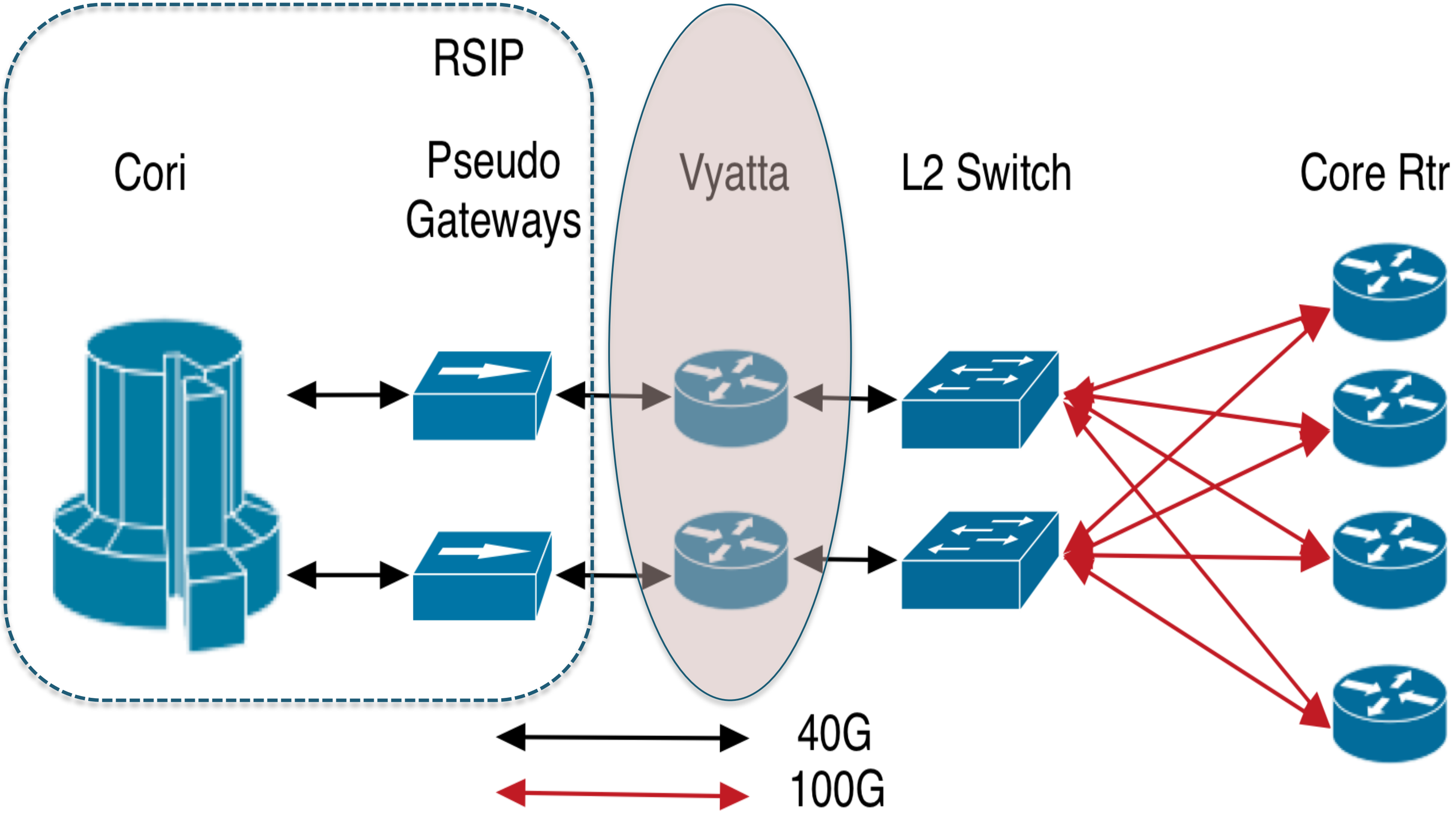
- **Supporting data-intensive use cases requires a new class of capabilities not traditionally important for HPC systems.**
  - **Compute nodes must be able to access external services and ingest data at high-bandwidths and high connection rates.**
  - **Compute nodes must also be accessible by external systems (e.g. for streaming uses cases).**
  - **Bandwidth and access to compute nodes can be allocated based on job placement and user needs.**

Approach

- **Completed:**
  - Repurpose RSIP nodes into “Bridge” nodes to pass traffic from Aries to external Gateway nodes
  - Introduce External Gateway nodes running a Vyos/Vyatta OS (software-based router) to do routing

- **Future: Integrate it all with SLURM**

Cori Software Define Network



Early Testing

Initial Science Uses Cases

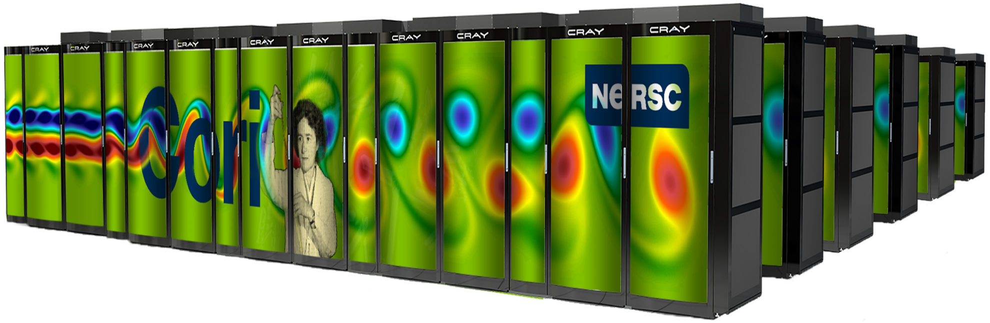
- General Atomics – 5x improvement talking to an external database used in a real-time workflow
- Globus-url-copy to CERN test point – 100x faster!
- LCLS to Cori now 100x faster

Note: Edison RSIP seems to perform much better. The large difference could be a symptom of RSIP configuration issues on Cori.

Conclusions - Shifter + Burst Buffer + SDN Bring HEP to NERSC

- **New framework and innovation are making running HEP workflows easy at NERSC**
  - Shifter framework can be extended to other Cray systems
  - Successful runs have been done with ATLAS, ALICE and CMS simulation and analysis jobs
  - Opportunity to run LHC jobs at NERSC at large scale
- **NERSC/Cray Burst Buffer offers new approach to dynamically allocate filesystems striped across high-performance SSDs**
  - Demonstrated here for experimental HEP Workflows
  - Substantially improves I/O over comparable Lustre filesystem
  - I/O is not (now) a significant barrier to these projects
- **Cori network upgrade provides SDN (software defined networking) interface to Esnet**
  - High speed external connectivity and data streaming

NERSC is Hiring!



We want exceptional individuals with

- Strong Systems Programming skills
- Deep understanding of Systems Architecture
- Interest in new and innovative Technology

You will

- Work on the largest systems anywhere in the world!
- Make an impact on how thousands of researchers use HPC systems!

For open positions see:

www.nersc.gov