EPJ Web of Conferences **245**, 01027 (2020) https://doi.org/10.1051/epjconf/202024501027
*CHEP 2019*

# The ALICE $O^2$ data quality control system

*Piotr* Konopka[1,2,*] and *Barthélémy* von Haller[1,**] for the ALICE Collaboration

[1]CERN, Experimental Physics Department, Geneva, Switzerland
[2]Faculty of Electrical Engineering, Automatics, Computer Science and Electronics, AGH University of Science and Technology, Cracow, Poland

**Abstract.** The ALICE Experiment at CERN LHC (Large Hadron Collider) is undertaking a major upgrade during LHC Long Shutdown 2 in 2019–2021. The raw data input from the ALICE detectors will then increase a hundredfold, up to 3.5 TB/s. In order to cope with such a large amount of data, a new online-offline computing system, called $O^2$, will be deployed.
One of the key software components of the $O^2$ system will be the data Quality Control (QC) that replaces the existing online Data Quality Monitoring and offline Quality Assurance. It involves the gathering, the analysis by user-defined algorithms and the visualization of monitored data, in both the synchronous and asynchronous parts of the $O^2$ system.
This paper presents the architecture and design, as well as the latest and upcoming features, of the ALICE $O^2$ QC. The results of the extensive benchmarks which have been carried out for each component of the system are later summarized. Finally, the adoption of this tool amongst the ALICE Collaboration and the measures taken to develop, in synergy with their respective teams, efficient monitoring modules for the detectors, are discussed.

## 1 Introduction

### 1.1 ALICE

ALICE [1] is the detector designed to cope with the high particle multiplicities produced in heavy-ion collisions at the CERN LHC to study the physics of strongly interacting matter and the quark–gluon plasma [2]. During the Long Shutdown 2 in the years 2019–2021 (LS2) the ALICE experiment is undergoing a major upgrade. The tracking detectors are entirely replaced in order to achieve a higher resolution and new detectors are added while some are decommissioned.

### 1.2 $O^2$

After the upgrade, the ALICE experiment will see its data input throughput increase a hundredfold, up to 3.5 TB/s. Moreover, the physics topics addressed by the ALICE upgrade [3] are characterized by a low signal to noise ratio making triggering techniques very inefficient

*e-mail: piotr.konopka@cern.ch
**e-mail: barthelemy.von.haller@cern.ch

if not impossible. Finally, the implementation of a continuous readout is required by the Time Projection Chamber in order to keep up with the 50 kHz interaction rate.

In order to cope with such requirements, a new Online and Offline Computing system, called $O^2$ [4], has been developed. It is characterized by the continuous readout of all interactions, their compression by means of partial online reconstruction and calibration and the sharing of common computing resources during and after data taking. In this scheme, only the reconstructed data is written to disk while the raw data is discarded.

As shown on Fig. 1, there are two major computing layers, the *First Level Processors* (FLPs) and the *Event Processing Nodes* (EPNs). Both are highly heterogeneous, with specialized acquisition cards embedding FPGAs on the FLPs and GPUs on the EPNs.

The $O^2$ software is based on a multi-process message-based system with a zero copy approach in the main processing flow. A Data Processing Layer [5] software framework is being developed on top of the FairMQ data transport layer [6].

## 2 Data Quality Control and Assessment

### 2.1 Definition

One of the key software components of the $O^2$ system is the "Data Quality Control and Assessment" (QC) that replaces the former online Data Quality Monitoring (DQM) and offline Quality Assurance (QA). The QC [7] is critical to identify and overcome problems during data taking, to provide good quality data for physics analyses and to ensure that the data processing behaves as expected, especially when running synchronously with the data taking.

The unification of the online and offline worlds, as well as the discarding of raw input data in favour of reconstructed data, make the need for a reliable data quality control even more essential. The challenge is made greater due to the more than 15 different detector teams involved, the very large amount of data to look after (3.5 TB/s) and the expected number of QC tasks (>100) that are going to run in parallel and produce more than ten thousand unmerged objects per second.

### 2.2 Architecture

The QC is split into a number of components shown on Fig. 1.

The *Data Sampling* (blue doted arrows) is in charge of selecting and distributing data samples based on configurable policies. Its key component, the *Dispatcher*, runs on every node where data should be sampled. Its key features include the pseudo-random sampling of parallel data distributed among many machines, the custom filtering as a plugin system and its reconfiguration during data taking.

The *QC tasks* (purple boxes) execute detector-specific algorithms either locally on the FLPs and the EPNs or remotely on dedicated Quality Control Servers. They publish their results as QC Objects, typically ROOT [8] histograms. As most tasks are running in parallel on many nodes, their output has to be merged. The *Mergers* are able to perform this task efficiently.

The *Checkers* then take care of evaluating the quality of the objects by running user algorithms developed under a common interface. A common set of checks is provided but users can develop their own. In parallel, the usage of Machine Learning is investigated as a mean to perform a similar evaluation of the QC Objects. Finally, the QC Objects and the Qualities are stored in the QC repository where they are available to the shifters and experts. Moreover, a system to aggregate these objects and use them to trigger alarms is planned. This database uses the same technology as the *Calibration and Condition DataBase* of ALICE $O^2$.

The *Post-processing* component encompasses any task running asynchronously to the main data flow. The main use is the correlation and trending of data derived from QC Objects and Qualities. It is triggered periodically, manually or on certain events (e.g. start of run or end of fill).

Once the QC Objects are stored, along with some Quality Objects, the shifters and experts use the QC GUI (QCG) to visualize them. A generic user interface is proposed which allows to navigate the objects and display them using JSROOT [9]. Users can create *layouts* to save the arrangement of tabs and objects, as well as their display options, and reuse or share them later.

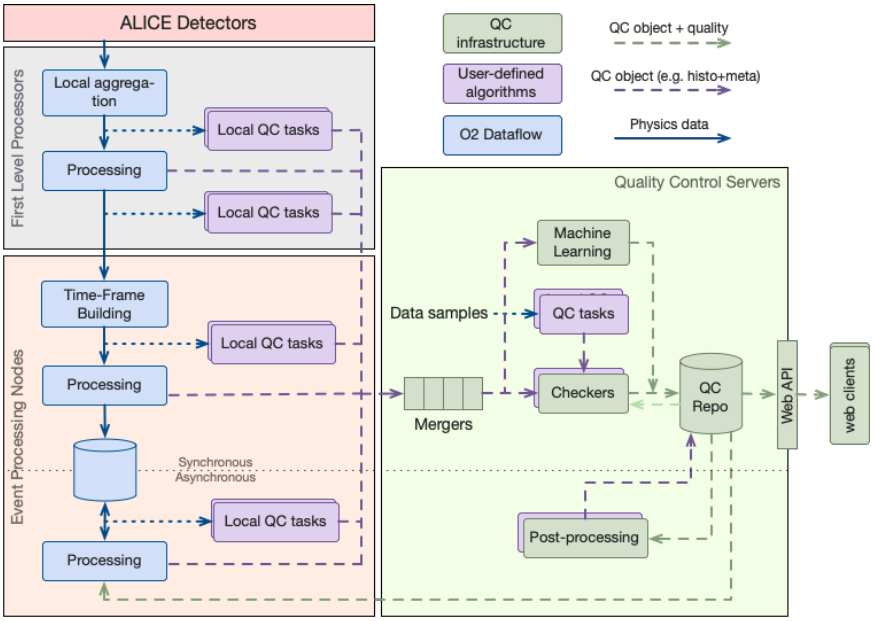


**Figure 1.** The QC architecture.

### 2.3 Data rates

Data volume is reduced with each processing step of the QC chain by means of sampling and replacement by objects of higher level abstraction, starting from the main processing data, down to high level information such as Qualities (see Fig. 2).

It is estimated that the full $O^2$ system will see around 3.5 TB/s of raw data as its input. After the first level processing a stream of 635 GB/s will reach the Event Processing Nodes, where it will be further reduced to 100 GB/s as we replace the raw data with reconstructed data. Some processing steps might generate additional temporary data as well. The granularity of data might differ significantly, starting from 2 MB pages up to timeframes of a dozen of GBs. That will have a direct influence on message rates in the system. Just one process might produce 7500 messages per second, if we assume 2 MB payloads. QC tasks will usually use between 1% and 10% of messages of different data types and only in a few cases they will require a full stream. An average QC Object might have the size of 250 kB. Out of partial QC Objects produced by QC Tasks, Mergers will have to generate complete versions of 10000

objects each minute. In total, the QC Tasks and Post-processing Tasks are estimated to produce about 25000 complete objects updated each minute. These will be evaluated, resulting in around 100000 Qualities.

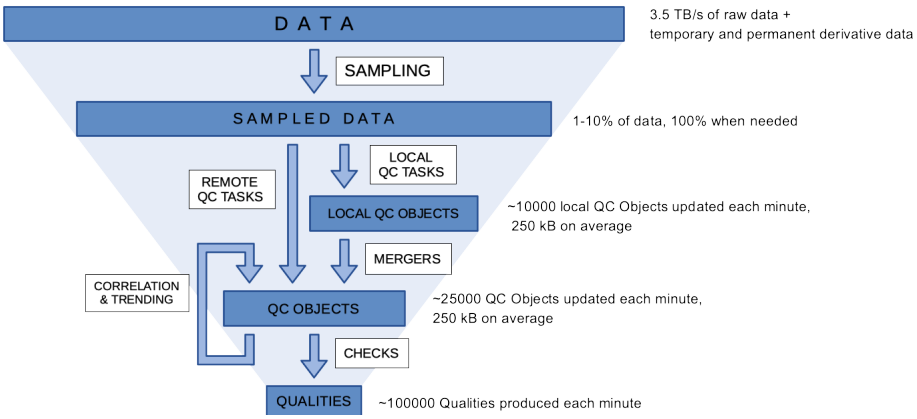


**Figure 2.** The QC chain and its data rates.

## 3 Performance study

The full QC chain involves multiple components and each one is a possible bottleneck if implemented without care. Also, as multiple processing steps will be sharing common CPU, memory and bandwidth resources, they might have an influence on others' performance. Generally, the first components in the processing chain appear to require more computing resources, since they handle higher data rates. In this chapter, results of benchmarks that were carried out to evaluate performance of each part of the system are presented.

All the tests results were obtained on a server with a Dual Intel(R) Xeon(R) CPU E5-2640 v3 @ 2.60 GHz and 128 GB RAM. Each benchmark, which involved measuring an amount of data that can be processed by a component, was carried out by running similar process topology. It consisted of multiple data producers and the benchmarked component which was receiving and processing messages. When justified, the latter sent data further to a data sink. The data producers would create as many messages as possible in order to saturate the compenent's input buffers. As one of these was not enough to do it, multiple data producers would be used in parallel. Each configuration was run 5 times to confirm reproducibility of the results.

Fig. 3 presents results of the Data Sampling benchmarks. It shows an amount of processed messages per second as a function of the message payload size. Two variants are shown - when the Dispatcher rejects each arrived message (0% sampled) and when it passes all of them (100% sampled). The difference of performance indicates an overhead when copying data and marks the expected low and high limits of possible message rates range. When sampling no data, a consistent rate of 115000 messages/s over the full payload size spectrum is reached. This is due to the usage of shared memory - receiving and accessing messages does not add any significant overhead. For payloads larger than 16 kB the cost of copying memory becomes clearly visible and shows a proportional relationship - multiplying the payload size drops the message rate by a roughly common factor. Still, transferring data

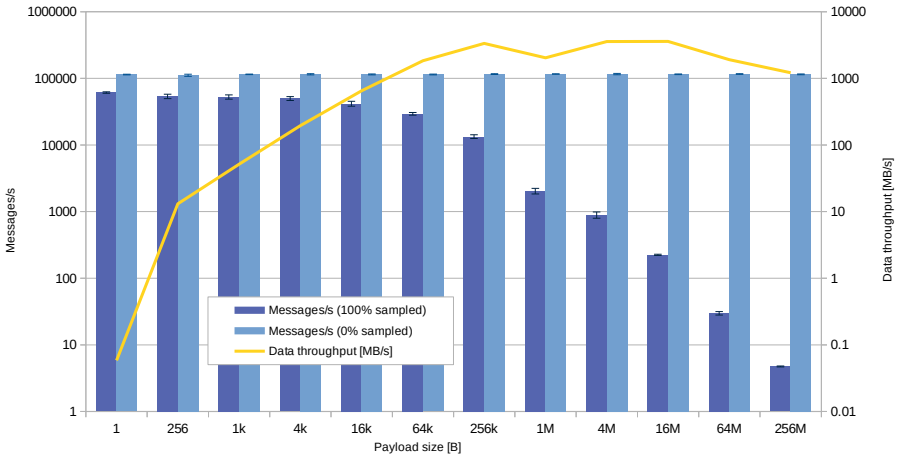


**Figure 3.** The performance of Dispatcher as a function of the message payload size.

**Table 1.** The performance of publishing 2D histograms by QC Tasks.

| Object size (in RAM) | 1 kB | 10 kB | 100 kB | 1 MB | 10 MB |
|---|---|---|---|---|---|
| Number of objects [/s] | >100000 | >30000 | >8100 | >1000 | >70 |

in larger chunks proves to be more efficient, as the highest data rates of about 3.5 GB/s are reached for 4–16 MB.

The QC Task maximum data input rates were found in a similar fashion. The highest message rates are achieved for payload sizes smaller than 4 kB, reaching 75000–85000 messages/s. On the other hand, one QC Task can process higher data volumes (~1 GB/s) for payload sizes in the range of 64 kB to 256 MB.

QC Objects are published in QC Tasks at the end of each cycle. Table 1 shows how many objects (2D histograms) one QC Task can publish per second. Again, the general trend indicates that the same amount of information is published more efficiently in larger chunks, reaching a maximum at around 1 MB.

The Mergers can be configured to work in different modes and it is yet to be found out what are their optimal settings. To have already a basic understanding of an expected performance, a rate of 1D histograms processed by one Merger process has been measured. Fig. 4 presents the results as a function of object's size. The performance drops linearly for objects larger than 16 kB, which suggests that the transport and summing up the histogram's bins takes the biggest toll.

Fig. 5 shows the framework performance of running empty checks (returning always the same Quality). Again, an influence of the size of the object on the message rate is observed, which is related to the transport cost. The total check rate increases when increasing the number of checks per object, especially for bigger objects.

The QC database was tested with 20 producer tasks publishing 20 objects per second. In this setup the database can sustain a throughput up to 1 MB objects.

The presented benchmarks did not include any real data quality control algorithms in order to measure the overhead of the framework only. The results indicate that the overhead should not have a big significance over the planned user code. The biggest factors which

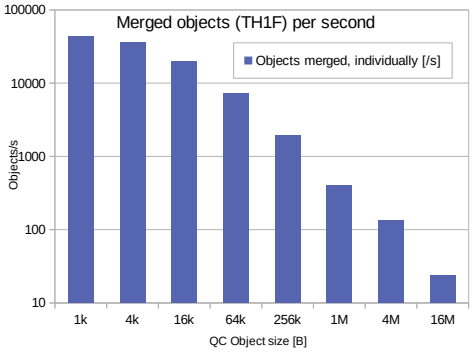


**Figure 4.** The performance of Merger as a function of the histogram size.

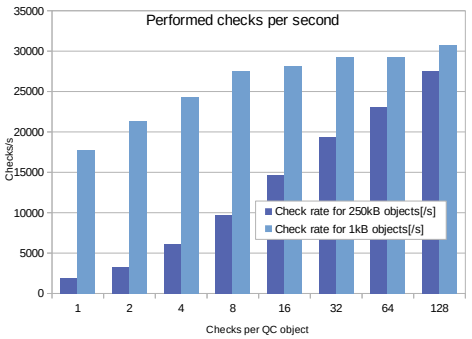


**Figure 5.** The performance of Checker as a function of the checks number per object.

slow down the framework are closely related to the amount of data needed to be transported between processes.

## 4 Status

Each component of the framework is already available and confirmed to work on standalone and development setups. The current version of Mergers offers multiple configuration options, such as a specification input objects' timespan (full histories or differences), or an option to merge objects in collections to benefit from potential processing speed-up. Having it as a base, the maximum rate of merged objects will be measured for different configurations. Knowing the best configuration will allow to remove unnecessary features, which would otherwise increase the code complexity, making it more difficult to mantain during the software's expected lifetime.

The ALICE experiment is now entering the recommissioning phase. The subdetector teams are already benefiting from the QC framework and developing their modules. For example, the software is used to measure the noise of the readout pads of the Muon Chambers detector (MCH) (see Fig. 6) and to monitor the occupancy, 2D hit maps and general status of the Inner Tracking System. Regular meetings are scheduled with the detector teams to discuss and plan their contribution to the QC and all the code is reviewed before merging.

## 5 Future work

The deployment of multinode setups has just started. During the upcoming time the data distribution between FLPs and EPNs will be integrated, as well as different software components, including e.g. the Control [10], Logging, Monitoring [11] and Configuration software. It will be the occasion to identify and fix the problems previously unseen on simpler setups. Mergers will be further benchmarked in order to find the most efficient configuration and a simplified version will be prepared. The configuration of the QC software is planned to be simplified by introducing a dedicated GUI which should replace the configuration files currently used.

Before the start of data-taking in the middle of the year 2021, the full $O^2$ system will undergo several test runs, including a simulation challenge and test runs with real data on

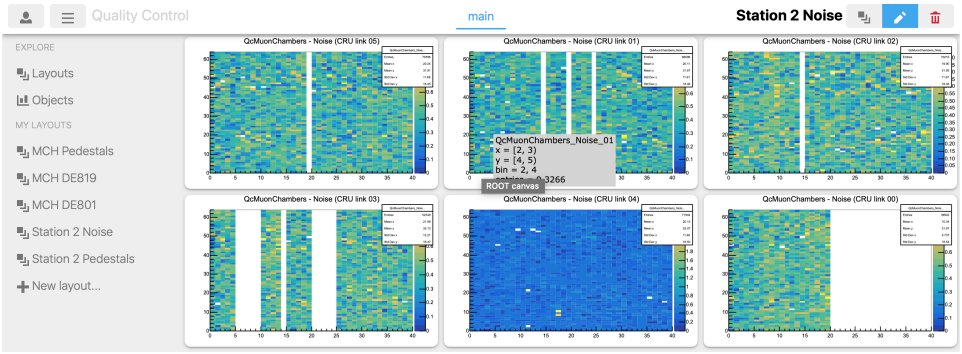


**Figure 6.** Pads' noise level of the Muon Chambers detector as shown in the QC GUI.

10% of the system and later on the full system. The Quality Control software will play an important role there as a tool to quickly identify possible problems with the data itself and indirectly with the detectors and data processing.

## References

[1] K. Aamodt et al. (ALICE Collaboration), Journal of Instrumentation **3**, S08002 (2008)
[2] N. Cabibbo, G. Parisi, Phys. Lett. **59B**, 67 (1975)
[3] B. Abelev et al. (ALICE Collaboration), J. Phys. **G41**, 087001 (2014)
[4] P. Buncic, M. Krzewicki, P. Vande Vyvre, Tech. Rep. CERN-LHCC-2015-006. ALICE-TDR-019 (2015), `https://cds.cern.ch/record/2011297`
[5] G. Eulisse, P. Konopka, M. Krzewicki, M. Richter, D. Rohr, S. Wenzel, EPJ Web of Conferences **214**, 05010 (2019)
[6] A. Rybalchenko, M. Al-Turany, GSI Report 2014-1 p. 283 (2014)
[7] B. von Haller, P. Lesiak, J. Otwinowski, Journal of Physics: Conference Series **898**, 032001 (2017)
[8] R. Brun, F. Rademakers, Nuclear Instruments and Methods in Physics Research A **389**, 81 (1997)
[9] B. Bellenot, S. Linev, Journal of Physics: Conference Series **664**, 062033 (2015)
[10] T. Mrnjavac, EPJ Web of Conferences **These proceedings** (2020)
[11] A. Wegrzynek, G. Vino, EPJ Web of Conferences **These proceedings** (2020)