# Castor status and plans

Sebastien Ponce, Hepix, October 22nd 2008

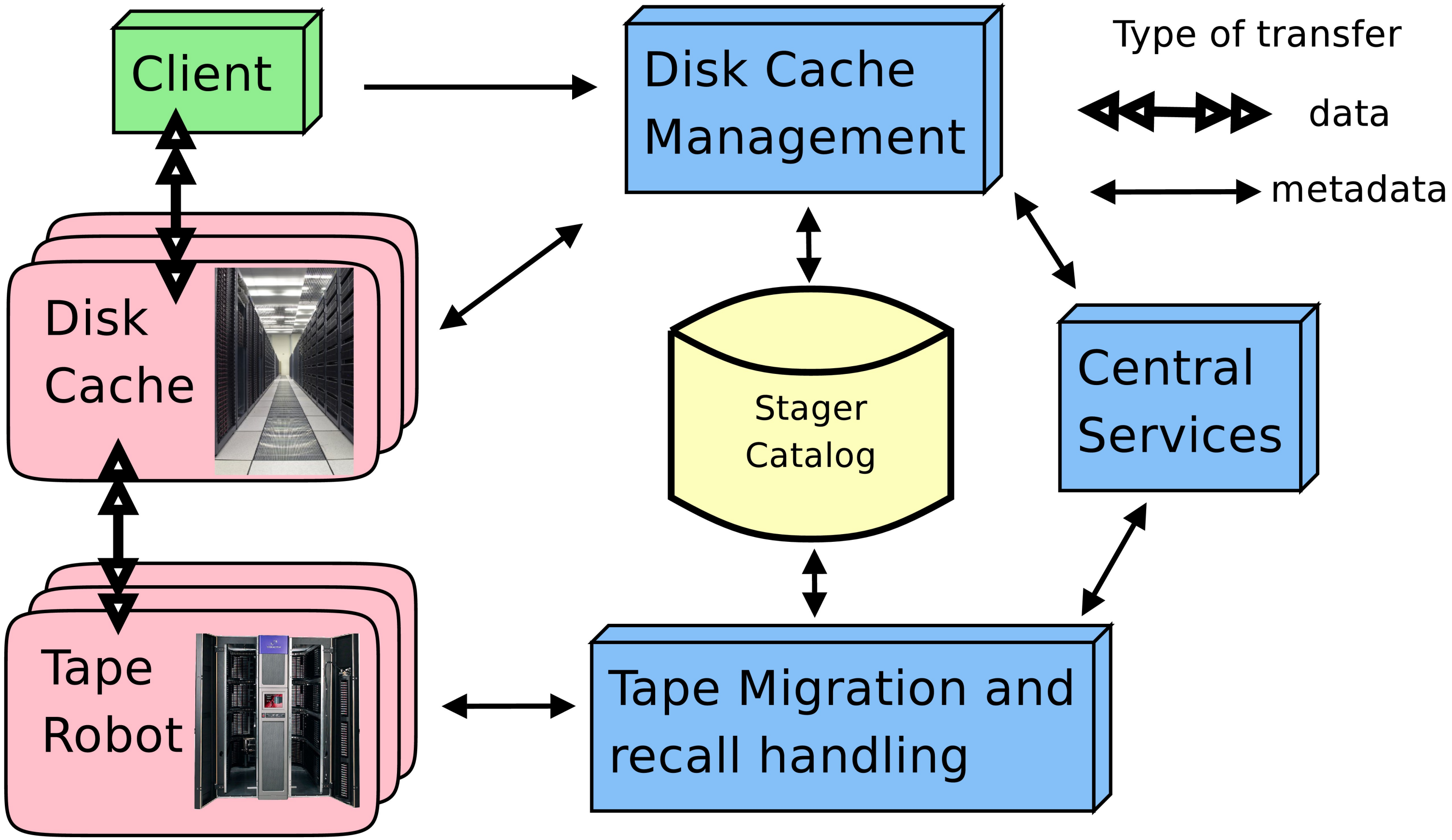
# Outline

- Overview of CASTOR
- Current status
  - And latest improvements
- Use CASTOR for analysis
  - XROOT-CASTOR integration
- Future plans
  - Ongoing and planned developments

# CASTOR overview

- a mass storage solution targeting the CERN Tier 0 and Tier 1s
- Handling 2 layers of storage
  - Tape archive
  - Disk cache
- Providing a unique namespace
- Able to handle large amounts of data
  - 100s PBs on tape
  - 10s PBs on disk
  - 10s GB/s of throughput

# Architecture

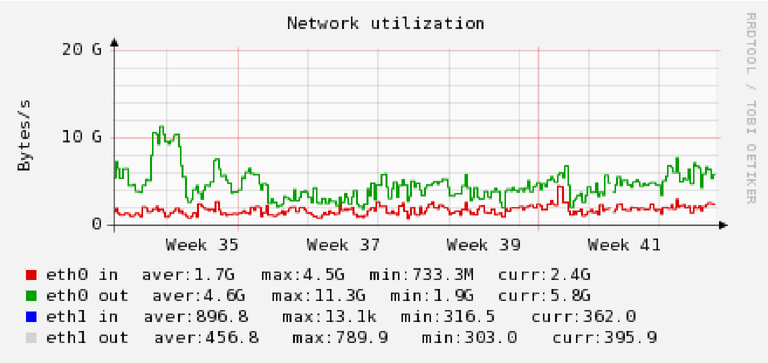


# Key points

- Database centric
  - All components communicate through ORACLE databases
- Robust
  - Redundancy at all levels
    - DB & all daemons
- Scalable
  - In data sizes : 18PB handled today
  - In throughput : 6GB/s av on a year

# Successful

- Handles the CERN Tier 0 activity
  - > 6GB/s average
  - > 10GB/s peak
  - 6 PB disk cache



- Met and exceeded expectations during data challenges
- Also installed in 3 Tier 1s
  - ASGC in Taiwan
  - RAL in UK
  - CNAF in Italy

# Continuously improving

- Security
  - All components have been secured (authentication)
    - Nameserver, disk cache
    - Protocols
  - Support for globus and kerberos 5
  - 1300 authentications/s with krb5 and no cache replay
- Accounting
  - Per user and per pool
- End to end checksumming
  - Possibility of presetting the checksum of a file
- Support for SLC5

# Analysis : a new use case

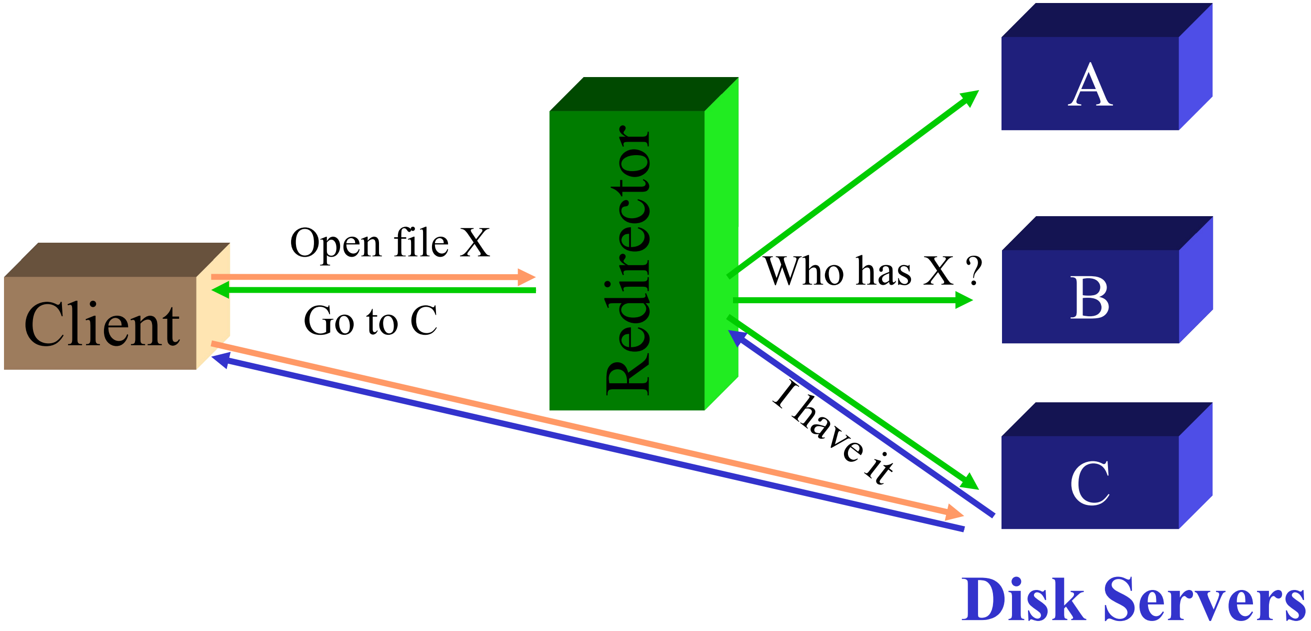
- CASTOR has been mainly foccussed on the Tier 0 activity until now
  - Data taking, export, reconstruction
  - Large files, heavy streams
  - Opening time was not an issue
- CERN is considering having an analysis facility
  - Mostly disk only
- New requirements have thus appeared
  - Support for small files
  - Low latency for file opening
  - Support for many parallel light weight streams

# Handle the analysis

- An architecture task force took place before the summer to analyze the needed architecture changes
- We need a protocol able to handle many streams concurrently per diskserver with limited overhead and low latency
- Conclusions are
  - The centralized (LSF) scheduling has too high latency
  - We focus on a single protocol and moved the scheduling within it
  - The protocol of choice is XROOT with CASTOR specific extensions

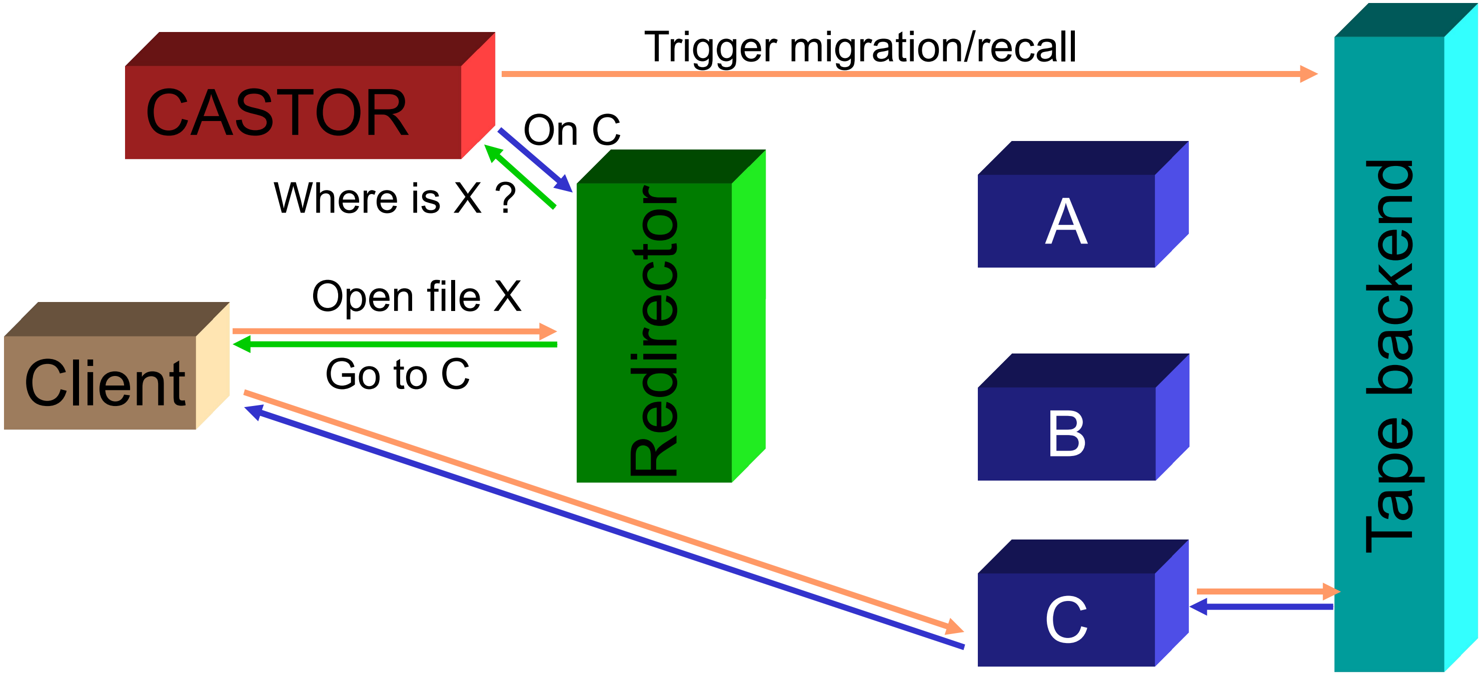
# XROOT overview

- Client connects to a redirector node
- This redirector finds out where the file is
  - It handles a cache of recent files for efficiency
- Client then connects directly to the node holding the data



# XROOT in CASTOR

- Client connects to a redirector node
- The redirector asks CASTOR where the file is
- Client then connects directly to the node holding the data
- CASTOR handles tapes in the back



Disk Servers

# The gains

- Benefits from low latency of XROOT
  - 80ms per file opening (1-2s for CASTOR)
- Many connections per second (small files)
  - >700 connections per second
- And native xroot for bandwidth optimization
  - Can serve concurrently 100s of streams per node
- Note :
  - New schema only used for analysis for now
    - Still using plain CASTOR-LSF based scheduling for Tier 0 related pools

# Practically

- New XROOT plugin in CASTOR
  - Tighter integration, aware of CASTOR concepts
    - e.g. service class, disabled disk server
- Extensions of XROOT
  - Security (Globus, kerberos)
  - Stream scheduling on a disk server
    - Ability to dynamically lower throughput dedicated to users when a tape stream starts
  - Configurable redirector
    - Can use its cache or CASTOR (or both)
    - Can use its scheduling or CASTOR's (or both)

# Future plans

- More improvements on xroot
  - Apply to writes what was done for reads
  - Study the move the file catalog to the xroot redirector to have fully native xroot access
- Tests of mountable CASTOR
  - Take advantage of XROOT via FUSE
  - Find out usage patterns and possible concerns
- Quotas
- Support of small files at the tape level
  - See Steven's talk

# Summary

- A robust and efficient Mass storage system
  - Used successfully in production for Tier 0, Tier 1s
- Evolving to answer new needs
  - accounting/quotas
  - Security
  - Small files
- Adapting to new technologies to tackle new use cases
  - Analysis scenario
  - Mountable mass storage