Performance metrics and measurements in the Data Lake mode

Jaroslava Schovancová on behalf of the WLCG Data Lake R&D group
CERN IT

Motivation

- HL-LHC storage needs are above the expected technology evolution (15%/yr) and funding (flat)
- Now it is time to focus on optimising the cost
  - including operational cost
  - During Run I and Run II the primary focus was to make things work, and then make things work reliably
- WLCG Data Lake R&D project: we are trying to understand if distributed storage saves cost

Distributed storage: concept

- distributed storage
- network links
  - latency, bandwidth
- storage media
  - disk/cache/tape
- evolving data access protocols
  - driven by the changes in networks
- evolving inter-storage communication

The core metric: event throughput

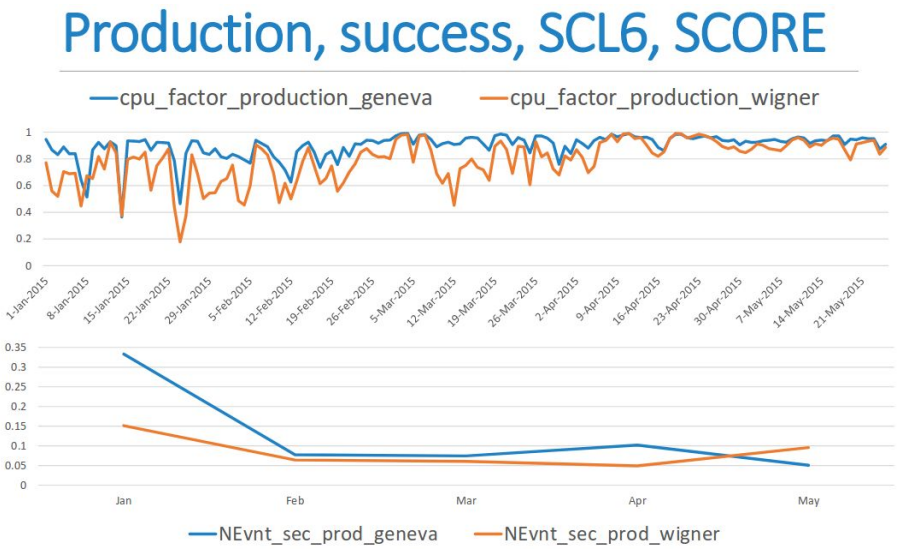
- the Compute side of things ⇒ all boils down to the **event throughput at the same cost**

  ⇒ Are we able to support the same or even better event throughput at the same cost with the evolving storage configuration?

- Easier said than done!
  - Which events? Which SW? How much I/O? How much memory? ...
- This talk is about methodology.
  - How to measure job performance? Storage performance?
  - How to benchmark?
  - What to take into account for the storage configuration?
  - Topology of resources? its transparency?
  - (Co-)location of data vs. compute resources?
  - Types of storage media vs. access policies?
  - Direct vs. remote access to data?
  - How to evolve tools to support the core mission

History: distributed Tier-0

Similar study, at smaller geo-scale 4 yrs ago: CERN multi-site Tier-0 and job efficiency study https://indico.cern.ch/event/302033/



✔ no indication of any significant difference of job efficiency between CERN Geneva and Wigner. https://indico.cern.ch/event/393550/

- for both analysis and production jobs
- Study carried out by Alessandro Di Girolamo, Edward Karavakis, Valentina Mancinelli, Maarten Litmaath

Measurements

- Methodology, how to measure and benchmark
- What to measure: event throughput
  - I/O rate
  - Stage-in / Stage-out time
  - SW init time
  - Time spent in event loop
- Production and Analysis workflows
- Core count preferences: MCORE (production) vs. SCORE (analysis)
- Local vs. remote data access

Benchmark

- Resources: standard storage vs. distributed storage
  - can compare these flavors of resources
  - in different configurations of the **distributed storage**
    - hot/warm/cold storage
    - caching
    - local vs. remote access
    - data replication policies/striping
    - downtime/recovery of subset of storage resources
  - benchmarking per resources, VM

⇒ study and benchmark both

- **job performance**, and
- **distributed storage performance**, at once

Workflows types

- Assumption: what works for ATLAS and CMS, works also for ALICE and LHCb
  - a reasonable assumption for ALICE
  - LHCb is sufficiently small, inefficiencies will not matter on the global scale
- Identified several typical workflows for ATLAS and CMS, with different demanding parts
- These workflows will serve for measurement of job/storage performance and benchmarking

Workflows types - ATLAS

- G4 simulation
  - CPU intensive, not so much RAM demanding, not much I/O intensive
  - ttbar full simul, reference workflow to compare HS06
- Digi+reco
  - some I/O (not that much IOwaits for jobs), RAM-demanding, sensitive to latency
  - Event mixing, digitization, trigger, trigger reconstruction
  - 50 GB in
- Production derivation
  - More I/O intensive
  - Skim, slim, …
  - 5 GB in
- Analysis - focusing on analysis derivation

✔☐ Recent developments/evolution: digi+reco and production derivation in HC, JS

Workflows types - CMS

- Understanding the equivalents
  - G4 simulation: quick
  - Reco takes more time
  - Premixed pile-up
    - CMS pre-mixes min bias ⇒ huge files, less copies. Perhaps lower I/O?
    - ATLAS does not pre-mix min bias ⇒ smaller files, more copies
  - No derivations
  - Analysis
- Production workflows in CMS: leverage the “1-chain” job https://doi.org/10.1007/s41781-017-0001-9
  - Generation - Simulation - Digitization - Reconstruction steps in 1 job, to save data stage-out and stage-in among jobs
    ⇒ very small input and 1 output of the full chain

✔ ☐ Recent developments/evolution: production jobs in HC CMS, Andrea Sciaba

✔ ☐ Recent developments/evolution: “1-chain” production jobs in HC CMS, Andrea Sciaba

Prototype setup - the sites

ATLAS

- CERN
  - distributed data centre between Meyrin and Wigner
- Nikhef & SARA
- LMU MUC
- RU Kurchatov
- Australia-ATLAS
- BNL
- Tokyo

CMS

- CERN
- JINR
- ...

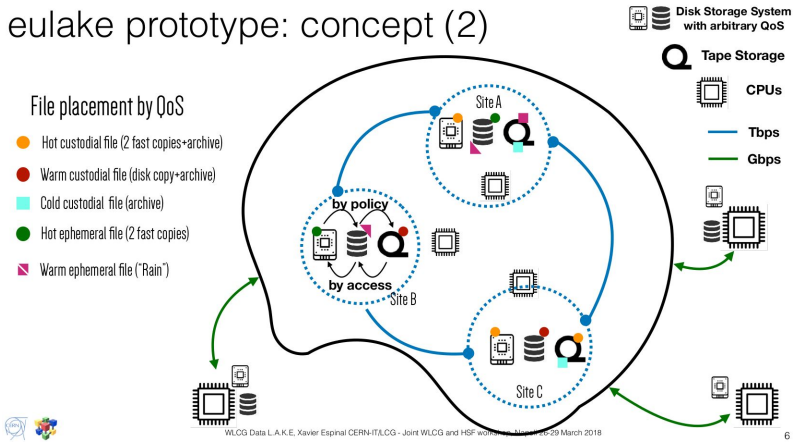
Data access modes

- ATLAS: copy to scratch vs. directIO from co-located storage vs. read over WAN
- CMS: remote read

ATLAS

storage vs. compute	Data access mode	Standard storage	eulake
co-located	copy to scratch	✔ ☐	✔ ☐
	directIO	✔ ☐	✔ ☐
not co-located	copy to scratch	?	✔ ☐
	directIO	?	✔ ☐

eulake prototype: concept (2)



by Xavier Espinal

CMS: investigation of data access modes ongoing

Zooming in on **eulake**

- Location of files is a part of metadata
- Want the DataLake to be “transparent” w.r.t. data location
  - however, ... for measurement/benchmarking we need to know the exact location of the data files ⇒ need for a information system capable to describe the complex topology, to configure the resources
    - Many thanks to Alessandro Di Girolamo and Alexey Anisenkov for their help with a eulake endpoint configuration in the information system!
- Small scale manual test, to get a “look and feel” how eulake works: ls, upload, download, delete
- Small scale test from within the job: ls, upload, download, delete
- Benchmark job performance w.r.t. different data access modes and workflow types, w.r.t. “standard sites with standard data access modes”

Performance metrics and measurements in the Data Lake mode

- Trying to understand if distributed storage saves cost
- With any distributed storage, we can study, measure, and benchmark
  - jobs and distributed storage performance
  - with different workflows
  - w.r.t. different data access modes

⇒ *Can we hide latency and average out bandwidth so that the data location becomes irrelevant?*

Jaroslava Schovancová on behalf of the
WLCG Data Lake R&D group


CERN