DAQExpert - An expert system to increase CMS data-taking efficiency

Maciej Gladki

on behalf of the CMS Collaboration, CERN, Switzerland

E-mail: maciej.gladki@cern.ch

Abstract. The efficiency of the Data Acquisition (DAQ) of the Compact Muon Solenoid (CMS) experiment for LHC Run-2 is constantly being improved. A significant factor on the data taking efficiency is the experience of the DAQ operator. One of the main responsibilities of the DAQ operator is to carry out the proper recovery procedure in case of failure in data-taking. At the start of Run-2, understanding the problem and finding the right remedy could take a considerable amount of time, sometimes up to minutes. This was caused by the need to manually diagnose the error condition and to find the right recovery procedure out of an extended list which changed frequently over time. Operators heavily relied on the support of on-call experts, also outside working hours. Wrong decisions due to time pressure sometimes lead to an additional overhead in recovery time.

To increase the efficiency of CMS data-taking we developed a new expert system, the DAQExpert which provides shifters with optimal recovery suggestions instantly when the failure occurs. This tool significantly improves the response time of operators and the success rate of recovery procedures. Our goal is to cover all known failure conditions and to eventually trigger the recovery without human intervention wherever possible. This paper covers how we achieved two goals - making CMS more efficient and building a generic solution that can be used in other projects as well. More specifically we discuss how we: determine the optimal recovery suggestion, inject expert knowledge with minimum overhead, facilitate post-mortem analysis and reduce the amount of calls to on-call experts without deterioration of CMS efficiency. DAQExpert is a web application analyzing frequently updating monitoring data from all DAQ components and identifying problems based on expert knowledge expressed in small, independent logic-modules written in Java. Its results are presented in real-time in the control room via a web-based GUI and a sound-system in a form of short description of the current failure, and steps to recover. Additional features include SMS and e-mail notifications and statistical analysis based on reasoning output persisted in a relational database.

1. Introduction

The data-taking efficiency of the Compact Muon Solenoid (CMS) [1] experiment at CERN in 2016 during proton-proton physics, understood as the recorded luminosity, was 92.7% according to Web Based Monitoring (WBM) [2]. One of the major roles in the operations is played by the Data Acquisition (DAQ) system which controls the detector's readout, event-building and operations of the event filtering farm. The smooth operations of the experiment and its resulting, high efficiency is ensured by constant supervision of crew of operators and human experts.

Being able to act within seconds during operations of LHC is one of the challenges of the operator. The DAQ system is a complex, therefore there are tools to automate certain operations, e.g. starting the run, stopping, resetting, resyncing, configuring etc. Those tools facilitate the actions initiated by operators, however, it's not uncommon that the system goes into erroneous state where deeper understanding of the system is required to overcome the problem in the optimal way. Experts of all subsystems prepare instructions for operators for dealing with specific problems. These are recovery instructions needed to resume the data-taking as soon as possible. Operators need to effectively select and issue them to make sure the operations of CMS go smoothly.

Over a time the list of recovery instructions grew. Despite its straightforward and easy form, it takes considerable amount of time for operator to find the correct remedy when needed. This part of intervention, the reaction time, considered to last between a problem occurrence and a subsequent recovery action taken, is one of two concerns. Wrong or suboptimal decisions were not uncommon as well. These led to increased recovery time, considered to last between a recovery action beginning and its end. Both reaction time and recovery time are referred as intervention time.

Reducing the reaction time and selecting the optimal recovery action are the two areas for improvement that will reduce the overall intervention time and directly affect the efficiency of CMS. There are also other chores that this project aims to streamline. First is to facilitate the post mortem analysis for DAQ experts, another is to aggregate and to report a statistical analysis of the performance of the system and individual subsystems. Last but not least is to reduce the need of a external help especially outside of working hours without a deterioration of CMS efficiency.

1.1. Selecting the optimal recovery action

The time spent recovering is the main contributor to the overall intervention time in CMS and amounts to 82%. Therefore its essential to define and select the quickest set of actions needed to bring the system back to operation. Many different recovery procedures may be distinguished, e.g.: stoping and starting the run, (re)configuring a specific subsystem, (re)initializing a subsystem, sending resynchronization or hard reset commands through the Timing Trigger and Control system (TTC). They differ significantly in terms of impact they have on the system and how much time they take to complete. Some of them take few seconds whereas others few minutes. One of the observed suboptimal decisions is issuing by the operator the most versatile and heavy recovery action for most problems. It gives the highest chance to recover but in most cases it means an overhead in recovery time. The aim of the project is to help operator make optimal decisions.

1.2. Reaction time

The time from the problem occurrence to the first recovery action is a second area to improve. There are sound notifications in the control room to attract attention of the shifter in a case data flow stops. Identification of the problem and selecting the recovery action takes from couple of seconds up to minutes. Figure 1 presents the histogram of a reaction time over the period of 6 months of data-taking (August 2016 - July 2017, excluding Year End Technical Stop (YETS) that lasted December 2016 - April 2016, month May of 2017 was excluded due to special conditions of data-taking), hereinafter referred as the *comparison period*. Most of the times the operator reacted in the time 10-20 seconds - 33 occurrences. The aim of the project is to reduce the reaction time.

1.3. Minimizing external help

In case the operators cannot fix the problem themselves they can always rely on help of DAQ on-call experts who are available 24/7. The scheme where operators cope with problems without the need of external help is noteworthy. Minimizing the number of these calls especially outside of working hours without a deterioration of CMS efficiency is one of the goals of the project.

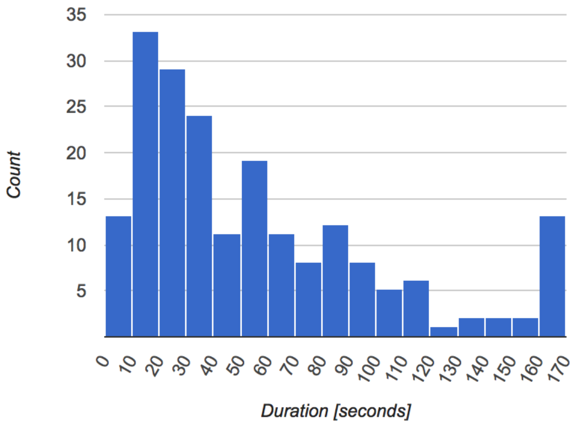


Figure 1. Reaction time histogram.

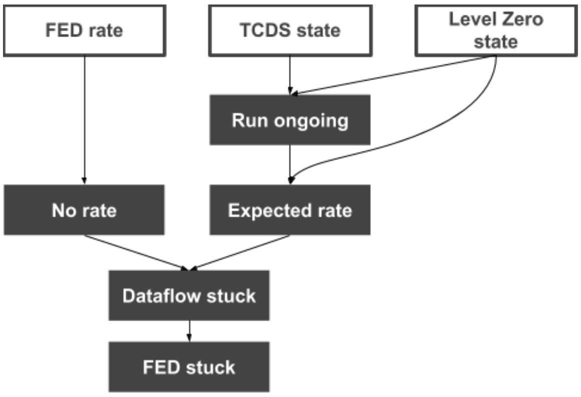


Figure 2. Subset of Logic Module hierarchy.

1.4. Facilitating post mortem analysis

After the problem occurs and is addressed, usually there is a lesson to be learnt from it to avoid similar problems in the future. Numerous monitoring systems enable the access to data necessary to analyze the problem albeit with some caveats. Some data are available after the problem has been resolved, some only while the problem is ongoing. The aim of the project is also to streamline the post mortem analysis by making both historic and real-time data easily available in the same way at any moment.

2. Expert system and architecture

Particular external forces influenced the design of the project and technologies used. Specifically no rule-based engine framework has been used to implement reasoning and define expert knowledge.

The system consists of three services. Firstly the *Snapshot Service* aggregates all monitoring information into a snapshot and persists it. Secondly the *Reasoning Service* applies the knowledge of the experts on the snapshots. Finally the *Notification Manager* manages and delivers notifications, including e-mail and live suggestions and sound notifications in the control room. The *DAQExpert* brings all parts together in a form of a web application allowing to browse the whole history of monitoring data, to view the analysis and to manage notifications.

2.1. External forces

It is highly likely that the project's lifetime, like in many other cases at CERN, will be more than a decade and will be maintained by different people over time. Thus, one of the main constraints was to avoid short-lived and highly specific technologies that would make it difficult to new team members to quickly become productive. It needs to be easy to understand and modify requiring little effort to learn. Rule-based engines have beed evaluated in the past by the CMS DAQ group and were considered not suitable due to the steep learning curve. Introducing additional language was not an option. The solution needed to be based on current skill set of the team members and as simple as possible.

Moreover, the nature of the project foresees frequent logic-related updates possibly introduced by many authors. If there is a new recovery solution advised by sub-system experts the according knowledge needs to be introduced in to the system. Finally a high availability needs to be assured, as each downtime of the system means no support for the operators and possibly extended downtime of CMS.

As a result the new expert framework has been designed from ground up. It allows to define the knowledge in imperative language popular in the group - Java. The knowledge is organized in small independent Logic Modules which are building blocks of the system.

2.2. Monitoring, aggregation and data flow

There are multiple steps before information from data sources is available to the clients. First the monitoring data sources are queried to get information about data taking health. There are multiple heterogenous sources of monitoring data. They differ not only in terms of what data is available but also when and how quickly the data can be retrieved. Diagnosing the problems in data-flow in real time requires quick access to key information. Post mortem analysis requires access to all relevant data at any time. To enable both, a so-called snapshot of the system is taken periodically. It contains all necessary information to identify and understand the problem. Thus, the first step was to aggregate all the information in snapshots that brings both structural data and instantaneous state of all components. The scope of the snapshot has been designed to give full picture of the system's state at a given moment. Both real-time and post-mortem analysis are based on the same data from snapshots.

While the snapshot is persisted for possible post-mortem analysis it is simultaneously sent to the *Reasoning Service* for real time analysis. As a result of the analysis, notifications may be yielded that will be later dispatched by the *Notification Service* and delivered to the clients. On the way, all of the services persist their results enabling to browse historic data.

2.3. Knowledge definition

Its common in the industry that the expert knowledge is defined in a declarative language. For instance in a form of if-then-else rules expressed in a high-level rules language. As a result of external forces this way of defining knowledge has not been used. The new custom made framework based on *Logic Modules* has been introduced. A *Logic Module* (hereinafter LM) is a building block of *DAQExpert* expert logic. Each LM focuses on one thing: expressing piece of knowledge about DAQ system:

- each LM defines one condition,
- the definition of condition is placed in the satisfy method,
- the method returns true if condition is satisfied and false otherwise,
- one LM can use results of another LM.

The knowledge defined in LMs is used to find the optimal solution when a problem occurs in the data-flow. The set of LMs is defined by DAQ experts. LMs operate on the snapshot and verify if key parameters are in expected states or in specified value ranges. As LMs may use outputs of other LMs there is a predefined order of firing them. Figure 2 presents a subset of LMs responsible for identifying one of many problems that may occur during data-taking. In the example there are five LMs and three parameters being monitored. The LMs are fired in top-down order allowing LMs rely on others as indicated by arrows.

A LM may play two roles: it might be a sub-step of the analysis (*Expected rate*, *No rate*, *Dataflow stuck*) and final step of the analysis (*FED stuck*, FED - Front End Driver) delivering a description of the condition and a recovery suggestion. Finding optimal recovery to a given condition in this hierarchy means that the chain of LM was activated:

- *Run ongoing* - there was a run started according to *TCDS state* (TCDS - Timing and Control Distribution System) and *Level Zero state* (Level Zero - Top level function manager),
- *No rate* - the average *FED rate* was equal to 0, there was no data flowing

- *Expected rate* - the *TCDS state* and *Level Zero state* indicate that no recovery action is being issued at the moment, all subsystems are running and the rate is expected to be non-zero, it also checks if there is currently a run ongoing using output of *Run ongoing* LM
- *Dataflow stuck* - this summarizes outputs of two LMs: *No rate* and *Expected rate*, it's activated when two related LMs are active meaning that the data-flow is stuck,
- *FED stuck* - starts the analysis when there is *Dataflow stuck*, it performs specific checks that reveal the specific FED to be stuck, causing the data-flow to be stuck, this LM identifies precisely the problem thus it consists final instructions for the operator including detailed description of the problem and recovery suggestion.

3. Results

The current version of the service (2.9.0) already improves all of the areas described in the first section. Operators of the DAQ system at CMS have now at their disposal a tool delivering recovery action suggestions in the real-time - the Dashboard (see figure 3). Whenever the problem occurs the tool is showing the description of the problem and the optimal steps to recover. It helps shifters to take right decisions, reducing both reaction time and need of external human expert help.

Measuring the success of the project was quite challenging task. The ultimate goal is to increase the efficiency of data-taking at the CMS. However the overall efficiency is not an adequate parameter as there are many different factors that interfere. The detector is constantly being improved, there were major upgrades in both hardware and software that influenced the final efficiency of the CMS. The feedback of operators and human experts is important as they are the target user groups of the system. As much as they appreciate the new system, the verbal feedback cannot be used as a measure of success as it is not quantitive and it is subjective. Three parameters have been identified to adequately represent the success of the project: accuracy of the recovery action selection, reaction time of the operators and demand of the external help. They are practically independent of other factors which make them reliable means of comparison.

The DAQExpert was introduced in mid 2016 but operators were instructed to use it in the early 2017. All comparisons are based on *comparison period* (see section 1.2) which gives four months from 2016 where there was no *DAQExpert* support and two months from 2017 where the support was available.

3.1. Recovery selection correctness

The improvement was mostly observed in the number of optimal recovery decisions taken by operators in case of data-flow upsets. As stated before the recovery time is major part of total intervention time (82%), thus it is essential to minimize it by choosing the most appropriate and time-efficient recovery actions (optimal recovery actions).

There are two main categories of recovery actions: heavy - lasting usually for 3-4 minutes, initiated by e.g. stopping the run; and light - lasting 7-10 seconds implemented with commands through the TTC system. Suboptimal decisions mean unnecessary time spent on recovery or irrelevant action being performed that will not solve the problem leading to same problematic condition after its completion. On average it means that each suboptimal decision adds roughly 93-125 seconds of overhead to recovery time.

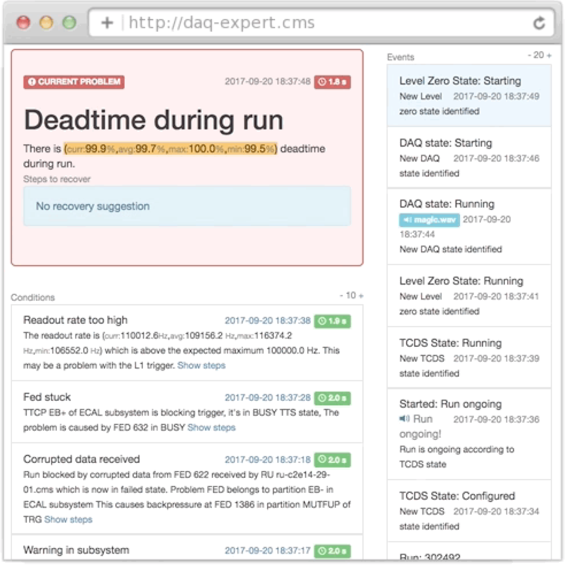
In 2016, during data-taking, operators had no guidance and out of 36 data-taking failures, 32 times the optimal decision was taken. In 2017 there were 29 data-taking failures and all of the decisions were optimal.

3.2. Reaction time

This may not be the most important area to improve as it is minor part of total intervention time - 18%. There is high variation depending on individual operator capabilities. The reaction time has slightly improved from 67 seconds in 2016 to 60 seconds in 2017. This is an area where human factor plays great role and bypassing the operators will certainly assure better results.

3.3. Non-quantitative results

An improvement in the field of post mortem analysis has been made as well. In case of more demanding data-flow upsets where further investigation is required the *DAQExpert* streamlines the chores of human experts. First of all the go-back in time functions has been introduced that enable human experts to access the historic data easily with the tool *Browser*, shown in figure 4. Second of all there are more data available to support post-mortem analysis than ever before. All parameters of the system are kept for some period of time when debugging demand is more likely. Later on the resolution of data is reduced so that the disk space requirements are met.



http://daq-expert.cms

CURRENT PROBLEM 2017-09-20 18:37:48 1.8 s

Deadtime during run

There is (curr:99.9%,avg:99.7%,max:100.0%,min:99.5%) deadtime during run.

Steps to recover

No recovery suggestion

Conditions - 10 +

Readout rate too high 2017-09-20 18:37:38 1.9 s

The readout rate is (curr:110012.6Hz,avg:109156.2 Hz,max:116374.2 Hz,min:106552.0 Hz) which is above the expected maximum 100000.0 Hz. This may be a problem with the L1 trigger. Show steps

Fed stuck 2017-09-20 18:37:28 2.0 s

TTCP EB+ of ECAL subsystem is blocking trigger, it's in BUSY TTS state, The problem is caused by FED 632 in BUSY Show steps

Corrupted data received 2017-09-20 18:37:18 2.0 s

Run blocked by corrupted data from FED 622 received by RU ru-c2e14-29-01.cms which is now in failed state. Problem FED belongs to partition EB- in ECAL subsystem This causes backpressure at FED 1386 in partition MUTFUP of TRG Show steps

Warning in subsystem 2017-09-20 18:37:17 2.0 s

Events - 20 +

Level Zero State: Starting
New Level 2017-09-20 18:37:49
zero state identified

DAQ state: Starting
New DAQ 2017-09-20 18:37:46
state identified

DAQ state: Running
magic.wav 2017-09-20 18:37:44
New DAQ state identified

Level Zero State: Running
New Level 2017-09-20 18:37:41
zero state identified

TCDS State: Running
New TCDS 2017-09-20 18:37:39
state identified

Started: Run ongoing
Run ongoing! 2017-09-20 18:37:36
Run is ongoing according to TCDS state

TCDS State: Configured
New TCDS 2017-09-20 18:37:34
state identified

Run: 302492

Figure 3. Dashboard - tool to show suggestions in real time.

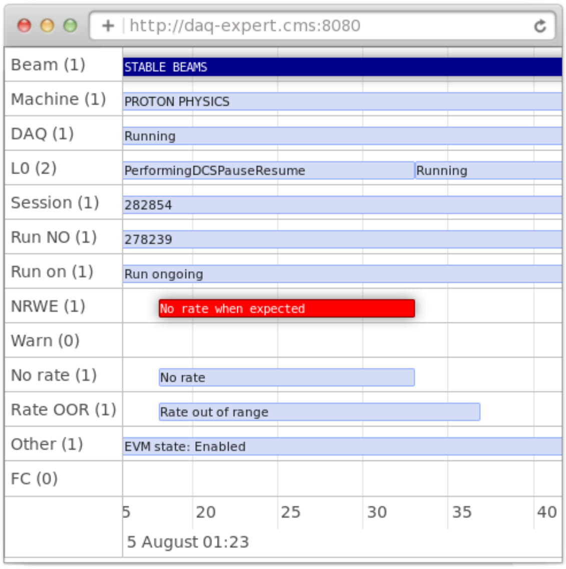


Figure 4. Browser - tool to browse in time to enable post mortem analysis.

3.4. Summary

Both quantitative and non-quantitative results were presented in this chapter. The recovery action selection accuracy was improved by introducing the expert system. There were no suboptimal recovery decisions observed anymore, whereas before they happened 11% of time. Additionally the reaction time shrunk from 67 to 60 seconds. It means that the total intervention time, including reaction time and recovery procedure time is now shorter by at least 10 seconds on average. One major improvement to be implemented in the future is to bypass the operator whenever possible. This will shrink the reaction time to few seconds. External help demand was not quantified yet. This area is of great interest for on-call human experts as it has a potential of limiting the intensity of their recurrent shifts. The number of calls (normalized by the number of problems) to the on-call experts before and after *DAQExpert* introduction may reveal trends in this area.

References

[1] CMS Collaboration, JINST 3 S08004 (2008)

[2] https://cmswbm.cern.ch