Benchmarking worker nodes using LHCb productions and comparing with HEPSpec06

P Charpentier
EP Department, CERN, Geneva, Switzerland

Email: Philippe.Charpentier@cern.ch

Abstract. In order to estimate the capabilities of a computing slot with limited processing time, it is necessary to know with a rather good precision its "power". This allows for example pilot jobs to match a task for which the required CPU-work is known, or to define the number of events to be processed knowing the CPU-work per event. Otherwise one always has the risk that the task is aborted because it exceeds the CPU capabilities of the resource. It also allows a better accounting of the consumed resources.

The traditional way the CPU power is estimated in WLCG since 2007 is using the HEP-Spec06 benchmark (HS06) suite that was verified at the time to scale properly with a set of typical HEP applications. However, the hardware architecture of processors has evolved, all WLCG experiments moved to using 64-bit applications and use different compilation flags from those advertised for running HS06. It is therefore interesting to check the scaling of HS06 with the HEP applications.

For this purpose, we have been using CPU intensive massive simulation productions from the LHCb experiment and compared their event throughput to the HS06 rating of the worker nodes. We also compared it with a much faster benchmark script that is used by the DIRAC framework used by LHCb for evaluating at run time the performance of the worker nodes.

This contribution reports on the finding of these comparisons: the main observation is that the scaling with HS06 is no longer fulfilled, while the fast benchmarks have a better scaling but are less precise. One can also clearly see that some hardware or software features when enabled on the worker nodes may enhance their performance beyond expectation from either benchmark, depending on external factors.

1. Introduction

There are many reasons why benchmarking the computing capabilities of a worker node is very important:

- Before attempting executing a given task, it is necessary to know whether that task will have enough compute resources to be able to complete successfully. This is in particular true when processing data files in which all events must be processed.
- In order to be able to report the actual usage that was done of compute resources made available to an experiment by resource providers, it is important to be able to keep an accounting of the consumed resources.

The types of computing resources (CPU architecture, caching capabilities, memory access time…) as well as their configurations are extremely diverse and each compute resource provider decides on
its own how to best provision and configure its resources: virtual or physical machines, hyper-threading, over-clocking… Therefore, knowing the type of processor is not enough to be able to assess what is the actual capability of a worker node (WN) in terms of CPU power.

It is therefore necessary to know these capabilities at the time when a task is about to be executed in order to know whether it will be able to complete. It is then possible to compare the estimated CPU power with the speed at which the task was effectively executed. If the task is calibrated, i.e. either executing a fixed application (typically a benchmarking suite like HEP-Spec06 [1]) or executing very similar sub-tasks (e.g. simulating a large number of HEP events of a given type), a large number of similar tasks allow a comparison of the actual performance for the users with the estimated CPU power.

2. Benchmarking metrics

2.1. Definition of benchmark metrics used in this study

LHCb is using for distributed computing the DIRAC framework [2] that provides an integrated fast benchmarking suite (taking about 90 seconds to execute), referred to here as DB16\(^1\) (for DIRAC Benchmark ‘16). This benchmark is executed on the WN when the overlay “pilot job” starts, prior to executing any actual task.

Event simulation campaigns (called productions) aim at generating LHC collisions and letting the produced particles propagate through the detector (using the Geant4 package). Although each collision (a.k.a. event) is different, the main characteristics of these events are on average very similar, and as long as many events (typically several hundred) are simulated in a single task, the amount of CPU resources required per event is very similar. Measuring the number of events simulated per unit of time is therefore a good metrics of the CPU power of the computing slot.

Similar techniques can be used for other types of applications (e.g. real events reconstruction), although it is less easy to guarantee that all events are on average similar.

WLCG has decided in 2007 to use a specific benchmarking suite (called HEPSpec06, or HS06) for evaluating the amount of CPU resources to be provisioned by sites. It is also the unit used for experiments to request CPU resources. This benchmark was shown at the time to scale well with the available HEP applications. However, the CPU architectures, the applications themselves have since then evolved whereas the definition of HS06 remained the same. It is therefore interesting to compare the HS06 evaluation with the DB16 fast benchmarking and with the simulation productions performance.

In this study, we shall compare thus for several types of productions (mainly simulation productions, but also real data reconstruction productions or event selection productions) the following three metrics:

- The DB16 fast benchmark (labelled as “Dirac” or “DiracPower” in the plots)
- The event rate performance metrics (labelled as “Job” or “JobPower” in the plots)
- The HS06 rating of the WNs when available (labelled as “MJF” in the plots because it is made available via the “Machine and Job Features” [3] mechanism)

2.2. Linearity of CPU-work with the number of events

In order to use the event rate performance as a valid metrics, it is important to verify that the amount of CPU-work is proportional to the number of events processed. CPU-work is defined as the product of the CPU time required to process one event by the CPU power of the worker node. It is expressed in HS06.s.

Figure 1 shows for two types of productions how the consumed CPU time scales with the number of processed events. These data were collected on a small subset of WNs that demonstrate a very

\(^1\) DB16 that was used in this study is a variant of the now well-known DB12 benchmark, just applying a scale factor of 0.65 in order to be closer to the HS06 unit. \(DB16 = DB12 \times 0.65\)
stable behavior. The linearity is pretty good. The dispersion around a perfect line is larger for reconstruction than for simulation, but this is expected as simulated events are by far more similar than real events. The accumulation at certain number of events are due to how the tasks are created: most simulation tasks generate around 400 events and most real data files have a fixed size of 3 GB which corresponds to roughly 59000 events.

**Figure 1**: CPU time (in seconds) consumed on a calibrated set of WNs as a function of the number of processed events for two types of productions (left: simulation, right: reconstruction).

As the DiracPower is evaluated in each job when the pilot job starts, the conditions in which this evaluation is done may vary (load on the worker node, turbo boost etc…), introducing a spread in the estimation as can be shown on **Figure 2** where the DB16 measurement is shown at a reference site that is running most LHCb jobs and has a very homogeneous set of worker nodes. The spread of the DB16 estimate is about 10% with an asymmetric share (larger values occur when worker nodes are idle and/or their clock speed is boosted). A measurement of DB16 in standard conditions (e.g. at boot time) would greatly help in reducing this spread.

**Figure 2**: Estimated DB16 power at the reference site.

3. **Comparison between JobPower and DB16**

In order to allow an easy comparison, the JobPower metrics was rescaled such as to have the same value as the HS06 CPU power on a reference set of WNs (arbitrarily chosen as CPUE5-2650v2@2.60GHz at the RAL site).
Using all tasks within selected productions, it is then possible to compare the metrics by looking at the ratio of these metrics for each task. This ratio can be studied as a function of the site, or for the various types of CPU deployed at these sites. For sake of simplicity, we show here only a subset of sites, namely the LHCb main sites (Tier1s) since these sites are used for running both simulation and reconstruction productions. It allows thus a direct comparison of the benchmarks for these two types of productions at the same sites.

**Figure 3**: Ratio of the measured JobPower to the DiracPower (DB16) at the LHCb Tier1 sites for two types of productions (left: simulation and right: reconstruction).

There are big differences (see Figure 3) in the comparison of central values between the various sites, with spreads as large as 20%. It can be noticed as well that the spread is larger for reconstruction tasks than for simulation tasks. Central values are as well different for these two types of productions, but not at all sites.

This difference of behaviour led us to extend the comparison to WN models in order to see whether this dispersion and the differences between productions could be related to the different types of WNs installed at these sites.

**Figure 4**: Ratio of JobPower to DiracPower for various CPU models at a subset of LHCb Tier1s, for the two types of productions (left: simulation, right: reconstruction).

The dispersion is smaller within each WNs model than when comparing sites as can be seen in Figure 4. It is also clear that the two types of productions have different behaviors depending on the
WN model. For example, the most populated WN models at RAL show a larger difference for reconstruction tasks than for simulation tasks. This indicates that the scaling for these two types of productions is different, probably because the simulation is much more CPU-bound than the reconstruction. Anyway the two types of productions behave differently and also differently from the DB12 benchmarking.

There is also a large spread within each model which is due partly to the observed intrinsic uncertainty in the determination of the DiracPower (10%) but also to the effects of certain worker node optimizations while running the applications (the actual power may vary while the application is running).

4. Comparison between JobPower and HS06
Although not many sites are providing the HS06 rating through the “Machine and Job Features” mechanism (MJF), it is possible to compare the power estimated from the jobs with that provided by the sites.

**Figure 5:** Ratio of JobPower to HS06 benchmark (a.k.a. MJF) for various CPU models at LHCb Tier1s that provide that information, for two types of productions (left: simulation, right: reconstruction).

**Figure 6:** Ratio of JobPower to HS06 power within the same Grid job for the simulation step (Gauss application, x-axis) and the reconstruction step (Brunel application, y-axis).
Figure 5 shows the ratio of the power estimated from the job performance to the HS06 rating for the same two productions. It is clear that the JobPower agrees much better with the HS06 benchmark for reconstruction jobs than for simulation jobs where for some CPU models the job performance is 20% better than expected from HS06 power estimate.

As the simulation jobs also contain a step of event reconstruction, it is possible to make a direct comparison, within the same job between these two steps. This is shown in Figure 6.

The comparison of the two steps within the same jobs show clearly the good scaling of the reconstruction application with HS06 while the simulation application (much more CPU-bound) has differences up to 20% that correspond to specific models of CPU.

5. Conclusions
LHCb have performed comparisons between the fast benchmark provided by DIRAC (DB16), by HEPSpec06 (through the MJF mechanism) and by the performance of jobs of different types (simulation and reconstruction).

DB16 benchmark provides a fair estimate of the CPU power, however with WN-model dependencies. This is however used by LHCb for estimating the capabilities of WNs before matching their jobs, or for estimating the number of simulation events that can be generated in a given job.

HS06 has bad scaling properties with the LHCb applications, in particular with the simulation (CPU-bound) application. Scaling with reconstruction applications is however much better. There are also WN-model dependencies for reconstruction jobs in particular, like for DB16.

References
[1] HEPSpec06 specifications, http://w3.hepix.org/benchmarks/.
[2] LHCb Collaboration (Stagni F et al.) 2012 The LHCb DIRAC-based production and data management systems, J. Phys. Conf. Ser. 368 (2012) 012010.
[3] Alef M et al 2016 Machine / Job Features, HEP Software Foundation note HSF-TN-2016-02.