EUROPEAN ORGANIZATION FOR NUCLEAR RESEARCH
CERN – ACCELERATORS AND TECHNOLOGY SECTOR

CERN-ATS-2012-159

LHC@HOME: A VOLUNTEER COMPUTING SYSTEM FOR MASSIVE NUMERICAL SIMULATIONS OF BEAM DYNAMICS AND HIGH ENERGY PHYSICS EVENTS

M. Giovannozzi, A. Harutyunyan, N. Høimyr, P. L. Jones, A. Karneyeu, M. A. Marquina, E. McIntosh, B. Segal, P. Skands, CERN, Geneva, Switzerland
F. Grey, D. Lombraña González, Citizen Cyberscience Centre, CERN, Geneva, Switzerland
L. Rivkin, I. Zacharov, Particle Accelerator Physics Laboratory, EPFL, Lausanne, Switzerland

Abstract

Recently, the LHC@home system has been revived at CERN. It is a volunteer computing system based on BOINC which boosts the available CPU-power in institutional computer centres with the help of individuals that donate the CPU-time of their PCs. Currently two projects are hosted on the system, namely SixTrack and Test4Theory. The first is aimed at performing beam dynamics simulations, while the latter deals with the simulation of high-energy events. In this paper the details of the global system, as well a discussion of the capabilities of each project will be presented.

Presented at the International Particle Accelerator Conference (IPAC'12) –

May 20-25, 2012, N. Orleans, USA

Geneva, Switzerland, May 2012

LHC@HOME: A VOLUNTEER COMPUTING SYSTEM FOR MASSIVE NUMERICAL SIMULATIONS OF BEAM DYNAMICS AND HIGH ENERGY PHYSICS EVENTS

M. Giovannozzi, A. Harutyunyan, N. Høimyr, P. L. Jones, A. Karneyeu, M. A. Marquina, E. McIntosh, B. Segal, P. Skands, CERN, Geneva, Switzerland
F. Grey, D. Lombraña González, Citizen Cyberscience Centre, CERN, Geneva, Switzerland
L. Rivkin, I. Zacharov, Particle Accelerator Physics Laboratory, EPFL, Lausanne, Switzerland

Abstract

Recently, the LHC@home system has been revived at CERN. It is a volunteer computing system based on BOINC which boosts the available CPU-power in institutional computer centres with the help of individuals that donate the CPU-time of their PCs. Currently two projects are hosted on the system, namely SixTrack and Test4Theory. The first is aimed at performing beam dynamics simulations, while the latter deals with the simulation of high-energy events. In this paper the details of the global system, as well a discussion of the capabilities of each project will be presented.

INTRODUCTION

Today, volunteer computing provides many petaflops of processor power to a wide range of scientific and technical projects and is also a good outreach channel for scientific projects. The most famous example is Search for Extraterrestrial Intelligence (SETI) SETI@home [1] at the Berkeley Space Sciences Laboratory. This attracted so much volunteer interest and CPU power that one of its inventors, David Anderson, went on to develop open source middleware, the Berkeley Open Infrastructure for Network Computing (BOINC) [2,3], which enables hundreds of institutes or individual researchers to access large amounts of contributed computing power otherwise unavailable to them.

The BOINC model is based on a project server which sends out jobs on behalf of a BOINC project to all volunteer client nodes which have attached to the project. The project server receives results from the submitted jobs, validates them, and deals with issues of client down-time or unreliability. There is also a comprehensive system of project message boards to allow communication between volunteers and project staff, and *credit* is awarded to volunteers to encourage their collaboration. This credit is neither material nor financial, but nevertheless represents a major incentive to some volunteer communities, who form teams and compete actively for the maximum amount of credit.

Until recently, each BOINC project required considerable effort to set up, e.g., to port its computing application to the wide variety of BOINC volunteer platforms. These are typically Windows, but the number of Linux and Mac OSX systems is increasing. The next step is developing suitable job submission scripts to manage the flow of work provided by the project scientists to the volunteers, via the project's BOINC server. A key development in this sense has been adapting BOINC to support Virtualization [4, 5, 6], which has been successfully pioneered at CERN. The effort to broaden the use of volunteer computing for Particle Physics, and earlier experience with the Sixtrack Accelerator Physics code [7], has led CERN IT Department to re-establish a BOINC-based computing service.

The SixTtrack application server that was hosted since 2007 at Queen Mary University of London was repatriated ahead of a major CERN outreach event featuring LHC@home, held at the Frankfurt book fair in October 2011. This event also marked the collaboration with EPFL for developing the SixTrack BOINC application and, if possible, extending LHC@home to other accelerator physics applications.

The BOINC cluster set-up currently consists of 5 virtual server nodes: one hosts the classic BOINC application SixTrack [7, 8] and another hosts the Virtual Machine BOINC project Test4Theory [9, 10]. A third node is used as a web-server redirector to the main LHC@home portal.The two remaining nodes are reserved for tests and upgrades. The cluster structure is flexible and will be extended with additional nodes as necessary.

The application servers are running with a MySQL [11] back end for the BOINC application and forums. The database is currently located on the local machine hosting the project and backed up with database exports and the Tivoli Storage Management (TSM) backup system [12]. In order to improve reliability and performance, it is planned to move the MySQL database from the BOINC servers to the DB on Demand service [13].

THE SIXTRACK PROJECT

An example of such a traditional BOINC project is the SixTrack project, the starting point for the LHC@home system. In 2003, the first tests started of an in-house screensaver called the Compact Physics Screen Saver (CPSS), running SixTrack on desktop computers at CERN [14]. CPSS has proved to be a vital tool for debugging this kind of distributed computing application before moving to a BOINC-based system. This occurred in 2004 as part of CERN's 50th anniversary celebrations and attracted many volunteers and much media attention. At that time the main focus was on the impact of the superconducting magnets' field quality and the dynamic aperture (DA), namely the area in phase space stable up to a specified number of turns. Both single-particle numerical simulations, for up to 10^5 turns, as well as including beam-beam effects in the weak-strong description, for up to 10^6 turns, were performed. Some 60,000 users with about 100,000 PCs have been active LHC@home volunteers since 2004. This provided huge computing power for special, dedicated studies [15, 16], otherwise not possible on more conventional CERN computing facilities. More recently, the donated CPU time was used to review the DA value for the LHC using the detailed information gathered during the installation stage of the machine. Instead of using randomly assigned magnetic errors those corresponding to the machine as-built were used. This allowed the computation of the DA for the different configurations foreseen for the beam commissioning, as well as quantifying the improvement to the LHC performance resulting from the sorting algorithm applied to the main cryodipoles [17].

The application code SixTrack is FORTRAN-based and was ported to Windows and Linux in the classic way, by incorporating calls to the BOINC application programming interfaces (API) library and recompiling and re-linking the source code to produce BOINC executables for each client platform.

For this specific project, two crucial issues had to be solved: the so-called checkpoint/restart and the independence of the numerical results from the platform used. The first is needed to implement the BOINC client in a non-intrusive way. If the volunteer returns to the workplace while a task is being processed, the BOINC task will stop instantly and hand back control to the user. The next time the BOINC client starts up it will reconfirm the task assignment and if not instructed by the server to drop the task, it will resume its execution on the volunteer computer. The overall efficiency of the computing process is highly enhanced as occasional, or even frequent, interruptions of the desktop PC can be tolerated and allow extended runs of one million turns or more. The cost is minimal, as the SixTrack code will generate a state file of only 10KB for resuming computations.

The second point is considered a very difficult problem: the numerical result should not depend on details such as the compilers, on the different operating system (OS), and on any model of PC (e.g., 32-bit/MAC). A solution, working well in practice, was found [19], but recently a definitive and elegant solution has been developed and will be used in the further SixTrack releases [20]. The methodology developed for SixTrack in FORTRAN, but equally applicable to C/C++, has already been shown to provide identical results using four different compilers, on a wide range of PCs, using Linux, Windows or MacOS at any level of standard compliant optimisation. It has been suggested in the literature that this might well be impossible due to loopholes and the incompleteness of the standard. In the case of SixTrack, even a one Unit in the Last Place (1 ULP) difference in a double-precision result, will lead to an exponential growth of the difference during the LHC tracking. The final results are so different as to make comparison impossible. This 0 ULP methodology allows a simple comparison of the results and furthermore provides an outstanding test of hardware and software standard compliance. During the test phase, several hardware malfunctions have been detected and a compiler optimisation bug identified. Parenthetically, this methodology might well outreach to the computer games domain.

THE TEST4THEORY PROJECT

After seeing the interest generated by the SixTrack application, the question was raised whether CERN could benefit from BOINC to assist with its LHC physics research programme as well as for accelerator design.

As already mentioned, volunteer computers are one of the most heterogeneous resources for running any type of scientific experiments because of different operating systems, hardware architectures, versions of libraries, applications, operating systems, APIs, etc.

In order to ease the access to such heterogeneous resources, BOINC provides an API and a set of wrappers. However, researchers have to either port their existing applications to the supported BOINC programming languages (C/C++, FORTRAN and Java), or start from scratch a new version of the application, or provide a statically linked version of the application that could be used with one of the provided wrappers.

While Microsoft Windows is the most popular volunteer platform OS, the vast majority of LHC experiments' code only runs on UNIX or GNU/Linux systems. Furthermore, the LHC experiments have very large code repositories of libraries, applications, modules, etc. which are updated frequently. Therefore, porting all the source code to Windows machines in order to run them in BOINC would be quite impractical.

A solution to these problems is to use virtualization technology [21]. In a guest virtual machine (VM) it is possible to load an OS and all its software applications. The VM is handled by a Virtual Machine Monitor (VMM) or hypervisor which works at the kernel level of the host machine. This technology provides a homogeneous application layer across all the described platforms and OSes.

In 2006, the CERN IT Department began work on inclusion of VMs into BOINC, primarily as masters or summer student projects. Results were promising, but the large size of the resulting VM images containing typical LHC physics code was a major problem.

In 2008, a CERN R&D project called CernVM [4] was launched in the CERN Physics Department, offering a general solution to the problem of virtual machine image management for physics computing at the LHC experiments [5]. Essentially, CernVM is a minimum virtual machine appliance – less than 250 MB in size – that then downloads by itself all the required libraries for LHC applications, and also provides a secure gateway to the experiments' own job management systems [6].

It was quickly realized that this CernVM technology could be incorporated into BOINC using a "CernVM Wrapper" - a modified version [25] of the official BOINC wrapper. This wrapper is in charge of uncompressing the

VM after it has been downloaded to the client, launching it and controlling it. For example, it will pause or resume the VM when the BOINC task has to be paused or resumed to follow the work flow of BOINC. This solution preserves the standard BOINC project model.

After a joint effort by the CernVM team and CERN-IT, the first BOINC project using virtual machines was born in 2011: the LHC@home Test4Theory project [10]. A particularly interesting feature of this system is that the BOINC-CernVM volunteer machines appear as *cloud nodes* to the application scientists [26]. Based on the described technology, a new web-based resource for validation of Monte Carlo (MC) models of high-energy collider physics has been established at CERN [22]. The aim of the web site [9] (MCPLOTS) is to provide a simple online repository of plots, comparing the most widely used MC event-generator tools (currently ALPGEN, HERWIG++, PYTHIA 6 & 8, SHERPA, and VINCIA, see [23]) to a large database of experimental results, encoded in the RIVET analysis tool [24]. The repository is continually updated, and the computing power is supplied by volunteers via Test4Theory.

The CernVM has allowed a very clean factorization between the physics software and the architecture of the volunteer machines. Another important aspect is a very high CPU-to-bandwidth ratio. The typical job will run for hours with zero load on the volunteer's network bandwidth (the events being analysed are simulated locally on the volunteer computer itself), and only a very small number of bytes is required to send job I/O back and forth at the start and end of jobs. After downloading the initial install package each volunteer machine only needs to be sent a single ASCII parameter card at the start of each simulation run, and the output sent back to the MCPLOTS server is likewise a single ASCII histogram.

At present, the volunteers connected to the Test4Theory project have generated a total of more than 300 billion collision events, providing a large amount of simulated MC statistics to compare to the experimentally measured reference distributions. The data come from a large range of collider experiments, including four at LHC (ALICE, ATLAS, CMS, LHCb), three at LEP (ALEPH, DELPHI, OPAL), two at SPS (UA1, UA5), two at the Tevatron (CDF, D0), and one at the RHIC (STAR), with both more generators and new data continually being added.

OUTLOOK AND FUTURE PLANS

Two different types of BOINC projects are currently operated at CERN. SixTrack is successfully running and producing data for beam dynamics studies for the LHC and eventually its upgrade. This project is using the standard BOINC architecture and relies on adapting the tracking framework to the BOINC philosophy. The experience with this project hopefully opens the path to porting other beam dynamics codes to BOINC.

The Test4Theory project is the first example of particle physics application using the computing power offered by volunteer computing. The strategy, in this case, is based on virtualisation technology successfully developed at CERN. As the virtual machine approach to Volunteer Cloud computing potentially opens up for a wider range of physics applications, the current project-based BOINC service could eventually become a general service to host volunteer computing applications at CERN.

ACKNOWLEDGMENT

We would like to express our gratitude to all volunteers that have subscribed to LHC@home, thus providing essential resources to our studies.

REFERENCES

[1] http://setiathome.berkeley.edu/
[2] D. Anderson, in proceedings of 5th IEEE/ACM International Workshop on Grid Computing, 4, 2004.
[3] BOINC: http://boinc.berkeley.edu/
[4] P. Buncic *et al.* , Journal of Physics: Conference Series, **219**, 042003, 2010.
[5] A. Harutyunyan *et al.*, Journal of Physics: Conference Series, **219**, 072036, 2010.
[6] A. Harutyunyan *et al.*, Journal of Physics: Conference Series, 331, 2011.
[7] F. Schmidt, CERN-SL-94-56, 1994.
[8] http://lhcathomeclassic.cern.ch/sixtrack/
[9] http://mcplots.cern.ch
[10] http://lhcathome2.cern.ch/test4theory/
[11] MySQL: http://www.mysql.com/
[12] TSM backup system: http://www-01.ibm.com/software/tivoli/products/storage-mgr/
[13] DB on Demand: http://cern.ch/dbondemand
[14] E. McIntosh, A. Wagner, CERN-2005-002, 1055.
[15] W. Herr, D. I. Kaltchev, E. McIntosh, F. Schmidt, LHC-Project-Report-927, 2006.
[16] M. Giovannozzi, E. McIntosh, LHC-Project-Report-925, 2006.
[17] S. Fartoukh, M. Giovannozzi, Nucl. Instrum. & Methods A **671** 10, 2012.
[19] E. McIntosh, F. Schmidt, F. de Dinechin, "Massive Tracking On Heterogeneous Platforms", 9th International Computational Accelerator Physics Conference, 2006.
[20] E. McIntosh, "IEEE-754 as intended - or how to obtain identical results with standard compliant compilers and hardware", in preparation.
[21] J. E. Smith and R. Nair, *Virtual Machines*, (S. Francisco: Morgan Kaufmann, Elsevier, 2005).
[22] P. Skands *et al.*, "MC production statistics", http://mcplots-dev.cern.ch/production.php
[23] A. Buckley et al., Phys.Rept. **504**, 145, 2011.
[24] A. Buckley et al., "Rivet user manual", arXiv:1003.0694 [hep-ph].
[25] D. Lombraña-González, "CernVM wrapper", Technical report, Citizen Cyberscience Centre, Online 2012. https://github.com/citizen-cyberscience-centre/cernvmwrapper
[26] B. Segal *et al.*, Proceedings of the XIII International Workshop on Advanced Computing and Analysis

Techniques in Physics Research (ACAT10), Jaipur, 2010.