

Monitoring System of the AMS Science Operation Centre

Wenchao Zhang¹, Baosong Shan^{1,*}, Vitali Choutko², Alexander Egorov², and Alexandre Eline²

¹Beihang University, School of Mathematics and System Science, Beijing 100191, China

²Massachusetts Institute of Technology, Laboratory for Nuclear Science, MA-02139, United States

Abstract. The Alpha Magnetic Spectrometer (AMS) is a high energy physics experiment installed and operating on board of the International Space Station (ISS) from May 2011 and expected to last through year 2024 and beyond. The Science Operation Centre (SOC) is in charge of the offline computing for the AMS experiment, including flight data production, Monte-Carlo simulation, data management, data backup, etc. This paper introduces the design and implementation for the new monitoring system of AMS SOC, from the monitoring data collection at the backend, to the visualisation at the frontend.

1 Introduction of the AMS Science Operation Centre

The Science Operation Centre (SOC) is responsible for science data processing for the Alpha Magnetic Spectrometer [1] (AMS) experiment, acting as a bridge between detector operations and physics analysis.

1.1 Flight data processing

The most important responsibility of SOC is to process the scientific data collected by the AMS detector, for detector monitoring, calibration, alignment, and for the physics analysis as well. The data processing flow will be described in detail in Section 2.

1.2 Monte-Carlo production

Monte-Carlo [2] (MC) is a simulation method, widely used in high-energy physics for detector design, optimisation, interaction simulation, and physics analysis. In AMS, MC production program reads simulation parameters and produces both the RAW and ROOT [3] files for data analysis. Since MC simulation is quite CPU consuming, AMS has several computing centres to run MC jobs. SOC manages the MC production at CERN (running on LSF [4] and HTCondor [5] batch farms), and the validation of the MC files produced by all the computing centres.

*e-mail: baosong.shan@cern.ch

1.3 SOC shifts and monitoring tools

To make sure data production is running correctly, shifts are arranged to monitor the above mentioned SOC activities and also the facilities for them, and various tools were developed to help the shift takers to check the status and to notify experts in case of issues.

As introduced in [6], SOC developed some tools to monitor mainly the first pass data production. These include the *frame monitor* for frame data arriving delay, the run list for the meta data display of all the science RAW files and their reconstructed ROOT files, the *production monitor* for the active data reconstruction processes, and the *net monitor* for monitoring the SOC hosts, network, and storage. For MC production, a web site is designed to show the dataset completing status and estimation.

2 Overview of the AMS data flow

Figure 1 shows the data transferring path of AMS. The data collected by the AMS detector are transferred by the Tracking and Data Relay Satellites (TDRS) from the ISS to White Sands, and then transmitted to the Payload Operations Control Centre (POCC) at CERN by the ground system. The original data, also known as *frame data* because it is in the form of one-minute frame files, arrives at POCC nearly in real-time, and POCC shares the data to SOC via Network File System [7] (NFS).

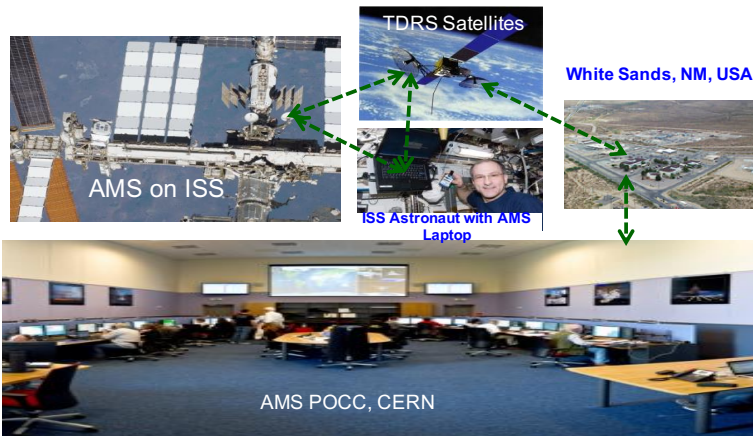


Figure 1. Data collected by AMS are transferred by the TDRS to White Sands, NM, USA, and then transmitted to AMS POCC at CERN by the ground transmission system.

SOC starts its flight data processing from reading the frame data.

2.1 Frame decoding

To facilitate the detector calibrations and alignments, we divide the ISS orbit into four segments, each one corresponding to the part between either pole and the equator, and the data of each segment is organised into one *run*, about 23 minutes. Data from the same run shares the same run ID and is decoded from the corresponding frame files, and repacked into one RAW file. Each RAW file is copied to EOS [8] after the consistency check, and the metadata is inserted into the Oracle Parallel Database [9] (PDB) hosted at CERN IT.

2.2 Reconstruction

After the metadata for the RAW files is recorded in the database, production jobs can be requested. A web server with CGI scripts is responsible to process the job requesting, and job requests are carried out by a cron script from a Puppet [10] managed host. Jobs are submitted to run on SOC computing facilities, as well as on CERN LSF and HTCondor batch farms. Production jobs produce ROOT files and Time Dependent Variable (TDV) files for the AMS conditional database. ROOT files are validated and uploaded to EOS, and the metadata are also recorded in the PDB.

The above mentioned production is the first pass of the flight data reconstruction which runs in a fully automated manner, the main purpose is for quick sub-detector performance evaluations. Usually the reconstructed data are available in a few hours after the raw data arrives. For physics analysis, a second pass of flight data reconstruction is required, which uses the calibrations, alignments and ancillary data from the ISS, as well as monitoring values (temperatures, pressures, and voltages), to produce final data ready for analysis. The second production usually runs every 6 months incrementally. In case of major upgrades of the reconstruction software, a full reproduction is needed. Figure 2 shows the data flow of the two-pass production.

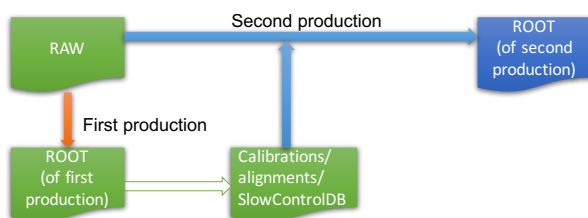


Figure 2. Data flow of the first and second production.

3 The redesign of AMS SOC monitoring system

Existing monitoring tools were designed for experts to monitor/manage the production, but lack of some important features, like:

- Modern monitoring visualisation interface;
- History records and data trends;
- Customisable monitoring time range;
- A glance view of production status;
- Alerts/notifications with configurable thresholds.

And as we are increasing our use of the resources/services provided by CERN, monitoring of those resources/services is also required. For example, the availability and quota status of AFS [11] and EOS, the status of the usage of LSF and HTCondor, and the activities on CASTOR [12] and FTS3 [13].

At the same time, the CERN IT monitoring infrastructure, MONIT [14], provides various tools to manage the monitoring metrics, to record in time based databases, and to virtualise them. Therefore, we decided to redesign the SOC monitoring system by using the MONIT infrastructure.

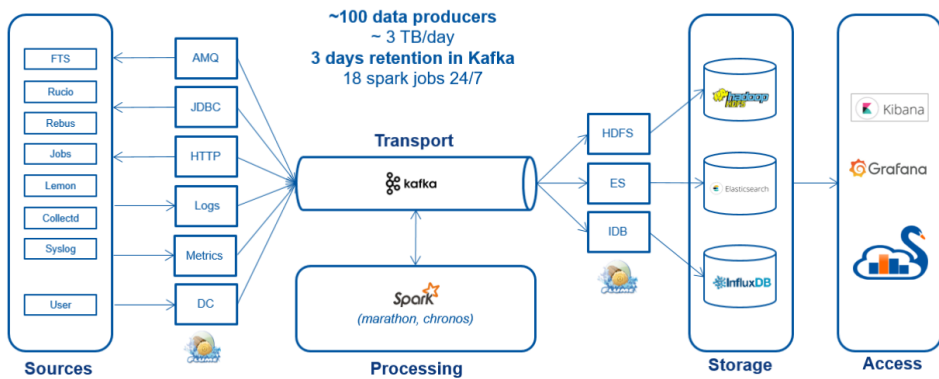


Figure 3. MONIT is “based on several widely-used open source technologies, selected tools are loosely coupled to each other adding flexibility to the system”. [14]

Figure 3 [14] shows the data flow of MONIT, which provides all the necessary tools for data storage, transportation, and visualisation. For most of the services/resources hosted by CERN IT, we only need to implement the visualisation part, based on the tools provided. For the others, we have to implement the data source part as well.

Taking a closer look at the data sources part, we can classify our monitoring targets and choose the corresponding sources for them.

3.1 SOC monitoring targets

As mentioned in Section 1, SOC is in charge of flight data processing, MC production, and the services/resources required for them. The following metrics are the most important targets for SOC monitoring.

3.1.1 Delay metrics of the first pass data production

The first pass data production is mainly for detector monitoring, so it must be done at the least possible delay. To describe the health status of it, we choose the following metrics:

Frame arriving delay: the time till now when the last frame file arrives. A large delay usually indicates a long time without data acquisition or that the transferring link is broken somewhere.

Frame delay: the time till now when the last frame file is generated by the detector. A large delay of this, with nominal delay of frame arriving delay, indicates the insufficient transferring speed.

RAW delay: the time till now when the last RAW file is validated. A large delay of this, with nominal delays of the above frame delays, indicates the issue with the program *frame decoder*, transmitter, or validator.

ROOT delay: the time till now when the last ROOT file is validated. A large delay of this, with nominal delays of the above frame and RAW delays, indicates the issue with reconstruction processes or the ROOT transmitter/validator.

Fresh run delay: the time till now when the most recent run is validated. A large delay of this, with nominal delays of all the above delays, indicates the issue with the production job requesting.

Apparently, these delay metrics belong to the “User Data” part in Figure 3, which means we have to prepare these metrics, send, and store them in a desired storage. These metrics describe the different delay values in a time sequence, so InfluxDB [15] is chosen as the storage.

We implemented a script to run routinely and get the delay values, pack them into JSON [16] format, and send to the MONIT metrics server by HTTP request. The MONIT server will take care of transferring and storing the data in the InfluxDB.

3.1.2 MC production metrics

As the AMS analysis is going to higher atomic number (Z) elements, the amount and importance of MC simulation are going higher as well. The memory consumption for high-Z MC, especially at high energy ranges, is increasing dramatically, and we observe more MC job abnormal due to these factors. To have a better understanding on the resource requirements of MC jobs, we decide to dig more information on MC jobs and add it to the monitoring system.

Similar to production delay, the MC production metrics are also “User Data” and needed to be prepared, sent to MONIT, and stored in InfluxDB. The following metrics are selected to describe the concerns of MC production.

Total trigger number: The total number of triggers processed during the specific period.

Completion ratio: The number of processed triggers over the preset.

CPU time: The total CPU time consumed during the period.

Average wall time: The average running time of MC jobs.

Average threads: The total CPU time divided by the running time.

All the above metrics are grouped by each site and each MC simulation template.

3.1.3 Batch metrics

CERN IT provides monitoring data for HTCondor batch system, however, it does not include everything we concern. For example, AMS has dedicated computing resource on HTCondor and we want to make sure that the resource is not idle (if idle, we will run MC jobs on it). And LSF batch platform does not have any monitoring data embedded in MONIT. So we also build our own “User Data” for batch platforms: LSF, HTCondor dedicated resource, and HTCondor shared resource.

Total slots: The total number of slots on the batch platform.

Busy slots: The number of busy slots.

Production slots: The number of busy slots which are used by production jobs.

Idle slots: The number of idle slots.

Pending jobs: The number of jobs which is waiting to be executed.

Pending production jobs: The number of production jobs which are pending.

3.1.4 Storage metrics

Different types of data are stored in different storages, and the storage resources need to be monitored.

EOS is the main storage of AMS science data and physics analysis data, powered by CERN EOS service; the NFS storage is the disk storage for both science data and user data, managed by SOC, accessible on SOC hosts by NFS protocol; CASTOR is the backup storage for AMS science data, powered by CERN CASTOR service; and AFS, which is expected to be replaced by EOS, is the main storage of AMS TDV database, codes, executables, libraries, job scripts, log files, user data, etc., powered by CERN AFS service.

For EOS and CASTOR, CERN IT provides the monitoring data, and we only need to do visualisation. For AFS, there is no monitoring data in MONIT so we build our own monitoring metrics on the volume quota, read/write latency, and directory sizes. For NFS, the availability on different host, the quota, and the read/write latency is monitored.

3.1.5 Host metrics

SOC has more than 40 hosts doing dedicated tasks, such as production management, web server hosting, job requesting, monitoring, frame decoding, validating, transferring, production executing, etc. Nearly half of them are managed by Puppet configuration service, and the number is growing over time. To monitor the health status of the Puppet managed hosts, MONIT provides “collectd” sensors to gather the information, like CPU usages, I/O, memory, disk space, etc. For the hosts which are running standalone, we keep use the “net monitor” to report their anomalies.

3.2 Visualisation

MONIT provides Kibana [17] and Grafana [18] as the visualisation tools. Since most of the monitoring targets are metrics, we choose Grafana as our platform.

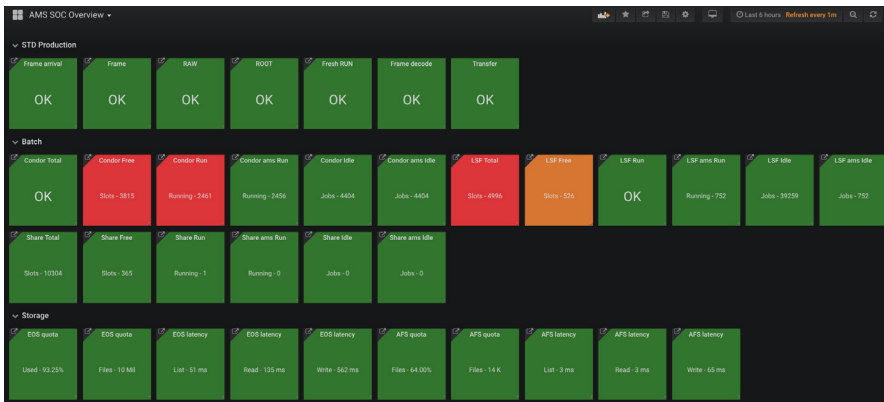


Figure 4. SOC Grafana graph: overview.

The monitoring tools we discussed in Sector 1.3 provide many detailed information to the shift takers/monitors, however, it would take quite some time for a person to examine each of them and confirm if the overall status is nominal or not. To overcome this, we redesigned the

visualisation style for the SOC monitoring system, by abstracting a glance view interface to indicate the overall service status, and linking the detailed monitoring curves to them.

As shown in Figure 4, the glance view page gives a clear view of the SOC status by the “singlestat” icons with different colors, green for OK, yellow for warning and red for critical. Usually there is a linked URL with each icon, and by clicking the user can be redirected to the detailed graphs.

For example, if one clicks on any of the icons related to production, the detailed curve for production delays will show up, as shown in Figure 5. And Figure 6 shows the detailed curve for HTCondor status, where the green line indicates the running slots, and all others show pending jobs for different users (legend part not included for privacy reason).

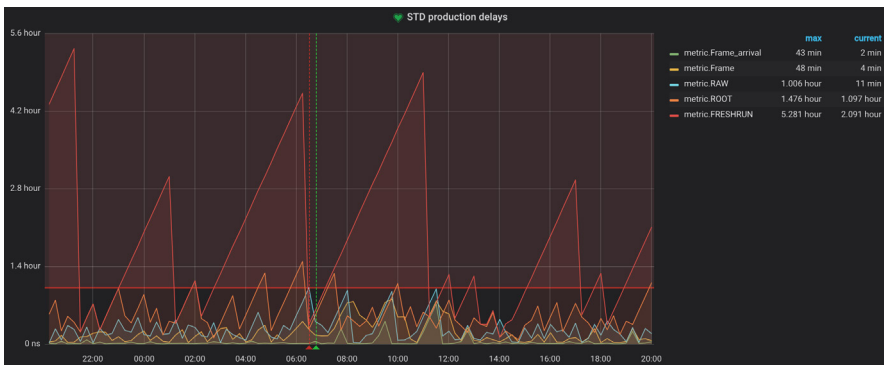


Figure 5. SOC Grafana graph: production delays.

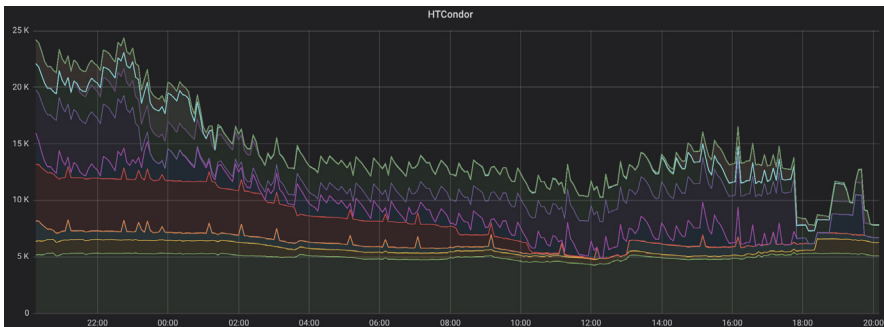


Figure 6. SOC Grafana graph: HTCondor status.

3.3 Notification and alerting

Grafana provides automatic notifications. Once the thresholds are defined in a graph, notifications with a snapshot will be sent by email when the monitored values go beyond the thresholds. Similarly, another mail will be sent when the values go back to nominal ranges.

There is also a special type of Grafana panel element called “Alert List”, where all the monitored graph will be listed with their current status (OK or Alerting).

4 Conclusion

The AMS SOC monitoring system is redesigned based on MONIT, the monitoring infrastructure of CERN IT, to provide modern monitoring interface for better checking and reporting the status of SOC activities and resources. The glance view of SOC status provides a tidy and clear view to show the overall states of the activities and resources, with links to the time-based monitoring curves, which provide the experts more detailed information for further diagnostics.

In future, we will add more monitoring targets as needed, and explore the application of machine learning on issue analysis.

References

- [1] S. Ting, Nuclear Physics B-Proceedings Supplements **243**, 12 (2013)
- [2] C.Z. Mooney, *Monte carlo simulation*, Vol. 116 (Sage Publications, 1997)
- [3] I. Antcheva, M. Ballintijn, B. Bellenot, M. Biskup, R. Brun, N. Buncic, P. Canal, D. Casadei, O. Couet, V. Fine et al., Computer Physics Communications **182**, 1384 (2011)
- [4] P. Computing, *Load sharing facility, lsf* (2007)
- [5] D. Thain, T. Tannenbaum, M. Livny, Concurrency and computation: practice and experience **17**, 323 (2005)
- [6] V. Choutko, O. Demakov, A. Egorov, A. Eline, B. Shan, R. Shi, *Evolution of Monitoring System for AMS Science Operation Centre*, in *Journal of Physics: Conference Series* (IOP Publishing, 2017), Vol. 898, p. 092053
- [7] S. Shepler, M. Eisler, D. Robinson, B. Callaghan, R. Thurlow, D. Noveck, C. Beame, Network (2003)
- [8] A.J. Peters, L. Janyst, *Exabyte scale storage at CERN*, in *Journal of Physics: Conference Series* (IOP Publishing, 2011), Vol. 331, p. 052015
- [9] D. DeWitt, J. Gray, Communications of the ACM **35**, 85 (1992)
- [10] J. Loope, *Managing Infrastructure with Puppet: Configuration Management at Scale* ("O'Reilly Media, Inc.", 2011)
- [11] J.H. Howard et al., *An overview of the andrew file system* (Carnegie Mellon University, Information Technology Center, 1988)
- [12] G.L. Presti, O. Barring, A. Earl, R.M.G. Rioja, S. Ponce, G. Taurelli, D. Waldron, M.C. Dos Santos, *CASTOR: A Distributed Storage Resource Facility for High Performance Data Processing at CERN.*, in *MSST* (Citeseer, 2007), Vol. 7, pp. 275–280
- [13] A. Ayllon, M. Salichos, M. Simon, O. Keeble, *Fts3: New data movement service for wlcg*, in *Journal of Physics: Conference Series* (IOP Publishing, 2014), Vol. 513, p. 032081
- [14] *Monitoring tutorial: Agile infrastructure and puppet for service managers*, <http://monit-docs.web.cern.ch/monit-docs/training/index.html>
- [15] I.O.S.T. Series, Website [Httpinfluxdb Com](http://influxdb.com) (2015)
- [16] D. Crockford, Tech. rep. (2006)
- [17] Y. Gupta, *Kibana Essentials* (Packt Publishing Ltd, 2015)
- [18] *Grafana*, <http://grafana.org/>