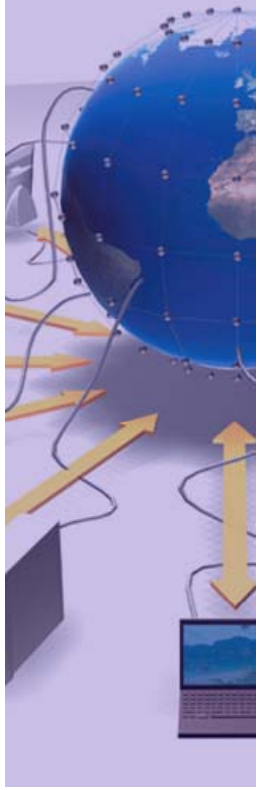


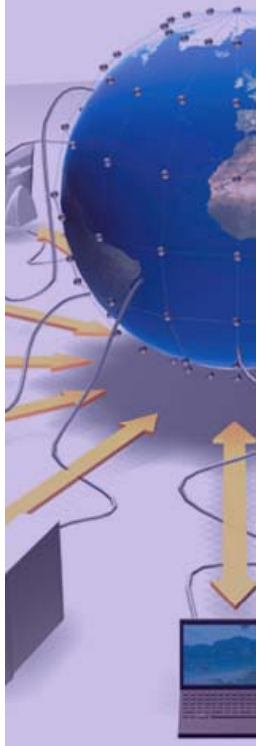
Summary of experiments' data management requests for 2009

Flavia Donno
Elisa Lanciotti
Andrea Sciabà

WLCG Collaboration Workshop
21-22 March, 2009
Prague, Czech Republic

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

ALICE data management



- All ALICE Grid sites must provide an xrootd enabled storage
- Mass storage
  - Accessed only by organized workflows
    - RAW data storage, replication and reconstruction
    - Recall and replication of ESDs to T1/T2s
  - Massive tape recalls
    - Tested at CERN via custom tools (parallel staging requests, minimizing multiple tape mounts)
    - Still an open question at the T1s
- Disk storage
  - Only type viable for analysis
  - Should allow for simultaneous access by a large number of clients, reading thousands of files

ALICE: disk access via xrootd (I)

- dCache
  - xrootd protocol implemented in Java, subset of full functionality
  - Experiences with large numbers of concurrent clients reading the same data (analysis T1/T2s, reconstruction T1)
    - Accurate dCache tuning required, performance depends on the level of expertise at the centre
    - Internal dCache server protection causes clients to wait, resulting in under-utilization of CPU

ALICE: disk access via xrootd (II)

- DPM
  - xrootd plugin works reasonably well, internal catalogue is not an issue
  - xrootd server version is obsolete (Aug 2007), missing advanced functionality
- CASTOR
  - Already very good experience with current xrootd implementation
    - Tested@T0 with prompt RAW reconstruction, analysis, access from CAF
  - New CASTOR 2.1.8 (further improvement in xrootd-CASTOR interoperability) is entering production

ALICE: disk access via xrootd (III)

- xrootd
  - Simple installation (choice of local compilation or RPMs)
  - No database needed – no risk of de-synchronization or loss, leading to loss of storage
  - Proven performance, reliability and scalability
  - ALICE uses xrootd native servers for some of the most critical data management tasks: conditions data on the Grid, configuration macros for production and analysis
  - Adopted by 30% of Tier-2s and some Tier-1s

ALICE requirements summary

- Deploy CASTOR 2.1.8 in production at T0/T1s
- Updates needed to dCache xrootd implementation and DPM server version
- xrootd native is the most straightforward management system for disk-only storage, especially relevant for T2s

ATLAS plans for 2009

- By May, run a full scale test (T0 reconstruction + distribution to T1 and T2), plus artificial traffic of small files to simulate analysis
- Reprocessing tests just done, other prestaging tests to come. To check
  - Achievable throughput
  - SRM interactions
- Analysis tests using Hammercloud
  - One goal is to verify that storage systems can cope with distributed analysis by testing all the components of the system
  - Data access: both direct via native protocol and via local copy to WN are tested

Issues seen by ATLAS

- Some files cannot be recalled from tape, manual interventions needed
- Checksum mismatches
  - Require cleaning and re-transfers
- Storage systems down
  - Cause inefficiencies
  - Sometimes caused by interference from other VOs
- Prestaging
  - Very low performances at some sites
- Scalability
  - PNFS bottleneck alleviated by upgrading to FastPNFS
  - Chimera?

ATLAS requirements for 2009

- Xrootd will be seriously investigated at CERN in the context of analysis. If performances are found to be very good, ATLAS could officially adopt it
- The focus should be on strengthening the services. For example, if Chimera is effective in improving the dCache performance, it should be adopted with a high priority
- Questions and requests
  - Is it a good practice to use *srmLs*, e.g. to get file size, checksum, etc.? Can any user do that or only a central service? In any case, *srmLs* should not block an SRM server
  - Less interference from other VOs
  - A way to check file checksum in FTS (in the works already)
  - A way to recalculate a checksum for a file already in the storage system
  - Protect storage systems by allowing prestaging requests only from special DNs or groups/roles

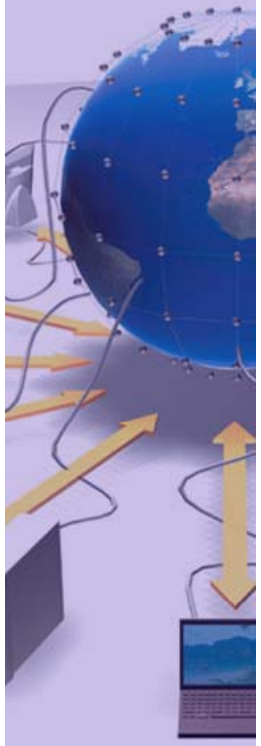
CMS plans

- Prestaging
  - Tested during CCRC08, manual prestaging by site manager
    - Decided that usage of tape families is essential
  - No definite plans yet for further tests but it would be desirable to do something at the same time as other VOs
- Analysis
  - Still need to measure the capability of Tier-2 sites to sustain a full scale analysis activity

Issues seen by CMS

- Internal inconsistencies can arise among DBS, PhEDEx database and storage
  - Consistency campaign produced tools to detect them
- Sites lose files from time to time
  - Disk servers die (sometimes before migration to tape), human errors, etc.
  - Tolerable now, much less with collision data!
- Problematic files, even if few %, cause waste of time
  - Work is often at the file level granularity
  - 30-40% of Savannah tickets for site problems are due to these
- Storage system instabilities

CMS wish list



- A much better SRM scaling (by a factor of 5-10)
  - Would pull data management farther from the sites
  - Easier prestaging
  - *srmLs* should not cause a “denial of service”
- Less failures on transfers and lost files
- A better authorization scheme on the storage
  - e.g. to forbid a random user from issuing a massive prestaging request
  - Tier-2’s: user X should not be able to delete Y’s files
- Quotas on users and groups
  - Tier-2’s: user X should not use up all space

LHCb data management activities for 2009

CERN IT Department

- FEST09: Periodic one week long full dress rehearsals of the whole online-offline processing chain
- Controlled prestaging tests at Tier-1 sites to measure throughput, possibly at the same time as other experiments
  - Prestaging works via SRM (*srmBringOnline*), polling via *srmLs*
  - Files should stay pinned for at least 48 hours
    - Raw data on T1D0 at Tier-1 sites

Issues seen by LHCb

- Inconsistencies among file catalogues, bookkeeping database and storage
  - Tools to correct them, but time consuming
- Interference from other experiments causing excessive load on storage system (e.g. at FZK)
- Hit by a temporary incompatibility between lcg-util and CASTOR
  - GridFTP sessions invalid after TURL is used
  - Requested to use GridFTP in “external” mode
- At IN2P3, files had locality NEARLINE even after a successful srmBringOnline (GGUS #45699)
  - Configuration issue specific to IN2P3
- lcg-cp with > 1 streams could time out while GridFTP processes didn’t exit and eventually killed the SE
  - Due to bad firewall configuration
- Site reliability improves when sites look proactively to LHCb SAM tests

LHCb wish list

- A tool to know the amount of free space in a disk cache
  - To put more intelligence in prestaging agents
- Access control lists in storage
  - To have a coherent system for file ownership
- Quotas

News from storage systems

- dCache
  - Splitting off a dedicated dCache instance for ATLAS was successfully done at FZK
    - Less interference among VOs
  - Transition to Chimera for Tier-1 sites waiting to see how it goes at NDGF
    - Completed at NDGF!
  - Plan to introduce with 1.9.3 an asynchronous *srmLs* to avoid “heavy” or too many *srmLs* from blocking all slots in SRM server

Other news

- Developing in WLCG a reference implementation of a prestaging tool
  - A Python script using GFAL as an SRM client
  - Can use either *srmStatusOfBringOnline* or *srmLs* to determine if files are online
- Agreed with client developers to improve the behaviour of SRM servers when under heavy load
  - Return a proper code (SRM_INTERNAL_ERROR) instead of “connection timed out”, gSOAP errors, etc.
  - In that case clients will use exponential back-off times

Conclusions

- Focus must be on stability
- Improve SRM scalability
- More support for xrootd
- More coordination among data management client and server developers
- Prestaging tests a must, concurrently from more VOs
- Still unclear if sites and computing systems are ready for the user data analysis
  - extensive tests are foreseen
- Better authorization on storage systems
  - To protect sites AND data