Experience with GPFS and StoRM at the INFN Tier-1

Luca dell'Agnello

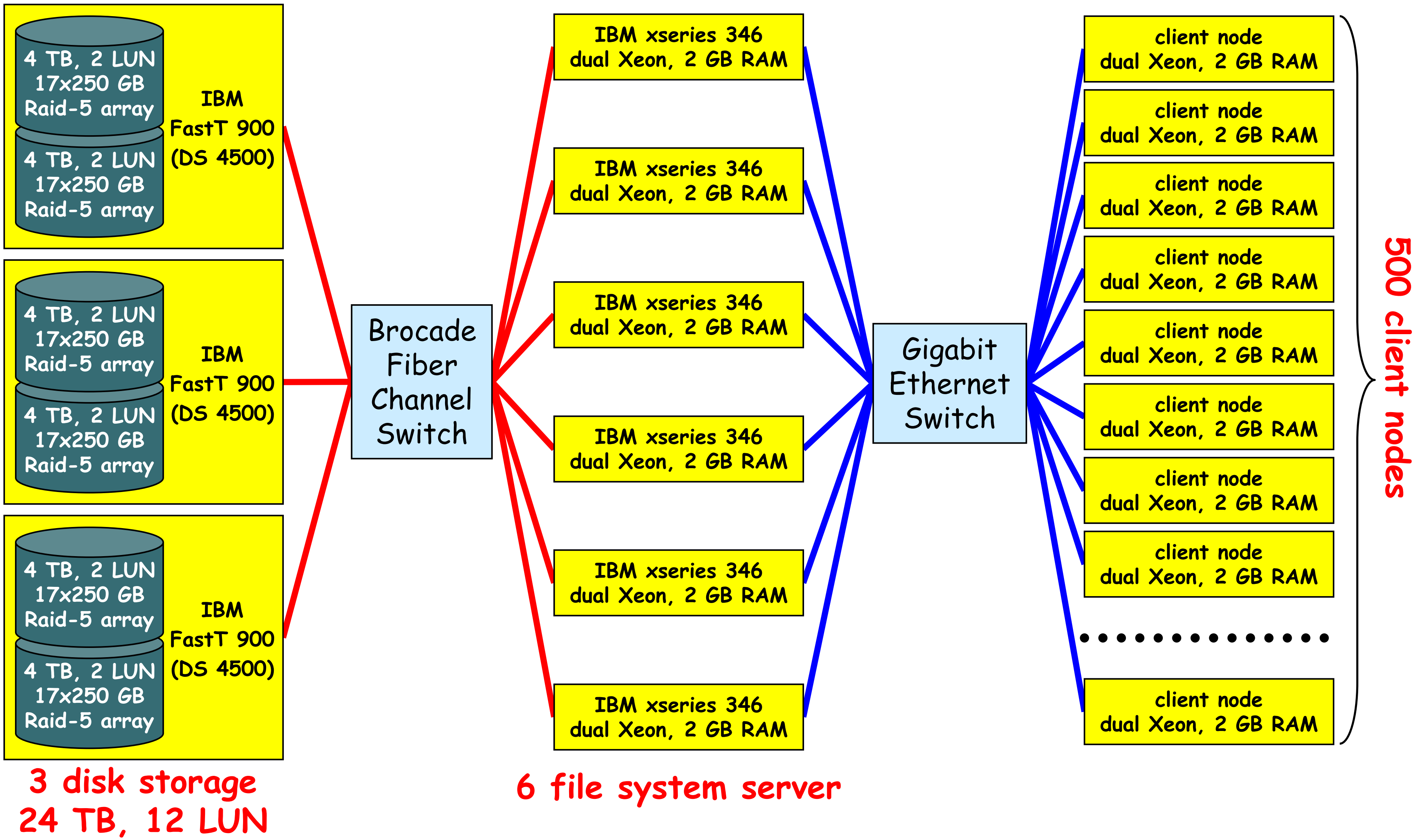
INFN-CNAF

Hepix, Roma 6th April 2006

Parallel File Systems at the INFN Tier-1: early studies in 2005

- Evaluation of **GPFS** for the implementation of a powerful disk I/O infrastructure for the TIER-1 at CNAF.
  - A **moderately high-end testbed** used for this study:
    - 6 IBM xseries 346 file servers connected via FC SAN to 3 IBM FAStT 900 (DS4500) controllers providing a total of **24 TB**.
    - 500 CPU slots (temporarily allocated) acting as clients
    - Maximum **available** throughput from server to client nodes using 6 Gb Ethernet cards in this study: **6 Gb/s**
- **PHASE 1**: **Generic tests**.
  - Comparison with **Lustre**
- **PHASE 2**: **Realistic physics analysis** jobs reading data from (not locally mounted) Parallel File System.
- Dedicated **tools** for **test** (PHASE 1) and **monitoring** have been **developed**:
  - The benchmarking tools allows the user to start, stop and monitor the test on all the clients from a single user interface
  - It implements network bandwith measurements by means of the **netperf** suite and **sequential read/write with dd**
  - The monitoring tools allow to measure the **time dependence** of the **raw network traffic** of each server with a granularity of one second

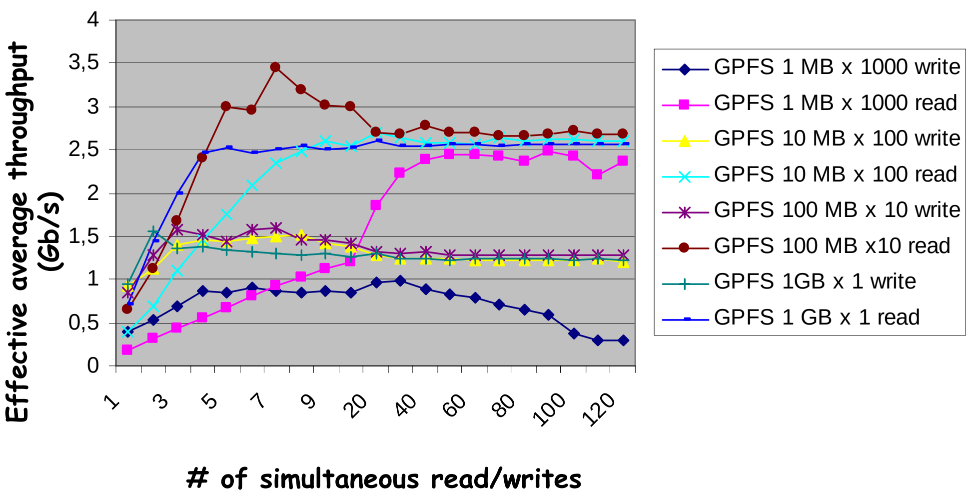
Early Parallel File System Test-bed



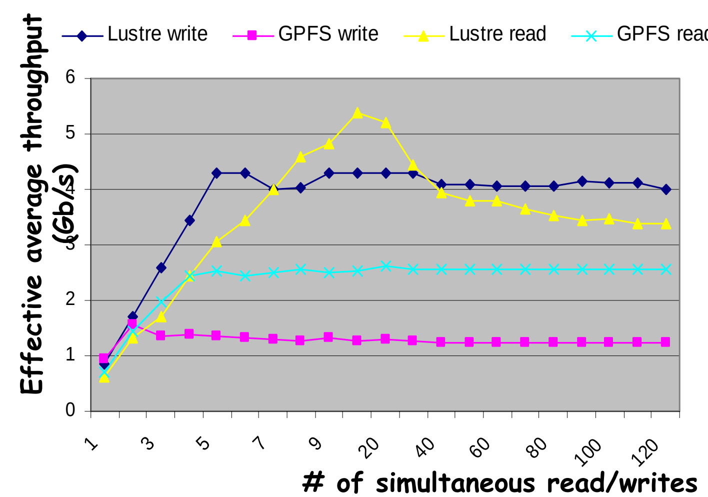
Test results

- Network tests (bidirectional saturation of 6 Gbps aggregate bandwidth to disk servers)
- GPFS robustness test
  - Done just with GPFS 2.2
  - 2.000.000 files written in 1 directory (for a total of 20 TB) by 100 processes simultaneously with native GPFS and then read back, run continuously for 3 days
  - No failures!
- Phase 1 – sequential r/w from several clients simultaneously performing I/O with different protocols (native GPFS/Lustre, RFIO over GPFS/Lustre, NFS over GPFS).
  - 1 to 30 GigaEthernet clients, 1 to 4 processes per client.
  - File sizes ranging from 1 MB to 1 GB.

Native GPFS with different file sizes

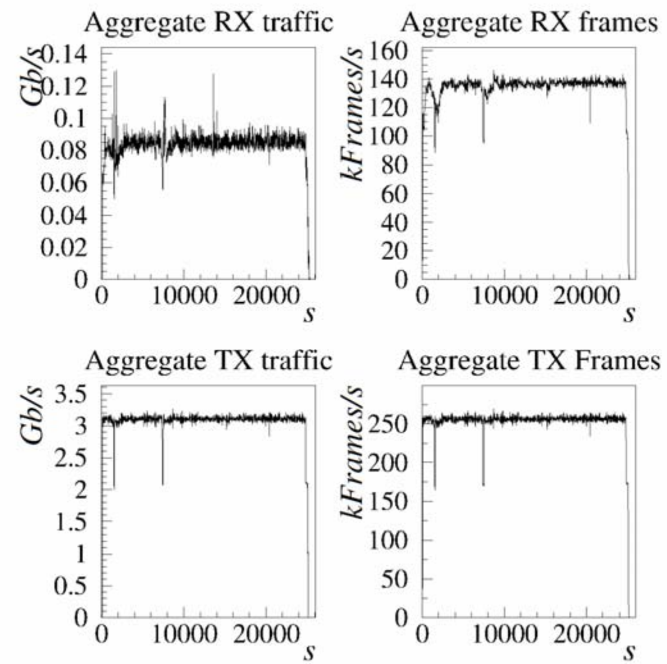


Results of read/write (1GB different files)



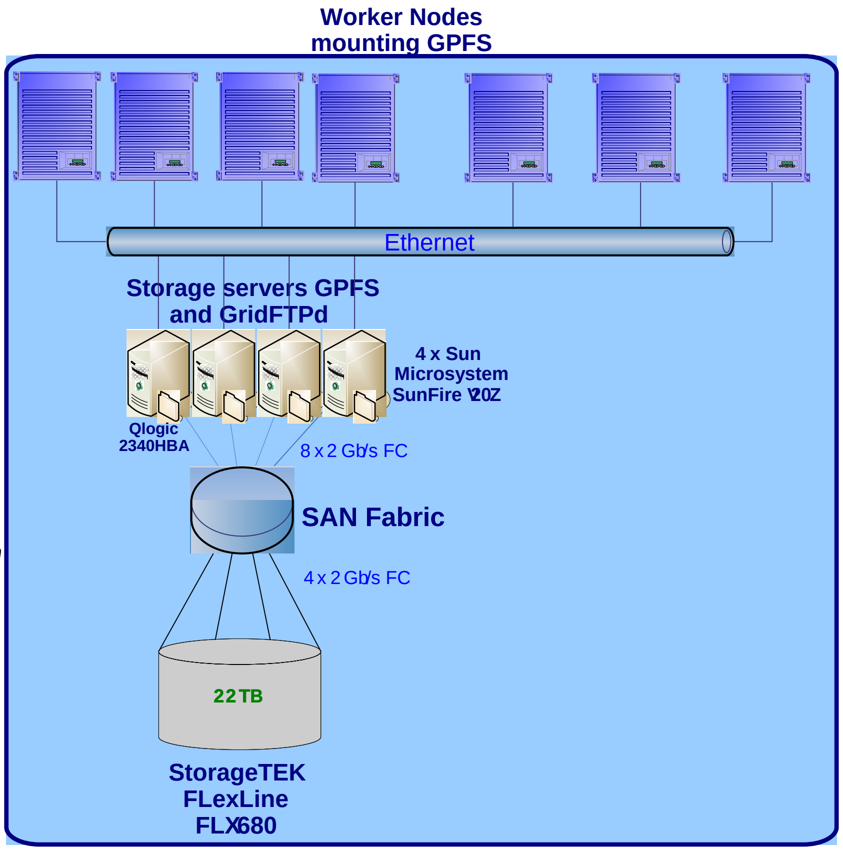
Test results : a realistic scenario

- Test with a realistic LHCb analysis algorithm
  - **Analysis Jobs** are generally the most **I/O bound** processes of the experiment activity.
  - The analysis algorithm reads sequentially **input data files** containing simulated events and produces **n-tuples** files in **output**
- Analysis jobs submitted to the production **LSF batch system**
  - 14000 jobs submitted to the queue, **500 jobs in simultaneous RUN state**
- **8.1 TB** of data served by RFIO daemons running on GPFS parallel file system servers (LUSTRE not tested for lack of time)
  - RFIO-copy to the local wn disk the file to be processed;
  - Analyze the data;
  - RFIO-copy back the output of the algorithm;
  - Cleanup files from the local disk.
- **All 8.1 TB** of data processed in **7 hours**, all **14000 jobs** completed successfully.
  - **>3 Gbit/s** raw sustained read throughput from the file servers with GPFS (about **320MByte/s** effective I/O throughput).
  - Write throughput of output data negligible (1 MB/job).
- Copying input files to the local disk is not the best approach (no guarantee for disk space availability)
- More cleaver approach (which requires SRM v2.1 and a reliable filesystem that allows to keep a file open for a while) would be to open remotely input and output file
  - SRM 2.1 functionalities needed to pin the input files and reserve space for the output files on the SE

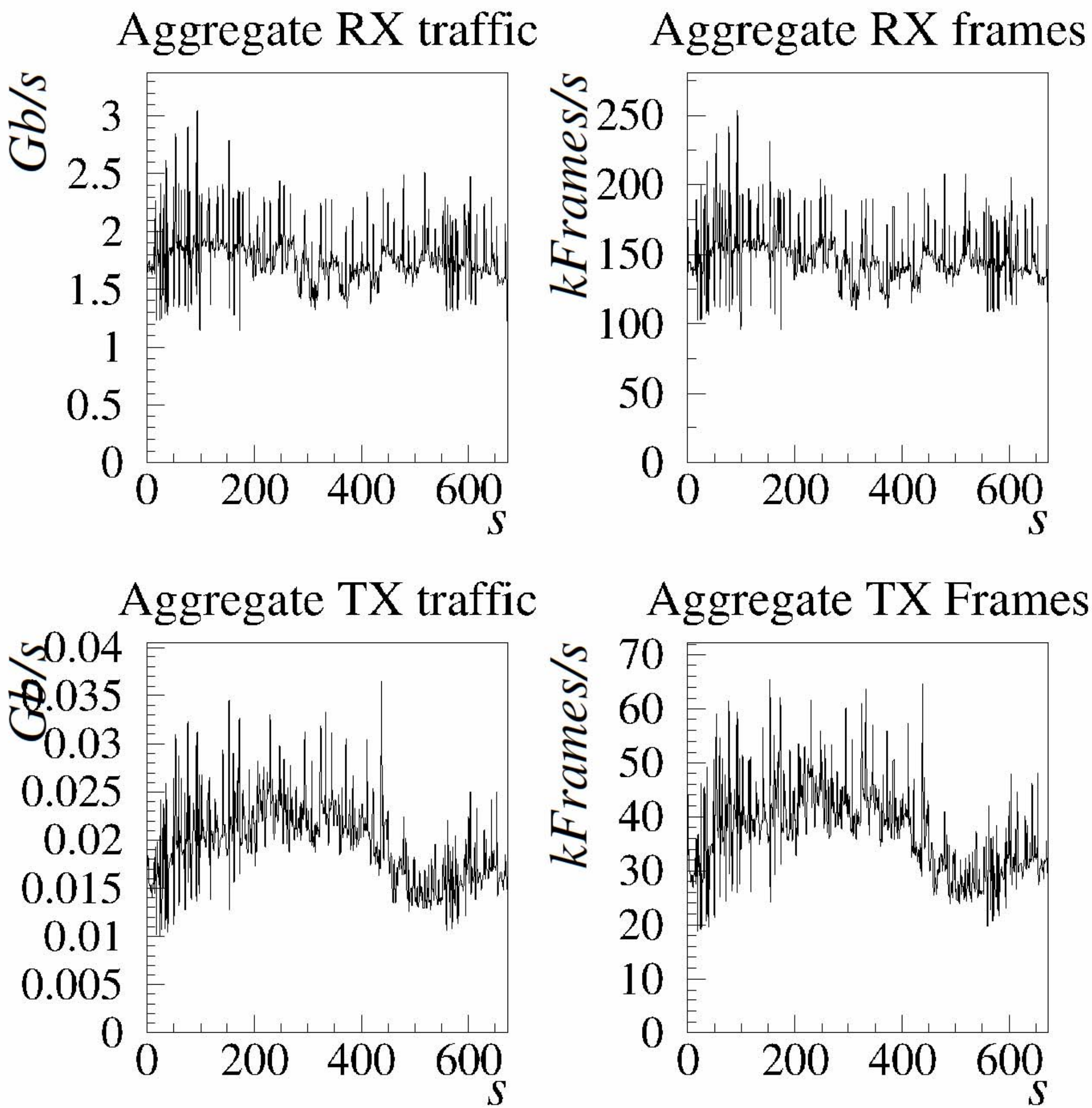


More recent studies with GPFS

- In 2006 new tests with local GPFS mount on WNs (no RFIO)
  - GPFS version 2.3.0-10
- Installation of GPFS RPMs completely “quattorized”
  - Minimal work required to adapt IBM RPM packages to become quattor compliant
- GPFS mounted on 500 boxes (most of the production farm)
- Why we (temporarily) dropped LUSTRE ?
  - Commercial product: it seems very promising and scalable (10000+ nodes) ☺
  - Stable and reliable ☺
  - Easy to install, but rather invasive ☹
    - Requires own Lustre patches to standard kernels either on server and client side
  - No support for ACL and space reservation ☹
  - GPFS already in production at Tier1……

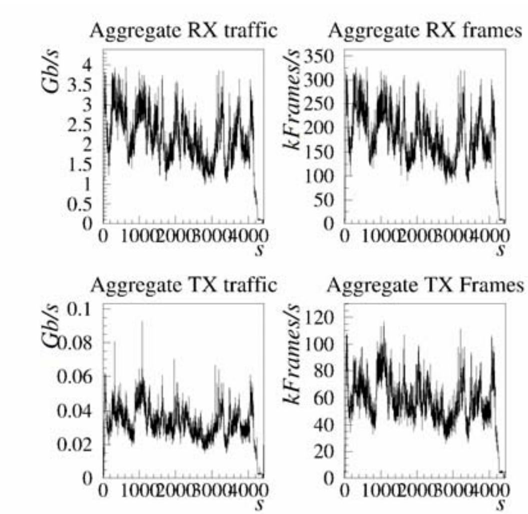


WAN data transfers



- Data transfers of pre-staged stripped LHCb data files from CERN (castorgridsc data exchanger pools) to the 4 GPFS servers via third party globus-url-copy
- 40 simultaneous transfers, dynamically balanced by the DNS, 5 streams per transfer
  - Typical file size 500 MB
- About 2 Gb/s of sustained throughput with this relatively simple testbed
- CPU load of servers: 35%
  - Including I/O wait: 15%

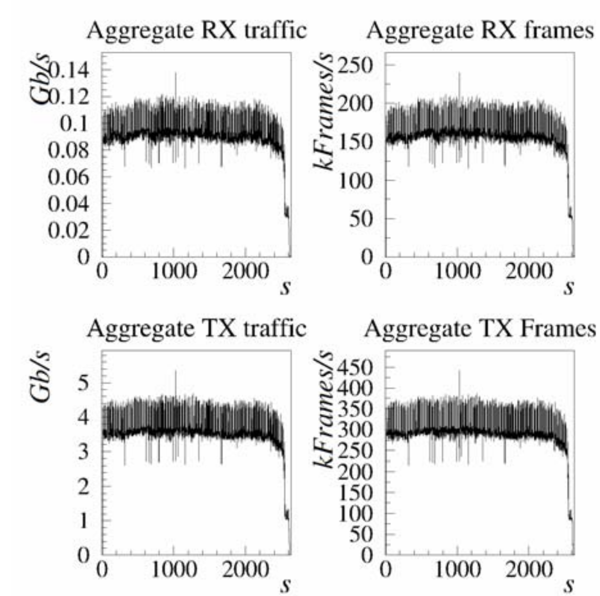
Sustained read & writes on LAN from production worker nodes



Sustained writes on LAN from production WNs

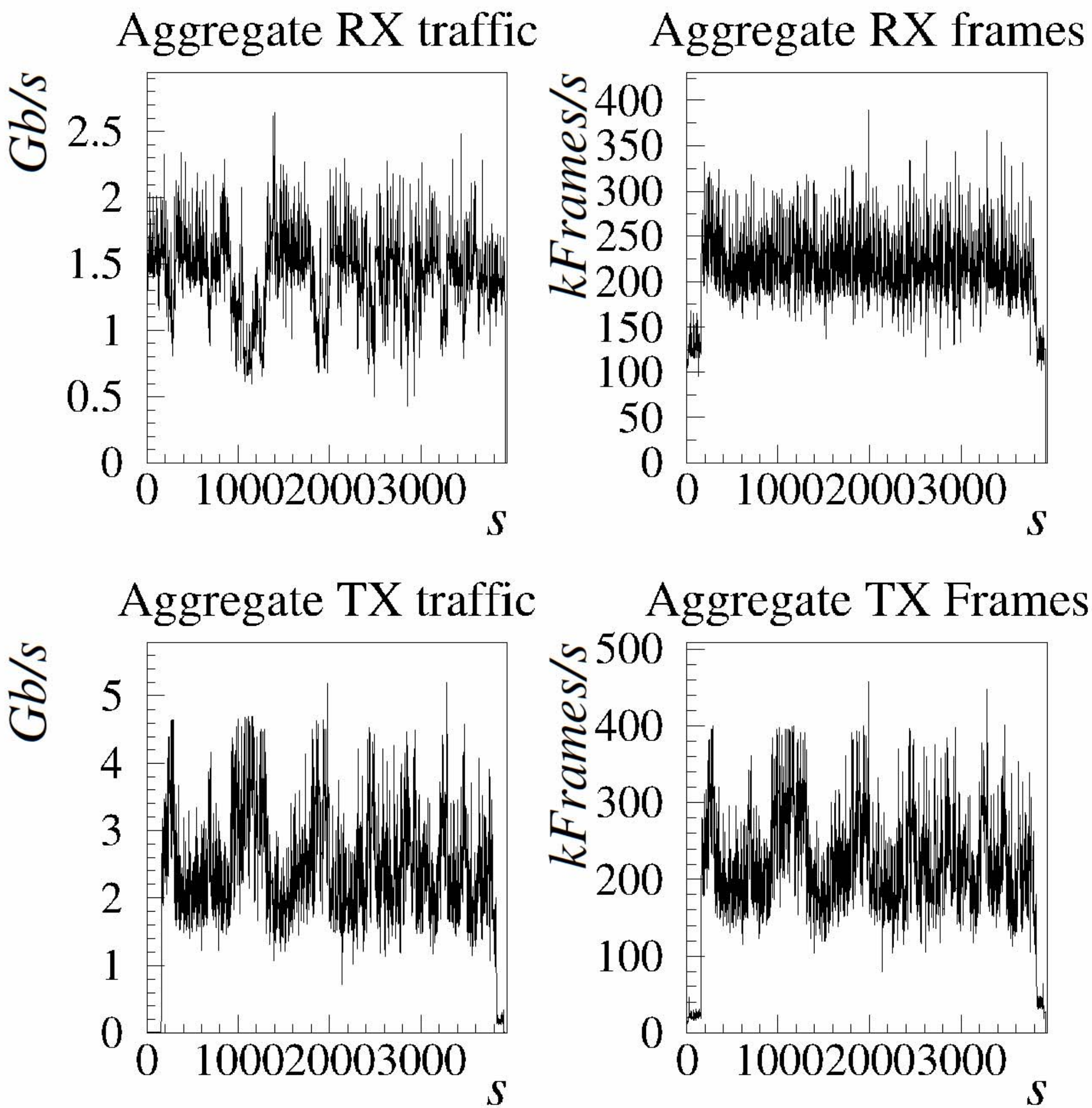
- 1000 jobs submitted to the LSF production batch
  - 400 jobs in simultaneous running state
  - 1 GB file written from each job at full available throughput
- About 2.5 Gb/s
- CPU load of servers: 70%
  - including I/O wait: 20
  - negligible on client side

- 1000 jobs submitted to the LSF production batch
  - 300 jobs in simultaneous running state
  - 1 GB file read from each job at full available throughput
- 4 Gb/s
  - Maximum available bandwidth used
- CPU load of servers: 85%
  - including I/O wait: 50%
  - negligible on client side



Sustained read on LAN from production WNs

A more realistic scenario: sustained WAN data transfers and local LAN read from worker nodes at the same time



- 40x5 streams from CERN to CNAF
- 1000 jobs submitted to the LSF production batch
  - 550 jobs in simultaneous running state
  - 1 GB file read from each job at full available throughput
- About 1.7 Gb/s from CERN and 2.5 Gb/s to worker nodes
- CPU load of servers: 100%
  - including I/O wait: 60%
  - negligible on client side

GPFS summary (1)

- Commercial product, initially developed by IBM for the SP systems and then ported to Linux
  - Free for academic use, but very difficult to have support from IBM (even paying…)
- Stable, reliable, fault tolerant, indicated for storage of critical data
  - Possibility to have data and metadata redundancy
    - Expensive solution, as it requires the replication of the whole files, indicated for storage of critical data
  - Data and metadata striping
  - Data recovery for filesystem corruption available, fsck
  - Fault tolerant features oriented to SAN and internal health monitoring through network heartbeat
  - Interesting performance figures, already at the scale of what required "one day" (not so far actually…)
- Easy to install, not invasive
  - Distributed as binaries or sources in RPM packages (smart repackaging needed for easy installation)
  - No patches to standard kernels are required (apart for small bug fixes on the server side already included in newer kernels), just a few kernel modules for POSIX I/O to be compiled for the running kernel
- POSIX I/O access, every existing application can use these filesystems as they are without any adaptation

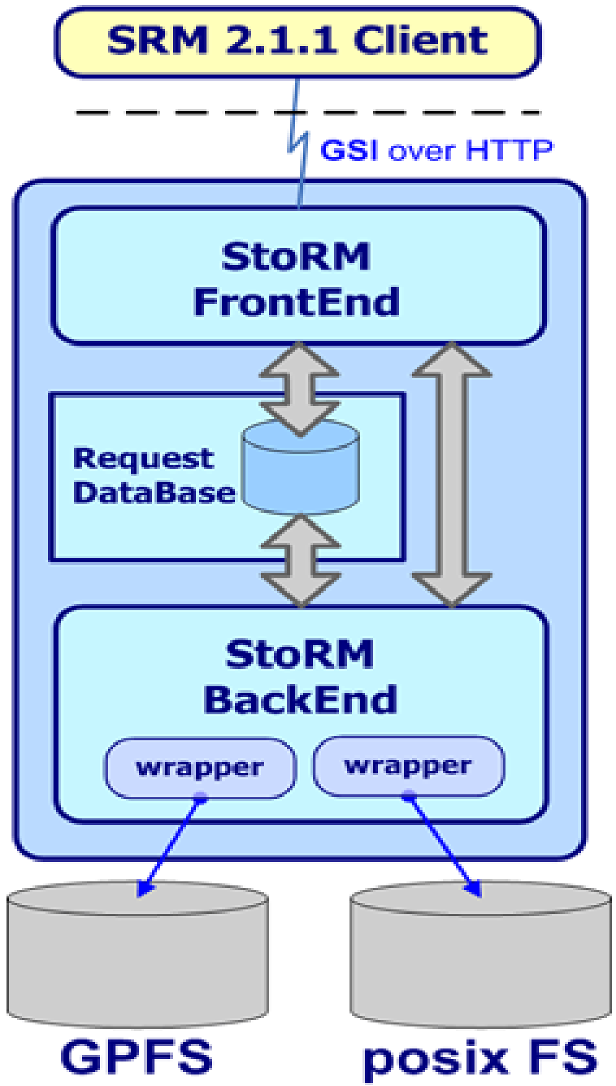
GPFS summary (2)

- In principle requires every machine in the cluster (clients and servers) to have each-other root authentication without password (with rsh or ssh)
  - In case one gets root privileges on one machine, all machines can be hacked
  - This is not a nice feature for security and seems like a quick and dirty way adopted when porting the software to Linux
  - We implemented a workaround restricting the access of the clients to the servers by means of ssh forced-command wrappers
- Advanced command line interface for configuration and management but…
- … the configuration of the cluster (tuning parameters, topology of the cluster, address of servers nodes, disks, etc.) has to be replicated on each node by means of ssh via a push mechanism
  - Pull mechanism however foreseen, e.g. in case the configuration has changed while a node was down, then the node can pull the new configuration when it comes up
  - Lustre solves the problem of deploying the cluster configuration by using an LDAP-based centralized information service
- For advanced storage management they require a dedicated SRM (see StoRM below), then naturally become fully GRID-compliant disk-based storage solutions, and can be solid building blocks toward GRID standardization in the I/O sector

SRM and StoRM

- StoRM is a disk based Storage Resource Manager which:
  - **implements SRM specification version 2.1.1**
    - **WS-I compliant version, named "2.1.1_modified".**
  - **is designed to support guaranteed space reservation.**
  - **supports direct access (native posix I/O calls).**
    - **Other access protocols remain available (e.g., rfio).**
  - **takes advantage of high performance Cluster File System with ACL support, such as GPFS.**
    - **Other posix file systems are supported (e.g., ext3)**
  - **Authentication and Authorization are based on VOMS certificates.**
- Current release (1.1.0) provides these functionalities:
  - **Data transfer** : srmCopy, srmPtG, srmPtP, srmStatus<XXX>
  - **Space Management** : srmReserveSpace, srmGetSpaceMetadata
  - **Directory** : srmLs, srmRm, srmMkDir, srmRmdir.
- Production release ready next May

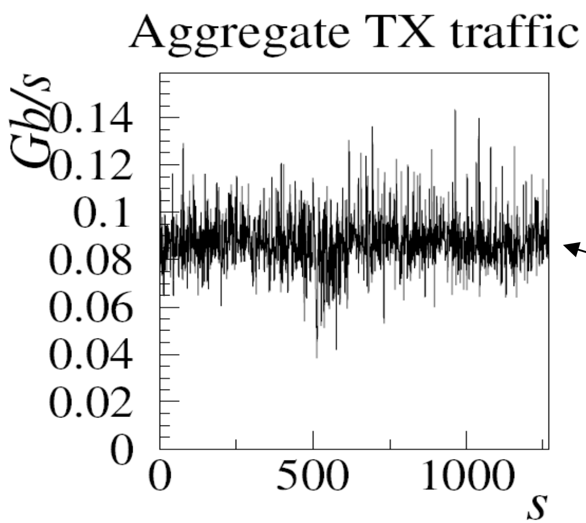
StoRM architecture



- Front end (FE) has responsibilities of :
  - expose a web service interface
  - manage connection with authorized clients
  - store asynchronous request into data base.
  - retrieve asynchronous request status.
  - co-operate with backend directly for synchronous call.
  - co-operate with external authorization service to enforce security policy on service.
  - manage user authentication
- Data Base :
  - Store SRM request and status
  - Store application data
- Back end (BE) has responsibilities of :
  - accomplish all synchronous (active) action.
  - get asynchronous request from data base.
  - accomplish all asynchronous action.
  - bind with underlying file system.
  - enforce authorization policy on files
  - manage SRM file and space metadata.

Preliminary tests

- Tests with 1.1.0
- 4 sites involved
  - Tier1 (22T) - Stress test and transfer test
  - Bari (2TB) - Transfer test
  - ICTP-Trieste (30GB) - Functionality tests
  - CNAF-Cert-SE (50GB) - Functionality tests



Data transfer T1 to/from Bari via srmCopy v.2.1.1

Only 100 Mb/s access bandwidth to BARI

50 parallel srmCopy with:

- From SURL at CNAF
- To SURL at BARI
- 1GB file size, everyone

Next planned transfertest with larger access bandwidth

People involved

A lot of people contributed to these test activities:

- INFN Tier1 staff (INFN-CNAF)
- StoRM development team (INFN-CNAF, ICTP)
- LHCb Bologna group