Dynamic sharing of tape drives accessing scientific data

Tape facility at INFN CNAF

Infrastructure

INFN CNAF provides storage resources for 4 LHC experiments (Alice, Atlas, CMS, LHCb) and ~30 non-LHC collaborations

- ✓ ~ 20 PB on disk
- ✓ ~ 42 PB on tape

Tape infrastructure components

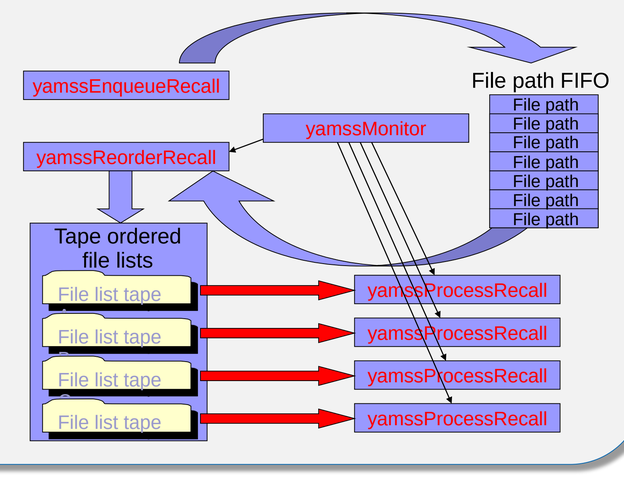
- ✓ **1 tape library Oracle-StorageTek SL8500** (10000 slots)
- ✓ **17 tape drive T10KD** for scientific data
- ✓ **GEMSS** (Grid Enabled Mass Storage System) software developed by INFN that provides a full HSM (Hierarchical Storage Management) integration of:
  - ✓ **StoRM** (Storage Resource Manager): software released by INFN based on SRM (Storage Resource Management) interface to access storage resources
  - ✓ **IBM Spectrum Scale:** the disk storage software infrastructure
  - ✓ **ISP (IBM Spectrum Protect)** software: the tape system manager

GEMSS recall workflow

- ✓ Files to recall taken by periodic scan of StoRM bring-online file list or direct user requests
- ✓ Requests enqueued by a FIFO method at first (yamssEnqueueRecall)
- ✓ Reorder of files to recall in tape ordered file lists to optimize reading and periodic regeneration in case of new requests (yamssReorderRecall)
- ✓ A recall process starts for each tape file (yamssProcessRecall)
- ✓ Supervision of the reorder and recall phases (yamssMonitor)

The maximum number of reading (yamssProcessRecall) and writing threads to send to ISP server for each HSM server is defined by 2 parameters:

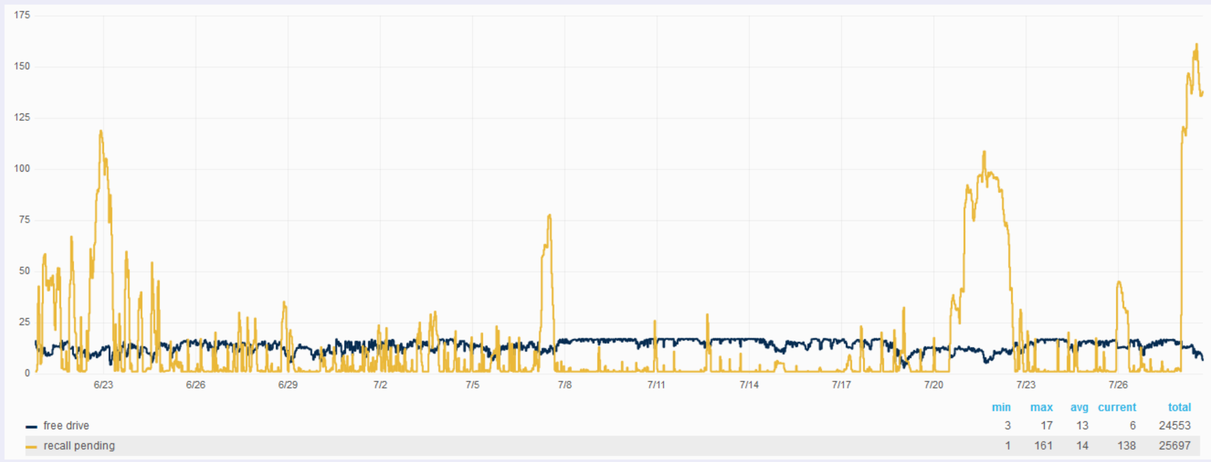
- ✓ **RECALL_RUNNING_THREADS**
- ✓ **MIGRATE_RUNNING_THREADS**



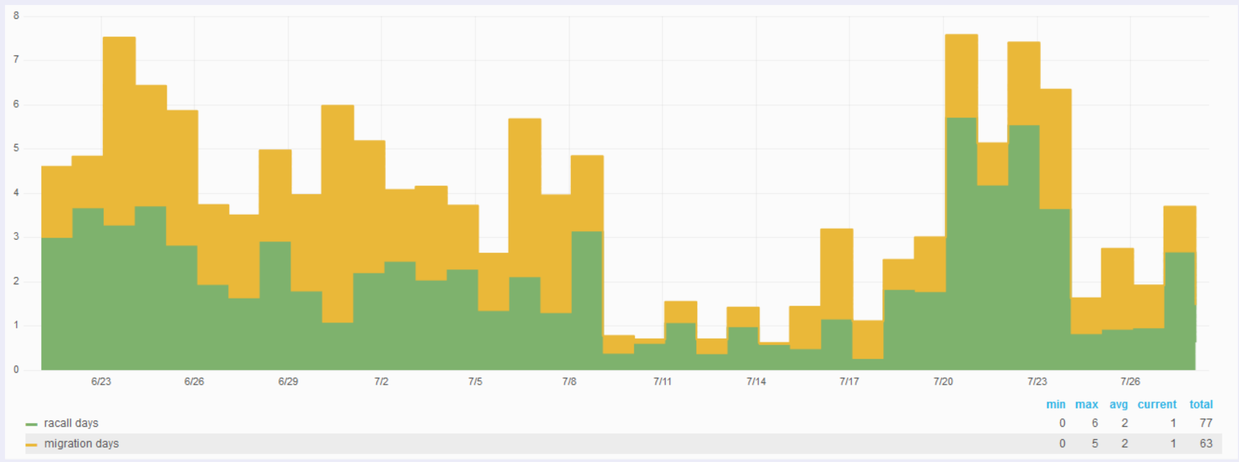
Tape drive usage

- ✓ Each experiment can use a maximum number of drives for recall or migration, statically defined in the GEMSS configuration file parameters (RECALL_RUNNING_THREADS and MIGRATE_RUNNING_THREADS)
- ✓ In case of scheduled massive recall or migration activity these parameters are manually preemptively changed by administrators
- ✓ We noticed several cases of free drives that could be used by pending recall threads (considering the limit of 8 Gbit/s on FC connection of each HSM server)

We designed a software solution to dinamically allocate additional drives to VOs and manage concurrent recall access



Total number of recall threads pending and free drives, June-July 2017. In several cases a subset of free drives could be used by recall threads.



Total duration (in days) of recall and migration processes, June-July 2017 – stacked plot aggregated by day. The total usage is never greater than 8 days (over a total of 17 drives).

Dynamic sharing of tape drives

InfluxDB stores monitoring information on:

- ✓ free drives from ISP server
- ✓ number of recall and migration threads running from each HSM server (e.g Exp1, Exp2, Exp3)
- ✓ number of pending recall threads

Orchestrator:

- ✓ performs comparison among pending threads and free drives every 5 minutes
- ✓ can change GEMSS parameter RECALL_RUNNING_THREADS on each HSM server for maximum number of recall threads
- ✓ manages the concurrent access to drives setting a dynamic priority on the basis of the following formula:

$$Exp_{priority} = \frac{Exp_{share}}{\alpha(usage\_time)+\beta(1+run\_recall)}$$

where: Exp_{share} is a static priority given to each experiment
$usage\_time$ is the total recall time used by the experiment in a certain period (e.g. last 24 hours)
run_recall is the number of recall running threads
α and β are tunable coefficients

Within an HSM server, priority can also be tuned using the RECALL_MAX_RETENTION parameter (default 1800s => ½ hour)

- ✓ *If* pending recalls threads *waiting time* ≤ RECALL_MAX_RETENTION ➜ priority to recall thread with the largest # of files
- ✓ *If* pending recalls threads *waiting time* > RECALL_MAX_RETENTION ➜ priority to FIFO pending recall threads

Tuning this parameter can avoid pending recall process starvation and can be considered for future GEMSS orchestrator implementation to provide a different priority method to dedicated recall threads.

A. Cavalli, D. Cesini, E. Fattibene, L. Morganti, A. Prosperini, P. Ricci, V. Sapunenko
INFN-CNAF Bologna (Italy)
enrico.fattibene@cnaf.infn.it