

Antonio Vivace

Modernising the CERN CMS Trigger Rates Monitoring software

Master Thesis Summary

CERN, LHC and CMS High Energy Physics experiments involve large amount of complex systems subject to anomalies and malfunctioning. They generate a lot of high dimensional data that must be monitored and analysed by various specialised teams to validate and deliver certified data for physics analysis. Most of these tasks are still carried out completely by humans, by looking at highly-contextual data and plots in complex workflows. Machine Learning techniques and approaches are being evaluated to aid and possibly automatize these processes.

CERN, the European Organization for Nuclear Research, operates the Large Hadron Collider (LHC), the world's largest and highest-energy particle collider, one the largest machine in the world.

2500 staff members design, construct and operate the research infrastructure, in addition to more than 17500 users and scientists of 110 nationalities, from institutes in more than 70 countries.

The Compact Moun Solenoid (CMS) detector in the LHC is a complex system which needs fast and reliable monitoring. Quick feedback on each of the subsystems is essential to spot and solve problems or the data taken might be unusable or not interesting for physics analysis. Experts from different systems need to correlate information to investigate underlying problems.

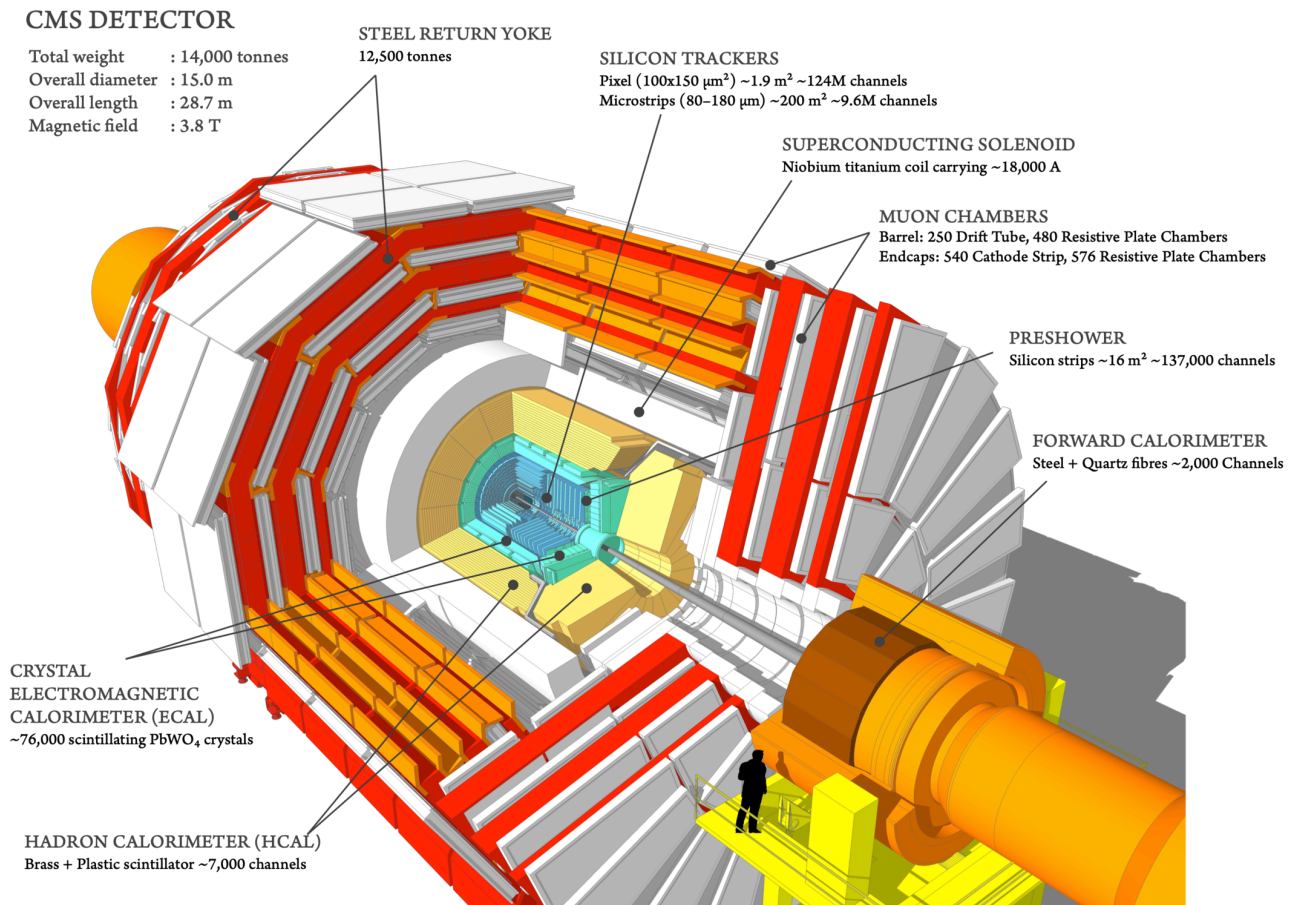


Figure 0.1: Cutaway diagram of the CMS detector [14]

A centralized monitoring solution must expose real time data, historical information, summaries and reports from a series of heterogeneous sources. The Web Based Monitoring (WBM) software covered this role since the commissioning of the CMS experiment (2008), evolving and integrating new services into a growing framework during LHC Run 1 (2010-2013) and Run 2 (2015-2018).

Monitoring the Trigger System A *Trigger system* is responsible of filtering the large majority of events, spotting the potentially interesting ones, triggering the detector's read-out system to retain data from the selected collisions.

The CMS experiment generates millions of events per second, each one carrying payloads averaging ~ 1.5 MB [5]. One of the major challenges is the task of reducing event rate from roughly 40 MHz down to a more manageable 1 kHz while keeping as many interesting physics events as possible. This is accomplished through the use of a Level-1 (L1) [1] hardware based trigger as well as a software based High-Level Trigger (HLT) [12].

This crucial phase is also sensible to malfunctions of many of the underlying parts, from sub detectors to the trigger algorithms configurations: monitoring the rates of such filters is essential to spot any anomalous behaviour in the underlying (sub)systems, software and/or hardware configurations, network and detector malfunctions.

A *Trigger* is a series or reconstruction steps and filters trying to select specific physics signatures like high momentum leptons or large energy deposits in the calorimeters [15].

Monitoring and understanding the output rates of the L1 and HLT triggers is of key importance for determining the overall performance of the trigger system and is intimately tied to what type of data is being recorded for physics analyses.

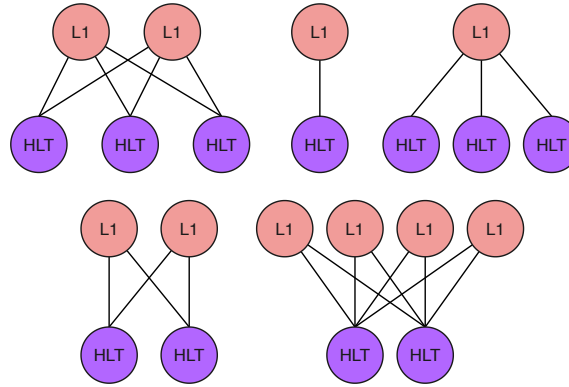


Figure 0.2: Simplified graph inspired by the trigger system configuration: the triggers connections can be seen as hierarchical directional graph. Red nodes represent L1 Trigger while purple HLT. Each link is directed from the red nodes to the purple ones. In reality, few hundred nodes spread approximately equally between HLT and L1 Trigger. [13]

This work concerns RateMon, the software framework providing the monitoring data on this Trigger system: it exposes Trigger Rates data, querying the CMS Online Master Database System (OMDS) and carrying out **normalisations** and **corrections** for a number of different configurations and conditions, allowing consistent comparisons. It is powered by ROOT [4], an open-source data analysis framework developed by CERN, and its Python interface, PyROOT.

Trigger Rates are presented in the form of *Rates (Hz) VS Pile Up luminosity (PU [9]) plots*. It is also responsible of **alerting** the Trigger Shifters staff when rates deviate too much from the **predicted** values. Those predictions are based on analytical models fitted on data collected in previous runs.

This software is also used by shifters in the CMS Control Room to monitor Trigger Rates and spot potential problems during the LHC runs.

Upgrading the monitoring framework During the second Long Shutdown of the LHC (2018-2021) the CMS detector will be upgraded and many CMS sub-systems will drastically change. WBM started

to show its age and problems: it unexpectedly and heavily grew with new features, arriving at a point where services were using vastly different technologies and it became harder and harder to maintain and expand. It has been decided to deprecate WBM in favor of a new software framework, called Online Monitoring System (OMS), decoupling the User Interface (UI) from the Aggregation (Data) Layer.

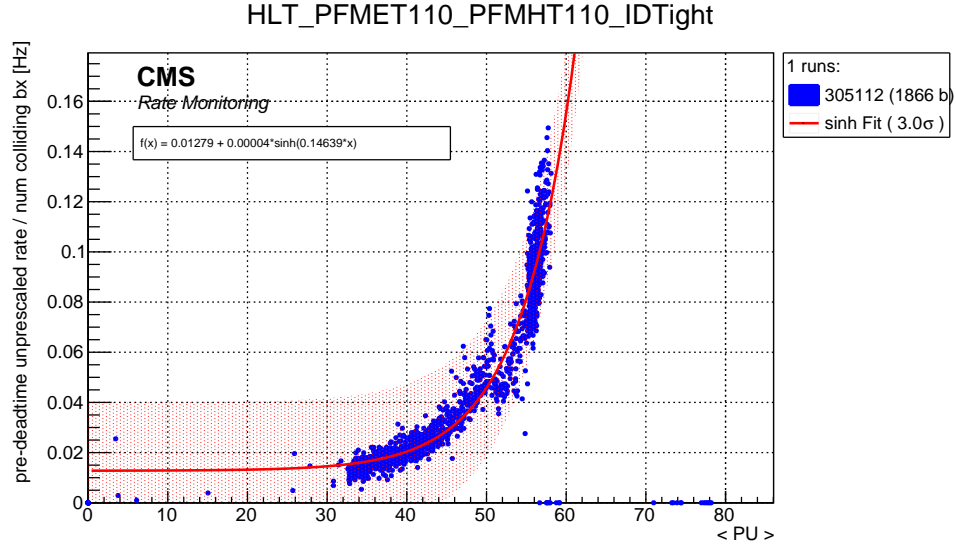


Figure 0.3: Rates for an HLT Trigger plotted with a fitted function on run 305112

Overview We proceed to improve and renovate the set of tools included in RateMon, getting them ready to be migrated to the new OMS infrastructure: the tool can be now run and deployed in a reproducible and isolated way. A new CI/CD pipeline is implemented to streamline the development process and deliver software updated directly to the Control Room machines. A new API surface exposes the software analysis, model fitting and plotting capabilities.

The following goals have been achieved and delivered in production [3] [2] [10] [11]:

- Upgraded the tools to run on a recent and supported version of Python and PyROOT.
- Moved the database connection configuration class to an configuration file.
- Formally defined the environment and the dependencies needed to run the software.
- Prepared scripts to package the software with the RPM package format, to be compliant (and integrate) with the Control Room deployment procedures. Prepared a basic systemd service to include in the package, to allow running the software as a system service.
- Prepared a Continuous Integration and Deployment pipeline running on the CERN infrastructure.
- Upgraded the CI/CD pipeline to use the CMS Cactus [7] auto DevOps tools.
- Upgraded the exporting feature of the tools, previously limited to shell output and small PNG image renders.

- Enabled the tools to export trigger rates data in ROOT binary files, allowing ROOT clients to read them.
- Enabled the tools to export trigger rates data also in a agnostic format, not tied to ROOT.
- Updated the documentation adding a way to consistently prepare any machine to run RateMon tools, without being tied to a particular machine cluster at CERN.
- Enabled any user to import the RateMon software as a Python module, adding some external-facing and transparent methods to get Trigger Rates as raw data or plots.
- Investigated a solution to integrate RateMon into the new CMS OMS, replacing the old WBM, now deprecated.
- Implemented an API exposing the exporting, processing and plotting functionalities of the RateMon software. This API is formally defined with an OpenAPI 3 Schema and an auto generated Swagger interface is available as documentation to users. This API can be plugged into the OMS stack to provide the old plots and many new features.
- Implemented a reactive web application in VueJS to demonstrate and showcase the API capabilities, plotting Trigger Rates (and their predicted values, based on the fits) in the browser with interactive and scalable plots (powered by JSROOT) instead of static image exports.
- Implemented a Trigger Selection interface on the UI, a feature requested by Trigger shifters and coordinators.
- Reported a series of bugs happening in the data processing phase of the scripts, detailing test cases and scenarios triggering them, allowing future bug fixing work.
- Deployed the implemented stack in production and in the CMS Control Room. Set up a separate data storage solution to cache computed data, limiting the bottleneck effect of the CMS database on the API and offering faster responses.

Building a new dataset to aid automated Anomaly Detection approaches Trying to fit traditional Machine Learning (ML) pipelines and models in this type of scenario is not a novel idea: an elaborated research study has been conducted on Anomaly Detection and Machine Learning approaches to similar problems at CERN: from Variational Auto Encoders solutions [13] aiming to improve the analytical models to fit and predict trigger rates to isolation forest and gaussian mixture models [8] to detect anomalies in the LHC injection magnets.

Similar papers and state of the art studies have been surveyed to frame and formally define our problem.

Obtaining large, proportionally labeled datasets and the intrinsic difficulty to codify and formalise the human expertise in a "consumable" form for traditional ML pipelines appeared as the most critical points of these attempts.

Therefore, we exploited the acquired knowledge on the software sources to produce a new integrated and labelled dataset.

Additional work has been done to properly export data from another important source at CMS, the new Run Registry [6]. It describes the status over time of every part of the detector, including the ones for which the failures are identifiable from the monitoring of the trigger rates, the process currently carried out by humans by looking at trigger plots, reference fits and other detector diagnostic data.

These improvements enabled trigger rates time series and detector data to be properly exported and integrated in a new, rich dataset of genuine and labeled anomalous runs, possibly providing a new starting point for new approaches to automatize this task.

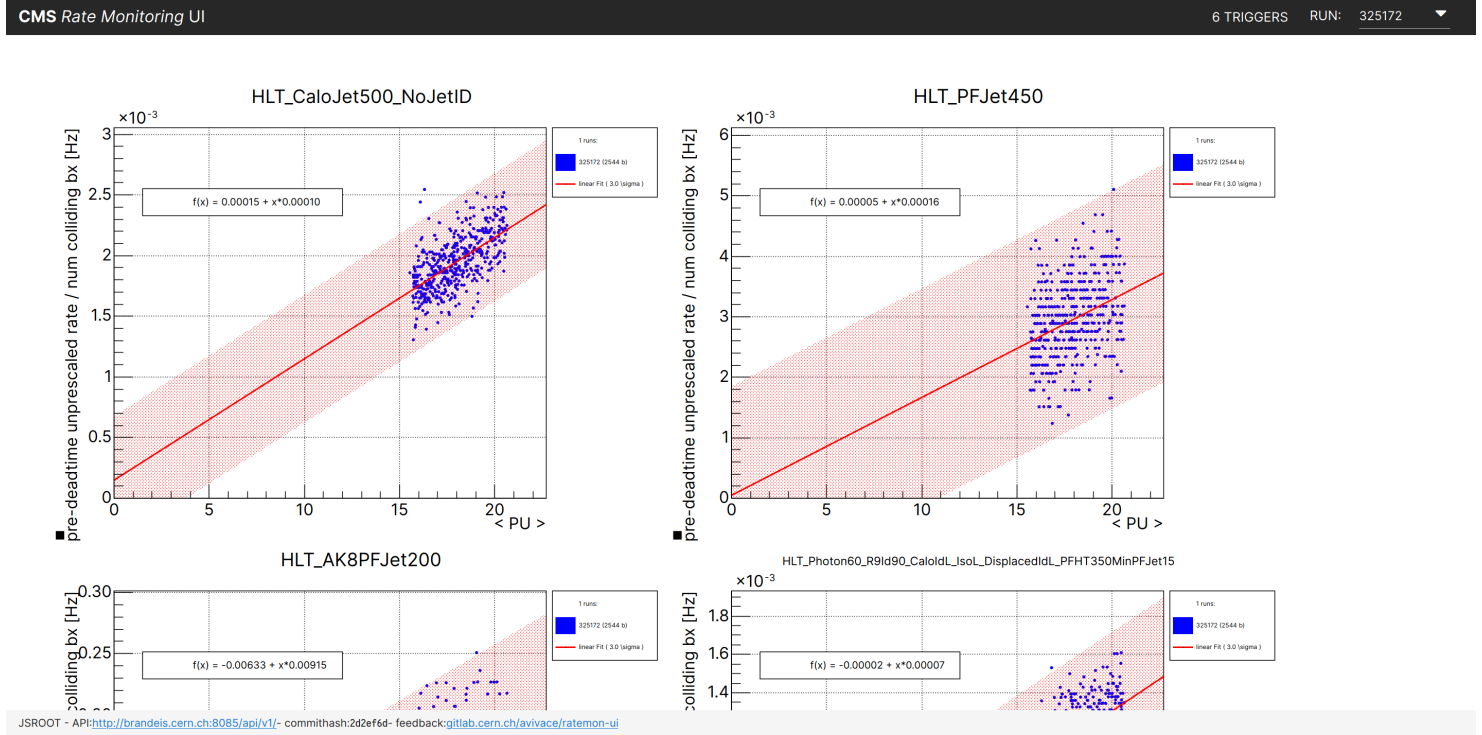


Figure 0.4: Main RateMon UI application view, showing interactive trigger rate plots and relative fit functions

Bibliography

- [1] CMS Collaboration (2000b). *CMS TriDAS project: Technical Design Report, Volume 1: The Trigger Systems*. Technical Design Report CMS. URL: <https://cds.cern.ch/record/706847>.
- [2] Antonio Vivace, on behalf of G. Dirkx, K. Mohrman, A. Wightman, J. Lawrence. “Integrating Rate Monitoring data on OMS”. In: *L1 DQM Meeting* (June 1, 2020).
- [3] Antonio Vivace, on behalf of G. Dirkx, K. Mohrman, A. Wightman, J. Lawrence. “Modernising CMS Trigger Rate Monitoring tools”. In: *CMS Trigger Studies Group meeting* (February 26, 2020).
- [4] R. Brun and F. Rademakers. “ROOT: An object oriented data analysis framework”. In: *Nucl. Instrum. Meth. A* 389 (1997). Ed. by M. Werlen and D. Perret-Gallix, pp. 81–86. DOI: 10.1016/S0168-9002(97)00048-X.
- [5] Srećko Morović on behalf of the CMS DAQ group. *The Run-2 DAQ System of CMS*. URL: https://indico.cern.ch/event/471309/contributions/1981075/attachments/1256226/1854567/sm-daq-at_lhc_2016_v3.pdf (visited on 10/12/2020).
- [6] CMS Collaboration and Fabio Alberto Espinosa Burbano. *The CMS Run Registry: Data Certification and Publication System tool*. Nov. 2019. DOI: 10.5281/zenodo.3599323. URL: <https://doi.org/10.5281/zenodo.3599323>.
- [7] G. Dirkx. “Online Software report”. In: *Level-1 Trigger Weekly Meeting* (May 5, 2020).
- [8] Armin Halilovic. “Anomaly Detection for the CERN Large Hadron Collider injection magnets”. Presented 28 Jun 2018. June 2018. URL: <http://cds.cern.ch/record/2665985>.
- [9] Werner Herr and B Muratori. “Concept of luminosity”. In: (2006). DOI: 10.5170/CERN-2006-002.361. URL: <https://cds.cern.ch/record/941318>.
- [10] K. Bunkowski and V. Sharma on behalf of A. Vivace, G. Dirkx, K. Mohrman, A. Wightman, J. Lawrence, B. Krikler, L1T DQM Team. “L1 Trigger OMS Developments”. In: *CMS Run Coordination Meeting* (April 28, 2020).
- [11] K. Mohrman, on behalf of A. Vivace, A. Wightman, J. Lawrence. “Update on RateMon status and plans for integrating into OMS”. In: *CMS Trigger Studies Group meeting* (May 13, 2020).
- [12] V. Khachatryan et al. “The CMS Trigger system”. In: *Journal of Instrumentation* 12.01 (Jan. 2017), P01020–P01020. ISSN: 1748-0221. DOI: 10.1088/1748-0221/12/01/p01020. URL: <http://dx.doi.org/10.1088/1748-0221/12/01/p01020>.
- [13] A. A. Pol et al. “Anomaly Detection with Conditional Variational Autoencoders”. In: *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*. 2019, pp. 1651–1657.

- [14] Tai Sakuma and Thomas McCauley. “Detector and Event Visualization with SketchUp at the CMS Experiment”. In: *Journal of Physics: Conference Series* 2 (), p. 22032. DOI: 10.1088/1742-6596/513/2/022032.
- [15] *Triggering and Data Acquisition | CMS Experiment*. URL: <http://cms.web.cern.ch/news/triggering-and-data-acquisition> (visited on 10/12/2020).