Proceedings of the LHCP2015 Conference, St. Petersburg, Russia, August 31 - September 5, 2015

Editors: V.T. Kim and D.E. Sosnov

The ATLAS Data Flow System for LHC Run 2

ANDREI KAZAROV[1,2]

[1]*CERN, CH1211 Geneva 23, Switzerland*
[2]*on leave from: Petersburg NPI Kurchatov NRC, Gatchina, Russian Federation*

Andrei.Kazarov@cern.ch

On behalf of the ATLAS Collaboration

Abstract. After its first shutdown, the LHC will provide pp collisions with increased luminosity and energy. In the ATLAS experiment, the Trigger and Data Acquisition system has been upgraded to deal with the increased event rates. The Data Flow element of the TDAQ is a distributed hardware and software system responsible for buffering and transporting event data from the readout system to the High Level Trigger and to the event storage. The DF has been reshaped in order to profit from the technological progress and to maximize the flexibility and efficiency of the data selection process. The updated DF is radically different from the previous implementation both in terms of architecture and expected performance. The pre-existing two level software filtering, known as Level 2 and the Event Filter, and the Event Building are now merged into a single process, performing incremental data collection and analysis. This design has many advantages, among which are: the radical simplification of the architecture, the flexible and automatically balanced distribution of the computing resources, the sharing of code and services on nodes. In addition, logical HLT computing cluster slicing, with each slice managed by a dedicated supervisor, has been dropped in favour of global management by a single master operating at 100 kHz. The Data Collection network that connects the HLT processing nodes to the Readout and the storage systems has evolved to provide network connectivity as required by the new Data Flow architecture. The old Data Collection and Back-End networks have been merged into a single Ethernet network and the Readout PCs have been directly connected to the network cores. The aggregate throughput and port density have been increased by an order of magnitude and the introduction of Multi Chassis Trunking significantly enhanced fault tolerance and redundancy. We will discuss the design choices, the strategies employed to minimize the data-collection latency, architecture and implementation aspects of DF components.

INTRODUCTION

The ATLAS experiment [1] is one of the major experiments of the Large Hadron Collider (LHC). It is a general purpose experiment, aiming at studying the Standard Model Higgs Boson, and looking for physics beyond the Standard Model of particle physics. It consists of various charged particle tracking detectors (the Inner Detector), a liquid-argon based electromagnetic and a hadronic calorimeter and a muon spectrometer. The detector provides many millions of read-out channels, able to capture data every 25 ns. This volume of data can not be recorded and kept for further data analysis. The ATLAS Trigger & Data Acquisition (TDAQ) system is responsible for the readout, selection and delivering to the permanent storage of the physics events and also for central infrastructure services like Controls, Configuration and Monitoring. TDAQ plays a fundamental role in the ATLAS experiment operation.

The TDAQ system performed very well during LHC Run 1 (2009-2012) [2], in many aspects well beyond its design values. It allowed ATLAS to collect events relevant for physics analyses with a high efficiency and desired event recording rates.

This paper focuses on the evolution of the Data Flow (DF) system. DF is the TDAQ component responsible for reading out and buffering ATLAS detector event fragments at Level 1 trigger rate, providing necessary fragments and events to the High-Level Trigger (HLT) system, building the selected events and finally sending them to the permanent storage as shown in Fig. 1.

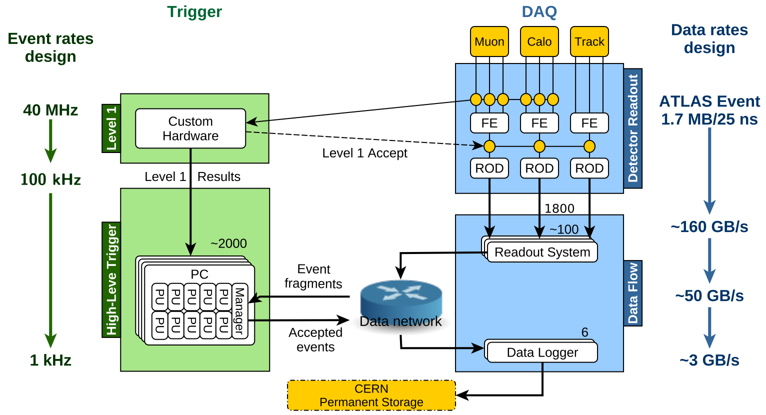

FIGURE 1: TDAQ architecture for Run 2.

New requirements for Data Flow system for Run 2

LHC Run 2 started in June 2015, bringing higher collision energy, luminosity and Level-1 trigger rate limitations as summarized in Table 1.

TABLE 1: Characteristic properties of LHC Run 1 and expected values for Run 2.

	Bunch spacing [ns]	Inst. luminocity $[cm^{-2}s^{-1}]$	L1 accept rate [kHz]	Readout fraction [%]	Event building bandwidth [GBps]	Peak recording rate [kHz/GBps]	N of readout channels	Event size (max) [MB]
Run 1	50	$8x10^{33}$	70	15	10	1/1.6	1600	1.6
Run 2	25	$1.7x10^{34}$	100	50	50	2/3	1800	2.0

In order to meet the updated requirements, the DF system was redesigned and reimplemented in the course of LHC long shutdown 1 (LS1). This process resulted in a simpler and streamlined design taking advantage of new hardware and software technologies.

New Data Flow architecture

The updated DF architecture (Fig. 2) is radically different from the previous implementation. The Region of Interest (RoI) concept has been kept, and processing and data collection proceeds in stages, beginning with fast algorithms based on RoIs. The decision when to build the full event is flexible, and afterwards more off-line style algorithms have access to the full event. Two levels of software filtering, known as Level 2 and the Event Filter are now merged into a single level, performing incremental data collection and analysis on the nodes of the flat HLT computing cluster. This design has many advantages, in particular:

- the radical simplification of the architecture, reducing number of components, dependencies and communication patterns;
- flexible and implicitly balanced utilization of the computing resources over the cluster, based on the active HLT algorithms;
- sharing of HLT code and services on the cluster nodes among all processing units, reducing the memory and resources utilization and allowing to run more HLT processing unit instances per node.

In addition, logical cluster slicing, with each slice managed by a dedicated supervisor, has been dropped in favour of global management by a single master (HLTSV) operating at 100 kHz. This simplifies management of the cluster utilization at the cost of higher reliability requirements for the central node. These were addressed by adding an idle backup element to the system.

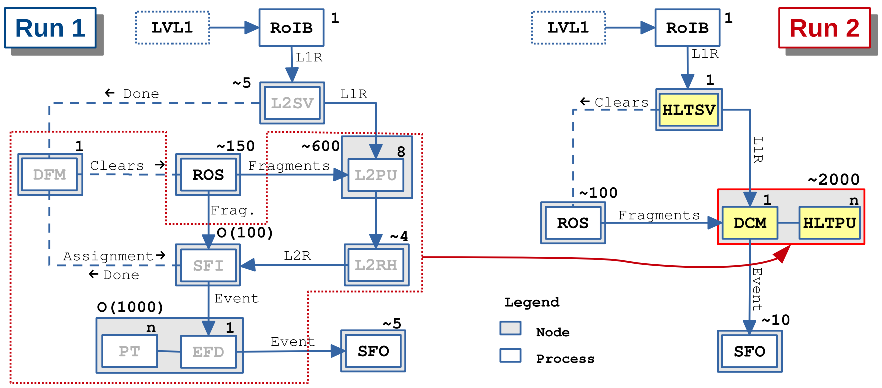

FIGURE 2: Evolution of DF architecture from Run 1 (provided as reference. A detailed description can be found in Ref. [2]) to Run 2.

As shown in Fig. 2, the system is composed of the following elements:

- the Readout System (ROS) buffers front-end data from the detectors and provides a standard interface to the DF;
- the Region of Interest Builder (RoIB) receives L1 trigger information and RoIs and combines the information for the HLT Supervisor;
- the HLT supervisor (HLTSV) schedules events to the HLT cluster, clears ROS buffers and handles possible time-outs;
- the Data Collection Manager (DCM) handles all I/O on the HLT nodes, including RoI requests from the HLT and full event building;
- the HLTPU processing tasks are forked from a single mother process to maximize memory sharing and run the ATLAS Athena/Gaudi framework performing event selection;
- the Data Loggers (SFO) are responsible for saving accepted events to disk, and for sending the files to permanent data storage.

All elements in the system are communicating via high-performance data and control Ethernet networks.

Data and Control networks

The Data network experienced a significant technological upgrade and simplification. With Multi-Chassis Trunking, a Brocade proprietary technology, the core routers provide load balancing and link redundancy to the network (Fig. 3). This is achieved by creating aggregated links on the devices connected to the routers. The latter are perceived as a single virtual network device. A proprietary protocol running between the two routers ensures the Routing and Forwarding table synchronization between the two network cores.

The internal architecture of the routers incorporates a major improvement in the packet forwarding capacity by introducing the Virtual Output Queuing (VOQ) technique which, already at the input stage, assigns different packet queues to each output port. Thanks to the VOQ technique, the packets addressed to a given output port can be forwarded independently of the traffic going to the other output ports, removing the head-of-line blocking phenomenon.

The Control network is a parallel network to the Data network in which the traffic for the control, configuration and monitoring of the TDAQ infrastructure flows. There are not strong performance requirements for the control network so the main upgrade activities concerned the redundancy and fail-over techniques. As part of the redundancy improvements, an Active-Backup setup has been installed for all important nodes in the system.

Figure 4 shows the achieved Data Collection bandwidth (which is a data flow from the ROS to the HLT cluster) during a test run (comparing to Run 1 capacity) and bandwidth utilization during a p-p run in 2015.

The following sections give more details on the DF components.

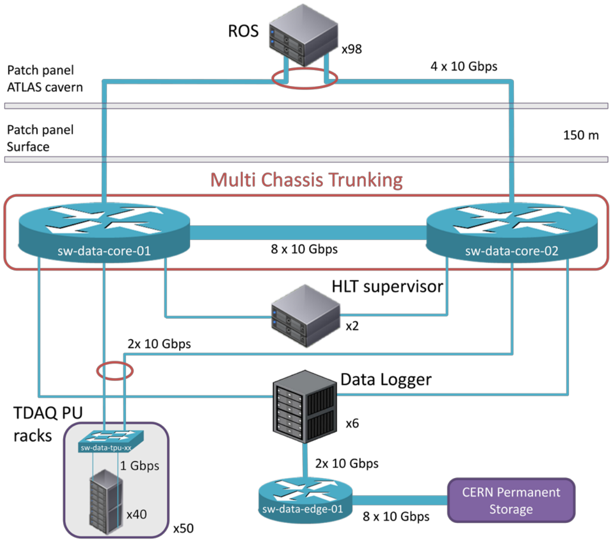

FIGURE 3: TDAQ Data network topology.

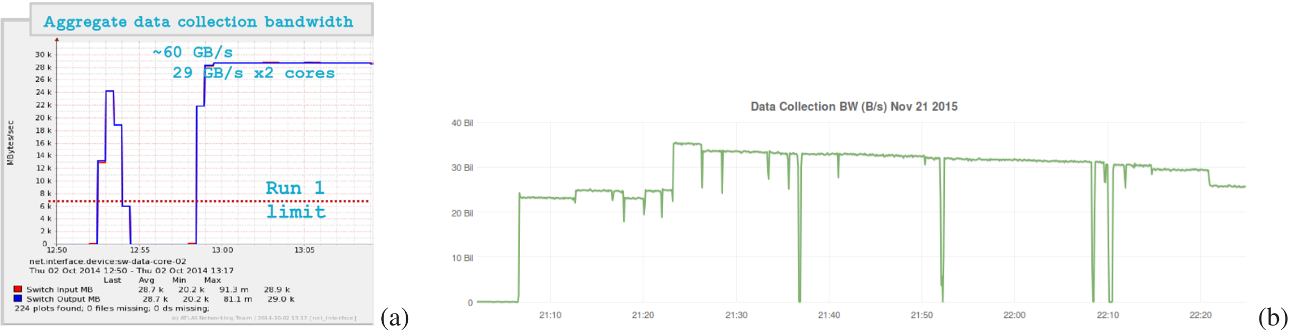

FIGURE 4: Data Collection bandwidth utilization in a test run (a) and in one of p-p runs in Nov 2015 (b).

Region of Interest Builder

The RoIB is a 9U VME based custom hardware solution, consisting of multiple cards. The original hardware from Run 1 is the current baseline. Larger input fragments for the Run 2 upgrade show that the hardware is close to its limits. A replacement based on a custom board is being developed, using common hardware between ROS and RoIB [3].

A single PCI Express board integrated directly into the HLT supervisor will suffice for all 11 inputs, and combining of the input fragments will be done in software. First test results in the lab are very promising, far exceeding the requirements.

Read-Out System

Addition of new detectors to ATLAS, higher luminosity and L1 trigger rates requires a more dense and a more performant readout system. Therefore the ROS system went through a full overhaul. Fully new ROS PCs feature: 2U form factor instead of 4U; 4x10GbE per ROS PC (was 2x GbE); higher density of optical links per server: 12 per input card, 2 cards per PC [4].

ROS PC hosts a new powerful input card (S-Link input and buffer hardware) which is based on ALICE C-RORC card [5] with ATLAS firmware and software. It includes: 8x PCI Express lanes Xilinx Virtex-6 @ 125MHz FPGA; DDR3-1600 SO-DIMM RAM 2x4GB Buffer memory; 12 input optical channels.

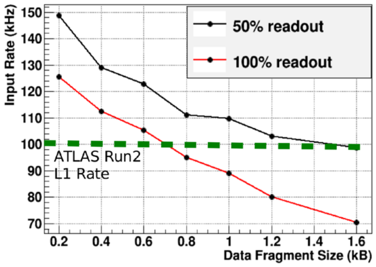

FIGURE 5: Sustained input rate of ROS as a function of readout fraction and fragment size. Taken from Ref. [4].

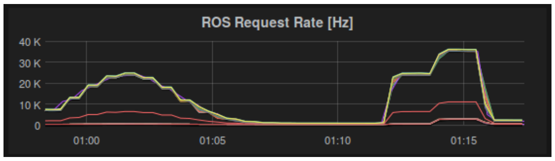

FIGURE 6: A screenshot of the online monitoring tool showing request rate to all ATLAS ROSes during a special run.

The connectivity of the ROS PCs to the HLT computing cluster is provided by four 10 GbE ports on two dual-port network interface cards.

A fully connected (24 links) ROS can sustain up to 50% readout for a wide set of request patterns (e.g. 35kHz of L1 during a scan run in 2015 as shown in Fig. 6). A ROS with fewer input links and/or small enough fragments can provide fragments at 100 kHz.

Software components

The following subsections describe the features of the main DF software components, running on more then 2400 commodity PCs in the HLT computing cluster.

Data Collection Manager

The DCM is a single application per HLT node. It deals with all data requests from multiple HLT processing tasks on the same node. It handles all requests to the ROS and communicates to the tasks via sockets and shared memory. Its design is essentially single-threaded based on non-blocking I/O using the Boost ASIO [6] library. A credit based traffic shaping mechanism [7] is used to prevent overloading the incoming network link on switches. For the selected events, DCM compresses the event payload before sending it to the data logger.

HLT Supervisor

A single HLT supervisor replaces the set of L2 supervisors used in Run 1. It uses a heavily multi-threaded, asynchronous design using the Boost ASIO library for communication and Intel Thread Building Blocks [8] for concurrent data structures. It reads RoIB fragments from 2x optical inputs links and sends assigned events to the HLT cluster via 2x10GbE links. A single HLTSV application can handle the input from the RoIB and manage the HLT cluster of 2000 machines at 115 kHz under realistic ATLAS conditions. In the future it is foreseen to merge the RoIB and HLTSV functionalities into a single PC equipped with C-RORC cards.

HLT Processing Unit

The HLT processing unit encapsulates the Athena framework [9] that executes the actual HLT algorithms. It communicates with the DCM for I/O requests and provides the trigger decision for each event. On each node a mother process is started first and goes through all the configuration process. A set of child processes is forked when a run starts. Thanks to the Linux kernel copy-on-write feature the memory sharing across the child process is maximized, hence the memory usage per node is minimized. Crashed HLT applications can be quickly replaced by forking another child instance. Tests with the full 2012 trigger menu show a memory consumption on a node of 1.8 GByte for the mother process plus 700 MB per child process. On a 12 core server this corresponds to 50% memory saving.

Transient Data Storage

In Run 1 the transient data storage [10] was implemented with storage server equipped with internal raid card. For Run 2 external SAS units with multiple front-ends and redundant data paths for fault tolerance and resilience are used (Fig. 7). Total capacity of the system is 340 TB, allowing ATLAS to record data being disconnected from offline storage

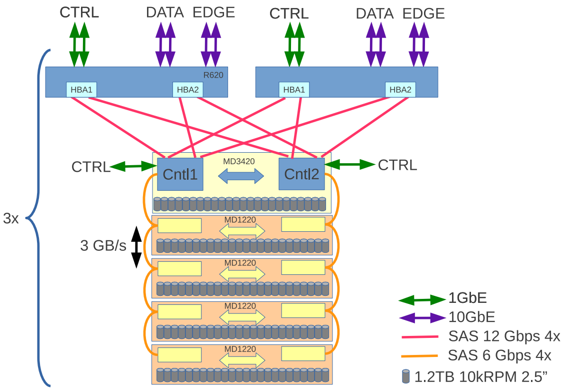

FIGURE 7: Internal architecture of a SFO unit.

for at least 24hrs. Performance depends on the trigger menu, number of streams, etc. In typical configurations, 4 GB/s of peak recording bandwidth is available.

Background jobs copy the files to permanent storage, deleting them on the local disk only when they are safely stored on tape.

Conclusions

The smooth operation of the TDAQ system has a direct impact on the efficiency of the ATLAS experiment and on the achievement of its Physics goals. In this paper we have shown how the Data Flow architecture has been reshaped during the LS1 in order to profit from the technological progress and to maximize the flexibility and efficiency of the data selection process.

REFERENCES

[1] The ATLAS Collaboration, JINST **3**, p. S08003 (2008).

[2] G. Avolio, S. Ballestrero, and W. Vandelli, Physics Procedia **37**, 1819 – 1826 (2012), proceedings of the 2nd International Conference on Technology and Instrumentation in Particle Physics (TIPP 2011).

[3] R. Blair, G. J. Crone, B. Green, J. Love, J. Proudfoot, O. Rifki, W. Panduro Vazquez, W. Vandelli, J. Zhang, and B. Abbott, “The Evolution of the Region of Interest Builder for the ATLAS Experiment at CERN,” Tech. Rep. ATL-DAQ-PROC-2015-052 (CERN, Geneva, 2015).

[4] A. Borga, “Evolution of the ReadOut System of the ATLAS experiment,” Tech. Rep. ATL-DAQ-PROC-2014-012 (CERN, Geneva, 2014).

[5] A. Borga, F. Costa, G. Crone, H. Engel, D. Eschweiler, D. Francis, B. Green, M. Joos, U. Kebschull, T. Kiss, A. Kugel, J. P. Vazquez, C. Soos, P. Teixeira-Dias, L. Tremblet, P. V. Vyvre, W. Vandelli, J. Vermeulen, P. Werner, and F. Wickens, Journal of Instrumentation **10**, p. C02022 (2015).

[6] Boost asio library, https://think-async.com, .

[7] T. Colombo and ATLAS Collaboration, Journal of Physics: Conference Series **608**, p. 012005 (2015).

[8] Intel Corporation, Intel threading building blocks library, https://www.threadingbuildingblocks.org, .

[9] The ATLAS Collaboration, *ATLAS Computing: technical design report*, Technical Design Report ATLAS (CERN, Geneva, 2005).

[10] A. Battaglia, H. Beck, M. Dobson, S. Gadomski, K. Kordas, and W. Vandelli, “Performance of the final data-logging system of the trigger and data acquisition for the ATLAS experiment at CERN,” in *Nuclear Science Symposium Conference Record, 2008. NSS ’08. IEEE* (2008), pp. 2173–2179.