Scidac-Data: Enabling Data Driven Modeling of Exascale Computing

Misbah Mubarak,[1] Pengfei Ding,[2] Leo Aliaga,[2] Aristeidis Tsaris,[2] Andrew Norman,[2] Adam Lyon,[2] and Robert Ross[1]

[1]Argonne National Laboratory, Lemont, IL, USA

[2]Fermi National Accelerator Laboratory, Batavia, IL, USA

E-mail: {mubarak,rross}@mcs.anl.gov {dingpf,laliaga,atsaris,anorman,lyon}@fnal.gov

Abstract. The SciDAC-Data project is a DOE-funded initiative to analyze and exploit two decades of information and analytics that have been collected by the Fermilab data center on the organization, movement, and consumption of high energy physics (HEP) data. The project analyzes the analysis patterns and data organization that have been used by NOvA, MicroBooNE, MINERvA, CDF, $D0$, and other experiments to develop realistic models of HEP analysis workflows and data processing. The SciDAC-Data project aims to provide both realistic input vectors and corresponding output data that can be used to optimize and validate simulations of HEP analysis. These simulations are designed to address questions of data handling, cache optimization, and workflow structures that are the prerequisites for modern HEP analysis chains to be mapped and optimized to run on the next generation of leadership-class exascale computing facilities.

We present the use of a subset of the SciDAC-Data distributions, acquired from analysis of approximately 71,000 HEP workflows run on the Fermilab data center and corresponding to over 9 million individual analysis jobs, as the input to detailed queuing simulations that model the expected data consumption and caching behaviors of the work running in high performance computing (HPC) and high throughput computing (HTC) environments. In particular we describe how the Sequential Access via Metadata (SAM) data-handling system in combination with the dCache/Enstore-based data archive facilities has been used to develop radically different models for analyzing the HEP data. We also show how the simulations may be used to assess the impact of design choices in archive facilities.

1. Overview

High energy physics (HEP) intensity frontier and energy frontier experiments are accumulating petascale datasets that need to be processed, managed, and archived effectively. At the same time the facilities, both storage and compute, that support these activities need to be designed and tuned to effectively deliver the needed throughput so that experimentalists can analyze their data in a timely manner. This process of design and tuning is extremely difficult because of the large design and parameter space that needs to be considered.

One method for exploring this space is through discrete-event simulation. This approach uses workflows from the actual physics experiments that are representative of the typical analysis and usage pattern. Applying these usage patterns, through the simulation, to a given model of a computing facility's data management system, can help HEP scientists understand the performance impacts that their analysis have on the underlying infrastructure. This then enables them to experiment with design choices relating to the infrastructure such as storage bandwidth, data cache sizes, retention policies and look-ahead algorithms and evaluate the impact they will have on overall system performance. To this end, we have used a discrete-event simulation to model the data center at the Fermi National Accelerator Laboratory.

Fermilab maintains and operates a mass data storage which services both the CMS experiment and a suite of intensity frontier experts which focus on neutrino and precision muon physics. The storage system provides archival (tape) storage as well as high speed disk cache access to experimental data. The Fermilab facility is integrated with a data management system, which provides a bridge between the NOvA, MicroBooNE, MINERvA, MINOS, and other experiment's analysis code bases, and the underlying data storage and movement tools. The schematic architecture of the Fermilab system is shown in Figure 1. The architecture depicted shows the high latency Enstore based tape library and robot system, which is fronted by a high performance, low latency distributed cache layer.

Users submit large collections of analysis jobs to a batch queueing system. The batch jobs requests data via the Sequential Access via Metadata (SAM) data management system [1], which is a comprehensive data management and catalog system for large-scale data storage and mining. The SAM system optimizes the retrieval of files from the Enstore tape system by using a scheduler layer which executes a look-ahead based algorithm to determine data retrieval lists. These lists are then used to trigger preemptive restoration of files from the tape system and to populate the dCache cache layer. The Enstore tape libraries have a storage capacity of over 570 petabytes of data based on a mixture of T10K and LTO tape systems. This tape archive system has been in operation since the Tevatron Run II era and has been described extensively [2]. The dCache base cache layer is comprised of a distributed set of high speed storage pools whose aggregated storage capacity in is approximately 4 petabytes. Data that is loaded to or restored from the tape archives, is routed through the dCache layer where a copy of the data is retained in the cache pool subject to a cache policy or algorithm[3].

To simulate the Fermilab data facility, the known static characteristics of the dCache, Enstore, batch systems and networking were used to construct the simulation components. Two sets of historic job activity logs were then used as driving input workloads to the simulations. The first set of workloads represented the actual observed activity from 5 years of analysis, simulation and data processing by a single experiment (NOvA) operating in the FNAL computing environment[4]. The second set of workloads represented the combined activity from three major experiments (NOvA, MicroBooNE and MINERvA) operating simultaneously over a period of one month in the full Fermilab computing environment [5]. The simulation was configured to report a series of metrics relating to the overall performance and efficiency of the data transport and storage layers of the combined disk, cache and tape system. In particular the simulation reported predictions for cache occupancy, file lifetimes within the cache (time to live), and tape drive activity.

2. Discrete-event Simulation Model

The SciDAC-Data simulation of the FNAL storage system was developed by using the CODES simulation framework with components that were specialized to represent the FNAL storage systems and their interactions. The CODES framework is an established toolkit which simulate large-scale high-performance computing (HPC) workloads and scientific workflows, high-performance networks, storage systems, and distributed data-intensive systems [6, 7]. CODES uses the Rensselaer Optimistic Simulation System (ROSS) discrete-event simulation framework, which has been used to process billions of events per second on modern HPC systems [8]. Simulations in CODES/ROSS comprise logical processes (LPs), each of which represents a distinct entity in the system. The LPs interact with each other via timestamped messages or events. By way of example, if simulating a computer network, an LP can be

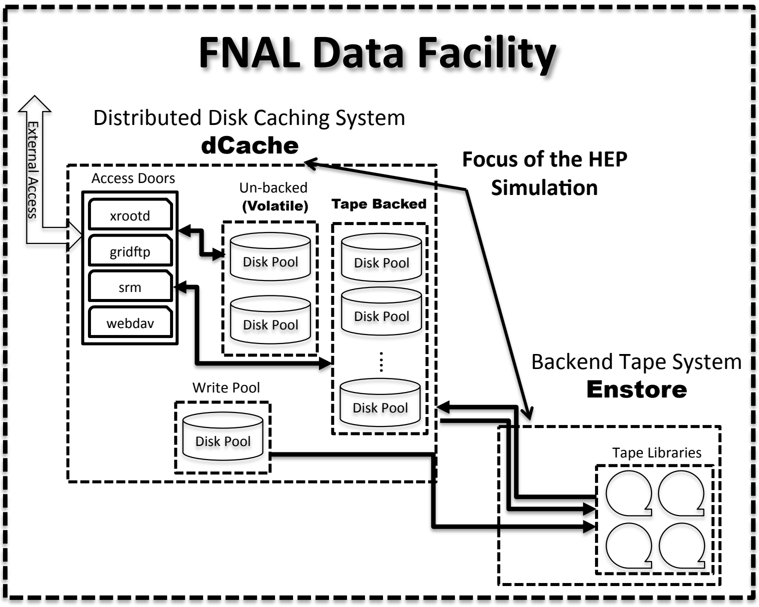

Figure 1. Schematic layout of components of the Fermilab archive facility that are modeled in the simulation. Components of the data storage and management system that are beyond the scope of the simulation are not shown.

implemented as a network node that can issue events such as the transmission of network packets at discrete times.

The model of the Fermilab HEP computing environment model, implemented in CODES, is comprised of four LP types: a batch scheduler, queue and SAM look-ahead scheduler, standard worker/compute cores, disk cache, and tape storage The scheduler LP in the simulation provides the combined functions of batch queue scheduler, which schedules jobs on a first-come, first-served (FCFS) basis and of the SAM look-ahead scheduler which optimizes the pre-staging and assignment of data to executing analysis jobs. Additionally, the scheduler LP provides the data management functions of keeping a catalog for the files and maintaining information on the files including their identifiera, storage locations, cache status, and file size. The cache and tape storage LPs implement performance characteristics of each hardware system, including startup and access latencies and bandwidth profiles associated with them. Each of these LPs was configured using the actual latencies and bandwidth numbers of Fermilab's mass storage hardware. Fermilab's cache system uses a "least recently used" (LRU) policy for cache eviction, the LPs in the simulation were configured to execute a similar LRU policy for deleting files from cache. Other parameters, including the cache capacity were left as runtime configurable parameters to the simulation which can be provided by the user. This allowed the simulation to be used to scan over and explore the impact of different cache sizes and other operational parameters on the HEP workflow execution.

In practice, analysis workflows submitted to the SAM data management system require that all files belong to a specific dataset. Files may reside either on the dCache or on Enstore. Once an analysis workflow is initiated by the SAM scheduler, a set of individual jobs are assigned to the worker cores, with each job processing a subset of files from the dataset. When a job is finished processing a file, it requests another file. The SAM scheduler prefetches files from

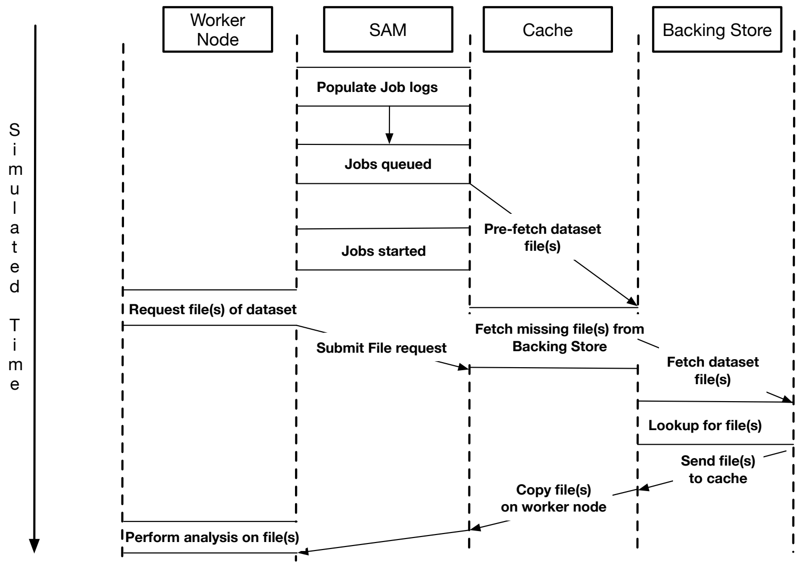

Figure 2. Components of the CODES HEP simulation and their interactions. Cache represents dCache, and backing store represents the Enstore system at Fermilab.

Enstore just before a job needing it is executed on a worker core (unless that file is already in dCache). Since the SAM scheduler keeps track of the location of files, if a file is not present in the cache, it requests the file from the Enstore. The process continues until all the files in the dataset have been consumed by the jobs. Correspondingly, in our simulation we have modeled the same interaction between worker cores, SAM, cache, and tape LPs by using realistic activity logs from the NOvA, MicroBooNE, and MINERvA experiments at Fermilab, where the workflow consists of a series of jobs submitted over a period of time that could span several years. Figure 2 shows the detailed interaction between different components of the simulation.

Collecting a variety of statistics is important for performance prediction and design space exploration of systems. Our simulation reports detailed statistics about the modeled storage system over an interval that is configurable. The statistics include the following:

File eviction times from cache : This metric provides information about the frequency of file evictions from the cache and the point in time when the eviction occurred. If a file gets evicted from cache and re-enters, the file life counter starts from the beginning.

Cache hits and misses : This metric records the file count and volume of data accessed from the cache and whether the access request was a cache hit or a miss.

Cache utilization over time : During each time interval, the simulation reports the occupied size of the cache and amount of data being accessed from the cache during that interval.

Start and end times of workflows : Statistics about each workflow are reported, such as the time that a workflow gets scheduled and the time to complete it.

Tape activity over time : This metric reports the amount of data and number of files being accessed from tape during a specific time interval.

3. Evaluation

Our simulation system was evaluated using the realistic workflow information from two different datasets, each derived from historic activity logs:

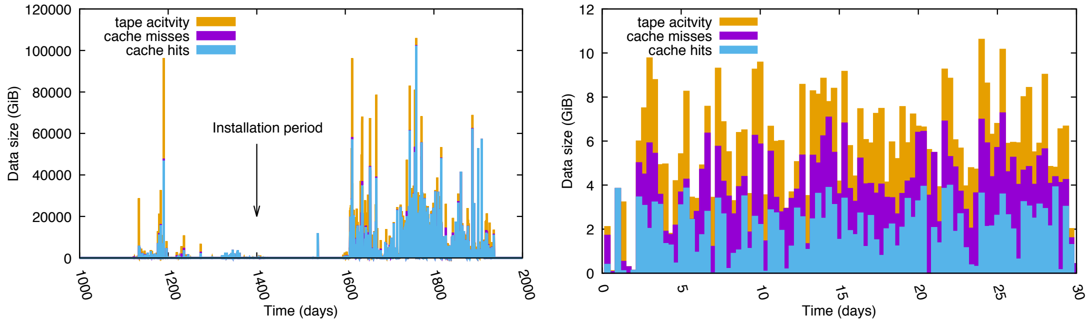

Figure 3. Tape activity, cache hits, and cache misses as reported by the CODES simulation for simulation of the FNAL facility utilizing workload input vectors based on a) 5-years of analysis workflows from NOvA (right) and b) One-month of combined analysis workflows from NOvA, MicroBooNE, and MINERvA experiments (left).

NOvA Analysis Dataset comprised of five and a half years of activity logs that describe the running and completion of HEP analysis workflow for the NOvA experiment. These workflows represent a total of 9,536 analysis campaigns which ran a total of 9,028,534 computational analysis jobs on the Fermilab computing clusters and transferred in excess of 5.5 PB of data between the storage systems and the analysis computing. To ensure full coverage of the dataset, the simulation was configured to simulate a six year time window with a 60,000 worker core batch farm and 5.6 PB of aggregated data transfer.

GPGrid Environment Dataset comprised of one month of activity logs that describe the combined running of the NOvA, MicroBooNE, and MINERvA experiments in the context of the Fermilab computing facility. This dataset comprised a more diverse set of 61,735 analysis workflows resulting in the submission of over 1 million jobs that were submitted and handled by the FNAL batch system in May 2016. The dataset represents an aggregate data transfer of 3.14 petabytes between the storage facility and the compute elements.

The simulation was configured with a disk cache bandwidth of 1.25 GB/s and a bandwidith of 200 MB/s for data retrieval/restore from the tape system. These parameters were chosen based on the Fermilab dCache and Enstore system's operational parameters. We present the results of this workload using the detailed statistics reported by the simulation in Section 2.

The first set of observables reported by the simulation were the tape activity, cache hits, and cache misses over the simulated time interval. This was done separately for the NOvA dataset and for the combined running of the NOvA, MicroBooNE, and MINERvA experiments. The left plot of Figure 3 shows the tape activity, cache hits, and cache misses during the high analysis activity periods for the NOvA experiment. The observed gap between days 1,400 and 1,600 of the simulation represents a period when few analysis jobs were submitted. This is consistent with the known detector installation phase of the NOvA experiment, followed by the flurry of analysis activity as the detector entered data taking operations. The right plot in Figure 3 shows the same observables for the combined activity of the NOvA, MicroBooNE, and MINERvA experiments over a one month period corresponding to May 2016.

In both cases, the simulation is configured to have a disk cache size of 2 PB corresponding to the actual Fermilab dCache storage size. The data handling system [SAM] optimizes and initiates data retrieval through a "prefetch" step when the compute jobs are scheduled and as a preemptive "look-ahead" when the jobs are running. This results in the prefetching phase accounts for a significant fraction of tape activity seen over the simulated time, when the analysis efforts switched within the experiment to a different primary set of detector data or Monte Carlo

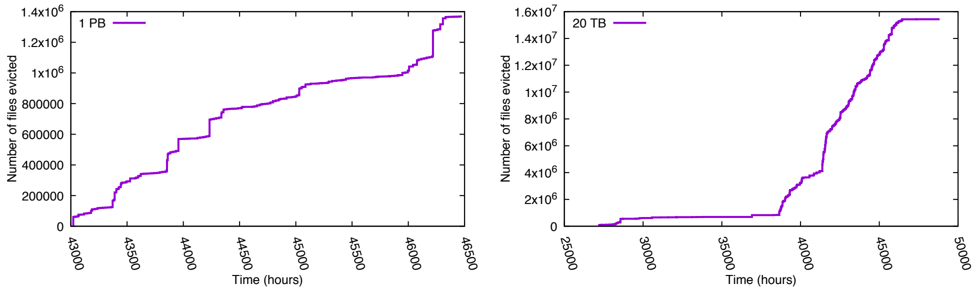

Figure 4. File eviction from cache via LRU scheme using NOvA 5-year job activity logs with a cache size of (left) 1 PB and (right) 20 TB (no cache evictions happen at 2 PB cache size).

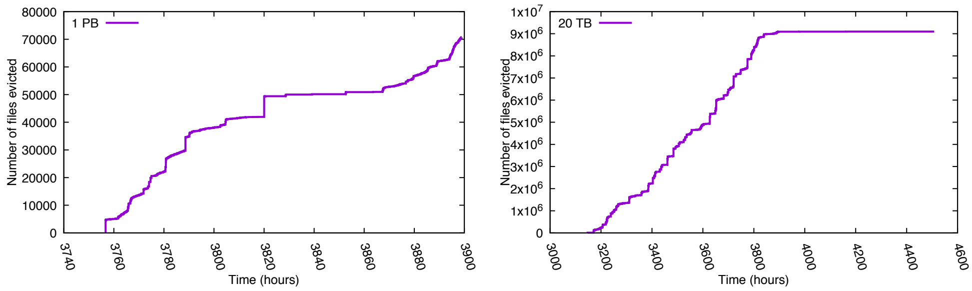

Figure 5. File eviction from cache via LRU scheme using job activity logs from NOvA, MicroBooNE and MINERvA experiments from May 2016 with a cache size of (left) 1 PB and (b) 20 TB (no cache evictions happen at 2 PB cache size).

simulation (i.e. the entire set of physics data had to be initially restored from tape). In this manner, if a requested file for an executing job is found in the cache, it is counted as a cache hit and if the file has to be restored from tape during the execution of the job (thereby stalling the job) it is counted as a cache miss.

For the five-year NOvA activity logs, the optimized prefetching phase brings the required files to be fetched from the tape, which decreases the number of cache misses. Since the cache size is large enough to fit the fetched files, no files are evicted, and minimal cache misses are observed. In contrast for the combined running of the NOvA, MicroBooNE, and MINERvA experiments, file requests start right after the prefetching phase is initiated, resulting in significant cache misses since not all files have been prefetched to the cache at the time they are requested by the worker cores.

Figures 4(left) and (right) show the impact of cache size on file lifetime when the cache size is decreased from 2 PB to 1 PB and 20 TB for the five-year NOvA activity logs. As expected with a cache size of 2 PB, none of the files get evicted from the cache since the aggregate data size requested by the jobs peaks at 1.6 PB. Once fetched from the tape, the files stay in the cache until the end of the simulation since the cache is big enough to hold all the data. In contrast, as the cache is decreased to half its size (1 PB) as shown in Figure 4(left) 1.4 million files are observed to be evicted from the cache. Further decreasing the cache size to 20 TB, results in the eviction of over 16 million files during the simulation window, as the cache is now to small to properly host the data being analyzed at any given time in the compute nodes.

Figure 5 shows the impact of cache size on the file lifetime of the one-month activity logs. In this case, there are 70,000 cache evictions for 1 PB cache size and 9 million evictions for 20 TB of cache size. Such experiments demonstrate the feasibility of using simulation to evaluate

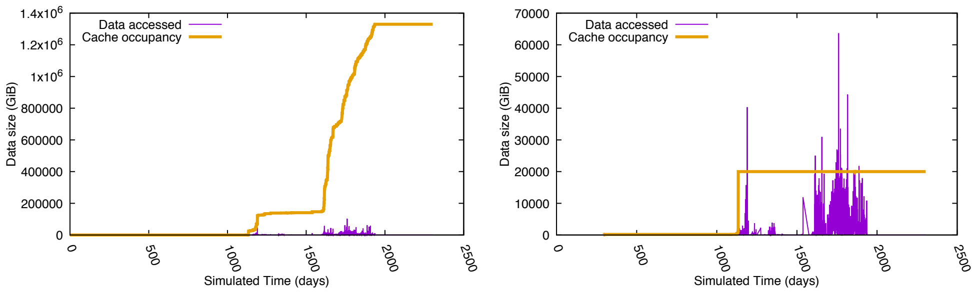

Figure 6. Cache utilization over simulated time using NOvA 5-year job activity logs where (left) cache size is configured to 2 PB and (right) cache size is configured to 20 TB.

effective cache sizes based on historical data processing from HEP experiments.

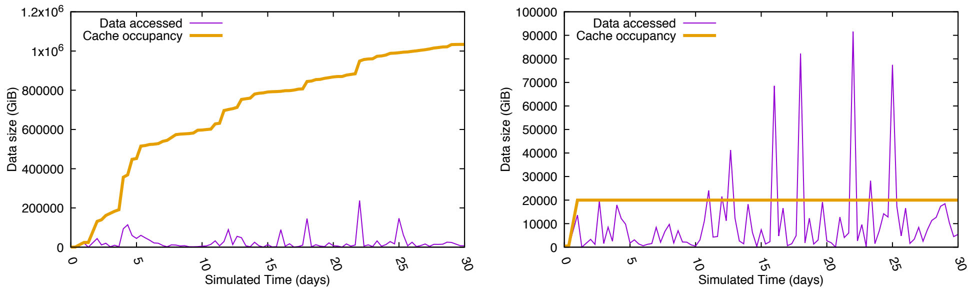

Figure 7. Cache utilization over simulated time using the NOvA, Minerva, and MicroBooNE dataset from May 2016 where (left) cache size is configured to 2 PB and (right) cache size is configured to 20 TB.

Figures 6 and 7 show the cache utilization for the five-year and one-month activity logs in terms of data being accessed and the occupied cache size. Initial cache activity is due to prefetching of files from the tape since the simulations start with an unpopulated cache. As soon as the job submissions start, the occupied cache size starts increasing. With a cache size of 2 PB, requests for data access from the cache stay well below the size of the cache; hence, we do not see any file evictions when using 2 PB. With 20 TB, however, requests for data access from the cache grows more than the occupied cache size and results in thrashing of the cache. Monitoring the cache utilization provides information on the percentage of the data in the cache that is actually being utilized at a given point in time.

4. Future Plans: Adapting HEP Workflows to an HPC Environment

As part of the CODES simulation framework, we have developed detailed network and storage models of current and future extreme-scale architectures [7, 9]. Low-latency HPC interconnects on which packets have to traverse a minimal number of hops because of their low network diameter are becoming an integral component of HPC systems. CODES provides detailed packet-level models of HPC interconnect network topologies such as dragonfly, fat tree, Slim Fly, and torus interconnects [10, 7, 11]. CODES simulations combines models of storage devices, high-performance interconnects and I/O forwarding layers of HPC systems and have been used successfully to demonstrate the usefulness of burst buffer storage nodes on a large-scale HPC system [9]. CODES provides the HPC interconnect and storage models as pluggable components so that model developers can experiment with different HPC components with minimal changes to the code.

Integrating these HPC storage and interconnect models with the HEP workflows will enable us to predict performance and throughput of the HEP analysis workflows on the next generation of extreme-scale systems. With these HPC models, we also will be able to do parameter tuning of HEP experimental workflows and adapt them to the HPC environment.

5. Summary

As part of the SciDAC-Data project, we are working to analyze the historical information collected by the Fermilab data center over the past two decades. Using the CODES simulation framework, we have developed simulations of the Fermilab data center that can assist researchers in effective data handling and cache optimizations. An accurate end-to-end simulation can provide insight into the performance of these complex data management systems and help scientists answer questions about data movement and resource allocation policies, identify potential bottlenecks, and quantify the value of adding new hardware such as disks and tapes.

We have taken a step in this direction by using realistic data from Fermilab's NOvA, MicroBooNE, and MINERvA experiments and replaying the historical data on our discrete-event simulation of Fermilab's storage system. We have designed the simulation to report detailed metrics on the performance of the storage model, such as cache utilization, cache lifetime, cache hits and misses, and tape activity over time.

Acknowledgments

Fermilab is operated by Fermi Research Alliance, LLC under Contract No. De-AC02-07CH11359 with the U.S. Department of Energy. The research at Argonne is based upon work supported by the U.S. Department of Energy, Office of Science, under Contract No. DE-AC02-06CH11357.

References

[1] Illingworth R 2014 A data handling system for modern and future Fermilab experiments *Journal of Physics: Conference Series* vol 513 (IOP Publishing) p 032045 URL `https://doi.org/10.1088/1742-6596/513/3/032045`

[2] Bakken J, Berman E, Huang C H, Moibenko A, Petravick D and Zalokar M 2003 The Fermilab data storage infrastructure *Mass Storage Systems and Technologies, 2003. (MSST 2003). Proceedings. 20th IEEE/11th NASA Goddard Conference on* pp 101–104 URL `https://doi.org/10.1109/MASS.2003.1194845`

[3] Ernst M, Fuhrmann P, Gasthuber M, Mkrtchyan T and Waldman C 2001 dcache, a distributed storage data caching system *Proceedings of CHEP 2001* pp 757–763

[4] Habig A, Norman A and Group C 2015 *J. Phys. Conf. Ser.* **664** 032011

[5] Group C *et al.* 2015 *J. Phys. Conf. Ser.* **664** 032012

[6] Cope J, Liu N, Lang S and Ross R 2011 CODES: Enabling co-design of multilayer exascale storage architectures *Proc. of the Workshop on Emerging Supercomputing Technol.*

[7] Mubarak M, Carothers C D, Ross R B and Carns P 2017 *IEEE Transactions on Parallel and Distributed Systems* **28** 87–100 URL `https://doi.org/10.1109/TPDS.2016.2543725`

[8] Barnes P D, Carothers C D, Jefferson D R and LaPre J M 2013 Warp speed: executing time warp on 1,966,080 cores *Proc. of the 2013 ACM SIGSIM Conf. on Principles of Advanced Discrete Simulation (PADS)* pp 327–336

[9] Liu N, Cope J, Carns P, Carothers C, Ross R, Grider G, Crume A and Maltzahn C 2012 On the role of burst buffers in leadership-class storage systems *2012 IEEE 28th Symposium on Mass Storage Systems and Technologies (MSST)* (IEEE) pp 1–11

[10] Yang X, Jenkins J, Mubarak M, Ross R B and Lan Z 2016 Watch out for the bully! job interference study on dragonfly network *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis, ser. SC* vol 16

[11] Wolfe N, Carothers C D, Mubarak M, Ross R and Carns P 2016 Modeling a million-node slim fly network using parallel discrete-event simulation *Proceedings of the 2016 annual ACM Conference on SIGSIM Principles of Advanced Discrete Simulation* (ACM) pp 189–199