System Performance and Cost Modelling in LHC computing

Andrea Sciabà
on behalf of the HSF/WLCG Systems performance and cost modelling WG

CHEP 2018
Sofia, 9-13 July 2018
Despite ongoing efforts, we do not know yet how we will manage to process HL data with the expected levels of funding.

- 10x increase in trigger rates, NLO & NNLO, 5x increase in pile-up
  - The latter involves >> linear growth in reconstruction time
- Price/performance advances slowing down
  - 20% yearly gains are very difficult
- CPU and disk short by a factor ≈ 5
  - Assuming no “revolutionary” changes
- Strong need to quantitatively understand our efficiency and how we can optimise performance

**HL-LHC baseline resource needs (LHCC Sep. 2017)**

**CMS**

**HL-LHC new working numbers**

- CMS does not have newer officially blessed numbers for HL-LHC
- Still, work has been ongoing also due to the DOE request to US-ONS for long time planning
- Main changes wrt to older models (see for example ECFA presentation by S. Campana) are
  - Expectation of 10% y year performance improvement
  - Rely largely on miniAOD(SIM) for operations; AOD(SIM) an archival thing

Take home messages for 2027:

- 50% within CPU
- 30% disk
- 20% tape

Wrt to 2017, assuming a >20% y by Moore and friends, the excesses are ~1 for CPU, ~4 for storage
The Working Group

- WLCG and HSF joined forces to study how we can achieve a more cost-effective computing on the Run3/4 timescale
  - Start by developing a deep understanding of current workloads, resource utilization, and their impact on site costs
  - Proceed to explore future scenarios, estimate possible improvements in efficiency (in software, infrastructure and computing systems)
  - Develop tools and methods to do the above, that can be used in the community
  - At the same time, establish a “culture of performance”
  - Site cost cannot be compared but locally optimised
  - Active participation by experiments, sites and IT experts
    - Conveners: J Flix, M Schulz, A Sciabà
    - About 35 active members → wlcg-SystemsPerformanceModeling@cern.ch
    - Links with HEPiX benchmarking working group
    - Web site: https://twiki.cern.ch/twiki/bin/view/LCG/WLCGSystemsPerformanceModeling
    - Meetings: https://indico.cern.ch/category/9733/
Several goals have been identified for the short, medium and long term and some are well under way or even completed:

- Identify representative experiment workloads that can be run in a controlled environment and package them for easy distribution.
- Define which metrics best characterise such workloads.
- Set up a distributed testbed to run tests.
- Establish a common framework for estimating resource needs.
- Define a process to evaluate the cost of an infrastructure as a function of the experiment requirements.

**Roadmap [preliminary]**

**SHORT**
- One reference workload
- 1st set of meaningful metrics

**MEDIUM**
- 1st set of models
- Local costs to 2011
- Structure of the WG

**LONG**
- Expansion of 1st model
- 2 experiments
- Model(s) completion
- Most NEFs from all experiments

**2018**
- Structure of the WG
- Definition of sub-groups
- One reference workload

**2019**
- Expansion of 1st model
- Local costs to 2011

**2020**
- Model(s) completion
- Most NEFs from all experiments

**COMPUTING TDR**
- WLCG
• Identify the metrics that best describe a workload
  – To understand if the hardware is used efficiently → software experts
  – To quantify the resource utilisation on the node → site administrators
  – Record time series and extract summary numbers (averages, 95th percentile values, etc.)
• Started with an already comprehensive list of basic metrics
• Will expand / contract as needed – work in progress
• The goal is to have the smallest amount of parameters that describes as completely as necessary the workloads

### Current metrics

![Current metrics table with metrics like CPU, Memory, Network]
Metrics measurement

- PrMon is a tool to monitor resource usage of a process tree
  - Derived from the ATLAS MemoryMonitor
  - [https://github.com/HSF/prmon](https://github.com/HSF/prmon)
  - It includes most of the previously listed metrics (from /proc)
    - VMEM, RSS, PSS
    - rchar/wchar (bytes read/written by the process), read_bytes/write_bytes (bytes read/written from/to the storage layer)
    - User time, system time, wallclock time
    - rx_bytes, tx_bytes, rx_packets, tx_packets
  - Actively worked on
- Trident
  - Measures CPU, IO and memory utilisation based on hardware counters
  - Very detailed, almost no overhead
  - See Servesh’ and David’s poster “Trident: A three pronged approach to analysing node utilisation” ([link](#))
- Collection of reference workloads from the LHC experiments
  - Event generation, Geant4 simulation, digitisation, reconstruction, derivation steps
  - Local file access or remote access via xrootd
- Making power and complex tools accessible for users and site managers on all levels
PrMon monitoring plots: examples

ATLAS Digi Reco - memory

CMS DIGI - IO

ALICE sim+reco - Memory

CMS DIGI - Network
• Several metrics calculated
  – CPU: IPC, top-down analysis (time spent on front-end/back-end, retiring/bad speculation), execution unit port utilisation
  – Memory: bandwidth usage, transaction classification (page-hit, page-empty, page-miss)
• Can be used to see how workloads differ (or resemble) each other and the benchmarks we use (HS06, SpecCPU2017?)
• CPU counters are a powerful (but complex) tool and Trident makes them accessible
The goal is to define a common framework for modelling the computing requirements of the LHC experiments

- Models as collection of parameters and standard calculations, to be as generic and customisable as possible
- Takes as one of its inputs the characteristics of the workflows
- Reproduce with reasonable accuracy (but not supersede!) the official estimates from the experiments
- Allow to play with different scenarios to explore potential gains

Current status

- A first iteration of the framework was obtained by refactoring and generalising (to a certain extent) a framework used by CMS
  - [https://github.com/sartiran/resource-modeling](https://github.com/sartiran/resource-modeling)
- Elicited strong interest from other LHC experiments
  - Agreed as a common basis for future development
Resource estimation (2/2)

- LHC parameters (trigger rates, live fractions, shutdown years, ...)
- Computing model (event sizes and processing times, improvement factors, ...)
- Storage model (numbers of versions, replicas, ...)
- Infrastructure (capacity model, T1 disk and tape, ...)
- Time granularity is yearly
  - While resource needs vary over the year
- No network estimates (for now)
- Extrapolation to HL-LHC relies on very uncertain estimates – the workloads don’t exist yet
Site cost estimation models

- Develop a method to assess how well an infrastructure is matched to the needs of the experiment workloads
  - Capacity can be matched to local cost
  - Fabric can be tuned to maximise the capacity over cost
  - Several site people in the WG went through a cost estimation exercise starting from an “example” workload
    - The goal is not to compare sites, but to provide tools to optimise expenditure
  - Actual model developed in IN2P3 and successfully applied to T1 to model yearly investment per sector
    - https://indico.cern.ch/event/304944/contributions/1672219/ (CHEP 2015)

- Main sectors
  - Hardware: servers, racks, switches
  - Electricity: to run the hardware, cooling
  - Infrastructure: rooms, routers
  - Manpower
**Infrastructure costs at CCIN2P3: hardware**

- **Main conditions**
  - Exponential decrease of costs
  - Flat budget
    - Used for capacity replacement + capacity increase
  - Replace hardware when warranty expires

\[
c^*(t) = c(t) \frac{r}{1 - (1 - r)^\tau}
\]

- \(c^*\) = modeled cost
- \(c\) = real cost
- \(\tau\) = warranty time
- \(r\) = cost decrease rate

- Model predictions checked within 20% of reality
  - Most of the uncertainty comes from tape

**Source:** R. Vernet
Infrastructure costs at CCIN2P3: power and total

- Power consumption cost changes more difficult to predict
- Predicting future costs is possible
- Other sites are invited to use the same principles

Hardware cost + Power cost = Total cost

Infrastructure costs at CCIN2P3:

- CPU: 39%
- Disk: 26%
- Tape: 2%
- Rest: 33%
Areas of potential savings

- Many “small” improvements can stack to provide significant gains
- A quantitative estimation is highly desirable
  - OK to quantify not very realistic scenarios as it still provides a measure of the “gap”
  - Numbers below are based on exploratory work and are not to be taken literally – the goal is to stimulate more accurate estimates
    - Some savings could be reduced by “side effects”. Eg.: storage consolidation could cause loss of resources for some funding schemes → another argument for advocating a careful evaluation
    - https://indico.cern.ch/event/704519/

<table>
<thead>
<tr>
<th>Change</th>
<th>Effort Sites</th>
<th>Effort Users</th>
<th>Gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>managed storage on 15 sites + caches</td>
<td>Some on large sites/gain on small sites</td>
<td>little</td>
<td>40% decrease in operations effort for storage</td>
</tr>
<tr>
<td>Data redundancy by tape backup</td>
<td>Some large sites</td>
<td>Frameworks some</td>
<td>30% disk costs</td>
</tr>
<tr>
<td>Reduced data replication and cold data</td>
<td>little</td>
<td>Frameworks some</td>
<td>15% disk costs</td>
</tr>
<tr>
<td>Scheduling and site inefficiencies</td>
<td>Some</td>
<td>Some</td>
<td>10-20% gain CPU</td>
</tr>
<tr>
<td>Reduced job failure rates</td>
<td>Little</td>
<td>Some-Considerable</td>
<td>5-10% CPU</td>
</tr>
<tr>
<td>Compiler and build improvements</td>
<td>None</td>
<td>Little</td>
<td>15-20% CPU</td>
</tr>
<tr>
<td>Improved memory access/management</td>
<td>None</td>
<td>Considerable</td>
<td>10% CPU</td>
</tr>
<tr>
<td>Exploiting modern CPU architectures</td>
<td>None</td>
<td>Considerable</td>
<td>100% CPU</td>
</tr>
<tr>
<td>Paradigm shift algorithms (ALICE HLT)</td>
<td>Some</td>
<td>Massive</td>
<td>Factor 2-100 CPU</td>
</tr>
<tr>
<td>Paradigm shift online/offline data (LHCb and ALICE)</td>
<td>Little</td>
<td>Massive</td>
<td>2-10 CPU 10-20 Storage</td>
</tr>
</tbody>
</table>

- Cumulative evolutionary changes
  - Storage costs: -45% less cost
  - Site operations for storage: -40%
  - CPU: +200% throughput

Source: M. Schulz
HL-LHC predictions

• What will change?
  – Running conditions (luminosity, pile-up, trigger rate)
  – Event generation (LO + NLO + NNLO)
  – Detector simulation (full + fast simulation)
  – Detectors (some completely new, with much more fine-grained information)
  – Reconstruction (new algorithms, momentum cuts)
  – Analysis (new data formats)
  – Software (new algorithms, machine learning, vectorization)
  – Fabric (many-core CPUs, GPUs, accelerators)

• Need to develop sensible models for future workloads

• Initially, lots of unknowns, huge uncertainties

• Create “fake” workloads?
HL-LHC computational complexity

- **Event size**
  - Linear in $\mu$, apart from the most compact analysis formats

- **Reconstruction time**
  - Dependency with $\mu$ is linear for
    - Calibration
    - Pattern recognition for low $\mu$
    - Linking of tracks and calorimetric objects for low $\mu$
  - It will be exponential for high $\mu$ for
    - Pattern recognition
  - Overall, it can be modelled as $t(\mu) = a\mu + b\mu - \mu_{\text{crit}}$

- **Simulation time**
  - Event generation and simulation independent from $\mu$
  - Digitisation linear in $\mu$

- **Analysis time**
  - Independent from $\mu$
The cost model WG is by construction tightly connected with other groups and communities

- **HEPIX**
  - Mainly on benchmarking and fabric technology evolution

- **WLCG DOMA (Data Organisation Management and Access)**
  - Aims at greatly reduce the cost of storage by consolidation, caching, rationalization of protocols and services

- **WLCG Archival Storage Working Group**
  - Improve understanding of the cost of tape archives

- **CERN EP**
  - R&D on software to meet the challenges of Run3 and HL-LHC

- **HSF**
  - Collaborating on software optimization and tools
The WLCG/HSF systems performance working group was established to improve our understanding of the evolution of the cost of computing for LHC (and HEP)

- HL-LHC requires us to squeeze all the performance we can get at all levels

The WG is active on many fronts and is already achieving important results

- Reference workloads and performance analysis tools
- Model for site cost estimation
- Framework on resource need estimation

Work is still in progress but the time scale is long

- One of the biggest challenges is to produce reliable estimates for HL-LHC

Several interactions with many other activities and bodies in the community

- Active participation from more people is always welcome and encouraged!
Author list

## Workload metric summary

<table>
<thead>
<tr>
<th>Type</th>
<th>Events</th>
<th>Duration (hours)</th>
<th>CPU efficiency (%)</th>
<th>PSS/process (MB)</th>
<th>Disk read rate (kB/s)</th>
<th>Disk write rate (kB/s)</th>
<th>Network traffic (kB/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATLAS sim</td>
<td>1000</td>
<td>9.4</td>
<td>98</td>
<td>500</td>
<td>140</td>
<td>70</td>
<td>negligible</td>
</tr>
<tr>
<td>ATLAS digi reco</td>
<td>2000</td>
<td>4.0</td>
<td>84</td>
<td>1500</td>
<td>2600</td>
<td>1900</td>
<td>negligible</td>
</tr>
<tr>
<td>ATLAS derivation</td>
<td>?</td>
<td>2.3</td>
<td>96</td>
<td>1400</td>
<td>5600</td>
<td>580</td>
<td>negligible</td>
</tr>
<tr>
<td>CMS GENSIM</td>
<td>500</td>
<td>0.5</td>
<td>97</td>
<td>200</td>
<td>600</td>
<td>240</td>
<td>negligible</td>
</tr>
<tr>
<td>CMS DIGI premix</td>
<td>500</td>
<td>0.25</td>
<td>58</td>
<td>400</td>
<td>1600</td>
<td>1900</td>
<td>3300</td>
</tr>
<tr>
<td>ALICE pp</td>
<td>1</td>
<td>0.3</td>
<td>100</td>
<td>700</td>
<td>600</td>
<td>60</td>
<td>negligible</td>
</tr>
</tbody>
</table>