High-Performance Storage System for the LHCb Experiment

Sai Suman Cherukuwada, CERN
Niko Neufeld, CERN

IEEE NPSS RealTime 2007, FNAL

Background

- Large Hadron Collider beauty
- One of 4 Major CERN experiments at the LHC
- Single-arm Forward Spectrometer
- b-Physics, CP Violation in the Interactions of b-hadrons

LHCb Data Acquisition

- Level 0 Trigger – 1 MHz
- High Level Trigger – 2-5 kHz
- Written to Disk at Experiment Site
- Staged out to Tape Centre

Storage Requirements

- Must sustain Write operations for 2-5 kHz of Event Data at ~30 kB/event, with peak loads going up to twice as much
- Must sustain matching read operations for staging out data to tape centre
- Must support reading of data for Analysis tasks
- Must be Fault tolerant
- Must easily scale to support higher storage capacities and/or throughput

Architecture : Choices

Fully Partitioned Independent Servers

Cluster File System

Architecture : Shared-Disk File System Cluster

- Can Scale Compute or Storage Components independently
- Locking Overhead restricted to very few servers
- Unified namespace is simpler to manage
- Storage fabric delivers high throughput

Shared-Disk File System Cluster

Hardware : Components

- Dell PowerEdge 2950 Intel Xeon Quad Core Servers (1.6 GHz) with 4 GB FBD RAM
- QLogic QLE2462 4 Gbps Fibre Channel Adapters
- DataDirect Networks S2A 8500 Storage Controllers with 2 Gbps host-side ports
- 50 x Hitachi 500 GB 7200 rpm SATA Disks
- Brocade 200E 4 Gbps Fibre Channel Switch

Hardware : Storage Controller

- *DirectRAID*
 - Combined Features of RAID3, RAID5, and RAID0
 - 8 + 1p + 1s
 - Very Low Impact on Disk Rebuild
- Large Sector Sizes (up to 8 kB) supported
- Eliminates host-side striping

•IOZone File System Benchmark with 8 threads writing 2 GB files each on one server

•Tested first in “Normal” mode with all disks in normal health, and then in “Rebuild”, with one disk in the process of being replaced by a global hot spare

Software

Software : Shared-Disk File System

- Runs on RAID volumes exported from Storage Arrays (called LUNs or Logical Units)
- Can be mounted by multiple servers simultaneously
- Lock Manager ensures consistency of operations
- Scales almost linearly up to 4 nodes (at least)

- IOZone Test with 8 threads, O_DIRECT I/O
- LUNs striped over 100+ disks
- 2 Gbps Fibre Channel Connections to Disk Array

Writer Service : Design Goals

- Enable a large number of HLT Farm Servers to write to disk
- Write Data to shared disk file system at close to maximum disk throughput
- Failover + Failback with no data loss
- Load Balancing between instances
- Write hundreds of concurrent files per server

Writer Service : Discovery

- Discovery and status updates performed through multicast
- *Service Table* maintains current status of all known hosts
- Service Table contents constantly updated to all connected *Gaudi Writer Processes* from the HLT Farm

Writer Process : Writing

- Cache every event
- Send to Writer Service
- Wait for Acknowledgement
- Flush and free

Writer Service : Failover

- Writer Processes aware of all instances of Writer Service
- Each Data *Chunk* is entirely self-contained
- Writing of a Data Chunk is idempotent
- If Writer Service fails, Writer Process can reconnect and resend unacknowledged chunks

Writer Service : Throughput

- Cached concurrent write performance for large numbers of files is insufficient
- Large CPU and memory load (memory copy)
- *O_DIRECT* reduces CPU and memory usage
 - Data need to be page-aligned for *O_DIRECT*
 - Written event data are not aligned to anything

- Custom test writing 32 files per thread x 8 threads
- Write sizes varying from 32 bytes to 1 MB
- LUNs striped over 16 disks
- 2 Gbps Fibre Channel Connections to Disk Array

Writer Service : Throughput per Server

- Scales up with number of clients
- Write throughput within 3% of maximum achievable through GFS

- Custom test writing event sizes ranging from 1 byte to 2 MB
- LUNs striped over 16 disks
- 2 Gbps Fibre Channel Connections to Disk Array
- 2 x 1 Gbps Ethernet Connections to Server
- CPU Utilisation ~ 7-10%

Conclusions & Future Work

- Solution that can offer high read and write throughput with minimal overhead
- Can be scaled up easily with more hardware
- Failover with no performance hit

- More sophisticated “Trickle” load-balancing algorithm in the process of being prototyped
- Maybe worth implementing a full-POSIX FS version someday?

Thank You

Writer Service

- Linux-HA not suited for Load Balancing
- Linux Virtual Server not suited for Write Workloads
- NFS in *sync* mode too slow, *async* mode can lead to information loss on failure