HEPiX Storage Working Group
- progress report 1.2010 -

Andrei Maslennikov

April 19, 2010 – Lisbon

Summary

- Activities Fall 2009 – Spring 2010
- Storage Questionnaire 2010
- Intermediate test results obtained at KIT
- Plans for the next months
- Discussion

Activities Fall 2009 – Spring 2010

- As of the late fall 2009 the group was building the new test facility at KIT. In parallel, the pre-existing CMS test case was renewed, and a new ATLAS analysis emulation was added. The new laboratory became operational in the beginning of March 2010, and we have already some first numbers to share.

- As well, the group prepared a new edition of the HEPiX Storage Questionnaire. This time 14 sites took part in the survey, and the group thanks all the site representatives who contributed for their help and patience.

Credits for the late period

- The new test laboratory at KIT was built on the top of hardware kindly provided by Karlsruhe Institute of Technology (rack and network infrastructure, load farm) and E4 Computer Engineering (new disk server). CERN had contrubuted with some funds to cover a part of human hours.

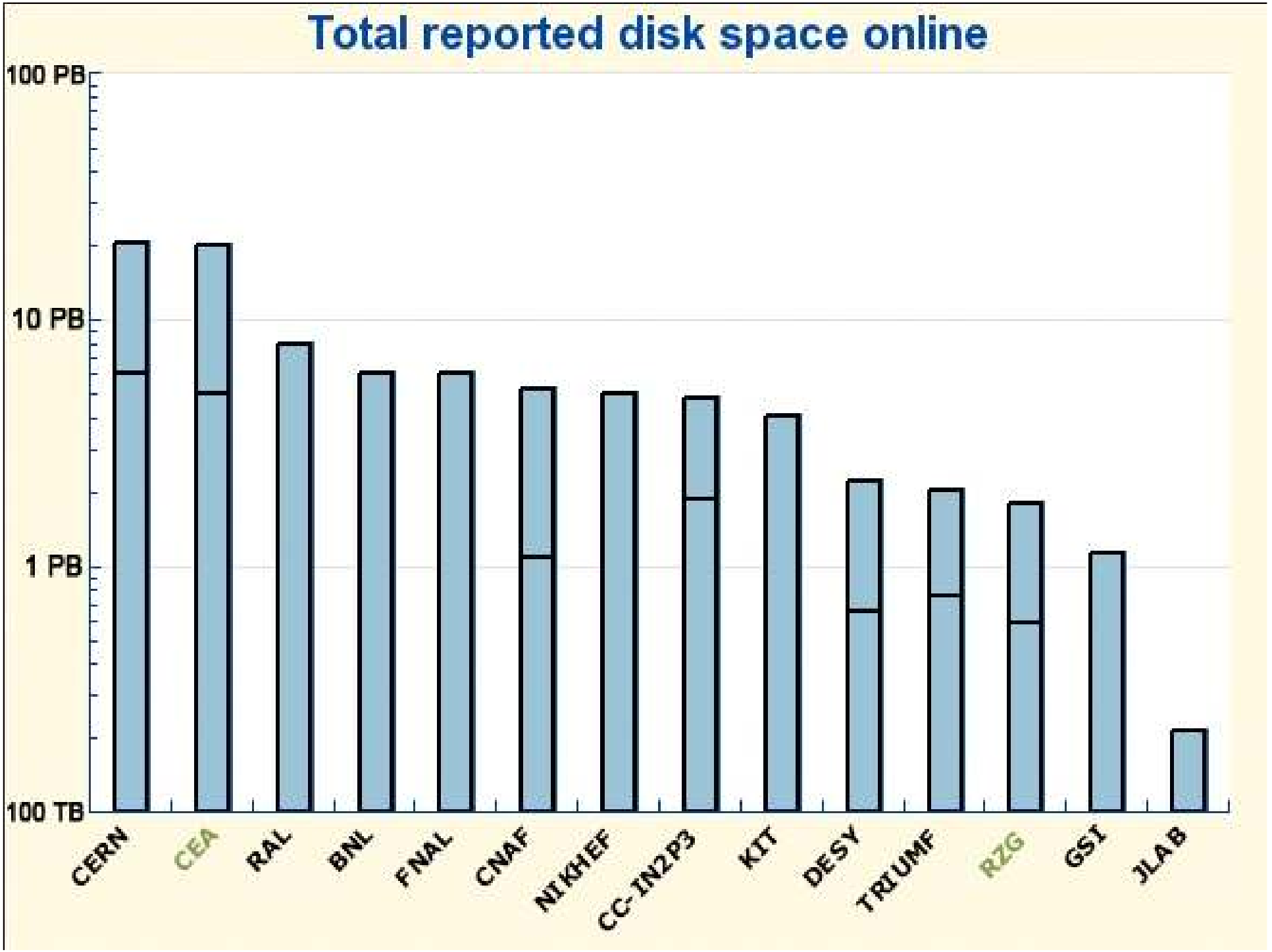
- These people participated in provisioning, funding, discussions, laboratory building, preparation of test cases and test framework, tests and elaboration of the results:

CASPUR	A.Maslennikov (Chair), M.Calori (Web Master)
CEA	J-C.Lafoucriere
CERN	B.Panzer-Steindel, D. van der Ster, R.Toebbicke
DESY	M.Gasthuber, P.van der Reest
E4	C.Gianfreda
INFN	G.Donvito, V.Sapunenko
KIT	J.van Wezel, A.Trunov, M.Alef, B.Hoeft
LAL	M.Jouvin
RZG	H.Reuter

Storage Questionnaire 2010

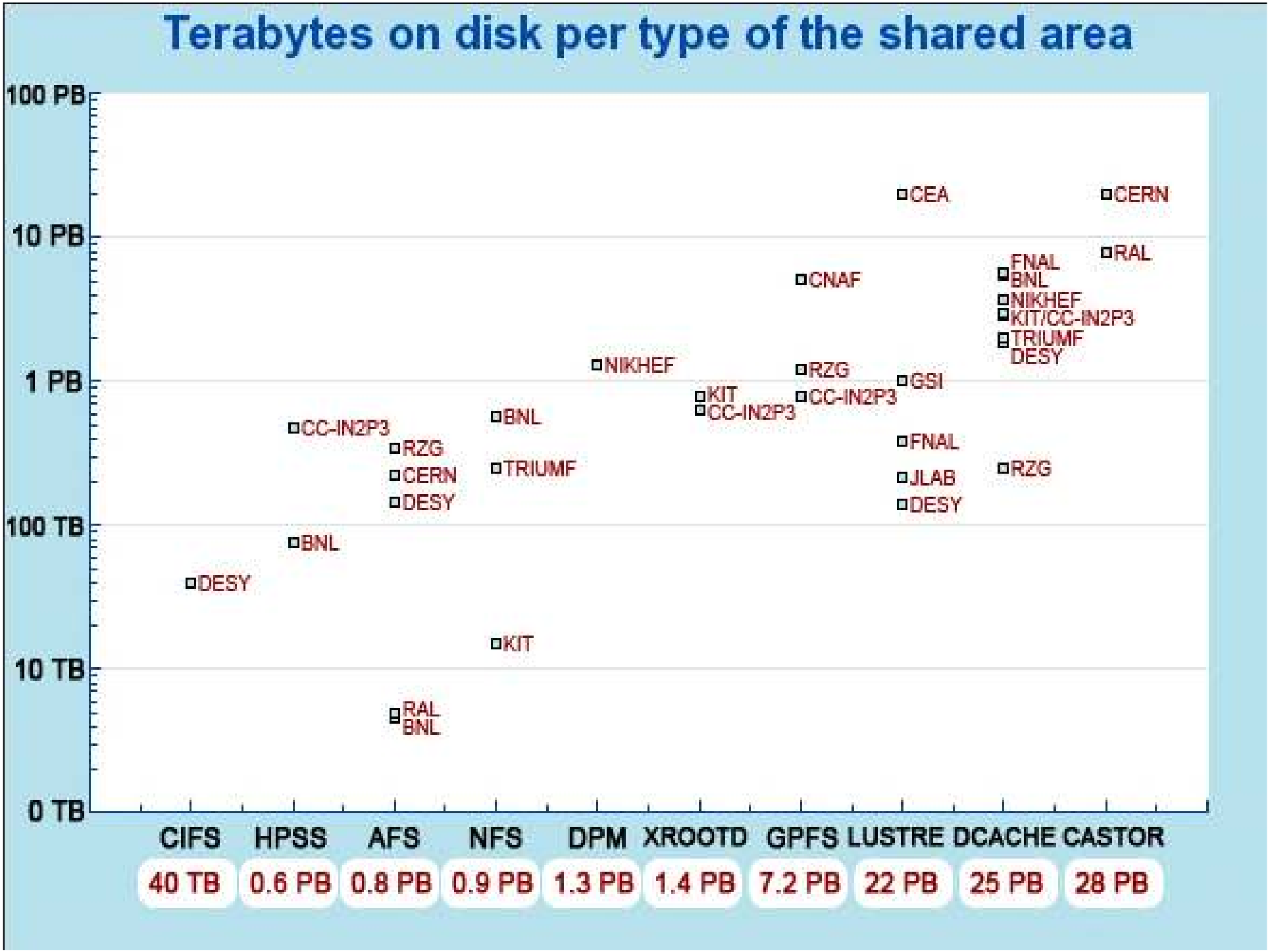
Storage Questionnaire 2010

- **The 14 participating sites were mainly of the HEP origin, CEA and RZG being the only exceptions. The total described space online summed up to 87 PB, to be compared with 14 PB reported in 2007.**


Total reported disk space online
100 PB
10 PB
1 PB
100 TB
CERN
CEA
RAL
BNL
FNAL
CNAF
NIKHEF
CC-IN2P3
KIT
DESY
TRIUMF
RZG
GSI
JLAB

Some facts

- Roughly one third of reported storage is in CASTOR, another third is in dCache and the remaining third is inside the shared file systems. CASTOR is only used at CERN and RAL, whereas dCache is in use at 8 sites out of 14.
- In 2007, no HEP data were stored inside Lustre. Today it is accounting for 50% of the shared file system space (another 50% is in GPFS). The shared file systems currently hold around 20% of HEP data, but they are visibly acquiring ground (new Lustre areas at GSI, DESY and FNAL etc). The recent migration from CASTOR to GPFS/STORM at CNAF demonstrated the feasibility of a large LCG compatible archive built on the top of a shared file system.
- Currently observed ratio N-of-clients/N-of-servers oscillates around 10, over all participating HEP sites. Servers are still mostly with 1G outlets, so this ratio will likely be growing towards 50-90 for 10G based servers.


Terabytes on disk per type of the shared area
100 PB
10 PB
1 PB
100 TB
10 TB
0 TB
CEA
CERN
RAL
FNAL
BNL
NIKHEF
KIT/CC-IN2P3
TRIUMF
DESY
CNAF
NIKHEF
RZG
GSI
KIT
CC-IN2P3
CC-IN2P3
BNL
FNAL
RZG
CERN
TRIUMF
JLAB
RZG
DESY
DESY
BNL
KIT
RAL
BNL
CIFS
HPSS
AFS
NFS
DPM
XROOTD
GPFS
LUSTRE
DCACHE
CASTOR
40 TB
0.6 PB
0.8 PB
0.9 PB
1.3 PB
1.4 PB
7.2 PB
22 PB
25 PB
28 PB

Observations

- So far, there seems to be no universally accepted data archival method for HEP data and situation continues to remain rather non-uniform. This non-uniformity in many cases has historical roots, and is often promoting a sane technological competition. However, one should never forget that all HEP sites have to deal with data of the same type and with similar access patterns.

- In this light, and in the view of the permanent growth of data volume, it is becoming more and more clear that a regular, methodical monitoring and comparison of TCO, reliability and efficiency of data archival and access solutions is and will be remaining a priority for HEP community.

Storage Laboratory 2010

Current goals

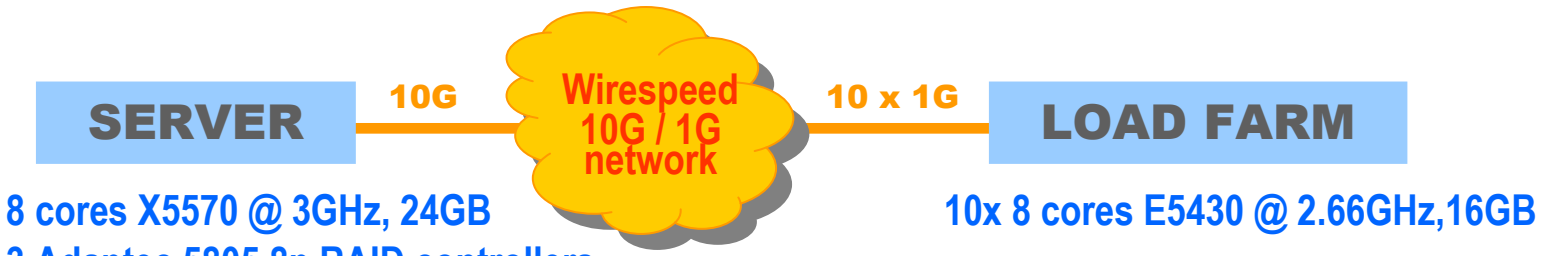
- As in the previous years, we aim at the performance comparison of most diffused storage solutions (AFS, GPFS, Lustre, dCache, Xrootd etc).
- Comparison is being done on the common hardware base, employing a set of realistic use cases relevant for the HEP community; one of our ancillary goals is thus to enlarge and keep up-to-date the use case library.

Disclaimer

- **We are constantly dealing with the “moving target”: data formats and use cases are evolving, hardware base is changing, new versions of storage access and archival software replace the old ones. This implies that results obtained in the storage laboratory are and will always remain a subject to change.**

- **Whatever we report should hence aways be seen as “work in progress”. We are not trying to provide any final recommendations but are rather sharing with you our findings and are ready to accept any advice and feedback.**

Hardware setup 2010 at KIT



8 cores X5570 @ 3GHz, 24GB
3 Adaptec 5805 8p RAID controllers
24 Hitachi drives of 1 TB
1 Intel 82598EB 10G NIC

10x 8 cores E5430 @ 2.66GHz,16GB

This setup reperesents well an elementary fraction of a typical large hardware installation and has basically no bottlenecks:

- **Each of the three Adaptec controllers may deliver 600+ MB/sec (R6)**
- **Ttcp memory-memory network test (1 server – 10 clients) shows full 10G speed**

(In 2009 we were limited by 4x 1G NICs and only one RAID controller)

Details of the current test environment

- RHEL 5.4/64bit on all nodes (kernel 2.6.18-164.11.1.lustre / -164.15.1)
- Lustre 1.8.2
- GPFS 3.2.1-17
- OpenAFS/OSD 1.4.11 (trunk 984)
- dCache 1.9.7

- Use Case 1: CMS "Data Merge" standalone job - fw v.3.4.0 (Giacinto Donvito)

- Use Case 2: ATLAS "Hammercloud" standalone job – fw v.15.6.1 (Daniel van der Ster)

Tunables

We report here, for reference, some of the settings that were used so far.

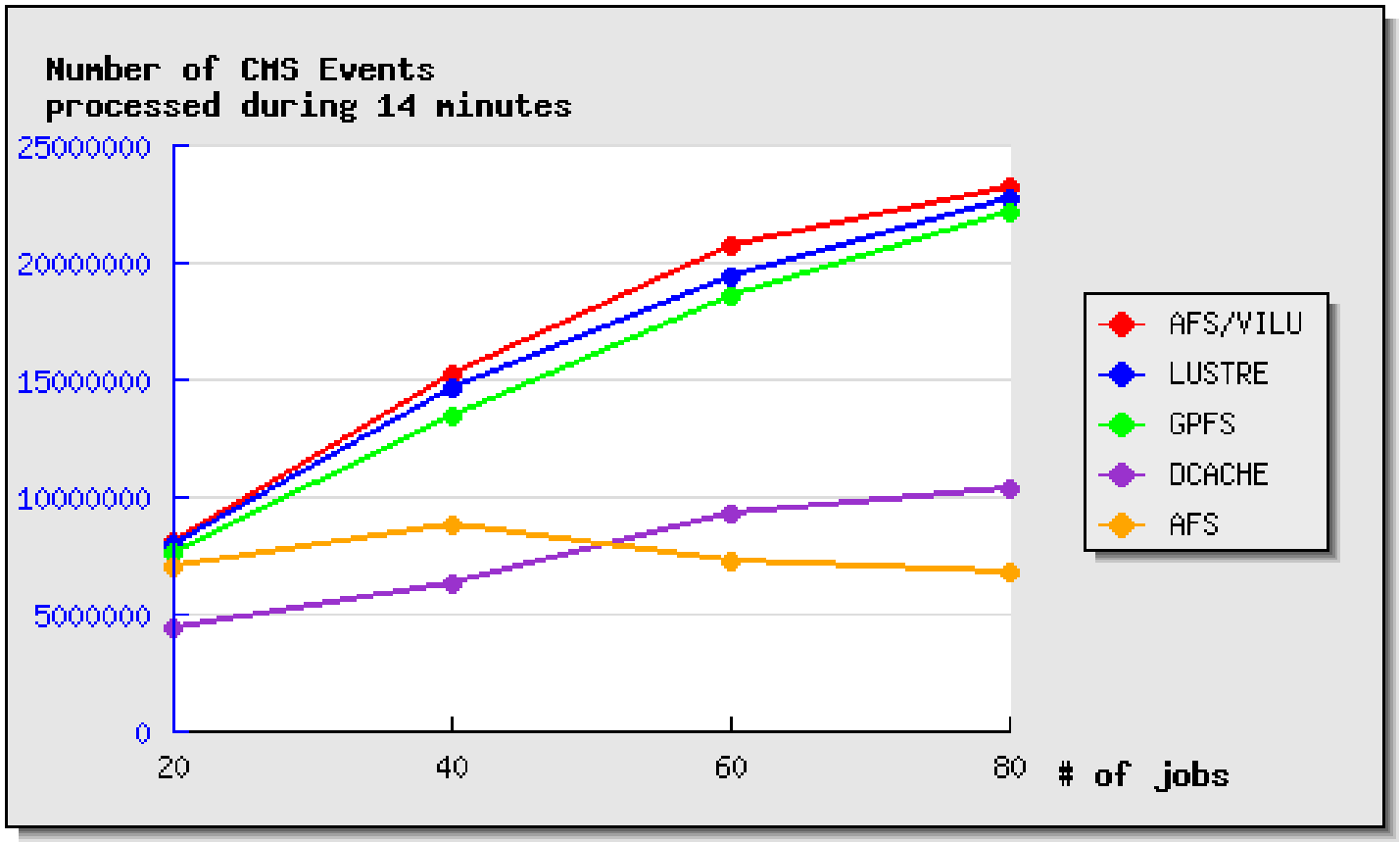
Diskware: three stanadlone RAID-6 arrays of 8 spindles, stripe size=1M

Lustre:
No checksumming, No caching on server
Formatted with: "-E stride=256 -E stripe-width=1536"
Data were spread over 3 file systems (1 MGS +3 MDT)
OST threads: "options ost oss_num_threads=512"
Read-aheads on clients: 4MB (CMS), 10MB (ATLAS)

GPFS:
3 NSDs, one per RAID-6 array
3 file systems (one per NSD)
-B 4M –j cluster
maxMBpS 1250
maxReceiverThreads 128
nsdMaxWorkerThreads 128
nsdThreadsPerDisk 8
pagepool 2G

AFS:
(dCache)
3 XFS vicep or dCache pool partitions (one per RAID array)
Formatted with: "-i size=1024 -n size=16384 -l version=2 -d sw=6,su=1024k"
Mounted with: "logbsize=256k,logbufs=8,swalloc,inode64,noatime"
Afsd options: "memcache, chunksize 22, cache size 500MB"
Dcache options: DCACHE_RAHEAD=true, DCACHE_RA_BUFFER=(100KB-100MB)

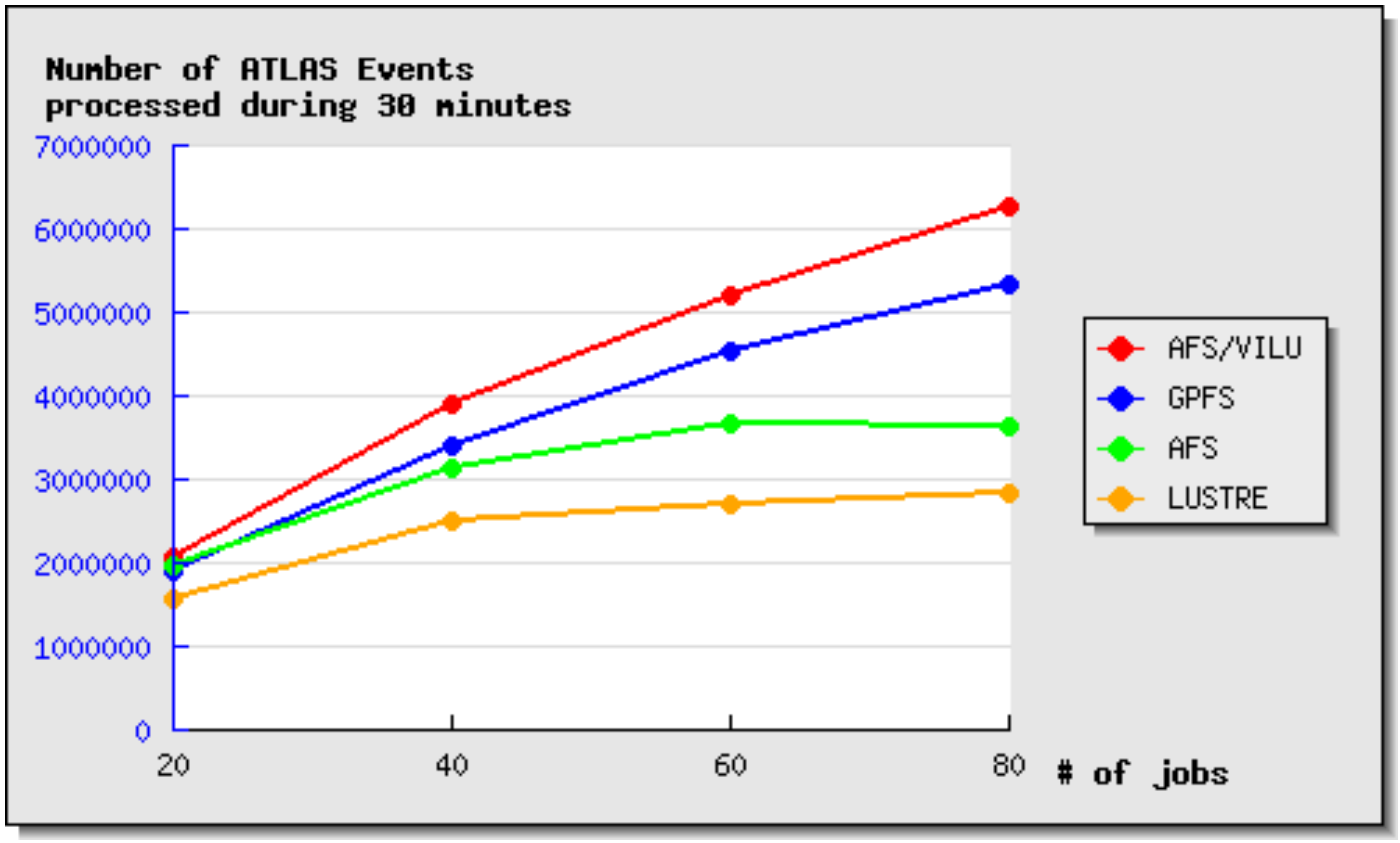
Current CMS use case results



For this test case, GPFS and Lustre are almost equally efficient. AFS/Vicep-over-Lustre looks surprisingly good.

The dCache result is very fresh and still has to be investigated. We however plot it here along with the others since the CMS test job was taken from the real life environment. The dCache team expressed an interest to verify the correctness of dCache and/or setup usage in this case, this will shortly be done in collaboration with them.

Current ATLAS use case results



The ATLAS job was prepared in the beginning of 2010; since then, ATLAS had migrated to a new data format and, consequently, to the new data access pattern. We were still using the previous version known for its high fraction of random access I/O. Thus it was of no surprise to discover that native Lustre was the most inefficient solution for this use case. However, AFS/Vicep with Lustre transport had shown the best results, like in the case of CMS. We were yet unable to run the dCache-based ATLAS test, this will be done soon.

More detail

For completeness, we quote here the numbers of events observed, along with the average number of MB per second entering all the client network interfaces during the test job execution.

(CMS)

	20 threads	40 threads	60 threads	80 threads
AFS NAToX a1M.22.4G	192 MB/sec 7015157 evs	279 MB/sec 8758298 evs	277 MB/sec 7243828 evs	277 MB/sec 6784062 evs
dCache.va	181 MB/sec 4447964 evs	300 MB/sec 6295403 evs	416 MB/sec 9237278 evs	510 MB/sec 10325946 evs
GPFS.R6.4 a512ra16	203 MB/sec 7568779 evs	389 MB/sec 13499568 evs	554 MB/sec 18579369 evs	702 MB/sec 22160217 evs
LU.4Mnsca a1M.6k	170 MB/sec 7961373 evs	331 MB/sec 14657741 evs	442 MB/sec 19358264 evs	551 MB/sec 22781409 evs
AFS/VILU a1M.22z.5	213 MB/sec 8152271 evs	412 MB/sec 15190647 evs	585 MB/sec 20686440 evs	720 MB/sec 23158250 evs

(ATLAS)

LU.10Mnsc ra512	114 MB/sec 1559890 evs	192 MB/sec 2488793 evs	199 MB/sec 2713095 evs	224 MB/sec 2840535 evs
AFS NAtoX a1M.22.4G	140 MB/sec 1960132 evs	232 MB/sec 3144659 evs	275 MB/sec 3659608 evs	279 MB/sec 3628869 evs
GPFS.R6.4 ra.512	183 MB/sec 1899171 evs	378 MB/sec 3409685 evs	541 MB/sec 4517282 evs	675 MB/sec 5346205 evs
AFS/VILU a1M.22z.5	148 MB/sec 2078669 evs	285 MB/sec 3887371 evs	392 MB/sec 5213502 evs	484 MB/sec 6280046 evs

Observations - GPFS

- This time we were able to obtain excellent GPFS results, much better than those that we have seen before. Most probably, this improvement may be explained by the elimination of the network bottleneck that we had in our previous setup (we stepped to 1000 MB/sec from 450 MB/sec). As well, we are now running a more recent version of GPFS software which is known to be more performing.

- GPFS is hence looking quite attractive. IBM had recently changed its licensing policies, and the product became more affordable. As of the next quarter, they promise to propose the even more convenient site licenses.

- GPFS technology allows for smooth addition and removal of storage devices which makes it much more manageable in comparison with Lustre. Its principal drawback today is the lack of the fragmented file system layout. Striping may not be switched off, thus a loss of just one NSD may result in a visible data outage across the file system.

Observations – AFS/Vicep-Lustre

- Somehow an amalgam of AFS and Lustre transport presented itself as the most performant solution for the two extreme cases (CMS use case with its modest random I/O component vs ATLAS/OldFormat with high random I/O),
- Running AFS with a speed of Lustre is especially attractive because of the value added features of AFS. It provides the fine-grained security level, and adds the possibility to add/remove Lustre OSTs without interrupting the file system activity. It is available at no cost; even if it is true that Lustre management on a large scale may require visible human resources, this hybrid solution is definitively deserving more attention.
- NB: The AFS/VILU tests were run as superuser. Some small overhead may be necessary to support the non-privileged user access.

Immediate plans

- The group is planning to run the lab tests at KIT for the next 4 months, and then to present its next progress report at Cornell.
- The test program includes migration to the updated ATLAS use case, thoroughful investigation of dCache in collaboration with developers, Xrootd measurements. We also plan to study the aggregate performance downgrade due to rebuild in progress, evaluate some new disk hardware. We might find time to look at other solutions like NFS v.4.1 that is now being integrated with dCache and Hadoop which is currently gaining momentum at the Tier-3 sites.

Discussion