LUSTRE AT FNAL

Lustre Users Group Meeting
April 18, 2013

Alex Kulyavtsev

Fermilab

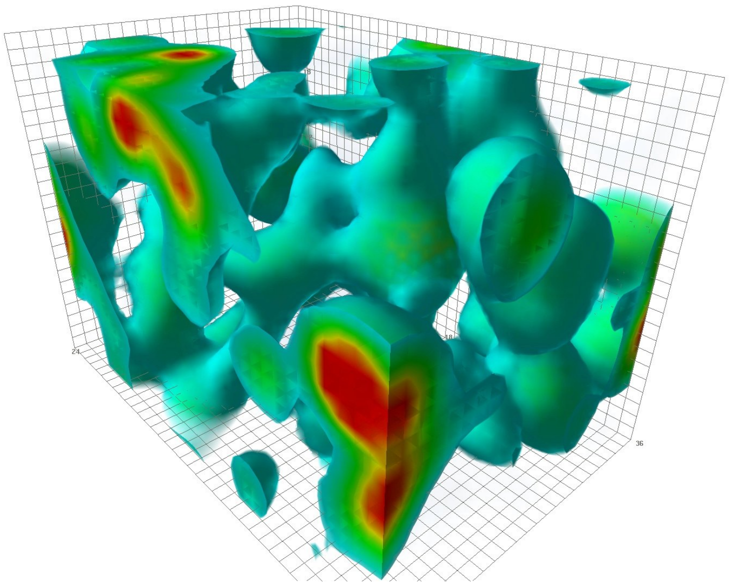
Outline: Lustre at FNAL

- Two largest lustre filesystems hosted at Fermilab are used for HPC computing:

  Lattice Quantum ChromoDynamic calculations and Cosmology calculations
- we discuss hardware configuration and experiences with lustre fs on these systems
- we shortly describe public lustre filesystem used for testing new Enstore tape system features and for kerbrized lustre WAN

Lustre for Lattice Quantum ChromoDynamics

- QCD is the theory of the strong force which binds quarks into nucleons (protons and neutrons) via interactions with gluons.
- The theory requires numerical simulation to make predictions and to compare with experiment
- The simulations are done on four-dimensional lattice and are called lattice QCD
- Otherwise it is typical HPC simulation task with different matrix and vector size and it uses complex numbers



'QCD Lava Lamp' courtesy Derek B. Leinweber

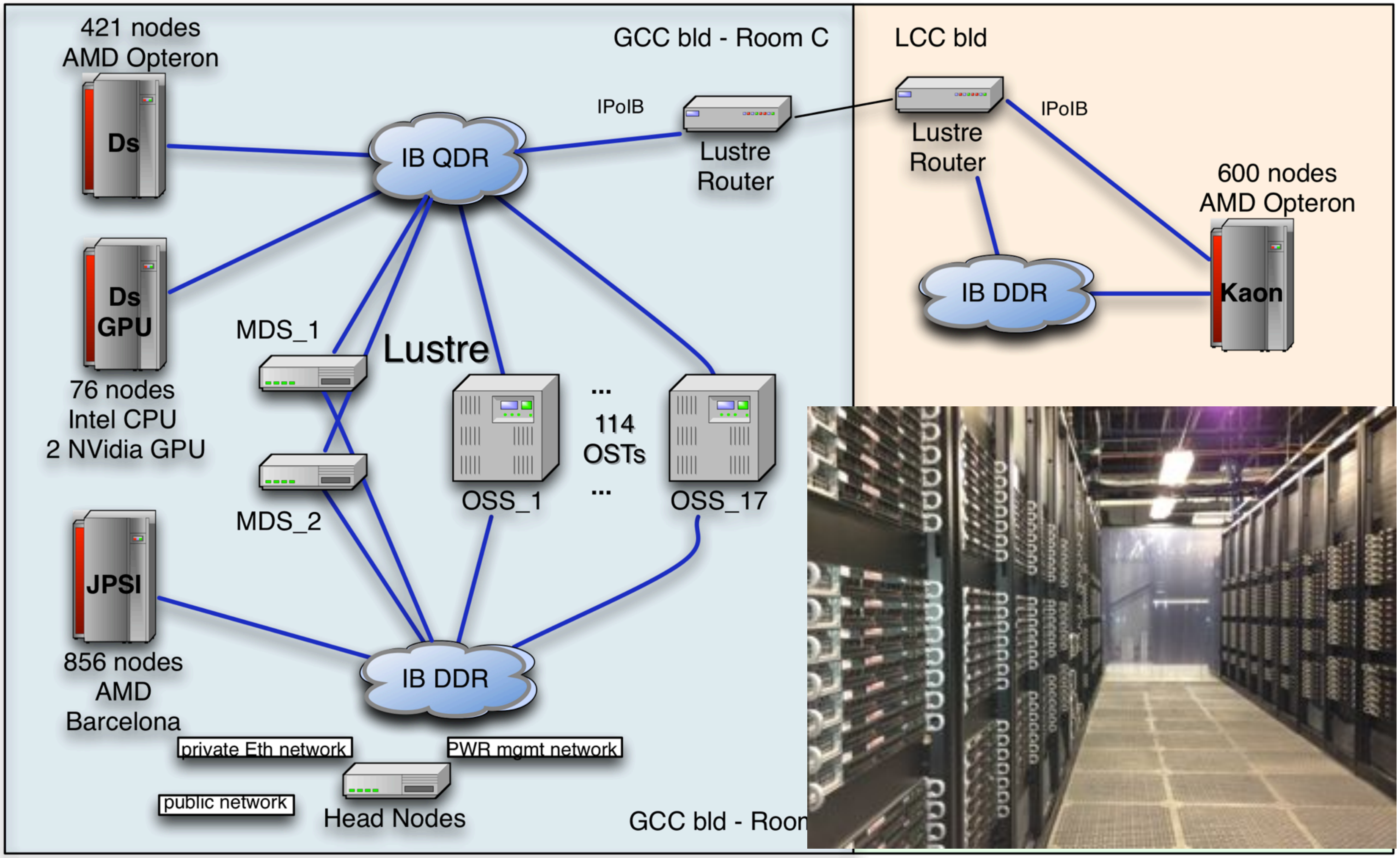
LQCD simulations

- Simulations require large parallel machines with parallel file system to store output which results in large dynamic range of files sizes, from hundreds of bytes to hundreds of gigabytes with mean below or about 1 MB.
- The simulation is performed in several stages.
- We generate large number of files representing ‘vacuum’ state (*gauge configurations*) – gigabytes each, 10k files/run
- On each of this configurations we calculate multiple simulations of quarks (*propagators*) – gigabytes each, 100k files/run
- Correlations between *propagators* correspond to physical measurements – 100k files/run (small files, 100-1000bytes)
- with hundreds computation runs going for month each

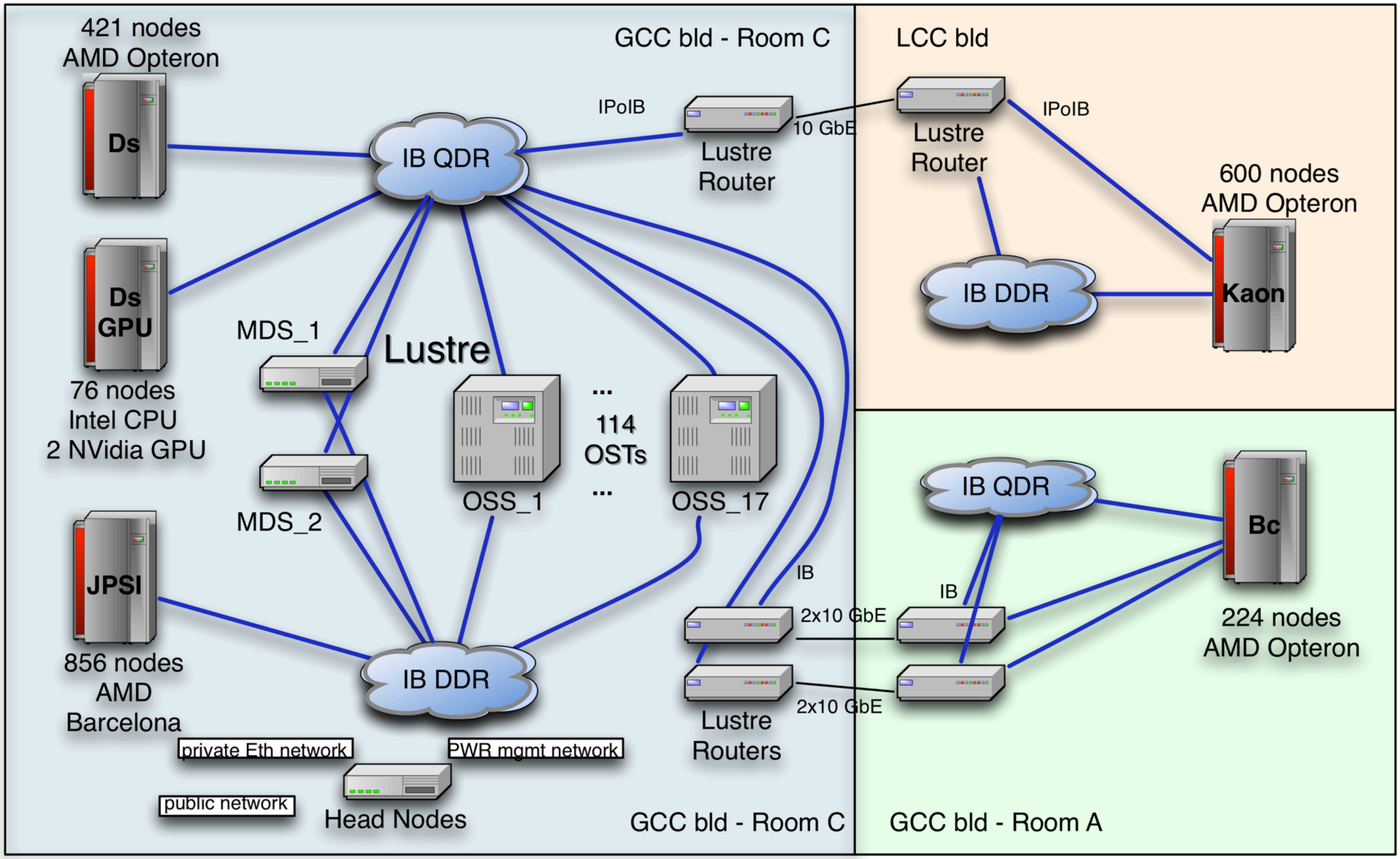
LQCD clusters

- common file system for four clusters (adding fifth)
- servers dual homed on two IB networks
- lustre 1.8.8 with 1.5.2 OFED
- servers use Scientific Linux 5.5 and kernel
- clients use SLF 5.5 and kernel.org kernel to be able to use most recent OFED, prefer lightweight kernel
- 614 TB, about 100 million files
- 17 OSS with 114 OSTs
- use lustre routers to extend LNET to another building (IPoIB) trough 10GE
- prototyped multiple IB-10GE-IB routers for new cluster

LQCD clusters



LQCD clusters – Lustre Routers



Cosmology Cluster

- configuration similar to LQCD cluster
- managed by the same group

- MDS/MGS pair
- six OSS with 48 OSTs
- 128 TB in five NexSAN SataBeast appliances
- DDR IB network
- lustre 1.8.8 on SLF 5.3
  - with RHEL OFED drivers
- initial deployment with Ethernet network
- replaced by IB that improved stability

Monitoring & Administration

- snapshot MDS every 12 or 6 hours to disk and backup snapshot image to the tape
- watchdog scripts:
  - kernel logs audits for LustreError
  - pings, servers and network status
  - monitor for OST space usage
- LMT: lwatch, cerebro (LLNL)
- collectl
- lltop (TACC)
  - helped to identify user overloading system with writes
  - xltop integration with Torque/PBS in progress

Monitoring

- special thanks to robinhood 2.4
  - enjoying web UI and fast find and du in new version
  - generate usage reports by user, group
- Handling large number of files ~100 millions can be challenging
- using DB in 120 GB RAM, rsync to disk
- scan takes 30 hours (fresh DB) or 11 hours update for 90 million files
- the load on MDT is different for initial scan (MDT stats bound) or rescan (DB bound)
- desire fast metadata scan for initial fill of DB
- metadata consistency MDT<->DB in general:
  - do MDT snapshot, scrub it and record changelog in parallel, then replay changlelog

Experiences

- stable lustre operation most of the time
- last summer get bitten by corruption when two inode pointed to the same (ost,oid) pair, resurfaced last month (20 files).
  - probably due to old corruption prior to 1.8.8 upgrade
  - rescanned all inodes to validate pointers <-> (ost,iod)
- it is vital to be able to do online or near online consistency checks:
  - lfsck is time consuming and can not be done online
  - operate fsck on MDT backup image, track fault history
  - store OID in DB and do consistency check

Large Metadata Challenges

- we used to have “project” area in lustre, then moved it to XFS
- need to store “history” for 81 million small files
- ‘rsnapshot’ perl utility provides unix TimeMachine functionality
- saves space and inodes:
  - 81 mil inodes -> 112 mil inodes (21 snapshot)
  - 4 TB -> 7.5 TB (21 snapshot)

  for 21 snapshots : 7 daily, 4 weekly, 12 monthly
- operation:
  - performs LVM snapshot
  - “rsync” to different disk creating hardlinks for unchanged files

  this results in high load on MDT to stat 81 mil files with 50 procs

Is there way to do correlated snapshot of MDT and some OSTs in lustre? ZFS is a future?

Space Management

- Hardware is installed and retired incrementally
- OST size (2.7-14.5 TB) and

storage per OSS vary a lot (16 to 85 TB) causing unbalanced read IO

Space Management

- watching disk usage and semi automatic rebalancing is regular duty
  - migrate data to new storage and/or replace OSTs
  - ability to set OST ‘read only’ to stop writes to existing files
  - reuse OST ID

Public Lustre File System

- Used by Data Movement and Storage Department for storage development of open source Enstore tape system features.
  - file cache for Small File Aggregation system preproduction tests.
  - streaming large files to/from Oracle STK T10000C tape drives at max drive rate ~240 MB/s.
  - Using ZFS-based NFS appliance in production for write cache.
- Data: NexSAN SataBeast
  - 40 TB, four arrays formatted RAID6 (8+2) = 32 TB
  - dual controller, twin 8 GBit FC to Nexan SataBeast
  - four OSS hosts with 10 Gbit Ethernet
- Metadata: NexSan SataBoy
  - FC to host, 1 GE network
  - failover pair of MDS/MGS
- SLF6.2, Lustre 2.1.3 (wc), cerebro, LMT

Lustre WAN

- FNAL participates in Extenci project supported by Open Science Grid, XSEDE and NFS
  - detailed reports on LUG’12, CHEP
- we host two OSTs in distributed lustre (FIU,UF,PSC)
- kerberized client crashes now fixed
- we installed Kerberized Lustre 2.0.62 (krb5n) to host lustre servers on DMS system
  - will upgrade to SLF 6.4 patched version (2.1.54) released last week

Plans: test kerberized lustre

- with remote clients in FIU
- with tens of clients on FermiGrid cluster (KVM)

Summary

- Lustre is in production for several years, it has gone through several upgrades and it has been stable for us.
- We less concerned with core lustre performance at this point as with ability to monitor system performance in consistent way integrated with batch system and network monitoring.
- It is essential to address Data Management to reduce administration burden
- Managing large number of files is a challenge.

Questions?