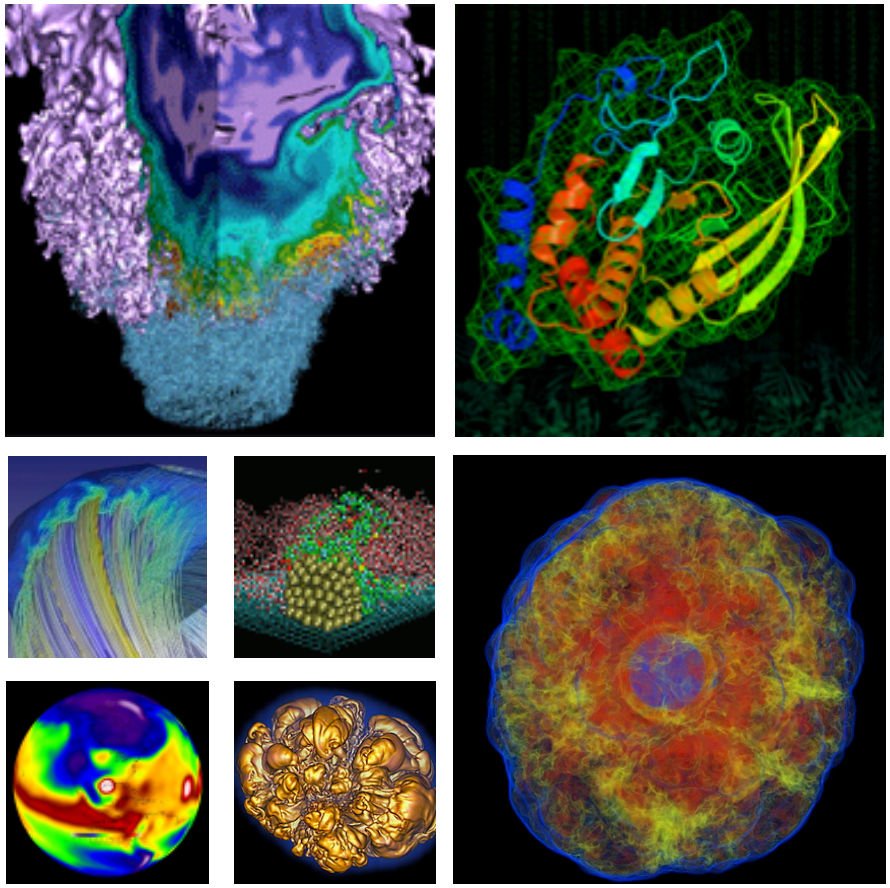
Running ATLAS, CMS, ALICE Workloads on the NERSC Cray XC30



L. Gerhardt, D. Jacobsen, J. Botts

October 12, 2015

HPC Computing at NERSC

- **NERSC-8, Cori, Cray XC40 is being installed now**
- **Phase 1 (coming ~1 month) 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: 9,300 Knights Landing Compute nodes (72 cores each) coming in 2016
- **Global GPFS file system for long term file retention and sharing**

CVMFS difficult in the HPC environment

- **To maximize the memory / node and reduce jitter, each compute node runs a highly optimized minimal environment**
  - Not a typical Linux environment
  - NO local disk
  - Root access only in special (rare) cases
- **Several methods were tried to deliver CVMFS, but none of them were feasible**
  - Required root permissions, file system issues (details in extra slide)
- **New method which leverages "Shifter" machinery to deliver user defined images to the Cray machines at NERSC**

Stuff that didn't work

- **Use CVMFS' alien cache to write to shared file system**
  - Requires root access and cvmfs client on each node, not usually done for a single type of workflow
- **NFS mount CVMFS install directory on compute nodes**
  - Requires root access and changes to compute node image, not usually done for a single type of workflow
- **Stratum-R: Install CVMFS on another node and rsync to shared file system**
  - At 20 M inodes Lustre file system performance starts to suffer
  - GPFS is delivered to computes through DVS gateway nodes, and is too slow when many small files are involved
  - ATLAS software must be retooled to find deal with new path

Deploying containers to HPC

- **Docker: open source, automated container deployment service**
  - Docker containers wrap up a piece of software in a complete filesystem that contains everything it needs to run (code, runtime, system tools and libraries)
  - Guaranteed to operate the same, regardless of the environment in which it is running
- **Docker-like container-based computing has yet to be fully recognized in HPC**
- **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**
- **Prototype implementation on Edison, full version 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

User loads image
"docker pull <image_name>"

Batch system
prolog stages
image on nodes

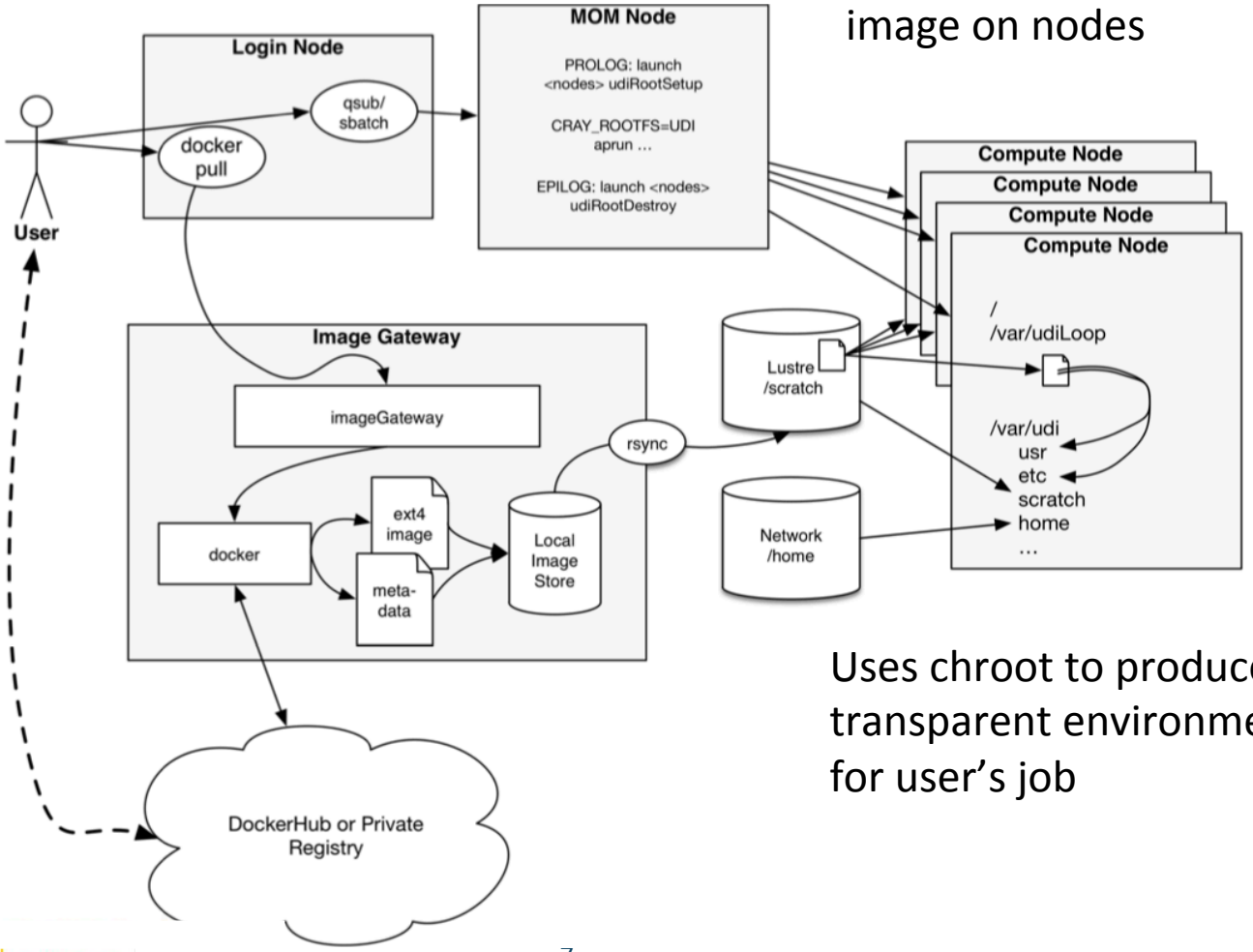
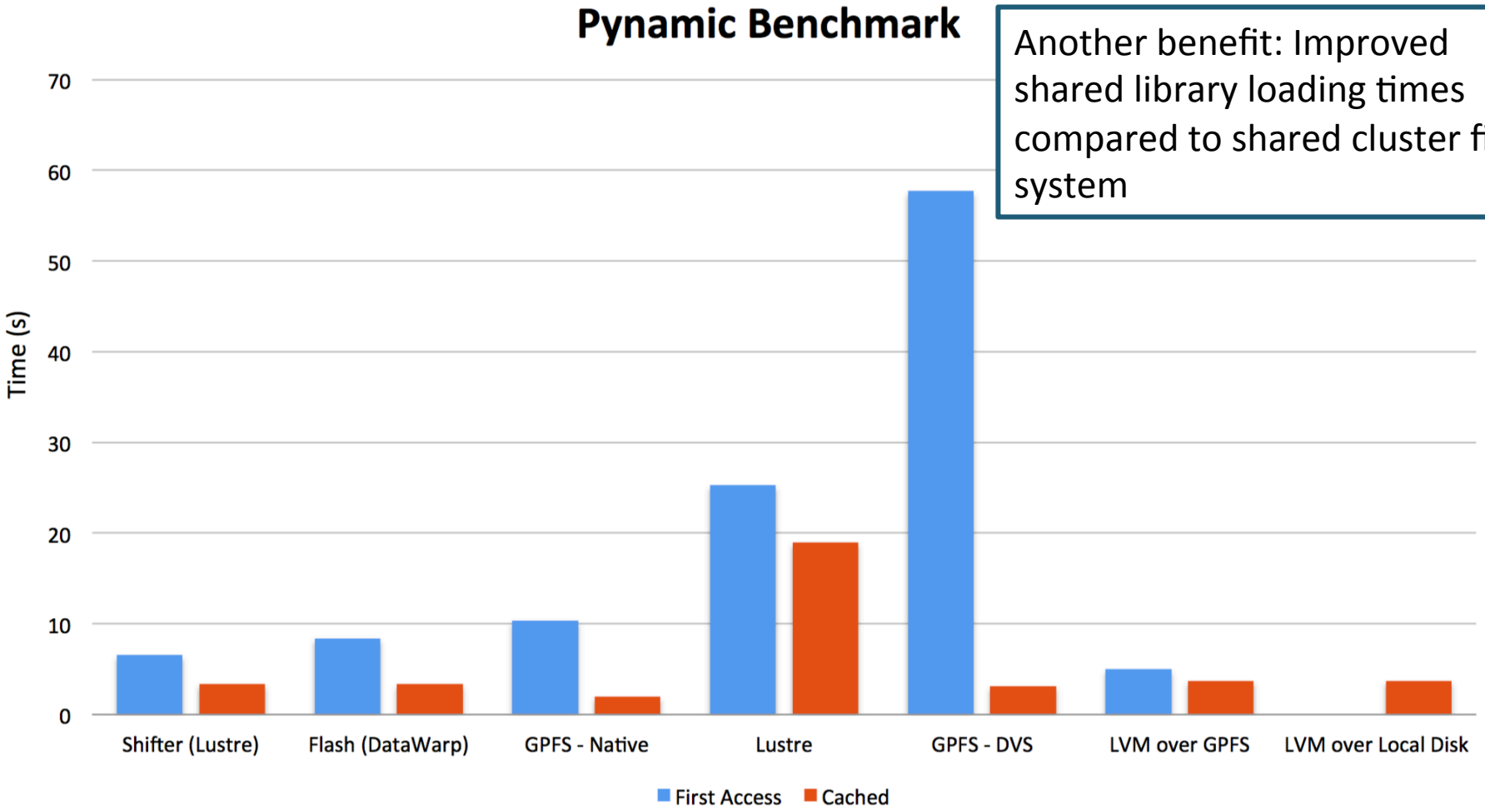


Image is stored
by gateway
service

Uses chroot to produce
transparent environment
for user's job

Shifter is Fast



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

Additional CVMFS Issues

- **One single Linux kernel for the whole node, so for security reasons it not possible to run images that require special kernel modules**
- **Unfortunately, this includes running CVMFS directly in the image**
  - FUSE, etc.
- **Get around this with a two-step process**

Getting CVMFS into Shifter

- **Install CVMFS on a standard Linux node**
- **Create an empty ext4 image file and mount it on the node**
- **Use cvmfs_snapshot to pull down full repository**
- **Rsync ATLAS CVMFS onto image**
  - Enormous!
  - > 3.5 TB and > 50 M inodes*

* Lower limits. I gave up at this point

Building the CVMFS Image Take 2

https://github.com/ic-hep/uncvmfs
By Simon Fayer

- **Use uncvmfs to dedupe files**
  - Python routines that crawls repository
  - Finds duplicate files and replaces them with hard links
  - Size dropped to 1.1 TB and 22M inodes
- **Convert ext4 image with to squashfs image**
  - Compresses data, inodes, and directories
  - Read only file system
  - Size dropped to 315 GB
- **Can make a fresh image ~daily**
  - CVMFS update ~2 hours
  - Squashfs conversion ~8 hours
  - Rsync into place ~1 hour

Running in the Real World

- **Use Shifter to load job**
  - Add a single flag to batch script "#PBS -v SHIFTER=<image name"
  - cvmfs repository is found at /cvmfs/<repo_name> like normal
- **Tested with ATLAS, ALICE, and CMS simulations out to 1000 nodes**
  - Load times scale with size of image, smaller is better
- **Testing with ATLAS analysis software underway**

Run Times at Different Scales

Number of Cores (24 cores / node)	Time to setup ATLAS environment (seconds)	Time to Initialize Threads (seconds)
24	32	102
240	11	98
2400	15	113
24000	24	256

All times for 313 GB sqashfs image.

Very good scaling of initialization time.
Ideal scaling of event processing time with number of cores.

Effect of the Image Size

- **Several images used for testing the performance of a 1000-node (24000-core) job**
  - **Large**: entire CVMFS software repo + big DB of detector conditions
  - **Medium**: Large – conditions
  - **Compact**: Only a few files

Image	Time to setup ATLAS environment (seconds)	Time to Initialize Threads (seconds)
Large ext4	280	600-650
Medium ext4	320	600
Medium squashfs	21	140-150
Compact ext4	66	240-270
Compact squashfs	24	150-200

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
- **There is a strong interest from Cray to make Shifter a mainstream capability for Cray systems**
- **Contact Doug Jacobsen (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 + Extras Bring CVMFS to NERSC

- **New framework and innovation are making running LHC 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
- **New Cori system will be friendlier to data intensive workflows**
- **Opportunity to run LHC jobs at NERSC at large scale**

Thank you! (And we are hiring!)