Quantitative storage analysis

Dirk Duellmann, CERN IT

Federated Storage Workshop,
21. April 2014, SLAC

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

DSS Statistical log analysis @ CERN

CERN IT Department

- Data sources
  - EOS logs per file access
    - some 20 metrics per open/close sequence
      - very similar to xroot f-Stream (collected by Matevz)
  - LSF - another 20 metrics per job
    - cpu, wall, os, virt/phys, RSS, swap, local I/O
  - Dashboards
    - experiment side classification and (in some cases) event rate
- Goals
  - can we see / quantify effects of main system level parameters ?
    - network distance, OS, virt/phys, Intel/AMD, TTreeCache use etc.
  - predict impact of an increased fraction remote/federated access
  - measure correlation / causal relationship between IO and CPU usage
  - note: this is done passively (no probe jobs) -> describes real job mix
- along the way..
    - spot unintended use, cost of internal balancing/draining
    - statistically detect component malfunctions
    - aggregates per server, protocol, buffer size etc.

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

CERN

DSS

Plots are easy… Analysis & conclusions can be work

CERN IT Department

- Many cross-checks could be made
  - but only after basic understanding and relation between IO and CPU and user metrics is established
- Some non-trivial aspects
  - cross-matching (eg based on user and job identity) between different monitoring sources is complex and can currently only be done for a subset of all jobs
  - OS / virtualisation / location are already correlated in large scale setups
- Popular metrics can be tested for their predictive value
  - For example: CPU/Wall = “CPU-efficiency”
    - or is it just how CPU-bound a particular job/IO mix is?
  - CMS/ATLAS jobs running from Wigner show significantly increased “CPU-efficiency”
  - …but, is this good or bad news?
    - hint: LHCb and Alice show decreased values

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

CERN

A proposal to experiments

- Event/s is the relevant metric for optimisation …
  - … but only well defined within one job type
- Short term:
  - check if evt/s and “CPU-efficiency” are correlated in a way which is useful for optimisation
  - for a “representative” workload compare both metrics
    - RH5 <-> RH6 (phys)
    - RH6(phys) <-> RH6 (virt)
    - RH6(virt @ Geneva) <-> RH6 (virt @Wigner)
- Maybe these can be extracted from existing/new HammerCloud runs

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Fully Remote Access Impact?

- One example from new LSF low-CPU-usage probes
  - one user with often only 2-3% CPU utilisation
  - 60-100kB/s average transfer rate
    - CPU@CERN - all data at US-T2
- After vector-read/TTC was enabled
  - improvement by factor 4.5 in turn-around and CPU utilisation remaining factor 6 wrt local EOS access
- "Visible" impact on users is mild (if sufficient batch slots are there)
  - even slow jobs are running stable in parallel
  - in many cases: no concrete expectation about what their job duration / transfer speed should be
- Impact on infrastructure
  - may rise: low CPU jobs block memory and hence other slots
  - not likely to improve with VM based CPU scheduling
- => low CPU & low throughput & no TTC
  - is easy to spot & warn about with existing monitoring

"Federation" Impact on CERN EOS (CPU outside)

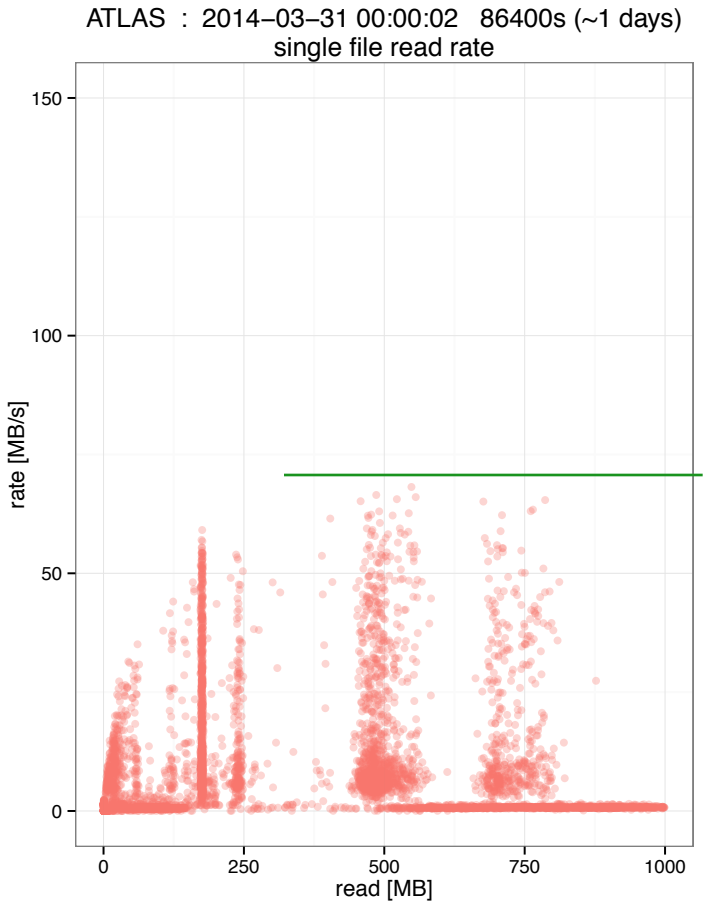
- How much is EOS used as storage fall-back?
  - numbers: less than 1% of opens / read-bytes for ATLAS and CMS
    - very low fraction
    - no visible impact on storage or network
- ALICE
  - one quarter of all opens and bytes read
    - not strictly xroot federation, but remote access from outside
    - may include some data distribution
- LHCb
  - no external user access

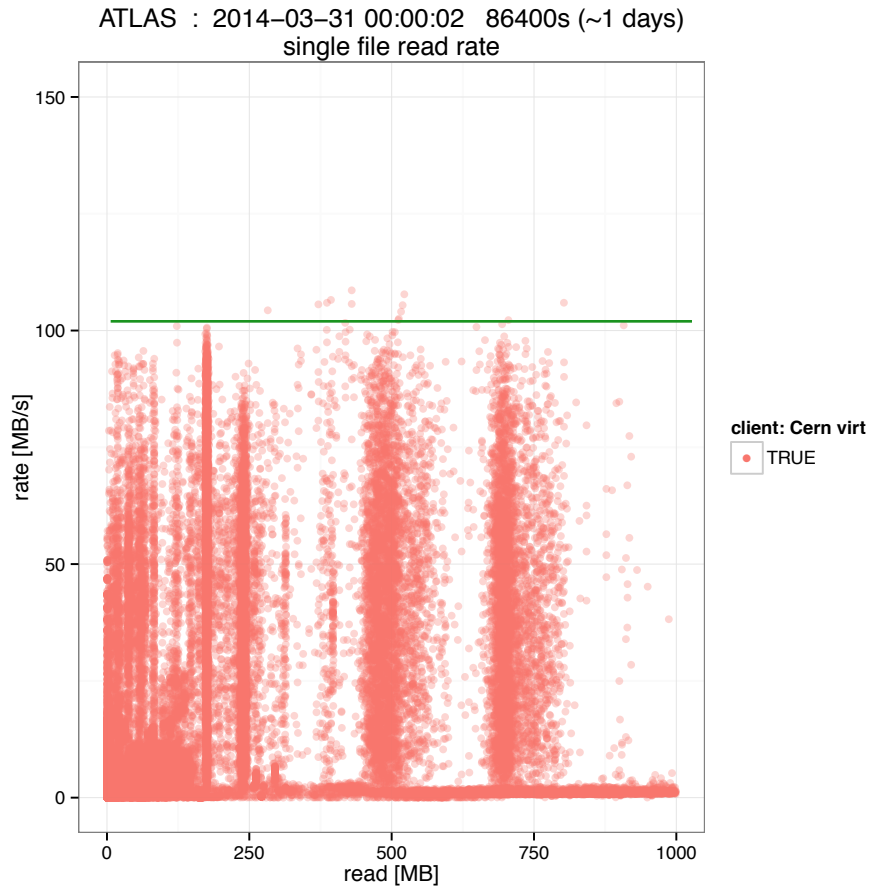
"Federation" impact on CERN CPUs? (data outside)

- How much are CERN jobs using outside storage?
  - data from March FAX/AAA logs
- ATLAS
  - very small fraction
  - > 99% Ilija's test jobs
  - average rate: 270 kB/s (but, again: test jobs)
- CMS
  - some enthusiasts
    - 1.6 million accesses by one user to one US site
    - 1.3 CPU years with 5.5TB transferred
  - normally: O(1k-10k) accesses to other sites
  - average rate: 55 kB/s
- ALICE & LHCb
  - no xroot monitoring data available via dashboard

Network distance to Wigner: Can we measure / predict the impact ?

- Wigner: RH6(virt) <-> Geneva: RH6(virt)





CERN IT Departmen
CH-1211 Genève 2
Switzerland
www.cern.ch/it

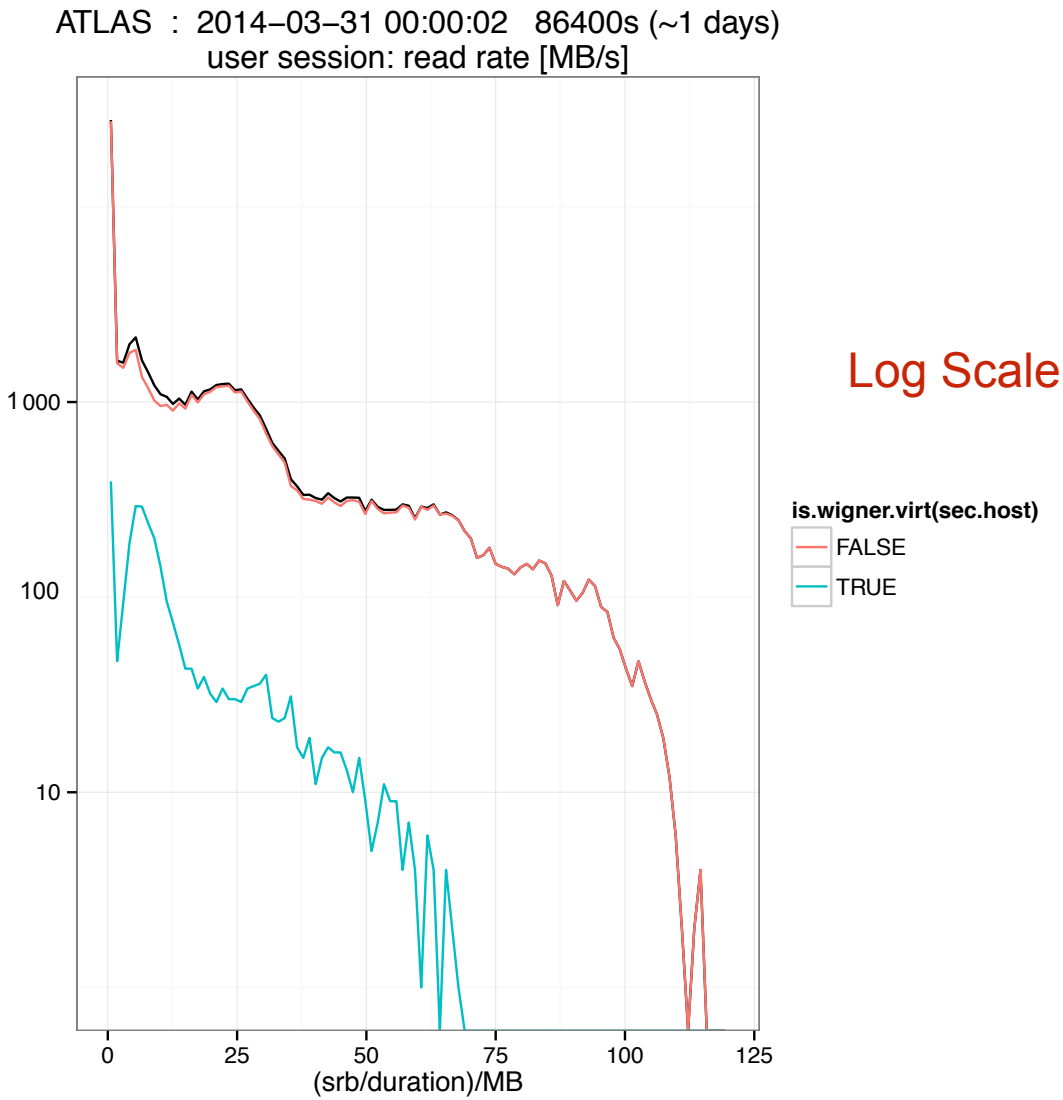
Is this expected? important?

- Single stream max bandwidth
  - depends on RTT, link speed, tcp buffer size
    - well known, but less well controlled
  - …and packet-loss frequency (if any)

- Ramp-up wrt transferred volume
  - can be passively measured (eg from copy jobs)
    - complementary to perfSONAR
- Fitting the expected function(RTT) allows to detect low-level packet drops / low buffer size

- Is the impact currently important for real performance (evt/s) ?
  - unlikely, as job profile is mainly CPU-bound
    - bandwidth is constrained by ROOT unpacking/decompression
  - this may change if s/w efficiency (or copy job fraction) should go up

Same data - rate distribution [MB/s]



Log Scale!

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

A proposal for the xroot side

- Correlating CPU and IO metrics is still difficult
  - no common id to match file access with batch metrics
  - soon a set of virtualisation platforms will be used
- Proposal
  - add xroot client plugin to collect cpu and user metrics per session
    - correlated (by session id) with f-stream
      - eg: cpu / wall / memory / swap / local IO
      - event rate from ROOT / exp. framework
  - define app info field for main workloads
    - Eg: app_info := “app/version/specific_setting”
    - EOS internally uses: /eos/gridftp, /eos/balancing, /eos/draining, etc
- This would allow passive analysis of experiment throughput per ‘app’
  - statistical comparison of sites/os/sw_versions…
- One would not be able to average evt/s over different apps
  - but one would be able to sum-up relative increases/decreases

Summary

- Impact of remote access (and much more) can be extracted statistically from existing log data
  - taking CPU/memory constraints into account
  - Wigner impact is smaller than the one of RH6 and phys/virt moves
- Remote (WAN) scenarios need closer tracking of TTC/vector-read usage (also for vanilla ROOT users)
  - correlated logs are sufficient to spot candidates
    - but not always to identify affected users / apps
- Popular use of CPU/Wall-ratio for evaluating optimisations may be a dangerous shortcut!
- Proposed xroot client plugin would allow to obtain correlated results for user, CPU and IO metrics
  - in an extensible and infrastructure neutral way