Plans for IN2P3

Lionel Schwarz <schwarz@cc.in2p3.fr>

Table of contents

CC-IN2P3 overview

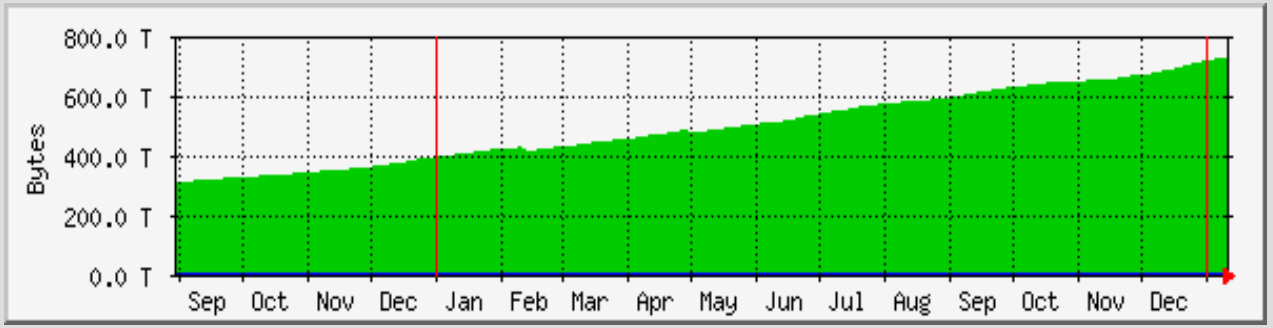
- > 40 experiments HEP (LHC, BaBar, D0 ...) , Astro (Auger, Snovae ...) & Biology
- ~60 FTE (~12 for LCG & EGEE)
- 1500 CPUs running Linux RH7.2, RH7.3, SL3
- ~2000 running jobs managed by BQS in one single farm
- 800 TB stored in HPSS
- 100 TB on disk

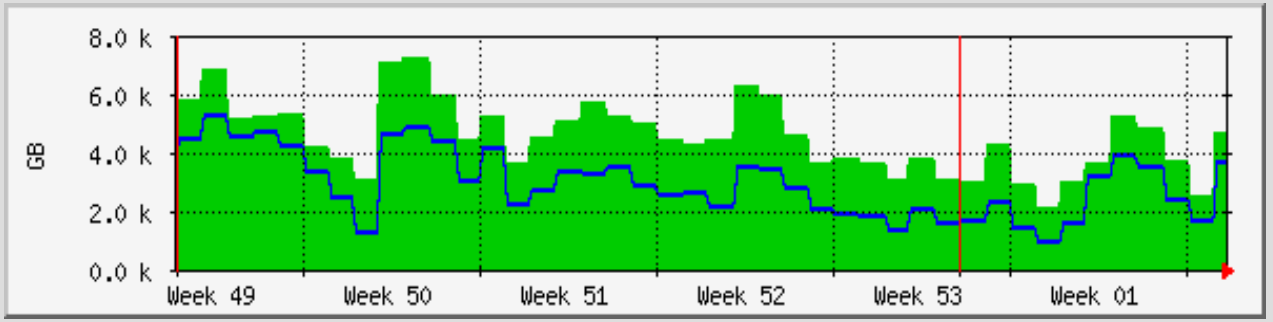
Disk storage

- Xrootd
  - 40 TB served by 15 servers
  - Up to 600 concurrent connexions
  - Easy to manage and scalable
  - Interfaced with HPSS (300 TB stored)
  - In production since 2003 (replaced Objectivity for Babar)
- “semi-permanent” storage
  - 15 TB served by 9 NFS servers (20 TB expected end 01/05)
  - Not scalable
  - Plans to use a global file system

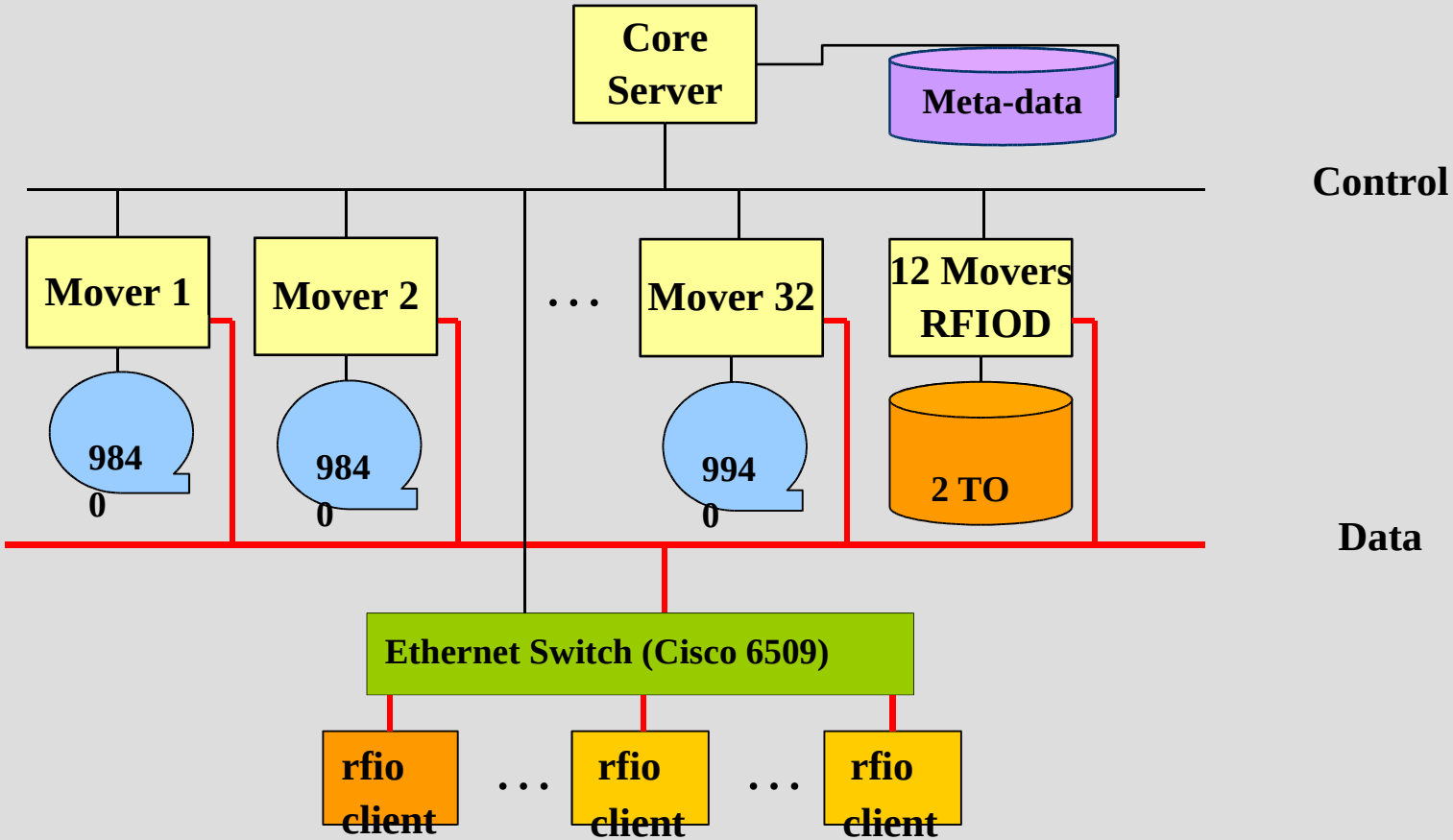
HPSS

- In production since 1999
- 800 TB stored
  - 6 STK robots
  - 9840, 9940 cartridges
  - 25 TB cache disk
- Average usage: 4TB/day





HPSS Architecture



HPSS Access

- Local access through RFIO (rfcp ou API)
- COS: Class of service
  - Dynamic resources affectation
  - Standard COS 0 (automatic search for best COS from file size)
  - Specific COS for specific needs (Babar, LHC)
- Interface with Xrootd in production
- Remote access: SRB, BBFTP, GridFTP

HPSS Grid interfaces

- EDG: interface with WP5-SE (RFIO API): not used anymore
- LCG2
  - 2 GridFTP servers (based on CERN's CastorGridftp) interfaced with HPSS in production (RFIO API)
  - Evaluation of LBNL's HRM v1 (pFtp)
  - SRM-dCache/HPSS interface in tests (based on BNL's work)

dCache SRM for HPSS

- HPSS-GridFTP provides poor performances:
  - Uses RFIO API (average 2MB/s)
  - COS 0 not used (file size not known before transfer starts: “PRET” not implemented in Globus server)
- dCache is better:
  - Uses rfcp (up to 30MB/s on 1G)
  - Customization for HPSS: COS and RFIO server
  - Load balancing (for gridftp transfers)
  - Asynchronous migration

SRM-dCache tests (so far)

- Setup
  - 1 head node (admin node) with SRM enabled
  - 2 pool nodes with 2 pools (3 GB) each with GridFTP servers
- Test description
  - 10 concurrent PUT (local to SRM) of 50MB files
  - Transfers (client-pool) done by gridftp
  - Client program: srmcp (part of dCache client)
- Results (so far)
  - Migration script => OK
  - dCache configuration for HPSS => OK
  - Still stability issues with gridftp servers on pool nodes

Challenge current situation

- Network: 1Gb link to GEANT (through Renater)
  - 2nd 1GB link? (Amsterdam meeting)
- 2 nodes (IBM x330, 1GB RAM, 500GB FC)
- GridFTP server configured with 1MB buffers
- Transfers done at 30MB/s on each node
- Tuning still being done

Plans for SRM-SRM challenge

- Use dCache as SRM front-end
- Modify hardware infrastructure
  - Gigabit interfaces on pool nodes
  - Increase disk space on pool nodes (~2TB per node)
  - Maybe increase # of pool nodes
- Further tests
  - Performances
  - Big files (1 GB)
  - Staging script (GET transfers)
  - Tape only COS (modify HPSS configuration)

Summary

- Network: 1Gb link with CERN
  - 10Gb end 2005
- SRM access:
  - dCache: the only solution so far
  - but needs much more testing
  - and new setup with more disk