Brookhaven National Laboratory

U.S. DEPARTMENT OF ENERGY

Tape Discussion
CERN 24/1/2023

Shigeki Misawa
US ATLAS Tier 1/SDCC

January 24, 2023

@BrookhavenLab

Tape Discussion
CERN 24/1/2023

Shigeki Misawa
US ATLAS Tier 1/SDCC

January 24, 2023

Purpose for Meeting

- Share information on tape system optimization techniques used at different sites with tape
- Stimulate discussion on potential techniques that may improve utilization of tape systems
- Enumerate the prerequisites for implementing each optimization technique (site dependent)
- Develop a formal proposal to find and evaluate new optimization techniques

Tape/Tape System Limitations

- Quantized access bandwidth
  - ~400 MB/sec per tape drive
  - ~400 MB/sec per tape
- Small single tape access bandwidth to capacity ratio
  - ~14 hours to read/write tape from end to end
- Sequential access media
  - Non trivial tape head seek time - measured in seconds to tens of seconds
- Serpentine media
  - Logically close on tape may not translate to physically close on media
- Long tape cartridge mount/initial seek/dismount time
  - 1 to 2 minutes
- Limited storage stack data buffering capacity

Detailed discussion of these limitations are in presentations made at multiple venues.

Simple Tape Ingest Service

- Ingested files treated as independent, opaque blobs
- FIFO algorithm used to send files to tape
- Write tape from end to end in one pass
- Multiple tapes written in parallel to achieve aggregate bandwidth requirements
- Mechanism for writing data to multiple tapes at the discretion of the tape system. Possibilities include
  - Stripe a file over multiple tapes
  - One tape per file
- On tape format of data depends on tape system capabilities and configuration

Simple File Staging Service

- Data consumers are oblivious to how data is stored on tape
- Requests for files from data consumers treated as uncorrelated
  - A consequence of a simplistic FIFO file staging system
- Without further information, correlation among file requests must be inferred from the stream of file requests
- Ability to correlate file requests dependent the local storage stack

Writes are significantly easier to optimize compared to reads.

Optimizing the Tape System

- Efficient use of tape resources is necessary for the success of ATLAS
- Increased ATLAS demand for data on tape makes optimization more important than ever.
- Optimization requires the participation of multiple parties
  - ATLAS - Provide information about data and change behavior (if possible)
  - “Middleware” providers - Alter “middleware “ to enable optimization
  - Facilities - Identify and enumerate optimization opportunities and implement optimizations
- Coordination between tape facilities needed to maintain clear communication with ATLAS and the middleware teams

No "*Lingua Franca*"

- Communication among tape sites requires a base set of shared concepts
  - Tier 1 tape drives sits at the "bottom" of a complex storage stack
    - Tape drives are typical front-end by one or more disk layers
  - Different sites use different software in the storage stack
    - Software differ in capabilities and limitations.
  - Even if software is the same, version and configuration differences can result in dramatically different system capabilities and limitations
- Discussion can rapidly devolve into conversations that require a detailed understanding of the tape system(s) involved
- First focus on the optimization, then how it can be achieved.

Optimization Techniques

- For a given tape system, a specific optimization technique may or may not improve system performance
- Some optimizations may already be in place at some sites
- Implementing an optimization may require changes by ATLAS, middleware or site storage stacks
  - All necessary modifications may not be possible in all cases or at all sites
  - Modifications at each level may differ by site for the same optimization
- Not all optimizations may be viable, allowable or improve performance
- Optimization to improve one aspect of system performance may degrade other aspects of system performance

Example : Write fewer but bigger files

- Goals :
  - Improve read performance by reducing tape head seeks
  - Improve write performance by reducing “cost” of writing tape marks
- Requires changes by ATLAS that may not be viable for all classes of data
- No effects with certain types of tape systems
  - Those that utilize buffered tape marks
  - Those that support and are configured for small file aggregation
  - Those that don’t utilize files as a unit of data storage
- Characteristics of optimization may change over time
  - Criteria for “bigger” changes as tape drives get faster

Optimization requires information

- ATLAS file requests are mostly by dataset
  - But only files not found on disk are requested from tape
- Snapshot of distribution of dataset sizes are known
- Time to receive all files in a dataset at a tape site varies
  - Time window distributions are known with some granularity
- Rucio contains detailed information on all files in a dataset that a tape site may receive
  - e.g. # files, file sizes, file names
  - All datasets are closed before any files they contain are transferred to a tape site [1]

[1] Raw data from detector was previously an exception, but this is no longer true

Optimizing Reads vs Writes

- End to end writing of data to tapes with a FIFO algorithm using multiple tape drives is optimal
- Techniques used to optimize reads need to minimize their impact on writes
- Optimizing reads require information about read patterns
- Read performance is dependent on how data is written in addition to how data is read
  - Write data as it will be read back, read data as it was written

Possibilities

- Enumeration of some possible optimization techniques
  - Some may be in use by sites
  - Efficacy of some methods open to debate
  - Other techniques are likely to be found
- Investigations to find other optimizations
  - ATLAS data generators/consumers may be a source
  - ATLAS data management system may also be a source
  - Analysis of site storage stacks including tape systems are another source
  - Analysis of data itself is another source

Optimization Techniques

- Sort file requests by tape
  - Minimizes tape mounts
  - Chances for optimization increases with # queued stage requests
  - Storage system must be able to handle volume of queued requests
  - Access latency increases with queue depth. Upstream ramifications?
    - Long tails may be a problem
  - Any time ordering of requests will be scrambled
- Sort request for files on a tape by tape order
  - Minimizes distance tape head move to read all requested data
  - Require RAO/oRAO capable drives or similar mechanisms

Optimization Techniques

- Write files in dataset contiguously on tape
  - Reduces tape head seeks when reading back dataset
  - Must be able to identify files in a dataset
  - Must be accompanied by file read requests by physical or logical tape order
- Write files in dataset to as few tapes as possible
  - Reduces tape mounts when reading
  - Increases amount of data read per tape mount
  - Potentially reduces demand for tape drives
  - But max read and write bandwidth is limited by # tapes used
    - May need to articulate access bandwidth requirements

Optimization Techniques

- Write dataset to tape only after all files in dataset have been received
  - Reduces tape mounts when reading
  - Requires sufficient staging space to hold all “inflight” datasets
  - May not be possible for all datasets
- Prefetch all files in dataset
  - Assumes access is by full dataset
  - Effects on disk staging area unclear

Investigations/Analysis

- Effectiveness of data placement techniques
  - Use simulations of writing into storage system to determine following:
    - Effectiveness of different data placement strategies on achieving desired layout
    - Determine impact on overall write throughput to tape
    - Determine impact on system requirements
      - Impact on disk buffer capacity/performance
      - Impact on effective write throughput to tape
    - “Re-play” write logs into storage system since the start of Run 3 to get an accurate simulation of the ingest environment. (Utilization of trace log replay techniques used by file system developers to analyze performance of files systems)

Investigation/Analysis

- Estimate impact of data placement on reads
  - Assume data is laid out on tape as desired
    - Or use results of write placement simulation
  - Use simulations of reading from storage system to determine if “optimal” data placement has the desired impact
  - “Re-play” read logs into storage system since the start of Run 3 to get an accurate simulation of the read environment
- Use above read and write simulations to gauge impact of ATLAS changing dataset write profiles (e.g., measure effects on reducing transfer window length)

Investigation/Analysis

- Use historical file staging logs to analyze impact of full dataset prefetching
  - Determine ratio of “hits” vs “misses”
  - Probe effectiveness of different prefetch decision algorithms
  - Investigate impact on cache to hold prefetched files
- Investigate segregation of classes of datasets onto separate media
  - e.g. Put main stream RAW data onto a dedicated set of tape cartridges
  - Examine ramifications of fine grained segregation

Investigation/Analysis

- More detailed/continuous dataset characteristic analysis
  - Transfer window distributions (by “class” and over time)
  - File size distributions within datasets (by “class” and over time)
  - Dataset size distributions (by “class” and over time)
  - Verify datasets are really “closed” or if they might get “reopened”
- Dataset correlation analysis
  - Examine dataset request logs to determine if groups of datasets (“retrieval group”) are retrieved together
  - Determine if datasets in a retrieval group can be co-located on tape(s) and if there is a benefit of co-location

Investigation/Analysis

- Dataset segregation
  - Investigate segregation of classes of dataset onto dedicated media
  - Identify candidate classes of data (e.g. main stream RAW data)
  - Determine impact of class granularity on both read and write performance
  -

Summary

- Several optimization techniques are known
  - More are likely to be discovered
- Additional information about the environment and the data are needed to configure possible optimizations and to discover more
- Information sharing and coordination of investigations among sites can help reduce the effort needed to identify, develop, and deploy new optimizations
- Once useful optimizations identified coordination with ATLAS and middleware developers is needed to make it possible to implement the optimizations