Data Management for Batch Systems

Ian G. Bird, Rita Chambers, Mark E. Davis, Andy Kowalski, Sandy Philpott, Dave Rackley, Roy Whitney

SURA/Jefferson Lab, 12000 Jefferson Avenue, Newport News, VA 23606

By late-1997, Jefferson Lab will collect over 1 TB of raw information for each day of accelerator operation. In this paper, we will present our implementation to manage the flow of data between a central mass storage system and a 50-CPU farm dedicated to batch mode data reconstruction of experimental raw data.

Key words: Batch; Data; Farm; HSM; RAID;

The Thomas Jefferson National Accelerator Facility ("Jefferson Lab") is operating the world's first high-energy (4 GeV), continuous beam, superconducting radio-frequency, electron accelerator for physics research. By summer, 1997, with all three experimental halls in simultaneous production, the laboratory will collect over one Terabyte (TB) of new raw data for each day of accelerator operation. This paper focuses on the design issues in managing the data flow to an off-line CPU farm of UNIX processors dedicated to the reconstruction and replay of these experimental results. During this process, which may be repeated 2 to 3 times for each data set, up to a byte of reconstructed data may be generated for each byte of raw input data. The aggregate data flow between the central mass storage facility and the CPU farm may reach several TB in a 24 hour period, and over a Petabyte (PB) each year. We will discuss a variety of magnetic disk configurations considered to manage this data flow, provide results of our data throughput testing, and present the implementation which will support a 300 SPECint95 farm required by late 1997.

Jefferson Lab's experimental data is stored in a central mass storage facility consisting of a StorageTek (STK) silo interfaced to four high performance STK Redwood tape transports. Using the hierarchical storage manager (HSM) Open Storage Manager (OSM, Computer Associates) running on a Sun Enterprise 4000 data server, a transfer rate of over 10 MB/s has been achieved copying data files between the silo and RAID disk areas. This transfer rate to a single tape drive, matches the expected 10 MB/s raw data stream incoming via dual ported Fibre Channel RAID from the CEBAF Large Acceptance Spectrometer (CLAS) in Experimental Hall B. The remaining tape transports support data from Experimental Halls A and C (approximately 2-3 MB/s each, via Fast Ethernet), plus manage the data flow between the off-line batch CPU farm and the silo. As many as 6-10 tape transports may be required at full operation depending on the level of data reduction achieved.

Approximately 200 SPECint95 of processing power will be required by late FY97 to process the CLAS event data. Considering the additional requirements (100 SPECint95) for the other experimental halls, a CPU farm of on the order of 50 processors must be considered in the design of the data handling operation. The efficiency of fanning the data out from the central silo to this array of processors will be affected by the transfer rates achieved through the data server between the Redwood tape drives and the RAID staging areas, as well as by network throughput and the I/O capabilities of individual batch nodes and their local disks. Hall B simulations indicate that processing one 2 GB data set will require approximately 2 hours and may generate an output file equal in size in early phases of understanding the system. Each batch CPU, which will independently process one data set, must therefore transfer 2 GB of data per hour, roughly 600 KB/s. The aggregate throughput for a 50 node farm then must be 30 MB/s. A number of possible scenarios to handle these requirements were considered. These included large central RAID staging areas, NFS servers, local storage on the farm nodes, and multi-host interconnects to RAID subsystems.

Preprint submitted to Elsevier Preprint *4 March 1997*

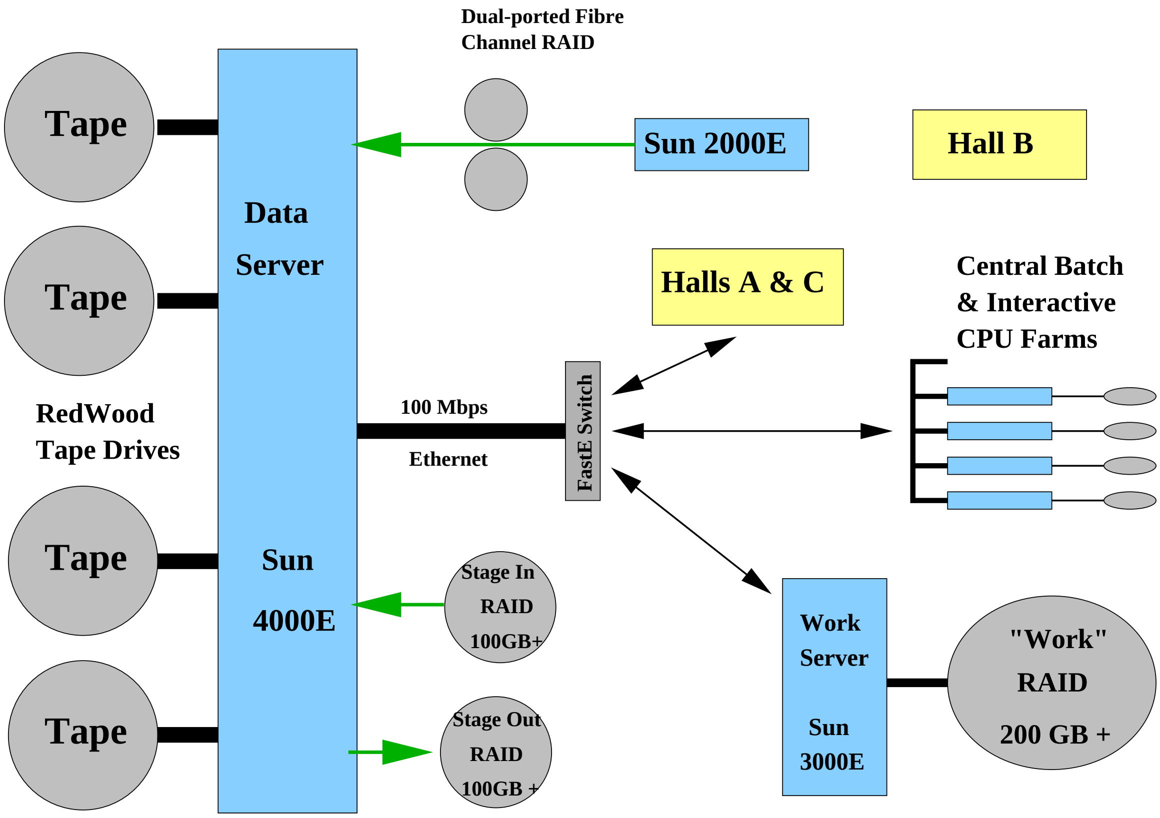

Fig. 1. Data Handling Configuration

Researchers also require longer term storage "work" areas to perform intermediate visualization, process feedback, and code tuning. Impending jobs may rely on previous results, underscoring the importance of efficient purge and reload algorithms. The large volumes of data alone demand attention to user- and experiment-level quotas. Processing, moreover, is performed in a cross platform CPU environment, tightly integrated with other central computer and file systems. Efficiency in both hardware and software design are critical.

Several configurations were rejected early in the design process. Although minimizing the number of times a data set is copied should reduce time and cost, direct network copies from tape to the local batch node dramatically increase the performance requirements (and hence cost) of the individual batch node, and reduce the efficiency of the Redwood tape drives. Alternatively, both input and output data could be stored in a central, higher performance, RAID subsystem and accessed by the local batch node via the network. Lessons learned with production network transfers of Hall C data rule out the use of NFS for writes, implying that at least some local storage of output files is mandatory. We tested the possible use of a dedicated high performance NFS server to receive data from the silo and stage it out via NFS to the farm nodes; however a maximum aggregate performance less than 4 MB/s for reads and 3 MB/s for writes effectively ruled this out. We considered multi-host connections to a large central RAID. Although multiple vendors now offer this option, the file system considerations in a cross-platform environment, plus the added complexity of managing a large number of small file system partitions, impact the feasibility of this option at the present time. Dual porting a smaller number of RAID subsystems as a high throughput medium to transfer data between central data servers may in fact be a part of the final Jefferson Lab design.

The planned implementation (see Fig. 1) will employ two UltraSCSI RAID subsystems (RAID 5, using moderate levels of parity disks and hot spares) directly connected to the Sun 4000E to handle staging in and out of the STK silo. This staging process, controlled by the Jefferson Lab Off-line Batch System (JOBS), under development [1], will distribute data over 100 Mbps switched Ethernet to batch nodes on dedicated network segments (approximately 10 processors per segment). Although the initial capacity will be 100 Gigabytes (GB) each, at least 150 GB per tape drive (3 times the 50 GB cartridge capacity) will be required to maintain tape streaming. We will employ the rfio utility (part of the CERN SHIFT package) for network copies to SCSI II Fast Wide data disks on the local CPU. Limiting the input and output files to 2 GB enhances CPU utilization by allowing two concurrent batch jobs to run on each processor with one 9 GB disk per CPU. The "work" area will be a separate RAID subsystem on a second Sun data server (3000E).

Three rounds of testing were performed to develop and validate the proposed design (see *http://www.cebaf.gov/ccc/raidtests.html*).
Initial tests to determine the maximum throughput possible between the Redwoods, disks, and networked CPUs used software RAID (0 and 5, Solstice DiskSuite) on SCSI II Fast Wide 7200 rpm drives. Our test results using RAID 0 achieved approximately 10 MB/s from tape to RAID, and 4 MB/s for a single network transfer between the 3000E and one batch node (a Sun Ultra 170); however, throughput was significantly degraded when the number of simultaneous transfers was increased. Depending on the final throughputs achieved with high performance, hardware-based RAID, the JOBS staging algorithm may consequently round robin to load each batch node rather than allow asynchronous transfer requests. A second tier of tests used an existing commercial NFS fileserver for read/write timed tests (same URL). As part of the procurement of the permanent RAID subsystems, an on-site benchmarking demonstration was required by each bidding vendor (see *http://www.cebaf.gov/ccc/vendortests.html*). While the results are procurement sensitive, we are confident that we can achieve transfer rates from the silo to RAID in excess of 10 MB/s and multiple simultaneous transfers from a single RAID subsystem to the networked batch nodes each in excess of 5 MB/s. To achieve the 30 MB/s aggregate required for the full 50-node farm will most likely require doubling the number of servers and RAID subsystems in the startup implementation.

Future enhancements to the design may include dual porting the staging RAID areas between the two data servers to enhance storage to and from the central work area. Overall transfer rates may be increased further by performing software striping across more than one hardware-based RAID subsystem. Vendor advances in the parallel transfer of data across multiple I/O interfaces to the same file system would dramatically increase the potential throughput achievable. The limitations in simultaneous loading of multiple batch nodes may be eased when Gigabit Ethernet becomes available in mid to late 1997.

References

[1] "Database Driven Scheduling for Batch Systems"; I .G .Bird, R. Chambers, M. E. Davis, A. Kowalski, S. Philpott, D. Rackley, R. Whitney. Paper submitted to this conference.