Monitoring CMS experiment data and infrastructure for next generation of LHC run

Ceyhun Uzunoglu*, Felipe Gomez, Brij Kishor Jashal, Valentin Y Kuznetsov, Federica Legger, Benedikt Maier, Oscar Fernando Garzon Miguez, Garyfallia Paspalaki.
CMS Offline & Computing, CMS, CERN, Switzerland.

CMS Monitoring in Run 3

CMS Monitoring had significantly evolved its infrastructure in the *Long Shutdown* 2 to face the challenge of Run 3 data taking. Main improvements had been focused on:

- Kubernetes on infrastructure management,
- Open-source products usage on almost all service needs,
- Adding new instruments and developing new critical data pipelines for collaboration's needs.

Figure 1: Software stack

With the solid capabilities of CERN IT services, which CMS Monitoring infrastructure deployed on, new gadgeteries were added to our toolbox and the existing ones are polished.

Infrastructure Overview

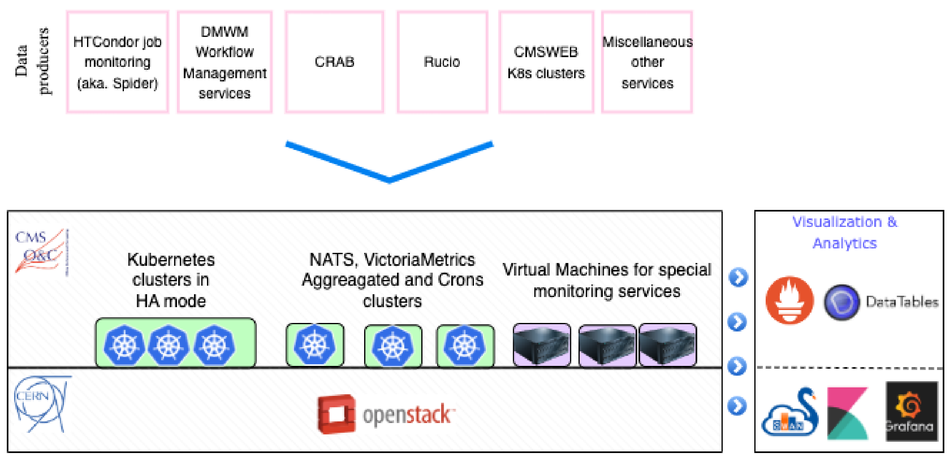

Figure 2: Infrastructure overview

- **Kubernetes**: 6 clusters with ∼100 of services
- **Prometheus**: 3 instances running in HA mode with total ∼100k metrics.
- **Victoria Metrics**: 3 instances in HA, 2 custom instances with total ∼900 billion data points
- **Apache Spark**: Number of spark jobs run on up to PB size data with daily/weekly/monthly schedule
- **Apache Sqoop**: Dumps of tens of database tables each day
- **Grafana**: ∼100 production dashboards with thousands of plots

High Availability

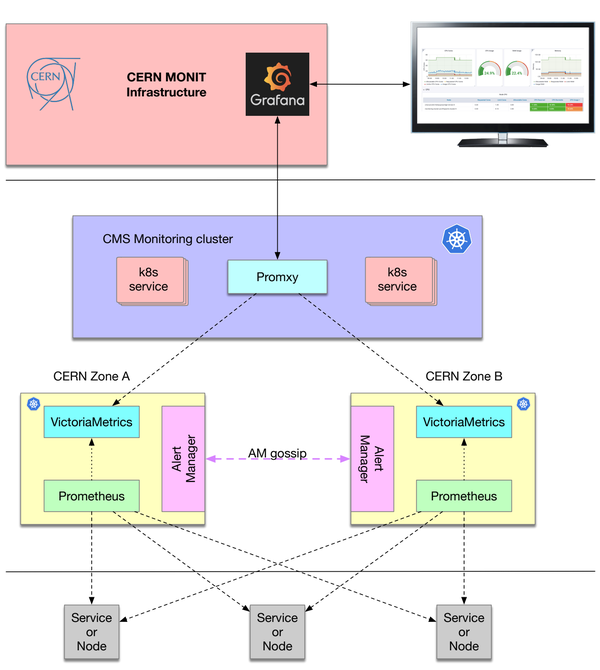

Figure 3: High-Availability.

High availability refers to the ability of a system to minimize downtime in order to avoid service interruptions. Currently 3 instances of each Prometheus, AM and VM services which talk each other enables HA mode. A *Promxy* service in main Kubernetes cluster plays a role of a single Grafana data-source for all Prometheus and VM instances.

Data Management Monitoring

Recent transition from *Phedex* to *Rucio* in data management of CMS brought new improvements to the collaboration and new responsibilities to our shoulders. The cooperation with data transfer team on monitoring of it had widened the data management monitoring view in a great extend.

Data management monitoring mainly uses Rucio and DBS(Dataset Bookkeeping Service) services data which are stored in OracleDB. Because of the limitations of relational databases, these tables are dumped to HDFS and all aggregations are handled by Spark jobs.

Datasets Monitoring - Time Series

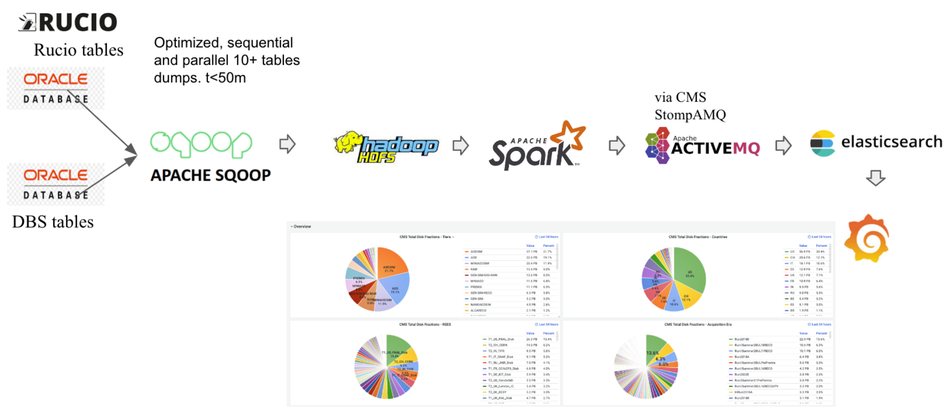

Figure 4: CMS Rucio and DBS(Dataset Bookkeeping Service) datasets monitoring using ElasticSearch storage with 3 months of retention time which will be increased to 3 years.

Complexity is in its nature. A spark job runs in Kubernetes joins Rucio and DBS(Dataset Bookkeeping Service) tables'(10+) dumps and handles more than 30 group-by.

Datasets Monitoring - Advanced Table View

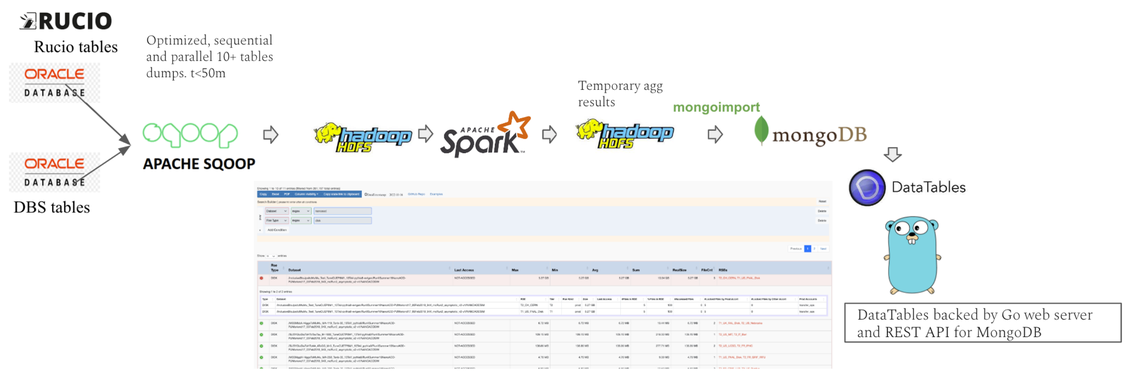

Figure 5: CMS Rucio and DBS datasets monitoring in MongoDB and DataTables

Because of the limitations of Grafana/Kibana, CMS Monitoring serves DataTables page for CMS Rucio-DBS datasets monitoring with easy-to-use search functionalities. Fast MongoDB collection management, fast&cool GoLang web server, which hosts REST API to MongoDB and DataTables, entails access to datasets information in all sites with daily granularity.

Summary and Future Works

- CMS Monitoring will improve its infrastructure by migrating rest of the services running in virtual machines to Kubernetes and by automatizing service deployments by implementing a CI/CD pipeline.
- OracleDB table dumps in HDFS opened a new era in Rucio and DBS data management monitoring. It will continue to evolve with the implementation of other critical monitoring needs. In the meantime, we are in search of close real-time monitoring of CMS Rucio-DBS data management.