Service monitoring system for JINR Tier-1

I. Kadochnikov, I. Kashunin, V. Korenkov, V. Mitsyn, I. Pelevanyuk, T. Strizh
Joint Institute for Nuclear Research, Dubna, Russia

Abstract

Tier-1 for CMS was created in JINR in 2015[1]. It is important to keep an eye on the Tier-1 center all the time in order to maintain its performance. The hardware monitoring system is based on Nagios: it monitors the center on the several levels: engineering infrastructure, network and hardware. It collects many metrics, creates plots and determines some statuses like HDD state, temperatures, loads and many other.

Apart from the infrastructure monitoring there is a need for consolidated service monitoring. The top-level services that accept jobs and data from the Grid depend on lower-level storage and processing facilities that themselves rely on the underlying infrastructure. There are variety of sources of information about the state and activity of the Tier-1 services.

The decision was made to develop a new monitoring system. The goals are to retrieve a monitoring information about services from various sources, to process the data into events and statuses, and to react according to a set of rules, e.g. to notify service administrators or to restart a service.[2]

Sources of information about Services

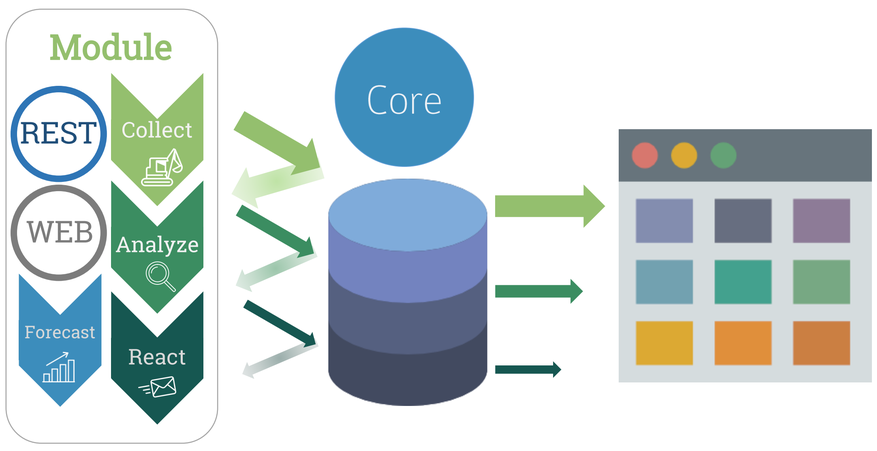
There are many sources of information about services running on a particular grid site. Now we can collect data from four types of sources:

JSON data | HTML data | Databases | Command Line

Without appropriate monitoring system administrator should manually check around 15 web-pages and run several commands in Linux Command Line at intervals between 15-minutes and 1 hour. And also, data in our sources are rarely presented in form which allow easy comprehension.

Architecture

The variety of monitoring information sources requires individual approach to every information source. That imposes modular style to our architecture, where for one source of data we have one module which is responsible for retrieving, storage, analysis, reaction and visualization.



With this architecture it is possible to focus on modules: add and change them without affecting the whole system. This architecture also allow to implement modules which do not need to use any part of module, for example: it is possible to make a module which has only web and rest part, so it does not collect anything, but it can display on the web page data from monitoring database.

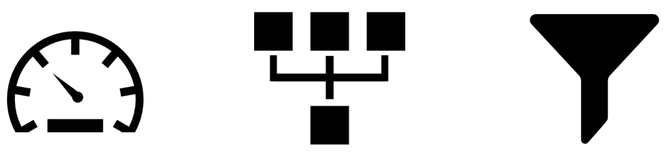
python | django | ANGULARJS by Google | Travis CI | PostgreSQL | JavaScript

For more information please visit:
https://github.com/tier-one-monitoring

Requirements

In order to simplify Tier-1 maintenance we need the monitoring system which would help administrators and shifters from control room. [2]

1. Shifters need to look at one page with all relevant information aggregated in summary - dashboard. In case of any service misbehavior there should be a possibility to get more detailed information about the service. If reason of a problem is not clear, then address the case to the administrator of the service.



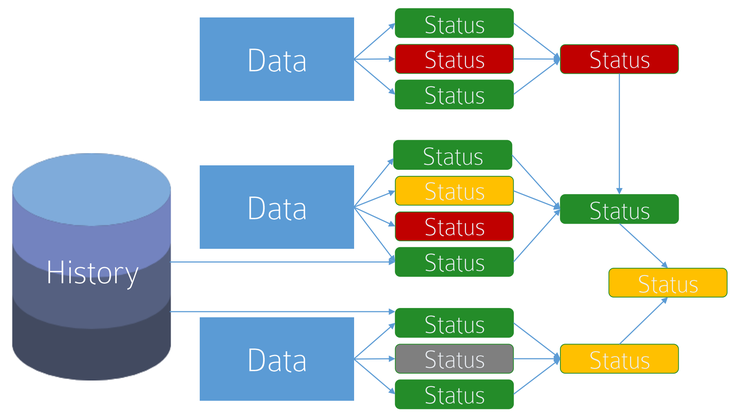
2. Administrator of a service should check historical data from time to time. But if some parameters goes out of configured boundaries administrator need to get and e-mail or in case of serious problem get an SMS message with notification and call to take an action. If necessary, monitoring system should provide tools for administrator to help him identify problem and fix it.

Events and statuses

Most of the time we receive raw data for our monitoring. This data by itself are only practical for system expert. After analysis they could be transformed to statuses and events.

Status of a service may consist of several minor statuses. Status consists of name, time of change, small description and value. Value is a number from 0 to 50. This numbers have mappings to state: Excellent, Good, Normal, Bad, Critical, Undefined. There could be more and it is up to module developer to add more.

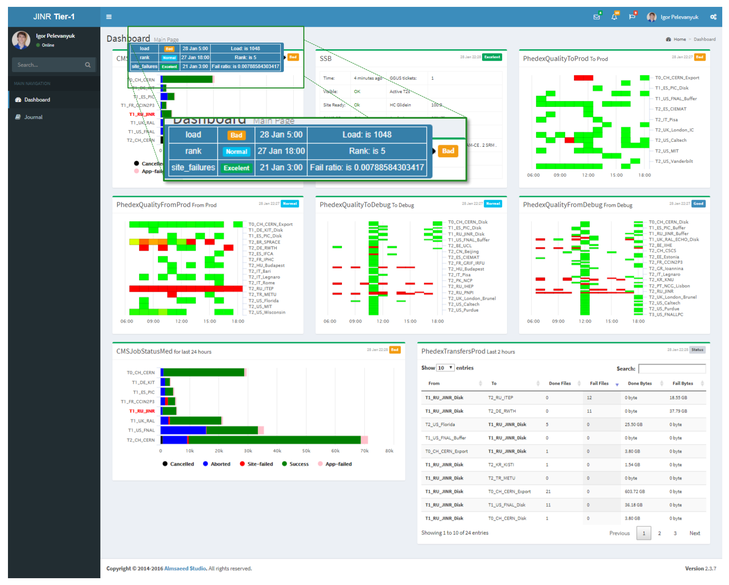
Another entities are events. Every status change is an event. Events could be also created on purpose depending on other events or data received by the module. This allows reaction to a set of different events.



Result

We were able to build a system which is successfully working now. It is capable of collecting from different sources of data information which is relevant only for Tier-1 in JINR, store it in the database and show on the one page. It was built with Python language and Django framework on server side and HTML, CSS and Javascript on the interface. Devised architecture allow to modify part of a system without affecting the entire system. And despite the fact that there are still a lot of work to do, it is already providing the fastest way to get information from 9 web-pages and there going to be more.

The following work will continue in three main directions: adding new modules, implementing reaction and evaluating possibility to use machine learning techniques for a forecasting. We are also planning to propose this tool as a general solutions to other groups in JINR.



References

1. Astakhov N.S. et al. JINR Tier-1 Centre for the CMS Experiment at LHC // Physics of Particles and Nuclei Letters – 2016. – Vol. 13, No. 5. – P.714–717.
2. Baginyan A.S. et al. Multy-level monitoring system for Multifunctional Information and Computing Complex at JINR // Proceedings of the XXVI International Symposium on Nuclear Electronics & Computing, September 25 - 29, 2017, CEUR-WS.org, - Vol. 2023 – P.226-233.

23rd international conference on Computing in High Energy and Nuclear Physics - CHEP, 9-13 July 2018, Sofia, Bulgaria