HEPiX FSWG – Final Report

Andrei Maslennikov

May 2008 - Geneva

Summary

- Reminder: raison d'être
- Active members
- Workflow phases (February 2007 - April 2008)
- Phase 3: comparative analysis of popular data access solutions
- Conclusions
- Discussion

Reminder: raison d'être

- Commissioned by IHEPCCC in the end of 2006
- Officially supported by the HEP IT managers
- The goal was to review the available file system solutions and storage access methods, and to divulge the know-how and practical recommendations among HEP organizations and beyond

- Timescale : Feb 2007 – April 2008
- Milestones: 2 progress reports (Spring 2007, Fall 2007),
  1 final report (Spring 2008)

Active members

- Currently we have 25 people on the list, but only these 20 participated in conference calls and/or actually did something during the last 10 months:

CASPUR	A.Maslennikov (Chair), M.Calori (Web Master)
CEA	J-C.Lafoucriere
CERN	B.Panzer-Steindel
DESY	M.Gasthuber, Y.Kemp, P.van der Reest,
FZK	J.van Wezel, C.Jung
IN2P3	L.Tortay
INFN	G.Donvito, V. Sapunenko
LAL	M.Jouvin
NERSC/LBL	C.Whitney
RAL	N.White
RZG	H.Reuter
SLAC	A.Hanushevsky, A.May, R.Melen
U.Edinburgh	G.A.Cowan

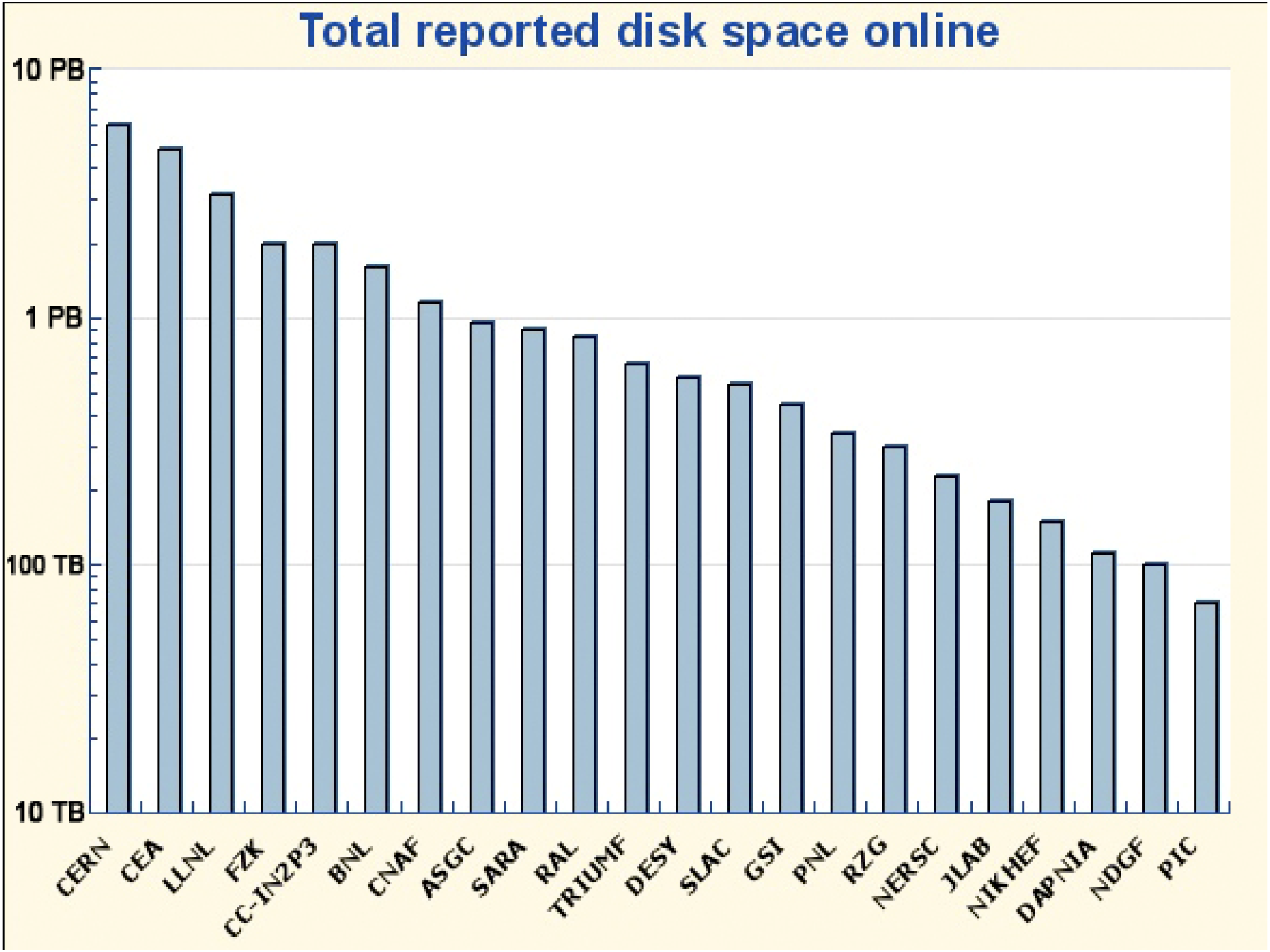
- During the lifespan of the Working Group: held 28 phone conferences, presented two progress reports at HEPiX meetings, reported to IHEPCCC.

Workflow phase 1: Feb 2007 - May 2007

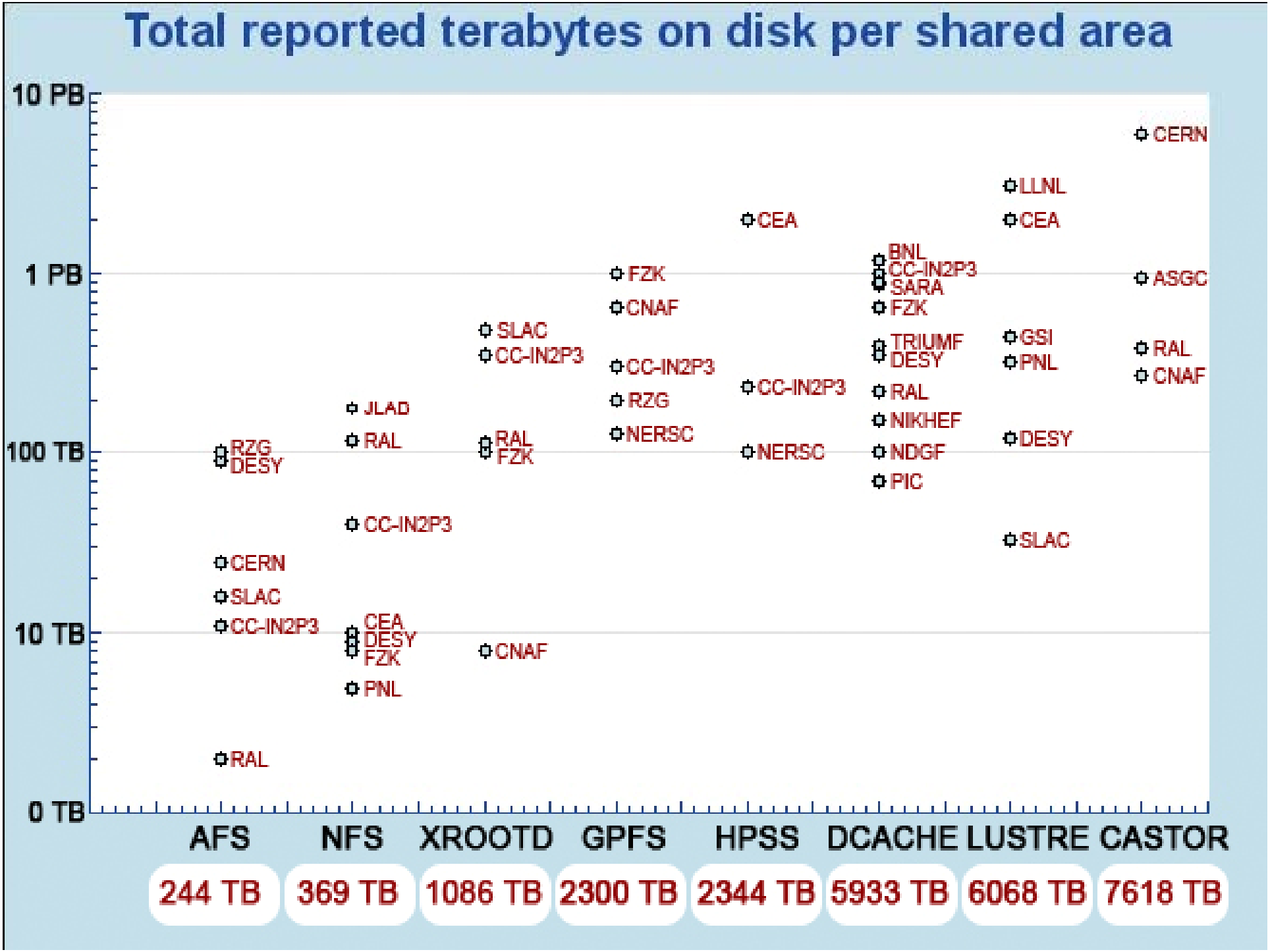
- Prepared an online Storage Questionnaire to gather the information on storage access solutions in use. Collected enough information to get an idea of the general picture. By now, all important HEP sites with an exception of FNAL have described their data areas.

- Made an assessment of available data access solutions. Decided to concentrate on large scalable data areas.

- Selected a reduced set of architectures to look at:
  - File Systems with Posix Transparent File Access (AFS, GPS, Lustre);
  - Special Solutions (dCache, DPM and Xrootd)


Total reported disk space online
10 PB
1 PB
100 TB
10 TB
CERN
CEA
LLNL
FZK
CC-IN2P3
BNL
CNAF
ASGC
SARA
RAL
TRIUMF
DESY
SLAC
GSI
PNL
RZG
NERSC
JLAB
NIKHEF
DAPNIA
NDGF
PIC

Total reported terabytes on disk per shared area



Workflow phase 2 : Jun 2007 - Oct 2007

- Collected technological information on storage access solutions, had numerous exchanges with site and software architects, learned about trends and problems. Started a storage technology web site.

- Main conclusions during phase 2:

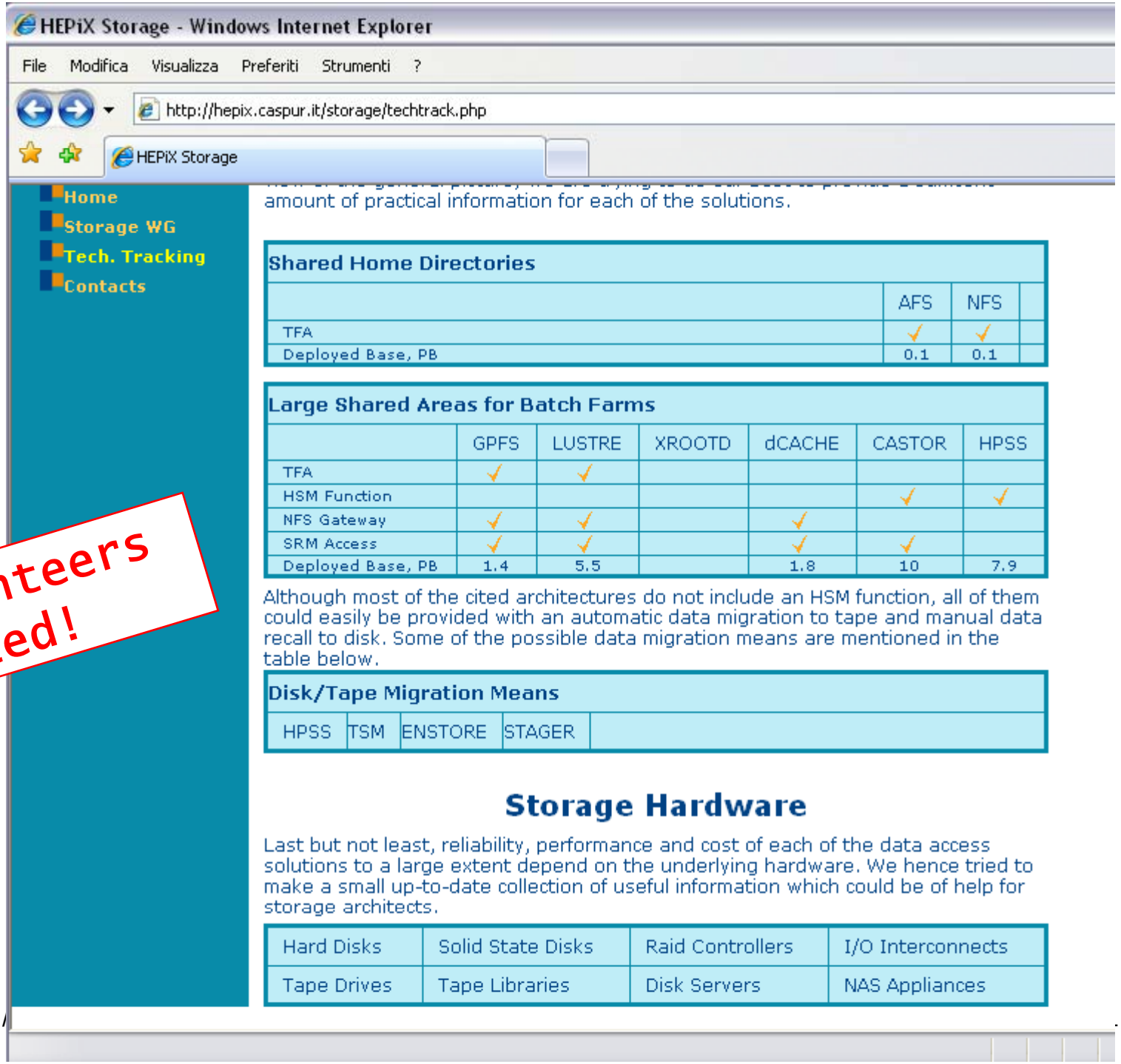
  - Storage solutions with TFA access are becoming more and more popular, most of the sites foresee growth in this area; HSM backends are needed, and are being actively used (GPFS) / developed (Lustre).

  - As SRM backend for TFA solutions (SToRM) is now becoming available, these may be considered as a viable technology for HEP and may compete with other SRM-enabled architectures (Xrootd, dCache, DPM).

- A few known comparison studies (GPFS vs CASTOR, dCache, Xrootd) reveal interesting facts, but are incomplete. The group hence decided to perform a series of comparative tests on a common hardware base for AFS, GPFS, Lustre, dCache, DPM and Xrootd.

HEPiX Storage Technology Web Site

- Consultable at http://hepix.caspur.it/storage
- Meant as a storage reference site for HEP
- Not meant to become yet another storage Wikipedia
- Requires time, is being filled on the best effort basis
- Volunteers wanted!

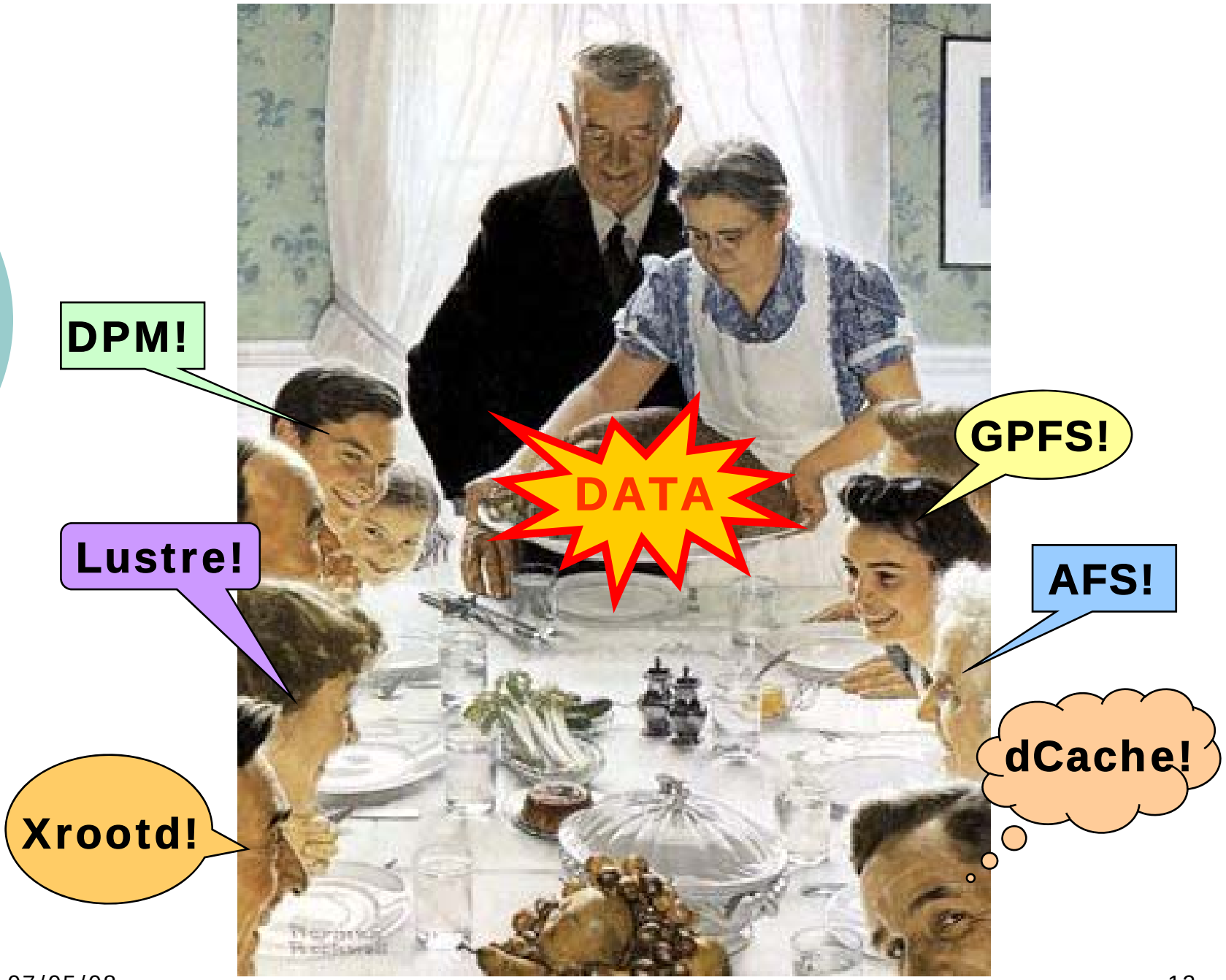


Volunteers wanted!

Workflow phase 3: Dec 2007 – Apr 2008

Tests, tests, tests!

- Looked for the most appropriate site to perform the tests; Offers to host them came from DESY, CERN, FZK and SLAC.
- Selected CERN as a test site, since they were receiving new hardware which could be made available for the group during the pre-production period.
- Agreed upon the hardware base of the tests: it had to be similar to that of an average T2 site: 10 typical disk servers, up to 500 jobs running simultaneously, commodity non-blocking Gigabit Ethernet network


DPM!
DATA
GPFS!
Lustre!
AFS!
dCache!
Xrootd!

Testers

- Lustre: J.-C. Lafoucriere
- AFS: A.Maslennikov
- GPFS: V.Sapunenko, C.Whitney
- dCache: M. Gasthuber, C.Jung, Y.Kemp
- Xrootd: A.Hanushevsky, A.May
- DPM: G.A.Cowan, M.Jouvin

- Local support at CERN:
  B.Panzer-Steindel, A.Peters, A.Hirstius, G.Cancio Melia
- Test codes for sequential and random I/O: G.A.Cowan
- Test framework, coordination: A.Maslennikov

Hardware used during the tests

- Disk servers (CERN Scientific Linux 4.6, x86_64):

  2 x Quad Core Intel E5335 @ 2 GHZ, 16 GB RAM
  Disk: 4-5 TB (200+ MB/sec), Network: one GigE NIC

- Client machines (CERN Scientific Linux 4.6, x86_64):

  2 x Quad Core Intel E5345 @ 2.33 GHZ, 16 GB RAM, 1 GigE NIC

  TCP parameters were tuned as follows:

```
net.ipv4.tcp_timestamps = 0
net.ipv4.tcp_sack = 0
net.ipv4.tcp_mem = 10000000 10000000 10000000
net.ipv4.tcp_rmem = 10000000 10000000 10000000
net.ipv4.tcp_wmem = 10000000 10000000 10000000
net.core.rmem_max = 1048576
net.core.wmem_max = 1048576
net.core.rmem_default = 1048576
net.core.wmem_default = 1048576
net.core.netdev_max_backlog = 300000
```

Configuration of the test data areas

- Agreed on a configuration where each of the shared storage areas looked as one whole, but was in fact fragmented: no striping was allowed between the disk servers, and one subdirectory would be residing fully on just one of the file server.

- Such a setup could be achieved for all the solutions under tests, although with some limitations. In particular, GPFS architecture is striping-oriented, and admits only a very limited number of “storage pools” composed of one or more storage elements.
  In case of dCache, some if its features like secondary cache copies, were deliberately disabled to ensure that it looked like the others.

Setup details: AFS, GPFS

- AFS: OpenAFS version 1.4.6; vicepX partitions on XFS; client cache was configured on a ramdisk of 1 GB; chunk size – 18 (256 KB). One service node was used for a database server.

- GPFS: version 3.2.0-3. In the end of the test sessions IBM warned its customers that some debug code which was present in this release, and that it could be not the most optimal version for benchmarks. The version mentioned, 3.2.0-3, was however the latest version available on the day when the tests began.

  We used 10 separate GPFS file systems, all mounted under /gpfs. (With only 8 storage pools allowed, we could not configure one storage pool for each of the 10 servers). Thus each server contained one GPFS file system, both data and metadata. No service machines were used.

Setup details: Lustre

- Lustre: version 1.6.4.3. Servers were running the official Sun kernel and modules, clients were running unmodified RHEL4 2.6.9-67.0.4 kernel.

  There was one stand-alone Metadata Server configured on a CERN standard batch node (2xQuadcore Intel, 16GB). The 10 disk servers were all running plain OSTs, one OST per server.

Setup details: dCache, Xrootd, DPM

- dCache: version 1.8.12p6. On the top of the 10 server machines, 2 service nodes required for this solution were used. Clients mounted PNFS to access the dCache namespace.
- DPM: version 1.6.10-4 (64 bit, RFIO mode 0). One service node was used to keep the data catalogs. No changes on the clients were necessary. GSI security features were disabled.
- Xrootd: Version 20080403. One data catalog node was employed. No changes were applied on the clients.

Three types of tests

1. "Acceptance Test":

   50 thousand files of 300 MB each were written on 10 servers (5000 files per server). This was done running 60 tasks on 60 different machines that were sending data simultaneously to 10 servers. In this way, each of the servers was "accepting" data from 6 tasks.

   The file size of 300 MB used in the test was chosen as it was considered to be typical for files containing the AOD data.

   Results of this test are expressed in average numbers of megabytes per second entering one of the disk servers.

Results for the Acceptance Test

	Lustre	dCache	DPM	Xrootd	AFS	GPFS
Average MB/sec entering a disk sever	117	117	117	114	109	96

Most of the solutions under test demonstrated to be capable to operate at speeds close to that of a single Gigabit Ethernet adapter.

Three types of tests, contd

Preparing for the further read tests, we have create another 450000 small or zero length files to emulate a “fat” file catalog. This was done for each of the solutions under test.

2. “Sequential Read Test”:

10,20,40,100,200,480 simultaneous tasks were reading a series of 300-MB files sequentially, with a block size of 1 MB. It was ensured that no file was read more than once.

Results of these tests are expressed in total number of files read during a period of 45 minutes.

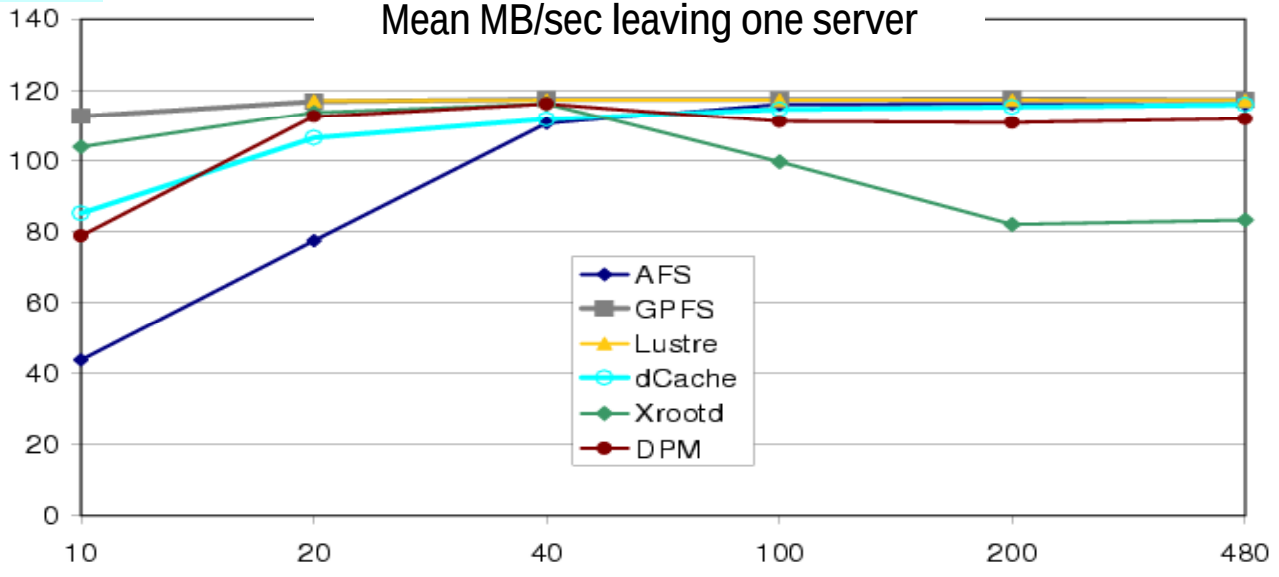
Results for the Sequential Read Test

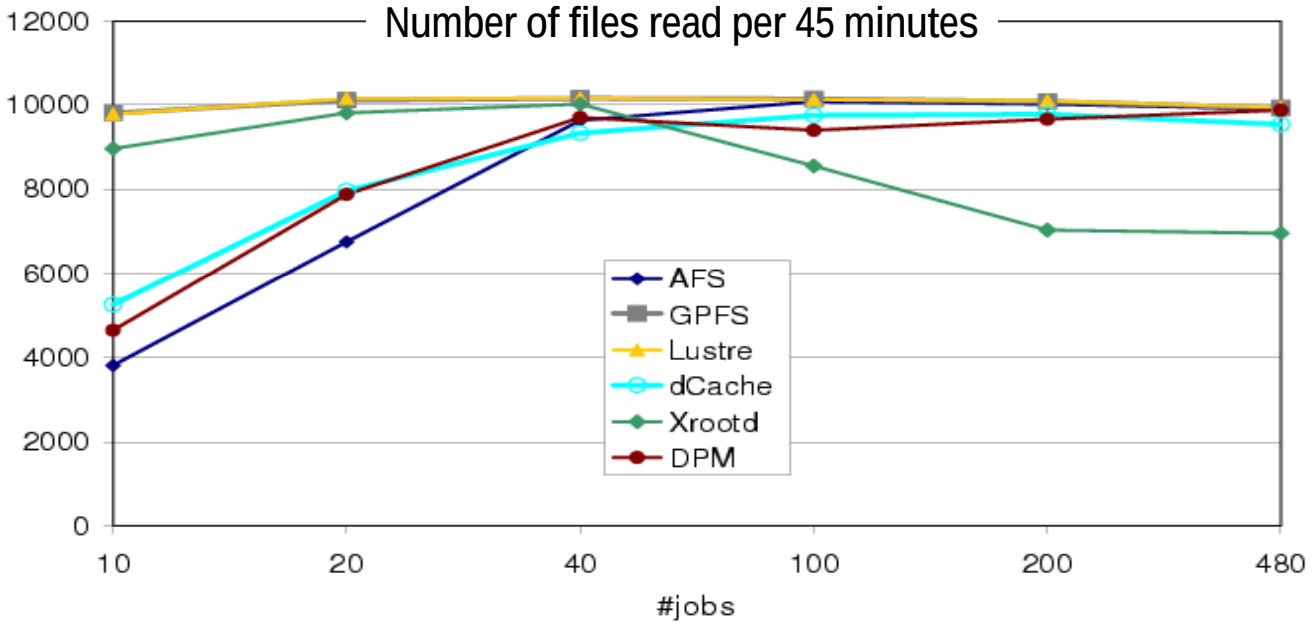
(numbers of the 300-MB files fully read over a 45 minute period)

	Number of jobs					
	10	20	40	100	200	480
AFS	3812	6751	9622	10069	10008	9894
GPFS	9794	10102	10144	10130	10073	9921
Lustre	9774	10138	10151	10117	10089	9935
dCache	5254	7959	9323	9744	9770	9531
Xrootd	8955	9801	10009	8545	7028	6953
DPM	4644	7872	9693	9390	9652	9866

Same results may also be expressed in MB/sec. With a good precision, 10000 files read correspond to 117 MB/sec per server, 5000 files correspond to 55 MB/sec per server. We estimate the global error for these results to be in the range of 5-7%.

Sequential Reads





Three types of tests, contd

3. "Pseudo-Random Read Test":

100,200,480 simultaneous tasks were reading a series of 300-MB files. Each of the tasks was programmed to read randomly selected small data chunks from within the file; the size of a chunk to read was set to be 10,25,50 or 100 KB and remained constant while 300 megabytes were read. Then the next file was read out, with a different chunk size. Each of the files was read only once.

The chunk sizes were selected in a pseudo-random way:
10 KB (10%), 25 KB (20%), 50 KB (50%), 100 KB (20%).

This test was meant to emulate, to a certain extent, some of the data organization and access patterns used in HEP. The results are expressed in the numbers of files processed in an interval of 45 minutes, and also in the average numbers of megabytes leaving the servers each second.

Results for the Pseudo-Random Read Test

	Number of jobs		
	100	200	480
AFS	6766	3802	1815
GPFS	13728	9575	6502
Lustre	12109	12062	11908
dCache	3185	4356	5530
Xrootd	3036	4194	5223
DPM	3216	4513	5988

Numbers of 300-MB files processed

	Number of jobs		
	100	200	480
AFS	79	112	87
GPFS	114	75	69
Lustre	117	117	117
dCache	35	49	65
Xrootd	34	47	60
DPM	35	48	64

Average MB leaving a server per second

Once this test was finished, the group was surprised with an outstanding Lustre performance, and tried to find an explanation for this (see the next slide).

Discussion on the pseudo-random read test

- The random read test allowed for reuse of some of the data chunks inside files (a condition which does not necessarily happen in real analysis scenarios). This most probably have favored Lustre before others as its aggressive read-ahead feature was effectively allowing the test code to “finish” faster with the current file and proceed with the next one.

- The numbers obtained are still quite meaningful. They clearly suggest that any sufficiently reliable judgment on storage solutions may only be made using a real-life analysis code against the real data files. We did not have enough time and resources to further pursue this. The group is however interested to perform such measurements beyond the lifetime of the Working Group.

Conclusions

- The HEPiX File Systems Working Group was set up to investigate the storage access solutions and to provide practical recommendations to HEP sites.
- The group made an assessment of existing storage architectures, documented and collected information on them, and performed a simple comparative analysis for 6 of the most diffused solutions. It leaves behind a start-up web site dedicated to the storage technologies.
- The studies done by the group confirm that shared, scalable file systems with Posix file access semantics may easily compete in performance with the special storage access solutions currently in use at HEP sites, at least in some of the use cases.
- Our short list of recommended TFA file systems contains GPFS and Lustre. The latter appears to be more flexible, may be slightly more performing, and is free. The group hence recommends to consider deployment of the Lustre file system in venue of a shared data store for large compute clusters.
- Initial comparative studies performed on a common hardware base had revealed the need to further investigate the role of storage architecture as a part of a complex compute cluster, against the real LHC analysis codes.

What's next?

- The group will complete its current mandate publishing the detailed test results on the storage technology web site.
- The group wishes to do one more realistic comparative test with the real-life code and data. Such a test would require 2-3 months of effective work, provided that sufficient hardware resources are made available all the time.
- The group intends to continue regular exchanges on the storage technologies, and to follow the technology web site.

Discussion