Experience in dynamic tape drive allocation to manage scientific data

A. Cavalli, D. Cesini, A. Falabella, E. Fattibene, F. Fornari, L. Morganti, A. Prosperini, V. Sapunenko

INFN CNAF, viale Berti Pichat 6/2 40127 Bologna, Italy

E-mail: enrico.fattibene@cnaf.infn.it

Abstract. The main computing and storage facility of INFN (Italian Institute for Nuclear Physics) running at CNAF hosts and manages tens of Petabytes of data produced by the LHC (Large Hadron Collider) experiments at CERN and other scientific collaborations in which INFN is involved. Most of these data are stored on tape resources of different technologies. All the tape drives can be used for administrative tasks (as repack, audit, space reclamation), as well to write and read data of all the experiments. Moreover, the usage of tape resources by scientific communities will become considerably more intense in the next years and the amount of data on tape will double by 2025. For these reasons, the issue of the concurrent access to tape drives is significant. We designed a software solution to optimize the efficiency of the shared usage of tape drives in our environment and put it in production in January 2020. In this paper we present the experience with such dynamic tape resources allocation in production. Comparing it with the previous static allocation method, we observed an improvement in reading throughput up to 85%. Moreover, we describe the new features added to our solution to optimize the efficiency of the shared usage of tape drives of different technologies.

1. Introduction

INFN CNAF operates the main data center of INFN, providing resources and services to communities involved in scientific collaborations. It is one of the 13 Tier-1s of the WLCG (Worldwide LHC Computing Grid) [1] that receives data produced by the LHC experiments (ALICE, ATLAS, CMS, LHCb) at CERN and provides computing and storage facilities for 30 other experiments in which INFN is involved, belonging to Astrophysics, Astro-particle Physics and High Energy Physics domains. Data are stored on both disk and tape resources. At the time of writing, ~41 PB of data reside on disk and ~93 PB on tape.

1.1. Tape facility and services

The tape infrastructure for scientific data at CNAF is based on two tape libraries: an Oracle-StorageTek SL8500 equipped with 16 T10000D tape drives (other 7 T10000C drives are used only for the backup and recovery service) and an IBM TS4500 equipped with 19 TS1160 tape drives. The overall capacity of the libraries is more than ~200 PB with the existing technologies. Data stored on tape are organised into IBM Spectrum Scale file systems. Tape storage is the highest latency tier within a HSM (Hierarchical Space Management) environment. Disk storage spaces are used as buffer, to enable writing and reading operations to/from tape.

In order to allow data access to scientific communities, the Storage Management group operates services based on a set of software packages:

- IBM Spectrum Scale [2]: formerly GPFS (General Parallel File System), a high-performance clustered file system developed by IBM. File systems can be partitioned into a set of storage pools implementing file placement policies and data migration rules from one pool to another according to some user-defined criteria.
- ISP (IBM Spectrum Protect) [3]: formerly TSM (Tivoli Storage Manager), a proprietary software designed by IBM. It offers a HSM extension to manage migrations from disk to tape and recalls from tape to disk of data hosted on Spectrum Scale file systems.
- StoRM (Storage Resource Manager) [4]: a software released by INFN based on SRM (Storage Resource Management) interface to access storage resources.
- GEMSS (Grid Enabled Mass Storage System) [5]: a software developed by INFN that provides a full HSM integration of Spectrum Scale, ISP and StoRM. It has been designed to optimize migration and recall operations. Recalls can be triggered by a periodic scan of the StoRM bring-online table (the table in the StoRM database that contains the files to be copied on the disk buffer). Otherwise, they can be requested through GEMSS command line or can be caused by a direct file access. GEMSS passes the list of files to be recalled to the ISP-HSM service, that groups the files by cartridge, ordered based on their position on the tape. At this point, a recall process can start for each cartridge, corresponding to a drive dedicated to that process. Among other things, GEMSS offers the possibility to customize, for each handled file system, the maximum number of recall threads running and performs periodic regeneration of tape ordered file lists to include new requests in already existing lists.

1.2. Static tape drive allocation

All the tape drives dedicated to scientific data are shared among experiments, i. e. any of them can be used by migration or recall threads running for each file system. Before January 2020 there were 16 T1000D drives in production and a static allocation method was in place. Each experiment could use a maximum number of drives for recalls or migrations, which was statically defined in GEMSS. In case of scheduled massive recall or migration activity, these parameters were manually changed by the administrators. Moreover, administrative tasks to manage data on tape, such as space reclamation or data move between storage pools, were also done manually. This could interfere with production, since tape drives busy for these tasks could not be made available to migrations or recalls unless they would be interrupted by the administrators.

By experience in administering such services, we noticed sometimes a certain number of free drives and, at the same time, pending recall threads for one or more file system. In several cases, a subset of free drives could have been profitably used to reduce the queue of pending recalls, but there was no way to dynamically change GEMSS parameters to give more drives to the appropriate file system.

2. Dynamic drive allocation

The inefficiency in tape drive allocation affected the tape system performance in terms of time needed to get data from tape to disk and time to perform the administrative tasks. Moreover, in the next years, the usage of tape drives was expected to become more intense due to the growing amount of scientific data to store and to the trend, disclosed by the main user communities, to use tapes as near-line (or "slow") disk, thereby increasing the reading traffic rate [6][7]. According to the latest projections, the required resources at CNAF at the end of 2025 will be 100 PB of disk and 220 PB of tape storage space.

For these reasons, we designed and implemented a software solution, called *Orchestrator* [8], to dynamically allocate tape drives to file systems and to manage concurrent requests.

The first version of the *Orchestrator* was implemented to manage the allocation of a homogeneous set of tape drives, since the only tape library operating at CNAF was the Oracle-StorageTek SL8500, equipped with tape drives of the same technology (T10000D) devoted to managing scientific data. GEMSS was implemented to handle a single queue of recalls for each file system. The *Orchestrator* ran at a configurable frequency (typically every five minutes) and retrieved monitoring information

from InfluxDB. Such information included the number of T10000D drives that were free or in use in a certain moment of time (taken from the ISP server), the number of running recall and migration threads, the number of pending recalls and the value of GEMSS parameter that indicated the number of recall threads that could become running for each file system (taken from the HSM servers). For each file system handled, the administrator set a maximum number of recall threads that could be reached, based on the speed limit of the Fiber Channel connection of each HSM server and the maximum native rate of each T10000D tape drive (250 MB/s). If free drives were found and, at the same time, pending requests waiting for drives for any handled file system, the *Orchestrator* established the number of drives that could be assigned to each interested file system, comparing the number of running recalls with the value of the maximum number of recall threads that could be reached. By modifying the appropriate GEMSS parameter for each interested file system, it was able to increase the number of running recall threads to reach the limit set by the administrator.

Because of the purchase of a new tape library (IBM TS4500), at the beginning of 2021 GEMSS was adapted to manage multiple recall queues, one for each set of tape drives in use. For each queue, a parameter indicates the number of recall threads that can become running. In the same way, the *Orchestrator* has been updated to consider multiple queues. For each file system and for each queue managed by GEMSS (i.e. for each set of tape drives), it considers the number of running and pending recall threads and the value of GEMSS parameter that indicates the number of recall threads that could become running. In order to optimize the usage of drives, the *Orchestrator* has been designed to consider the overall Fiber Channel traffic of each HSM server in order not to exceed its nominal speed limit. Therefore, the maximum number of drives that can be assigned to a file system is calculated considering the current number of recall threads running for each set of drives and the native rate of drives. The technologies currently used rely on native rates of 400 MB/s for IBM TS1160 drives and 250 MB/s for Oracle T10000D.

Whenever there is no concurrency among different file systems for the available resources, i.e. the number of available free drives is large enough to satisfy all the requests, the *Orchestrator* mitigates the pending requests. This is performed by means of modifying the appropriate GEMSS parameter for each interested file system on the relevant HSM server. In case of concurrent pending recalls, i.e. if the number of tape drives is not sufficient to satisfy all the requests, the system orders the file systems by the number of recalls that can become running, considering the limit given by the Fiber Channel connection. Afterwards, it assigns a tape drive to each file system starting from the one with highest number of requests and repeats this loop until the maximum number of runnable threads is reached or all the free drives are assigned. Of course, ideally one would like to lower the number of pending recall threads by filling all the available free drives except for a reasonable reserve (which we set to 2 for each of the groups of drives).

The *Orchestrator* performs on-line scheduling, so it makes decision without any knowledge about the kind and amount of workload that will come in the future.

The administrator can configure the possibility to start space reclamation processes for a particular storage pool managed by each group of drives. Space is reclaimable because it is occupied by files that are expired or deleted from the ISP database. If further tape drives are available after the assignment for pending recalls, a reclamation process can be triggered on the ISP server for a storage pool, specifying a reclamation threshold, that indicates how much reclaimable space (in percentage) a tape volume must have at least to be made available by this process. The administrator can set also the duration of such process, so that, at the end of the process, another one can be automatically triggered in case of sufficient free drives. In this way, the reclamation task is performed without the manual intervention of the administrators and minimizing the impact on the production tasks (migrations and recalls).

3. Drive allocation methods comparison

The *Orchestrator* is in production at CNAF since January 2020. In order to compare the recall performance of CNAF tape system before and after the adoption of the dynamic allocation method implemented through the *Orchestrator*, two plots taken from the CNAF monitoring web interface are

reported in figure 1. The reference period for both plots is two years from February 2019 to January 2021. During this period a unique homogeneous set of drives was in production (16 T10000D). In the first half of this period the static allocation method was in place. The red lines represent the starting point of adoption of the *Orchestrator* in production.

With the static method, the administrators used to set 3 as maximum number of recall threads for ATLAS (plot on the left), in order to leave other drives available for possible migrations or recalls of the experiments. Starting from January 2020, a significative increase in number of drives used is visible since the *Orchestrator* was able to assign free drives to ATLAS pending recalls. In the same way, the plot on the right, that reports the average recall throughput every 60 minutes, shows a significant improvement thanks to the adoption of the dynamic method, switching from a rate always under 1 GB/s to several peaks above 1.5 GB/s. Of course, this comparison is biased by the real production activity, i.e. the number of recall pending threads and availability of free drives overtime, but it gives a hint of the improvement of the performance of the system.

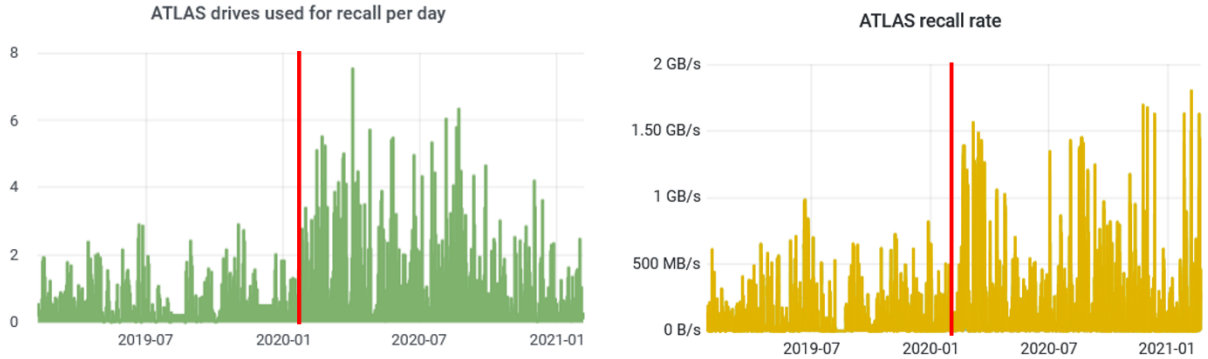

Figure 1. Number of drives used for recalls and recall rate for ATLAS in production (February 2019 – January 2021).

We performed another comparison by considering two real CMS bulk recalls with similar number of files and amount of data read, with the two different allocation methods in place. Table 1 reports details about these two sets of recalls. In this case of similar sets of data, i.e. similar number of files and amount of data read, an improvement of 85% is visible in terms of waiting time to have all data available on the buffer (duration) and of average throughput.

Table 1. Sample comparison between two real CMS bulk recalls using CNAF production tape infrastructure.

	Static allocation method	**Dynamic allocation method**
Recall period	18-23 Apr 2019	17-19 Jan 2021
Number of files	98000	92000
Data read	319.5 TB	313.5 TB
Duration	**138 hours**	**72 hours**
Average drives used	**3.7**	**6.3**
Average throughput	**650 MB/s**	**1.2 GB/s**

Referring to these CMS bulk recalls, figure 2 shows the average number of tape drives used and the amount of data read per day with both methods. Again, the capability of the *Orchestrator* to assign drives to pending recalls depends on the number of free drives overtime. There could be periods with intense activity by other experiments, so that the improvement is less important (day 2). On the other hand, in case of availability of resources, the added value of *Orchestrator* using the dynamic allocation method is relevant. The peak in the amount of data read per day jumps from 65 TB with static allocation to 135 TB with the new method.

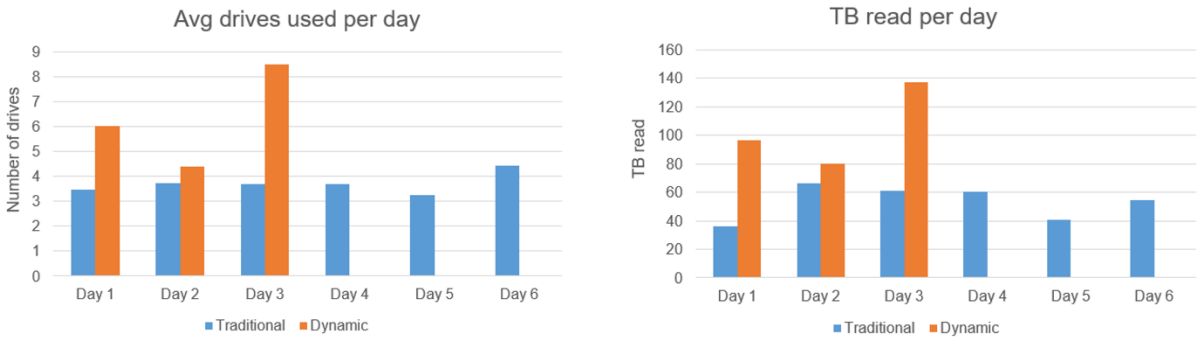

Figure 2. Average number of drives used and amount of data (in TB) read per day for two real CMS bulk recalls, with the two different allocation methods in place.

4. Conclusions

Dynamic drive allocation, implemented through the *Orchestrator*, is in production at CNAF since January 2020. It allows to decrease users' waiting time for recalls and to perform administrative tasks, such as space reclamation, without interfering with production. By a sample comparison to traditional allocation, based on real recall activity, we observed an improvement of 85% in throughput and, consequently, in users' waiting time to have all data available on the disk buffer. This system has been also improved to manage different recall queues for different sets of tape drives, by working with multiple tape libraries.

This solution is an important enhancement for CNAF mass-storage facility in view of the future growth of writing and reading rate. Moreover, the ability to maximize the drive exploitation will help CNAF in lowering hardware purchase, by reducing the number of tape drives to buy in the future, as compared to the static allocation method.

References

[1] WLCG home page: https://wlcg.web.cern.ch/
[2] IBM Spectrum Scale web page: www.ibm.com/systems/storage/spectrum/scale
[3] IBM Spectrum Protect web page: www.ibm.com/systems/storage/spectrum/protect
[4] R. Zappi R et al, *An efficient Grid data access with StoRM*, S.C. Lin and E. Yen (eds.), Data Driven e-Science: Use Cases and Successful Applications of Distributed Computing Infrastructures (ISGC 2010), Springer Science + Business Media, LLC 2011
[5] Bonacorsi D et al., *The Grid Enabled Mass Storage System (GEMSS): the Storage and Data management system used at the INFN Tier1 at CNAF*, 2012 J. Phys. Conf. Ser. 396 042051 Proceedings of 2012 CHEP conference.
[6] ATLAS HL-LHC Computing Conceptual Design Report: https://cds.cern.ch/record/2729668
[7] Evolution of the CMS Computing Model towards Phase-2: https://cds.cern.ch/record/2751565
[8] Cavalli A, Cesini D, Fattibene E, Morganti L, Prosperini A, Ricci P P, Sapunenko V; *Dynamic sharing of tape drives accessing scientific data*; 2017. In the proc. of the 18th International Workshop on Advanced Computing and Analysis Techniques in Physics Research (ACAT 2017), August 2017, Seattle, USA