# CASTOR 2

***History, architecture, development process, running service***

*Felix Ehm CERN / IT / GD*
*April, 2007*

# Outline

- **Scope and constraints**
- **Short history**
  - SHIFT, CASTOR 1, CASTOR 2
  - Architecture overview
  - Breakdown, DB centric
  - Core framework
- **Deployment**
- **Performance**

# Scope

- **A hierarchical Mass Storage System**
  - Storing data on tapes
  - Handling the tape drives/robots
  - Handling a level of disk caches
  - Providing easy access to the data

- **Some numbers today**
  - 50M files on tape
  - 7 PB of data on tape
  - 1,5 PB disk cache deployed (2,2 PB avail.)
  - Sustained 1-2 GB/s data rate
    - Goal is 4 GB/s of sustained, incoming data
    - 14 PB/year

# Constraints

- **Assure that CERN can fulfill the Tier-0 and Tier-1 storage requirements for the LHC experiments**
  - Central Data Recording (CDR)
  - Data reconstruction
  - Data export to Tier-1 centers

# Outline

- Scope and constraints
- **Short history**
  - SHIFT, CASTOR 1, CASTOR 2
- Architecture overview
  - Breakdown, DB centric
  - Core framework
- Deployment
- Performances

# History - SHIFT

- **S**calable **H**eterogeneous **I**ntegrated **F**acili**T**y
  - Started in early 90's
  - All user file access on disk. No direct tape access
  - Users access files by Tape volume (VID) + tape file sequence number (FSEQ)
  - The experiments normally had their own catalog on top (e.g. FATMEN)

- **Limits**
  - Data rate: 10MB/s per stream
  - Stager does not scale over 10,000 files
  - SHIFT does not support many concurrent accesses
  - HSM (automatic migration/recall of files) is not available

# History – CASTOR 1

- **Cern Advanced STORage manager**
  - Started in 1999
  - Managed storage : tapes hidden from the users
  - Hierarchical UNIX directory namespace added via the name server
  - Users access file by their CASTOR file names
  - Full HSM with automated migration and recall

- **Limits**
  - Stager unresponsive beyond 200k disk resident files
  - Stager code had reached a state where it had become practically unmaintainable
  - Everything in Memory (data loss, if machine crashes)
  - and more …

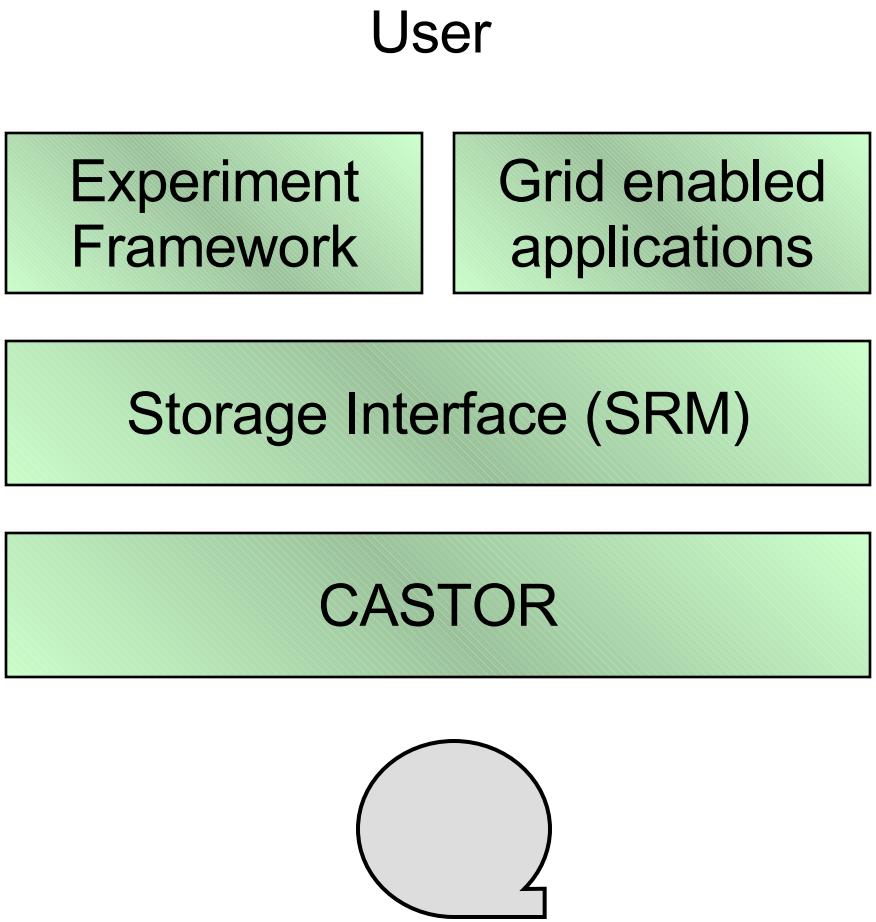
# History – CASTOR 2

- **CASTOR 2**
  - Started in June 2006 as replacement for CASTOR 1
  - Pluggable framework rather than all-in-one-wonder solution
  - Stateless components
  - Scheduling for incoming Requests
  - Introducing Policies (Migration/Garbage Collection)
  - Raising the system limits for handling
    - Requests
    - Number of files
  - Sophisticated File system selection
  - API
  - Max. Backward compatibility

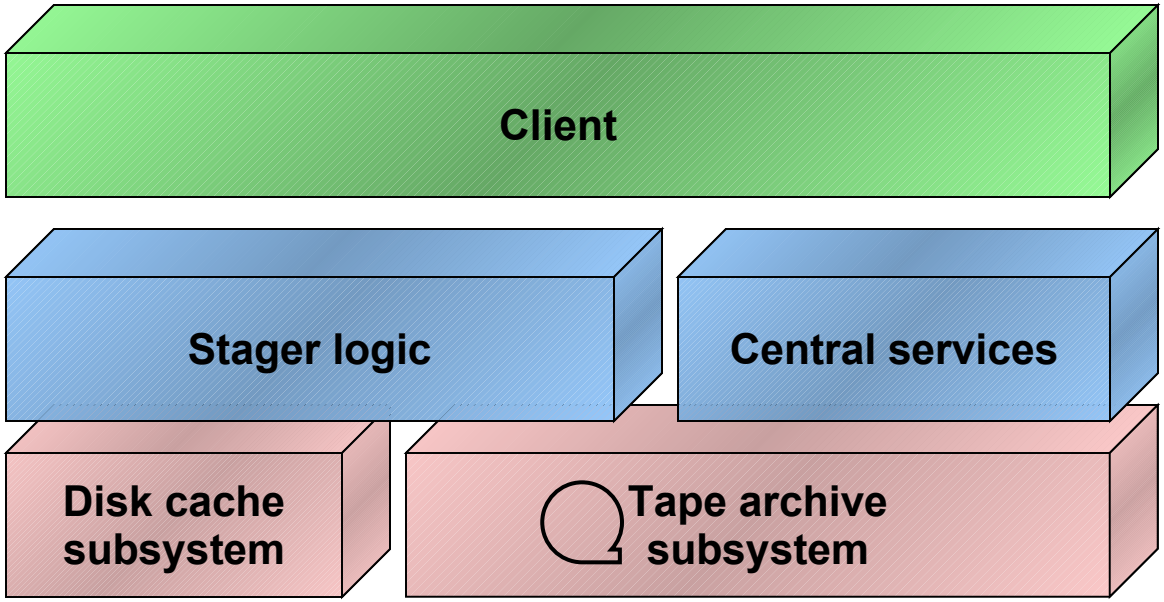
# Outline

- Scope and constraints
- Short history
  - SHIFT, CASTOR 1, CASTOR 2
- **Architecture overview**
  - Breakdown, DB centric
  - Core framework
- Deployment
- Performances

# Context View



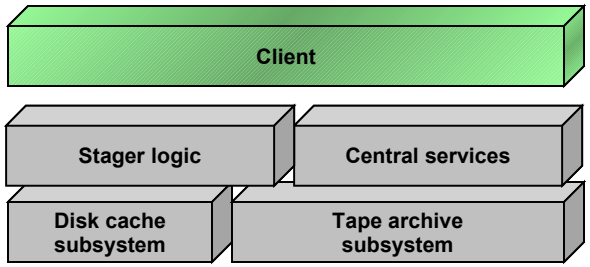
# General Picture of Castor




- **Shared with Castor 1**
  - Central services (e.g. NameServer), tape part
- **New in Castor 2**
  - Client API, Stager, disk management part

# Client deliverables

- **CLI for end users**
  - stager commands (**`stager_xxx`**)
  - RFIO commands (**`rfxxx`**)



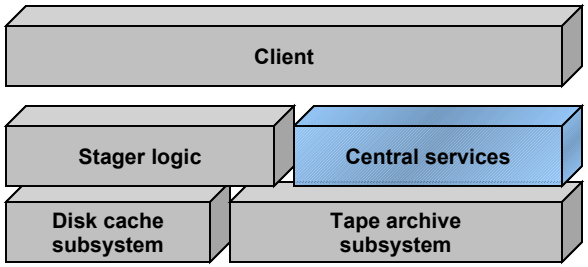

- **Client API**
  - only C
  - internal API in C++

- **Supported Protocols**
  - RFIO: rfio://server:port//castor/cern.ch/...
  - ROOT: root://server:port//castor/cern.ch/...
  - XROOT
  - GridFTP v1 : gsiftp://server:port//local/mnt/point/...

- **SRM v1 and v2 interface**

# Central Services

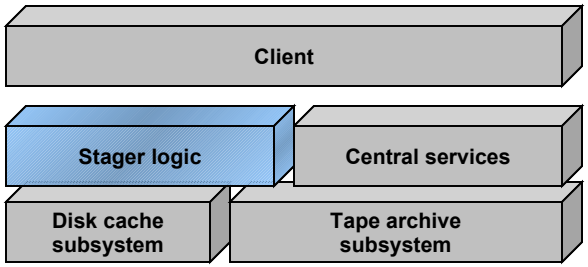
- **NameServer**
  - Database for the Castor “FileSystem”
  - Stores tape-related info as well




- **Volume and Drive Queue Manager (VDQM)**
  - Daemon for drive queue management
- **Volume Manager (VMGR)**
  - Archive of all tapes available in the libraries
- **Castor User Privileges (CUPV)**
  - Authorization daemon: provides rights to users and admins for tape related operations

# Stager

- **Database centric architecture**
  - "Surrounding" daemons are stateless
  - Well defined database interfaces separated from the rest of the code
  - **Oracle** fully supported, **PostgreSQL/MySQL** partially implemented

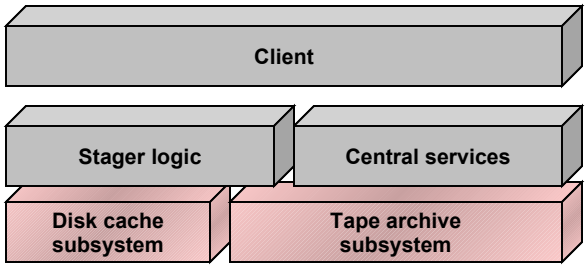



- **Stateless components**
  - can be restarted/parallelized easily
  - Stager split in many independent services
    - fully scalable

- **Minimal footprint of inactive requests**
  - Requests are not instantiated in terms of processes until they run
    - Stored in DB and/or scheduler while waiting for resources

# Disk Cache and Tape Archive

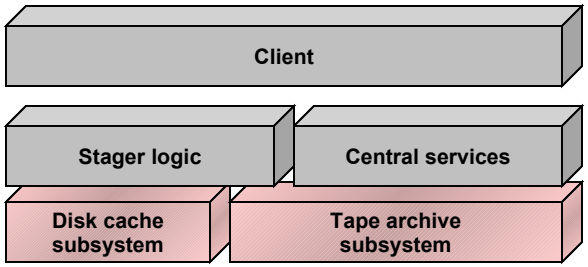
- **Scheduled disk access**
  - 2 schedulers supported
    - **LSF**, expensive, fully supported and recommended for Tier1 size setups
    - **Maui**, open-source (development frozen in Mar 2005)




- **Dynamic *migration / recall* streams to / from tape**
  - Multiple concurrent requests for same volume will be processed together
  - New requests arriving after the stream has started are automatically added to the stream

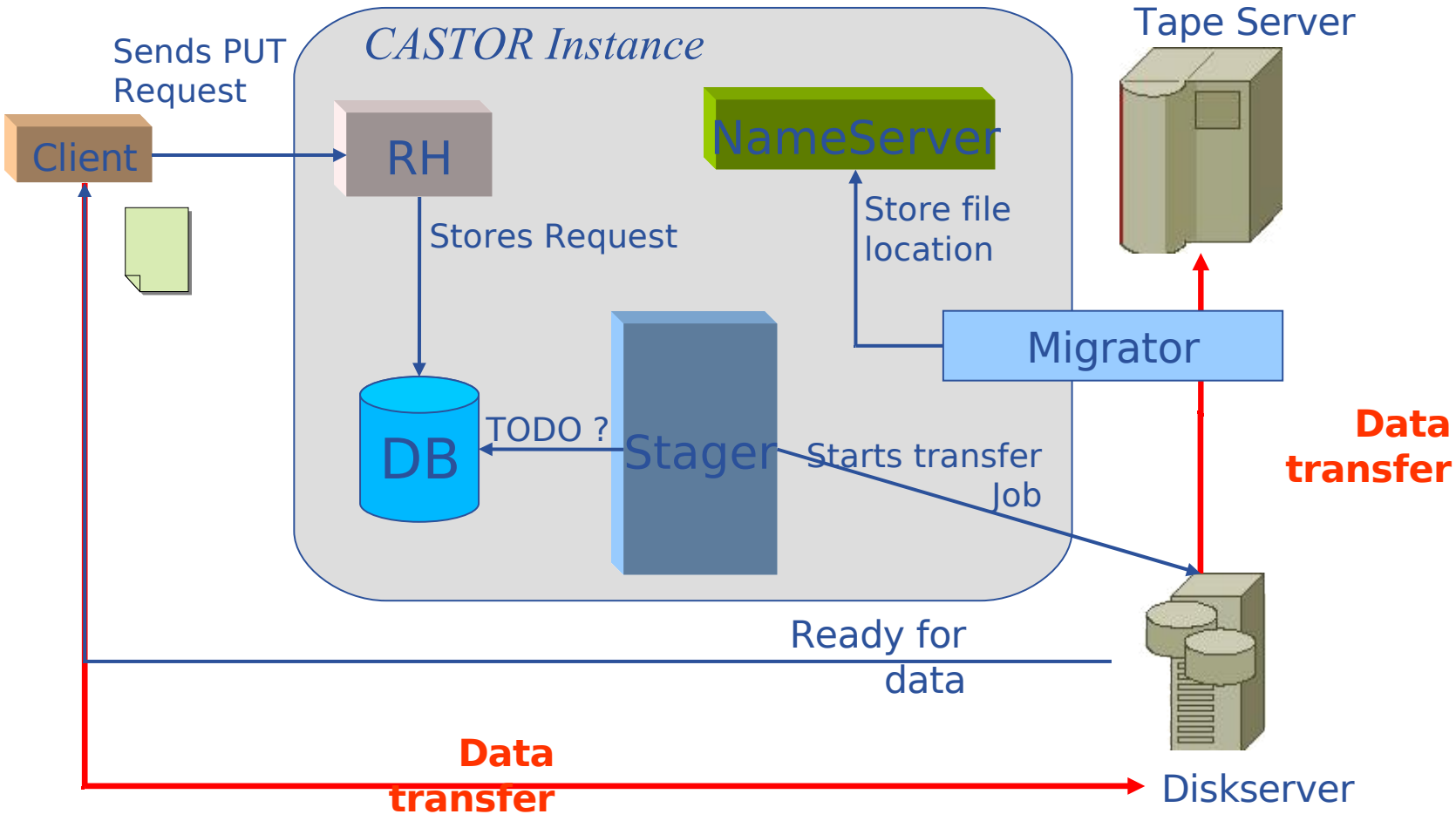
# Disk Cache and Tape Archive

- **Pluggable policies**
  - For recall, migration, I/O scheduling, GC
  - Allows support for
    - volatile storage (GC, no migration)
    - durable storage (no GC, no migration)
    - permanent storage (GC, migration)

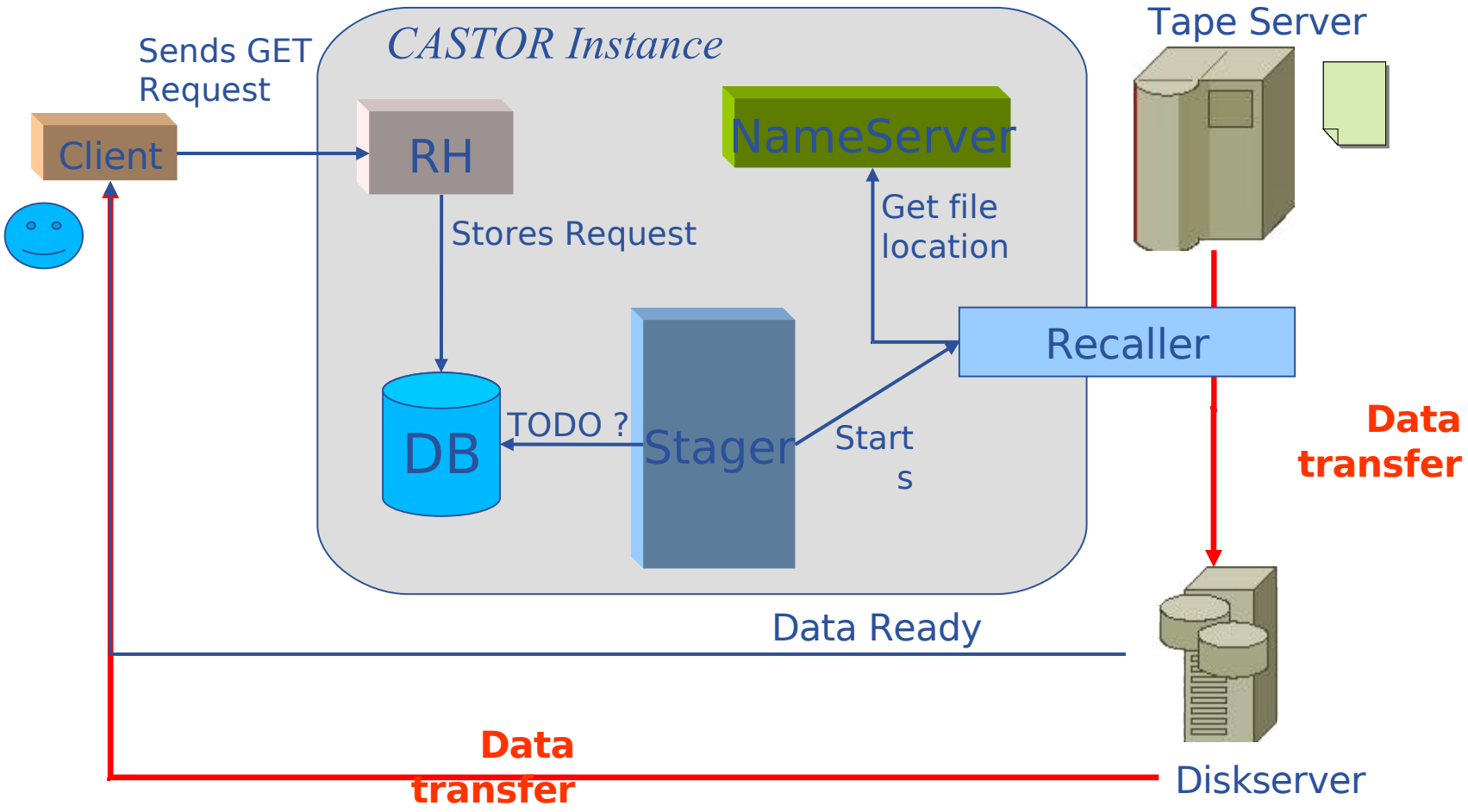



- **"Pluggable" protocols**
  - RFIO and ROOT internal, XROOT coming, GridFTP external, will become internal

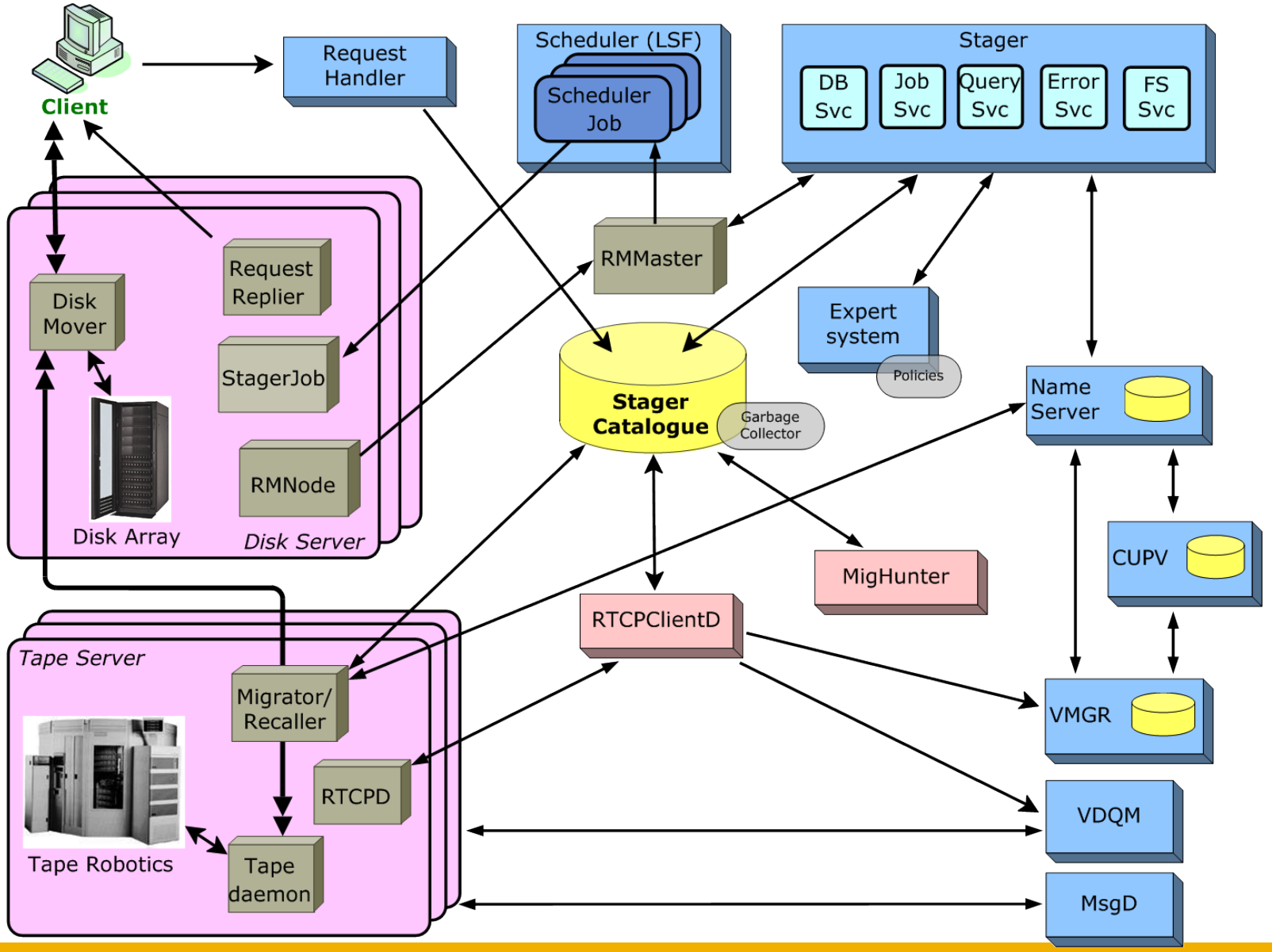
# Workflow PUT



# Workflow GET



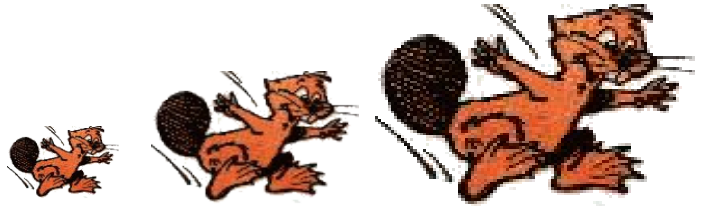
# Detailed picture of CASTOR



# Why Database Centric ?

- **Reliability**
  - No single point of failure
    - By replicating all components
  - Locking handled by the DB
  - Backups handled by the DB

- **Scalability**
  - All component can be replicated
  - No catalog in memory
  - Limited by DB scalability
    - No risk from the space point of view
    - CPU is the limit. A lot of tuning done by DB people.
    - Has to be measured properly once tuning is over
    - No fear so far



# Outline

- **Scope and constraints**
- **Short history**
  - SHIFT, CASTOR 1, CASTOR 2
- **Architecture overview**
  - Breakdown, DB centric
  - Core framework
- **Deployment**
- **Performances**

# Hardware used for Castor

- **Anatomy of a Castor-2 instance**
  - cluster of headnodes *to run main services*
  - lots of diskservers *provide disk cache, grouped in pools*
  - two database servers *Stager and DLF Oracle databases*

- **Shared infrastructure *both for Castor-1 and Castor-2***
  - nameserver cluster *database server + 4 CPU servers*
  - admin cluster *4 CPU servers*
  - tape libraries *robots, drives, servers, media*

# Hardware used for Castor

- **Today**
  - 6 Castor-2 instances
    - 4 LHC experiments, SC4, ITDC
    - 400 diskservers, ~2,2 PB
    - 40
  - 18 Castor-1 experiment stagers
    - Compass, NA48, LEP, Harp, Ntof, public, lhcb, …
    - 110 older diskservers, ~200 TB
    - plus ~20 infrastructure stagers…
- **Tomorrow**
  - more Castor-2 instances
  - fewer Castor-1 stagers…

# Scaling Castor-2 instances

| | May 2006 | | Sep 2006 | | Feb 2007 | |
|---|---|---|---|---|---|---|
| | *space [TB]* | *servers* | *space [TB]* | *servers* | *space [TB]* | *servers* |
| Alice | 78 | 20 | 231 | ~60 | 500 | |
| Atlas | 123 | 25 | 176 | ~45 | 370 | |
| CMS | 138 | 27 | 176 | ~45 | 370 | |
| LHCb | 121 | 26 | 188 | ~45 | 370 | |
| **total LHC** | **460** | **98** | **771** | **~180** | **1610** | **~480** |
| SC4 | 187 | 40 | | | | |
| ITDC | 169 | 42 | 169 | 42 | 170 | ~40 |
| public | | | | | ~200 | ~100 |
| **total** | **816** | **180** | **940** | **220** | **~2000** | **~600** |

*Experiment capacity should grow to 1.6 PB by Feb 2007, by adding ~300 servers*

***we will need to operate 600 diskservers***

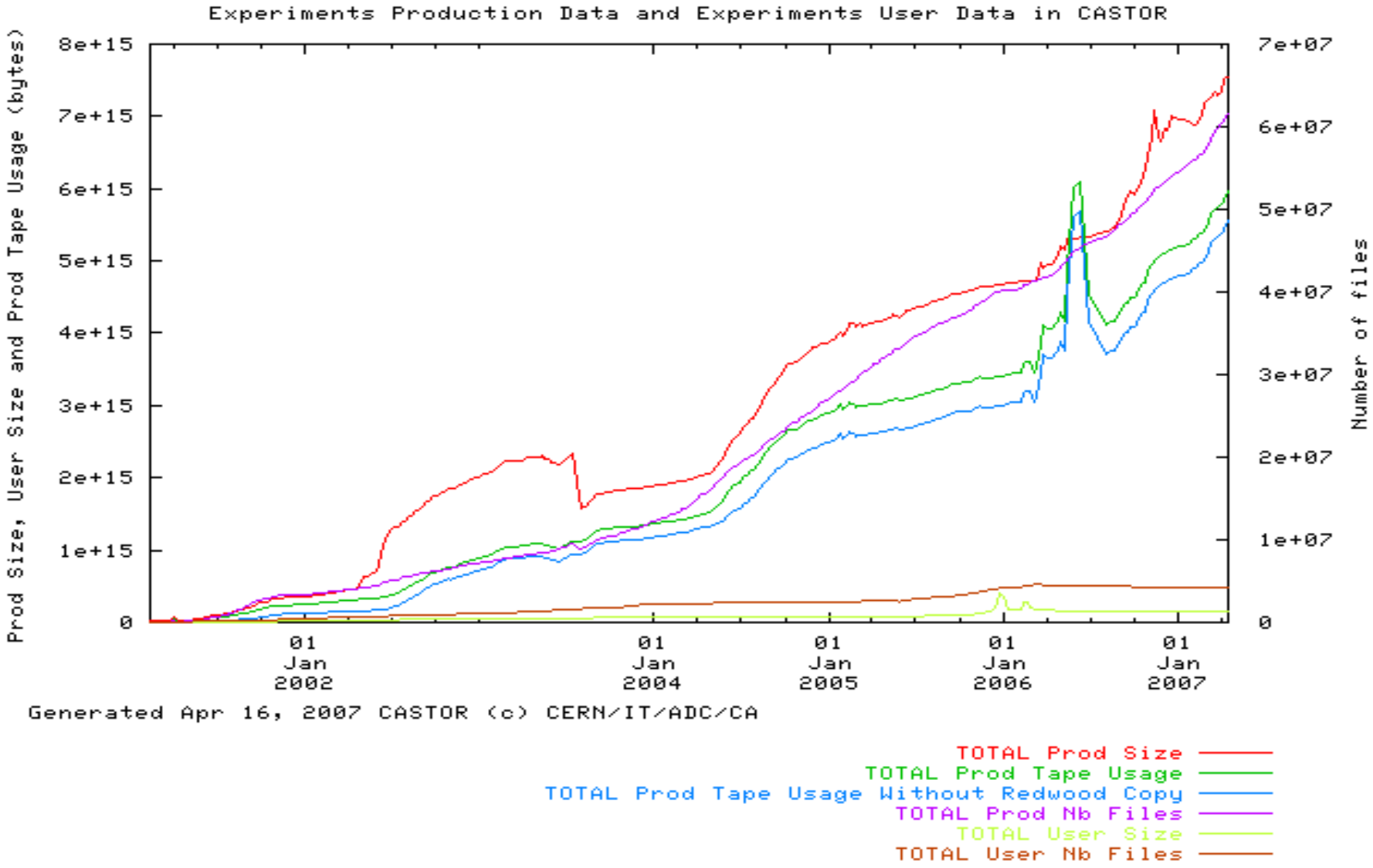
# Outline

- **Scope and constraints**
- **Short history**
  - SHIFT, CASTOR 1, Motivations for CASTOR 2
- **Architecture overview**
  - Breakdown, DB centric
  - Core framework : services, converters, multithreading
- **Deployment**
- **Performances**

# Performance measurement

- **Mainly in data challenges**
  - Service challenges are not really stressing CASTOR
- **Dedicated instance "ITDC"**
  - Production like
  - 40 disk servers, 30-40 tape drives, 200 CPU nodes as data source
- **Basic numbers / limitations**
  - 60 MB/s per tape drive (40 MB/s for LTOs)
  - 1 GB/s of bandwidth per switch
  - 100 MB/s of bandwidth per diskserver

# Latest Numbers

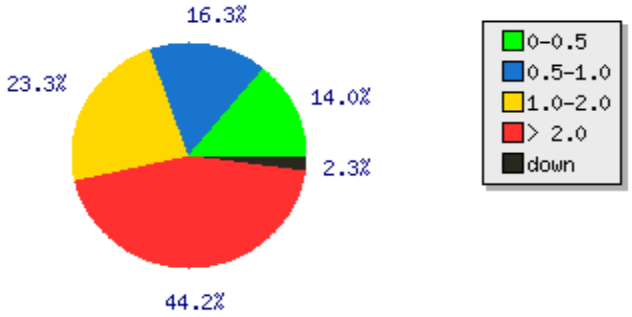


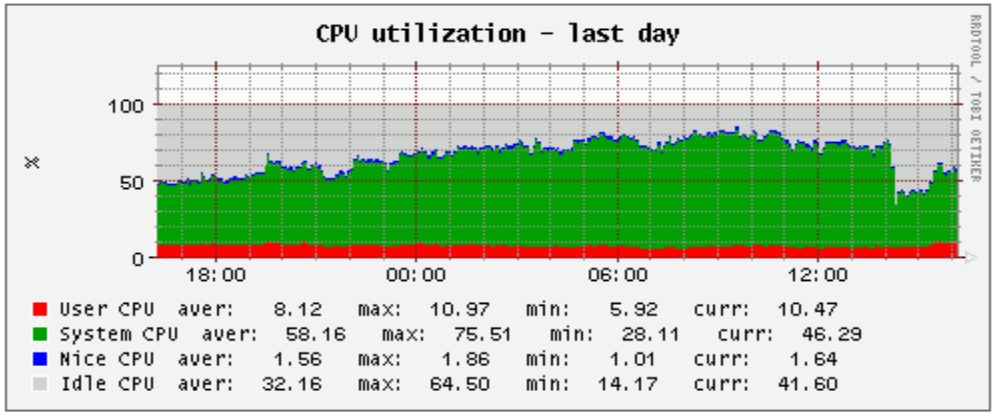
# SC4 throughput challenge

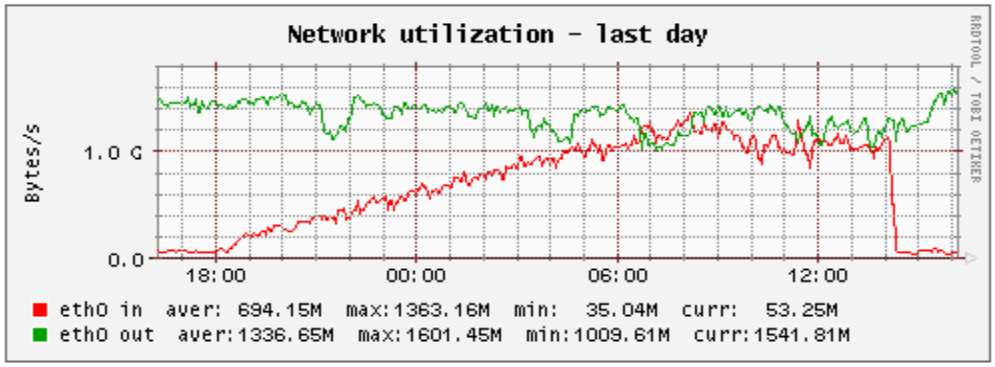
Cluster info: c2sc4 subcluster wan — 19 Apr 2006 Wed 16:14:27

| Cluster Information | |
|---|---|
| # of hosts (down): | 43 (1) |
| operating system(s): | 2.6.9-34.EL.cernsmp |
| # of CPUs (down): | 51 (2) |
| average up time: | 20 days, 18h:50m (boots per host) |
| hosts down: | lxfsra3004 |
| exceptions: | RPC_STATD_WRONG, MIRROR_BROKEN, |
| ITCM history | View template |
| Select from hosts: | None |
| Metric Distributions | Correlations |

Load Percentages




Last day | Get Fresh data | Get Detailed data | Auto Update

**1.3GB/s sent to Tier 1s**
**Added emulation of DAQ input streams without impact**

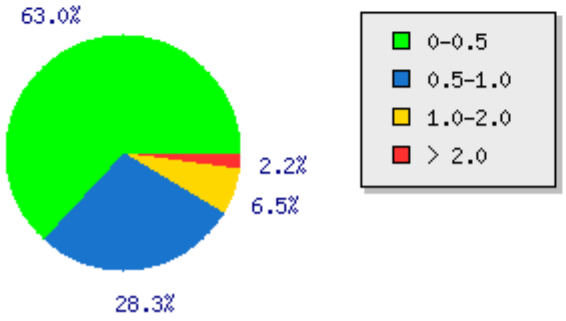
# ITDC : DAQ -> Tape

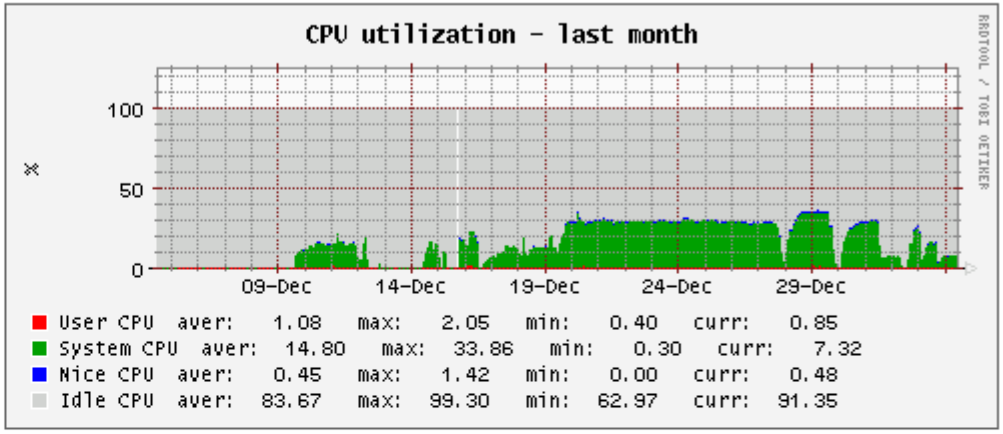
Cluster info: ITDC

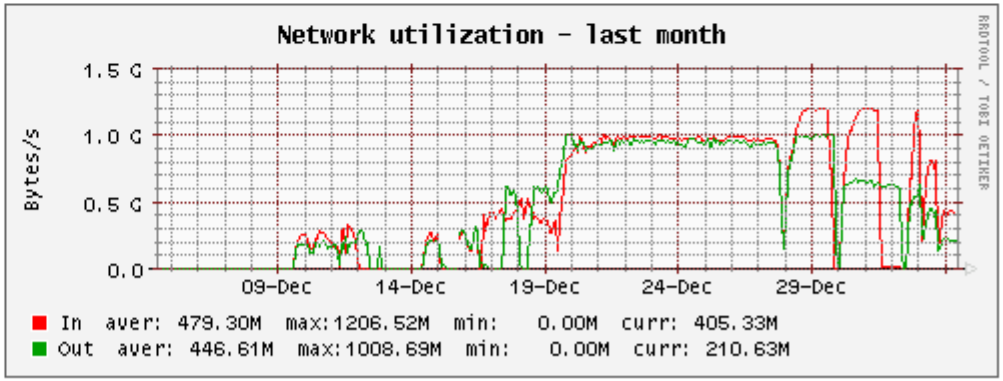
03 Jan 2006 Tue 10:19:12

| Cluster Information | |
|---|---|
| # of hosts (down): | 46 (0) |
| operating system(s): | 2.4.21-37.EL.cernsmp |
| # of CPUs (down): | 62 (0) |
| average up time: | 47 days, 14h:14m (boots per host) |
| hosts down: | none |
| exceptions: | FILESYSTEM_ERROR |
| ITCM history | View template |
| Select from hosts: | None |
| Metric Distributions | Correlations |

Load Percentages




Last month | Get Fresh data | Get Detailed data | Auto Update

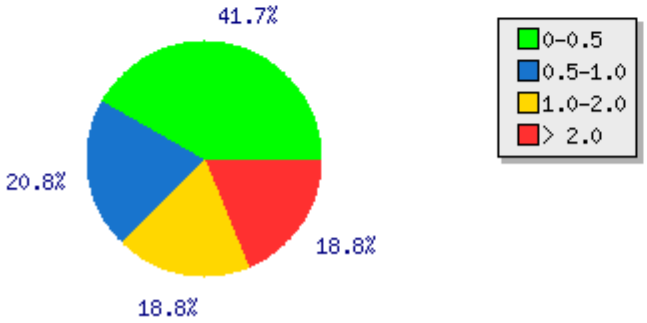
**Entering data to the disk buffer and writing them to tape**
**Stable running for one week at 1 GB/s**

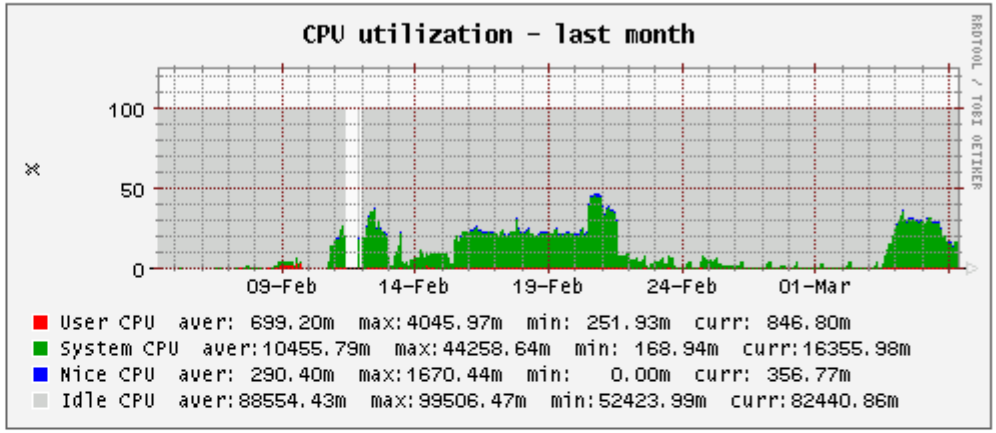
# ITDC : Tier 0 emulation

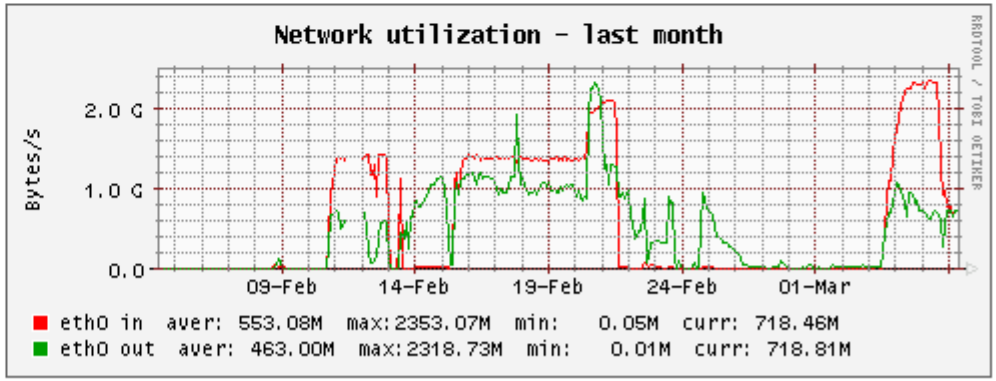
Cluster info: castor2 subcluster ITDC — 06 Mar 2006 Mon 08:15:45

| Cluster Information | |
|---|---|
| # of hosts (down): | 48 (0) |
| operating system(s): | 2.4.21-37.EL.cernsmp, 2.4.21-37.0.1.EL.cernsmp |
| # of CPUs (down): | 88 (0) |
| average up time: | 38 days, 7h:43m (boots per host) |
| hosts down: | none |
| ITCM history | View template |
| Select from hosts: | None |
| Metric Distributions | Correlations |

Load Percentages




Last month | Get Fresh data | Get Detailed data | Auto Update

**Emulating DAQ + Reconstruction + Export to Tier 1s + Tape writing**
**2.1 GByte/s input data rate + 2.2 GByte/s output data rate sustained 24h**

# Conclusion

- **CASTOR 2 is still 'young' (first use 2 years ago)**
  - not fully mature and debugged
- **But already widely used and quite performant**
  - SC3, SC4, data challenges
  - LHC experiments, CNAF, RAL , PIC, ASCG
  - 4.3 GB/s, 5M files staged, 30K concurrent requests
- **Scalable architecture**
  - DB centric, stateless daemons, no single point of failure

# Questions ?

**Related Info Pages:**

http://castor.web.cern.ch/castor/
http://it-dep-fio-ds.web.cern.ch/it-dep-fio-ds/services.asp
http://sls.cern.ch/sls/service.php?id=CASTOR