ATL-SOFT-PROC-2017-049
14 February 2017

Evolution and experience with the ATLAS Simulation at Point1 Project

S Ballestrero[1], F Brasolin[3], D Fazio[2], A Di Girolamo[2], T Kouba[5], C J Lee[2,9], D A Scannicchio[4], J Schovancová[2], M S Twomey[6], F Wang[7], A Zaytsev[8]

[1] University of Johannesburg, South Africa
[2] CERN, Switzerland
[3] Istituto Nazionale di Fisica Nucleare Sezione di Bologna, Italy
[4] University of California, Irvine, USA
[5] Institute of Physics, ASCR, Czech Republic
[6] University of Washington Department of Physics, USA
[7] University of Wisconsin-Madison, USA
[8] Brookhaven National Laboratory (BNL), USA
[9] University of Cape Town, South Africa

Email: Tomas.Kouba@cern.ch, atlas-adc-sim-p1@cern.ch

Abstract. The Simulation at Point1 project is successfully running standard ATLAS simulation jobs on the TDAQ HLT resources. The pool of available resources changes dynamically, therefore we need to be very effective in exploiting the available computing cycles. We present our experience with using the Event Service that provides the event-level granularity of computations. We show the design decisions and overhead time related to the usage of the Event Service. The improved utilization of the resources is also presented with the recent development in monitoring, automatic alerting, deployment and GUI.

1. Introduction

During the LHC [1] Long shutdown 1 (LS1) period, the Simulation at Point1 (Sim@P1) project utilized the TDAQ HLT [2] (Trigger and Data Acquisition system, High Level Trigger) resources very well in a continuous manner. The design decisions, the performance observed and the results of the commissioning of the project are described in [3].

LHC Run 2 period presented slightly different challenge as the Simulation at Point1 project successfully continued in using the resources reserved for Sim@P1 and we focused on better utilization of the resources given to Sim@P1 during the periods when the resources are not needed for TDAQ HLT computing. Such periods are usually Machine Development (MD) or Technical Stop (TS) periods which usually last several days. We also deployed a Web User Interface (so called "red button") to ease the switch the resources from/to TDAQ and Sim@P1 role.

There was no major change in the cloud setup and the major version of the operating system setup for Sim@P1. The software stability is a key aspect for TDAQ HLT smooth running, therefore we stayed on SLC6 as the operating system and OpenStack in version "Icehouse".

So far the overall performance of the Sim@P1 was very good during the Run 2. In terms of generated and simulated events the performance of Sim@P1 is comparable to the performance of the largest tier one (T1) centers in the WLCG [4].

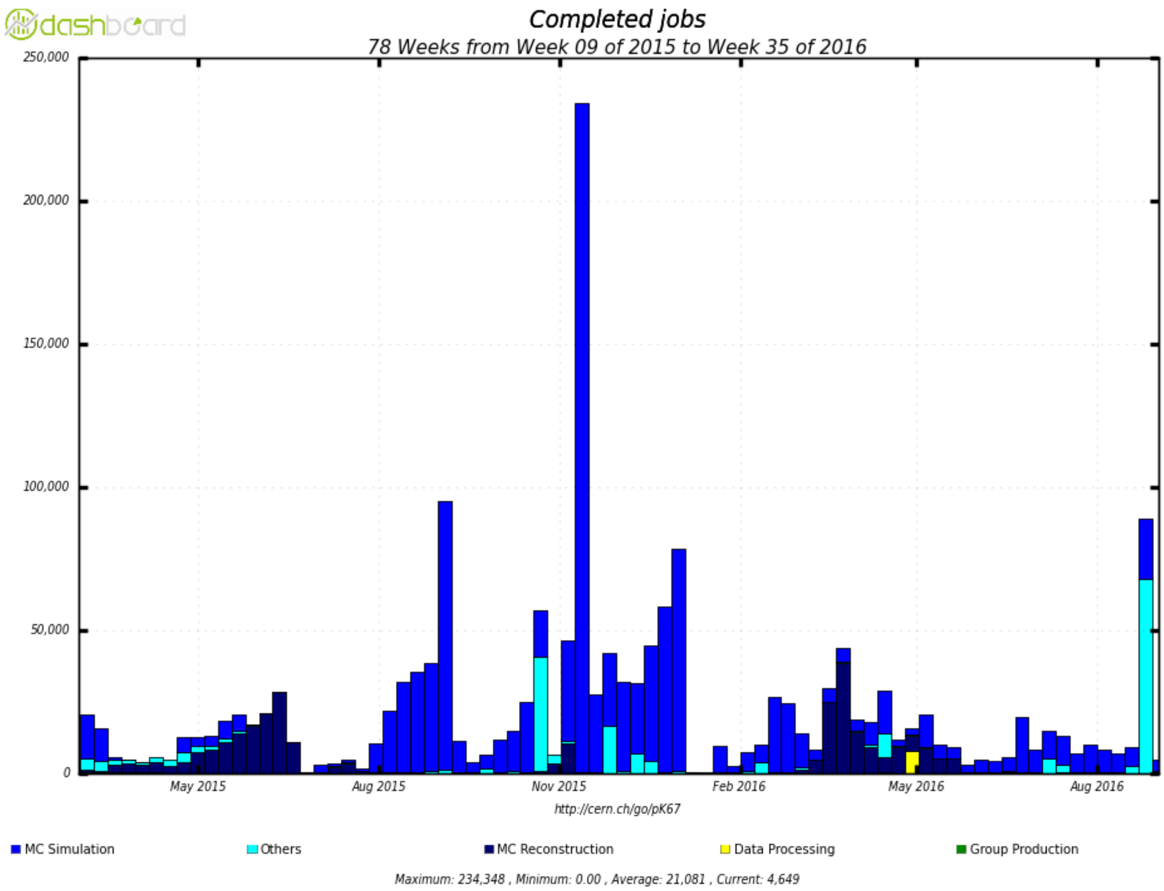

Figure 1 – Sim@P1: Jobs completed since March 2015

The Figure 1 shows the number of completed jobs since March 2015. Each column represents one week. The various colors represent different job types.

The Figure 2 depicts the biggest portions of all simulation jobs processed by each ATLAS computing site. The chart shows that Sim@P1 contributed more than 12% of all the events simulated for ATLAS since March 2015. Please note that the pie chart in the Figure 2 is limited only to simulation jobs and other sites are not dedicated to simulation only jobs (Sim@P1 is exclusively used for simulation jobs except short periods of tests of other job types – as seen on Figure 1).

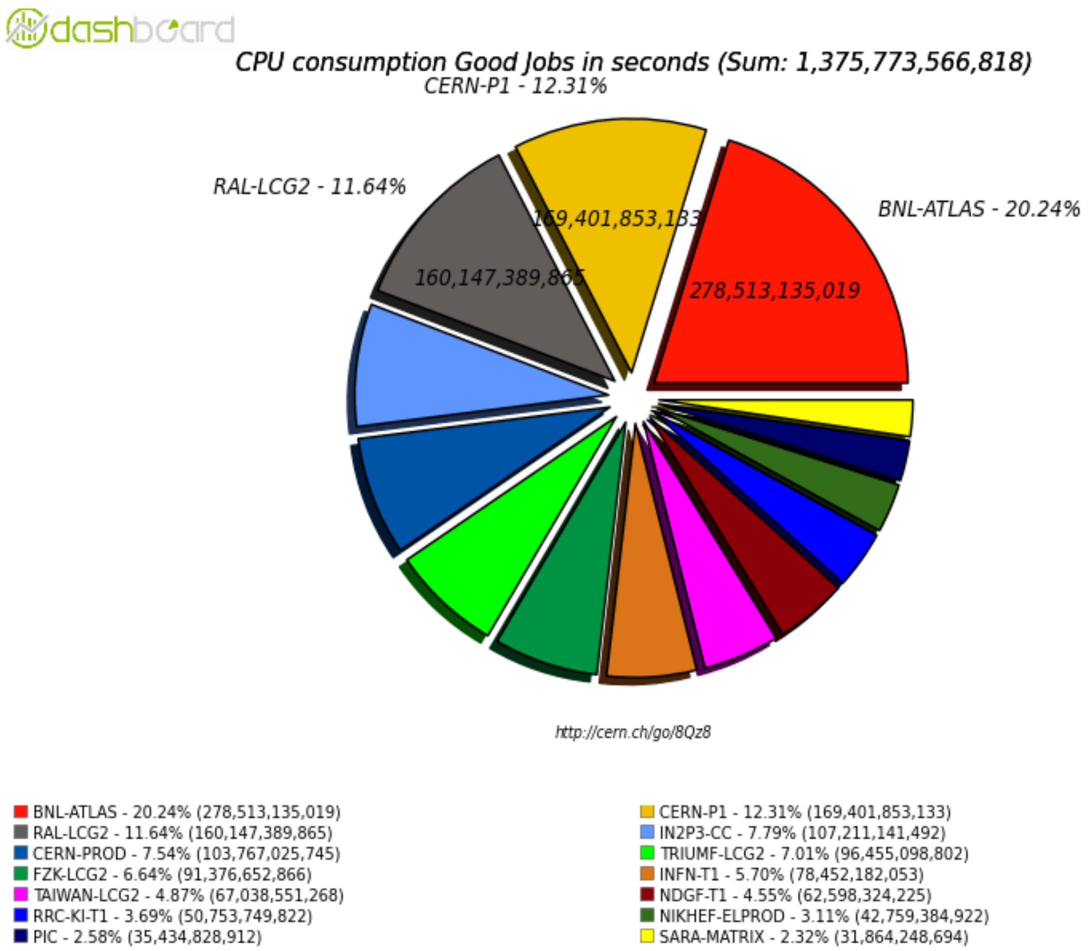

Figure 2 - Simulation jobs CPU consumption around the grid since March 2015

2. Event Service pilot test

As described in [5] “the ATLAS Event Service (ES) implements a new fine grained approach to HEP event processing, designed to be agile and efficient in exploiting transient, short-lived resources”. This perfectly fits for the short (i.e. hours) bursts in available CPUs in Sim@P1. These short periods are not long enough to finish a regular grid job but still long enough to process significant number of events. The regular simulation jobs will fail and lose all progress when the resources vanish, but the Event Service jobs with the event-by-event processing and staging-out can minimize the loss of the resources changes.

The Sim@P1 project has piloted in the Event Service (ES) test as the first opportunistic site.

Many feedbacks and suggestions have been provided to the Event Service team during the test, which helped to improve its performance and fix bugs.

The Figure 3 below shows the amount of lost jobs when the resources are taken back from Sim@P1 to the regular TDAQ HLT operation.

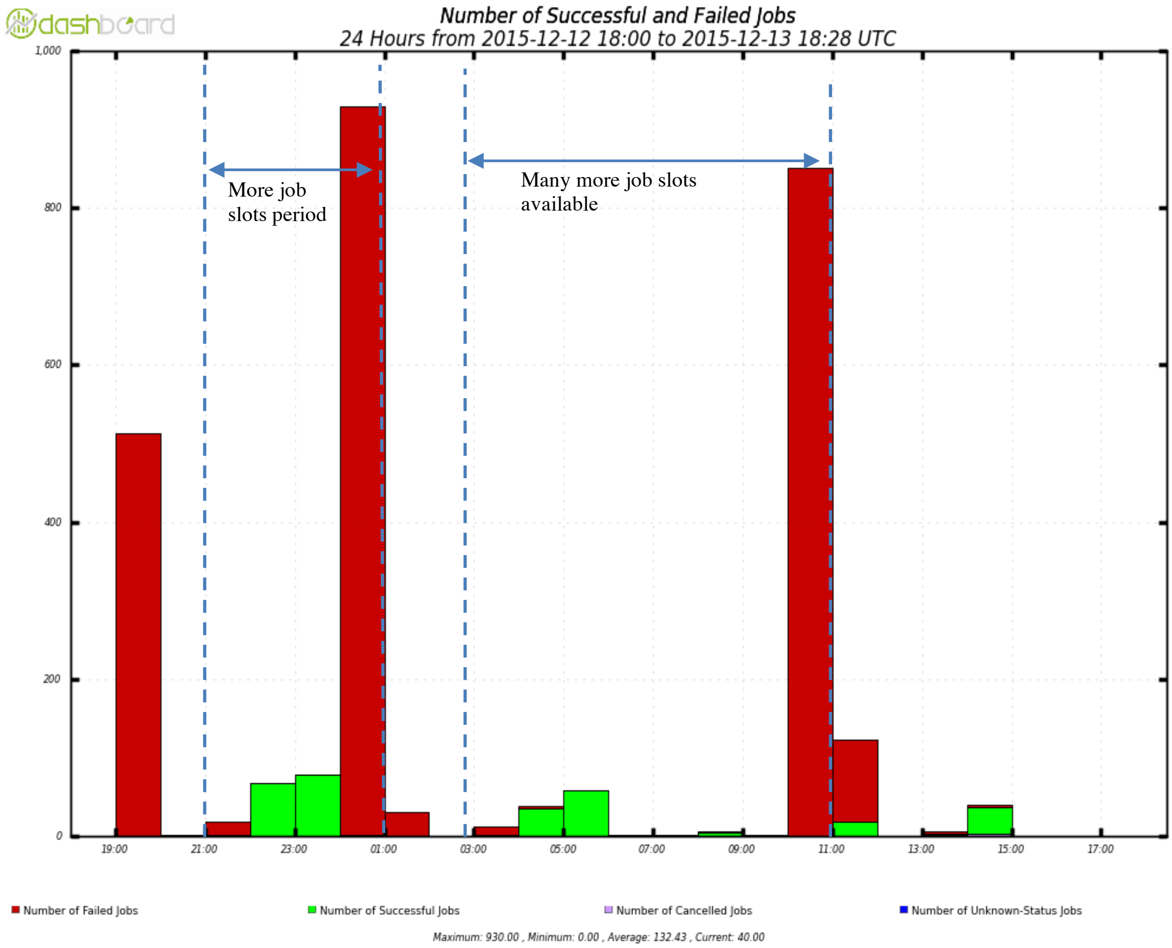

Figure 3 - Failed jobs in regular simulation tasks for short periods

3. Automated switcher: "red button"

The process of switching the resources from TDAQ HLT to Sim@P1 role is quite complex and needed a supervision of TDAQ experts. To exploit short periods of time when TDAQ servers are available for simulation purposes we needed a mechanism that would allow shifters to easily switch the role of the resources (rack by rack) from TDAQ to Sim@P1 and back. Such mechanism must be easy to use for shifters, granular enough to allow switch the role rack by rack and very robust so the resources do not stay in undefined state (somewhere between Sim@P1 and TDAQ). Another critical feature of the whole switch is that the resources must be safely and quickly released by Sim@P1 (even on short notice) when they are needed by TDAQ.

The web interface implementing all these required features is shown in figure 4.

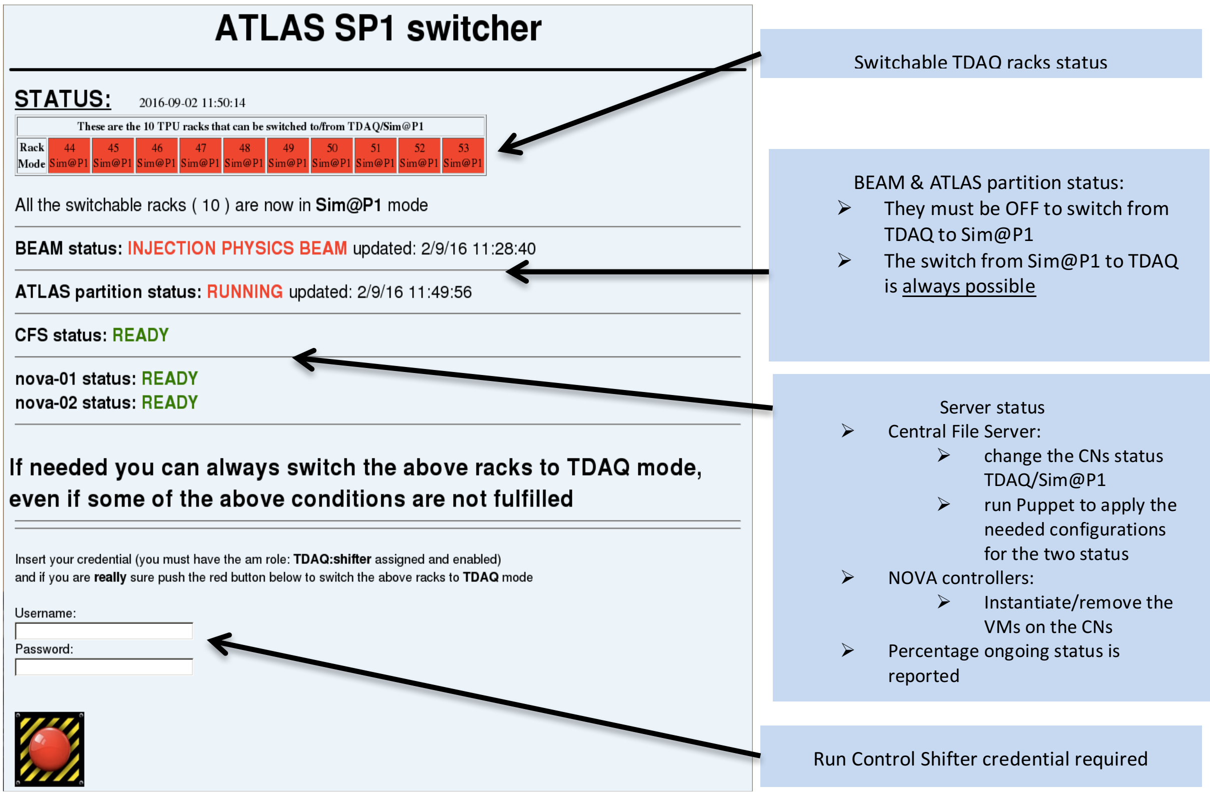

Figure 4 - Web interface for the "red button"

4. Monitoring developments

The monitoring of all the Sim@P1 related activities had been consolidated on one live page [6] and the Sim@P1 responsibles are notified if the utilization (ratio between existing job slots and running ATLAS jobs) drops below 80%. These two values are depicted in figure 5.

During the year 2016 we tested using the Sim@P1 resources not only for simulation jobs but we tried reconstruction and reprocessing jobs too. These jobs brought higher I/O load to the nodes. This triggers notification to TDAQ administrators and also to the Sim@P1 administrators. We have not yet fully identified the exact root cause of the high I/O so we went back to simulation jobs only at the end of 2016.

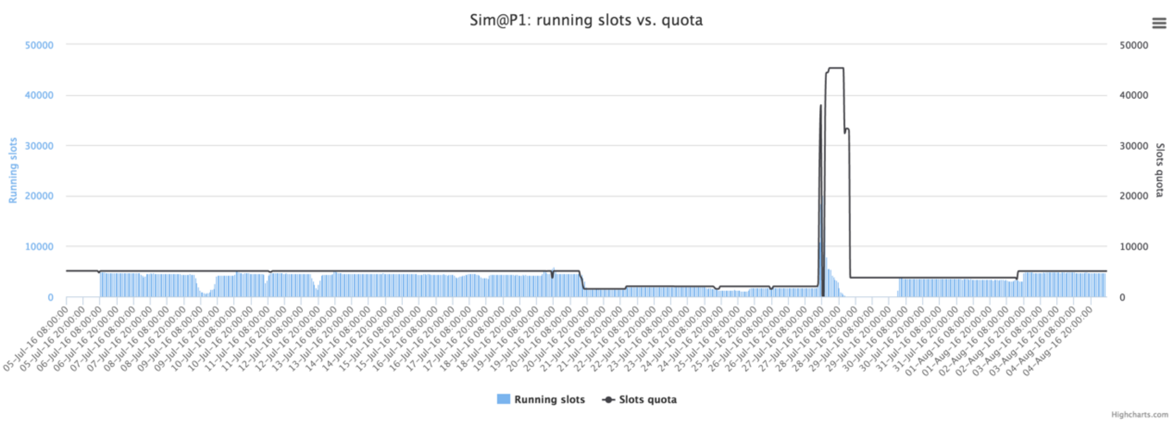

Figure 5 - Available resources (job slots) vs running jobs, Jul-Aug 2016

5. Dynamic partitioning and Cgroups

We switched from static HTCondor slots (8 CPU, no RAM limit) to dynamically allocated slots of two types (8 CPU, 8GB RAM + 8 CPU, 15 GB RAM) and one additional type (1 CPU, 2 GB RAM) for single core jobs (particularly merge jobs).

The limits are enforced with Linux Cgroups [7] with the goal not to overcommit the memory usage on the virtual machines and so not to trigger swap usage on the hypervisors outside of VMs. This helped to test the reconstruction jobs that often had caused memory exhaustion. However, we still see heavy I/O load caused by the reconstruction jobs. This needs to be fully understood before we allow more reconstruction and reprocessing jobs to be run at Sim@P1.

6. Summary

During the LHC Run2 the Sim@P1 project continued to deliver a significant portion of simulated events for ATLAS. The resources were also used for testing new activities in ATLAS distributed computing (Event Service, new version of Auto Pilot Factory [8]).

There was no significant upgrade in the infrastructure as it needs to be kept stable for correct TDAQ HLT operations. Any discussed upgrades were postponed to LS2. This includes OpenStack version upgrade which requires operating system upgrades (from SLC6 to CC7). Such major upgrade needs careful planning as it involves also validation of ATLAS TDAQ software.

The future plans are now focused mainly on resolving spurious problems with high I/O load caused by grid jobs on the TDAQ nodes and on commissioning Event Service jobs from test phase to production.

We are also aware of the suboptimal settings of the virtual machines in terms of memory. The Non-Uniform Memory Access (NUMA) topology information is lost between the hardware and our Sim@P1 virtual machine. This can be fixed again in the recent version of OpenStack with more modern version of the Libvirt [9] library.

References

[1] Brüning O S, Collier P, Lebrun P, Myers S, Ostojic R, Poole J, Proudlock P, LHC Design Report, ISBN 92-9083-224-0

[2] ATLAS Collaboration, ATLAS High Level Trigger, Data Acquisition and Controls: Technical Design Report, CERN/LHCC/2003-022 (2003), ISBN 92-9083-205-3

[3] Zaytsev A *et al.*, Design, Results, Evolution and Status of the ATLAS Simulation at Point1 Project

[4] Worldwide LHC Computing GRID: http://wlcg.web.cern.ch

[5] Calafiura P, De K, Guan W, Maeno T, Nilsson P, Oleynik D, Panitkin S, Tsulaia V, Van Gemmeren P, Wenaus T, The ATLAS Event Service: A new approach to event processing, J.Phys.Conf.Ser. 664 (2015) no.6, 062065

[6] ATLAS Sim@P1 processing overview, http://adc-mon.cern.ch/SimAtP1/ATLAS-SimAtP1-live.html

[7] Introduction to Cgroups, https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Resource_Management_Guide/ch01.html

[8] Caballero J, Hover J, Love P, Stewart G, AutoPyFactory: A Scalable Flexible Pilot Factory Implementation, ATL-SOFT-PROC-2012-045, ATL-COM-SOFT-2012-021

[9] Libvirt virtualization API, https://libvirt.org/