Storage and Data Management Plans

Robert Illingworth

Muon g-2/SCD Computing Workshop

2 March 2022

Storage and Data Management Plans

- dCache will remain the primary system for experiment data

- Will transition Enstore tape management to new system – CTA

- SAM Data Management is essentially feature frozen
  - Future is Rucio/MetaCat

- Investigating possible alternative for interactive use (BlueArc/NAS equivalent)
  - Early stages; no promises

Data Management

- SAM is g-2’s data management replica and metadata catalog
  - 43,241,372 files with metadata (including 5,291,760 retired files)
  - 43,215,800 file locations
  - 367 GB database size (about a quarter of the largest SAM DB)

- SAM is essentially feature frozen at the moment – we address bugs and have implemented a few minor features, but we are not actively developing it.

- Activity has moved to newer projects: Rucio, MetaCat, etc
  - Targeting new experiments
    - DUNE
    - SBN
    - etc

SAM replacements

- SAM is a monolithic system
  - Metadata catalogue
  - Replica catalogue
  - Workflow integration (SAM station)
  - Data upload tool

- Replace with separate components; use external products where it makes sense
  - Provides more flexibility
  - Common HEP (and beyond) solutions benefit everyone

Rucio

- Data management system developed by ATLAS, now used by CMS, Belle II and either in use or planned to be used by various other experiments.
- Replica catalogue; rule-based replication and data transfer
- Currently used by DUNE and Icarus in parallel with SAM; will transition fully to Rucio

- Anticipate that SCD supported method of getting data to/from HPC will be Rucio based

- For more details see:
  - https://indico.fnal.gov/event/18468/attachments/29644/36533/Rucio_for_FIFE.pptx
  - https://indico.cern.ch/event/1037922/

MetaCat

- New metadata catalogue
  - Compatible with, but not integrated with Rucio
  - Attempts to address some of the performance issues we’ve seen with SAM
    - Worse case query complexity should be bounded
    - De-emphasize frequent evaluation of queries in favour of static datasets

- Conversion from SAM metadata is possible
  - Process likely to need some customization for each experiment

- Functionality is broadly equivalent, but not API compatible

- https://docs.google.com/presentation/d/13q-belkZyUJe-XUNj6qZpZ-xv_4h_7nW1znKvB4lRPU/edit?usp=sharing

Other DM components

- Will create “data dispatcher” to replace SAM station
  - DUNE needs this for ProtoDUNE II, so work is underway
  - Not API compatible, but expect similar “get next file” functionality

- Replacement for Fermi-FTS
  - Automatically register metadata with MetaCat & upload file to Rucio
  - Will be simpler as Rucio will take over data movement part
  - New name! (Not yet decided – may be File Declaration Daemon if no-one comes up with anything better)

- User level functionality
  - This part is less settled, but expect to use Rucio features to move/archive user data
  - This would provide a way to move user output from jobs to storage

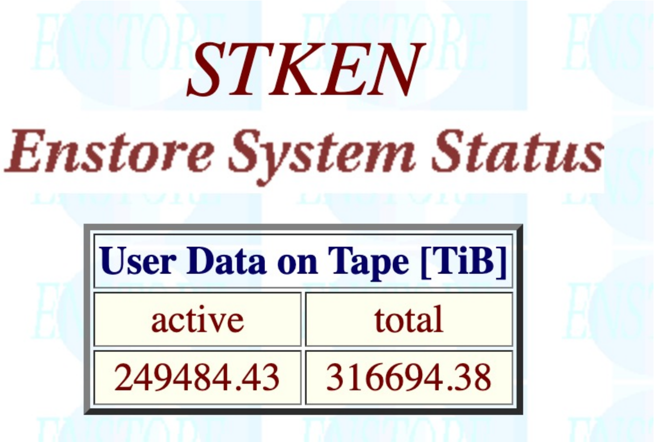
DM choices for g-2

- Carry on with SAM
  - Routine support load is now fairly low; can likely keep core services operating
  - Don’t expect improvements or new features
  - Performance could cause problems which are unlikely to be addressed
  - Changes in computer security requirements/OS/etc might require disruptive changes

- Migrate to new tools
  - Will be the supported IF tools going forward
  - Will be more scalable
  - Likely to require changes in workflows and interfaces

Enstore & CTA

- Enstore is the system used to manage the tape libraries
  - Design goes back over 20 years
  - Data volumes have increased massively in size
    - 2000: T9940A tape cartridge capacity 60 GB
    - Now: LTO-8 12 TB (LTO-9 will hold 18 TB)
  - Enstore was designed for petabyte scale; now at 250+ PB and looking towards exabyte for HL-LHC
- Enstore is already showing some scalability issues and we have concluded it would require significant development effort to continue



STKEN

Enstore System Status

User Data on Tape [TiB]	
active	total
249484.43	316694.38

CTA

- CERN Tape Archive https://cta.web.cern.ch/cta/
  - CERN previously used a locally developed system called CASTOR
  - Similar concerns about scalability
  - Decided to develop a new system for exabyte scale
  - Deployed in production at CERN starting June 2020
  - Also adopted by RAL in the UK (previous CASTOR site)
  - Under consideration by DESY

- We have decided to adopt CTA, with modifications necessary for our environment
  - Principal change is to enable reading of our existing data
    - Conversion should be a metadata (database) change only
  - Second concern is how to handle SFA
    - Right approach still under consideration

What does this mean for experiments?

- Plans for CTA still under development

- For most people not much will change
  - Uploading and staging files will still be via SAM/dCache/Rucio interface (see later)
  - Few people interact directly with enstore commands now, and I don’t expect that to change

- Likely new monitoring and information

- Support and development will be spread over more people in more institutes
  - More viable long-term support

New tape library purchase

We are in the process of purchasing a new tape library as the existing public libraries are approaching capacity.

The pictures show last year's new CMS library. (Not necessarily indicative of what we buy next)

Data lifetimes

- Data archived on tape is an ongoing cost – even if it's not being accessed
  - Requires space in the tape libraries and every few years must be migrated to a new generation of tape media

- It's an ongoing struggle to get experiments to consider the useful lifetimes of datasets, *and to delete them when they are no longer useful*
  - Deleting datasets potentially allows us to recycle tapes for other purposes, or to free slots in the library
  - At the very least it's data we know we don't have to eventually migrate, which reduces the long term cost

dCache

- Evolutionary development
  - lots of internal improvements I’m not going to discuss here

- Protocol changes
  - Gridftp deprecated; no immediate plans to disable it
  - https/WebDAV & xrootd are the preferred protocols

- Authentication changes
  - X509 auth is going away
  - Tokens are implemented
    - Tokens work by capabilities **not** by identity
    - A token authorized transfer can only write to the paths permitted by the token regardless of existing directory ownership
    - May require reorganization of some user directories

dCache quotas

- Quota feature now available on dCache
  - Not enabled by default
  - Group and user quotas available
  - Can put separate quotas on persistent disk usage and tape usage
    - But since this seems to confuse people: you cannot use quotas to control tape cache usage

- Primary use case is to control persistent usage
  - Keep users under control…

- See https://indico.fnal.gov/event/51486/ for more details
- Make a servicedesk request if you are interested in pursuing this

Tape staging performance

- We know staging performance has been an issue
  - We're never going to get enough tape drives or disk space to keep everybody happy, but there is scope to improve the efficiency of what we've got

- A reoccurring problem is due to poor request queuing between dCache and Enstore
  - dCache is not tape location aware and each pool has an independent queue of files to stage
  - To avoid overloading Enstore the number of requests dispatched to it must be limited
  - Consequence is that dCache sends an unordered subset of files from each pool to Enstore, which then can't efficiently order tape accesses
- Too many unnecessary mounts and dismounts; too much spooling back and forth of tape

Tape staging performance improvements

- Switching to Enstore to CTA may help with this as it should be able to accept more pending stage requests
  - Doesn’t solve all the issues

- We are investigating making dCache more tape aware
  - Fetch location on tape; direct requests for same tape to the same pool
  - Grouping requests like this should make tape accesses more efficient

- Seen benefits of similar approach while optimizing the media migration process
  - As much as 4x throughput (real world results may vary)
  - Semi-manual implementation
  - Needs to be built into standard staging process

- Planning to try this in the relatively near term

Disorganized file staging

- The dCache r/w disk works as a cache (dedicated or shared)
  - Files are staged to disk on access (very inefficient) or an explicit request
  - Dropped from cache according to LRU algorithm

- Current situation is simple but effectively a “free-for-all”
  - User access can be very disorganized
  - Production type workflows are better, but still vulnerable to other experiments (especially on the shared cache)

- We encourage a semi-manual process with “prestage” SAM tasks
  - No control when files drop out of cache

Organized file staging

- We want to move large experiments towards a more controlled system
  - Require files to be staged and unstaged explicitly

- The SAM tools are not designed for this; will really require Rucio

- More than one approach
  - Split out a separate staging pool (not accessible to users/jobs) and require the experiment to transfer the data to the disk pools before use – the CMS configuration
  - Turn off implicit staging and require explicit dCache QoS (Quality of Service) transitions to make the data available (not supported by Rucio yet, but it is planned)

- Is g-2 a large experiment for this purpose
  - Requires more active involvement from experiment computing to decide what deserves to be on disk

Scientific NAS replacement?

- We’re aware that the functionality of the Scientific NAS (“BlueArc”) is popular
  - POSIX filesystems are convenient
  - dCache pnfs does not fully replace it
  - But the NAS is provided by an expensive hardware appliance; we do not plan to replace it when the warranty expires

- We are currently in the early stages of investigating a *possible* replacement using CephFS (https://docs.ceph.com/en/latest/cephfs/), a distributed open-source file system
- We do not expect this to replace dCache for large scale data storage; this would be for use-cases like interactive analysis of ntuples
- Needs more investigation before we can decide if this is the right approach

Conclusions

- Think about SAM->Rucio transition

- Enstore -> CTA transition should put us in a better position for the long term, but will have relatively little direct impact on end users

- dCache will remain as main data store
  - Have to keep up with protocol and authorization changes

- We're looking at ways to improve tape staging performance, but need to think about changing to a more controlled staging model

- Considering NAS alternative for use cases where dCache is less suitable