ALICE experiences with CASTOR2

Latchezar Betev

ALICE

Outline

- General ALICE use cases
- DAQ ALICE Data Challenges
- Offline Physics Data Challenges
- Plans for performance/stability tests
- CASTOR2 and xrootd
- Support
- Conclusions

General ALICE use cases

- CASTOR is used as a custodial storage for:
  - RAW and condition data from the experiment – transferred from the disk buffer in ALICE P2
  - Offline production - ESDs, AODs, user analysis results – through the Grid middleware and CAF
  - Direct storage of user files – from applications running on lxbatch

Brief history

DAQ ALICE Data Challenges:

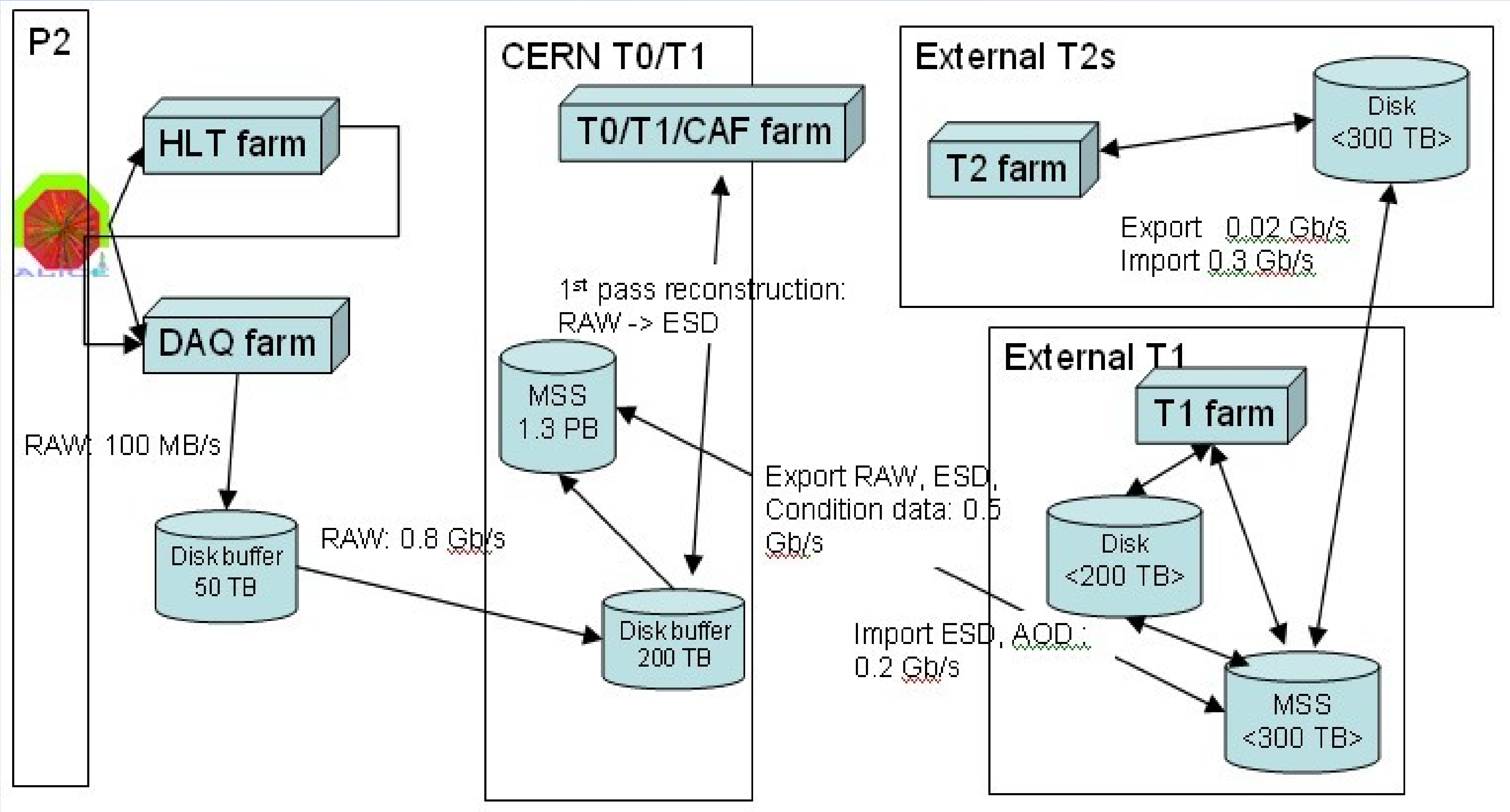
- 2001 – ADC III – CASTOR1, 85MB/sec sustained transfer for one week
- 2002 – ADC IV – CASTOR1, 300MB/sec sustained transfer for one week
- 2004 – ADC VI – CASTOR1, failed to reach the challenge goals of 300 MB/sec
- 2005 – ADC VI – ***CASTOR2,*** 450 MB/sec sustained transfer for a week
- 2006 – ADC VII – ***CASTOR2 (July/August),*** 1 GB/sec sustained for a week
  - Last data challenge before data taking

Offline Physics Data Challenges

- 2004-2005 – PDC’04 – CASTOR1 – storage of simulated events from up to 20 computing centres worldwide to CERN, test of data flow ‘in reverse’
  - Exposed the limitations (number of concurrently staged files) of CASTOR1 stager - partially solved by creating several stager instances
  - Exposed deficiencies of AliRoot – too many small files, inefficient use of taping system
- 2005-2006 PDC’06 – CASTOR2 (ongoing)
  - Tests of data transfers through xrootd and gLite File Transfer System (FTS) – stability of CASTOR2 and Grid tools
  - Goal – up to 300MB/sec from CERN to ALICE T1s, sustained for one week

Data flow schema for first data taking period

- p+p data taking and reconstruction data flow



Brief History (2)

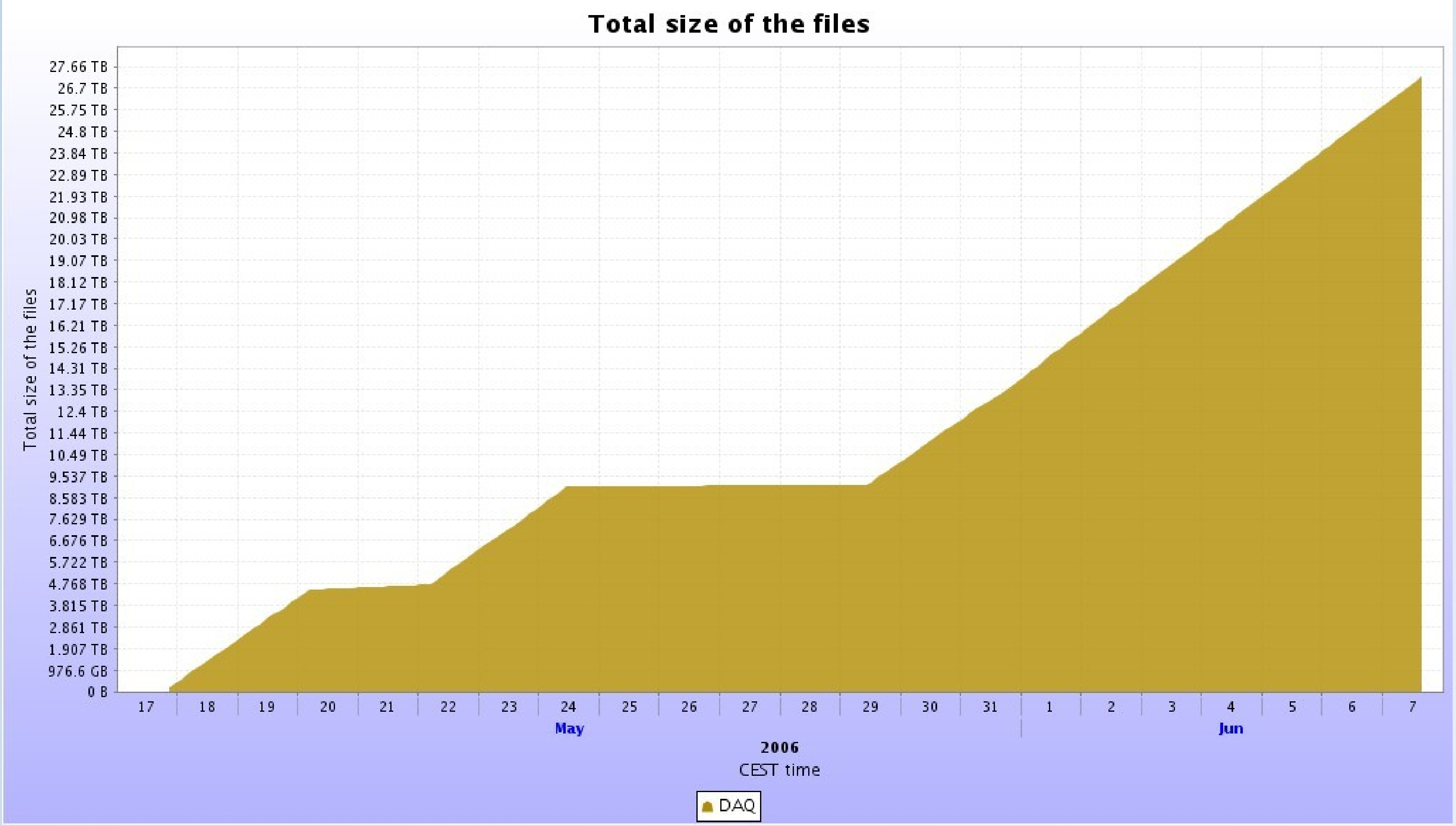
- User access
  - Substantial user interaction with CASTOR1 (one stager)
  - Progressively all ALICE users are migrating to Grid tools, direct interaction with CASTOR is minimized
  - Puts less stress on the system from ‘uninformed’ parties
- Summary
  - ALICE was the first LHC experiment to migrate completely to CASTOR2 – both for major tasks (DAQ, Grid - August 2005) and users (February 2006)

DAQ ADC VII with CASTOR2

- Data is transferred to a ‘no tape’ CASTOR2 service class dedicated for Data Challenges (3 TB) with a garbage collector
- Functional tests of ‘rfcp’, ‘rfdir’… and registration of data in the ALICE Grid catalogue (AliEn)
- Only few files are read back for testing
- No stress tests of resources yet, services interruptions are acceptable
- Production tests (***scheduled for July/August 2006***):
  - With a ‘to tape’ storage class
  - Test of CASTOR2 API using a DAQ software package dedicated to ROOT objectification
  - Online data transfer from the DAQ machines in P2 to CASTOR2

Data rates – example

- In the past 20 days, approximately 27 TB in 35000 files registered in CASTOR2 and AliEn



Current issues

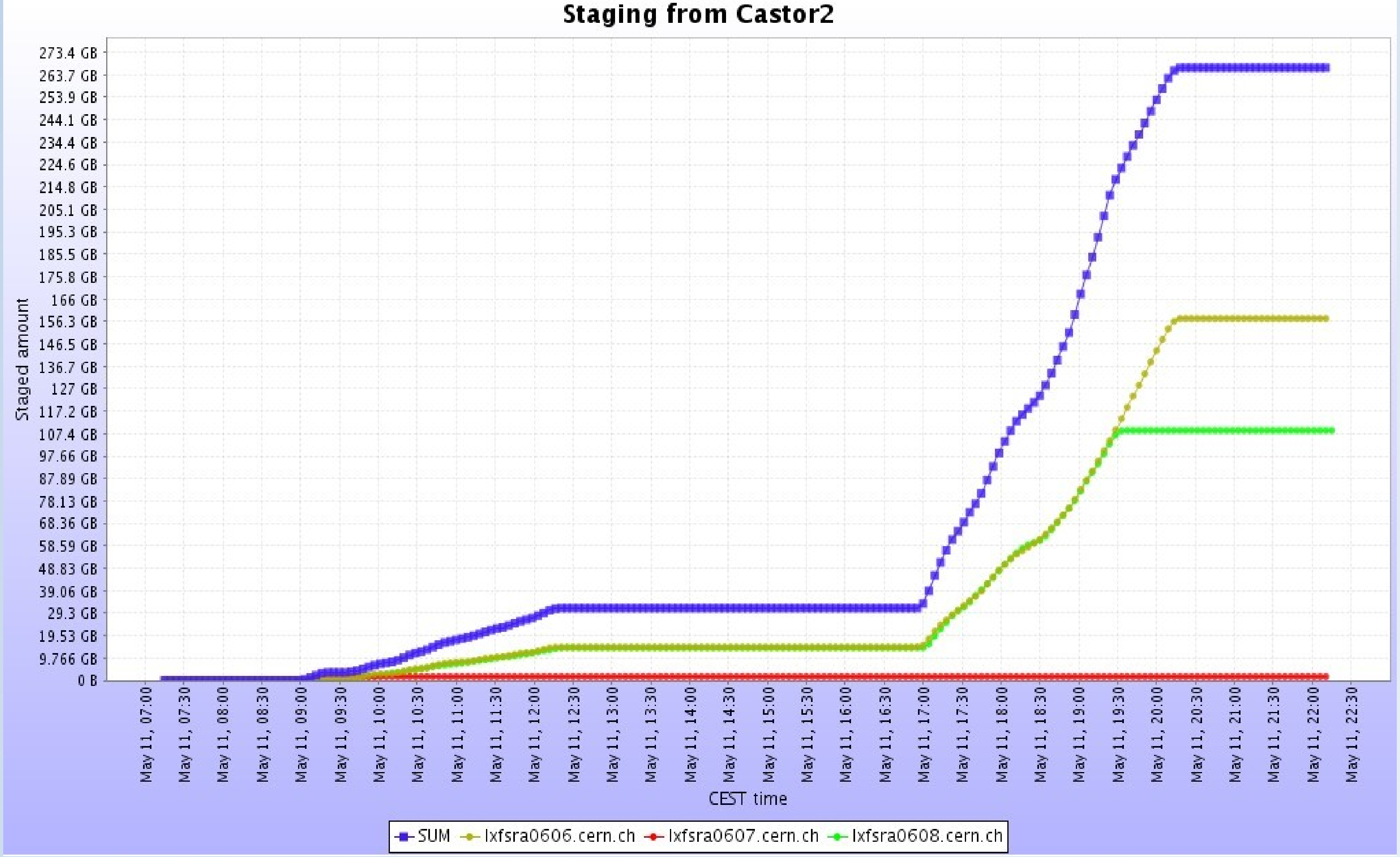
- Modification of rfcp to include calculation of checksum ‘on-the-fly’, submitted to CASTOR development team
- Running a CASTOR2 client on a standard CERN PC require substantial modifications of the default firewall settings
- Adverse effects of power outages on the CASTOR2 services – clients are blocked and cannot recover ‘graciously’

Offline PDC’06 with CASTOR2

- Offline uses a single instance of CASTOR2 (castoralice)
- Files are transferred currently from/to CASTOR2 through 3 dedicated xrootd servers (lxfsra06xx), running a backend migration scripts and through FTS
- ALICE Grid tools pack all output files from an application into an archive for optimization of taping – a reduction of number of files registered in CASTOR by factor of 4 to 30
- Currently there are 40 TB of data in 214K files registered in CASTOR2

Data rates

- Readback of production files from CASTOR2



Current issues (2)

- Addition of monitoring tools for the stagers – re-implementation of 'stageqry –s'
- Reduced latency for interactive file open (currently ~8 sec/file)
- Guesstimate of time needed to stage a file from tape:
  - The Grid tools are 'hiding' the type of storage (MSS, disk) from the user/application
  - Naturally, the behavior of these two basic storage types is different
  - Any file stored in a MSS (like CASTOR) can be returned immediately (if it is in the MSS disk cache) or be delayed if it is on tape and has to be staged
  - In the second case the MSS should return an estimate (f.e. based on the information in the staging queue) when the file will be reasonably available to the application
  - This will help optimize the Grid tools and job behavior

Test plans in July/August 2006

- Part of the integrated DAQ ADC VII, Offline PDC’06 and the LCG SC4:
    - Test of CASTOR2 API (with xrootd) – clarification later in the talk, using a DAQ software package dedicated to ROOT objectification
    - Online data transfer between DAQ "live" streams and CASTOR2 from the DAQ machines in P2, rate 1 GB/sec for 1 week
    - FTS transfers from CERN CASTOR WAN pool to 6 T1s, rate 300 MB/sec aggregate from CERN for one week (export of RAW data)
    - Functional tests of the CASTOR2-xrootd interface
    - Full reconstruction chain RAW data -> CERN first pass reconstruction storage and export, T1 second pass reconstruction and storage
- The above will test both the throughput and stability of the CASTOR2 service for all basic computational tasks in ALICE

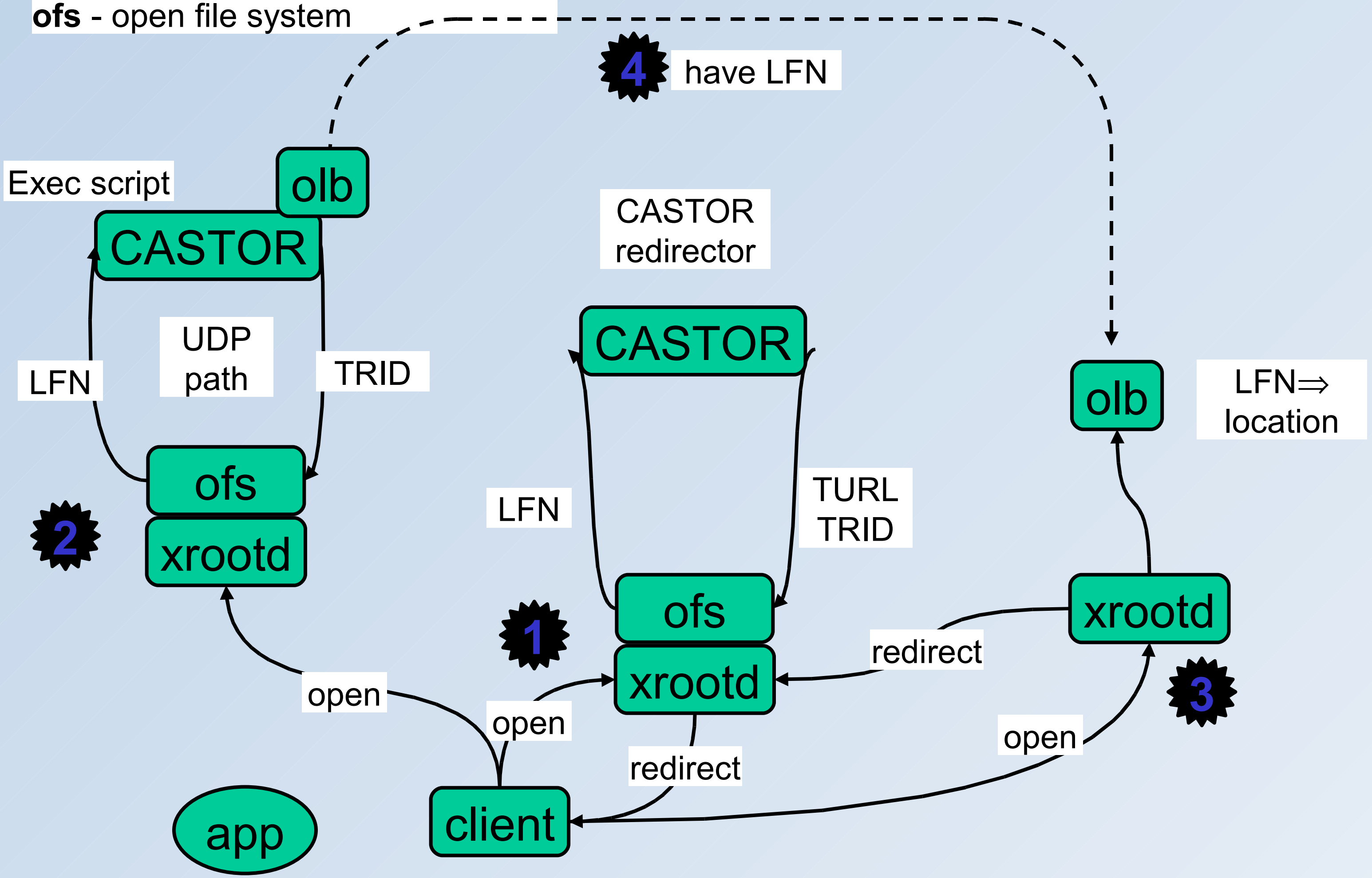
xrootd-CASTOR2 interface

- xrootd – file server/disk manager and a transport protocol developed at SLAC, incorporated in ROOT and as such a natural choice for ALICE
- ALICE will to use ***only*** xrootd from the applications to access data stored anywhere on any storage system:
  - Avoids the need for the applications to carry multiple libraries for different protocols and storage systems (issue on the Grid/CAF)
  - Avoids the splitting of the experiment's disk pool in many sections
- In March 2006, discussion have started between the CASTOR2 and xrootd development teams to incorporate xrootd in CASTOR2
  - This has resulted in a prototype implementation

Architecture

olb - open load balancing

ofs - open file system



Explanation of interactions

- Forward an open request to CASTOR, the xrootd redirector (via CASTOR) provides the best copy to open
- For CASTOR, schedule the I/O (1st open only)
- Better option again stack a redirector in front of everything, perhaps with normal disks
  - If the file is on disk, redirect to normal disks or to CASTOR disks buffer
  - If the file is on tape, redirect as before to CASTOR for staging
- CASTOR stager would tell the olb of xrootd that the file is on disk for subsequent immediate opens

❖ Similar implementation in other storage management systems (DPM, dCache)

Experiences with CASTOR support

- ALICE was the first experiment to use CASTOR2 on extended basis:
  - As such, we have met with some ‘teething’ problems of the system
- The stability of service has improved greatly in the past several months
- The reaction to problems was/is typically very fast and competent
  - Few (named) experts are almost always on-line
  - We are worried about support sustainability, especially with many concurrent and continuously running exercises

Conclusions

- ALICE has successfully migrated to CASTOR2
- Initial tests of the system has shown that the basic functionality required by the experiment's DAQ, Grid and user software is adequate
  - However the performance and stability tests are still to be done
- Concern: at the time of this review, we can only partially answer the set of questions asked by the review committee
- ALICE's major development requirement is the integration of xrootd in CASTOR2
  - A good progress has been made in few (short) months
- Few minor issues with functionality/monitoring - listed within the presentations
- The CASTOR2 team response to problems is very quick and competent
  - This will also be tested extensively during the scheduled July/August exercises