Scalable Petascale Storage for HEP using Lustre

C.J. Walker D.P. Traynor and A.J. Martin
Queen Mary University of London, School of Physics and Astronomy, Mile End Road, London
E1 4NS, UK
E-mail: C.J.Walker@qmul.ac.uk

Abstract. We have deployed a 1 PB clustered filesystem for High Energy Physics. The use of commodity storage arrays and bonded ethernet interconnects makes the array cost effective, whilst providing high bandwidth to the storage. The filesystem is a POSIX fileysystem, presented to the Grid using the StoRM Storage Resource Manager (SRM). We describe an upgrade to 10 Gbit/s networking and we present benchmarks demonstrating the performance and scalability of the filesystem.

1. Introduction
Analysis of the large quantities of data from the Large Hadron Collider (LHC) is performed using a distributed network of computing centres, the Worldwide LHC Computing Grid (WLCG). Queen Mary, University of London (QMUL) hosts a Tier-2 WLCG site, and this paper describes the configuration and performance of the clustered computing resources at this site.

One of the major challenges is to provide a cost effective storage solution that can deliver the large volume of storage required whilst providing sufficient bandwidth from the storage to the computational resources. We provide this storage using the cluster filesystem, Lustre[1], with an X509 authenticated interface provided by the StoRM Storage Resource Manager (SRM).

The work presented here describes the system design, tuning, and benchmarks demonstrating the performance and scalability of a system that provides 1 PB of storage. We also present some preliminary benchmarks after part of the cluster was upgraded to 10 Gbit/s networking (phase II).

2. Cluster Design
The system is based on a modular “brick” structure that combines commodity storage and compute. Physically mixing the servers within a rack balances power and cooling between racks (although storage and compute servers consume similar amounts of power, compute servers take up 1/4 the space). For LHC analysis, data are typically written once and read many times, so network traffic is predominantly from the storage to the compute. Combining storage and compute connections on the same ethernet switch allows traffic to be spread evenly across many switches. If we didn’t do this, traffic would tend to saturate the network uplink on storage switches and downlink on compute switches. Furthermore, a proportion of traffic won’t even need to traverse the uplink, though unfortunately, it isn’t possible to schedule jobs to take advantage of this data locality with the current grid infrastructure.

The cluster was designed to run a mixture of IO bound data analysis and CPU bound Monte Carlo simulation. We have maximised storage capacity, whilst providing sufficient IO bandwidth
to efficiently run more than 50% data analysis. A further requirement was that the cluster be easily expandable in future, potentially with different hardware.

2.1. Phase I
Our configuration\(^1\), deployed in 2011, is depicted in figure 1. The base system consists of 8 bricks, each with 6 Lustre Object Storage servers (OSS) with 20TB RAID 6 storage, and 12 Dual Intel X5650 compute servers. Each brick has a 48 port Gbit ethernet switch with a 10 Gbit/s uplink to a core switch. Bonded links were used to cost effectively increase the bandwidth: 4 from each of 6 storage servers and 2 from each of 12 compute servers, so half of the switch ports were connected to storage and half to compute servers.

Lustre Object Storage Servers: we have used 2U storage servers (Dell R510) with 12*2TB SATA disks in a RAID 6 configuration for bulk storage and 2 internal 160GB disks for the OS. Compute Servers: We have used Dell C6100 cloudservers - each 2U chassis has 4 independent motherboards with dual X5650 CPUs and 24GB RAM.

With 8 bricks, if data and jobs are randomly distributed, 12.5% of the network traffic will remain within a brick.

![Figure 1. Network topology: A core switch connected in a star topology at 10 Gbit/s to 8 bricks each with 12 compute servers and 6 storage servers connected using bonded Gbit/s ethernet.](image)

2.2. Commissioning
Initial commissioning of the disk servers involved running a series of stress and performance tests to identify faulty or misconfigured hardware. We used storage stress tests provided by RAL[4], HepSPEC benchmarks[3], and the Lustre ost-survey command. These tests identified faulty disks and backplanes, failed RAID controller batteries and incorrect BIOS settings throttling the CPU.

\(^1\) In addition to the configuration described here, the cluster has an additional 800 CPU cores and a further 300TB of storage on legacy servers.
We found that increasing the readahead setting for the disks gave a significant performance increase. The results presented are for a readahead of 8 megabytes².

2.3. Network bonding
Bonding is a cost effective way of increasing bandwidth, but does have some limitations: Bonding n Gbit links gives n*Gbit streams, not a n Gbit link. There are several different modes of bonding, and we chose IEEE 802.3ad Dynamic link aggregation (bonding mode 4) as it load balances across all links and provides fault tolerance in case of failure of one of the links.

The default (Layer 2 based) hashing algorithm used to distribute traffic across bonded links lead to uneven traffic distribution (figure 2). We found that changing the algorithm to use layer 3 and layer 4 information³ resulted in a more even traffic distribution. Inbound traffic distribution is dependent on the switch hashing algorithm.

![Figure 2. Distribution of traffic across network cards with different hashing algorithms.](image)

2.4. Performance
Iozone⁵ was used to measure read and write performance of a brick. With 6 Object Storage Servers (Fig 3), write performance was found to increase linearly to 2.5 Gbyte/s with 12 clients and 2 threads (close to 24Gbit/s network). Read performance increases to 1.5Gbyte/s with 1 or 2 threads. Write performance is better than read performance. This is typical of Lustre systems⁶ as the backend storage can aggregate writes efficiently, but reads from clients coming in a different order require seeks from disk.

3. Phase II
Motivated by the increase in data from the LHC, we are currently (Spring 2012) performing a major network upgrade. We plan to upgrade disk and compute servers to 10 Gbit/s ethernet, and deploy new switches with much increased bandwidth in a distributed core topology (figure 4). In addition, we have expanded the system with an additional 24 compute nodes and 480 TB of storage.

With the improved network, we have reconfigured our cluster to consist of larger bricks which contain twice the compute (24) and storage (12) nodes of the previous solution, a top of rack switch (Dell S4810) with 48 10 Gbit/s ports and 4 40 Gbit uplink ports giving 160 Gbit/s connectivity to a redundant pair of switches with 32 40 Gbit/s ports (Dell Z9000). The new system has 5 such bricks, so 20% of the traffic remains local to a switch.

² We set the disk readahead to be 16384 sectors, with each sector 512 bytes using the command: `/sbin/blockdev --setra 16384 /dev/sdb`
³ `xmit_hash_policy=layer3+4`
Figure 3. Performance of the phase I network. Write performance increases linearly up to 2.5 Gbyte/s with two threads, and read performance peaks at 1.6 Gbyte/s with two threads.

Figure 4. Phase II (2012) network topology. Each machine connected at 10 Gbit/s, and each top of rack switch connected at 160 Gbit/s to a distributed core.

The new Lustre object storage servers are Dell R510s like the previous batch, but contain 12*3TB nearline SAS NL-SAS disks - giving a 50% increase in storage capacity per server. 16 of these provide an additional 480 TB of storage. The 24 new compute servers have identical Dual Intel X5650 CPUs, but twice the RAM - at 48 GB per machine.

3.1. Performance
Preliminary results for one brick of the new system are included in this paper (figure 5). Write performance increases linearly to 6 Gbyte/s with 12 clients (one per OSS), and has a maximum of 8 Gbyte/s with 24 clients each with 2 threads. Read performance increases linearly to 8 Gbyte/s then drops off. With this cluster configuration of 5 bricks, we expect the maximum cluster performance to approach 40 Gbyte/s under a balanced load.

The new system has been deployed as an ATLAS Tier-2 resource, which routinely runs more than 2600 ATLAS analysis jobs in parallel, and peaks at over 20,000 jobs/day. Network traffic over the core switch (figure 6) sees short bursts as a bunch of analysis jobs start, and can peak
at 60 Gbit/s. This is well within the capabilities of the system, and demonstrates the scalability of our solution.

![Graph](image1)

**Figure 5.** Performance of the phase II network (machines connected at 10 Gbit/s)

![Graph](image2)

**Figure 6.** Network traffic for a typical week (18-25 Sept 2012) from the top of rack (s4810) switches to the core (Z9000) switches.

4. Conclusions
We have built a performant petabyte storage system from commodity hardware that routinely analyses more than 100TB/day of ATLAS data. With the upgrade to 10Gbit/s networking to
the servers, a brick can deliver read performance up to 8 GBytes/s. We currently have 5 such bricks and routinely run with the cluster fully loaded, with most of the jobs performing data analysis for the LHC. We are confident that this solution is scalable to at least 10 petabytes.

References
[1] The Lustre Filesystem - http://www.lustre.org
[2] The StoRM Storage Resource Manager http://storm.forge.cnaf.infn.it/home
[3] HEP-SPEC https://twiki.cern.ch/twiki/bin/view/FIOgroup/TsiBenchHEPSPEC
[4] http://www.sysadmin.hep.ac.uk/svn/fabric-management/ral-tier1-disk-tests/
[5] IOzone Filesystem Benchmark http://www.iozone.org/
[6] Lustre 1.8 operations manual - section 23.3.2 “Write performance better than read performance”
   http://wiki.lustre.org/images/3/3c/821-0035-12.pdf