ALICE HLT Run 2 performance overview.

Mikolaj Krzewicki and Volker Lindenstruth for the ALICE collaboration
FIAS, Johann Wolfgang Goethe-Universität Frankfurt, Germany
E-mail: mikolaj.krzewicki@cern.ch

Abstract. For the LHC Run 2 the ALICE HLT architecture was consolidated to comply with the upgraded ALICE detector readout technology. The software framework was optimized and extended to cope with the increased data load. Online calibration of the TPC using online tracking capabilities of the ALICE HLT was deployed. Offline calibration code was adapted to run both online and offline and the HLT framework was extended to support that. The performance of this schema is important for Run 3 related developments. An additional data transport approach was developed using the ZeroMQ library, forming at the same time a test bed for the new data flow model of the O² system, where further development of this concept is ongoing. This messaging technology was used to implement the calibration feedback loop augmenting the existing, graph oriented HLT transport framework. Utilising the online reconstruction of many detectors, a new asynchronous monitoring scheme was developed to allow real-time monitoring of the physics performance of the ALICE detector, on top of the new messaging scheme for both internal and external communication. Spare computing resources comprising the production and development clusters are run as a tier-2 GRID site using an OpenStack-based setup. The development cluster is running continuously, the production cluster contributes resources opportunistically during periods of LHC inactivity.

1. Introduction

The ALICE High Level Trigger (HLT) is an online reconstruction and data compression system used in the ALICE experiment at CERN, comprising 180 worker nodes each containing 24 CPU cores. Unique among the LHC experiments, it extensively uses modern accelerator technologies like general purpose graphic processing units (GPGPU) and field programmable gate arrays (FPGA) in the data flow.

Real-time data compression is performed using a cluster finder algorithm implemented on FPGA boards and subsequent format optimisation and Huffman encoding stages. These data, instead of raw clusters, are stored for offline processing. The compression scheme is being improved to provide higher compression ratios for Run 3. The processing power of a single FPGA board used for cluster finding in the HLT is roughly equivalent to 15-30 CPU cores performing the same task using a sufficiently large memory buffer. Track finding is performed using a cellular automaton and a Kalman filter algorithm on GPGPU hardware, where CUDA, OpenCL and OpenMP (for CPU support) technologies can be used interchangeably. The gain of using GPGPUs for tracking was roughly $0.5M in Run 1 and with the higher processing requirements in Run 2 amounts to about $1M.
2. Consolidation in the new readout system
In the context of the upgrade of the readout system the HLT processing framework was optimised to handle the increased data and event rates due to the increased LHC luminosity in Run 2. The ALICE time projection chamber (TPC) readout was upgraded from the readout control unit 1 (RCU1) to RCU2 technology [1] using detector data links version 2 (DDL2) links [2] approximately doubling the maximum data throughput compared to Run 1. The data input and output in the HLT is handled by the readout receiver card (C-RORC) [3], see figure 1. This PCIe device was jointly developed by the ALICE HLT and DAQ groups and is also used by the ATLAS TDAQ ReadOut System and ATLAS Trigger. It provides state of the art interfaces to the host machine and made it possible for detectors to increase their readout bandwidth for Run 2.

The HLT C-RORC firmware was upgraded to handle both RCU1 and RCU2 formats and comply with the new data layout [4]. A single firmware version covers both the older RCU1 and the new RCU2 protocols. This eases the HLT testing and development cycle as older RCU1 data can be replayed from the onboard memory modules without a firmware change. Table 1 gives an overview of covered operating scenarios.

3. Data compression
The HLT performs online TPC data compression using a combination of techniques. The first stage of the procedure is cluster finding using the C-RORC FPGA which is followed by a data format optimization and a subsequent Huffman compression [5]. A new, differential Huffman compression scheme has been developed to replace the non-differential coding scheme of Run 1, bringing the compression ratio of TPC clusters from 4.3 to 5.5. The overall raw data compression improvement with respect to Run 1 is about 20%, thus significantly reducing the
Table 2. HLT processing chain performance for various input and processing configurations. In high luminosity pp and Pb–Pb scenarios the processing bottleneck is the TPC input link bandwidth.

|            | running reconstruction (all events) | max rate    | bottleneck          |
|------------|-------------------------------------|-------------|---------------------|
| pp (22 bunches) | TPC, ITS, EMCAL, VZERO, ZDC       | 4.5 kHz     | CPU                 |
| pp (1495 bunches) | TPC, ITS, EMCAL, VZERO, ZDC      | 2.4 kHz     | RCU2 bandwidth      |
| Pb–Pb      | TPC, ITS, EMCAL, VZERO, ZDC       | 0.95 kHz    | RCU2 bandwidth      |
| Pb–Pb      | ITS, EMCAL, VZERO, ZDC           | 6.0 kHz     | Event merger        |
| Pb–Pb      | TPC only, no framework overhead   | 2.5 kHz     | CPU/GPU             |

Figure 2. New data compression ratio as function of the number of TPC clusters (left) and the annual raw data size projections for the ALICE experiment for the previous and the current compression scheme.

ALICE data storage needs (see figure 2).

In view of the upgrade program for Run 3 and its high demand for efficient data storage studies are ongoing in the areas of track model compression, smart cluster charge encoding and junk removal, see e.g. [5].

4. HLT framework optimisations

In order to cope with the increased data load in Run 2, the HLT software frameworks underwent significant optimisations. The following points were optimized:

- The TPC tracking (running on GPU), see [6].
- Data flow topology was simplified.
- Data transport framework interprocess communication.

After the optimisation, the HLT system is able to reconstruct and process the TPC data at maximum data rate allowed by the input and output links. The data transport framework limits the processing rate in fast detector clusters to about 6 kHz, twice the highest rate possible before the optimisations. Currently the HLT is able to process all foreseen data taking scenarios.

In addition to the data flow and processing related framework optimisations, the configuration and initialisation times were reduced by an order of magnitude, see [7].

5. New HLT developments

The ALICE HLT has seen a number of new developments extending the functionality described so far. The new developments include real-time, closed-loop online calibration, real-time physics
and high level detector monitoring as well as a new data transport mechanism based on the ZeroMQ messaging queue library. Much of the implemented new functionality serves as a testing bed for concepts related to the ALICE O² system [8, 9] - a joint online-offline processing system being developed for LHC Run 3.

**Figure 3.** Overview of the HLT functionality. Each block represents a functional component in the processing chain. Data flow is from left (detector links) to right (output links and external data consumers). The lower blocks represent the calibration feedback loop.

### 5.1. Message queue based transport

The native HLT data transport framework only supports unidirectional synchronous data flow. In order to accommodate the need for asynchronous processing of user analysis tasks and the calibration feedback loop, an additional transport layer based on the ZeroMQ library [10] was added. ZeroMQ support for scatter/gather IO (multi-part messages) was utilized to maintain the association between the data and its metadata: a single message carries multiple payloads, each preceded by its header (see figure 4). The scatter/gather technique eliminates the need for the explicit construction of contiguous IO buffers, saving both CPU and memory. Separation of data from metadata at this level also facilitates efficient dispatch and parsing as payloads are not touched (unpacked or possibly decrypted) to parse the meta information. The new messaging model used here forms a prototype and a test bed for the data flow model of the ALICE O² system. Figure 3 shows how the new messaging framework is being utilized in the HLT.

### 5.2. Online TPC calibration

TPC calibration (and most other high level ALICE calibration procedures) runs as a user analysis task inside the physics analysis framework using reconstructed data as input. Offline calibration code was adapted to run both offline in the physics analysis framework (fully backward compatible) as well as online using the new HLT analysis manager framework. The HLT infrastructure allows any adapted user analysis code to run online irrespectively of its...
Figure 4. The message layout. Each payload in a scatter-gather list (a multi-part message) is preceded by metadata. The messaging library (ZeroMQ) guarantees in-order delivery of all parts of a multi-part message.

The tasks run asynchronously in separate processes to avoid stalling the main data processing chain. Process separation also provides failure resilience with respect to user code misbehaviour. The procedure allows the calibration code to gather statistics for a predefined period of time (typically in the order of 10 minutes for the TPC). Subsequently and periodically merges the produced outputs. After a processing step the calculated calibration parameters are shipped to the beginning of the chain via the feedback loop and are applied to the incoming TPC clusters (see figure 3 and [11]). The comparison between the correction factors calculated online and offline in figure 5 shows a similar trend as function of time. The difference is caused by a different set of boundary conditions: not all sensor data was accessed online in this exercise. Comparing the cluster positions with clusters calibrated using the standard offline procedure yields a difference of at most 0.5 mm, well within the intrinsic TPC cluster resolution, despite the difference in the correction factor which compensates for the missing sensor data, see figure 5. Online calibration, next to being an important exercise for Run 3, can reduce the computing workload during the offline calibration and reconstruction cycle already in Run 2.

Figure 5. Time dependence of the drift velocity correction calculated online and offline (left). Differences in $z$ positions of clusters corrected by the online and offline procedures (right).

5.3. Online physics QA and monitoring
The HLT provides reconstructed data for many detectors in real-time making it possible to monitor high level observables (including physics observables) online. A scheme was implemented to allow real-time inspection of the reconstructed data using all processed statistics fully synchronously with data taking or, optionally, asynchronously using limited sampled data subsets. The fully synchronous monitoring requires processing components that run inside the main data processing chain. This puts strict requirements on the performance of the monitoring
code. This scheme is used to monitor relatively simple observables. The procedure has access to most of the raw and reconstructed data and can be reconfigured in real-time, accommodating the need for fast feedback during contingencies. The new HLT analysis framework allows for straightforward integration of offline analysis tasks into the online data processing chain. It is used to run data quality monitoring and analysis code out of the critical (synchronous) path, limiting the processed statistics to what any particular procedure is able to process, with the benefits of failure resilience.

The data produced by the calibration and monitoring components running in parallel on all nodes is periodically pushed to merger processes, where the incoming data (typically ROOT objects) is merged and made available to outside consumers. In addition, the mergers add run and LHC condition metadata and on request can add the ROOT schema evolution information (streamers) to aid external clients with data unpacking and interpretation. The mergers use the ZeroMQ based scheme described in section 5.1 for both input and output channels.

5.4. Offline use

The two HLT computing farms (the development and production clusters) are used opportunistically to run offline tasks in virtualised environments using OpenStack (see [12]). Due to the networking layout which is more restricted than for a regular data centre, only tasks are run which do not require large amounts of data transfers, most frequently long running Monte Carlo simulations. The smaller development farm is used for offline tasks during most of the year barring occasional use for software validation and large scale tests. The main production resource was used for offline tasks only during longer breaks in data taking, mostly during technical LHC shutdowns. Since the start of operation of the OpenStack setup in February 2016, approximately 650 thousand jobs were completed, circa 2.5% of the entire ALICE simulation workload, see figure 6.

![Figure 6. Total wall time in hours for ALICE jobs on the HLT vs. date in 2016. Lower, continuous line represents jobs running on the development cluster, the upper lines represent jobs running on the main production cluster during technical LHC shutdowns.](image)

6. Summary

After the successful readout consolidation and deployment in LHC Run 2 online data processing, the HLT has seen many new developments pertaining to the upgrades of the ALICE experiment for Run 3. New concepts like real-time calibration and physics monitoring were tested and are being developed further. Some of the new developments served a role of a prototype for the upcoming O² system, like the new data transport model. Efforts will continue to develop the
new ideas on the current infrastructure and, at the same time, to implement (or port) them in the new O² framework.

References
[1] Alme J et al. 2013 RCU2 - The ALICE TPC readout electronics consolidation for Run 2 JINST 8:C12032
[2] Carena F et al. 2015 DDL, the ALICE data transmission protocol and its evolution from 2 to 6 Gb/s JINST 10(04):C04008.
[3] Borga A et al. 2015 The C-RORC PCIe card and its application in the ALICE and ATLAS experiments JINST 10(02):C02022
[4] Engel H et al. 2017 FPGA based data processing in the ALICE High Level Trigger in LHC Run 2 JPCS
[5] Richter M et al. 2017 Online Data Compression in the ALICE O2 facility JPCS
[6] Rohr D et al. 2017 GPU-accelerated track reconstruction in ALICE High Level Trigger JPCS
[7] Rohr D et al. 2017 Improvements of the ALICE HLT data transport framework for LHC Run 2 JPCS
[8] https://github.com/AliceO2Group
[9] Buncic P and The ALICE Collaboration 2015 Technical Design Report for the Upgrade of the Online-Offline Computing System (ALICE-TDR-019) CERN-LHCC-2015-006
[10] http://zeromq.org
[11] Krzewicki M and Lindenstruth V 2017 Support for online calibration in the ALICE HLT framework JPCS
[12] Lehrbach J et al. 2017 ALICE HLT Cluster operation during ALICE Run 2 JPCS