

MONIT: Monitoring the CERN Data Centres and the WLCG Infrastructure

Alberto Aimar^{1*}, *Asier Aguado Corman*², *Pedro Andrade*¹, *Javier Delgado Fernandez*¹, *Borja Garrido Bear*¹, *Edward Karavakis*¹, *Dominik Marek Kulikowski*³ and *Luca Magnoni*¹

¹ CERN, European Laboratory for Particle Physics, Geneva, Switzerland

² Universidad de Oviedo, Oviedo, Spain

³ Wroclaw University of Science and Technology, Wroclaw, Poland

Abstract. The new unified monitoring architecture (MONIT) for the CERN Data Centres and for the WLCG Infrastructure is based on established open source technologies to collect, stream, store and access monitoring data. The previous solutions, based on in-house development and commercial software, have been replaced with widely- recognized technologies such as Collectd, Kafka, Spark, Elasticsearch, InfluxDB, Grafana and others. The monitoring infrastructure, fully based on CERN cloud resources, covers the whole workflow of the monitoring data: from collecting and validating metrics and logs to making them available for dashboards, reports and alarms. The deployment in production of this new DC and WLCG monitoring is well under way and this contribution provides a summary of the progress, hurdles met and lessons learned in using these open source technologies. It also focuses on the choices made to achieve the required levels of stability, scalability and performance of the MONIT monitoring service.

1 Introduction

The CERN Data Centres (DC) and the Worldwide LHC Computing Grid (WLCG) [1] have been monitored, for more than a decade, with in-house central solutions gathering and storing in the CERN storage facilities a large amount of metrics and logging information. Both monitoring systems have been developed separately, supported by different software teams at CERN and with a limited amount of common software components.

The monitoring of the CERN Data Centres, in Geneva and in Wigner, has been using LEMON (LHC Era Monitoring for Large-Scale Infrastructure) [2] a client/server-based monitoring system developed in the Laboratory. LEMON has been gathering, transporting and storing the information collected on all DC computers, IT services and other entities like network switches, electrical power distribution systems and environmental sensors (temperature, dust, etc.). On every monitored node, every few minutes, a local agent retrieves

* Corresponding author: alberto.aimar@cern.ch

samples of monitoring information and detects possible anomalies and alarms. Such information, typically metrics and status of the services running on that node, is forwarded to a central repository where data are curated, stored and displayed in quasi real time.

In the same period, the Large Hadron Collider (LHC) experiments have relied on an advanced and dedicated toolset, WLCG Monitoring [3], for collecting and monitoring information about the whole LHC Grid infrastructure. The WLCG Monitoring has grown to include a number of specialized applications developed at CERN, widely used in the experiments, that enabled the central monitoring of several grid and experiments activities, including execution of jobs, data transfers, and site availability, over the more than 250 sites of the WLCG.

Starting from 2013, both LEMON and WLCG Monitoring, initially based on relational database technologies, had both started to evolve towards more recent data analytics technologies such as Elasticsearch, Apache Hadoop and Apache Spark. Therefore, at the end of 2015, it was decided to take advantage of these major refactoring activities of the two monitoring services and, after merging the two CERN teams, aiming at a new single Unified Monitoring service satisfying both Data Centres and WLCG Infrastructure requirements.

2 Unified Monitoring Services (MONIT)

2.1 Architecture and Open Source Technologies

While the previous monitoring solutions had been improved and extended for more than a decade, several new technologies have been developed for data collection, storage, transport and processing. Many of these new technologies are open source and were developed by highly successful major software vendors and communities to gather, store and distribute information in highly scalable and distributed infrastructure.

The new unified architecture [4], designed by the engineers of the new team, is shown in Figure 1 and is based on a standard modular approach, with specific tasks on the dataflow of the monitoring information:

- data sources: to collect and validate the monitoring data sets from a variety of producers;
- transport: to gather data sets from the original sources to the storage components, and also provide a large buffer in case of downtime of any of the downstream storage layers;
- processing: to perform data processing on real-time data or on historical data;
- storage: to store the data, offering different retention policies according to the type of the data (raw metrics, logs, stats, etc.) and based on different technologies;
- data access: to offer to users many visualizations, reporting and analytics technologies.

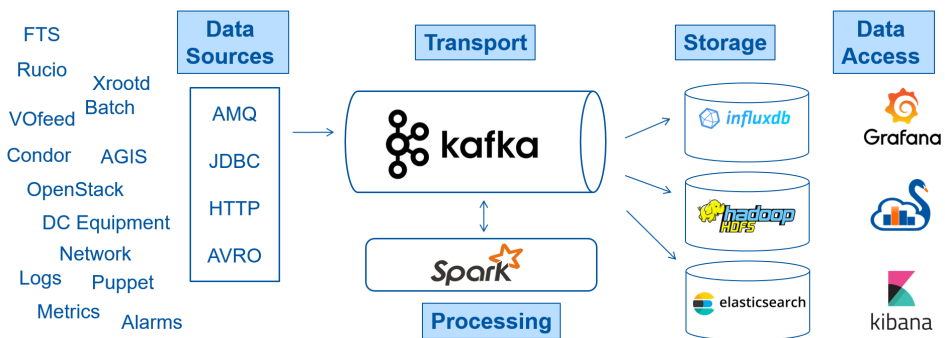


Fig. 1. MONIT Architecture and Technologies

In addition to design and implement a single unified architecture, the other key decision was to exclude any in-house development and to base the new monitoring infrastructure only on mainstream open source technologies. These technological solutions chosen are those already widely used by providers of internet services and main social networks for collecting and transporting, in a scalable manner, much larger amounts of data.

2.2 Data Sources

Data are sent to MONIT [5] in a schema-free JSON format, but must have a set of predefined fields (e.g. producer, timestamp, etc.) in order to be routed, validated, processed and stored adequately. The MONIT infrastructure provides a set of standard ways to integrate data from different protocols and technologies, both from the Data Centres and the WLCG Grid resources.

Adding new data sources to the MONIT system is documented and there are standard options that best cover different types of data and producers:

- Collectd [6]: is the new local agent sending metrics and alarms that replaces the LEMON agent on every node of the Data Centres. Collectd has a wide set of plugins for all metrics for all kinds of hardware, operating systems and common software services;
- HTTP feed and ActiveMQ [7] messaging: for external data producers that produce and push monitoring information to the central monitoring;
- JDBC query: to periodically pull external data from a variety of existing databases of monitoring information.

2.2 Transport and Processing

All data received via the MONIT Data Sources are routed to an Apache Kafka cluster [8] that provides a unified, high-throughput and low-latency transport platform. Data are published in different producer-specific topics that are partitioned, replicated and distributed throughout the Kafka cluster. The data in Kafka is distributed in real time (time of flight from source to destination is below 10 seconds) towards the storage component and buffered for a retention time of 72 hours in order to prevent a data loss in any downtime of the data consumers. This approach, based on Kafka as transport layer which decouples producers and consumers of the monitoring data, enable full isolation of workflows and protect users from being affected by each other activities and by infrastructure problems.

MONIT allows data handling via Apache Spark [9] both in real time stream processing of the data from Kafka and batch processing of historical data from HDFS[10]. Stream processing is used to serve multiple needs and operations on the monitoring data:

- data enrichment when a data source is joined and enriched with information from several other sources (e.g. the WLCG topology, DC node information);
- data aggregation over time such as the creation of summary statistics over a time bin, or over other dimensions, i.e. the computation of a cumulative metric for a set of machines hosting the same service;
- data correlation: to detect simultaneous anomalies, perform advanced alarming to detect failures originating from multiple sources, i.e. DC topology-aware alarms.

Batch processing is used to reprocess and recalculate historical data, compress old historical data and to extract statistics and reports used by management bodies at CERN and WLCG.

The processing platform, also available to teams and users, relies on a *Mesos cluster* [11] with *Marathon* and *Chronos* for the job orchestration and scheduling, and on *Docker* [12] for an isolated environment and a lightweight deployment of the Spark processing jobs.

2.3 Storage

All data received and transported by MONIT (3 TB per day in compressed) are stored in different storage endpoints in format and granularity for different types of display and access:

- HDFS to store all data for long-term data archival and offline analytics. Compressed JSON or Parquet for metrics and logs;
- Elasticsearch [13] (ES) for short-term storage and indexing of the data three large instances. Raw data, metrics and logs, are kept for one month;
- InfluxDB [14] for the medium and long-term storage of time-series data either in a raw or in an aggregated format. Data are being kept for 5 years and are down-sampled and aggregated (raw data for 1 week, 5m bins for 1 month, 1h bins for 5 years).

2.4 Data Access

The monitoring data in MONIT can be accessed using well-known visualization and data analytics technologies:

- Kibana [15] is provided for the visualization of metrics and logs, connected to the Elasticsearch backends. This offers search/filtering capabilities, and to interactively discover monitoring data and logs;
- Grafana [16] is for the visualization of time-series data serving data for both Elasticsearch and InfluxDB storage. Grafana supports multiple back ends: ES/InfluxDB and provides templates, ad-hoc filters, advanced queries, ACLs and high-level alarms;
- Swan [17], a CERN project that is based on Jupyter [18], gives users access to data stored in HDFS, Elasticsearch and the integration with multiple widely used High Energy Physics (HEP) tools to create interactive notebooks.

3 Monitoring the CERN Data Centres

The MONIT service in 2018 is progressively replacing the old metrics collection agent based on LEMON. MONIT provides a standard and solid dataflow where collectd, installed on all nodes, sends all the metrics gathered to the central infrastructure. Additionally it was possible to integrate use cases previously not achievable, such as the central collection of IT Services Logs, which been successfully adopted by many service managers and is now generating almost 1 TB data/day.

The MONIT service provides by default:

- automated deployment: all monitoring agents are configured, via Puppet [19], and deployed on all nodes (virtual machines, hypervisors) and equipment of the CERN DC;
- preconfigured dataflow: data collected transported and stored using MONIT services;
- generic dashboards: for host and services metrics that can be tailored for different views and services;
- standard alarms: a set of alarms that can be used and customized by Service Managers.

The IT service managers can then add additional metrics, dashboards and alarms to use the standard collectd data available but also extending it with service-specific features:

- service metrics: configuring or developing specific collectd plugins;
- service dashboards: to provide real-time and trends of main indicators of their service;
- services alarms: to create custom alarms for their services, in collectd or Grafana.

Currently, all main CERN IT Services are moving to the MONIT infrastructure. Some examples of dashboards that they have developed are in Figures 2 to 4 below.

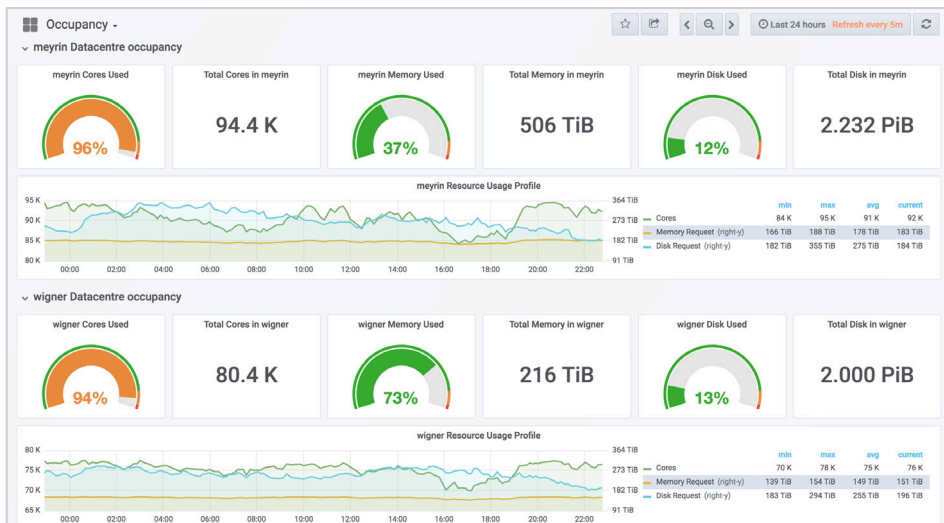


Fig. 2. Example of Dashboard of the Batch Computing Service

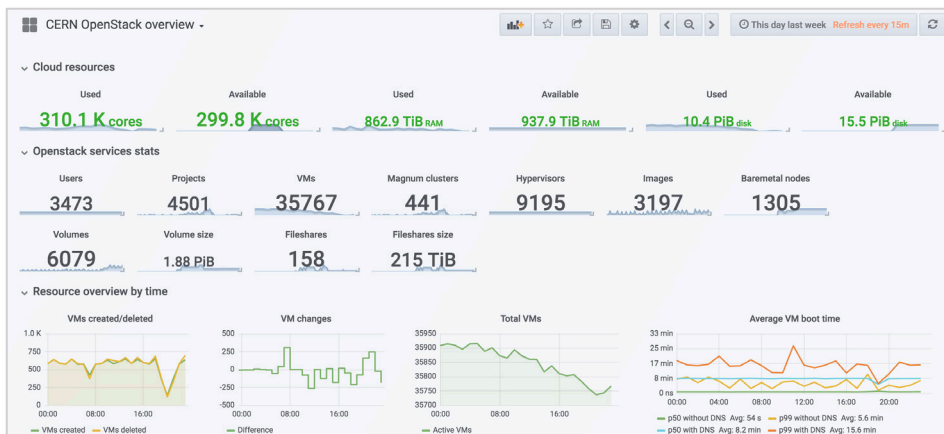


Fig. 3. Example of Dashboard of the Cloud OpenStack Service



Fig. 4. Example of Dashboard of EOS Storage Services

4 Monitoring the WLCG Infrastructure

The second major target of the new Unified Monitoring is the replacement of WLCG monitoring. The WLCG data is collected, from the existing data sources and channeled through the standard MONIT workflow (Figure.1), based on Kafka. Spark jobs provide enrichment and aggregation to the data (consistent naming, add topology, create hourly bins, etc.). Once the data is stored, the users and experts can develop new dashboards with the support of the MONIT team. Currently all previously collected data are being collected and dashboards are developed in MONIT:

- WLCG Transfers, FTS and XRootd: retirement of the all system completed ;
- ATLAS Distributed Data Management: new dashboards being used by shifters;
- ATLAS Job Accounting: new and more reliable data sources being completed;
- CMS Job Monitoring: new dashboards being finalized with the CMS experts;
- WLCG Sites Availability and Status Board: initial prototype being promoted to stable flow.

Figures 5 to 7 show examples of these new dashboards for WLCG and experiments in MONIT Grafana.

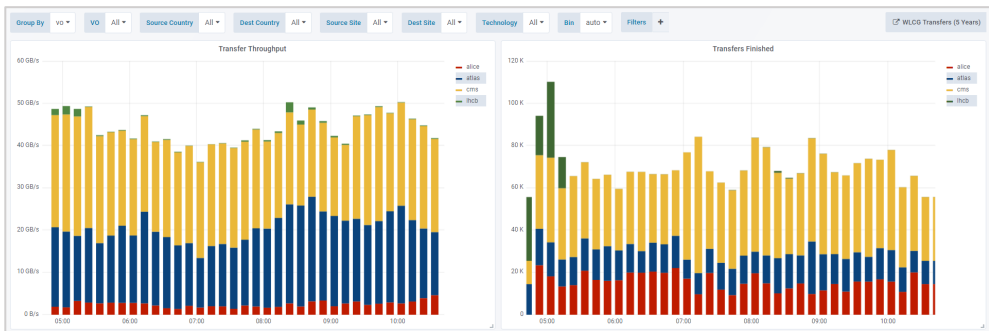


Fig. 5. Dashboard of the WLCG Transfers of each Experiment

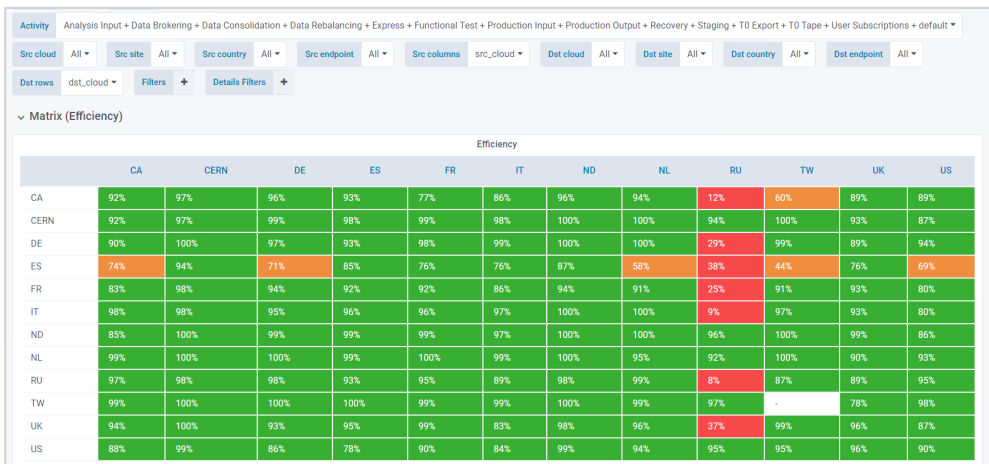


Fig. 6. Dashboard of the ATLAS DDM Transfers

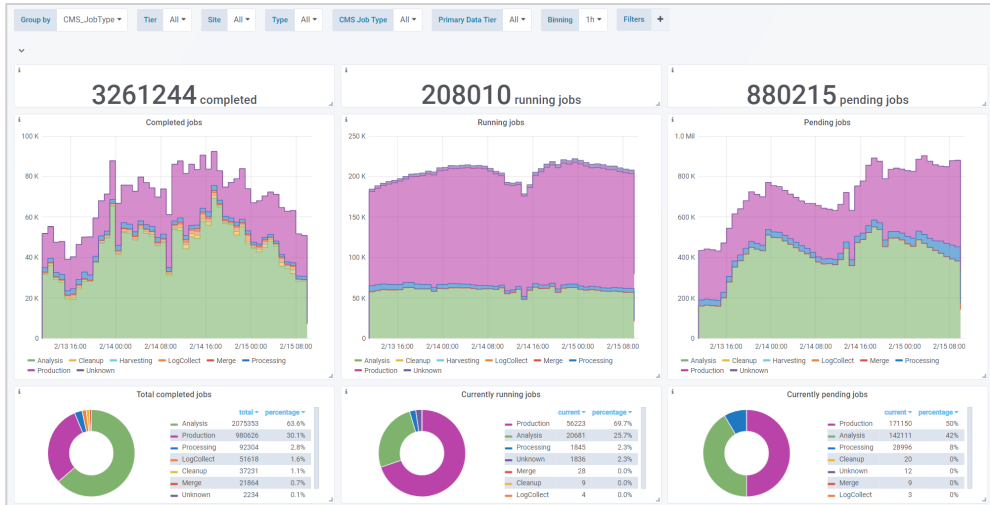


Fig. 7. Dashboard of the CMS Job Monitoring



Fig. 8 Dashboard of the WLCG Network Performance

5 Conclusions and Current Status

The new monitoring architecture is a solid, modular and highly-scalable infrastructure built on mainstream open-source technologies and tools, which proved capable of gathering, transporting processing 3TB/day and ready for the CERN DC growth in the coming years.

The main challenges introduced with the new Unified Monitoring infrastructure were the full adoption of established open source solutions and refraining from developing any in-house customization, that would imply any significant development and maintenance costs, but rather promote the contribution upstream to join the communities for development and maintenance efforts. The adoption of this “develop services by configuring and tuning external solutions” approach forces to follow the priorities of the open source projects but has also brought major upgrades with new versions of those external open source products, without any development costs for the MONIT team.

This change of mentality is also difficult for the users, both in IT services and Experiments, who were used to simply request, and obtain, ad-hoc improvements and features. Now they are encouraged to do them themselves using the open source tools provided (e.g. Grafana).

The open source products chosen for MONIT, such as Kafka and Spark, are respected and popular, in the community and outside. This makes it easier to hire and integrate new team members in an environment like CERN where interns and students join for very limited time.

Based on these products, the architecture designed is linear, modular and decoupling all key functions. This allows, for instance, having tens of types of data sources and several types of storage that best fit for the type of granularity and usage of the data: from slower long-term storage in HDFS to faster automated time series aggregations provided by InfluxDB.

Currently the new Unified Monitoring infrastructure is running in parallel to the old DC and WLCG monitoring solutions that will be progressively replaced in 2018 and 2019:

- Collectd is installed and sending metrics to MONIT for all the modes and most services. The option to turn-off LEMON is available for the IT services;
- Kafka is transporting 3 TB/day, more than 90Kdocs/s from about 100 data sources to the MONIT storage layer;
- Grafana provided by MONIT has more than 1500 registered users and is the standard display for monitoring the Data Centres, IT services and the WLCG infrastructure.

References

1. Bird I, *Computing for the Large Hadron Collider*, Annual Review of Nuclear and Particle Science, 61:99–118 doi:10.1146/annurev-nucl-102010-130059 (2011)
2. Babik M et al, *LEMON - LHC Era Monitoring for Large-Scale Infrastructures*, J. Phys.: Conf. Ser. 331 052025 doi:10.1088/1742-6596/331/5/052025 (2011)
3. Andreeva J et al, *Experiment Dashboard for monitoring computing activities of the LHC virtual organizations*, J. Grid Comput. 8 323-339 doi:10.1007/s10723-010-9148-x (2010)
4. Aimar A et al, *Unified Monitoring Architecture for IT and Grid Services*, Journal of Physics: Conference Series. <http://stacks.iop.org/1742-6596/898/i=9/a=092033> (2017)
5. MONIT Portal, <http://monit.web.cern.ch/monit>
6. Collectd, <https://collectd.org>
7. Apache ActiveMQ, <http://activemq.apache.org>
8. Apache Kafka, <http://kafka.apache.org>
9. Apache Spark, <http://spark.apache.org>
10. Apache Hadoop, <http://hadoop.apache.org>
11. Apache Mesos, <http://mesos.apache.org>
12. Docker, <http://docker.com>
13. Elasticsearch, <http://elastic.co>
14. InfluxDB, <https://www.influxdata.com/time-series-platform/influxdb/>
15. Kibana, <https://www.elastic.co/products/kibana>
16. Grafana, <http://grafana.org>
17. Piparo D et al, *SWAN: A service for interactive analysis in the cloud*, Future Generation Computer Systems doi: 10.1016/j.future.2016.11.035 (2016)
18. Jupyter, <https://jupyter.org>
19. Puppet at CERN, <https://puppet.com/presentations/puppet-infrastructure-cern>