U.S. Department of Energy Best Practices Workshop on
File Systems & Archives
San Francisco, CA
September 26-27, 2011
Position Paper

Todd M. Heer
Lawrence Livermore National Laboratory
theer@llnl.gov

ABSTRACT / SUMMARY

Specific to our center, archival data integrity emanates from dual copy files, intensive preproduction environment analysis, and ongoing HSM verification testing. Availability falls from the judicious application of a redundancy model. Efficiency can be obtained by leveraging the large procurements part and parcel of HPC center operations as well as the thinning out of unnecessary costly equipment.

A newly implemented soft quota model constrains growth. Flexibility and communication with users ensure success.

INTRODUCTION

The deployment and administrative tasks of an HPC data archive tax credos of data integrity, service availability, and operational efficiency. Couple that with the specter of prodigious growth, and you have a witches' brew of daunting missions.

DEPLOYING TO OUR CREDOS

I. DATA INTEGRITY

Arguably, the ultimate responsibility of an archive is to protect the data.

- ***Dual Copy, Dual Technology***

Data integrity is achieved by dual copy of files over a specific size range. It is often sufficient to simply have dual copies of a file, unless a specific underlying technology is the source of the problem (e.g. firmware bug causing corruption on a data pattern). In this case, a differing technology must store the second copy.

We dual copy over two tape drive technologies in order to avoid such scenarios. Currently these technologies are Oracle T10000C and IBM LTO-5.

The recent leap in capacity resulting from the barium-ferrite particle of the T10000C media realizes an average of 7.9TB per cartridge with our customer data profile. We have recently been afforded the opportunity to dual copy all files up to 256MB as a consequence. We offer a special class of service customers can specify to obtain dual copy files on tape regardless of size.

Each technology is further separated in two distinct robotic library complexes (Oracle SL8500) separate by a distance of approximately 1 kilomoter.

- ***Offline Testing***

As tape drives are either purchased or replaced due to failure, they are tested for integrity and performance before being placed into production. A suite of tools was created to facilitate this out-

of-band testing. Files of known size and composition are written to and read from test media. Timing is conducted and data is examined by means of a checksum. It is important to understand that a performance threshold exists below which drives should be considered faulty for the environment, even if integrity checks pass.

- ***End to end verification***

The largest stride in the quest for complete data integrity can only be realized by testing the entire stack of software and hardware in use by the archive application. We employ a homegrown utility called DIVT – Data Integrity Verification Tool.

DIVT runs as a client on various center platforms while using various source file systems. It transfers files into the archive. The files land on level 0 disk cache. They are then pulled out of the archive and compared against the original. The files are stored again, except this time the file is pushed down to level 1 tape and purged off of level 0 disk. Again it is retrieved from the archive and compared against the original.

Should any anomaly exist, email notification will be sent.

This push and pull against the disk and tape levels of the HSM is constant. Finding problems is a game of percentages. In the last two years, DIVT has found two major problems. The first was a file stat() bug with Lustre parallel filesystem reporting inconsistent file size, the result of which were corrupted tar archive images. The second problem was a tape drive that was silently truncating files, thereby corrupting them on tape. None of these would've been found had it not been for the utility. The opportunity for silent corruption is rampant.

II. AVAILABILITY

The focus is on "nines of availability". Simply stated, it means reducing the length of planned outages. Our goal is often said to be "two and a half nines," or 99.5% annual uptime, which translates into 3.65 hours of outage per month or 1.8 days per calendar year. For this reason, each second of outage is tracked.

- ***Pre-Production***

A "Pre-Production" environment is an absolute necessity to an archive. All new device firmware, device drivers, operating system fixes and version upgrades, and application versions are tested rigorously. It is here where the methods and order of complex integrations take shape. Tuning parameters are also sorted. A substantive subset of the exact hardware used in production should be represented in pre-production.

With such an environment comes the need for discipline. A pre-production system must be fed and cared for in the same way a production system would be, otherwise it quickly achieves a state of neglect, requiring significant resources to restore its usefulness.

We have traditionally run two production environments – unclassified and classified. Each of those has a dedicated pre-production environment. Deployments start in unclassified preproduction. Depending on the nature of the changes, testing can be from a couple weeks to a couple months, after which time it's deemed suitable for production and a planned downtime date is set.

Then the process is started all over for the classified side on its pre-production system. These cycles tend to be much shorter as most software has been battle-hardened in our unclassified environment by this time.

More typically, due to its larger scale, unclassified production will uproot a bug that wasn't caught in preproduction testing. All future deployments are put on hold while problems are researched and remedied.

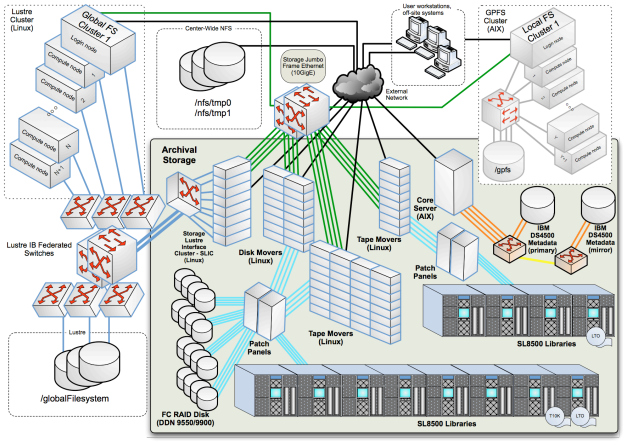
The net result is a well-sorted production rollout that minimizes chances of users finding issues before the deployment staff.

- ***Redundancy***

The current number one reason for loss of service is planned and unplanned electrical outages. Our archive spans four different raised floor

environments. Consequently, we often react to regional raised-floor power work for nearby projects and our own expansion. This is exacerbated by newer electrical safety rules that prohibit electricians from performing "hot work".

To mitigate, certain hardware has been duplicated in redundant configurations.



Metadata disk for the HPSS Core Server is replicated in two different rooms 1km apart. Either can go down and the application will continue operation. Using operating system disk mirroring on top of RAID controllers across highly available duplicated fiber channel switching technology ensures no one single point of failure between the main core server and its metadata.

Further, robotic software control servers (for Oracle ACSLS) have been made redundant in a cold spare configuration, also located 1km apart.

Identifying single points of failure allows us to concentrate on the biggest bang for our redundancy dollar. The disk and tape mover nodes exist in smaller commodity hardware configurations, in sufficient quantities, so as to allow for individual node failures. Failed nodes are fenced out by our scalable application, HPSS, all while the remaining movers handle the load.

Core server hosts, on the other hand, can be found to have redundant internal drives, fiber HBAs, fans, ethernet cards, power supplies, and ECC memory.

PDUs are specified for twin tailed power sources and are fed from two panels where available.

- ***Measured doses of code patching***

Keeping up the nines of availability requires resisting the urge to over-patch the production systems. Security concerns should be thought out and patches tested cohesively in pre-production environments. With few exceptions, the most egregious software security vulnerabilities can be handled by a workaround or an efix which keeps the main archive service available without interruption. Constant patching equals constant downtime.

III. EFFICIENCY

In many ways, data archives are a study in how to do more with less. Budgets and personnel tend to not grow in step with storage requirements.

- ***Trim the fat***

With enough inexpensive data mover hosts, expensive-to-purchase and even more expensive-to-maintain fiber switch technology is not required.

Our data movers are commodity hardware based x86_64 systems running Linux. All devices are direct attached to the HBA on the host in either FC4 or FC8 native speeds. Fiber trunks running to patch panels handle the interconnects. No electrical is required to these panels.

Should one of these systems crash, there are plenty of remaining nodes to shoulder the load. We mark their associated devices unavailable to the archive application, thus no need exists for a switching architecture to swing devices to online hosts.

- ***Piggyback procurements***

Given this commodity hardware data mover design, we are able to leverage the sorts of purchases HPC centers make all the time, namely large cluster and file system disk purchases.

With modest adjustments of node configurations, what was a compute node can be a quite capable and inexpensive I/O data mover machine if tied into the larger procurement process.

- ***Vendor manpower***

Our center has dedicated operations staff well versed in the various hardware types and associated common failure scenarios. Specific vendor gear exists onsite in considerable quantities. Accordingly, we find it possible to negotiate daily onsite vendor CSE/CE support at modest rates. This allows us to have a specialist available for the inevitable unique problems falling outside the scope of an operations staff, as well as for providing a fast track to backend developer support at a moment's notice. This speeds time to resolution and frees our staff to concentrate on the administration of the archive and center at large.

- ***Authoritative sources of information***

An essential component of archive management involves reliably answering questions whose result set changes from frequently to hardly ever. Sources for such questions range from automated scripts to reports written for management. Examples include:

- What milestones were achieved last year?
- What are the firmware versions on the tape drives?
- What fixes make up our previous production code release?

Establishing a single authoritative source abates confusion. The authoritative source often differs for each question, but needs to be identified and communicated to avoid future errors based on incorrect or drifting information gathered from substandard sources (e.g. a file in team member's home directory).

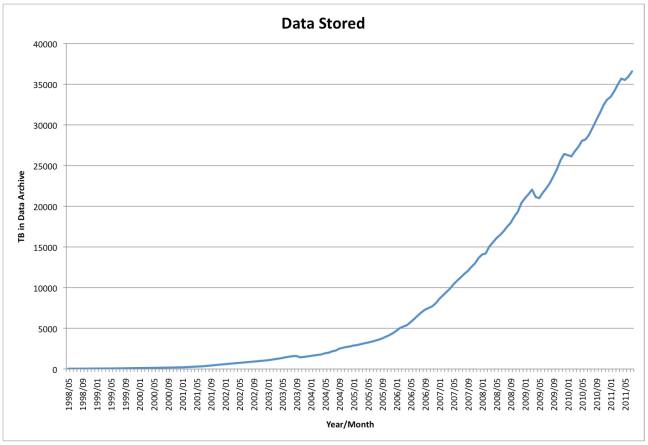
For example, the archive team coalesced on a TeamForge (SourceForge) web utility, which provides a wiki and source control among others. We track project progress here, create How-To's, load key diagram documentation, etc.

Using tape drives as an example, we write utilities that get information in real time by accessing drives over their built in Ethernet connections. Items such as dump status, firmware version, currently mounted cartridge, feet of tape processed, etc. can be gleaned in this fashion.

Our application code and the various local modifications are kept in subversion. We track preproduction and production series. The team members checkout the code, interact with it, and check it back into the central repository. All changes are logged.

MANAGING GROWTH

Fiscal year 2011 marks the first production year of our new Archival Quota system (a.k.a. Aquota). Traditionally, users have been allowed to grow our data archives with few restrictions.



Growth in the last few years suggested that we would need to construct vast new buildings to hold data if this growth curve was to be sustained.

- ***Unique to this quota system***

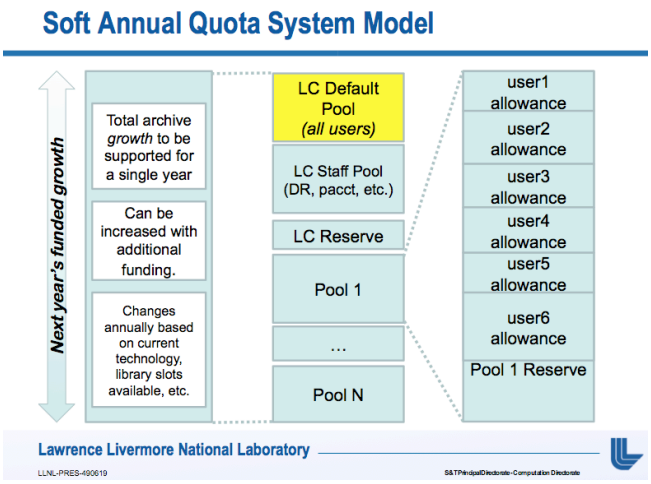
Two key differences exist comparing Aquota and a traditional disk quota. First is that only annual growth is measured. Data stored the fiscal year prior and before is not considered. Quotas are reset each new fiscal year.

Secondly, it is "soft" enforcement only. Users are still allowed to store after their limits are reached. Users as well as their responsible program managers are contacted when quota is met. It is reported that they have grown beyond their

default allowance and need to seek additional resources.

- ***Aquota Model***

Most users live within their yearly budget. The center allocates "pools" of storage to projects. Individuals exceeding their default allowance need to be given space from project pools.

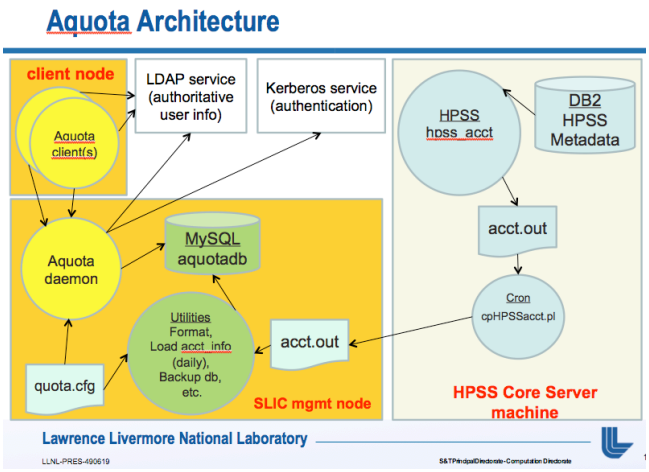


This model of growth control allows the center to predetermine the amount of growth it is willing to sustain for the upcoming fiscal year, rather than attempting to budget based on the previous year's unabated growth.

Once a budget is set, a reasonable set of growth constraints is arrived at based on the amount of media the budget will allow (including potential technology refreshes).

- ***Aquota Architecture***

Aquota was built in-house. It is comprised of a server daemon written in C, any number of multiple interactive clients written in C, and a variety of administrative tools written in Perl.



Nightly exports of HPSS accounting data are imported into a MySQL database. The Aquota daemon handles all client Aquota requests, which can run on a variety of hosts in the center. Users, Pool Managers, and Administrators have increasing levels of authority and interface with the system via the command line client.

- ***Impact***

Early evidence for FY11 suggests that overall annual growth will have dropped 14 points from the previous three-year average. The tangible impact is that a tool to facilitate a dialogue has been opened between users, responsible managers, and those of us tasked with offering the archive service. This did not exist in previous years. A common language is now being spoken.

CONCLUSIONS

Data archives outlive architectures, operating systems, and interconnects. They grow with wild abandon. Bytes churn in a maelstrom of activity as new data arrives and old data is repacked.

Even with a cadre of the latest technological advances and efficient models of deployment, the primary elements of a successful data archive are the people and their willingness to strive to meet the credos of the archive. Key skills in computer science - particularly in languages interpreted and compiled - don't hurt either.