# CASTOR 2 in SC 3

## Operational aspects

Vladimír Bahyl
CERN IT-FIO

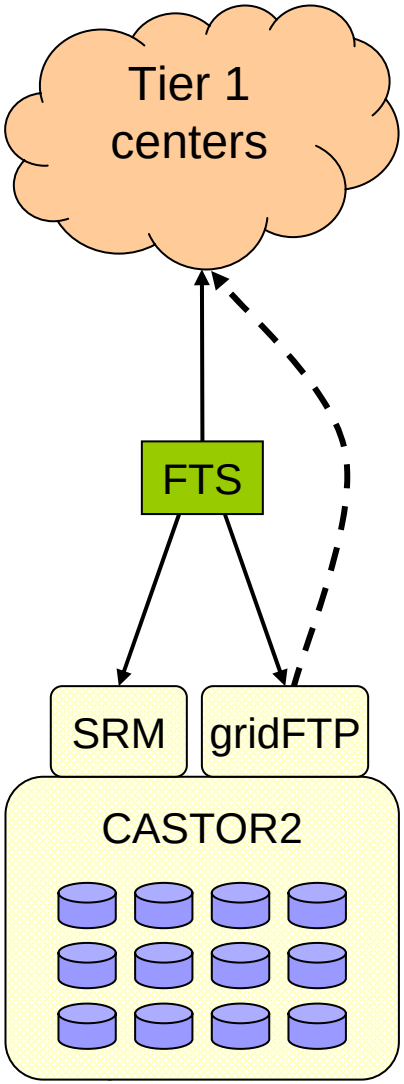
# Outline

- CASTOR2 and its role in SC3
- Configuration of disk pools
- Issues we came across
- Making it a service
- Conclusions

# What is CASTOR2 ?

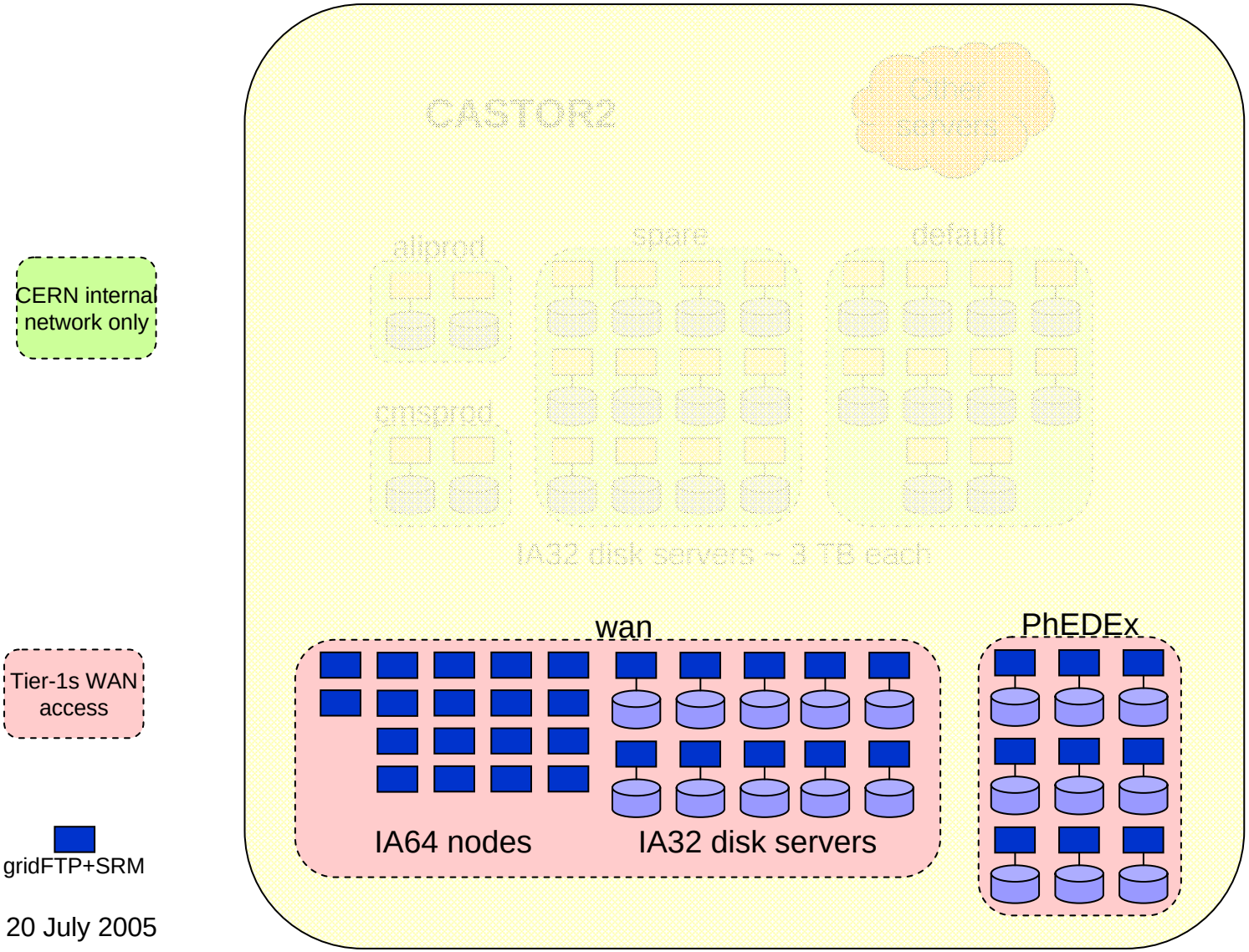
- Complete replacement of the stager (disk cache management)
- Database centric
  - Request spooling
  - Event logging
- Requests scheduling externalized (at CERN via LSF)
- More interesting features:
  - Stateless daemons
  - Tape requests bundling
- Scalable (tested by Tier-0 exercise)
- Central services stayed unmodified:
  - Name server, VMGR, VDQM, etc.
- More details:
  - Talk of Olof Bärring at PEB meeting 7th June 2005

# Role of CASTOR2 in SC3

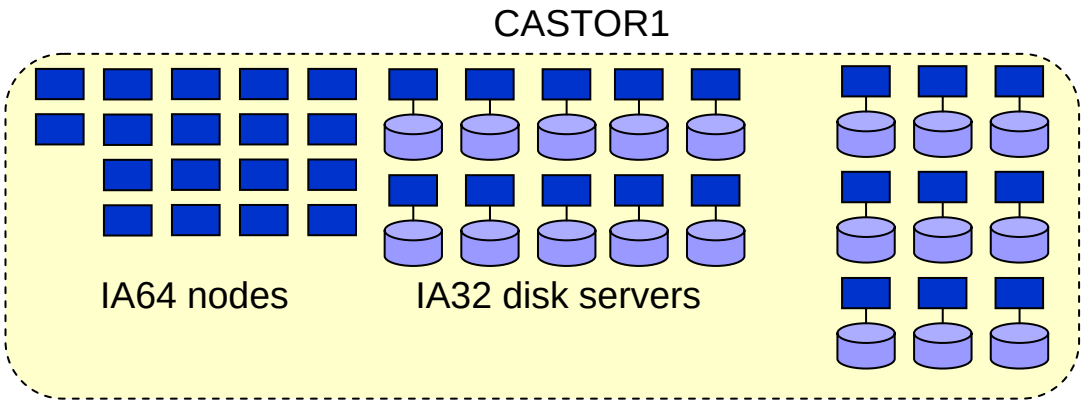



- CASTOR holds the data hence it is a founding element in the transfers
- FTS does the negotiations
- Transfers done over the gridFTP protocol as 3rd party copy

# CASTOR2 disk pool configuration

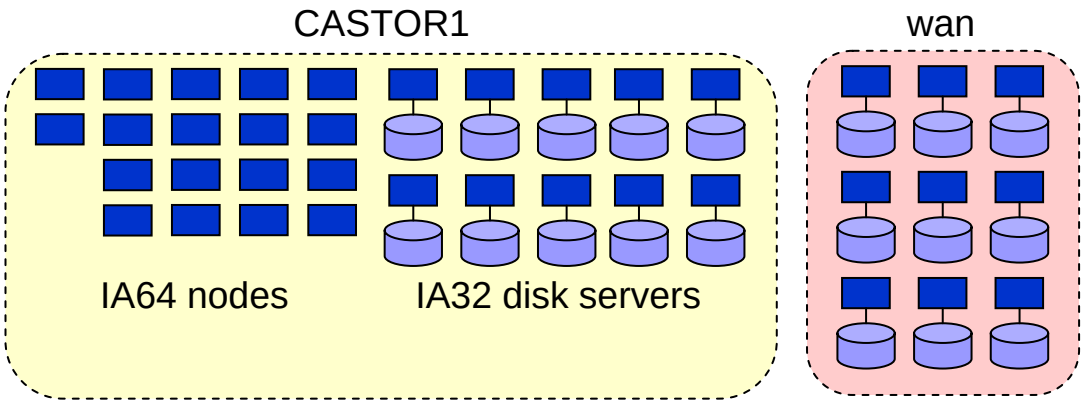


# WAN service class evolution 1/4



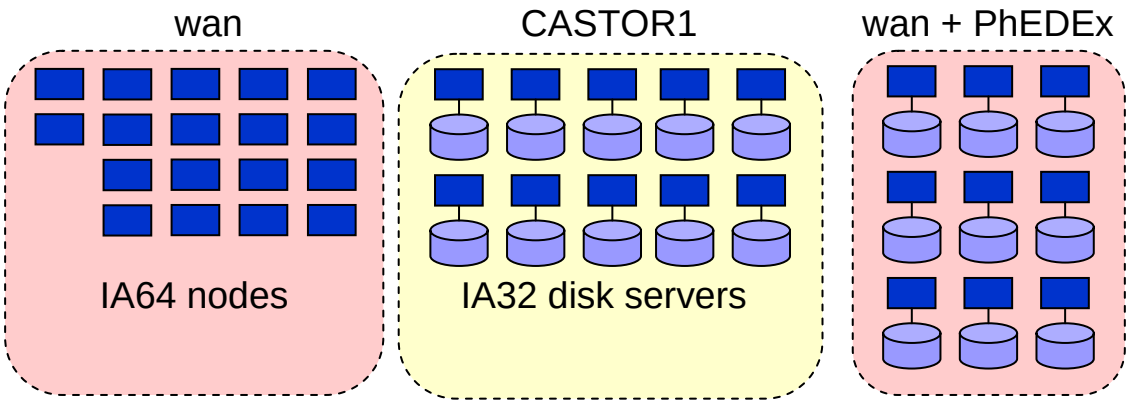

1. All nodes were in CASTOR1 disk pool with a dedicated stager.

# WAN service class evolution 2/4



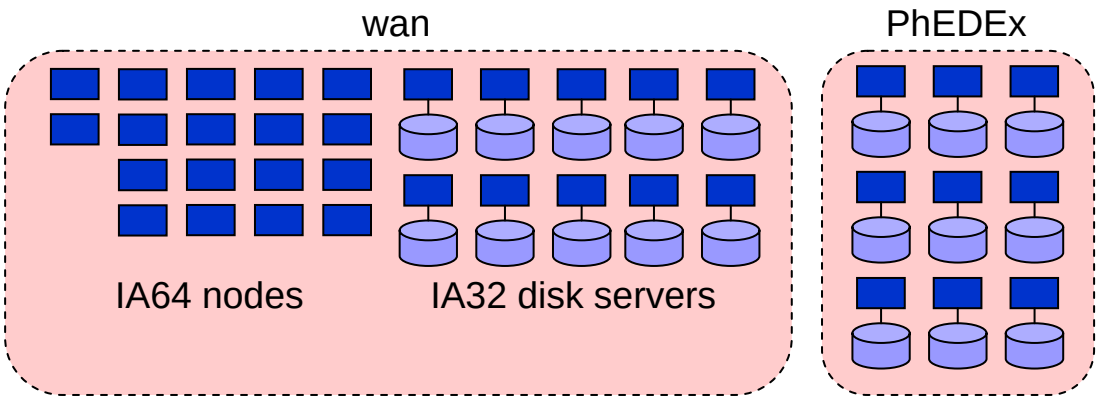

2. Move of 9 nodes to CASTOR2.
3. Creation of the WAN service class.

# WAN service class evolution 3/4




4. Introduction of IA64 oplaproXX nodes into the WAN service class
5. PhEDEx joined the party.

# WAN service class evolution 4/4




6. Separation of PhEDEx and WAN service classes to prevent interference.
7. Elimination of the need for CASTOR1.
8. Merge of remaining IA32 CASTOR1 nodes into the WAN service class.

# Issues confronted – external

- Load distribution with SRM
  - Previously, SRM returned static hostname (which was supposed to be a DNS alias) in TURLs
  - Now it can return hostname of the disk server where the data resides
  - It can also extract hostnames behind a DNS alias and rotate over the given set in the returned TURLs
    - Good for requests of multiple files
- Support of IA32 and IA64 architectures
  - Reason to use IA64 was to compare different architectures doing high performance transfers
  - Compiling CASTOR2 for IA64 architecture uncovered missing include files and type inconsistencies in the C code
  - Installation procedures had to be extended
  - IA64 hardware support not of a production quality

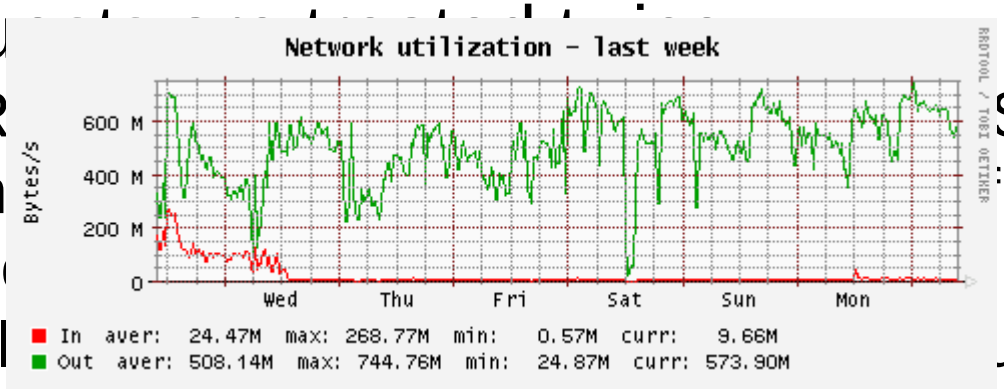
# Issues confronted – internal 1/2

- CASTOR2 LSF plug-in problems
  - Selects candidates to run jobs based on external (CASTOR2 related) resource requirements
  - Initialization occasionally fails = system halts
  - All incidents understood (caused by misconfiguration)
  - Operational procedures put in place ☺
- Waiting for tape recalls
  - SC3 is run with disk only files
  - If a file is removed, system will look for it on tape
  - The fact, that the file is not on tape was not correctly propagated back to the requester
  - Fixed promptly on the database level ☺

# Issues confronted – internal 2/2

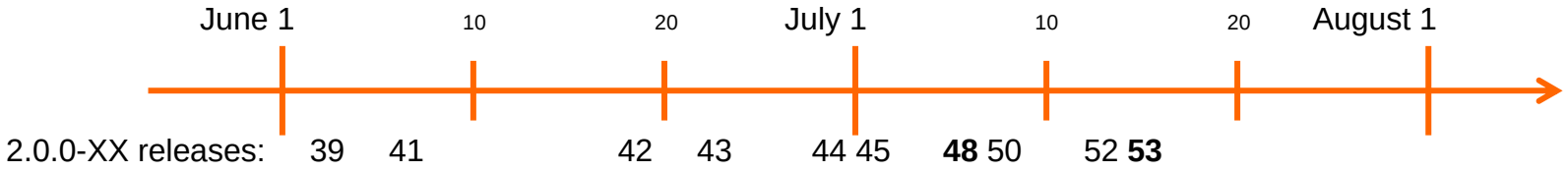
- Replication causing internal traffic
  - SRM returns exact location where the data is
  - gridFTP calls stager_get() (via rfio_open())
  - Scheduler returns a replica ☹
  - Not an issue as this is an unlikely usage pattern

- Requ
  - SR SF job
  - On io_open()
  - An
  - Wil ully supported



# Release marathon

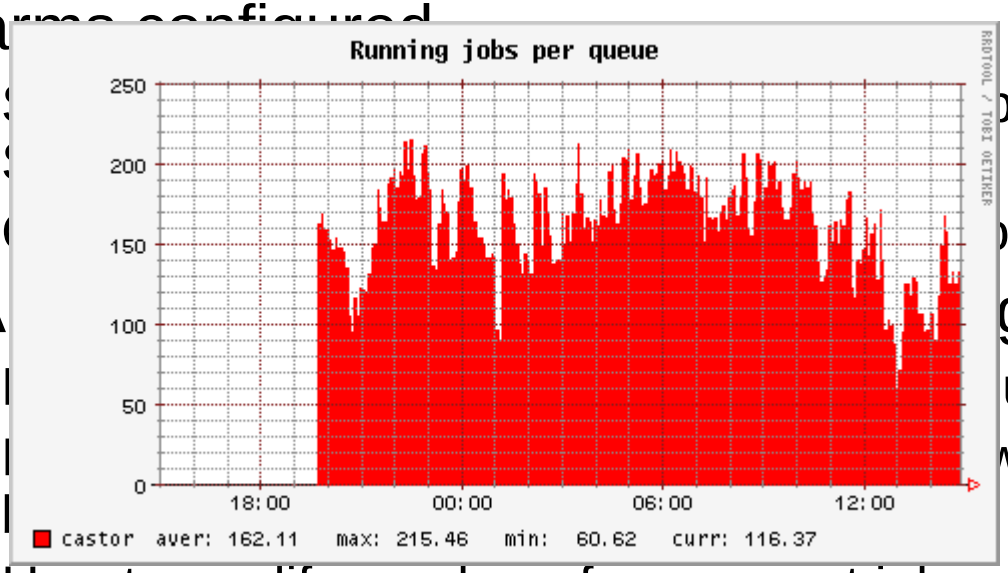
- Circa 2 revisions per week
- Schedule:




- Deployment point of view – dynamic system
  - Packaging level changes:
    - Structure of the configuration files
    - Names of the startup scripts
    - RPM dependencies are modified
  - Database bottlenecks are being cleaned up
    - Function based indexes configured
  - LSF configuration changes required
    - Removed *eexec* script

# Making it a service

- Monitoring enabled
  - New daemons on the server nodes as well as disk servers
  - Check the log files for new patterns
  - CASTOR LSF status integrated in Lemon
- Ala
  - S … by the Operators &
  - C … pers 24x7 ☹
- CA … g drafted
  - I … uction ?
  - I … w to move disk servers
  - How to modify number of concurrent jobs running on a disk server ?



# Conclusions

- SC3 is proving valuable in shaping up CASTOR2
- CASTOR2 as it is now has not been fully optimized for SC3 usage pattern
  - Disk only file copy configuration is unlikely to be the mainstream usage pattern once LHC starts
  - Native support for gsiftp protocol not yet built in (only ROOT & RFIO supported for now)
    - Replications of hot files are affected
    - Internal transfers over the loopback interface exist
  - → This in fact doubles the load on the system (which in itself is good so that we can find and fix the problems)
- Recommended CASTOR2 hardware setup has proved suitable for the SC3 tests
  - See my talk from 17th May 2005 at GDB and check 4 scenarios
- CASTOR2 can throttle requests per hardware type (in fact per machine)
  - Something that CASTOR1 stager couldn't

- None of the issues mentioned above is a show stopper and no major problems are expected ahead
- For the tasks required by the service phase of SC3, we have already reached production quality and are confident that we can handle it

# Thank you

- Vladimir.Bahyl@cern.ch