Performance evaluation of computing resources with DIRAC interware

Cite as: AIP Conference Proceedings **2377**, 040006 (2021); https://doi.org/10.1063/5.0064778
Published Online: 24 September 2021

Igor Pelevanyuk

Performance Evaluation of Computing Resources with DIRAC Interware

Igor Pelevanyuk[a)]

Joint Institute for Nuclear Research, 6 Joliot-Curie, 141980, Dubna, Moscow region, Russia

[a)]*Corresponding author: pelevanyuk@jinr.ru*

Abstract. The evaluation of the performance of computing resources is important for workload distribution and workload planning. It is usually done with standard benchmark tools. The most well-known benchmark in grid infrastructures in High Energy Physic is HEP-SPEC06. This benchmark is based on industrial benchmark SPEC06. Running of HEP-SPEC06 takes a substantial amount of time but gives a precise estimation of the computing performance of one worknode. DIRAC interware uses the DB12 benchmark for performance estimation. It takes less than a minute to run and gives estimations very close to HEP-SPEC06. When a job is executed on a resource by DIRAC, DB12 benchmark test is performed and results are saved in the database. That means that there is no need to stop operations on computing resources to estimate full performance. It just requires to aggregate information about all individual DB12 results. Aggregation of benchmark results was done in JINR on all big computing components: Tier1, Tier2, Govorun supercomputer, Cloud, and NICA cluster. It did not require to send dedicated jobs since DIRAC was actively used for Monte-Carlo generation.

The studies in this direction were supported by the RFBR special grant (“Megascience – NICA”), №18-02-40101.

INTRODUCTION

Scientific studies today hugely rely on computing resources. For some of them, it is just a minor side tool to achieve results, but for some of them, computing is a crucial component on the path to discovery: Higgs Boson discovery [1, 2], SARS-CoV-2 modeling [3], the second proof of quantum computational advantage achieved recently [4], genome research, brain modeling, artificial intelligence study, and many other domains. The possibility to get results during research may depend on the performance of computing resources.

To complete massive computational tasks two types of systems are generally used: High-Performance Computing (HPC) and High Throughput Computing (HTC). HPC consists of many individual processors united by high throughput and low latency network. That allows HPC to calculate one task on many nodes in parallel mode. HTC also consists of many individual processors but requirements for the network are lower since the purpose of HTC is to complete a massive amount of individual jobs that do not rely on each other and do not need inter-process communication. HPC and HTC exist for different tasks. HTC tasks usually may be completed on HPC resources but not another way around.

The focus of this paper is the performance evaluation of computing resources in the scope of HTC computing. In the Joint Institute for Nuclear Research a wide range of studies is performed. Several computing components with different features were established to fulfill different needs in computing. Tier1 and Tier2 grid clusters, Govorun supercomputer, computing cloud, and LHEP Cluster (cluster of Laboratory of High Energy Physics). Tier1, Tier2, and LHEP clusters are HTC resources. Govorun supercomputer is an HPC resource. Cloud is used to run services and provide virtual machines but also may be used as an opportunistic resource.

All computing resources have different batch systems, authentication and authorization approach, quota, and some other features. That prevented their mutual consistent use, users had to integrate their workflows with all resources independently. To simplify mutual use of these resources they were integrated into DIRAC Interware. DIRAC Interware is an open-source platform originally developed for the LHCb experiment but later became a general-purpose product. DIRAC provides all necessary tools, services, and interfaces to interconnect heterogeneous resources allowing their consistent use.

DIRAC Interware was installed and configured in JINR [5]. All major computing resources in the institute were integrated into the system. That allowed to submit HTC jobs to all these resources in a similar way hiding most of the complexity from users. Up to this moment, more than a half-million of jobs were successfully executed. Total wall-clock time exceeds 350 years. To understand the capabilities of united resources it is important to know its overall performance as well as the performance of individual components.

Proceedings of the 24th International Scientific Conference of Young Scientists and Specialists (AYSS-2020)
AIP Conf. Proc. 2377, 040006-1–040006-4; https://doi.org/10.1063/5.0064778
Published by AIP Publishing. 978-0-7354-4132-3/$30.00

BENCHMARKS

To estimate the performance of a computer special programs exist. They called benchmarks. The goal of a benchmark is to provide a relative performance of an object by running a standard test. There are many aspects of performance in modern computers: CPU speed, RAM speed, disk i/o per second, and many others. Some benchmarks provide absolute numbers on different metrics: Flops, MB/s, reads/s, writes/s. Others provide just a number that represents relative performance on some artificial workload.

The main reason to get performance estimation with a benchmark is to predict how much time a real workload will consume. The most accurate way is to run a real workload. But the variety of different kinds of it may be big and it may take a lot of time to perform a complete set of estimations. That is why with benchmarks it is always a tradeoff between accuracy of the estimation and efforts to run it.

While writing a custom benchmark suite would be the best way of measuring the performance for HEP jobs, it would be too difficult to do. Physics code changes very frequently and support of benchmarks from the different experiments would take a lot of effort. This would also be harder for vendors to execute, as experiment code generally has more dependencies. So, for HEP applications industry-standard benchmark SPECcpu2006 [6] was taken and tuned to satisfy requirements [7]. Since HEPSPEC2006 is based on proprietary software, it is accessible only to resource administrators. It also requires a lot of time to run this benchmark, so resource administrators perform this test only if the structure of their resources has changed.

DIRAC Interware uses a benchmark called DIRACBenchmark2012 (DB12). It is an open-source benchmark that was designed as a general fast benchmark that may be used as a replacement of HEPSPEC2006 by different projects. It is written in Python and does not need any dependencies to be installed. Basic benchmark run takes less than a minute to run. This benchmark is used by DIRAC for two purposes: to estimate if the resource is fast enough to complete the job in time, and to estimate the number of consumed resources and normalize it across different resources [8].

PERFORMANCE ESTIMATION IN JINR

DIRAC uses a pilot mechanism to submit jobs to different resources. That means that if a user submits a job in DIRAC Job Queue, DIRAC would not send it directly to a resource. First, the pilot job will be sent. Pilot jobs go in the local job queue on a resource and wait to be executed. Once the pilot starts working on a worknode it will perform basic checks including the DB12 benchmark. After that, it will submit results to DIRAC and request a user job that could be completed by worknode with estimated characteristics. After the user job is completed pilot may either get another user job or it may finish its execution.

DB12 benchmark results received by pilots are mostly used to normalize CPU consumption across different resources for use in the DIRAC Accounting system. With that estimations, it is possible to compare the amount of work done by different resources. But in the accounting database, the data are aggregated and it is impossible to get detailed information about each particular DB12 run. For that information job database is used. That database provides current information about jobs running, waiting, failed, and done. By default data are deleted in that database after one week. Since we do not have so big a flow of jobs compared to the LHCb experiment, the period of keeping information about jobs was increased to one month in JINR.

There was no need to send some artificial jobs since at the time the DIRAC was heavily used by users. The jobs were related to the Monte-Carlo simulation. This kind of job is characterized by heavy use of CPU. For each job, the following information was recorded: DB12 result, the final status of the job, wall-clock time to compete job, hostname, processor model, and some other information. This data was saved from the database as a CSV file for the following analysis with Jupyter.

RESULTS

The question which was asked: does the DB12 result correlate with the execution time of the job? That is a very important question. Answer "yes" would allow us to use DB12 results to provide a general estimation of different resource CPU performance.

To see that the data were organized on the scatter plot in the following way. One axis represents the value of the DB12 benchmark. The second axis represents wall-clock time to complete a job. Every dot on the plot represents one job. The coordinates of a dot are chosen according to the information about the DB12 result and wall-clock time consumed by a job. The color of a dot represents the resource the job was running on. The described plot is in Fig. 1.

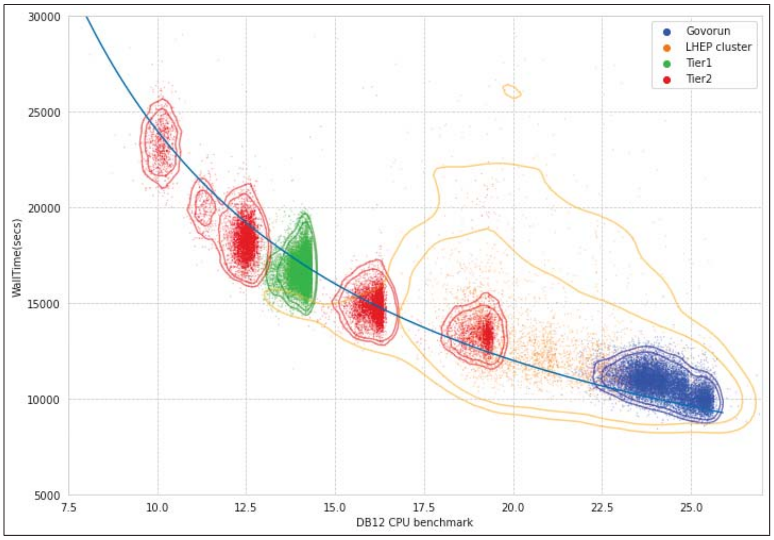

FIGURE 1. Correlation between DB12 benchmark and wall-clock time to complete a job

Computing resources Tier1, Tier2, and Govorun being in the same local network with the storage demonstrate a consistent correlation between DB12 and Wall-clock. Data related to the LHEP cluster is more scattered since at that time jobs were running not on disk but rather on a network file system. Another feature that appears on the plot is a clustering of jobs that run on Tier2. The reason for that is the heterogeneity of the Tier2 cluster itself. It consists of servers with different models of CPUs.

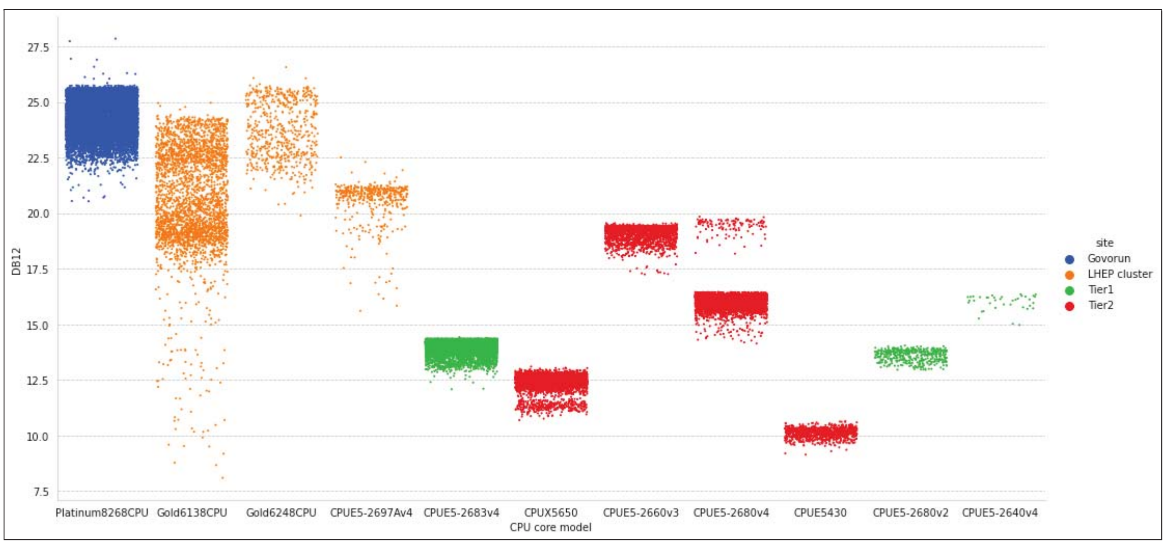

FIGURE 2. Results of single-core DB12 benchmark on different CPU models

The colored lines around dots on the plot showed to highlight the area where most of the jobs appeared. This area represents consistency between wall-clock time and the DB12 results. Inside the outer border, 95% of all jobs completed. Inside the inner border - 90% of all jobs completed. For successful jobs, the DB12 benchmark predicts wall-clock time with accuracy +/- 15%. That means that the DB12 benchmark could be used for performance evaluation. It is important to note, that wall-clock time includes the time that required to download initial data, perform calculation, and upload results.

The total CPU performance could be calculated by multiplying the average DB12 benchmark on the resource and amount of job slots on the computing resource. That does not mean that any workload will work with expected performance since all of them require different aspects of computer performance: disk speed, RAM speed, network, etc. But for CPU-intensive workload like Monte-Carlo generation, the estimation will give a precise estimation of jobs duration time.

The next step was to see how the DB12 benchmark varies between different processors. In addition to wall-clock time and DB12 results, there is information about the CPU model for each job. Most of the jobs which are completed now in DIRAC in JINR are single-core. So, only a single core performance will be evaluated. Different CPUs have different amounts of cores, and to get full performance estimation it is necessary to consider that.

The results are in the Fig. 2. And on this plot, the more deep characteristics of different computing resources appear. The Govorun supercomputer is the only resource with just one CPU model. LHEP cluster uses three types of processors, and single-core performance of them is either comparable with Govorun cores or almost the same (Gold6248). Tier1 on Fig. 1 looked homogeneous, but in fact, it uses three types of CPUs.

CONCLUSION

DB12 fast benchmark results correlate with the amount of time required to complete CPU intensive task like Monte-Carlo generation. Network operations, disk operations, and heavy use of RAM may increase execution time injecting dispersion to the Wall-clock time/DB12 plot. Monte-Carlo generation that was run by DIRAC users has a slightly variable execution time. For our case on some resources with the DB12 benchmark, it is possible to predict that time with accuracy +/- 15%.

DB12 benchmark only checks the speed of operations on the processor. The processor models and their single-core performance are visible on the DB12/CPU model plot demonstrated in the Fig. 2. It gives a good insight into computing performance on different resources.

All data used to create demonstrated plots were received from the inner DIRAC Interware database. That is an advantage since there is no need to send some heavy artificial workload to distributed resources and occupy it instead of useful user jobs.

REFERENCES

1. ATLAS Collaboration, Physics Letters B **716**, 1-29 (2012).
2. CMS Collaboration, Physics Letters B **716**, 30-61 (2012).
3. M. I. Zimmerman et al., bioRxiv, 10.1101/2020.06.27.175430 (2020).
4. H. Zhong et al., Science, 370 (6523), 1460-1463, 10.1126/science.abe8770 (2020).
5. V. Korenkov, I. Pelevanyuk, and A. Tsaregorodtsev, Communications in Computer and Information Science **1223**, 31-46 (2020).
6. *SPECcpu2006 official website*, https://www.spec.org/cpu2006/ (accessed December 1, 2020).
7. M. Michelotto et al., Journal of Physics: Conference Series **219**, 052009 (2010).
8. P. Charpentier, Journal of Physics: Conference Series **898**, 082011 (2017).