The Fermilab data storage infrastructure

Jon Bakken
bakken@fnal.gov

Eileen Berman
berman@fnal.gov

Chih-Hao Huang
huangch@fnal.gov

Alexander Moibenko
moibenko@fnal.gov

Donald Petravick
petravick@fnal.gov

Michael Zalokar
zalokar@fnal.gov

Fermi National Accelerator Laboratory

Abstract

Fermilab, in collaboration with the DESY laboratory in Hamburg, Germany, has created a petabyte scale data storage infrastructure to meet the requirements of experiments to store and access large data sets. The Fermilab data storage infrastructure consists of the following major storage and data transfer components: Enstore mass storage system, DCache distributed data cache, ftp and Grid ftp for primarily external data transfers.

This infrastructure provides a data throughput sufficient for transferring data from experiments' data acquisition systems. It also allows access to data in the Grid framework.

1. Introduction

In 1994 Run I of the Fermilab Tevatron finished with a very important result: evidence of the existence of the top quark, closing a gap in the high energy physics Standard Model. For the data storage system at Fermilab it also had a major impact. Upcoming Run II experiments would require a tremendous amount of data transferred, stored, and accessed in robotic tape libraries. Physicists needed to effectively store and access petabytes of data with data acquisition system rates. Estimates were at about 250 MBytes/s of aggregate throughput sustained for about a month of uninterrupted operation.

Existing mass storage systems were evaluated and researched and the outcome was not very promising. The majority of mass storage systems were designed basically as backup systems. They could not group stored data according to experiment needs. They did not map to the specific needs of high energy physics data processing. They could not sustain a constant data flow at a very high rate of uninterrupted work over a substantial time period. The ratio cost/effectiveness was not satisfactory. Most of the systems were strongly coupled to the hardware, reducing flexibility in the selection of machines and robotic libraries. It was also difficult to get the code modified to meet user requirements.

Looking at the systems around High Energy Physics laboratories we found a very promising prototype used at DESY in Hamburg, Germany. Based on this system Fermilab decided to develop its own mass storage system, that would satisfy the requirements of the High Energy Physics experiments at Fermilab. The goal was to create a highly reliable, scalable, flexible mass storage system having a lack of time, money, and man power. We wanted to use very inexpensive computers to run the system and have the capability to choose between a wide range of robotic storage libraries and tape drives. The Enstore project started in July 1998 and the major components of the system were developed and put into test production in approximately 1 year. Since then, the DCache data caching system, has been layered on top of Enstore. Grid ftp servers were recently implemented to provide world wide user access to the data.

2. Enstore

Enstore[1] is implemented as the primary data store for experiments' large data sets. It provides distributed access to data on tapes or other storage media. Enstore is designed to provide high fault tolerance and availability. It has a namespace which presents the storage media library contents as though the data files existed in a hierarchical UNIX file system. This restricts the use of Enstore to only on-site machines. Off-site machines can access data stored in Enstore via DCache using DCap (DCache access prtocol), ftp and Grid ftp clients. The architecture of DCache allows it to implement any access protocol.

Enstore is designed using a client-server architecture, providing a generic interface for users. The modularity of

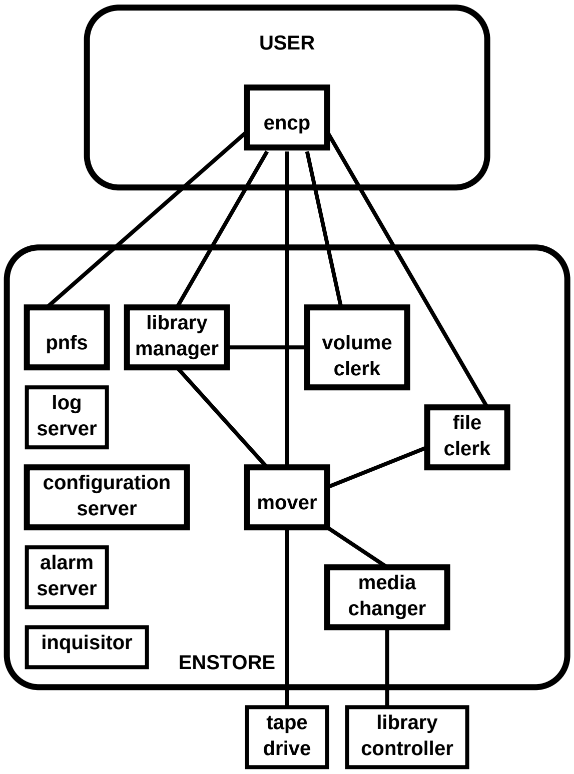

Figure 1. Enstore system

the Enstore architecture allows easy addition and replacement of hardware and software components. Enstore has the following major components (Fig. 1):

- servers
  - Configuration server - maintain system configuration information and present it to the rest of the system components.
  - Volume clerk - maintain volume database.
  - File clerk - maintain file database.
  - Multiple, distributed library managers - provide queuing, optimization and distribution of user requests to assigned movers.
  - Multiple, distributed movers - transfer data between user computers and storage devices (tape drive, disk).
  - Media changer - mount/dismount requested media in the tape storage.
  - Log server - log messages from the Enstore components.
  - Alarm Server - generate alarms upon requests from Enstore components,
  - Accounting Server - under construction and is intended for maintaining the accounting database.
- namespace - implemented by the PNFS package from DESY.
- encp - a program used to copy files to and from tape libraries.
- monitoring system (includes the Inquisitor)
- administration tools

The number of user computers is unlimited, and the Enstore system can run an unlimited number of physical tape libraries and tape drives. Components are connected via IP, and great care has been taken that the system will function under extreme load conditions. Like TCP, the system is architected with a distributed, end-to-end approach to reliability. Each request originating from an encp program is branded with a unique i.d.. The system can instruct encp to re-try if there is an internal error.

Enstore supports both automated and manual storage media libraries. This allows for a larger number of storage volumes than slots in the robotic storage. The file size to be stored in Enstore is practically unlimited (limited by the size of a tape).

Users can search and list contents of media volumes as easily as native file systems. The stored files appear to the user as though they exist in a mounted UNIX directory.

The encp program has a syntax similar to the Unix cp command with some additional options, allowing users to specify request processing parameters such as priority, number of retries, whether to calculate CRC, to set a tape dismount time, etc.

The namespace is implemented using PNFS[2] from DESY. PNFS is a database used by Enstore for keeping the metadata of data stored in Enstore. Externally it looks like a set of Unix network file systems, which must be mounted on the machine where the user application runs, thus allowing Enstore access for on-site users only. Off-site users can send read/write requests via ftp to a disk caching system DCache and DCap protocol.

Enstore stores its operating configuration in a file which is distributed to many components of the system. This file allows user specification of many features of the running system including individual server parameters. Alterations to these parameters may be made and downloaded to a running system, and will not disturb operations.

Data stored in Enstore are grouped based on the storage group unique to each experiment, and file families inside of the storage group. The storage group is assigned by the system administrator while the file family is assigned by the user. Files in the same file family are written to the same set of tapes moderated by a file family width. The file family width controls the amount of simultaneous write transfers for a certain file family.

Enstore allows users to specify the priorities for data transfer. There are 2 kinds of priorities: regular and administrative or Data Acquisition (DAQ) Priority. The library manager will dispatch DAQ priority requests ahead of any regular priority requests. In this case the mover may dismount the currently mounted tape and mount the DAQ one. Priorities can be assigned to requests according to configuration parameters in the Enstore configuration file.

Another important feature of the system is its capability to specify the number of tape drives dedicated to an experiment (storage group). Experiments have separate budgets. Some of them may have their own tape drives installed into the general purpose robotic library. These drives are in the common pool but the experiment will preferentially be given access to the number of drives equivalent to its contribution. This amount can be specified in the configuration file and is used while processing the request queue.

Enstore has a powerful monitoring system that monitors a large set of hardware and software components:

- states of the Enstore servers
- amount of Enstore resources such as tape quotas, number of working movers
- user request queues
- plots of data movement, throughput, and tape mounts
- volume information
- generated alarms
- recently completed transfers
- computer uptime and accessibility
- amount of resources used (memory, CPU, disk space, etc).

The monitored information is available on the web in a set of static and dynamic web pages published on the Enstore web site at http://www-hppc.fnal.gov/enstore/. This includes the presentation of the state of the Enstore system on a single web page, useful for operator monitoring. Monitoring of the system does not effect system performance or data throughput.

Several web pages are dedicated to displaying the details of the user request queues. This information allows users to determine the status of their requests without additional assistance.

When an error is detected, servers can be automatically restarted. Otherwise a detailed e-mail message is sent to the Enstore administrator mailing list. Severe problems will result in an administrator being paged.

Currently the Fermilab Enstore Mass Storage System is represented by 3 independent systems:

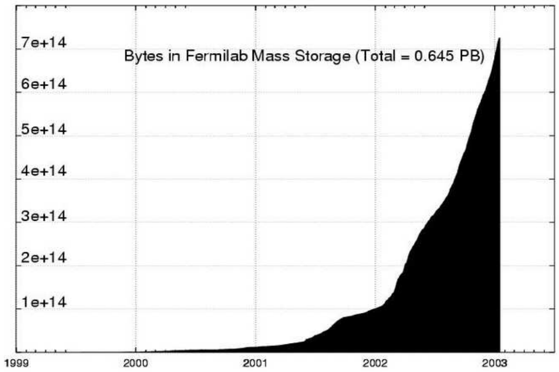

Figure 2. Total Bytes stored in the Enstore

- D0en - for the D0 experiment
- CDFen - for the CDF experiment
- STKen - for the rest of the Fermilab user community

Altogether they have 1 ADIC and 4 STK robotic tape libraries with 28 STK 9940A and 12 STK 9940B tape drives, 9 IBM LTO tape drives, and 8 STK 9840 tape drives. Ten STK 9940B tape drives will be installed soon, and 9840 and 9940A tape drives will be phased out. Data transfer rates between clients and mover nodes are equal to the rates of the tape drives (assuming the network is faster). Currently the total amount of data stored in permanent storage is more than 600 TB with an average daily transfer rate of about 8 TB, and a maximum of more than 13 TB. Figure 2 shows the total amount of data stored in Enstore since it has been in production, not excluding deleted data.

3. DCache

The next element in the data storage hierarchy is the DCache data caching system[3]. It allows on-site and remote users to store and retrieve their data in the underlying data storage system. DCache itself uses disk as a temporary data storage. Data written into this temporary storage eventually migrates to the permanent data store. The big advantage of using DCache is the rate adaptation and buffering of the data. DCache can sustain very high aggregate data transfer rates, while Enstore depends on the integral rate of the tape drives configured into the system. DCache uses Enstore as a lower layer to permanently store and retrieve data written on disks. It also serves to adapt the potentially slow network transfers with fast storage transfers and implements deferred data write staging and read ahead requests.

DCache runs on on-site machines. Data files uploaded to DCache from a user's remote machine are stored on highly reliable RAID disks pending transfer to Enstore. Files that

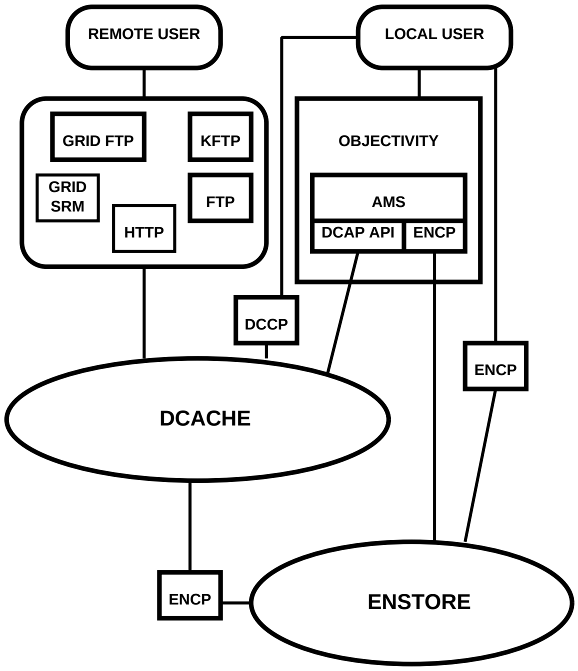

Figure 3. User connectivity to Fermilab data storage

get downloaded to DCache from Enstore are stored on ordinary disks pending further download to the user's remote machine. User connectivity to Fermilab data storage is shown in Figure 3.

DCache uses servers called "doors" to communicate with user client applications. Currently there are the following connectivity possibilities:

- DCap protocol supported by dccp utility and libdcap.so. Posix IO is available.
- Http, Grid SRM and Grid FTP including mode E parallel transfer, supported by DCache.
- Objectivity AMS server with a DCache and Enstore interface.
- Kerberized and certicate based (GSS, GSI) ftp for read/write access.
- Standard ftp for read-only access.

There are 4 independent DCache systems configured for use by the major Fermilab experiments and collaborations as well as for general use. The overall capacity of the system is about 100 Terabytes of disk space on more than 150 nodes, and is rapidly growing with the purchase of new file servers. Both Enstore and DCache systems use inexpensive computers, typically Pentium processor based PCs. There also are some Sun file servers for DCache write pools. DCache monitoring is available over the web at http://www-dca.fnal.gov and is currently undergoing extensive development in accordance with user requirements. It includes system utilization and current status as well as some useful graphical information.

4. Future Development

A new component implementing permanent storage on disks in Fermilab farms will be added soon. Fermilab farms consist of hundreds of nodes, where CPU intensive calculations are performed. These nodes always have a substantial amount of unused disk space that can be used for permanent data storage. The system which implements such functionality was developed and is currently used for temporary storage. This system uses replicas for assuring data integrity on the disks, which can not themselves be considered as permanent storage devices. It will be modified to provide the functionality necessary for implementing permanent data storage. Integration of this system will allow users very fast access to data without the need to stage files as is the case of data stored on tapes.

5. Conclusion

The Enstore mass storage system and DCache data caching system have been successfully used at Fermilab for several years. They are built using inexpensive hardware, are robust and highly fault tolerant. Administration of these systems is easy. They have a flexible distributed configuration allowing for the addition of new components without restarting the system. They easily scale satisfying Fermilab throughput requirements. Integration of the DCache system into a Grid environment allows users access to the data stored in Fermilab from any location.

References

[1] J.Bakken et al. Enstore Technical Design Document, http://isd/enstore/design.html

[2] P.Fuhrmann, A Perfectly Normal File System, http://www-pnfs.desy.de/info.html

[3] M. Ernst et al. DCache, a distributed storage data caching system, http://www-dcache.desy.de/chep2001/talk-4-005.pdf