Statistical Comparison of CPU Performance for LHCb Applications

Ricardo Graciani

Universitat de Barcelona

E-mail: graciani@ecm.ub.es

on behalf of the LHCb Collaboration

Abstract. The usage of CPU resources by LHCb on the Grid is dominated by two different applications: Gauss and Brunel. Gauss is the application doing the Monte Carlo simulation of proton-proton collisions. Brunel is the application responsible for the reconstruction of the signals recorded by the detector converting them into objects that can be used for later physics analysis of the data (tracks, clusters, . . .).

Both applications are based on the Gaudi and LHCb software frameworks. Gauss uses Pythia and Geant4 as underlying libraries for the simulation of the collision and the later passage of the generated particles through the LHCb detector. While Brunel makes use of LHCb specific code to process the data from each sub-detector. Both applications are CPU bound. Large Monte Carlo productions or data reconstructions running on the Grid are an ideal benchmark to compare the performance of the different CPU models for each case. Since the processed events are only statistically comparable, only statistical comparison of the achieved performance can be obtained.

1. Introduction

A large fraction of the CPU resources needed by LHC experiments on the Grid will be dedicated to Monte Carlo simulation. Although there are small variations for each experiment, in all cases this is a very well defined cpu intensive computing task. In the case of LHCb [1], this simulation is done by a combination of two different applications Gauss [2] and Boole [3], based on the common LHCb Gaudi [4] software framework. The dominant contribution comes from Gauss, being responsible of the simulation of the proton-proton interaction produced by LHC using a general purpose package (Pythia [5]), and for the transport of the produced particles through the LHCb detector apparatus using another general purpose package (Geant4 [6] and [7]). Boole is responsible of converting the energy depositions reported by Gauss into digitized signals, simulating the response of the detector electronics and trigger. Gauss is the application consuming most of the cpu (over 95 % of the total).

The purpose of this paper is to establish a procedure that can be used for ABSOLUTE normalization of the cpu delivered by the grid computing resources to LHCb. A procedure based on the contribution by each single job calculated from the raw cpu time consumed and the CPU model being used is presented.

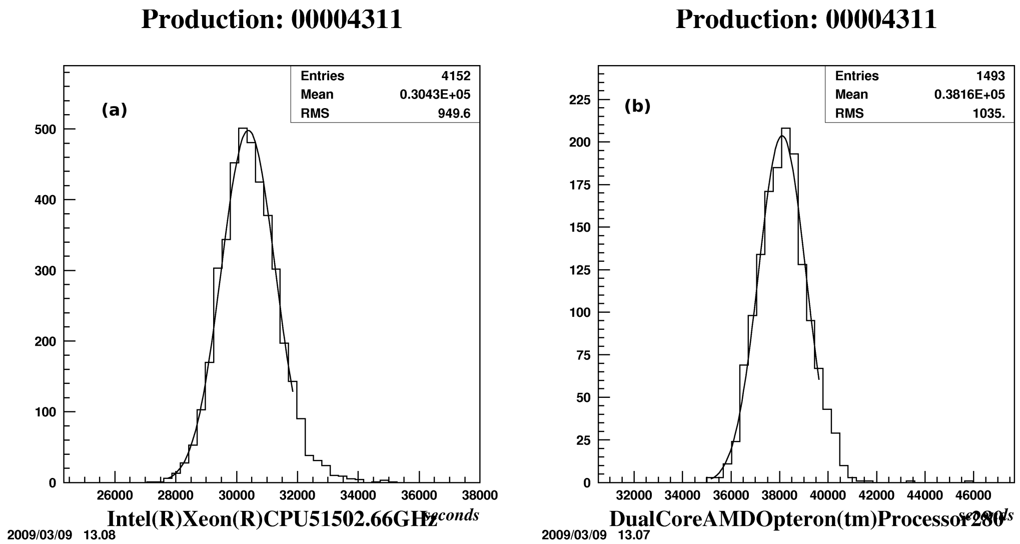

Figure 1. Distribution of raw cpu time for jobs from production 00003145 on Intel Xeon 5150 (left) and AMD Opteron 280 (right).

Using the information collected from recent LHCb large Monte Carlo simulation campaigns described in section 2 the relative performance of different CPU models is estimated. The resulting table and the comparison with standard CPU benchmarks like Spec© CPU2000 [8] and Spec© CPU2006 [9] are presented in section 3. Finally some conclusions are drawn in section 4.

2. Running Conditions

From October 2008 until February 2009, LHCb has produced a large amount of simulated data with the aim of injecting it into the LHCb Online Farm as they were coming from the real detector. This is done to test all the algorithms and procedures that are being prepared for the beginning of data taking. Among other productions [1], 00003145, 00003793, 00004311 correspond in total to over 400.000 independent jobs. While executing on the grid, each of these productions made used of more that 64 different CPU models.

Productions 00003145 and 00003793 correspond to the so called 2008 LHCb detector configuration (describing the geometry as it was prepared for the LHC startup in September of that year), and used two different versions of Gauss, v35r1 and v36r0 respectively. Production 00004311 uses Gauss version v35r1 but in this case with the new detector description for 2009. In all cases the same compilation options, those corresponding to the LHCb tag "slc4_ia32_gcc34", have been used.

For each simulation job the raw cpu time consumed is kept. Other parameters about the running environment, like the CPU model [2], the CPU clock frequency, the amount of memory and number of cores available, the flavour of the operating system (including the 64 or 32 bit), etc. are also kept for each job. After detailed analysis of these data, the first conclusion is that, for a given production, the only parameter on which the measured raw cpu time depends is the CPU model. Figure 1, presenting the distribution of raw cpu time measured for all jobs with two production/CPU model combinations, shows consistently less than 3% variation from job

[1] For LHCb, a production is a self consistent set of simulated data, produced with the same versions and options of the simulation programs, and only varying the initial random seeds for each simulated collision.

[2] The CPU model is taken as reported by the field "model name" in the /proc/cpuinfo file of the WN where the jobs is running removing blanks.

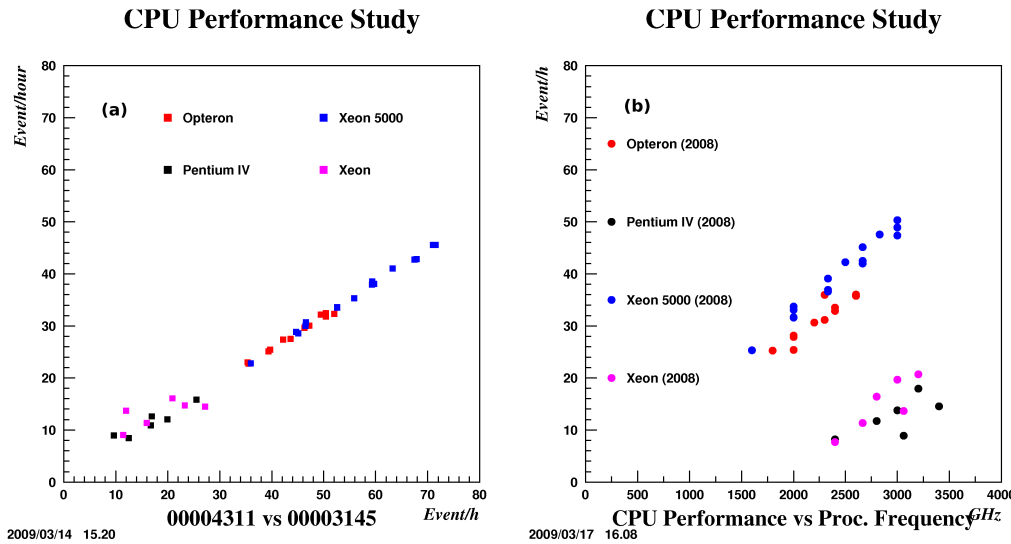

Figure 2. Performace measured in event/hour for production 00004311 versus production 00003145 (a). Performance measuread for production 00004311 as a function of CPU processor clock frequency (b).

to job for all combinations. A small tail to large values due to the intrinsic fluctuations of the Monte Carlo techniques used in the simulation can be seen.

3. Result of the Test

Table 1 summarizes our measurements. Each different production and CPU model combination (only those CPU models with enough jobs for all productions to allow a proper fit of the resulting distribution are included) is fitted using a Gaussian function as shown in figure 1. The resulting mean value of the fitted Gaussian is used to determine an event per hour value for each combination. This value is used hereafter as a measurement of the computing power of the given processor. As mentioned before, the typical sigma of the fitted Gaussian is between 2 and 3 % of the mean.

First, a check of the self consistency of the data is shown in figure 2(a). The details of the simulation depend essentially on the details of the detector apparatus description. Therefore one can not expect the absolute event/hour values for each production to be the same. However, if the application is well behaved, the different performance numbers should scale. In fact, this is what happens. When the cpu performances (event/hour) obtained for the different models are compared for different productions [3] (in the figure the case of production 00004311 versus production 00003145 is shown) one can see an excelent correlation. Figure 2(b) shows how the computing power obtained from different CPU models scales with the processor clock frequency. CPU models have been grouped in four families, Opteron from AMD and Pentium IV, Xeon and Xeon 5000 from Intel. Reasonably good scaling is found for Opterons and Xeon 5000s, with 10 to 15 % more performance for the Xeon models with respect to the Opterons at the same frequency.

With these results in hand, one can then go to the Spec© database [8,9] and try to compare. The comparison is restricted to the Opteron and Xeon 5000 families and only to those models for which there are published results in each case. All public results available for each CPU

[3] From now on production 00004311 is used as reference for simplicity, conclusions are not affected by this selection.

CPU Model	**00003145**	**00003793**	**00004311**
AMD Opteron™Processor 246	78179.8	70894.7	50888.7
AMD Opteron™Processor 250	60795.2	54779.6	38838.6
AMD Opteron™Processor 252	55974.6	50311.4	36377.2
Dual-Core AMD Opteron™Processor 2212HE	70824.5	64031.1	45309.3
Dual-Core AMD Opteron™Processor 2214	65807.5	58792.3	42650.9
Dual-Core AMD Opteron™Processor 2218	55723	49962.7	34617.9
DualCore AMD Opteron™Processor 265	79018.6	71213.6	50622
DualCore AMD Opteron™Processor 270	71709.6	64523.9	45672.3
DualCore AMD Opteron™Processor 275	65403.9	58690.3	41291.5
DualCoreAMDOpteron™Processor 280	59881.3	53658.8	38106
Intel®Core™2 CPU 6600@2.40GHz	51878	46959	31583.3
Intel®Pentium®4 CPU 2.40GHz	200297	220264	186981
Intel®Pentium®4 CPU 2.80GHz	165795	153901	107300
Intel®Pentium®4 CPU 3.00GHz	142991	130632	106410
Intel®Pentium®4 CPU 3.06GHz	212203	201879	143399
Intel®Pentium®4 CPU 3.20GHz	113528	100457	70521.3
Intel®Pentium®4 CPU 3.40GHz	149270	123961	90277.6
Intel®Pentium®D CPU 3.40GHz	81705.2	71799.8	50371.1
Intel®Xeon®CPU 5130@2.00GHz	62374.6	56908.3	40236.9
Intel®Xeon®CPU 5140@2.33GHz	53662.9	48714.2	34232.6
Intel®Xeon®CPU 5150@2.66GHz	47405.8	42850.4	30384.7
Intel®Xeon®CPU 5160@3.00GHz	42082.4	38011.6	26674.5
Intel®Xeon®CPU E5310@1.60GHz	78994.7	71072	50007.8
Intel®Xeon®CPU E5335@2.00GHz	62905.6	56858.3	39923.3
Intel®Xeon®CPU E5345@2.33GHz	53641.9	49140.8	34237.6
Intel®Xeon®CPU E5405@2.00GHz	58637.1	53384.9	38654.7
Intel®Xeon®CPU E5410@2.33GHz	50944.3	46030.7	32200.2
Intel®Xeon®CPU E5420@2.50GHz	47274.6	42606	30141.1
Intel®Xeon®CPU E5430@2.66GHz	43889.1	39868.9	28454.4
Intel®Xeon®CPU E5440@2.83GHz	42058.3	37866.8	26531.8
Intel®Xeon®CPU E5450@3.00GHz	39481.5	36794.8	25177.1
Intel®Xeon®CPU L5335@2.00GHz	60031.3	54427.1	38714.2
Intel®Xeon®CPU X5355@2.66GHz	46759.9	42329.4	30348.6
Intel®Xeon®CPU X5450@3.00GHz	39478.2	35758.1	25358.2
Intel®Xeon™CPU 2.00GHz	72135.8	64007.9	46068.4
Intel®Xeon™CPU 2.40GHz	199974	235056	157127
Intel®Xeon™CPU 2.66GHz	158437	159017	112814
Intel®Xeon™CPU 2.80GHz	122246	109780	77324.4
Intel®Xeon™CPU 3.00GHz	124349	91513.3	66093.2
Intel®Xeon™CPU 3.06GHz	131417	131616	149736
Intel®Xeon™CPU 3.20GHz	112073	87047.4	86055.9
Quad-Core AMD Opteron™Processor 2356	55542.8	50084	35716
Quad-Core AMD Opteron™Processor 8356	56588.9	57834	35694

Table 1. Average cpu time (in seconds) measured for all jobs of a given production/CPU model combinations.

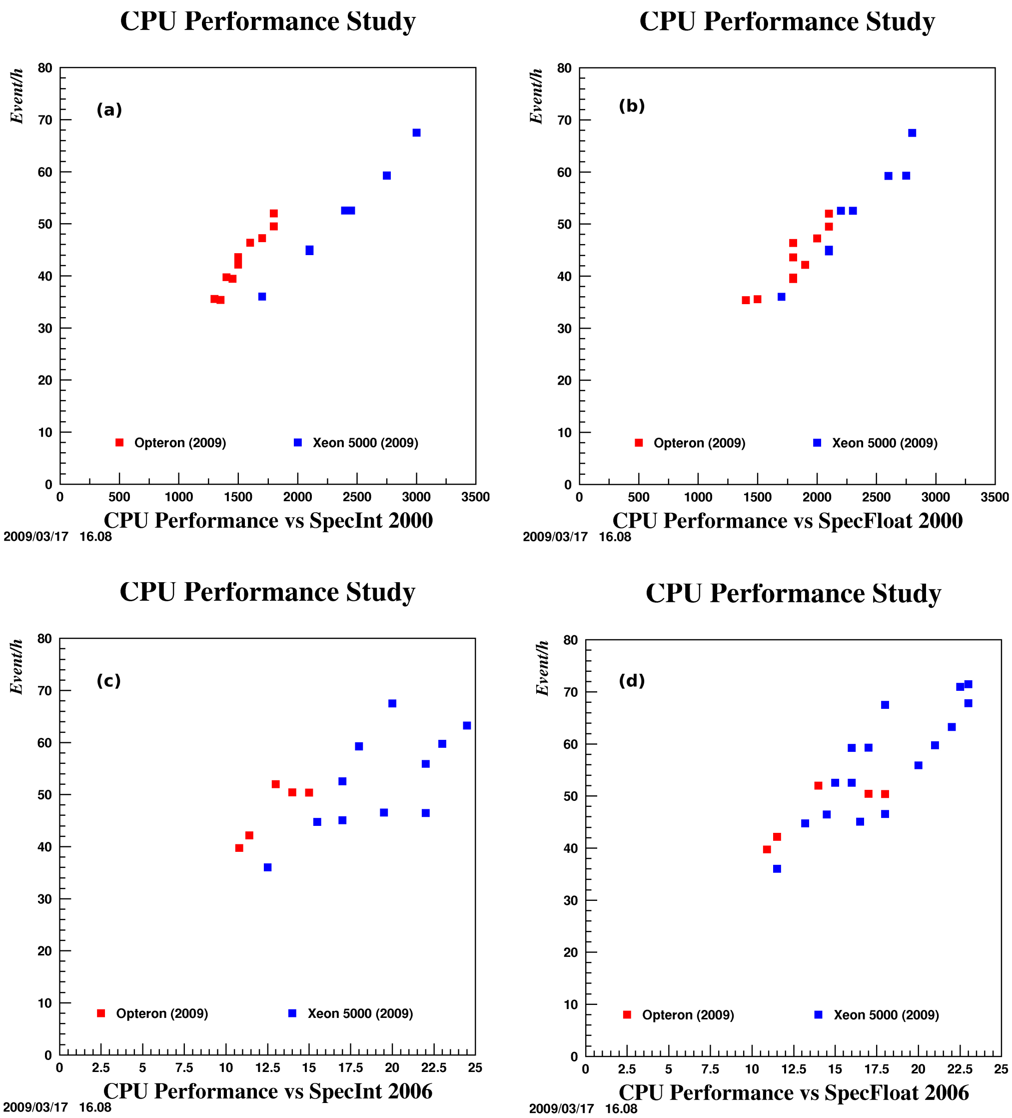

Figure 3. Comparions of the measured cpu performance with Spec© CINT2000 (a), CFP2000 (b), CINT2006 (c) and CFP2006 (d) benchmarks published averages.

model were searched and an average of the published values was taken.

3.1. Comparison with Spec© CPU2000

First, Spec© CPU2000 results were considered, since so far all WLCG sites are using this unit to publish their CPU normalization factors. Figure 3(a) shows the correlation between the performance numbers obtained and the averages CINT2000 for each of the CPU models. As can be seen, there is a good correlation for processors in the same family, but CINT2000 does a systematic underestimation of about 20 % of the performance Opteron processors can deliver to LHCb simulation with respect to Xeon 5000s. Figure 3(b) shows the correlation found between our measurement and CFP2000. In this case, the in-family correlation is not that good any more, but on the other hand there is some overall better behaviour and the systematic bias has almost disappeared.

3.2. *Comparison with Spec© 2006*

Since Spec© retired its CPU2000 benchmark in 2007, it was found interesting to compare the measured performance values with published results for this new benchmark. Figure 3(c) shows the observed correlation of the measured performance with the average of the CINT2006 published results. It can be seen that the in-family correlation observed with CINT2000 breaks. One can see how different Xeon 5000 processors delivering the same performance for LHCb Monte Carlo simulation differ up to 50 % in their CINT2006 published values. Finally, figure 3(d) shows the correlation with CFP2006 published values.

4. Conclusion

Large samples of LHCb Monte Carlo simulation jobs have been used to determine a cpu performance value for different CPU models. The dispersion of the job to job variation observed is in the 2-3 % level, perfectly compatible with the intrinsic statistical variation expected from the underlying Monte Carlo model. Results from different versions of the simulation programs and for different detector descriptions show an excelent correlation. A reasonable scaling is found with respect to the cpu clock frequency for the processors of the Opteron and Xeon 5000 families, with a 15 % higher performance for the Xeons with respect to the Opterons for the same processor frequency.

When comparing with Spec© published results for the corresponding CPU models the following features are observed: both for 2000 and 2006 benchmarks there is a better correlation of the measured performances with the CFP values than with the CINT ones; within one of the processor families considered CINT2000 shows an excelent correlation with the measured values, it also shows an overestimation of 20 % for Xeons with respect to Opterons; for CINT2006 there are huge differences for processors of the same family (up to 50 %) that are not observed by the LHCb Applications; the best correlation is found for CFP2000 benchmark results.

References

[1] Antunes-Nobrega R *et al.* (LHCb) CERN-LHCC-2003-030
[2] 2009 The GAUSS Project `http://lhcb-release-area.web.cern.ch/LHCb-release-area/DOC/gauss/`
[3] 2009 The BOOLE Project `http://lhcb-release-area.web.cern.ch/LHCb-release-area/DOC/boole/`
[4] 2009 The GAUDI Project `http://proj-gaudi.web.cern.ch/proj-gaudi/`
[5] Sjöstrand T, Mrenna S and Skands P 2009 Pythia `http://home.thep.lu.se/ torbjorn/Pythia.html`
[6] 2009 Geant4 `http://geant4.web.cern.ch/geant4/`
[7] Allison J *et al.* 2006 *IEEE Trans. Nucl. Sci.* **53** 270
[8] 2000 Standard Performance Evaluation Corporation `http://www.spec.org/cpu2000/`
[9] 2006 Standard Performance Evaluation Corporation `http://www.spec.org/cpu2006/`