

Overview of Storage Services

The Data Movement and Storage (DMS) department provides nearline and archival services for storing scientific data. We provide both direct tape based storage and tape storage with a disk cached front end for lower latency access to frequently accessed files. We assume tape storage to be permanent and provide custodial level of service for data on tape (see below for a description of the custodial service we provide to maintain data integrity). DMS will **not** provide storage for any data with confidentiality or privacy concerns.

Our tape storage system uses LTO tape technology with the capacity of tens of Petabytes and aggregate I/O bandwidth between 5 and 10 GB/s. Direct tape I/O can be performed on-site with our enstore software. Off-site access to tapes must be performed through a disk caching front end to the tape system. This disk cache also provides low latency access to frequently used files both on and off site.

For disk caching we currently use the dCache system. dCache uses a LRU eviction policy, and if a file is not resident in dCache, it will automatically stage it in from tape. With dCache you can remotely access data by several protocols: GSI ftp (aka grid ftp), kerberized ftp, and SRM. dCache also has a posix-like protocol, called dcap, that can be used on-site for low latency posix-like I/O.

DMS provides a shared Public dCache space for experiments that don't require a lot of dedicated disk space or long retention periods. DMS will purchase and provision dedicated disk space on behalf of an experiment by agreement.

Making a Request for Service

Requests for scientific data storage services should be made by creating a service desk ticket. The Computing Division will arrange for a meeting between the Computing Division and the experiment or organization (hereafter referred to as "experiment") to discuss details, service level agreement and cost. Please include as much of the following information that you can in the service desk request:

1. Experiment or organization
2. Experiment contacts (for permissions to export pnfs and recycle tapes with all deleted files, and for notification of planned and unplanned disruptions to service)
3. Email addresses for notification of planned and unplanned disruption of services

4. Desired storage group (a short name, usually the experiment number or name, e.g. “minos”)
5. The amount of tape storage requested (in Terabytes). We request that you provide and estimate for the current fiscal year and the following two years.
6. Range of file sizes. It is important to know if you are writing files > 8GB, as files greater than 8GB require special considerations.
7. Whether on and off site access is required
8. Whether disk caching is required.
 - a. What your maximum disk working set size would be (in Terabytes)
 - b. What lifetime the working set would need (in days)
 - c. How many simultaneous users will be accessing files in the cache
9. VO information, including mapping to UID/GID
10. Whether two copies of the data at different Fermilab sites is required (see custodial service section below)

It is understood that it may not be clear what of the above information (e.g. mappings of VO/roles to UID/GID) is needed until after meetings between DMS and the experiment are held.

Things You Should Consider About File Size and Access Latency

Latency

When a request to read or write is made to the enstore tape management system, the tape must be mounted and threaded to its load point, then a seek to the location of the file on tape is made. The mount time depends on the load of the system – the quickest time to load is about 45 seconds. A seek from the beginning of tape takes about 50 seconds on average and a seek from one location on tape to another random location is about 33 seconds. On average, you should expect the best time to access a file on tape to be about 2 minutes. LTO4 drives stream at 120 and 60 MB/s.

For reads enstore queues requests for files on a tape in the order they are on the tape. Best performance is achieved for consecutive files.

For writes best performance is achieved by queuing up enough files to make the mount time insignificant, and for file sizes large enough to keep the tape drive streaming for a significant amount of time.

Using the front-end disk cache can significantly reduce latency for frequently accessed files.

File size

It takes about 2 seconds to write a file mark. File sizes should be large enough that their time of transfer is significantly more than 2 seconds. For an LTO4 tape and a 2.5 GB file at 120 MB/s, the file mark overhead is 10%. An LTO4 tape (800GB) can hold 320 2.5 GB files. Just writing the filemarks alone takes around 11 minutes for this file size. Compare this to writing 1600 500MB files on a tape: the time spent writing filemarks is nearly an hour. That is inefficient use of limited resources and it also results in significantly more wear and tear on the tape drives.

File sizes should be large enough to keep a drive streaming for a significant amount of time, and be supplied at a rate at which the drive can stream (> 60 MB/s).

We ask that experiments not write small files (according to the above guidelines). If the experiment has small files they should tar them up into files large enough not to impact performance, availability of resources, and tape drive life. Note though that we are working on an internal file aggregator that will transparently package up small files for the user.

Storage Group and File Families

Each experiment or organization needs a storage group for writing to tape. This is usually a short experiment name (e.g. "minos" "CMS" "CDF"). The storage group is used for accounting and also for operational functions such as assigning tapes

Enstore has the concept of file families. File families are "tags" that the experiment can specify in the pnfs namespace used by enstore. Each tape volume being written to has a volume family (storage group + file family) associated with it. Only files with the same volume family can be written to the tape. In addition, file families can have file family widths (with a default of 1) associated with them. If the file family width is N, then enstore can write up to N tapes with that file family in parallel.

While file families are useful tools for grouping together similar files on tape, their use must be considered very carefully. If an experiment has too many file families, many of the tapes may never be filled. This has two consequences: wasted storage space and, since the tapes are never filled, the tapes will not have their write protection tabs locked, leaving them more vulnerable to accidents.

Files written to dCache can be directed to certain pools based on the storage group or volume family (and also by host name, read/write and other criteria). Pool selection criteria on the Public dCache system will be configured by DMS in consultation with the experiment.

Custodial Service for data on tapes

By default, the Computing Division assumes the retention period for data on tape to be permanent and that the owners of the data will bear the cost of maintaining the integrity of this data. The cost includes maintenance costs, operation costs and the costs associated with migration to newer and/or denser media.

Enstore has numerous features built in to maintain the integrity of data on tape, and the Storage Services Administration group has procedures to maintain the integrity:

1. Enstore uses end-to-end checksumming to ensure the data of each file has remained unchanged. This checksum is generated at the user's end by encp, calculated at the tape mover (or dCache) receiving end before being written to tape, is stored in the file's metadata, and is calculated on every read from tape, as it goes out over the network, and on the user receiving end of encp. This assures that the data written to enstore is exactly the same as retrieved from enstore.
2. When a tape gets filled, it will be ejected and have its write protection tab physically locked to protect against accidental erasure. Jobs to write lock batches of filled tapes are generated automatically when enough filled tapes have accumulated, or a week has passed, whichever comes first.
3. Tapes are randomly sampled and the first, last and a random middle file on the selected tapes are read (and implicitly their checksum verified).
4. The number of mounts for each tape is maintained in a database and alarmed if a threshold is crossed. The Storage Services Administration group "clones" the data to a new tape in response to these alarms. This protects against data loss due to tape wear.
5. Data is refreshed on an approximately 6 year cycle by migrating to new, denser, media. (New generations of media that are about 1.9 times denser are rolled out around every 3 years., so we migrate to 3.5 times denser media every 6 years.)
6. Since tapes and tape drives are mechanical devices, (very) infrequently tape drives can damage tapes, a tape cartridge can fail mechanically, or a portion of the media can be damaged from wear. In these cases it usually requires vendor intervention to read past the bad spots on the tape to recover the data. The Storage Services Administration group sends damaged tapes to data recovery vendors to recover what they can from the tape and restore the recovered data to new tape.
7. There are tape libraries at the GRID Computing Center and the Feynman Computing Center, which are separated by about a mile. Users can maintain two geographically separated copies at these two centers. They can do this on their own, or use an enstore feature to do this.
8. We are just starting to monitor tapes at the level of soft errors (recovered reads or writes) and will be determining if we might use this information to proactively detect tape integrity issues.

In addition to the above precautions, an experiment can arrange to have their data automatically duplicated between tape libraries located in buildings separated by about a mile. This provides protection against tapes getting damaged by drives and against a disaster at one of the sites. It is highly recommended that two or more copies be made for experiments with small amounts of data (such that losing a tape constitutes a significant statistical loss), since rarely a tape drive will damage a tape or a tape cartridge will fail.

DRAFT