CMS ZDC DATA MONITORING FOR RUN 3*

Aivaras Silale, Mantas Stankevicius, Valdas Rapsevicius

Vilnius University, Lithuania

Cole Douglas Le Mahieu, Sorina Popescu, Michael Murray

University of Kansas, USA

Received 2 February 2023, accepted 23 September 2023, published online 27 October 2023

The CMS Zero Degree Calorimeters (ZDCs) are used to measure very forward and backward neutrons and photons from heavy-ion (and possibly *pp*) collisions at the LHC. Their purpose is to characterize the geometry of heavy-ion, photon–nucleus, and photon–photon collisions. The ZDCs are built from layers of tungsten and quartz fiber, and detect Cerenkov light produced by the showers of particles generated from incoming neutrons and photons. They will serve as a basic minimum bias trigger for 2022 PbPb run. To operate the ZDCs efficiently, it is vital to have a comprehensive monitoring system. This paper will present design considerations and results of prototype testing of the new ZDC monitoring system. This system operates within the framework of the CMS Online Monitoring System (OMS). A dedicated workspace for the ZDC allows for the organizing of monitored metrics in folders and pages. The most important metrics are energy distributions, the shower shape profile, and the single neutron peak. At a lower level charge and time distributions for individual channels are available. CMS OMS supports correlation of multiple data sources which allows for monitoring of rate per layer of ZDC average flux *versus* luminosity. Different pages give access to the current status of the detector as well as access to historical data.

DOI:10.5506/APhysPolBSupp.16.8-A2

1. ZDC

The two Zero Degree Calorimeters (ZDCs) [2] of the CMS experiment are located at ± 140 m from the collision point and measure very forward and backward neutrons and photons. The ZDCs complement the main CMS detector especially for heavy-ion studies. They reside in special detector slots

* Presented at *Excited QCD 2022*, Sicily, Italy, 23–29 October, 2022.

in the neutral particle absorber (TAN), which protects the first superconducting quadrupole magnet from radiation. The detector is built from layers of tungsten and quartz fibers. Data is digitized via the QIE electronics.

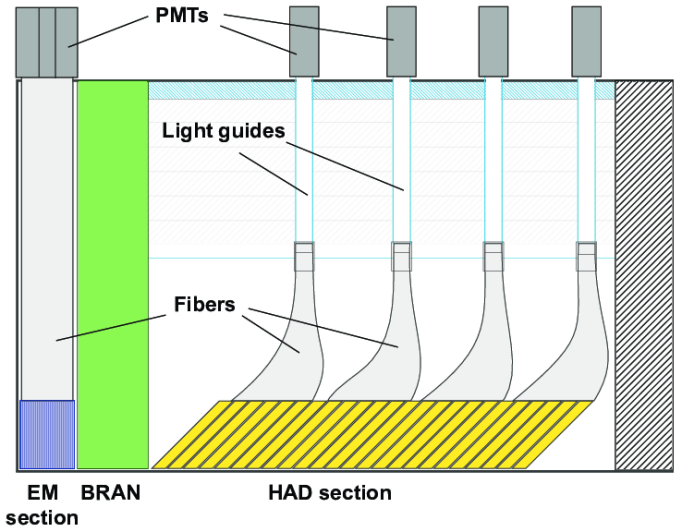

Fig. 1. The schematic side-view of the CMS ZDC.

2. Data preparation

Data collected by ZDC are stored in ROOT files in the EOS files system. In order to make it available via browser, data must be located, extracted, and transformed into a suitable format for loading into the Oracle database. Data extraction from ROOT files is a challenging task as different graph types (1D/2D) store data in a different format. Data is enriched with metadata which consists of 4 fields, see Table 1.

Table 1. Description of metadata.

Field	Definition
Name	Name of the graph
Local run number	Number of a test run in the lab
Fiber	ID of a fiber connected to the board
Channel	ID of a channel of a board

We use Python3 script to create the ETL pipelines, for extracting data from ROOT files we use the Uproot library [1]. Uproot is a library for reading and writing ROOT files in pure Python and NumPy. Uproot does not depend on C++ ROOT. Instead, it uses NumPy to cast blocks of data from the ROOT file as NumPy arrays.

Data is transformed and loaded into the Oracle database provided by the CERN services. Each graph has its own table:
`ZDC_ALL_SUM_CHARGE` [see Fig. 2], `ZDC_AC_PER_CHANNEL`, `ZDC_CAP_ID_PLOT`, `ZDC_MAX_CHARGE`, `ZDC_QIE10ETAPHISPACE`

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	PLOT_ID	NUMBER	No	(null)	1	(null)
2	NAME	VARCHAR2(255 BYTE)	No	(null)	2	(null)
3	FIBER	NUMBER	No	(null)	3	(null)
4	CHANNEL	NUMBER	No	(null)	4	(null)
5	DATA_ARRAY	CLOB	Yes	(null)	5	(null)
6	LOCAL_RUN_NUMBER	NUMBER	Yes	(null)	6	(null)

Fig. 2. `ZDC_ALL_SUM_CHARGE` table schema.

3. API

3.1. AGG API

The CMS OMS is divided into two layers — aggregation and presentation. RESTful JSON:API is responsible for communication between layers. The aggregation layer is responsible for collecting data from various sources, storage of transformed and pre-calculated (aggregated) values, and exposure of data via the RESTful API. Aggregation API was written using the JAVA programming language. Each subsystem which uses OMS has to use Virtual Machine (VM) in the CERN OpenStack.

3.2. ZDC endpoints

For development, the ZDC group has a dedicated VM in the CERN OpenStack project for aggregation API (AGG API for short). The OMS Core team provided a GitLab repository for us with all instructions. We had to pull the code and start to create endpoints. In order to create an endpoint, a user has to upload a JSON file for the DB table, which he would like to expose. After that, a user has to run Python3 script to create a boilerplate for endpoints. At this moment, ZDC has 6 endpoints which we use to filter and display the data, see Table 2.

Table 2. List of endpoints.

Endpoint name	URL
ADC per Channel	`/zdc-agg/api/v1/zdc/adcperchannel`
All Sums Charges	`/zdc-agg/api/v1/zdc/allsumcharges`
Cap Ids	`/zdc-agg/api/v1/zdc/capids`
Local Runs	`/zdc-agg/api/v1/zdc/localruns`
Max Charge	`/zdc-agg/api/v1/zdc/maxcharge`
Eta phi space	`/zdc-agg/api/v1/zdc/qie10etaphispace`

4. GUI

To operate the ZDCs efficiently, it is vital to have a comprehensive monitoring system. OMS is a web-based application to display data from various sources.

4.1. OMS

OMS — Online Monitoring System is a web-based Monitoring System, which is a main tool used to display data from various sources (real-time as well as historical data). OMS is used by many groups such as GEM, HGCAL, CTTPS, PPS, ECAL, *etc.* These groups have their own dedicated workspaces, where they can create many folders inside which users are allowed to create pages containing data tables, different types of charts such as Highcharts, Error Bar, HeatMap, Map, Pie Chart, *etc.* OMS is user-friendly and most of the work can be done with only the "drag and drop" action using the web interface. Originally, OMS was created with Python and React.

4.2. ZDC workspace

During our work with OMS, we have created 5 different types of charts: Linear chart, HeatMap, Error Bar, Scatter, and Histogram. Linear chart and Histogram were created only by using configuration, without programming effort. In order to create that kind of charts, we had to use generic charts which were created before. In order to create a chart from a generic template, we had to write a specific configuration which is shown below, see Fig. 3.

The other 3 types were created by coding. The HeatMap, Scatter, and Error Bar charts were created as a template. That allows all the OMS users to use these templates without writing any line of code, by only editing portlet configuration. An example of HeatMap which indicates the correct fiber every time data are received and number of ADC counts per channels is shown in Fig. 4.

Portlet Configuration For This Page (64000/663)

```
{
  "aggpath": "/zdc-agg",
  "endpoint": "zdc/capids",
  "filters": [
    {
      "attribute": "plot_id",
      "operator": "EQ",
      "value": 1234
    }
  ],
  "highcharts": {
    "chart": {},
    "colorAxis": {
      "maxColor": "yellow",
      "min": 0,
      "minColor": "#FFFFFF"
    },
    "plotOptions": {
      "histogram": {
        "accessibility": {
          "point": {
            "valueDescriptionFormat": "{index}. {point.x:.3f}
          }
        }
      }
```

Fig. 3. Portlet (Chart) configuration example.

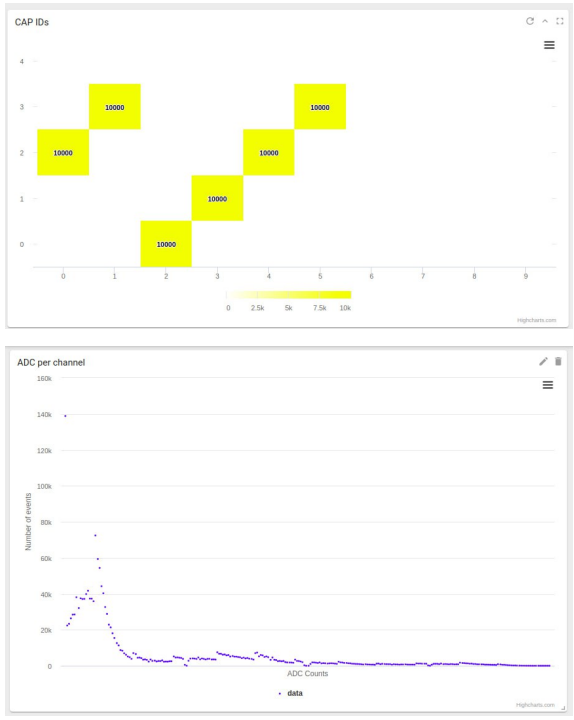

Fig. 4. Top: Reading the correct fiber every time data are received. Bottom: ADC count per channel.

5. Summary

The main goal of this project is to show how ZDC can leverage the CMS OMS framework to display data in an efficient and user-friendly way. We have created an ETL pipeline to extract data from the source systems, store it in an Oracle database, and create an aggregation API to expose the data. We have also created front-end assets using the CMS OMS framework to display the data in a visually-pleasing way. Our proof of concept is now ready for testing and integration.

REFERENCES

[1] J. Pivarski *et al.*, Uproot, September, 2017.
[2] O. Surányi *et al.*, *J. Instrum.* **16**, P05008 (2021).