Data & Storage Services

X.Espinal, J.IVEN, L.Mascetti, A-J.Peters
Contact: jan.iven@cern.ch

Di - EOS
stributed

running EOS across 2 computer centers

CERN IT Department

Background

CERN Meyrin computer center is limited to 3.5MW

- No extension inside CERN (financial & political)
- Call for tender, won by

  Wigner Institute / Budapest / HU
- 2x 100Gb/s links = plenty of bandwidth, redundant

Benefits

Can double CERN computing capacity (+2.5MW)

- Allows for business continuity even on major mishaps
- Free trips to Hungary (not - all managed remotely..)

Timeline

- During 2012: construction work
- Spring 2013 : network and first hardware
- Autumn 2013: first production services at 10%

EOS* at Wigner

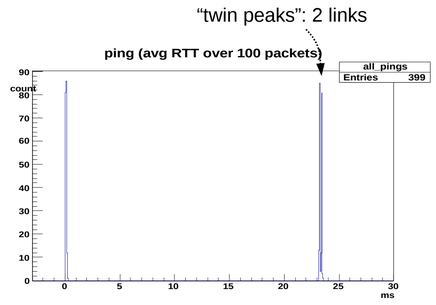
*) EOS: XRootD-based disk-only storage system developed at CERN - http://eos.cern.ch

Same:

- Hardware (split delivery for single order)
- Software (SLC6, EOS/xrootd)
- Configuration (same puppet hostgroup)
- Tools (same `new` "*agile*" toolchain..)

Different:

- *(remote) operations team*
- *LATENCY: 23ms*



Network inside each site is "flat"

Service-internal tests:

✔ Functionality: OK

✔ Internal performance: rate-limited anyway

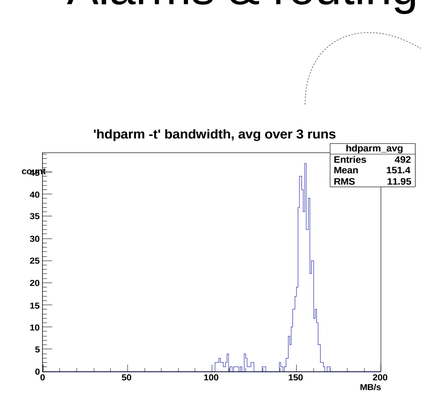
- Rebalancing,
- Draining,
- Consistency checks

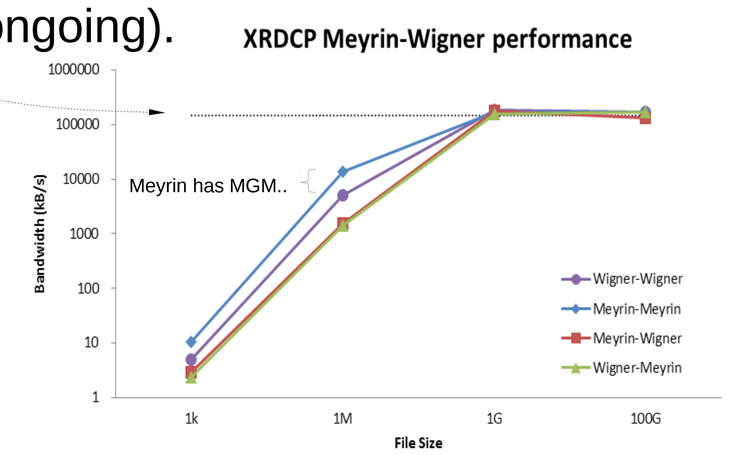
Review & test workflows

✔ Installation + EOS auto-registration

✔ Manual remote reboot, remote console

- Various repairs (ongoing)
- Alarms & routing (ongoing).





Single Stream transfers capped by single disk speed.
Remember: EOS scales via concurrency

Performance Impact?

Initial (Gu)es(s)timates:

✔ Low impact for streaming data

- TCP windows; capped by single disk speed
- Most writes in this category, & capped by *slowest* disk

✔ Low impact for short data transfers:

- anyway limited by disk seeks & overhead

▼ Moderate impact for metadata-only operations

- Assume: intense activity is close to MGM

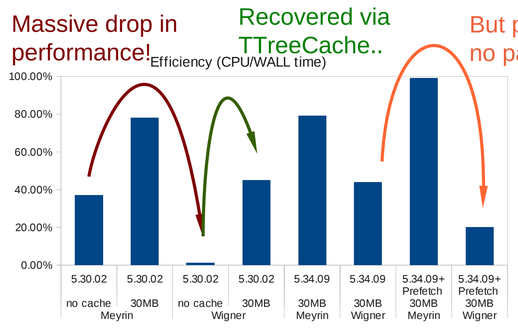
✘ **high** impact on repeated+small+direct-I/O (worst-case)

- TCP windows don't help for small transfers
- n × Δ-latency
- ("real" jobs would compute something..)

Massive drop in performance!

Recovered via TTreeCache..

But prefetch is no panacea.



ROOT reads 10% random subset of events; client in Meyrin, data from Meyrin or Wigner; without/with ROOT TTreeCache

Countermeasures

Local caching – experiments' decision

- ROOT TTreeCache recovers most of the lost performance even for "remote" access. Default in ROOT-6..

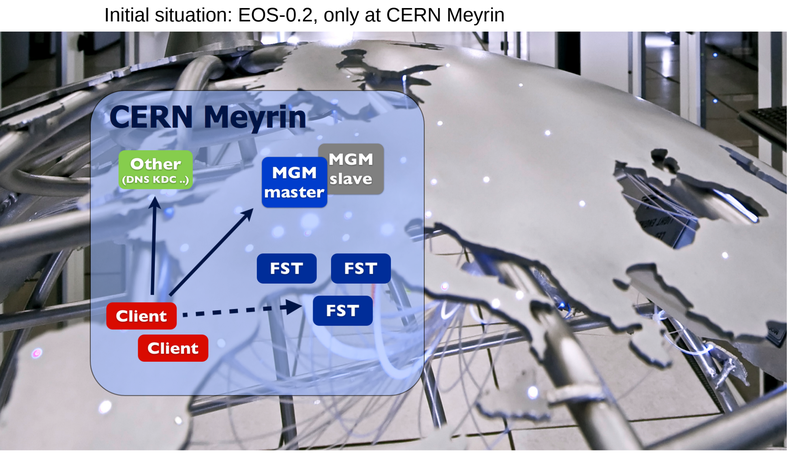
Data locality and GEO-Scheduling (EOS-0.3)

- Place file replicas "far apart"; on access prefer closest replica
- can **completely hide** data read latency once `size(Wigner)` ≳ `size(hotdata)`

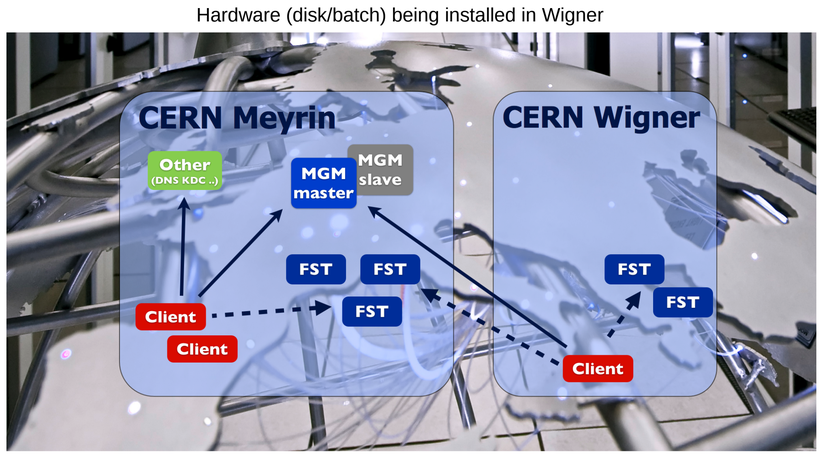
Per-site replicated services (EOS-0.3)

- Clients talk to local MGM (readonly, writes go to master) – no more penalty for metadata (read) operations

Initial situation: EOS-0.2, only at CERN Meyrin



Hardware (disk/batch) being installed in Wigner



Goal: fully location-aware, replicated service