Evoluzione dell'accesso allo storage e della gestione dati

Giacinto DONVITO
INFN-Bari

Slides: Vincenzo Vagnoni, Stefano Bagnasco, Daniele Bonacorsi, Alessandro De Girolamo, e altri

Outlook

- Data Management activity report from experiments
  - Experiments global activities
  - INFN related contribution

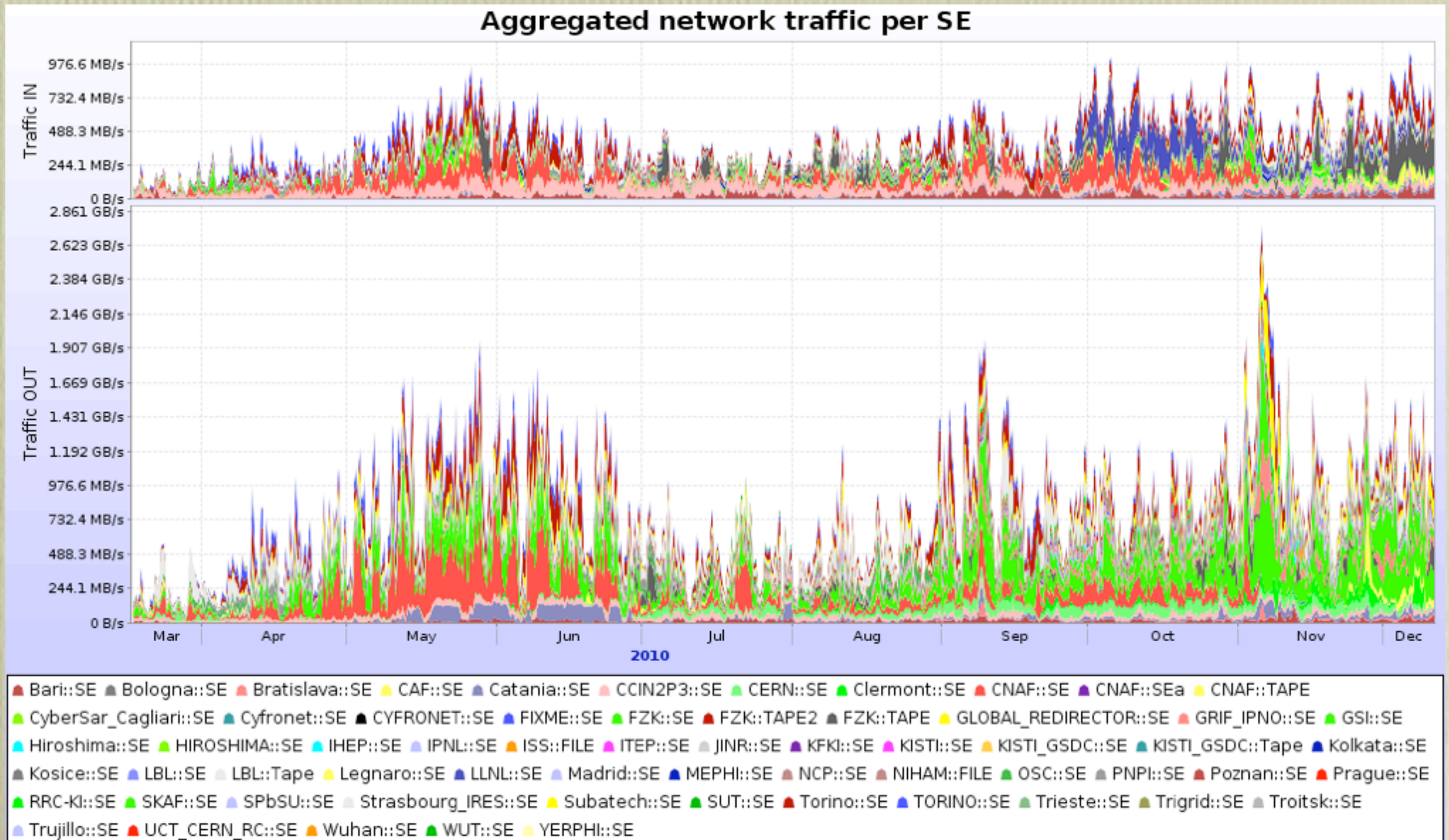
- CNAF experience with data management in last year of data taking

Overall status of experiment's data management

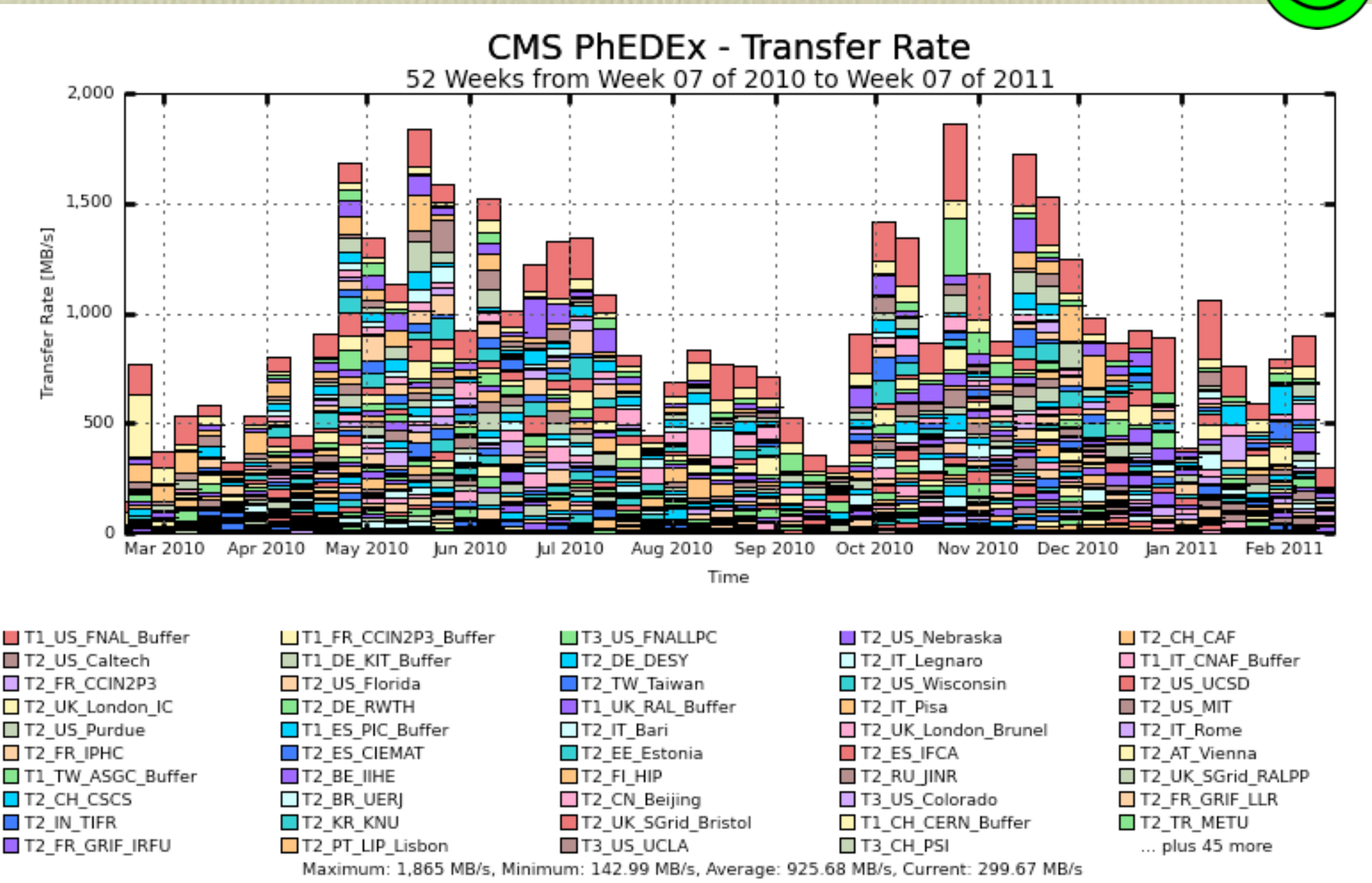
- Generally speaking data management/access is able to cope with the amount of data available at the moment

- Still few open issues:
  - Tape
  - Disk usage
  - number of accesses per dataset
    - Job queue time, site load, etc
  - human effort
    - Tier3, Tier2 deletion requests

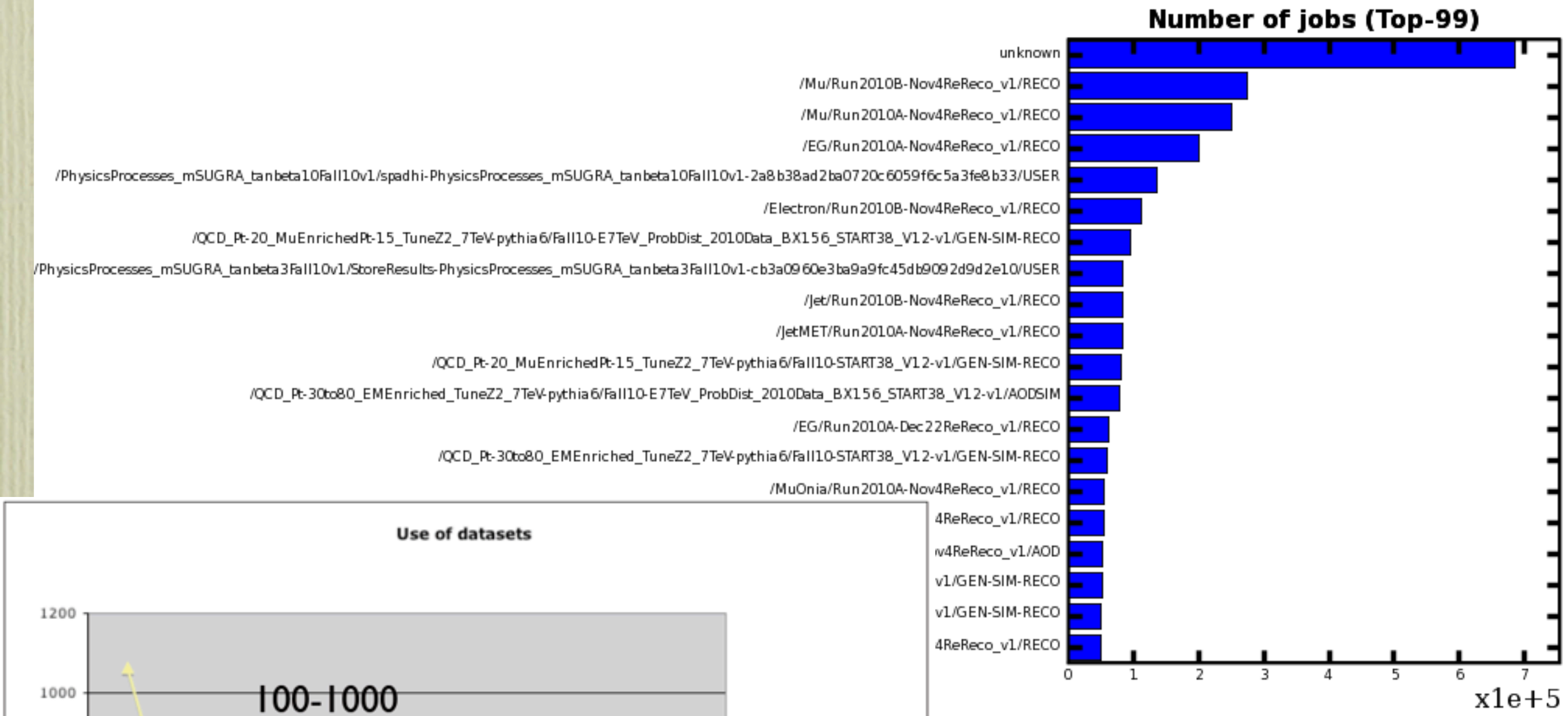
ALICE SE USAGE

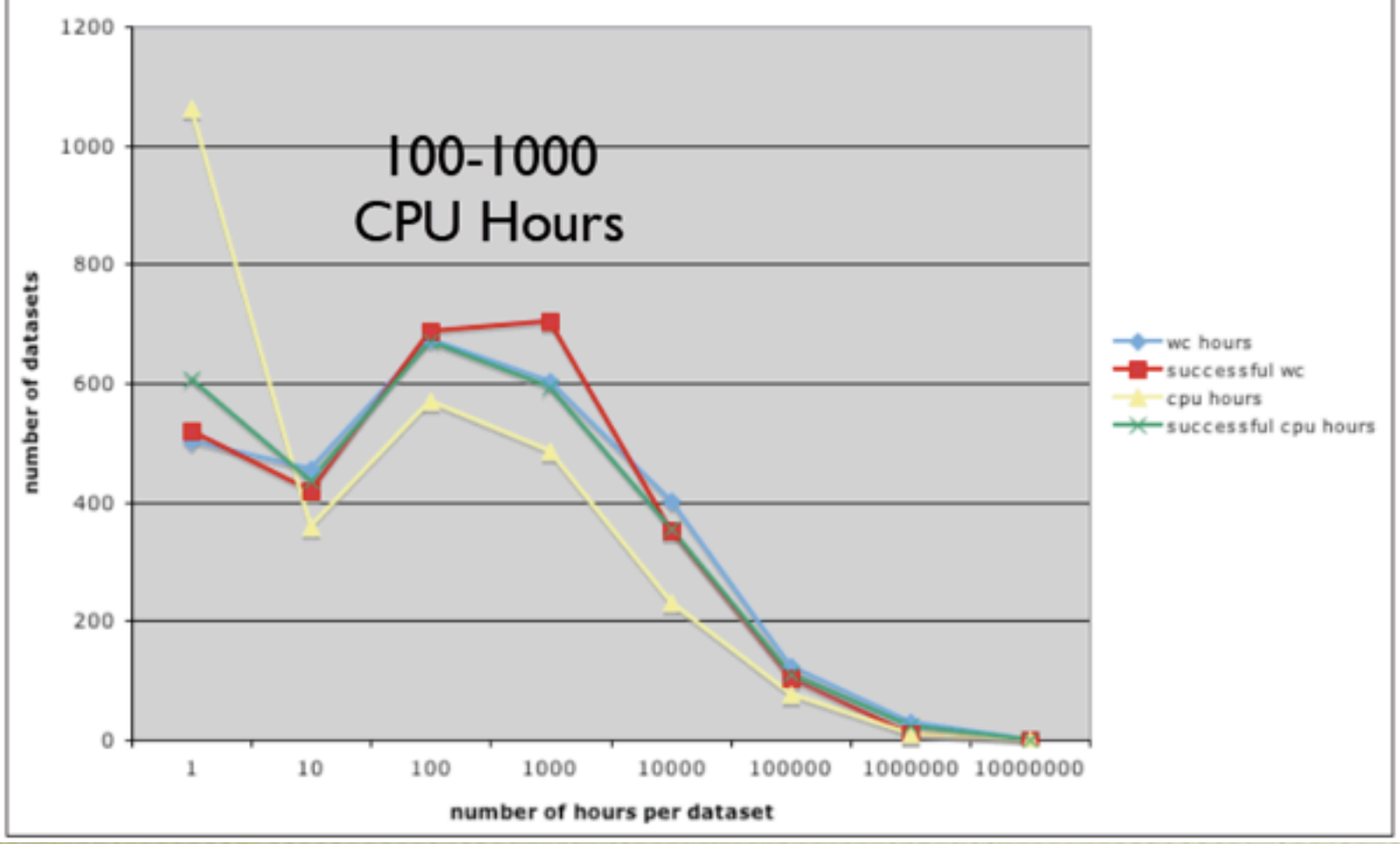


CMS WAN Transfers rate



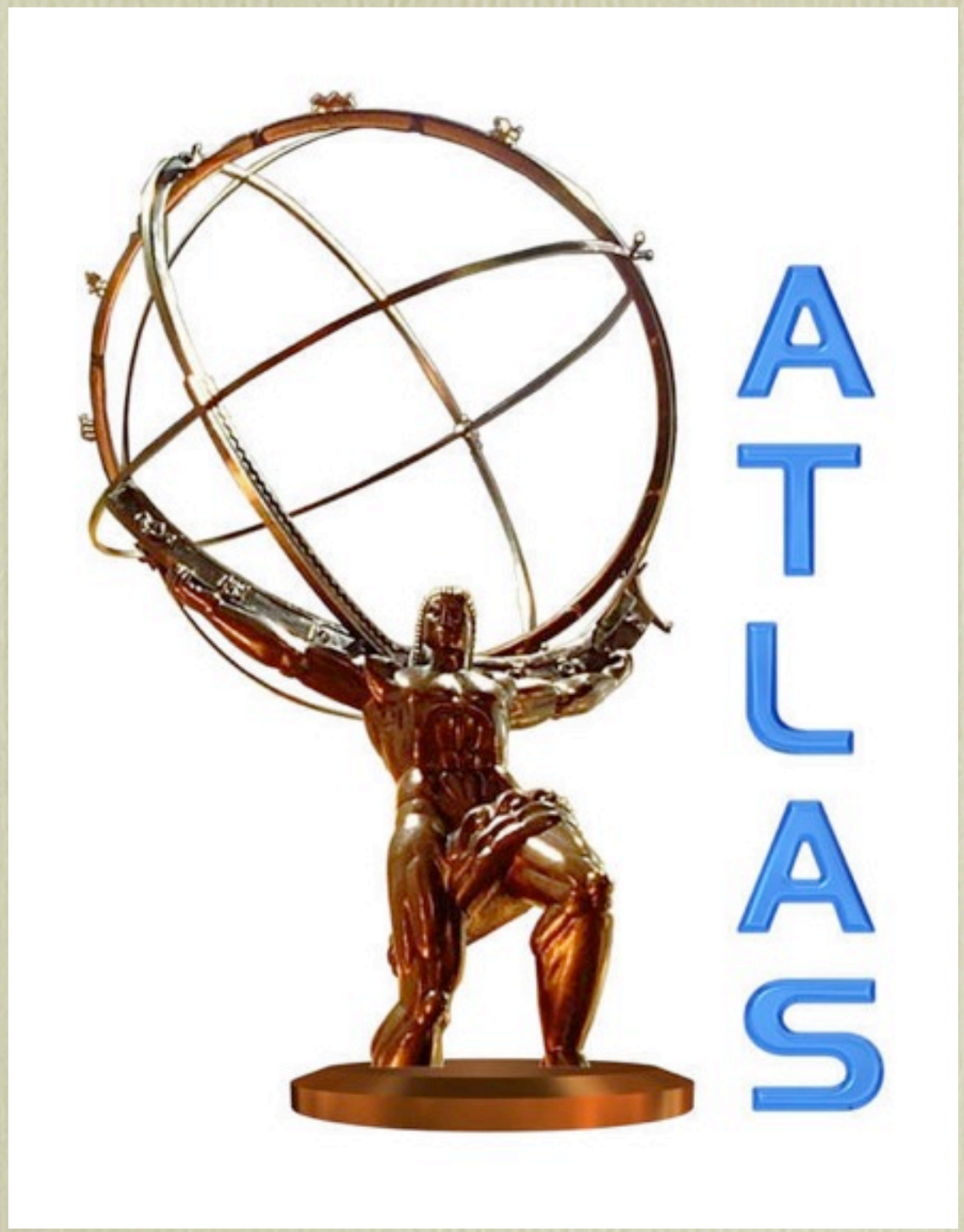
Destination Sites


Number of jobs (Top-99)
unknown
/Mu/Run2010B-Nov4ReReco_v1/RECO
/Mu/Run2010A-Nov4ReReco_v1/RECO
/EG/Run2010A-Nov4ReReco_v1/RECO
/PhysicsProcesses_mSUGRA_tanbeta10Fall10v1/spadhi-PhysicsProcesses_mSUGRA_tanbeta10Fall10v1-2a8b38ad2ba0720c6059f6c5a3fe8b33/USER
/Electron/Run2010B-Nov4ReReco_v1/RECO
/QCD_Pt-20_MuEnrichedPt-15_TuneZ2_7TeV-pythia6/Fall10-E7TeV_ProbDist_2010Data_BX156_START38_V12-v1/GEN-SIM-RECO
/PhysicsProcesses_mSUGRA_tanbeta3Fall10v1/StoreResults-PhysicsProcesses_mSUGRA_tanbeta3Fall10v1-cb3a0960e3ba9a9fc45db9092d9d2e10/USER
/Jet/Run2010B-Nov4ReReco_v1/RECO
/JetMET/Run2010A-Nov4ReReco_v1/RECO
/QCD_Pt-20_MuEnrichedPt-15_TuneZ2_7TeV-pythia6/Fall10-START38_V12-v1/GEN-SIM-RECO
/QCD_Pt-30to80_EMEnriched_TuneZ2_7TeV-pythia6/Fall10-E7TeV_ProbDist_2010Data_BX156_START38_V12-v1/AODSIM
/EG/Run2010A-Dec22ReReco_v1/RECO
/QCD_Pt-30to80_EMEnriched_TuneZ2_7TeV-pythia6/Fall10-START38_V12-v1/GEN-SIM-RECO
/MuOnia/Run2010A-Nov4ReReco_v1/RECO
4ReReco_v1/RECO
v4ReReco_v1/AOD
v1/GEN-SIM-RECO
v1/GEN-SIM-RECO
4ReReco_v1/RECO
0
1
2
3
4
5
6
7
x1e+5

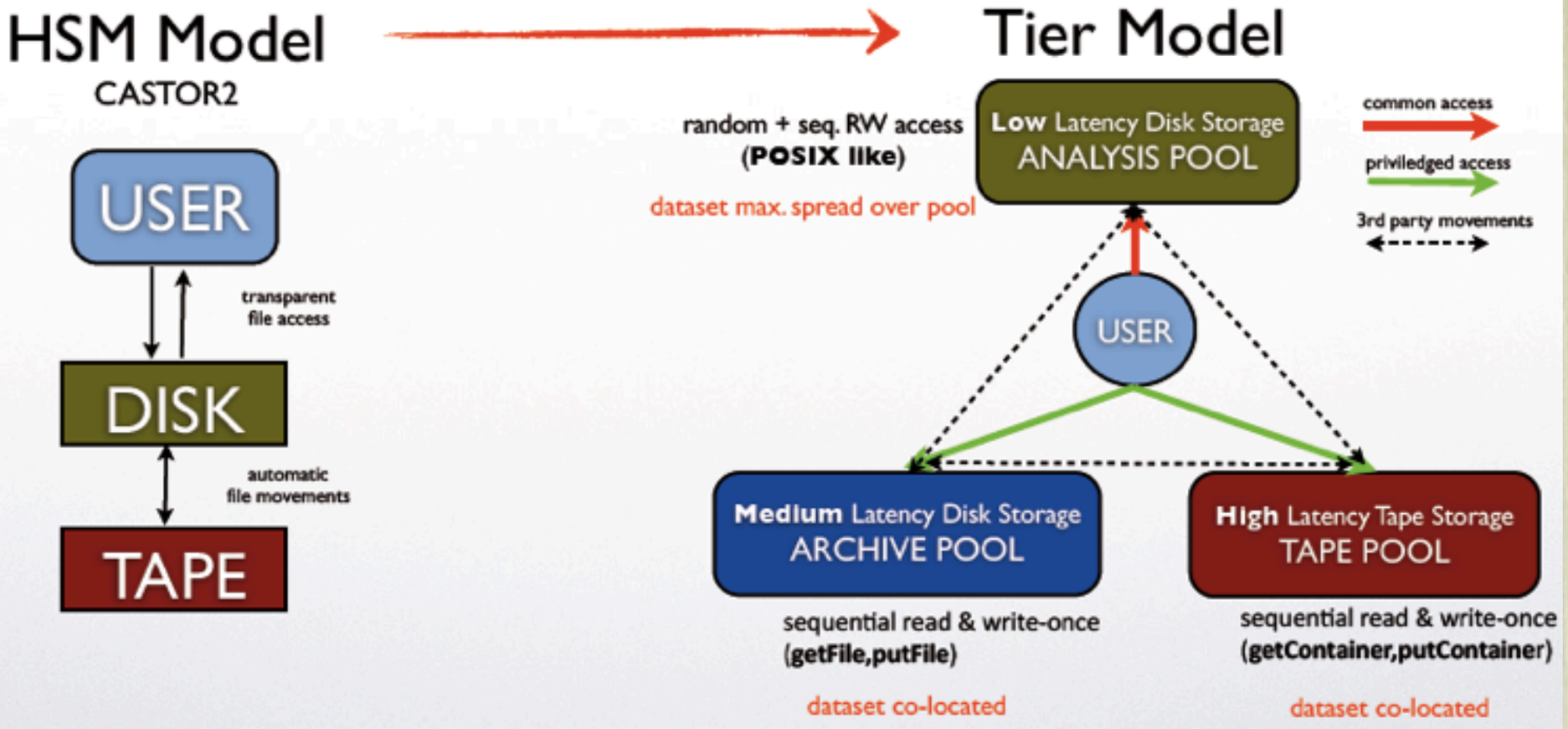

Use of datasets
100-1000
CPU Hours
number of datasets
1200
1000
800
600
400
200
0
1
10
100
1000
10000
100000
1000000
10000000
number of hours per dataset
wc hours
successful wc
cpu hours
successful cpu hours

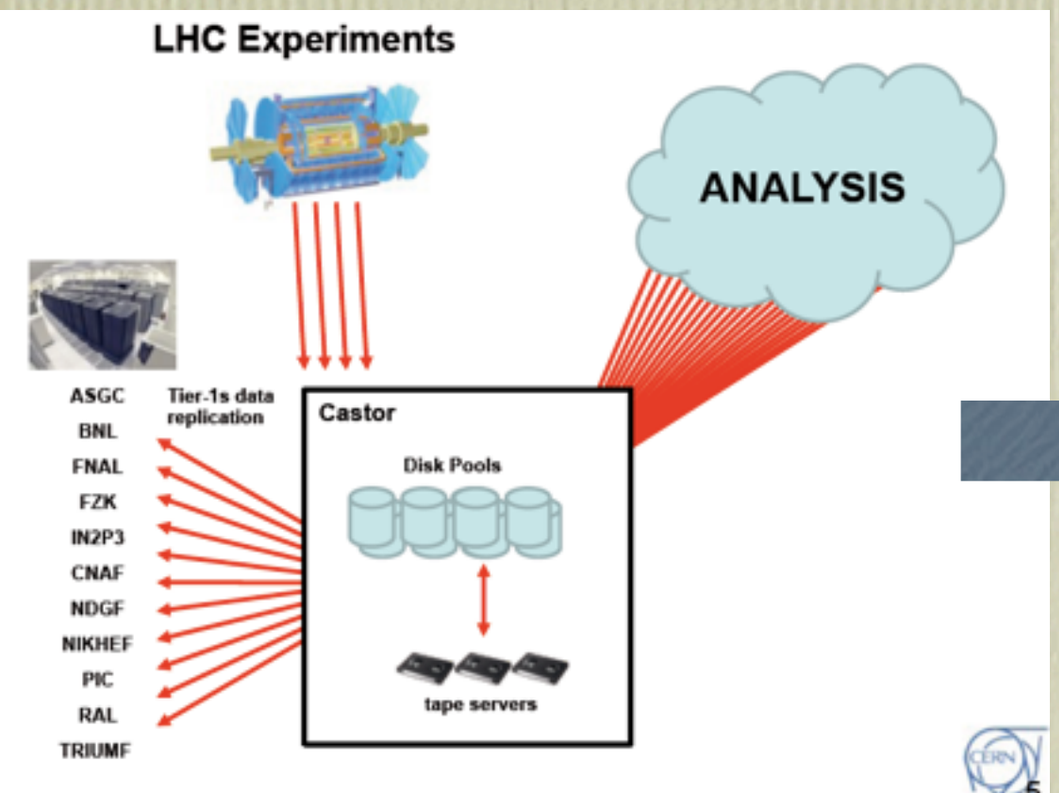

shutterstock

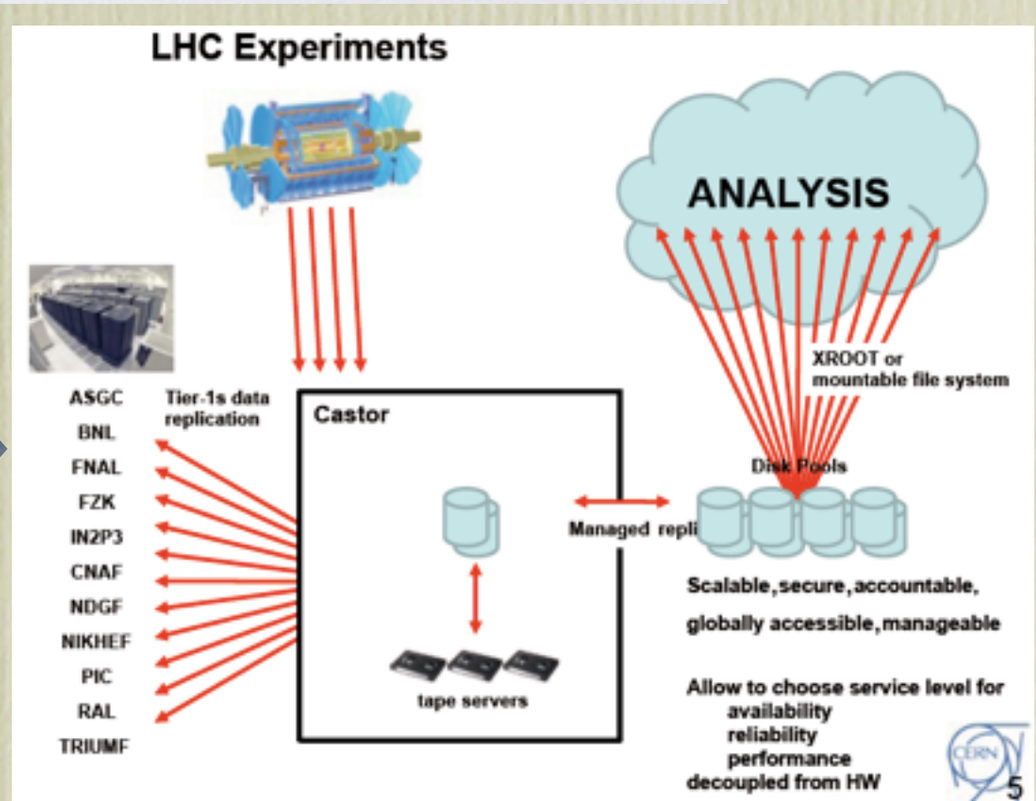
ATLAS



Model Transition







CERN is trying to move away from a central monolithic solution

EOS Architecture

Management Server

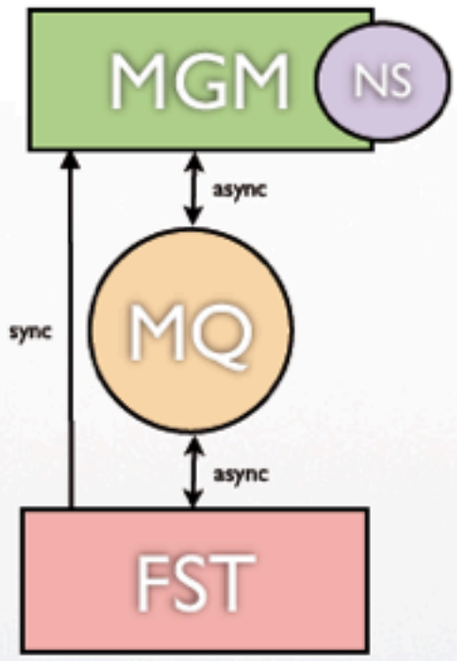
Pluggable Namespace, Quota
Strong Authentication
Capability Engine
File Placement
File Location

Message Queue

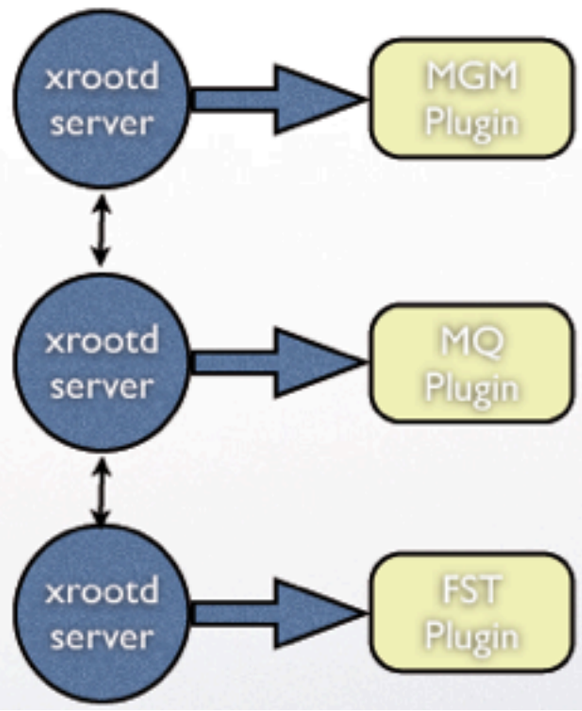
Service State Messages
File Transaction Reports

File Storage

File & File Meta Data Store
Capability Authorization
Checksumming & Verification (adler,crc32,md5,sha1)
Disk Error Detection (Scrubbing)



Implemented as plugins in **xrootd**



EOS NS Scalability

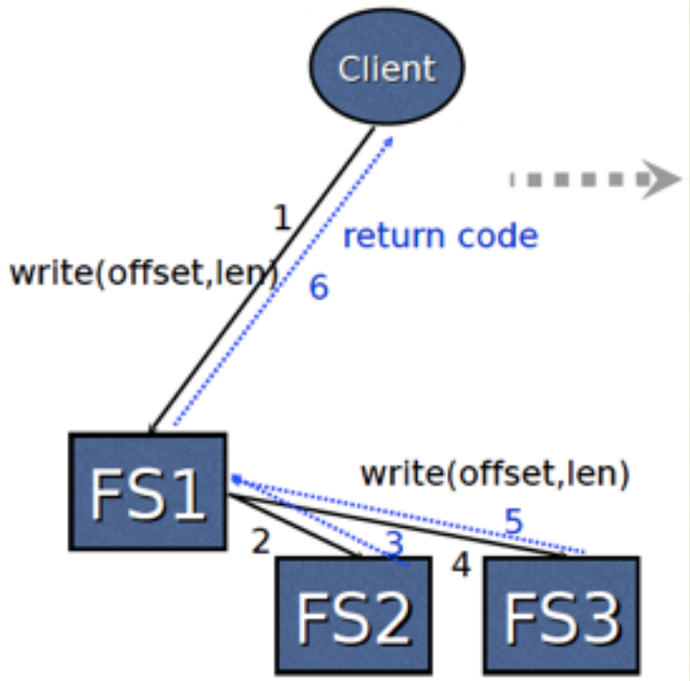
Namespace	V1	V2*
Inode Scale	100 Mio. inodes	1000 Mio inodes
In-Memory Size	80-100 GB (replicas have minor space contribution)	20 GB x n(replica)
Boot Time	~520 s **	15-30 min ** (difficult to guess)
Pool size assuming avg. 10 Mb/file + 2 replicas	2 PB	20 PB
Pool Nodes assuming 40 TB/node	50	500
File Systems assuming 20 / node	1.000	10.000

* estimates
** boot time plays a minor role with active/passive MGM pairs

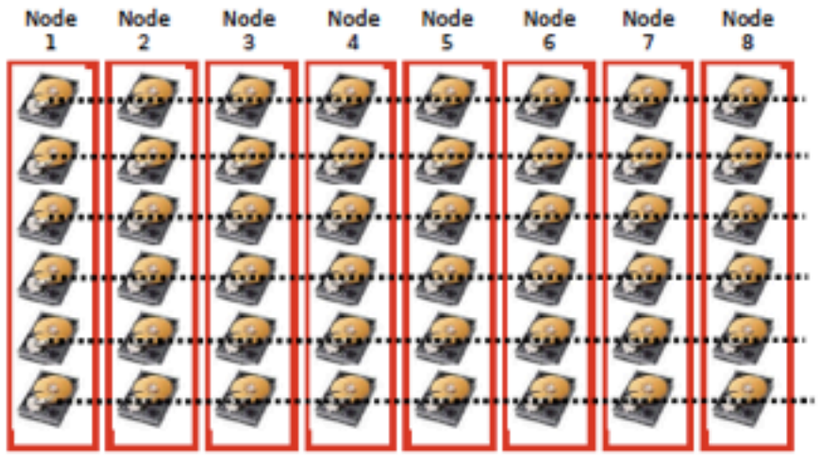
EOS Philosophy

- Storage with single disks (JBODs-no RAID arrays)
  cheap & unreliable
- Network RAID within node groups
  (scheduling groups & round-robin rings)
- Self-healing
  from a clients point of view all files are always readable & writable
- Online filesystem migration
- Tunable quality of service
  via redundancy parameters
- Tradeoff in Scalability vs Latency (pluggable hierarchical namespace - scale out for read - scale out for write only by namespace split)

Replica layout

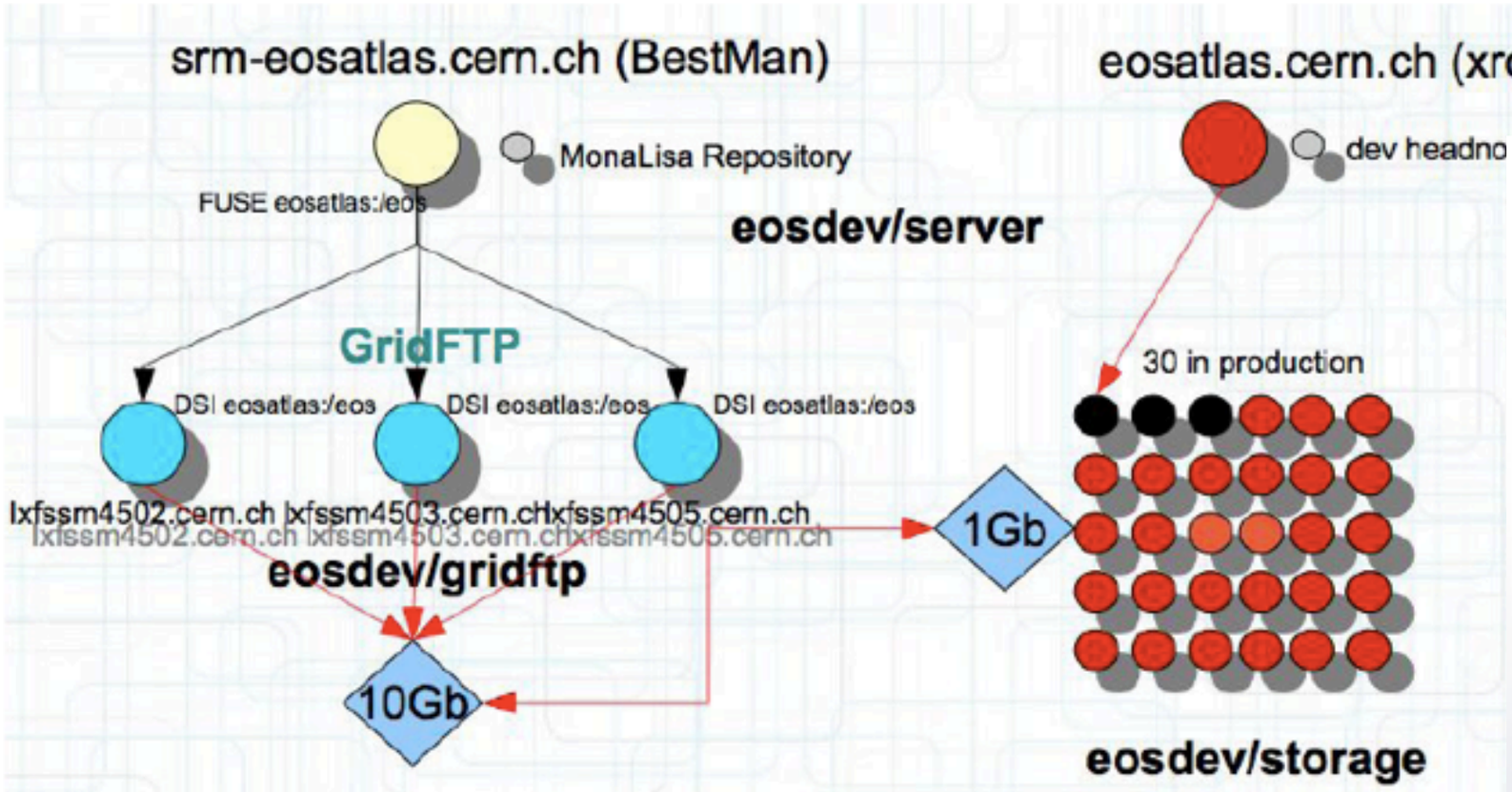


Replica placement



In order to minimize the risk of data loss we couple disks into scheduling groups (current default is 8 disks per group)

- The system selects a scheduling group to store a file in in a round-robin
- All the other replicas of this file are stored within the same group
  - Data placement optimised vs hardware layout (PC boxes, network infrastructure, etc...)



EOSATLAS Usage

1.11.2010-05.01.2011 66 days

File Opened Read	Data Read	File Opened Write	Data Written	Logical Space	Inodes max used
1.9 M	2.12 PB	3.2 M 1.6 M*	3.55 PB 1.77 PB*	650 + 40 TB	0.9M Files 31k Directories

* incl. replication factor 2

830k FTS transfers
nominal import at **1.1-1.3 GB/s**

Initial Setup August 2010

Space (11/12 RAID-0 FS per Server).

data	958 TB	45	sched. groups
user	344 TB	15	sched. groups
scratch	93 TB	5	sched. groups

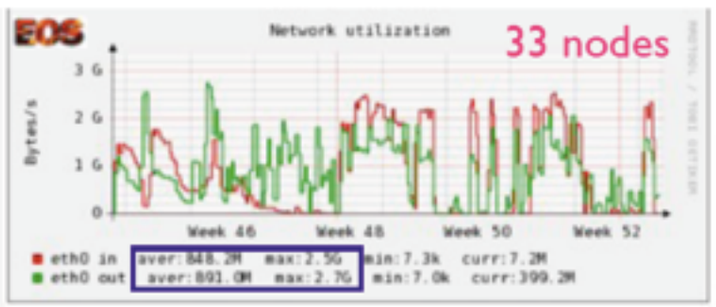
Read test

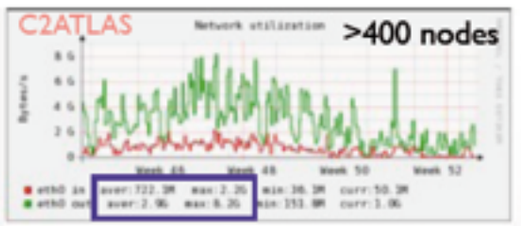
NS Size: 10 Mio Files
* 100 Million read open
* 350 ROOT clients 7 kHz
* CPU usage 20%

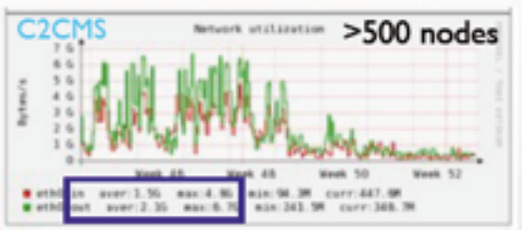
Write test

NS Size: 10 Mio Files
* 22 ROOT clients 1 kH
* 1 ROOT client 220 Hz

EOSATLAS Usage







Scale Comparison

EOS	C2ATLAS	C2CMS	C2ALL
Nodes	8%	7%	2%
IO Read	30%	42%	11%
IO Write	110%	56%	28%

EOS Server well tested:
avg. running at 25% of available IO bandwidth

ES LST/EOS schedule

CERN IT Department

- July - Kick-off meeting, draft test plan proposed
- August - Phase 1, 6 weeks:
  - Preparation and pre-testing
- October - Phase 2, 4 weeks:
  - Tuning
- November - Phase 3, 3 months:
  - Test Running

July 6th:
First draft proposed
https://twiki.cern.ch/twiki/bin/view/Main/LST2010

August:
DDM, FTS and panda setting up

October:
Up to 2k HC parallel jobs running

18 Nov
ANALY_CERN_XROOTD
EOSATLAS in production

- Large Scale Test on EOS is still ongoing
  - Cap to 500 parallel jobs running now removed
  - Direct access has been successfully tested (functional tests, still to run stress tests)
- LST plan was effective to evaluate up to here the behavior of EOS as an ATLAS Grid Site
  - Can LST 'procedures' be re-used to perform similar test of different storage technologies?
  - …. EOS ….

Dynamic Data Placement

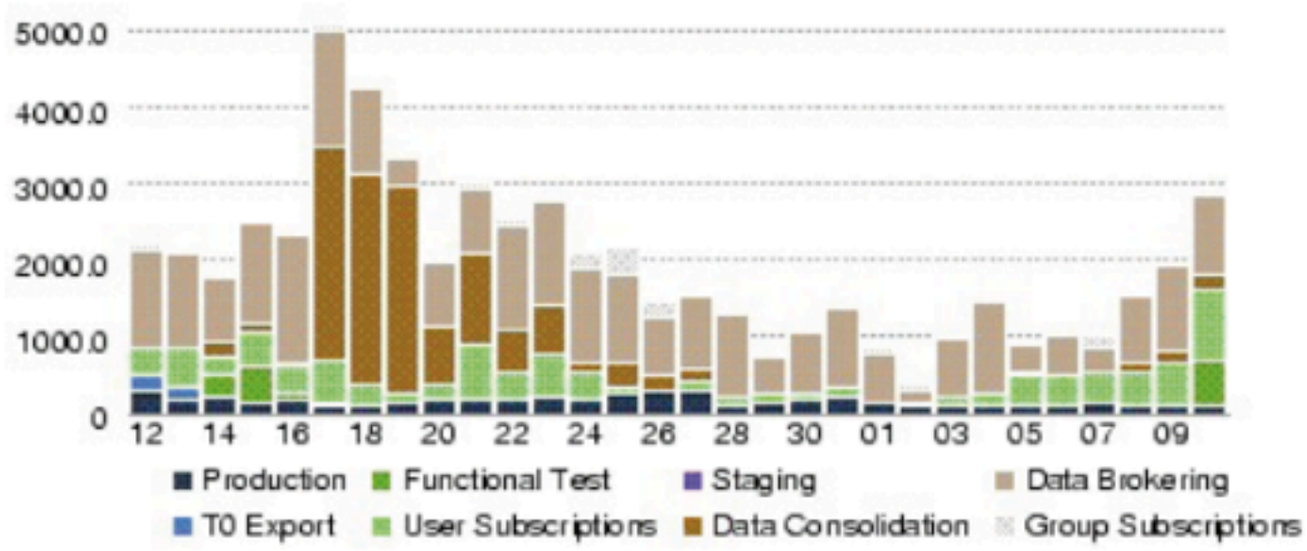
Problems with pre-placement strategy of early LHC data:

- Big fraction of data is not used
- Suboptimal usage of network and storage resources: Uninteresting data is preplaced equally to the interesting data

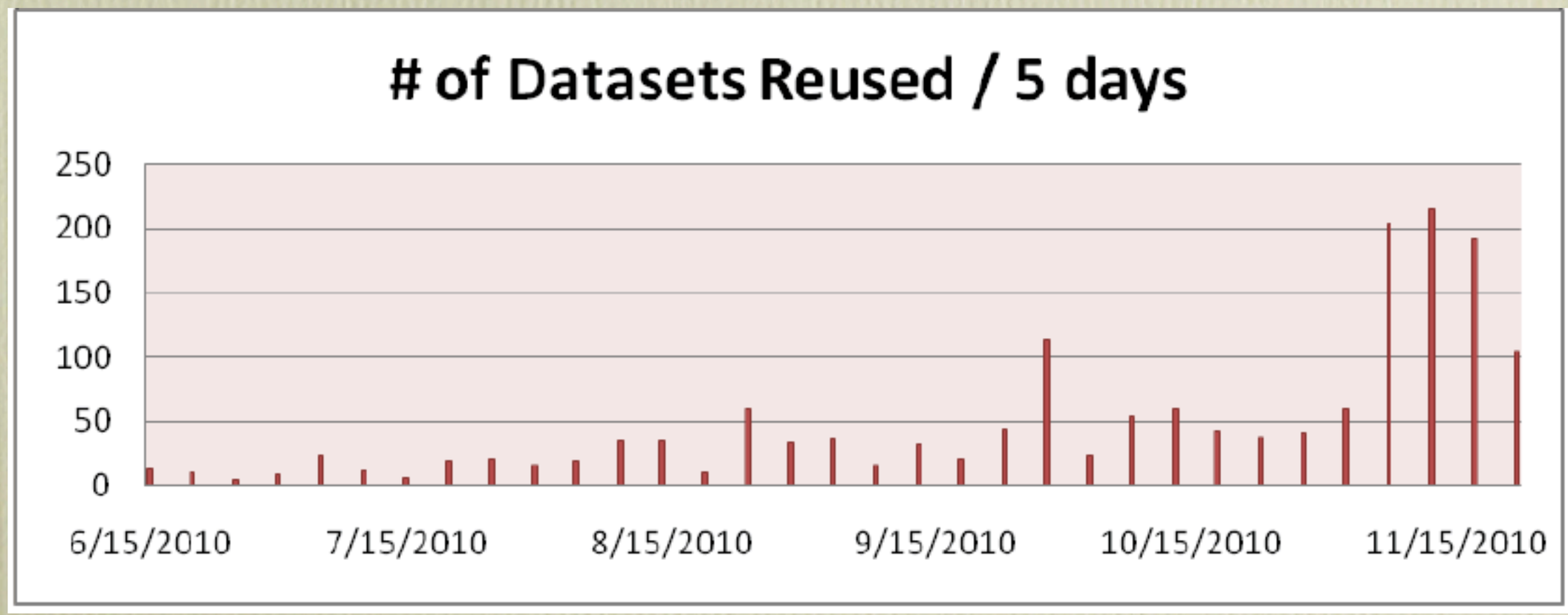
Evolution: PanDA Dynamic Data Placement PD2P

- Data is only preplaced at T1s
- Trigger secondary replication to T2s for **used** data
- Replication is based on decisions taken by the workload management system. It is **not** based on DDM Popularity

1. Data is preplaced at T1s only
2. User submits analysis jobs on a particular dataset
3. PanDA runs analysis jobs in the T1 initially and simultaneously triggers additional replication requests to a T2
   - Consideration: PanDA server is centralized
   - T2 selection is based on free space, queue depth, past performance…
4. Once the dataset has been replicated to the T2, pending jobs at the T1 can be re-brokered
5. If the dataset is considered very hot the dataset can be replicated further
   - based on the backlog of jobs
6. Cleanup of unused datasets done by Victor (see following slides)



- PD2P now responsible for significant data movement on the grid

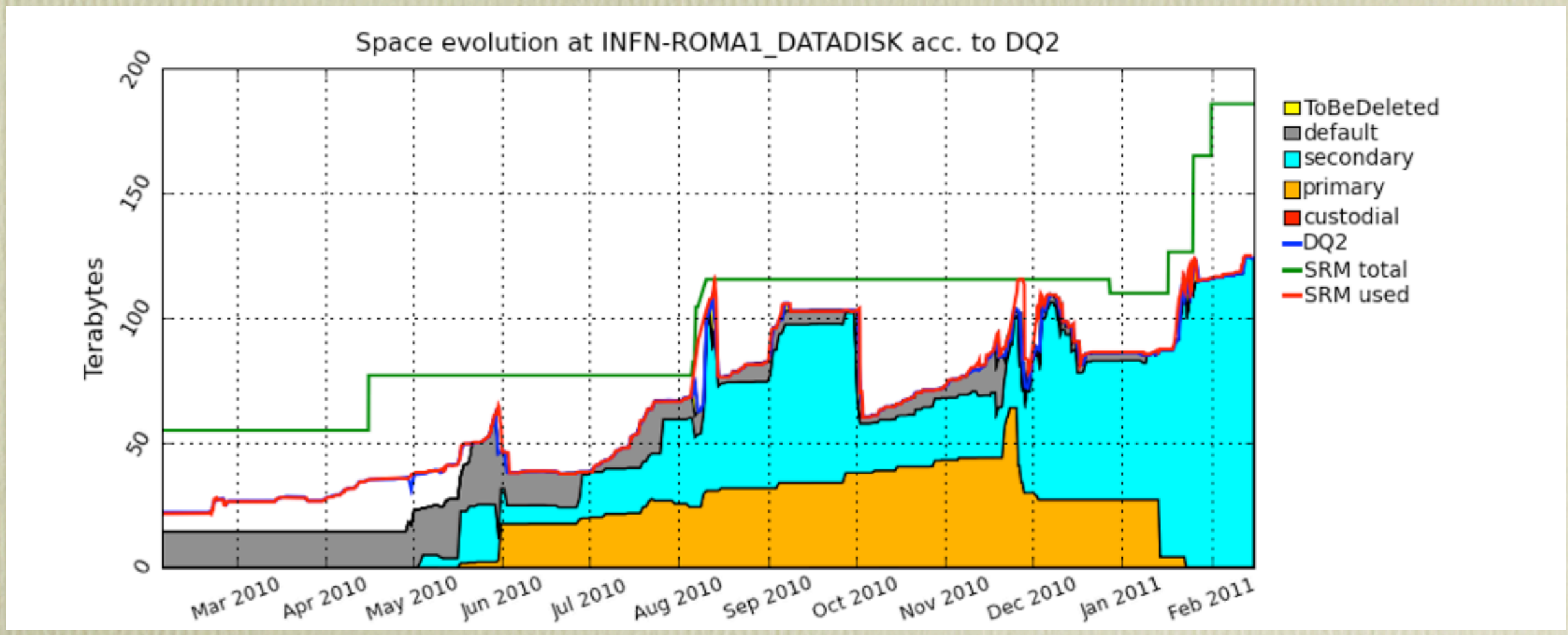


Automatic site cleaning: Victor

- Fully automated system for cleaning that replaces laborious (occasionally error prone) manual work
- Reduction of **secondary** replicas only
  - Downgrading primary to secondary is a decision done by the physics coordination
  - In case of disaster data can be re-replicated again (but no disasters have happened so far!!!)
- Running on **centrally managed spacetokens**
- Allows ATLAS to fully use the deployed storage space
- Built on top of existing DDM components: DDM Storage Accounting, DDM Popularity and Centralized Deletion Service

Atlas Data evolution

- Direct I/O from the storage
- Remote (WAN) reading (see CMS xrootd activity)
- Broke cloud boundaries => “big tier2” (in italy: napoli e roma)



- Custodial => Tape
- Primary => “Master” copy on disk (not to be deleted)
- Secondary => “Cache” copy on disk (could be deleted)

CMS

Groups Short Term

- WAN Data Access
  - Roll-out of xrootd redirector
  - Operational Issues
  - Timescale, how to measure success
  - Benefits of caching at Tier-3s, test plans
- Data Popularity
  - Development Needed
  - Schedule for potential functionality
- CERN Analysis Disk
  - Testing needed
  - Proposed new architecture for Default pool
  - Schedule for roll-out
  - Impact on workflows

Short term activities

Longer term activities

- A number of forward looking items were addressed
  - More automated dynamic data placement.
  - Broader use of of the wide area access to data
  - Dynamic use of local storage
  - Improving IO by orders of magnitude
  - Plans for data archives
  - Report from the HEPIX Storage Working Group

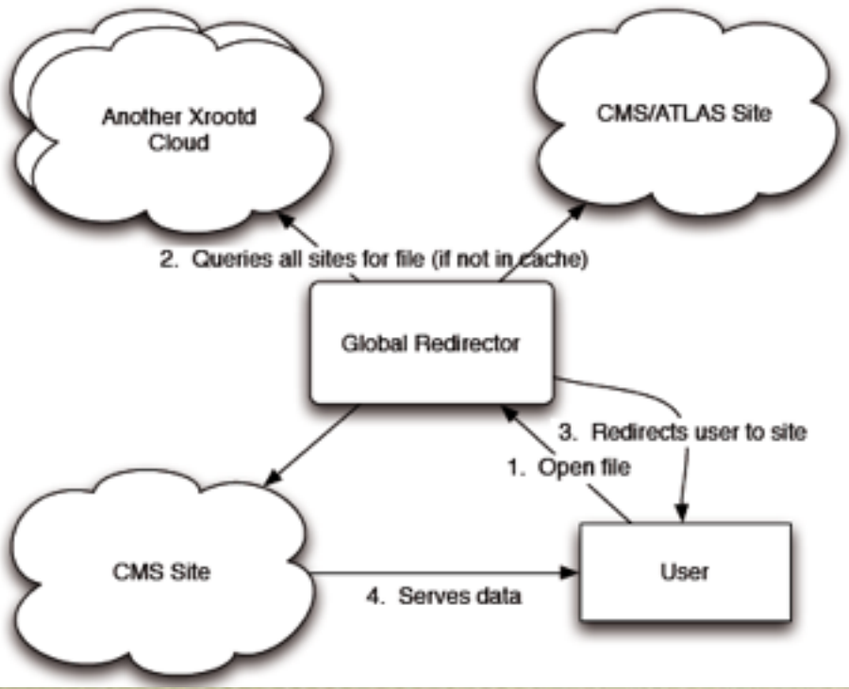
Targets

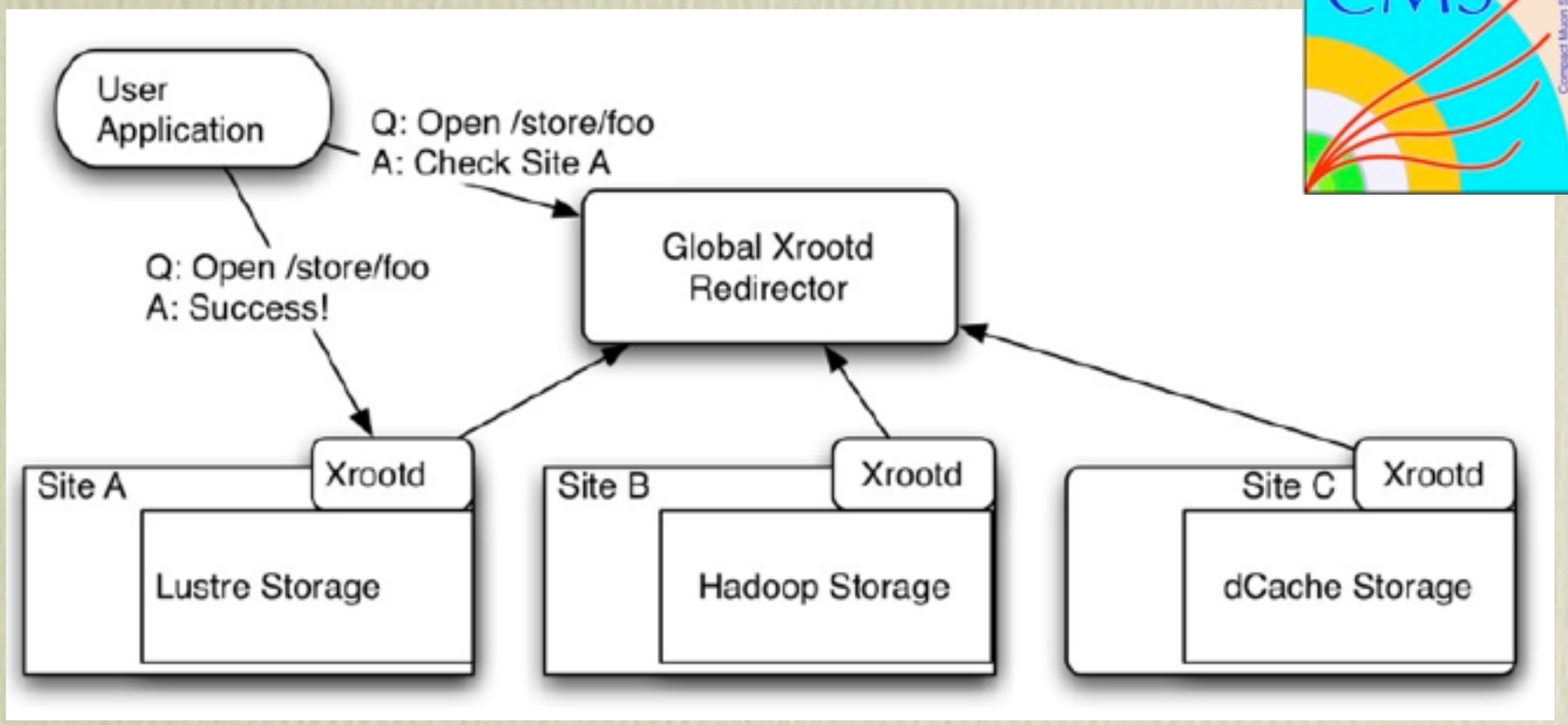
- The target users of this project:
  - *End-users*: event viewers, running on a few CMS files, sharing files with a group.
  - *T3s*: Running medium-scale ntuple analysis
- None of these users are well-represented by CMS tools right now.
  - So, a prototype is better than nothing...

Xrootd Prototype

- Have a global redirector users can contact for all files.
  - Can use any ROOT-based app to access the prototype infrastructure! Each file only has one possible URL
- Each participating site deploys at least 1 xrootd server that acts like a proxy/door to the external world.

Global Xrootd Federation





CMSSW Improvements

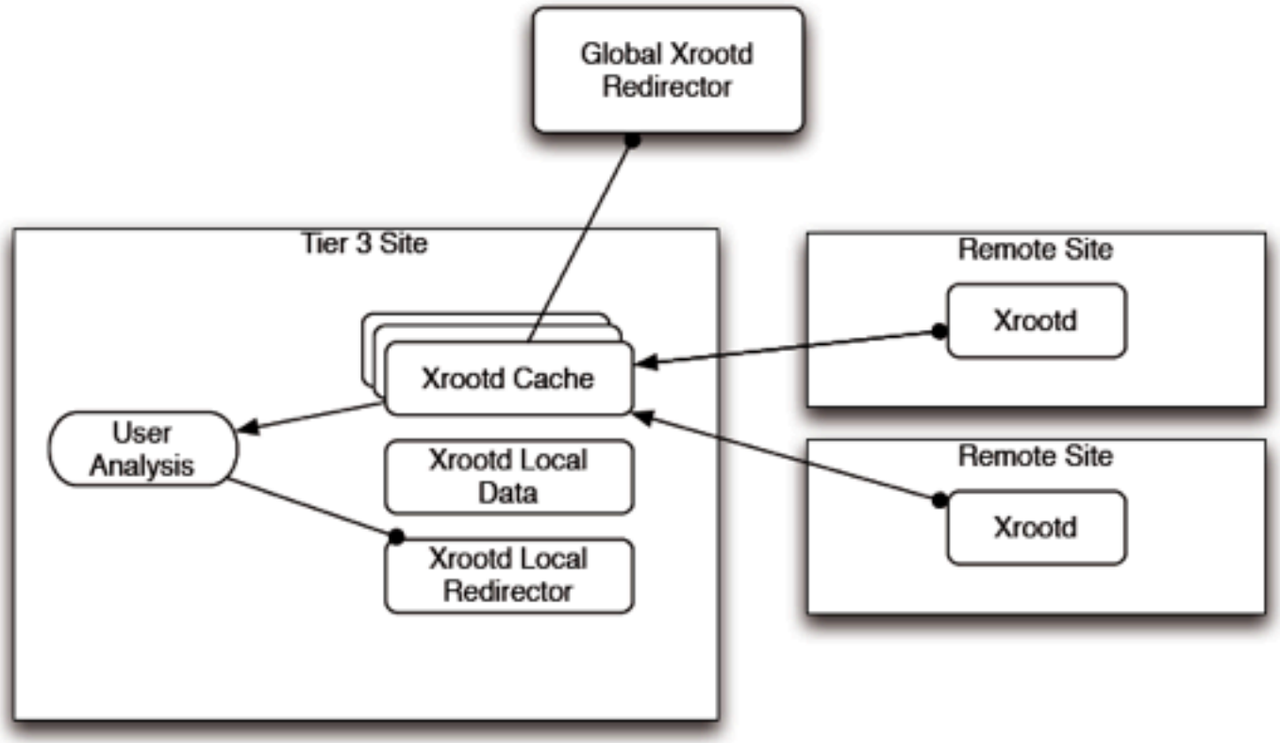
- In order to improve WAN streaming performance, we worked hard with the CMSSW team to optimize the I/O code.
- A sample, I/O-intensive analysis of 60k evts reading data from FNAL dCache/Xrootd:

Site	Ping time	Wall time
FNAL	.1ms	80s
Nebraska	17ms	80s
CERN	128ms	161s

T3 Benefits

- A T3 no longer needs to learn CMS data movement tools to access data.
- If the T3 is xrootd-based, we can use caches to improve data locality.
- If the T3 is not xrootd-based, they can just "fall back" to the global T3 cluster if the file is not local.

Caching Case

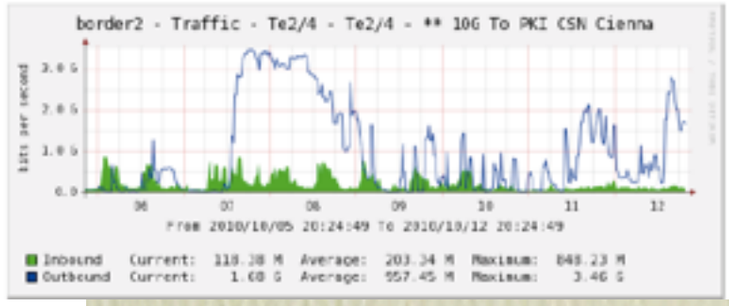


Notice xrootd can download from multiple sites at once!

This helps one avoid overloaded sites; bittorrent-like.

Example: T3 at Omaha

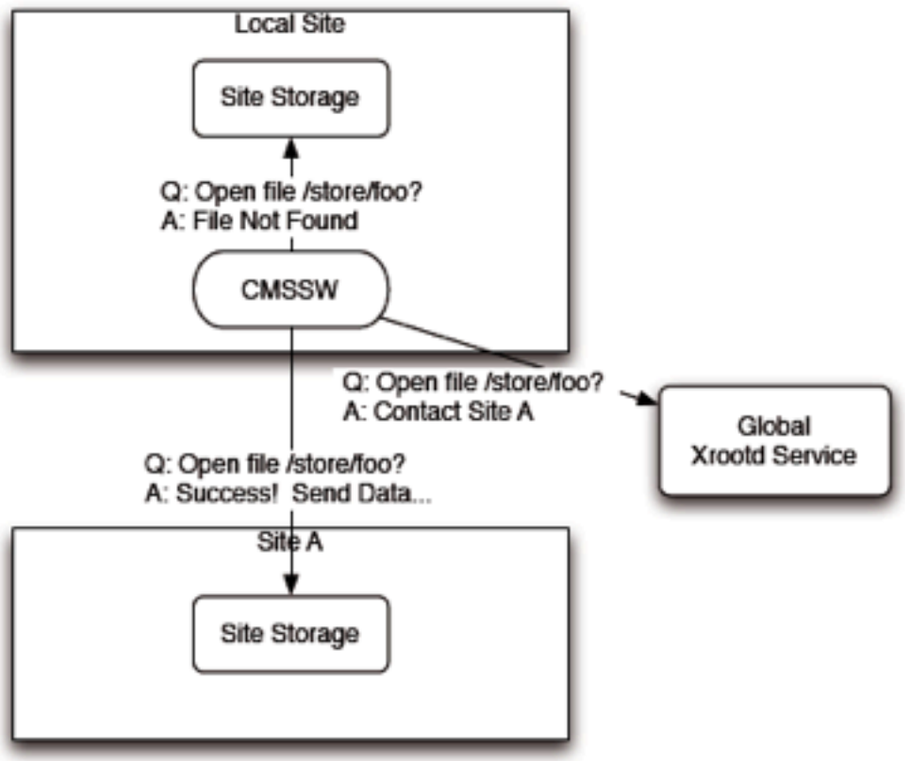
- We don't have the effort to efficiently maintain CMS PhEDEx at Omaha.
- This T3 only reads from the global xrootd system. Good continuous test.
- 6,000 wall hours in the last day.



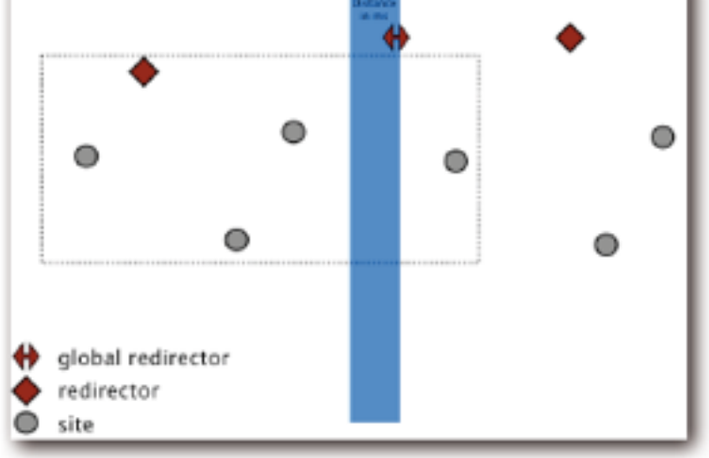
Xrootd traffic to Omaha

Fallback Case

- CMSSW_3_9_x includes ability to open a file remotely if the local file is missing.



Cost



Global xrootd federation

- Thoughts on a dynamic test infrastructure
  - More redirectors, >= 6 sites, thresholds to join
  - Different xrootd implementations, and storage solutions
  - Subset of resources could work on different test scopes

- The capabilities of a geographically partitioned wide area access system are attractive
  - Start with Tier-3s and interactive use plus the fall back channel
  - Maybe add access to CAF systems through an EOS pool as available
  - Gain some experiment
  - Use the popularity service to better predict what datasets will only be used once.

Milestones

Early March

- Minor code updates by xrootd team
- Central services available for interested parties to join
- Base-level monitoring available
- Deliverable: a plan for YOU to join (thresholds, actions needed, base-level documentation)

Early May

- JR/LoadTest equivalent infrastructures available
- Improved documentation/monitoring (feedback collected since March)
- Define metrics from next milestone
- Deliverable: fallback use-case

Early June

- Organized job overflow to sites which are part of the integration instance
- Measure according to the metrics defined in previous milestone
- Deliverable: a report

August / September

- Touchbase with sites after few months of constant usage: is the service stable?
- Deliverable: interactive use-case
- Deliverable: disk-less T3 use-case

- The Popularity framework will be the system responsible for:
  - providing usage statistics on datasets/blocks on the grid.
    - information in terms of dataset name, remote site, local site, and user...
  - providing data service for further applications
    - e.g. a dynamic replica reduction agent

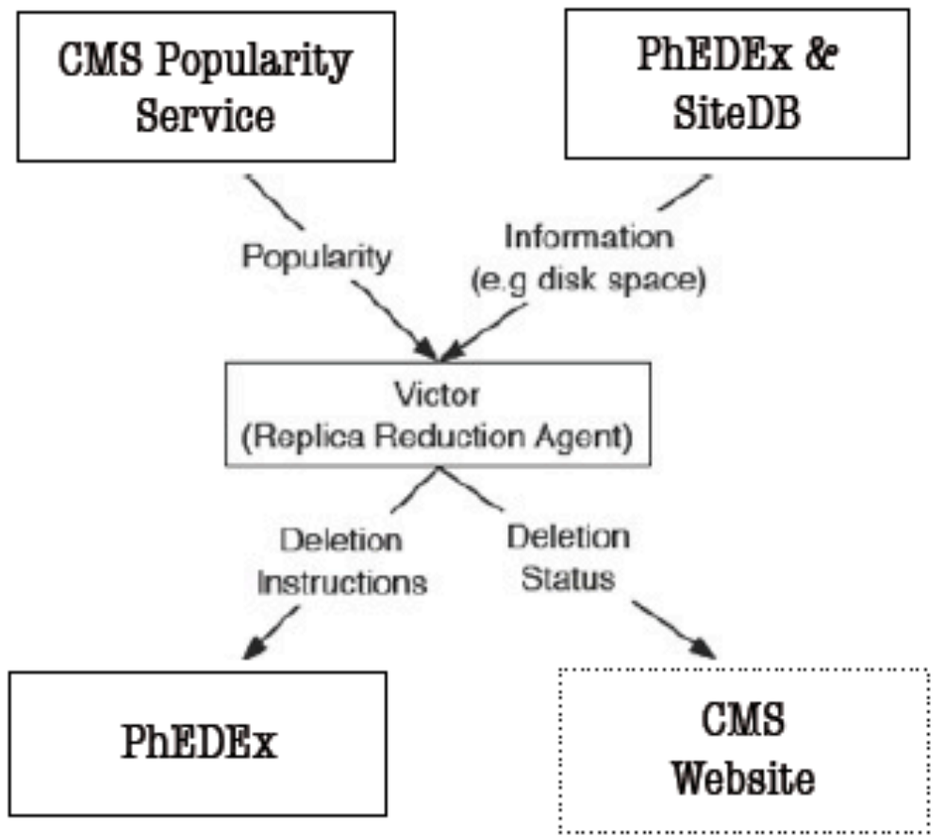
Data Popularity

- Take the already existing dashboard aggregator and summary table at dataset granularity
  - validate the current summary table
- Extend the aggregator at block granularity:
  - enable CRAB sending block info to the dashboard collector
  - Pull and factorize the information from the dashboard and aggregate it

“Victor” per CMS

On CMS Replica Reduction Agent

Reuse the ATLAS replica reduction model for the CMS central space clean-up



- Re-factorize the existing Victor code
  - Implementing a ATLAS and CMS plug-in
- Integrate CMS Victor to
  - Block Popularity Service
  - PhEDEx/SiteDB for Accounting
  - PhEDEx Request for Replica Reduction

EOS per CMS

Discussion : Disk-only (EOS) solution

- Constraints of EOS solution, compared to DEFAULT
  - Disk-only space management needs to be addressed carefully. We may need a dynamic data placement tool according to popularity (à la ATLAS)
  - Need small PhEDEx development (optimize staging via xrd3cp)
    - Note : EOS could just become a new PhEDEx site
  - Might be disruptive for CERN users at the beginning, but as Ian remarked in the introduction, not changing a system because it is (more or less) working now is not a good strategy if there are better options for the future

Proposed new Architecture (I)

- Basically, EOS should look almost like any other CMS Tier-2
  - PhEDEx node + optimize stager as done for CAF (currently CMSCAF going via stager_get instead SRM)
- For data reading, need the redirector to go to either: root://eoscms//eos/cms/... or root://castorcms//castor/cern.ch/cms/... When opening root://<newRedirector>//…
  - Needs adaptation to TFC
  - Existing castorcms redirector already accomplishing something similar for CAF-T2 so expect this should be possible
- For writing, default is CASTOR, or EOS if specified
  - Needs adaptation of T0 code to be able to write to EOS
- Architecture needs to be compatible for both CERN (kerberos) and GRID (certificate) jobs. SRM interface to GRID jobs to EOS exists ("bestman" solution for ATLAS)

Proposed new Architecture (II)

- Quotas in EOS :
  - Volume quotas for users : hard and soft limit can be set, CMS needs to decide on a policy. Assumes that the mapping of GRID certificates to cern-account is working.
  - Quotas for groups (e.g. CMS PH groups) : need to understand if we could map PhEDEx-groups to EOS-group quotas ? Maybe easier to do everything within PhEDEx ?
  - Bandwidth quotas : option for the future
- Service Classes
  - Do we want to split EOS in different areas depending on reliability, replication definition ? Need to decide what is most suitable for CMS use-case.

CMS/EOS Testing Time Schedule

- February-March
  - TFC tests
  - switch CMSCAF (CAF-T1) to xrootd
  - test reading/writing from/to EOS or CASTOR via new redirector
- April-May
  - Test new PhEDEx node + special stager
  - Include Hammercloud tests
  - Test mapping of GRID certificates to CERN account and SRM interface to EOS
  - Integration of CAF-T2 into EOS
- May++
  - Adapt/Test T0 writing to EOS
  - Test Data Popularity Service on EOS

CMS

INFN activities

Xrootd Regional Redirector

Perugia Tier3

Bologna Tier3

Users Desktop

INFN-Bari

Xrootd Redirector

Xrootd door

Lustre FS

INFN-Pisa

Xrootd door

dCache SE

Xrootd door

XYZ SE

XYZ-Site

CMS

CMS performance test - 2011

- **New CMS test, new framework, and new configurations:**
  - **Preparing the new tests to be reported to HEPIX meeting in may 2011**

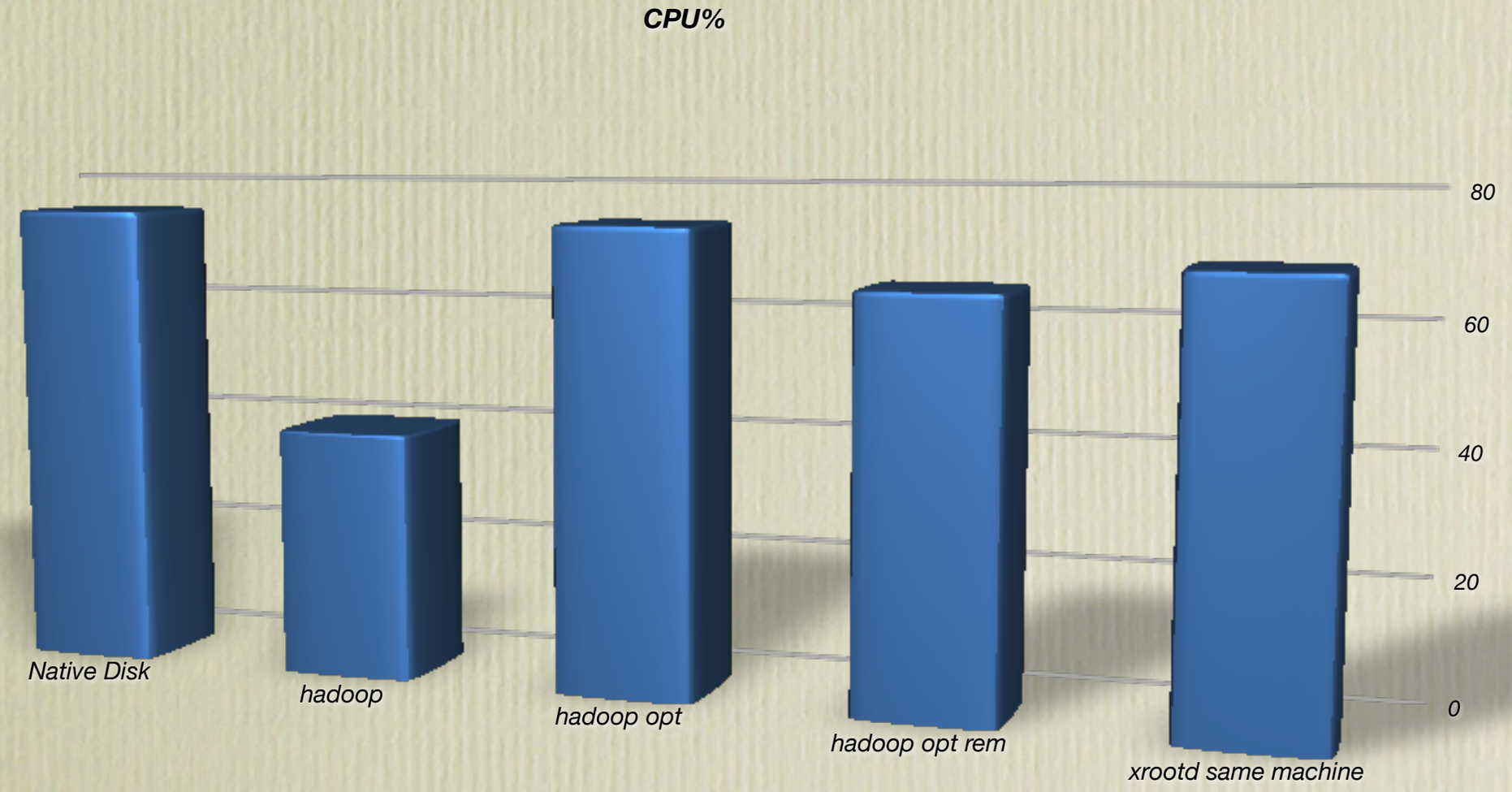
  - **Server:**
    - Lustre 2.0:
      - 3 RAID5 FS. Stripe-unit size: 128 KB. 5 Data disk each
    - Xrootd 3.0.0:
      - 13x1TB single disks. EXT3 FS
    - hadoop-0.20.2 (from http://newman.ultralight.org/)
      - 13x1TB single disks. EXT3 FS

- **Clients:**
  - SLC5.4 kernel 2.6.18-194.11.3
  - Fuse: fuse-libs-2.7.4-8
  - FUSE mount on the client (rdbuffer=32768)

Optimising the Single job

- "Hadoop opt"=> rdbuffer=32768
- The CMSSW (cacheHint,readHint,cacheSize) tuning parameters are always used and tested until the best result is found
- "blockdev --setra" on each drive, was tuned in order to find the best solution
- **Lustre** is not reported in the plot, but it was **83% of CPU efficiency**

- It is possible to obtain the same performance with up **to 4-5 concurrent job** per **single native disk**



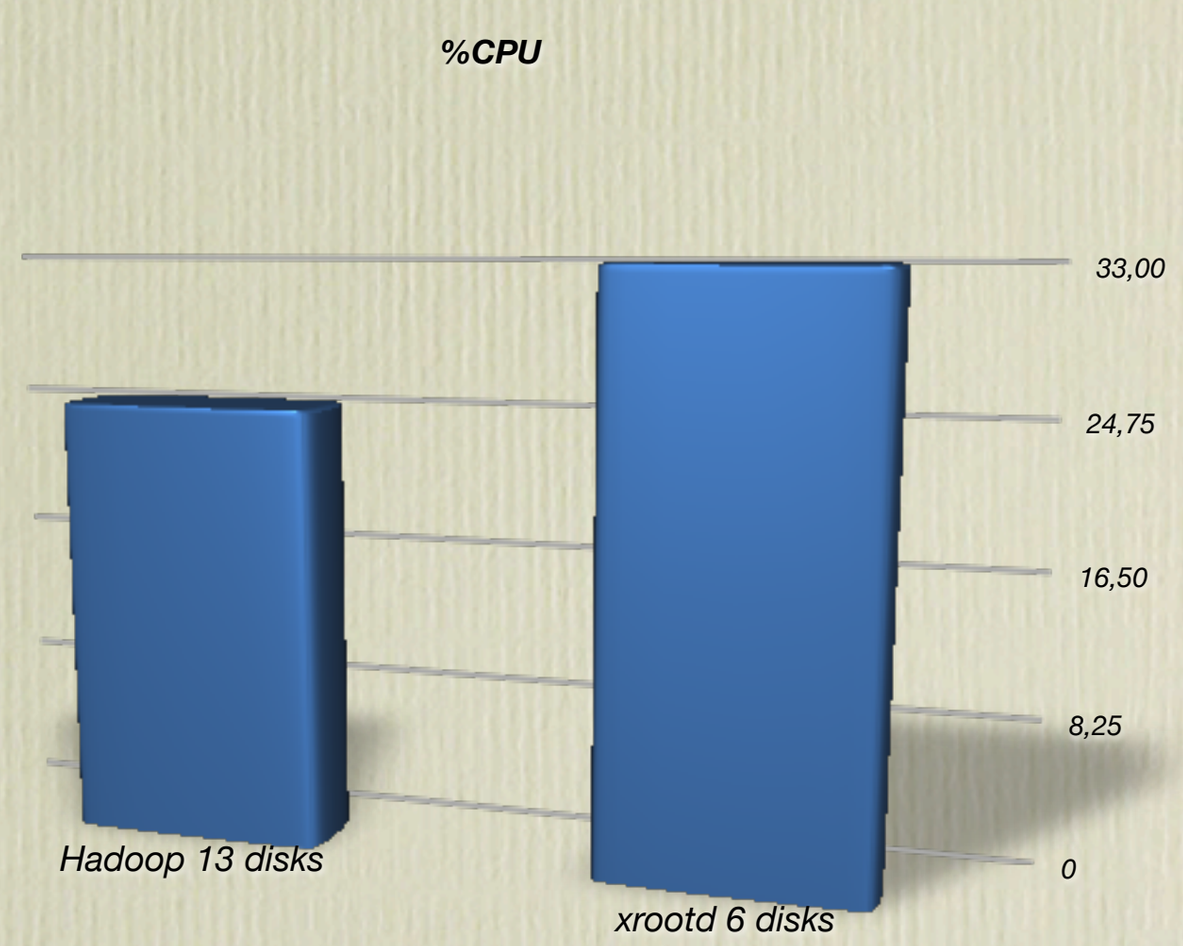
Performance Tests

- up to 116 concurrent jobs
- production farm used to run the jobs
- Each file on the server is used only by a single job
  - There is no “concurrency” on each file
- A single disk server:
  - 10Gbit/s network card
  - deep network testing to assure there are no network bottleneck
  - >400MB/s measured disk-to-network bandwidth

Performance test: hadoop vs xrootd

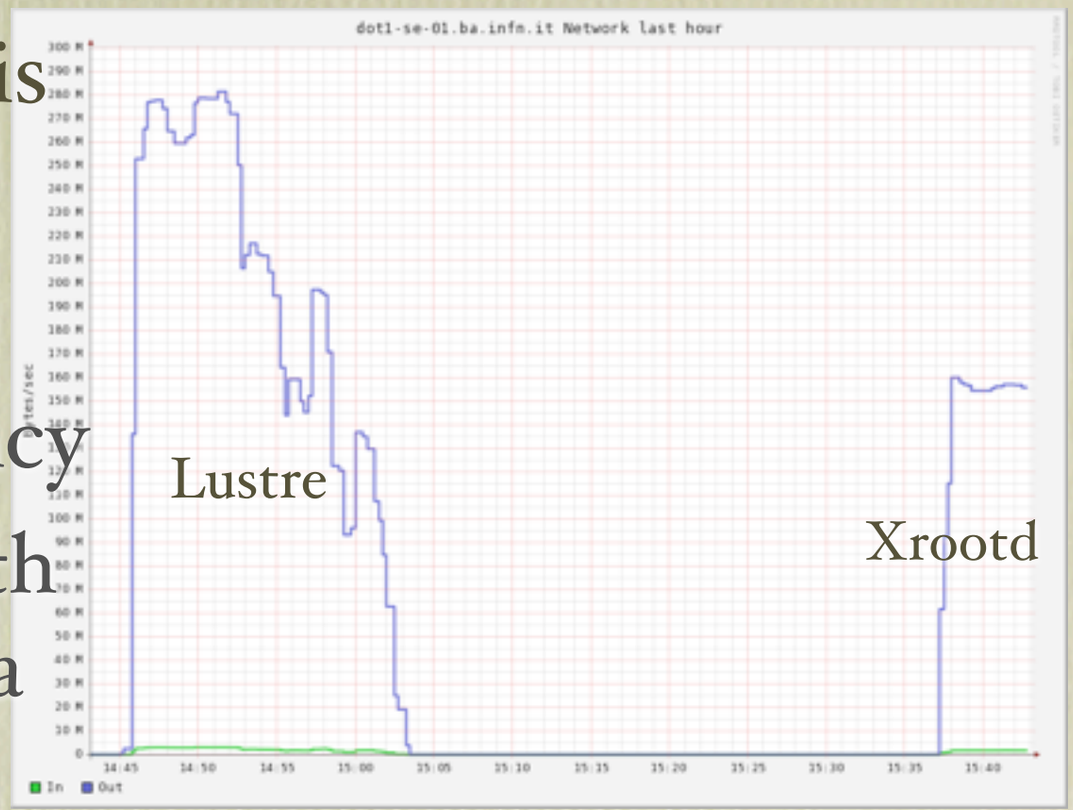
- Running 56 concurrent cms analysis jobs
- Using **6 disks** for xrootd
- Using **13 disks** on hadoop installation
  - Reading data using “fuse optimized“
  - Single server: no “block replica”

- We have observed huge load on the server while running “hadoop test”
  - tuning “blockdev setra” did not improve the situation
  - increasing the memory for java produced only small improvements

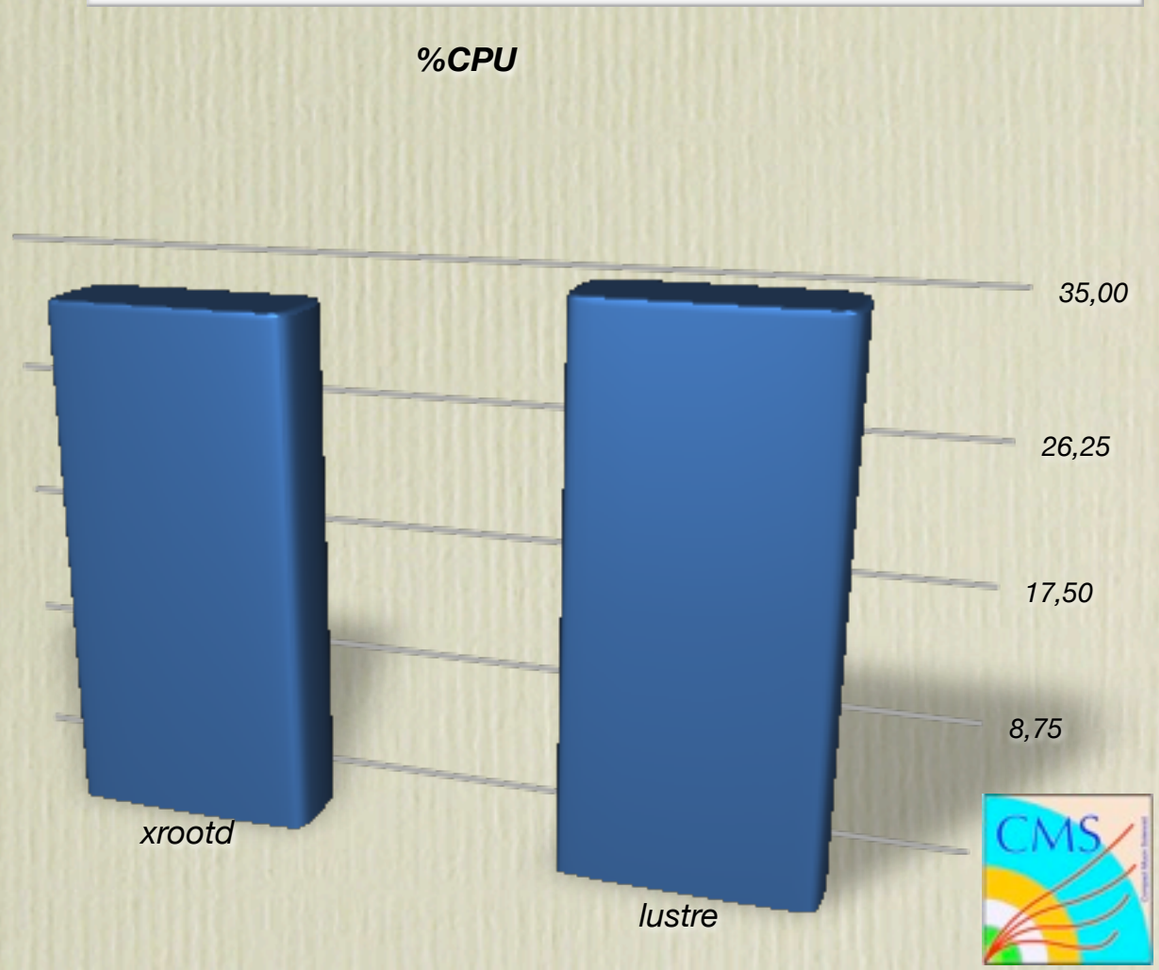


Performance tests: lustre vs xrootd

- Running 116 concurrent cms analysis jobs
- Reading ~1TB of data
- Always measuring the CPU efficiency
  - This is an interesting parameter both from user's point of view and from a site admin



- The network usage of the two solution is completely different
- Different configuration were tested: it looks like this is the best result we can achieve
- In both cases the disk subsystem on the server is the bottleneck



Future Works

- Try to run the same tests with new CMSSW based on ROOT 5.28
  - We have to produce new files to do this test

- Testing also other use cases (different kind of analysis)

- **Run tests on dCache using NFS4.1 (LNL will be actively involved)**

- To start measuring the performance in case of "remote" access (both xrootd and NFS4.1)
  - With different RTT? 10ms, 50ms,100ms

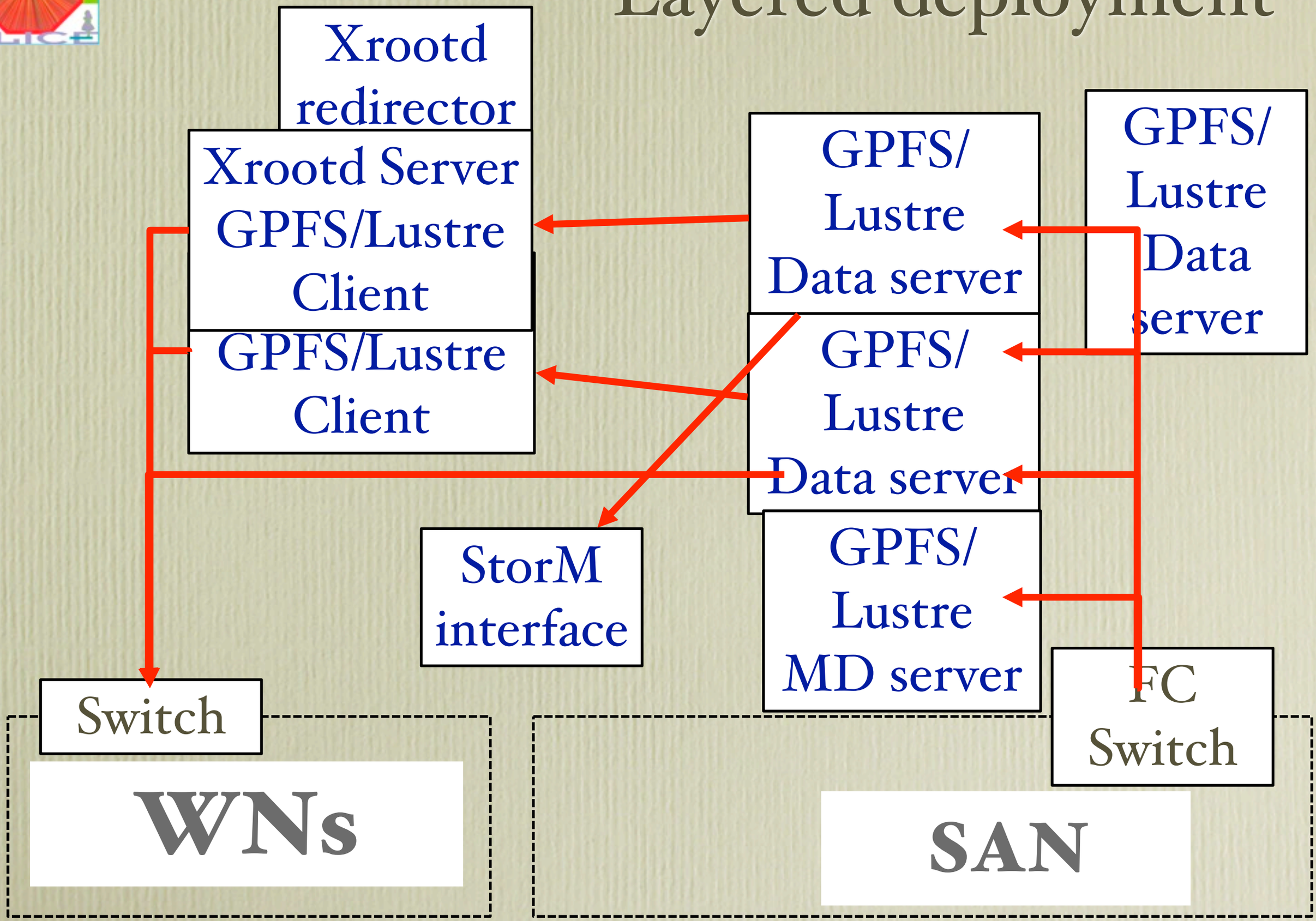
ALICE

INFN activities

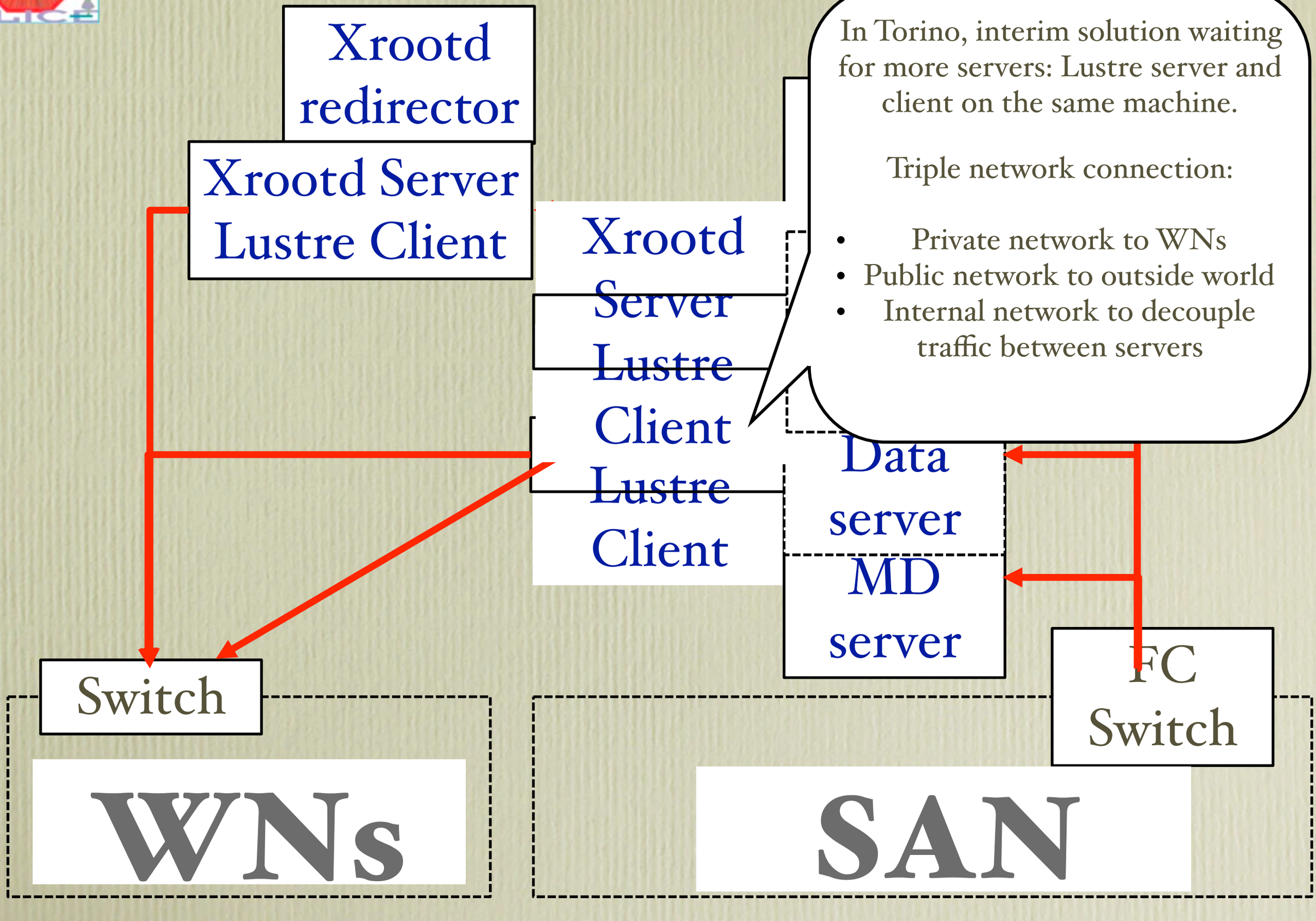
Layered deployment

- Xrootd provides only rudimentary management tools
  - e.g. filesystem migration, metadata backup,...
  - Also, documentation is scarce
- Several sites need to cater for different requirements
  - Multi-VO T2s, consolidated storage for different applications,...
  - Or just existing infrastructure to build upon (e.g. Trieste)
- Xrootd is highly efficient, very stable and simple to configure
  - But only in the "standard" use cases
  - Not very flexible. It is intended to be used "as is"
- Solution: **xrootd on top of a parallel filesystem**
  - Both GPFS (CNAF, CT, TS) and Lustre (TO, BA) in use

Layered deployment

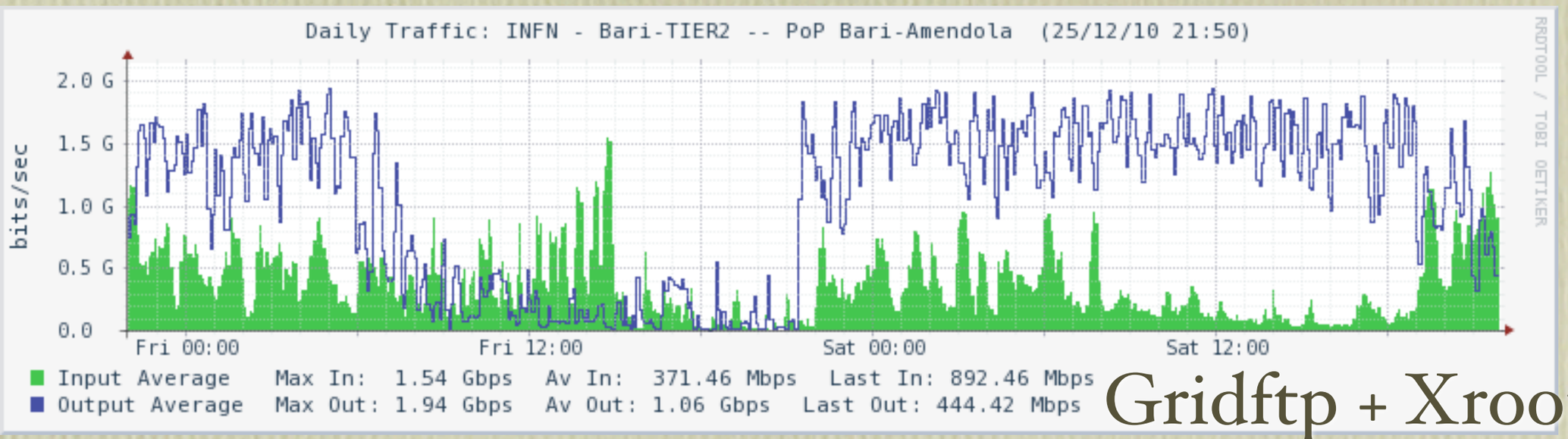


Layered deployment

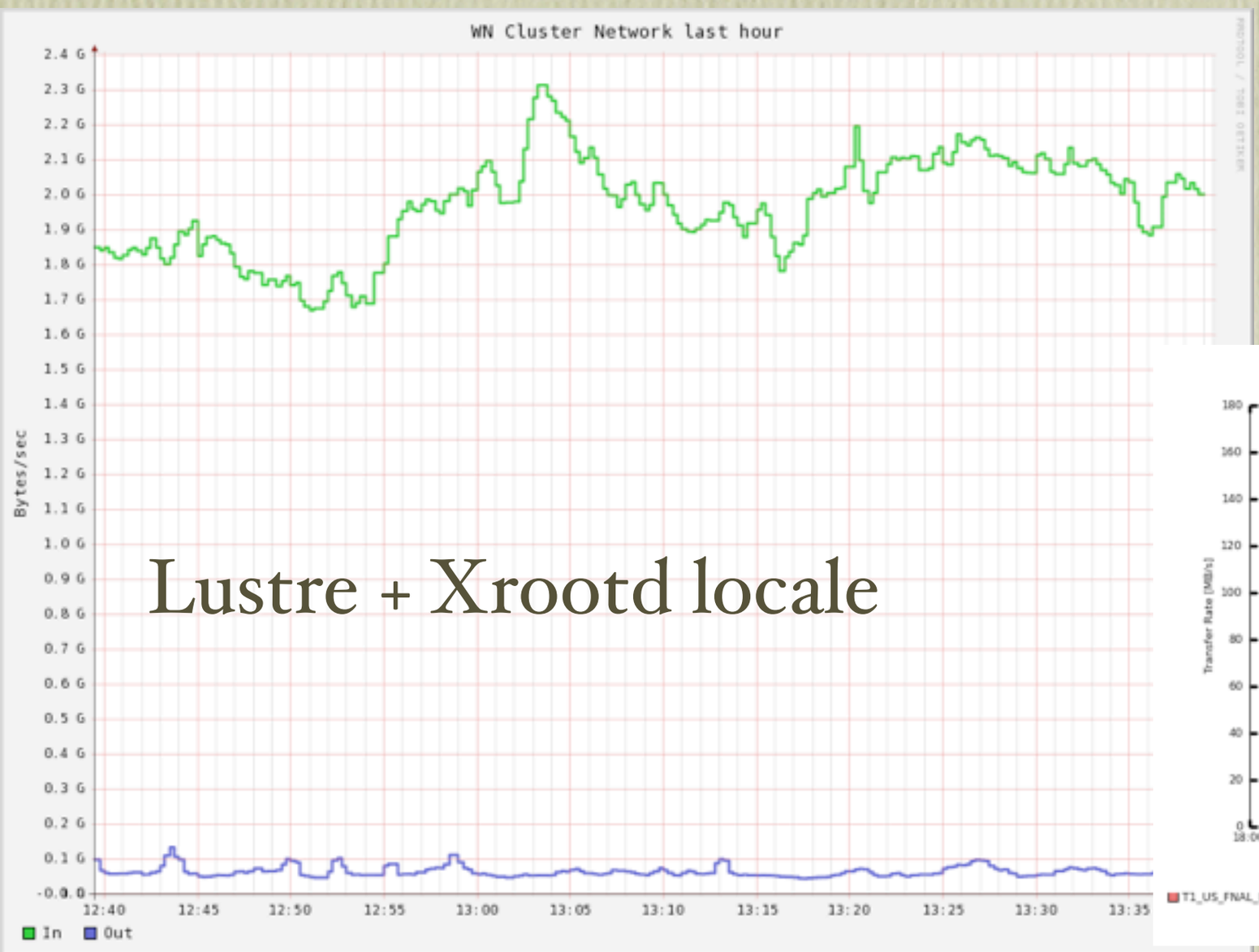


Operational experience

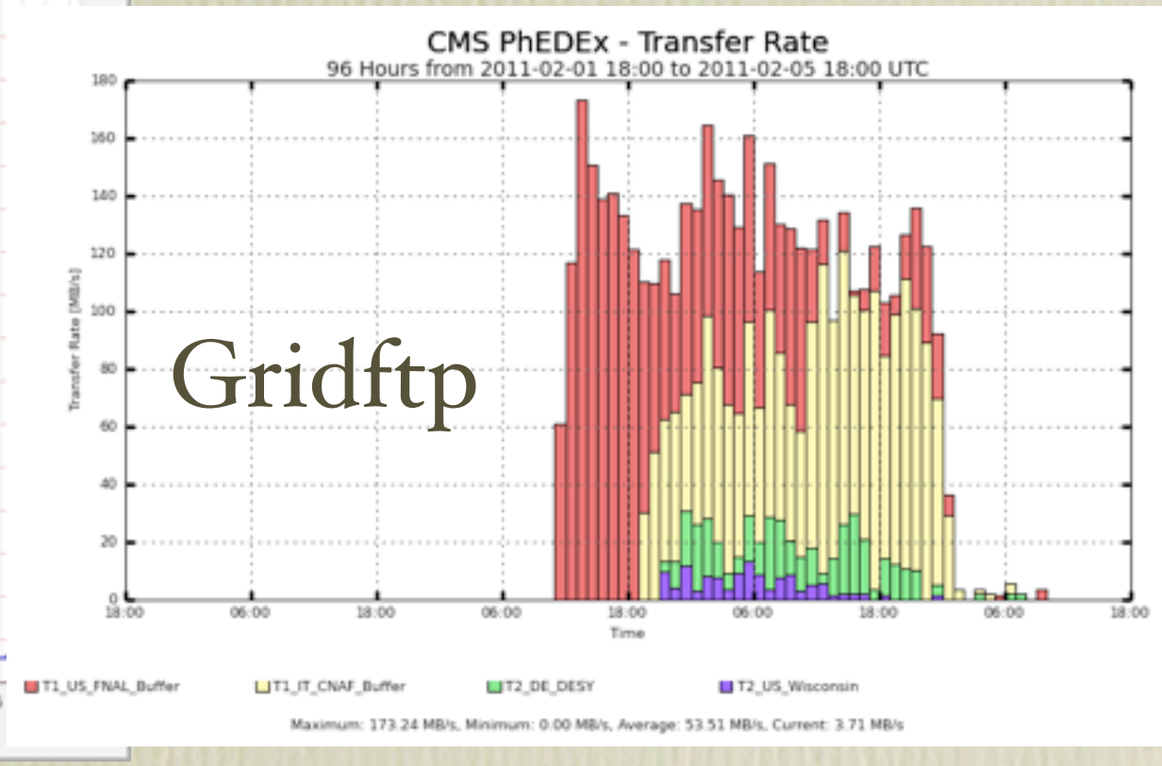
- Xrootd itself needs little maintenance
  - See e.g. “native xrootd” deployment al LNL
  - Also, getting better with age
  - Releases are backward-compatible (no mandatory updates)
  - Upgrade is usually painless
- Interaction with underlying FS can be messy
  - Cross-optimisation (e.g. readahead conflicts)
  - No support for redundant servers
    - Space double-counting
    - Xrd3cp failures
- At CNAF, e.g.:
  - All servers are independent redirectors behind a DNS alias



Gridftp + Xrootd WAN



Lustre + Xrootd locale



Gridftp

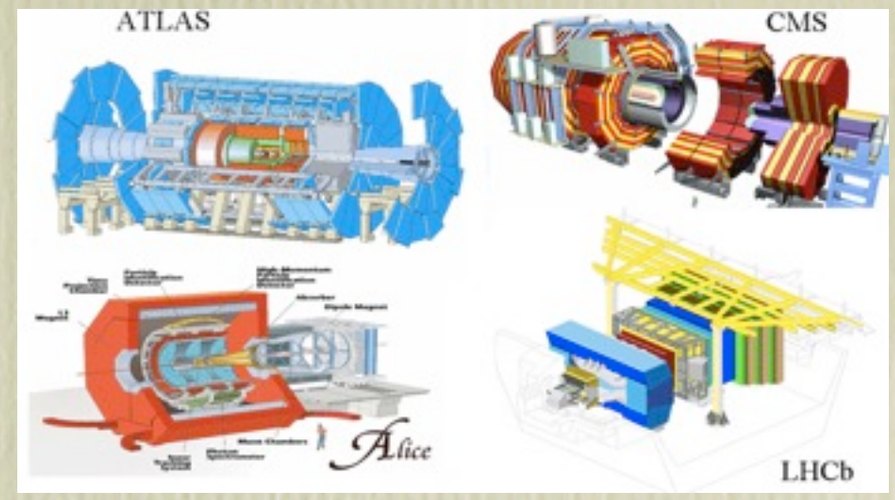
Various & sundry

- Very conservative data deletion policy in ALICE
  - Underlying FS tools allow to create different storage areas for high-performance and "cheap" storage
  - Just something we're thinking about in Torino
- Xrootd "Virtual Mass Storage" feature
  - Allows to create caches e.g. for interactive facilities
  - Under study in Torino, but already in use in several proof-based Analysis Facilities
- We use StoRM to provide SRM access to data
  - On top of Lustre/GPFS, alongside xrootd
  - Little integration needed: xrootd writes always as same user
  - No idea of performance – not used by ALICE
- Plugin for TSM tape backend at Tier-1
  - Manages tape recalls directly from GPFS
  - By F. Noferini and v. Vagnoni

INFN Ti

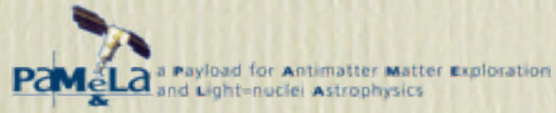
Disk storage at CNAF

- Large number of users with independent requirements
- 6.4 PB of disk on-line served by GPFS
  - 5 DDN S2A 9950
    - SATA disks of 2 TB for data
    - SAS disks of 300 GB for metadata
  - 11 EMC² 3-80
  - 1 EMC² 4-960
- GPFS disk servers 10 Gbit/s
  - 26 = 8(Atlas)+6(Alice)+12(CMS)
- GPFS disk servers 1 Gbit/s ~ 60





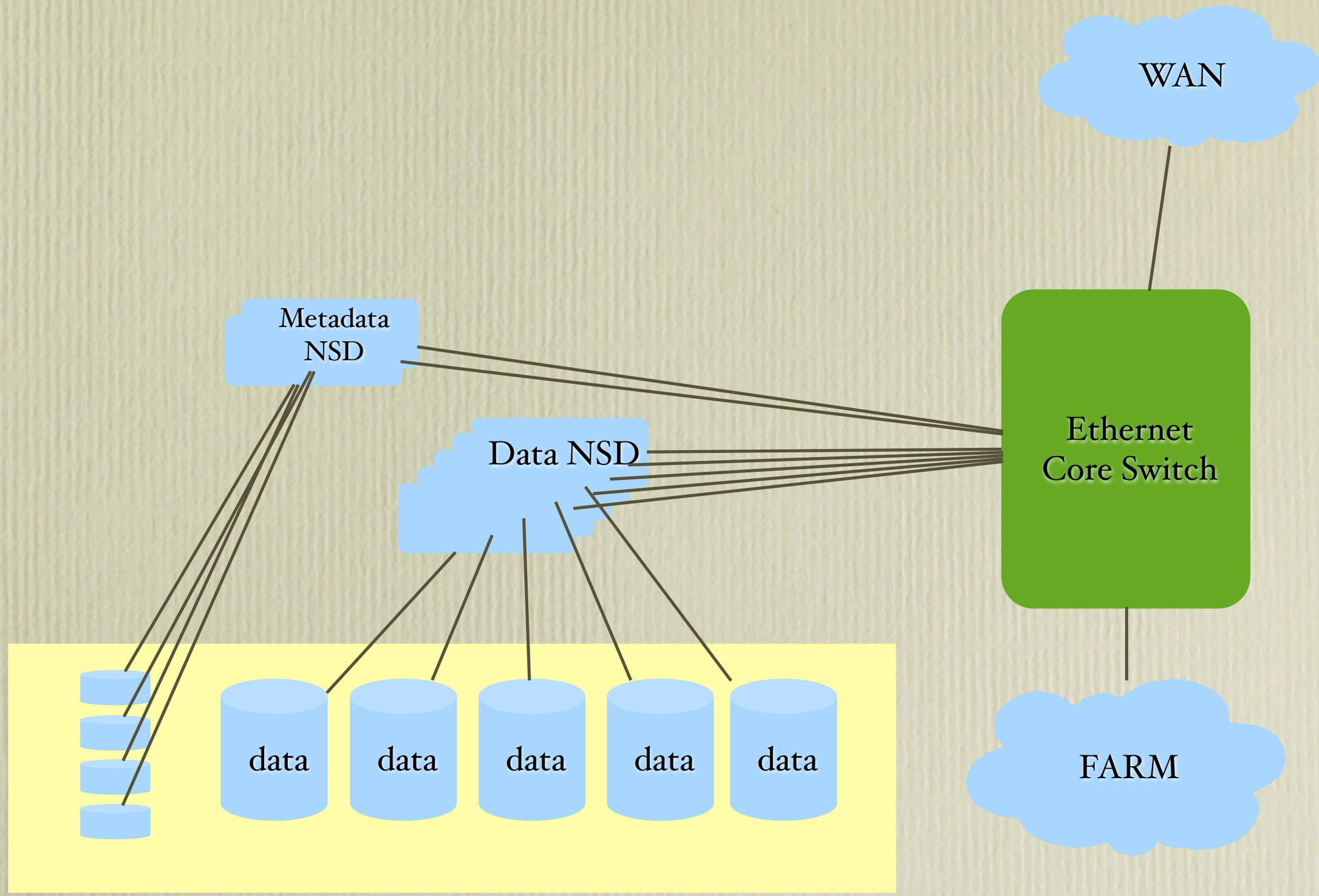


GPFS: multi-cluster environment

- Version
  - 3.2.1-23 and 3.4.0-3 (+efix)
- Multi Cluster environment
  - 1 cluster for WN (real) diskless
  - 1 cluster for VWN (virtual) diskless
  - 6 clusters for the larger experiments (Atlas, Alice, BaBar/SuperB, CMS, CDF, LHCb)
  - 2 clusters for the other experiments (Argo, AMS, Virgo, ...)
  - 2 CNFS clusters (software area, home directories)

A typical GPFS cluster



Mass Storage Sysytem

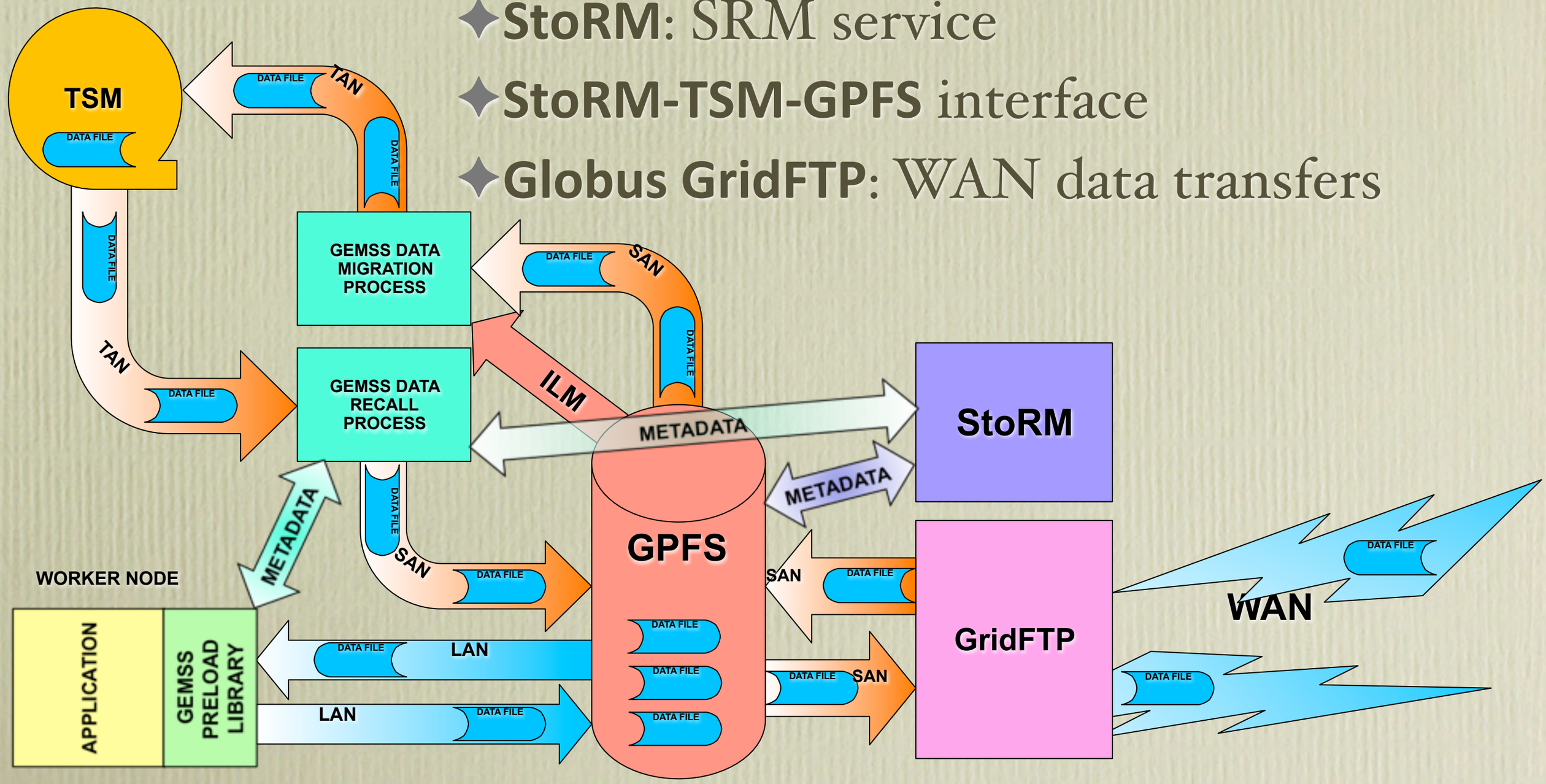
- CASTOR phased out on 15.02.2011
- GEMSS is in use by all LHC and non-LHC experiments

Experiment	# tapes used	Tape space (TB)
LHCB	165	126.9
CMS	2702	2357.9
ATLAS	555	476.2
ALICE	208	174.5
ARGO	407	381.6
AMS	53	47.7
AGATA	80	74.8
MAGIC	32	27.1
PAMELA	75	68.9
VIRGO	64 (ongoing)	55.6 (+105)

Building blocks of GEMSS system

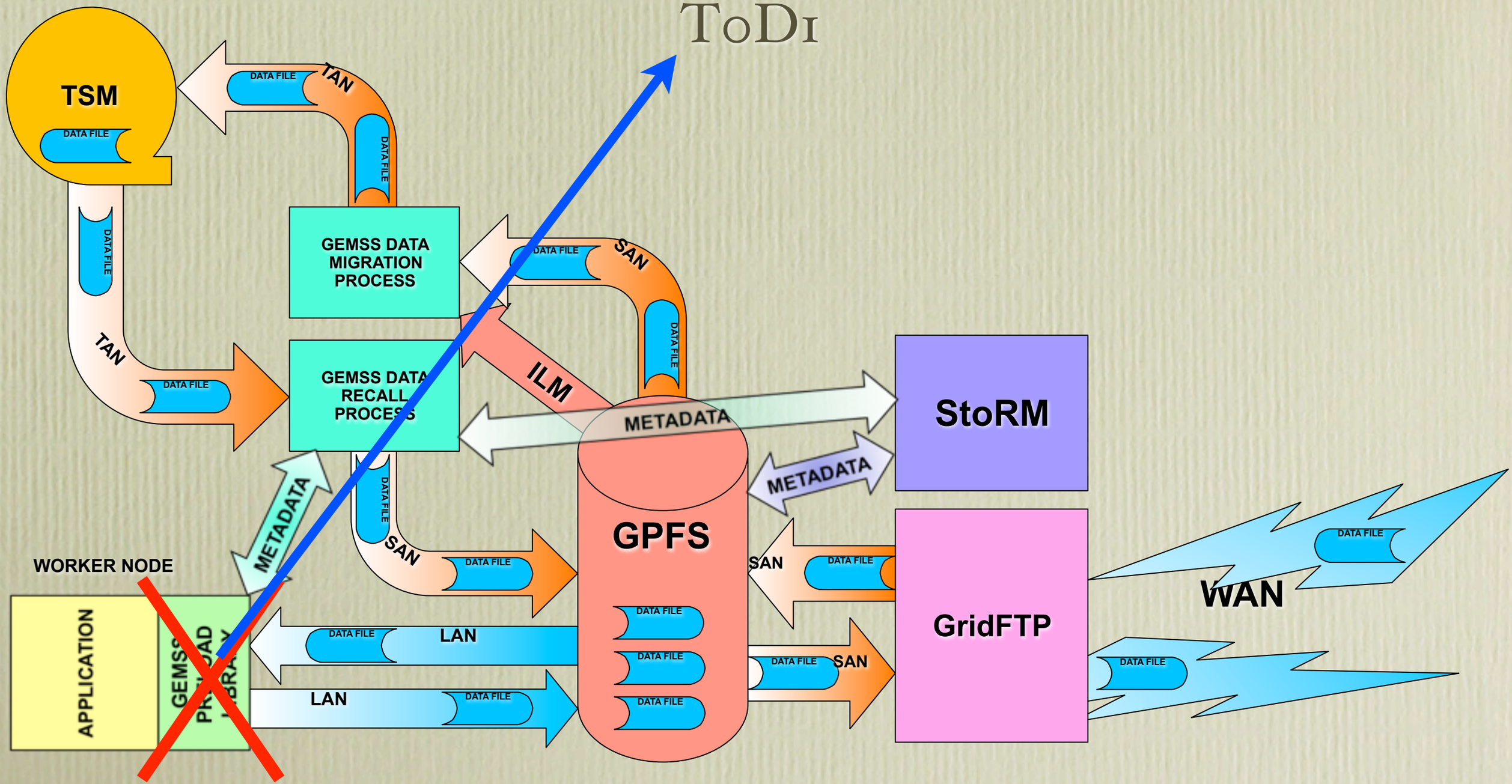
Disk-centric system with five building blocks

1. **GPFS**: disk-storage software infrastructure
- **TSM**: tape management system
- **StoRM**: SRM service
- **StoRM-TSM-GPFS** interface
- **Globus GridFTP**: WAN data transfers

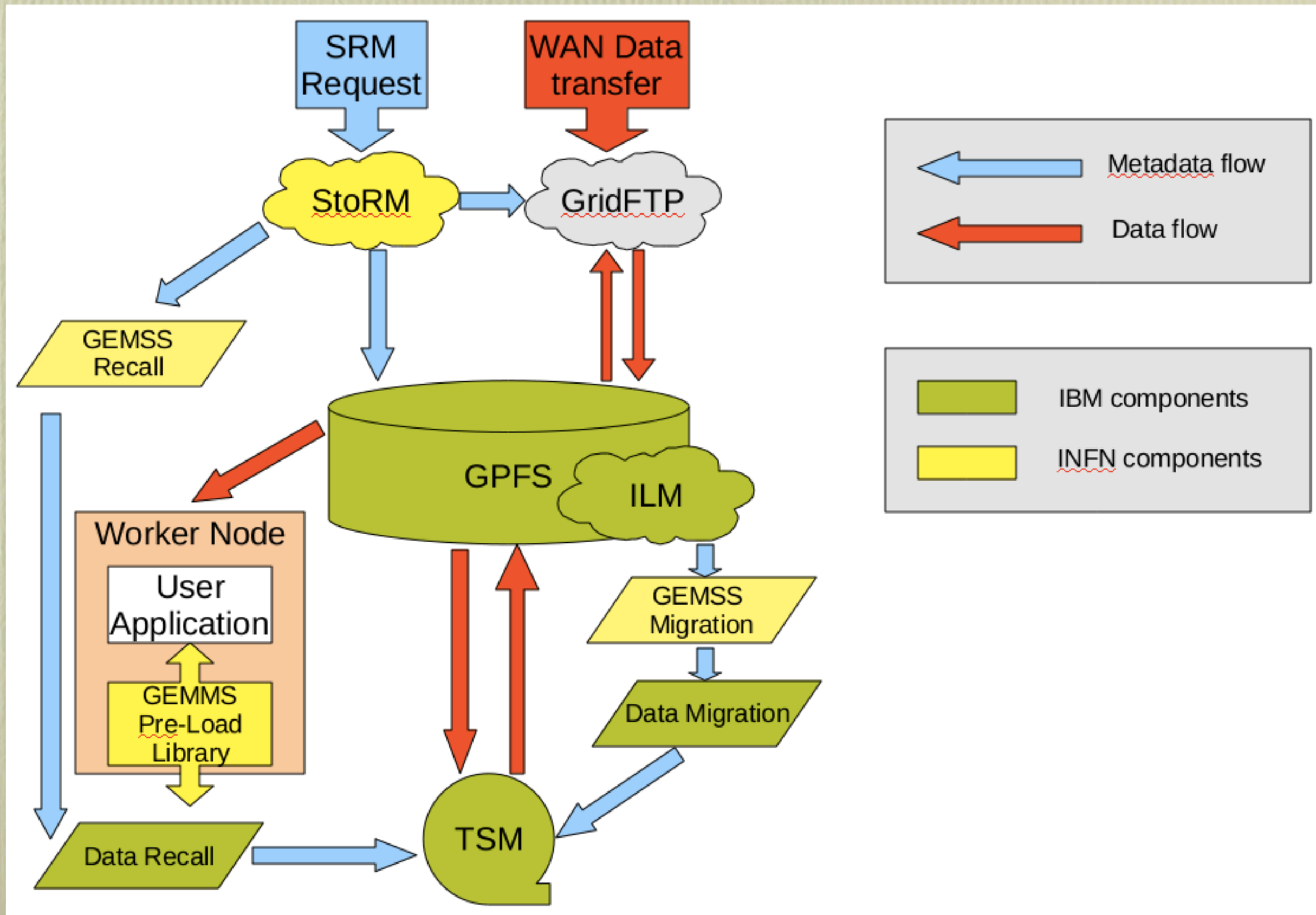


Building blocks of GEMSS system

In the last version the preload library is not needed: it is a purely posix system for both TiDo and ToDi

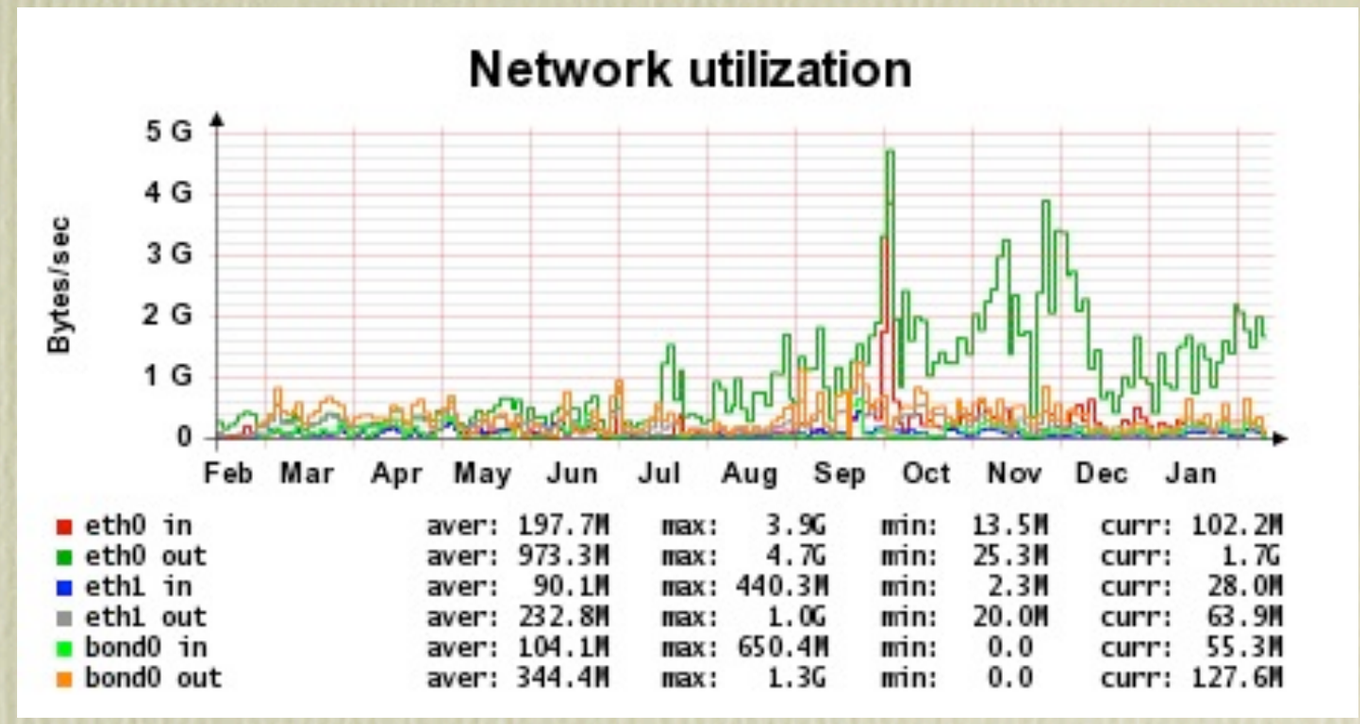


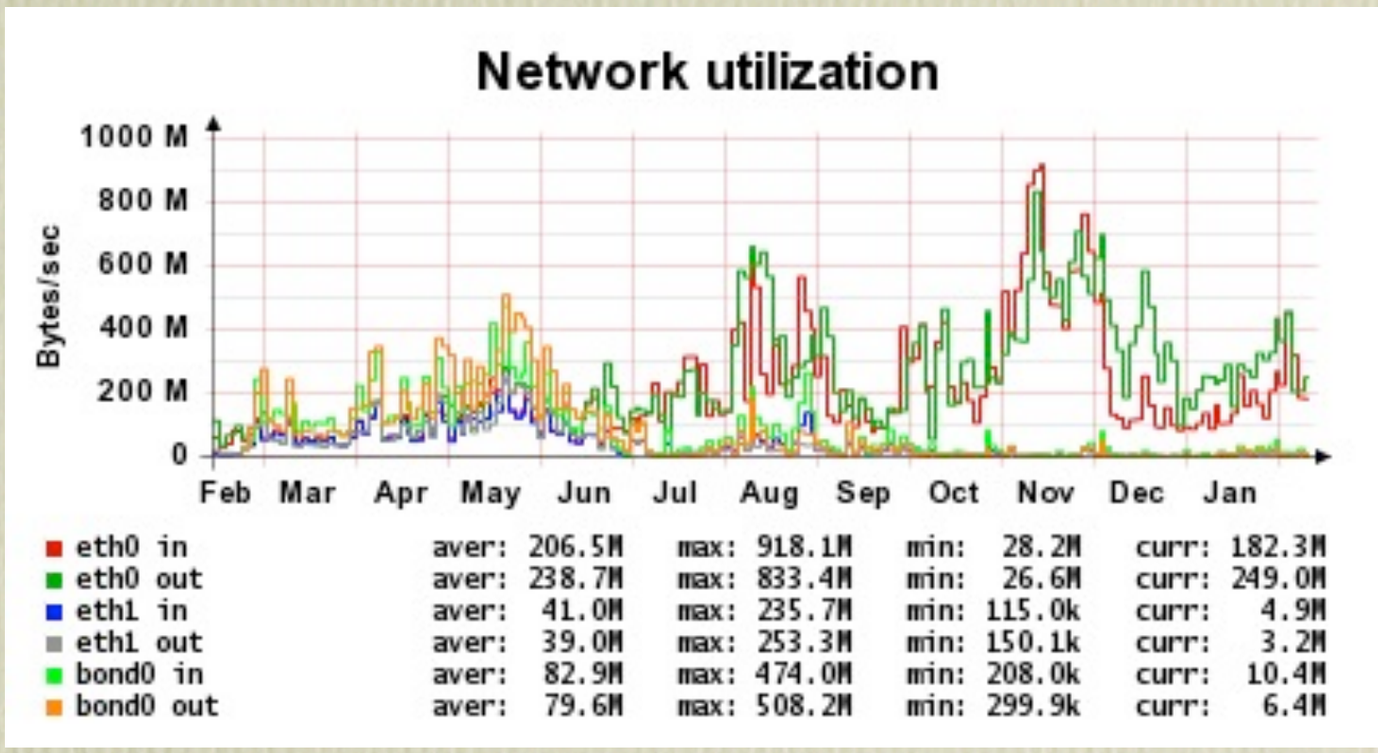
GEMSS layout



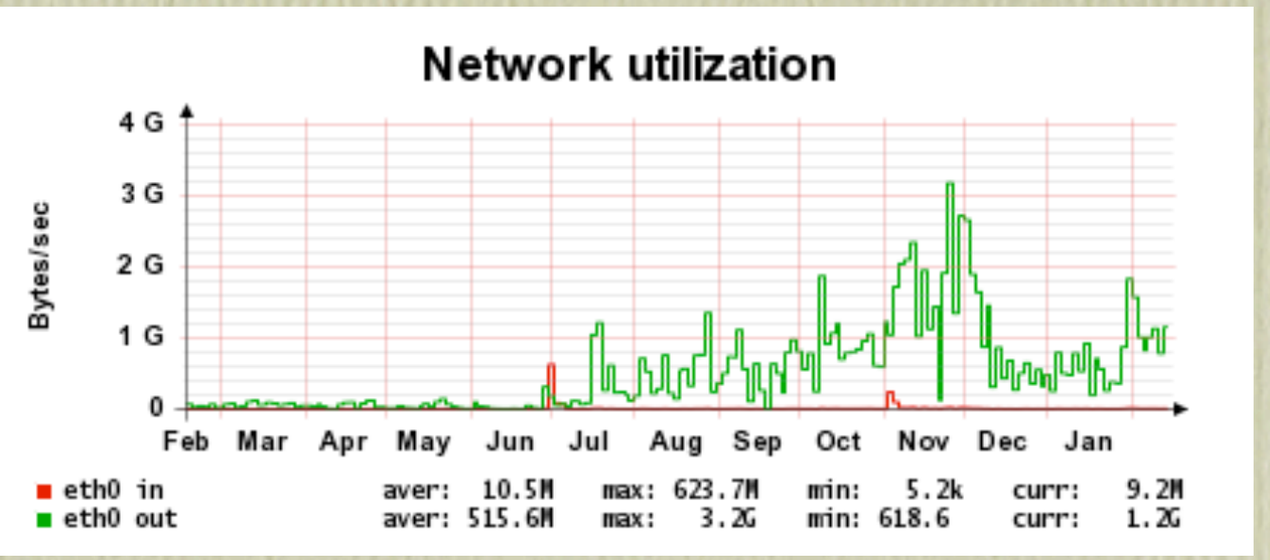
Last year's statistics

- Native GPFS (only LAN)
  - Several GB/s sustained from disk servers to worker nodes
- GridFTP (mostly to/from WAN)
  - Up to 1 GB/s in reads and writes

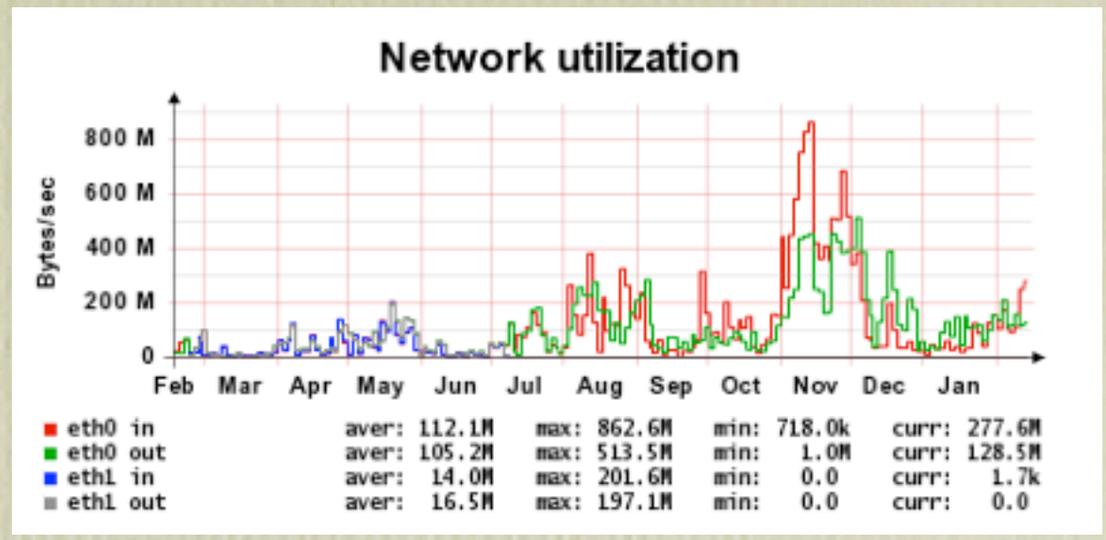




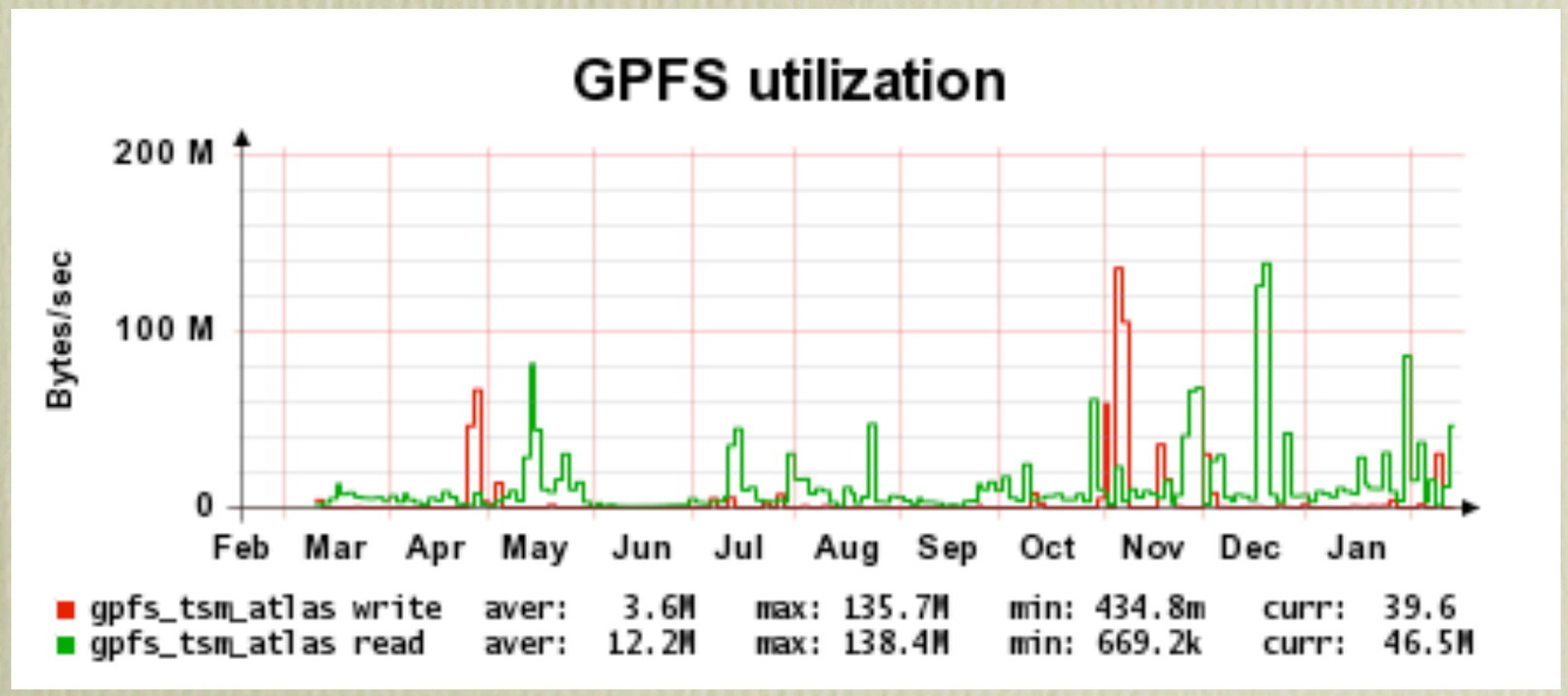
ATLAS GPFS



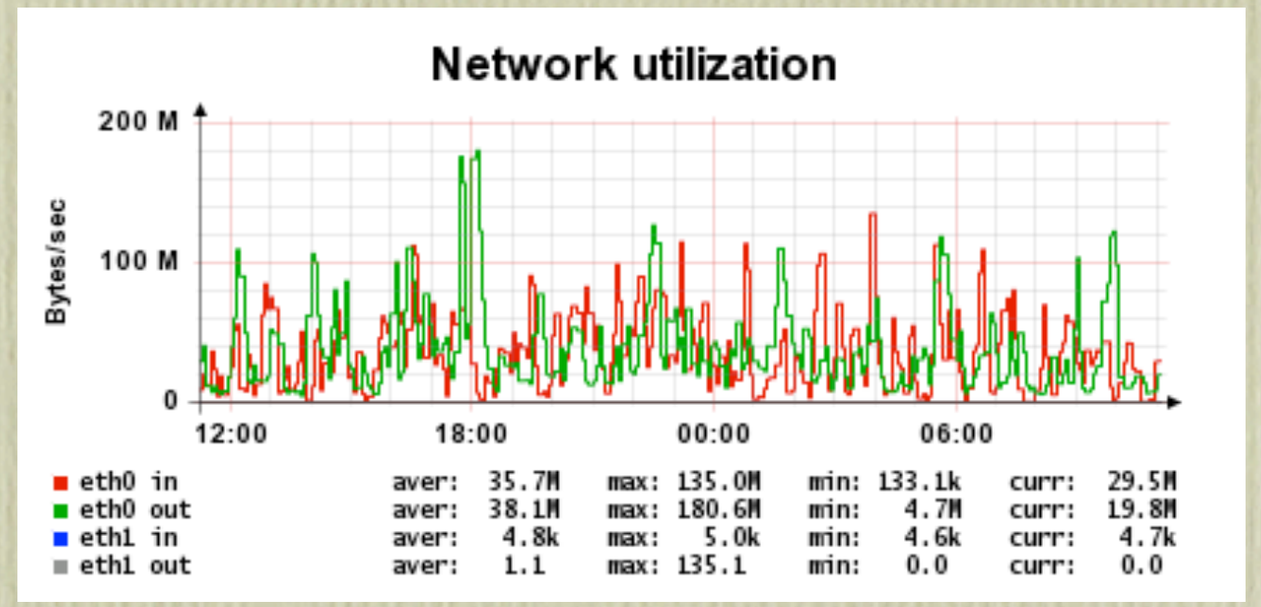
ATLAS GRIDFTP



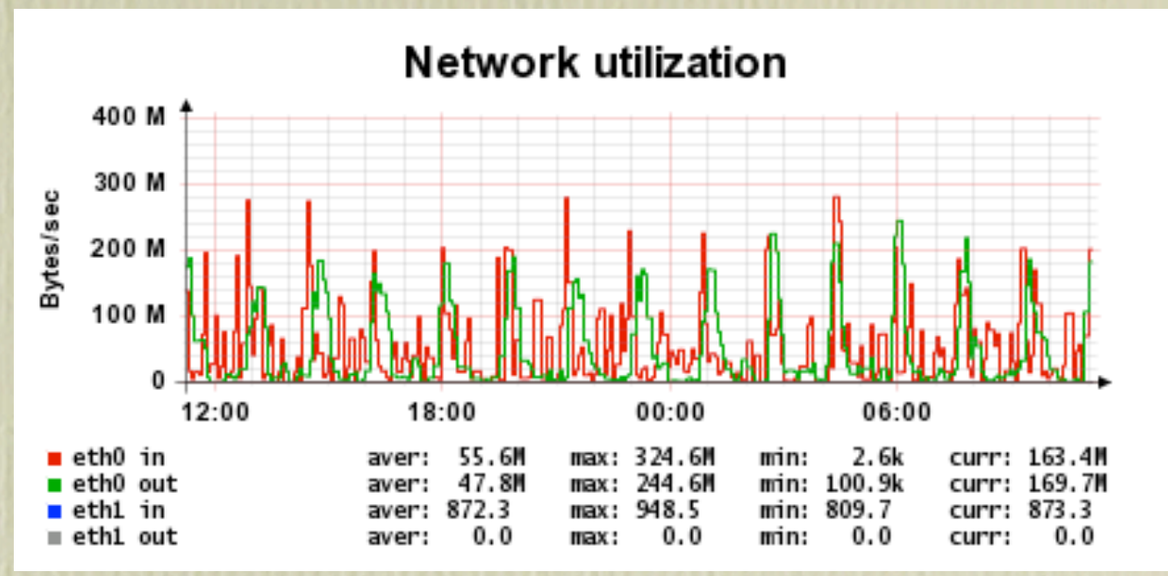
ATLAS TAPE TRAFFIC



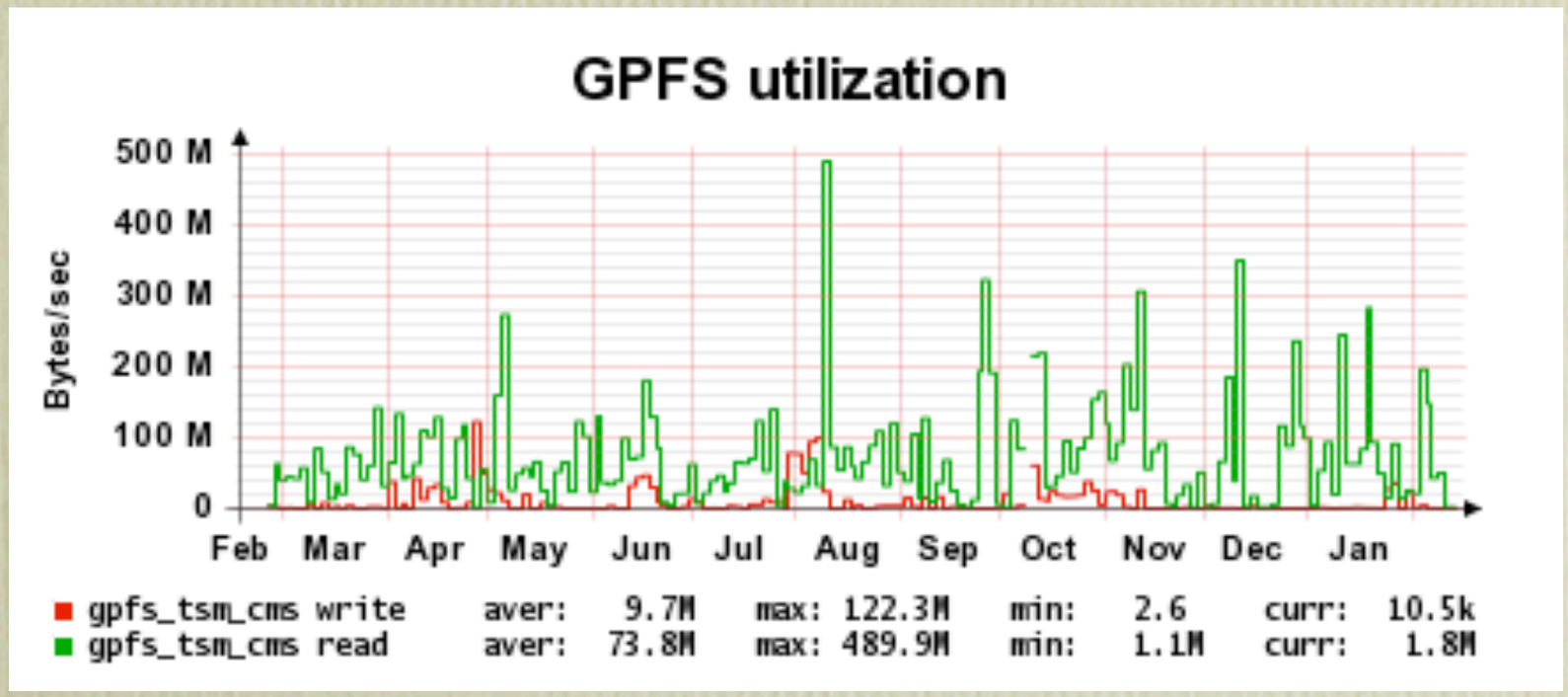
CMS GPFS



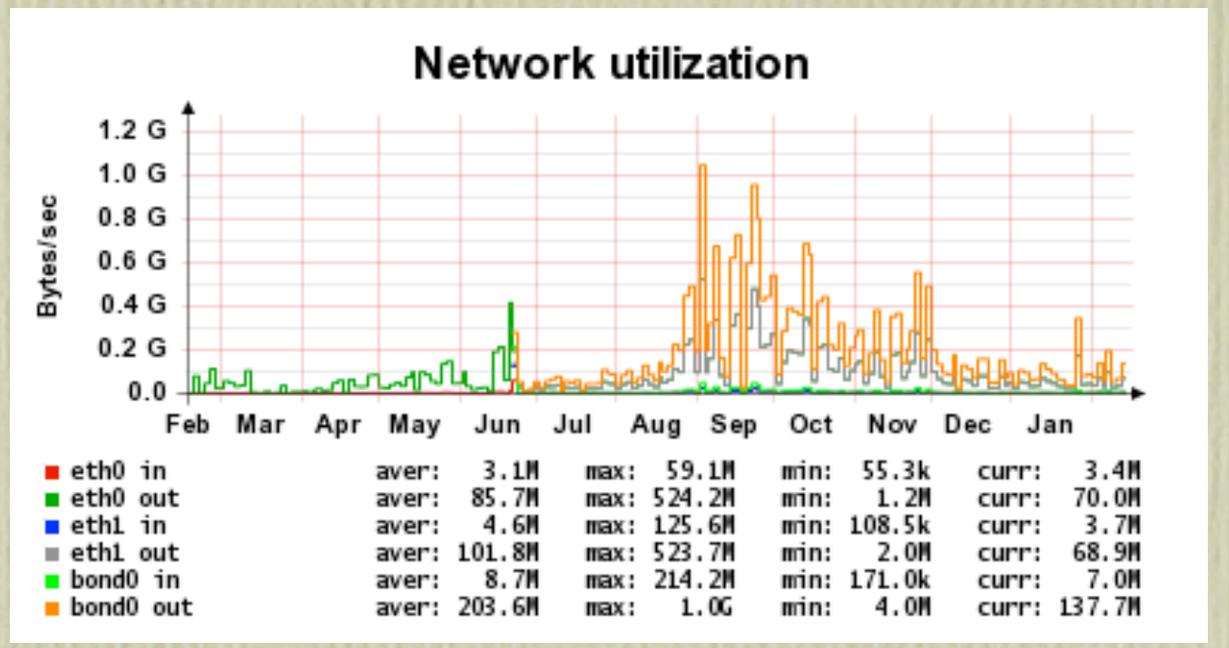
CMS GRIDFTP



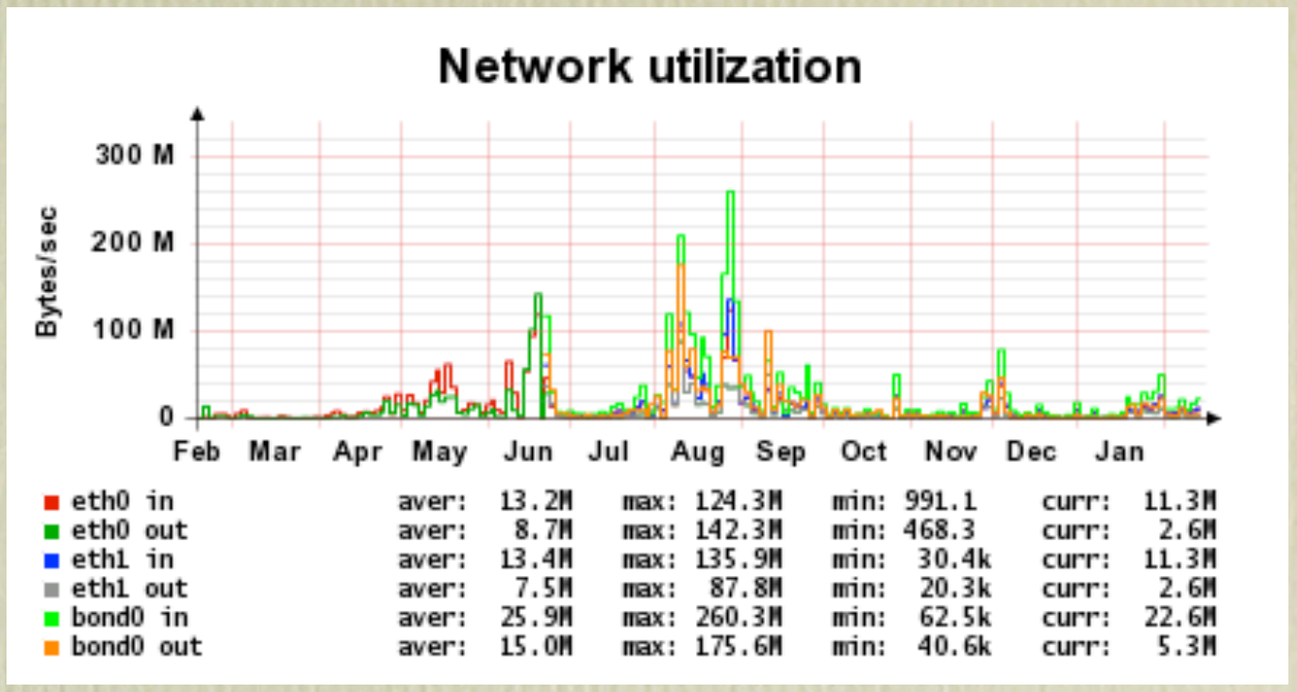
CMS TAPE TRAFFIC



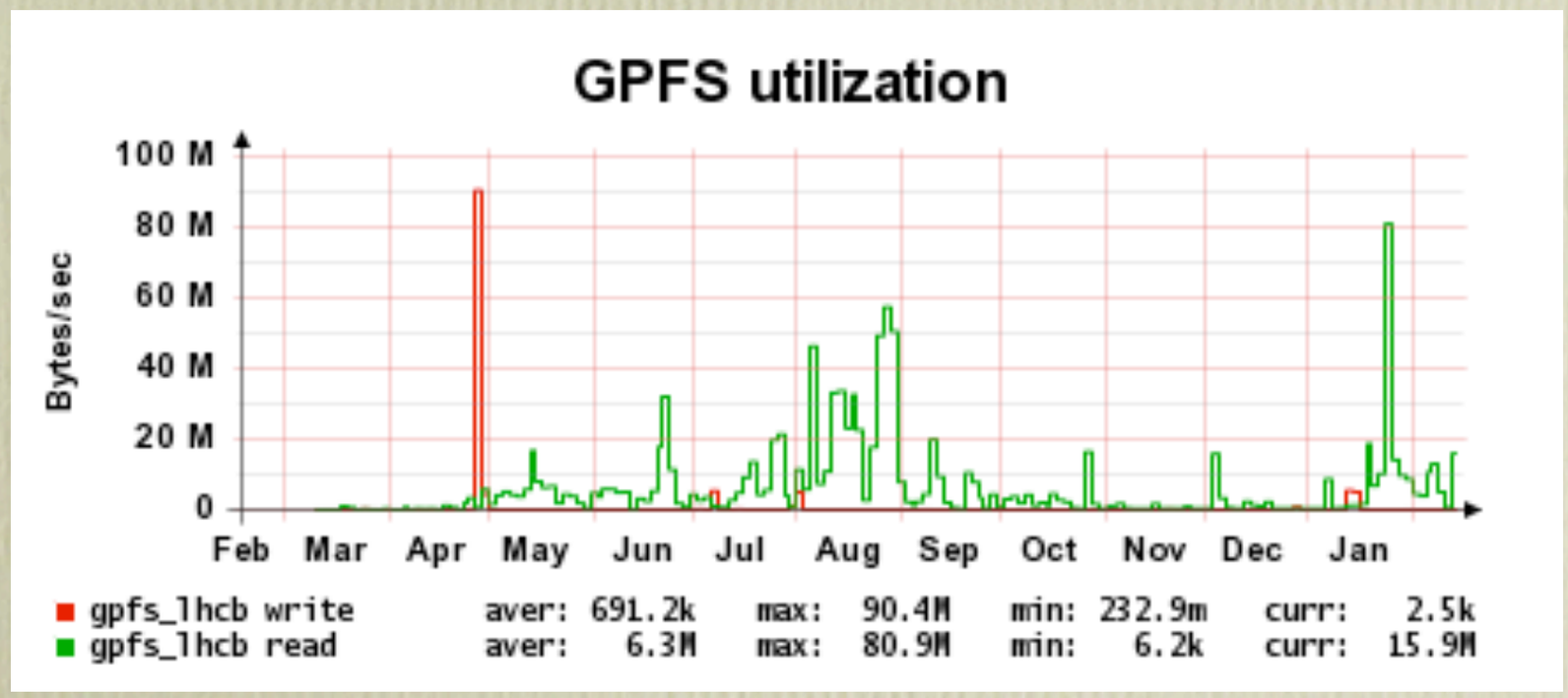
LHCb GPFS



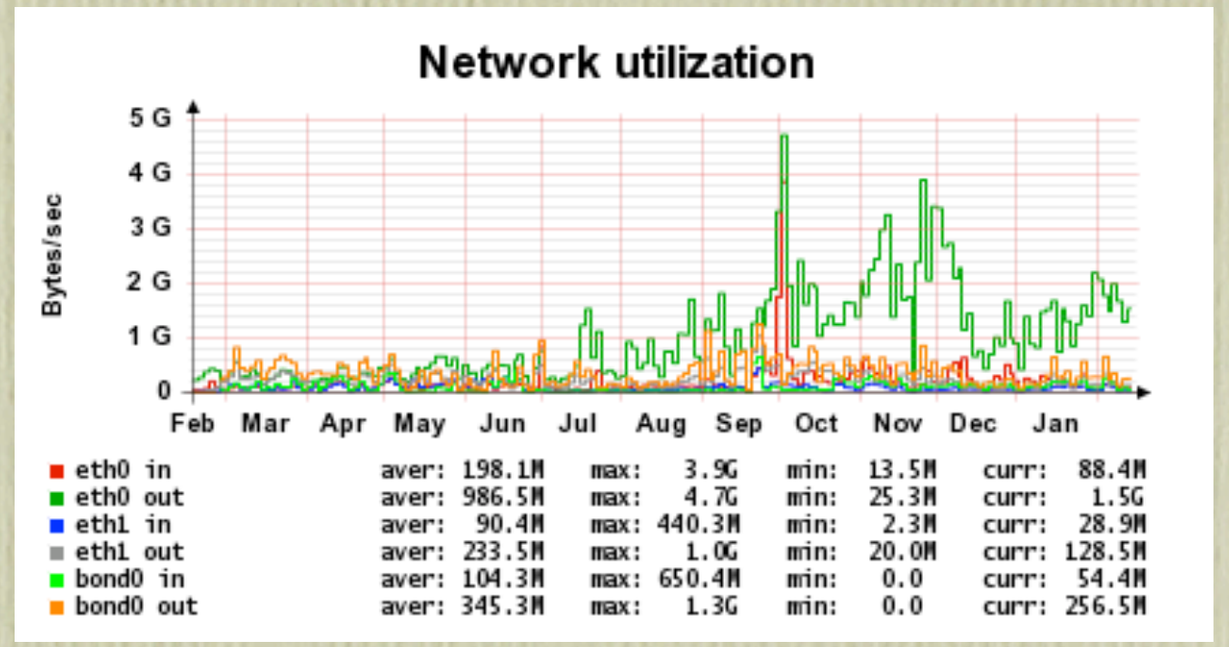
LHCb GRIDFTP



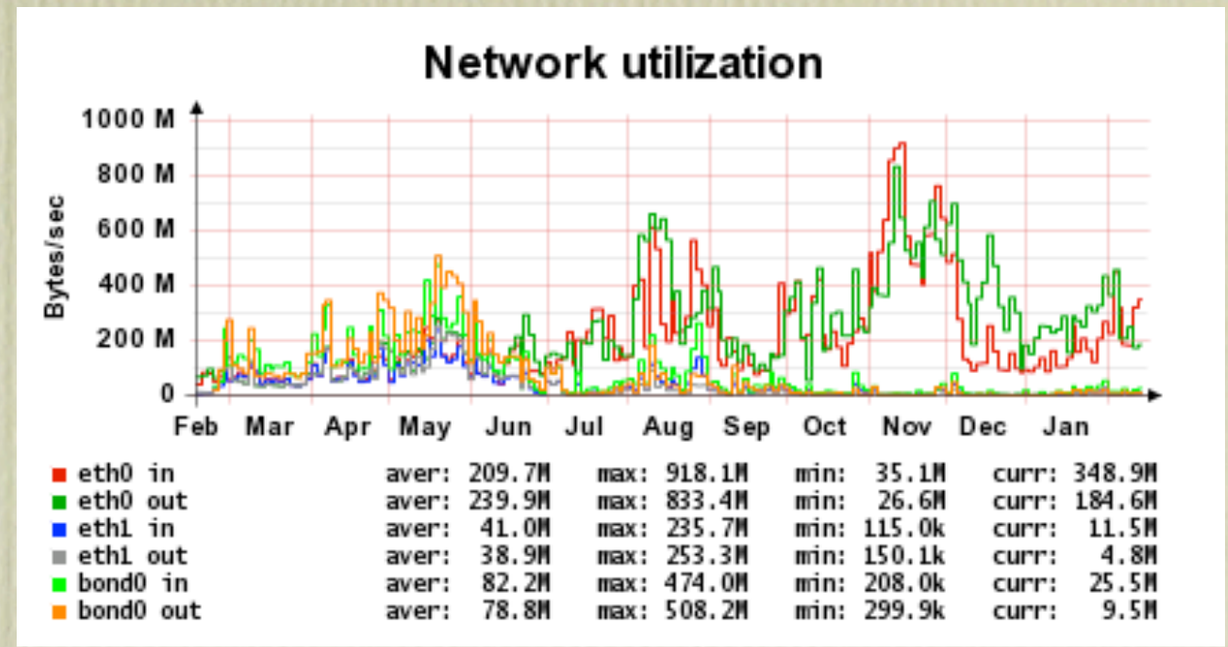
LHCb TAPE TRAFFIC



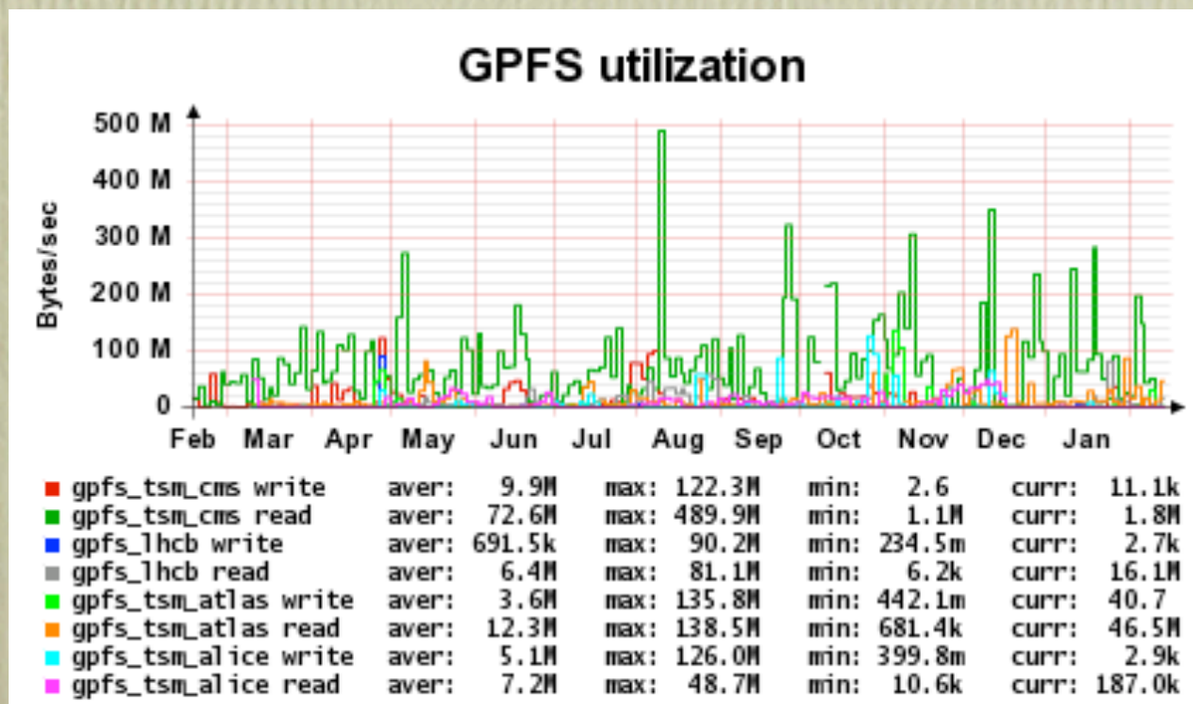
CNAF GPFS



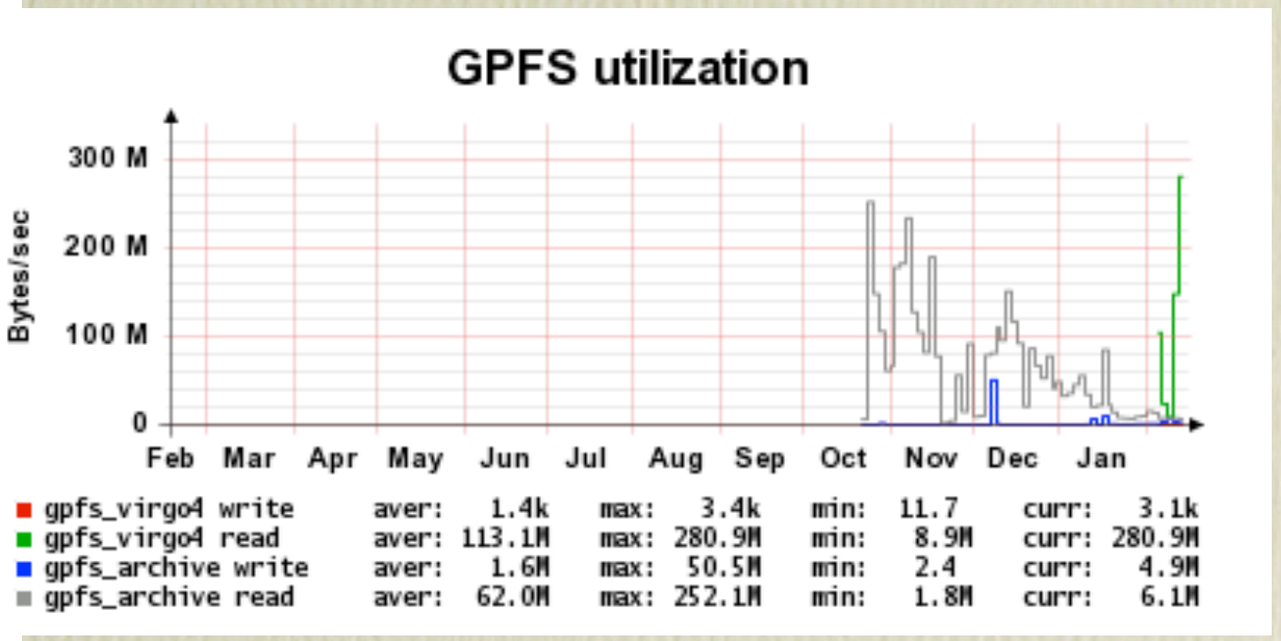
CNAF GRIDFTP



CNAF TAPE (LHC)



CNAF TAPE (no LHC)



Conclusions

- Starting from something that works now, it is needed to improve tools and strategies to be prepared to the increase in amount of data
- There is room for improvements not only at the computing infrastructure level but lot of work can be done in the application optimization (lots of things are already happening)
- Man power to keep to infrastructure running should be taken into account
- We have at least two “checkpoints” behind:
  - small improvements that could be introduced without disrupting the production infrastructure (fully in production by the end of 2011)
  - production ready after LHC shutdown (~2013-14)
- Xrootd shows a very good shape, but the support in long term should be taken into account (it is not a “standard”)
  - while it could be easily adopted as the short term solution, in the long term we should keep the road open to other solutions
- *Maybe INFN could be a bit more “active” in proposing new technologies and strategies*