Designing a data acquisition system for performance in the context of the CMS experiment

Fábio Carneiro
fabio.carneiro@cern.ch
BTU, Cottbus, Germany

Leen Lambers
leen.lambers@b-tu.de
BTU, Cottbus, Germany

Zhen Xie
zhen.xie@cern.ch
Princeton University, USA

The Large Hadron Collider (LHC), the largest particle accelerator in the world, provides particle collisions to four particle physics experiments [1]. These experiments not only record collision data but also report key parameters back to the operators of the accelerator. One of these parameters is the instantaneous delivered luminosity, which describes the rate of particle collisions at an experiment's interaction point. For the Compact Muon Solenoid (CMS) experiment, the Beams Radiation Instrumentation and Luminosity (BRIL) group operates a data acquisition system, BRILDAQ, responsible for acquiring, processing, and storing luminosity data from a pre-selected set of CMS subdetectors and for providing luminosity feedback to the accelerator operators within a soft limit of 1 to 2 seconds [2].

The acquisition from a diverse set of subdetectors is challenging because BRILDAQ must interface with the different hardware, software, and transport protocols used by each subdetector [3]. All subdetectors' hardware aggregates luminosity histograms for 1 second and releases them in a synchronized way, so the data arrives in bursts. In total, up to 82 back-end boards across 7 subdetectors produce up to 26 MB per burst, and each board sends between 30 KB and 8 MB. In order to minimize single points of failure, the data for each subdector is acquired and processed independently from another. The high-throughput CMS network is reserved for other collision data, so luminosity data travels over a hierarchical general-purpose 1 Gb/s network shared with configuration and monitoring traffic [4]. All acquired data must also be preserved for offline analysis. Lastly, luminosity processing depends on secondary sources, including the state of the accelerator. The design must therefore let applications receive data from several sources without much coupling to them.

This last requirement is usually met by adopting a publisher-subscriber pattern where each application subscribes only to the information it needs. MQTT, a publisher-subscriber protocol widely adopted by CMS, is an attractive option (Figure 1a). A benchmark of a bridged MQTT broker [5] reported mean latencies of about 230 to 260 ms, for QoS 1, for payloads of 125 KB. The largest BRILDAQ messages reach 8 MB, more than 60 times that size. A broker-only design must also either send the full data twice, to storage and to processing, or let the processing application store it, coupling persistence to processing failures.

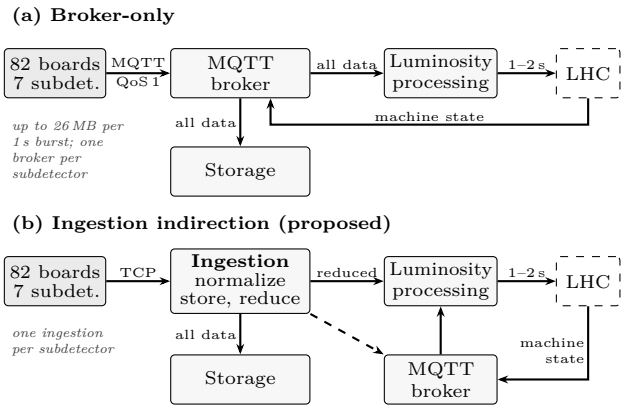

Figure 1: Alternative designs for the BRILDAQ acquisition. Solid arrows show the data flow. The dashed arrow shows a possible variant of design (b) in which the reduced data is forwarded through the broker instead.

An alternative design is to separate the latency-critical data path from the MQTT integration path (Figure 1b). The boards of each subdetector send their data over TCP to an ingestion component, which converts the subdetector payloads into one internal format, stores the complete stream, reduces the data needed online, and forwards the reduced data either directly to the processing application or through the broker. Because the ingestion component stores the complete stream before forwarding anything, the full data crosses the network only once, and a processing failure can no longer stop persistence. TCP is used only for this critical link, and other consumers remain on MQTT. The drawback of bypassing the broker is that connection handling and message buffering are no longer provided.

Data reduction lowers the load on processing stages at the cost of granularity, but only for the online data, since the stored copy is complete. Averaging data across sources of the same subdetector is the simplest approach. Additional lossless compression could be added on top, but its benefit depends on the entropy of the data, which still has to be studied.

The current analysis is qualitative, and the next step is to verify the design against the performance requirements. Palladio [6] is one tool being considered for modeling the different design approaches and evaluating them against the quality requirements of the system.

References

[1] L. Evans and P. Bryant. "LHC Machine". In: *Journal of Instrumentation* 3.08 (Aug. 2008), S08001–S08001.

[2] T. C. Collaboration et al. "The CMS experiment at the CERN LHC". In: *Journal of Instrumentation* 3.08 (Aug. 2008), S08004–S08004.

[3] C. Collaboration. *The Phase-2 Upgrade of the CMS Beam Radiation Instrumentation and Luminosity Detectors*. Tech. rep. Geneva: CERN, 2021.

[4] C. Collaboration. *The Phase-2 Upgrade of the CMS Data Acquisition and High Level Trigger*. Tech. rep. Geneva: CERN, 2021.

[5] K. Lima et al. "Evaluation of MQTT Bridge Architectures in a Cross-Organizational Context". In: *2025 IEEE 22nd International Conference on Software Architecture (ICSA)*. IEEE, 2025, pp. 243–254.

[6] S. Becker, H. Koziolek, and R. Reussner. "The Palladio component model for model-driven performance prediction". In: *Journal of Systems and Software*. Special Issue: Software Performance - Modeling and Analysis 82.1 (Jan. 2009), pp. 3–22.