HEP Benchmarks

Considerations about how to transition HS06 → HEPscore

D. Giordano (CERN / IT-CM-RPS)

WLCG MB

26 May 2020

Outlook

- Why are we doing this activity?
  - WLCG has to abandon (at some point in the future) HS06
    - EoL support, targets only CPUs,
      scales well with HEP WLs mainly on Intel x86 CPUs

Scenarios	HS06	New Benchmark
x86 CPUs (y. 2010-2020)	✓	
New CPUs (AMD) and/or arch (ARM/..)	?	
New Exp Sw	?	
CPU + GPU/FPGA/…	✗	

- Transition to a **field-specific** benchmark (**HEPscore**) is demonstrated
  - The HEPiX Benchmarking WG has extensively documented design, build/run process, code and results

- **Opportunity** for the WLCG community to review the concepts and challenges of pledging, accounting, procurement, and **define the policies**
  - Technically several pledging/accounting scenarios are possible ⟹ Policies shall drive implementations

Some clarifications

There are some recurring questions/doubts that can be clarified here

1. Does HEPscore favor a WL or an Exp. respect to another? Example the fastest one?
   - [*Question received at the HSF-WLCG workshop*]
   - **Answer**: **NO**. It's based on speed factors *(see slide 4)*

2. How to track the configurations?
   - **Answer**: All configurations are **versioned** (by hash string), and the versions are part of the **score report**
     - NB: The versions of the WLs are also part of this versioning. E.g.: two LHCb gen-sim versions ⟹ two score versions

3. Does HEPscore weights all WLs in the same manner?
   - **Answer**: It depends on **WLCG policy**
   - Technically both possibilities are in place: it can weight all WLs equally, or adopt tuned weights (see slide 4,5)

4. How often shall the HEPscore change **default** configuration?
   - **Answer**: It depends on **WLCG policy.**
     It could be never.
     Or only when a new production sw provides a major change in performance to be reproduced by the benchmark score.

5. How to assess pledges of old CPUs with new software?
   - **Answer**: **WLCG policy**, to be addressed in the **accounting infrastructure.** Scenarios: fixed score, or evolving score, etc
     - This aspect is true for any benchmark

Comparing by “speed factors”

- In order to compare servers HS06, as many other benchmark suites, implements the **geometric mean** approach
  - Ingredients
    - a set of reference workloads (**WLs**)
    - a measure of performance per WL (m_i), that typically goes as [1/s] (eg. can be the event throughput)
    - a reference machine
- The score **S** of a server (**srv**) is defined as the **geometric mean** of the **speed factors** x_i(**srv**,ref) = m_i(**srv**)/m_i(**ref**) respect to the reference machine (**ref**)
  - i.e. “speed” is *normalised* respect to the reference machine “speed”

$$\left(\prod_{i=1}^{n} x_i\right)^{\frac{1}{n}} = \sqrt[n]{x_1 x_2 \cdots x_n}$$

https://en.wikipedia.org/wiki/Geometric_mean

- The relative score between srv_A and srv_B is defined as the ratio of the scores S(srv,ref)
  - This is still a geometric mean of speed factors

	WL_1		WL_2		WL_n		Score	S(A,B)
Ref. Srv	m_1(ref)	1 (by def)	m_2(ref)	1 (by def)	m_n(ref)	1 (by def)	$\left(\prod_{i=1}^{n} x_i\right)^{\frac{1}{n}}$	
Srv A	m_1(A)	x_1(A,ref)	m_2(A)	x_2(A,ref)	m_n(A)	x_n(A,ref)	S(A,ref)	$\frac{S(A,ref)}{S(B,ref)}$
Srv B	m_1(B)	x_1(B,ref)	m_2(B)	x_2(B,ref)	m_n(B)	x_n(B,ref)	S(B,ref)	

HS06 → HEPscore

HEPscore implements the **same approach** as HS06

- A set of workloads is used to measure the **speed factors**
  - WLs suggested by the experiments' liaisons
- A **reference machine** has been identified
  - CPU Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40GHz (32 cores, SMT ON): α_{ref} = **355 HS06** (measured HS06)
- HEPscore is the **geometric mean** of the **speed factors**
  - NB: it can be easily extend to the **weighted geometric mean**, if the workloads have to be differently weighted (*to be defined as **WLCG policy***)
- In order to provide scores in the same order of magnitude as HS06, a **normalization factor** is introduced, equal to α_{ref}
  - HEPscore(srv,ref) = $\bar{x} = \alpha_{ref} \sqrt[n]{\prod_{i=1}^{n} x_i}$

Workload	ATLAS gen	ATLAS sim	ATLAS digi-reco	CMS gen-sim	CMS digi	CMS reco	LHCb gen-sim
Robustness	✅	✅	✅	✅	✅	✅	✅
Reproducibility	0.8%	2%	0.6%	1.5%	1%	1%	1%
Memory	✅	✅	✅	✅	✅	✅	✅
Image size (unpacked)	1.5 GB	2.0 GB	6GB	10 GB	6.5 GB	5.5 GB	2.6 GB
Readiness	✅	✅	✅	✅	✅	✅	✅

✅ okay
❌ blocker

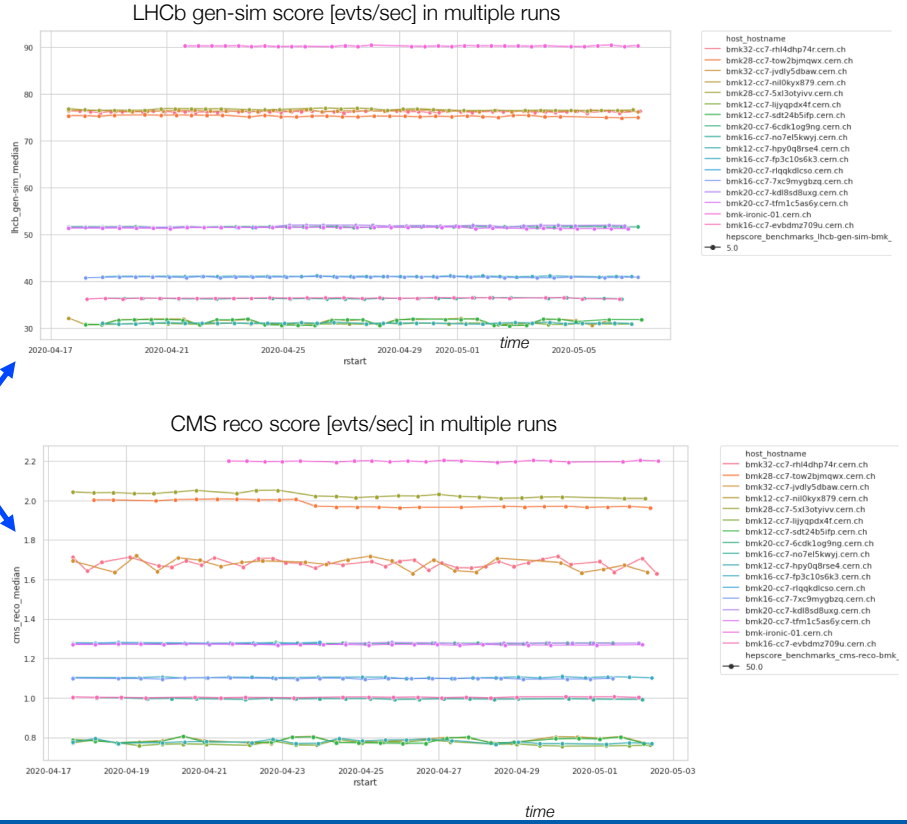
Reproducibility, evaluated as spread in repeated measurements $(score_{max} - score_{min})/score_{mean}$

$$\bar{x} = \left(\prod_{i=1}^{n} x_i^{w_i} \right)^{1/\sum_{i=1}^{n} w_i}$$

https://en.wikipedia.org/wiki/Weighted_geometric_mean

Test: Compare HS06 Vs HEPscore results

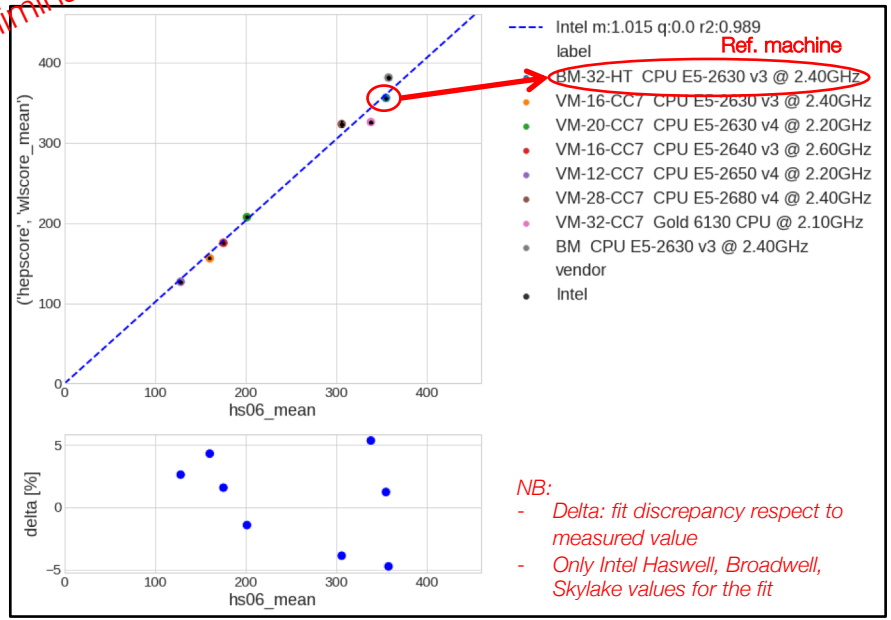
- **HS06** has been used for a decade because **highly correlated** with the Exp. WLs on x86 CPUs
- We should find some level of **correlation**, comparing machines with HS06 and HEPscore
  - We should also **expect discrepancies** when including new CPU models
- This has been **verified** collecting **hundreds** of benchmark events, for several **CPU models**
  - All CPU models of CERN T0 (see next slides)
  - CPU models from other sites (still under study)
  - Notice the stable WL results!



The linearity

- Intel Haswell, Broadwell, Skylake model examined
- HEPscore, built with CMS, Atlas and LHCb workloads scales extremely well with HS06
  - Delta fit < 5% , $r^2 = 0.989$
- Fitted coefficient **m~1** ⇔ approx. same CPU score measured with HS06 and HEPscore
  - NB: Obtained simply normalizing by $\alpha_{ref} = 355$ HS06

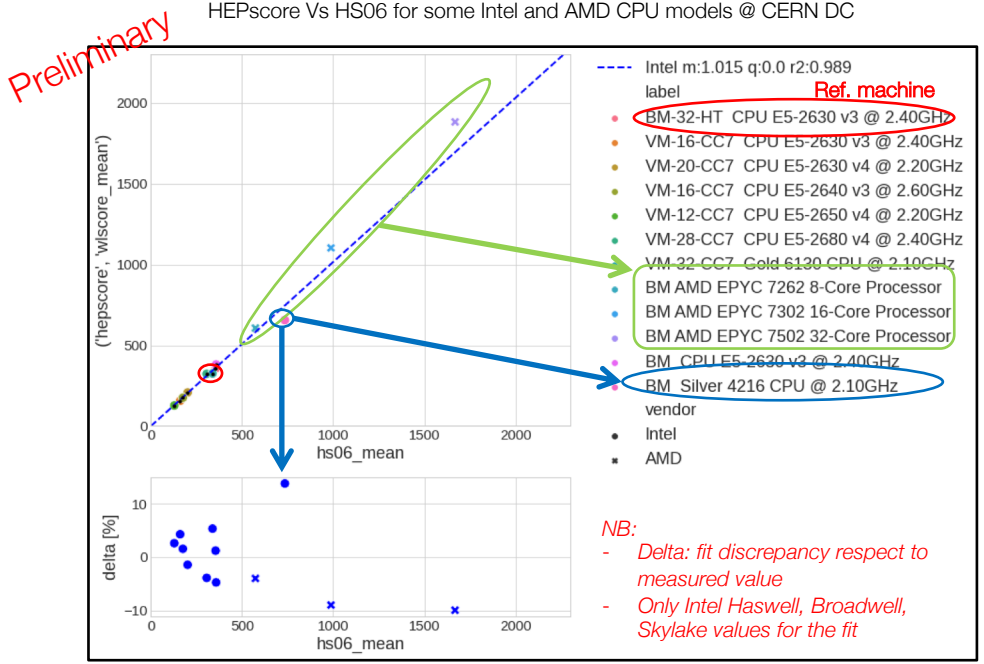
Preliminary



Let's add other CPU models

- Added Intel Cascade Lake (Silver) and 3 AMD EPYC Rome
  - Benchmarked production and test servers in collaboration with **CERN procurement team**

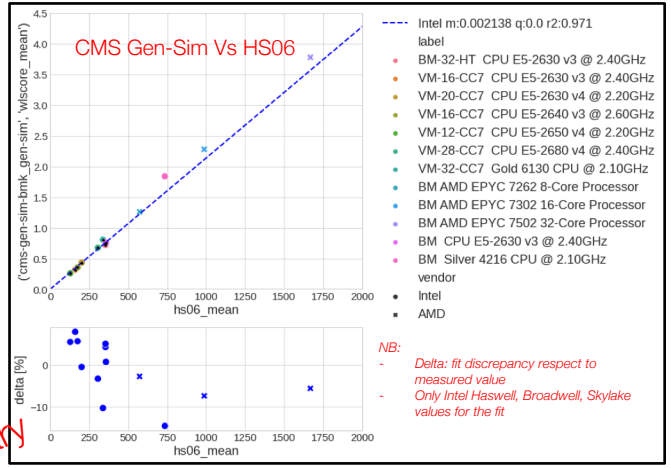
- Respect to HEPscore
  - HS06 score **under-estimates** AMD performance of ~10%
  - HS06 score over-estimates Cascade Lake of ~13%



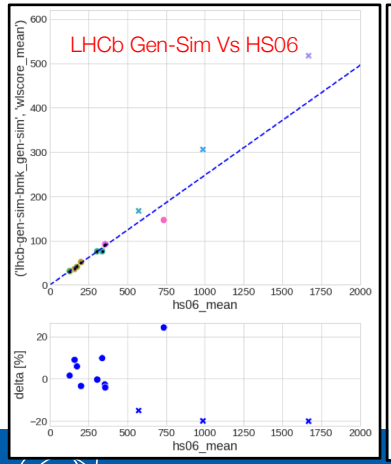
A look at the individual Exp. WLs

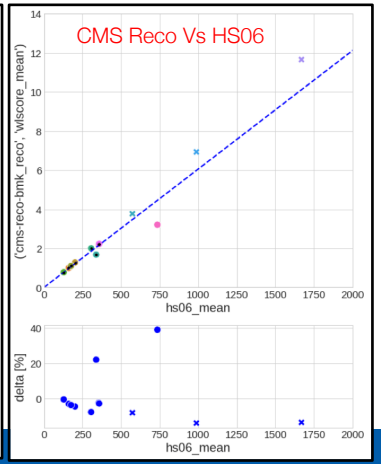
- Understand performance changes across CPU models for each individual HEP WL is possible with **HEPscore** (by construction)
  - It is not possible with **HS06**, because HS06 is not HEP specific
- What we can already see:

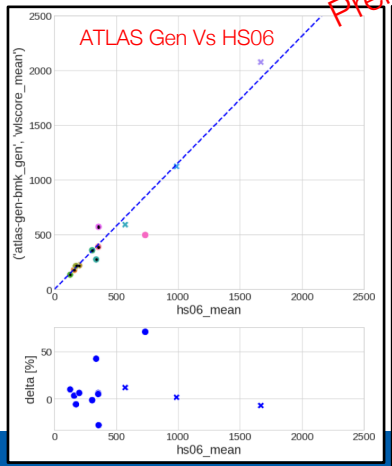
  across CPU models, HS06 scales within **10%** only for a subset of the HEP WLs
  - In some cases differences are **>20%** and explain why ***some Experiments notice more discrepancies of their Grid job performance with respect to HS06 score***

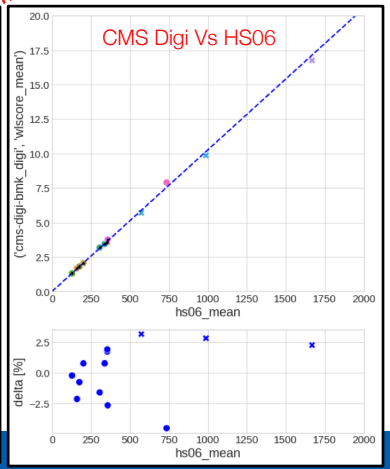


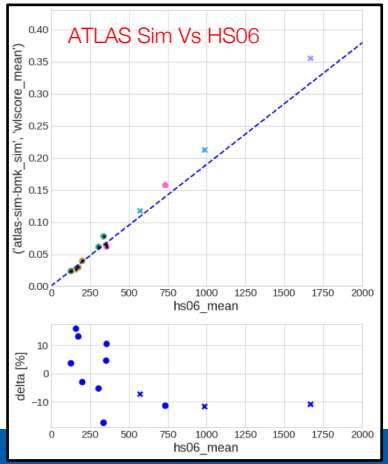
Preliminary











Conclusions

- **By construction** HEPscore offers an intrinsic relation with the performance of production jobs
  - Makes easier the prediction/review of the performance of the running WLCG jobs
  - Simplifies the transition to **heterogeneous** resources
    - The score definition remains the same, based on event throughput
- **Technically** HEPscore is versatile and as robust as the HEP SW running on the same HW
  - Adopts frozen WL container versions; allows configurable set of WLs (and weights); tracks version changes
- For WLCG pledging/accounting several scenarios are possible
  - WLCG **policies** should define the requirements

Scenarios	HS06	HEPscore
x86 CPUs (y. 2010-2020)	✓	✓
New CPUs (AMD) and/or arch (ARM/..)	?	✓
New Exp Sw	?	✓ (w/ new reference WLs)
CPU + GPU/FPGA/…	✗	✓ (same speed definition: event/s)