PROCEEDINGS OF SCIENCE

Software Upgrades for the High-Luminosity LHC

David R. Shope[a,*] **on behalf of the ALICE, ATLAS, CMS, and LHCb collaborations**

[a] *Department of Physics, University of Oslo,*
Sem Sælands vei 24, 0371, Oslo, Norway

E-mail: david.richard.shope@cern.ch

The High Luminosity upgrade of the LHC (HL-LHC) presents a fresh set of challenges that will need to be overcome in order to fully utilize the prodigious amount of data set to be collected throughout the course of its operation. These proceedings provide an overview of the various ongoing efforts to prepare the software of the ALICE, ATLAS, CMS, and LHCb collaborations for this exciting new era of research. Center stage to these developments are the topics of accelerator technologies and machine learning algorithms, both of which have seen a tremendous amount of activity and progress in recent years. Cross-experiment R&D projects are also highlighted, in addition to software that must be written for new sub-detectors as well as for the support of legacy data.

12th Large Hadron Collider Physics Conference (LHCP2024)
3-7 June 2024
Boston, USA

*Speaker

PoS(LHCP2024)193

1. Introduction

The High Luminosity phase of the Large Hadron Collider (HL-LHC) is due to begin operation in the late 2020s and run until the early 2040s. During this time, the ATLAS and CMS experiments are expected to have collected over ten times more data than they will have recorded during the first three LHC runs. In order to achieve this unprecedented milestone, ATLAS and CMS will operate at an instantaneous luminosity which is over three times higher than today, with a peak leveled pileup of 200 (up from 30-60 currently). Given these conditions, it's clear as illustrated in Figure 1 that computing resources will fall short unless significant R&D occurs. The following HL-LHC

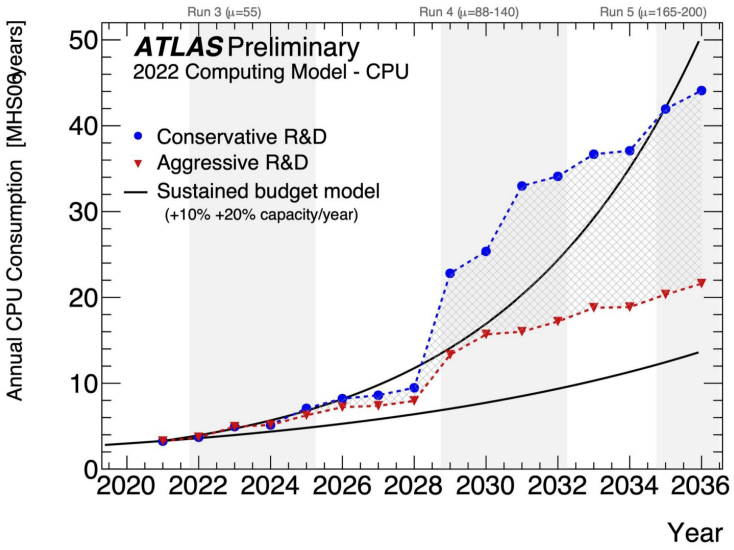

Figure 1: Projected evolution of ATLAS compute usage from 2020 until 2036, under the conservative (blue) and aggressive (red) R&D scenarios [1].

software upgrades fall into several broad themes:

- **Adapting to heterogeneous platforms** - GPUs offer large parallel processing capabilities and excel at tasks such as the training of deep learning models. However, the paradigm for programming these devices differs significantly from classical x86 processors and supporting a heterogeneous set of accelerator technologies therefore also represents a major software development challenge.
- **New approaches with Machine Learning (ML)** - ML algorithms have long since been used in HEP, although the recent rapid advancements in hardware and algorithms are proving to play a central role for many improvements in nearly all areas of the field.
- **Cross-experiment R&D projects** - the sharing of common software between different experiments will free up resources that were previously spent on duplicate solutions to problems observed with multiple detectors.
- **"Meat-and-potatoes" developments** - this category encompasses all of the needed developments that follow from the installation of new sub-detectors and includes the detector

descriptions in simulation, digitization (electronics simulation including extensive modeling of radiation damage), and associated reconstruction algorithms.

2. Software Upgrades By Domain

2.1 Triggering

During HL-LHC operation, ATLAS and CMS will receive a ~threefold increase in both trigger rate and data size per event, necessitating significant trigger upgrades (both in hardware and software) in order to cope with such conditions. Currently, custom hardware triggers reduce the data rate from 40 MHz down to $O(100)$ kHz, while heterogeneous computing farms lower it further to $O(1)$ kHz through partial reconstruction and triggering of the events in software. The direction of research in this domain lies in the further utilization of hardware accelerators such as GPUs and FPGAs, which implies a significant change in the way that the associated software is written. In CMS, GPU-enabled reconstruction in the High-Level Trigger (HLT) since the start of Run 3 has reduced execution time by ~40% while simultaneously reducing power consumption by 30% [5, 6]. ALICE and LHCb, on the other hand, have successfully deployed a model in which hardware triggers have been removed since the start of Run 3. In the case of ALICE, the triggers have been removed entirely, leading to a continuous readout with online compression of the 50 kHz Pb-Pb raw data in software and an online computing farm in which 95% of the processing workload runs on GPUs [2, 4]. For LHCb, a fully software-based trigger system has been employed, running at the full 30 MHz readout rate of the detector [3, 4]. In order to further investigate innovative computing technologies (both hardware and software) in the design of future data acquisition strategies, a project by the name of NextGen Triggers has been established this year and is expected to play a significant role in shaping the future of triggering at the LHC [7].

2.2 Event Generation

Monte Carlo event generators are projected to use 10-20% of the computing resources at the HL-LHC [1]. Run 4 will see the need for both high-statistics inclusive samples as well as the efficient population of exclusive phase-spaces, all while maintaining the best available accuracy. Many avenues for speeding up the generation of events are being explored, as outlined by the HEP Software Foundation (HSF) Generator Working Group [10]. For example, the MadGraph4GPU project aims to speed up the matrix element calculation in MC5aMC on GPUs and vector CPUs [11] and has successfully demonstrated up to an 80-fold increase in total throughput for generating pairs of top quarks with additional gluon emission using an NVidia Tesla A100 GPU. Efforts to parallelise the entire parton-level event generation are also underway, including the Pepper project [13] which emphasises portability between different architectures through the use of the Kokkos programming model [12].

2.3 Detector Simulation

Detector simulation is (today) the largest CPU consumer on the computing grid, with time overwhelmingly spent in the calorimeter [15]. The careful optimization of the Geant4 parameters in full detector simulation has proven fruitful in reducing its CPU consumption, with a factor of

~two improvement in speed observed in both ATLAS and CMS since Run 2 [21]. GPUs also offer a promising solution to the problem of slow simulation, with two projects under development for electromagnetic particle transport - AdePT [23, 24] and Celeritas [22, 25] - and LHCb pursuing options such as Opticks [26] and Mitsuba3 [27] for the simulation of optical photons.

Despite these improvements in the full simulation of detectors, producing physics-accurate simulations in a fraction of the current time will be critical for HL-LHC programs. Traditionally, fast simulation methods have relied on parameterizations of the detector response using e.g. principal component analysis (PCA). In recent years, generative ML techniques have shown significant promise as a replacement [19]. For example, ALICE has established an impressive 100x speed-up of the Zero Degree Calorimeter using Variational Auto-Encoders (VAEs) and Generative Adversarial Networks (GANs) [14], while ATLAS has employed GANs for calorimeter simulation [16] and coupled with a Fast ATLAS Track Simulation (FATRAS) is able to achieve an order of magnitude improvement in speed as compared with full simulation of the detector [17].

Parameterization of the high level response of a detector can also enable a fast reconstruction. For example, CMS is exploring the usage of normalizing flows (generative models for PDFs) in a workflow referred to as Flashsim in order to produce their reconstructed event format directly usable for downstream physics analysis directly from generated events [18]. LHCb is also developing LAMARR, a Gaudi-based framework for ultra-fast simulation which currently uses Gradient Boosted Decision Trees (GBDTs) for efficiency modeling and a GAN for reproducing high-level physics distributions [20].

2.4 Event Reconstruction

Historically, reconstruction software has largely been developed independently by each experiment. In recent years, there's been a shift towards common software written to be adaptable to multiple experiments. One example of this is A Common Tracking Software Project (ACTS) [28, 29], which offers an experiment-independent toolkit for the reconstruction of charged particle trajectories with a core implementation based on modern (C++) and thread-safe code. Work is also ongoing to interface ACTS with accelerators through the traccc project [30], which has the goal of mapping procedural (non-ML) tracking algorithms such as the Combinatorial Kalman Filter (CKF) onto GPUs without algorithmic compromises. ML solutions have also been explored for track reconstruction, for example the GNN4ITk project aims to apply a Graph Neural Network (GNN) to the problem of track finding in the ATLAS Inner Tracker (ITk) to be installed for Run 4. A first look at the physics performance of the GNN-based pipeline compared with the CKF algorithm demonstrates that the GNN provides competitive tracking efficiency, as well as a high quality track parameter resolution [31].

Due to the increase in event pileup during HL-LHC operation, a significant contamination of the hard-scatter interaction with uncorrelated activity would severely degredate physics performance with current detectors and algorithms. One common solution to mitigate this issue is the introduction of new sub-detectors with enhanced timing capabilities (often with time resolution $\mathcal{O}(10\text{ ps})$) [8, 9]. Fully exploiting this additional timing information requires in many cases the redesign of existing reconstruction algorithms.

2.5 Data Analysis

Analysing the vast amount of data that will be available at the HL-LHC is itself a challenge, as current workflows do not scale. Even the storage of the data is an issue if adaptation isn't made, due to the limited availability and cost of disk/tape resources. Several efforts are ongoing to make the data more compact and easier to process downstream, including an improved encoding in ROOT's RNTuple [32, 35] and smaller (~10 kb/event) data formats such as PHYSLITE in ATLAS [33] and NanoAOD in CMS [34]. Simultaneously, a clear trend is also emerging (aided by a budding ecosystem of associated python libraries) in how events are processed - namely through a new paradigm known as *columnar analysis* [36], which proceeds through operations on arrays for batches of events (as opposed to one event at a time in an event loop).

In the future, it may become possible for data analysis to be a highly interactive experience, minimizing the time it takes to iterate between ideas and plots [37]. As a result, the concept of an *analysis facility* [38] is taking shape, which would feature several key characteristics: data consolidation at the site, accessibility to those outside the site, and the ability to perform both a distributed and interactive analysis style with all the tools readily availabie (including ML resources).

3. Summary and Outlook

Meeting the challenges brought on by the HL-LHC requires not only hard work and innovation in hardware, but in software as well. It's critical that baseline upgrade-related software workflows are already in place today, as they inform studies impacting new detectors as (or before) they are built. Many promising ideas for leveraging new technologies in the areas of heterogeneous platforms and machine learning applications are actively being pursued. In particular, Large Language Models (LLMs) have seen an explosion in popularity in recent years and are being adapted for not only conventional natural language use-cases in HEP such as the summarization of collaboration documentation and the auto-parsing of log files, but also to process scientific data e.g. during particle reconstruction [43].

As we look toward the future, it's important at the same time to provide support for legacy data. Firstly, this entails compliance with the CERN Open Science policy [39] by providing open data to the public [40, 42] and with the preservation of analysis through the reproduction of workflows using tools such as REANA [41]. In addition, consideration must also be given to the feasability of running old data and MC in the latest version of software. On the one hand, this potentially poses a real challenge if older data can no longer be accessed and utilized. On the other hand, major investment is required for maintenance, including for detectors that are no longer taking data.

References

[1] The ATLAS Collaboration, "ATLAS Software and Computing HL-LHC Roadmap," Tech. Rep. CERN-LHCC-2022-005, LHCC-G-182, CERN, Geneva, 2022. https://cds.cern.ch/record/2802918

[2] P. Buncic, M. Krzewicki, and P. Vande Vyvre, "Technical Design Report for the Upgrade of the Online-Offline Computing System", CERN-LHCC-2015-006, ALICE-TDR-019, 2015, https://cds.cern.ch/record/2011297.

[3] The LHCb Collaboration, "The LHCb Upgrade I", Journal of Instrumentation, vol. 19, no. 05, P05065, May 2024, arxiv:2305.10515, doi:10.1088/1748-0221/19/05/p05065.

[4] M. Zanetti, "Novel triggering strategies (HW and SW) at the HL-LHC." https://indico.cern.ch/event/1253590/contributions/5843734/, 2024.

[5] G. Parida, "Run-3 Commissioning of CMS Online HLT reconstruction using GPUs." https://indico.jlab.org/event/459/contributions/11822/, 2023.

[6] S. Donato, "Trigger performance (including data scouting and GPU) at CMS and ATLAS." https://indico.cern.ch/event/1253590/contributions/5832657/, 2024.

[7] "NextGen Trigger website." https://nextgentriggers.web.cern.ch/

[8] T. Evans, "Detectors with timing capabilities." https://indico.cern.ch/event/1253590/contributions/5843726/, 2024.

[9] S. Griso, "Pileup suppression with timing detectors." https://indico.cern.ch/event/1253590/contributions/5832662/, 2024.

[10] The HSF Physics Event Generator WG., Valassi, A., Yazgan, E. et al. Challenges in Monte Carlo Event Generator Software for High-Luminosity LHC. Comput Softw Big Sci 5, 12 (2021). https://doi.org/10.1007/s41781-021-00055-1

[11] S. Hageboeck, "Madgraph5_aMC@NLO on GPUs and vector CPUs: experience with the first alpha release." https://indico.jlab.org/event/459/contributions/11829/, 2023.

[12] "Kokkos website." https://kokkos.org/kokkos-core-wiki/

[13] E. Bothmann, "Pepper – A Portable Parton-Level Event Generator for the High-Luminosity LHC." https://indico.cern.ch/event/1330797/contributions/5791236/, 2024.

[14] J. Dubiński, K. Deja, S. Wenzel, P. Rokita, and T. Trzciński, "Machine Learning methods for simulating particle response in the Zero Degree Calorimeter at the ALICE experiment, CERN", 2023, arxiv:2306.13606.

[15] D. South, "R&D in ATLAS Distributed Computing towards HL-LHC." https://indico.jlab.org/event/459/contributions/11502/, 2023.

[16] The ATLAS Collaboration. Deep Generative Models for Fast Photon Shower Simulation in ATLAS. Comput Softw Big Sci 8, 7 (2024). arxiv:2210.06204, doi:10.1007/s41781-023-00106-9

[17] R. Wang, "FATRAS integration for ATLAS fast simulation at HL-LHC." https://indico.cern.ch/event/1330797/contributions/5796509/, 2024

[18] F. Vaselli, "Flashsim: a ML based simulation for analysis datatiers." https://indico.jlab.org/event/459/contributions/11718/, 2023.

[19] L. Mijovic, "Fast simulation with generative models at the LHC." https://indico.cern.ch/event/1253590/contributions/5832658/, 2024.

[20] M. Barbetti, "Lamarr: LHCb ultra-fast simulation based on machine learning models deployed within Gauss", 2024, arxiv:2303.11428.

[21] M. Schmidt, "Optimizing the ATLAS Geant4 detector simulation." https://indico.cern.ch/event/1330797/contributions/5801194/, 2024.

[22] J. Esseiva, "Celeritas: evaluating performance of HEP detector simulation on GPUs." https://indico.cern.ch/event/1330797/contributions/5796653/, 2024.

[23] A. Gheata, "Accelerated demonstrator of electromagnetic Particle Transport (AdePT) status and plans." https://indico.jlab.org/event/459/contributions/11427/, 2023.

[24] "AdePT project." https://github.com/apt-sim/AdePT

[25] "Celeritas project." https://github.com/celeritas-project/celeritas

[26] Y. Li *et al.*, GPU-based optical photon simulation for the LHCb RICH 1 Detector, arxiv:2307.10823, 2023.

[27] A. Davis *et al.*, Optical Photon Simulation with Mitsuba3, arxiv:2309.12496, 2023.

[28] X. Ai *et al.*, A Common Tracking Software Project, arxiv:2106.13593, 2022.

[29] "ACTS Common Tracking Software documentation." https://acts.readthedocs.io/en/latest/

[30] A. Krasznahorkay, "Traccc: Track Reconstruction on GPU in ACTS." https://indico.cern.ch/event/1369601/contributions/5898656/, 2024.

[31] H. Torres, "Physics Performance of the ATLAS GNN4ITk Track Reconstruction Chain." https://indico.cern.ch/event/1252748/contributions/5576737/, 2023.

[32] J. Blomer, "Evolution of the ROOT Tree I/O." https://indico.cern.ch/event/773049/contributions/3474746/, 2019.

[33] J. Schaarschmidt, "PHYSLITE - a new reduced common data format for ATLAS." https://indico.jlab.org/event/459/contributions/11586/, 2023.

[34] M. Peruzzi *et al.*, "The NanoAOD event data format in CMS," *Journal of Physics: Conference Series*, vol. 1525, p. 012038, apr 2020. doi:10.1088/1742-6596/1525/1/012038

[35] J. Blomer, "ROOT's RNTuple I/O Subsystem: The Path to Production." https://indico.jlab.org/event/459/contributions/11594/, 2023.

[36] L. Gray, "Fine-Grained HEP Analysis Task Graph Optimization with Coffea and Dask." https://indico.jlab.org/event/459/contributions/11533/, 2023.

PoS(LHCP2024)193

[37] R. Gardner, "Computing at the HL-LHC and beyond." https://indico.cern.ch/event/1253590/contributions/5843742/, 2024.

[38] D. Ciangottini *et al.*, Analysis Facilities White Paper, arxiv:2404.02100, 2024.

[39] CERN, "CERN Open Science Policy", CERN-OPEN-2022-013, Geneva, 2022. https://cds.cern.ch/record/2835057.

[40] CERN, "CERN Open Data Policy for the LHC Experiments", CERN-OPEN-2020-013, Geneva, 2020. https://cds.cern.ch/record/2745133.

[41] "Reana website." https://reanahub.io/

[42] D. Fitzgerald, "An Ntuple production service for accessing LHCb Open Data: the Ntuple Wizard." https://indico.jlab.org/event/459/contributions/11696/, 2023.

[43] X. Ju, "Leveraging Language Models for Particle Reconstruction." https://indico.cern.ch/event/1330797/contributions/5796831/, 2024.