



Evaluation of Monitoring System at the Belle DIRAC

Evgeniy Kovalenko
Master Course I
HEP, BINP
Supervisor: Hideki Miyake

August 7, 2019



The Belle DIRAC system store its data in a SQL (mySQL) database that is a bit old and do not provide with comfortable application to work with logs.

- Strict structure
- Not optimised for monitoring system and slow (sometimes up to 10 min for one-month plot)

The goal of the present project is to evaluate NoSQL database (ElasticSearch and related apps) at the DIRAC as a Monitoring System and log analyzer.

- No strict structure, allow to analyse in real-time a large number of observables.
- Includes cute web-interface (Kibana)



- Elastic Search is a NoSQL (Non Structured Query Language) database
- Scales horizontally
- Kibana enables to interactively explore, visualise data
- Includes machine learning for detecting anomalies
- DIRAC communities started evaluation of the ES but not at Belle II



First tests was performed on a laptop using CentOS 7 in virtual machine

- Elasticsearch v6.8.0 in docker with official image
- Kibana v6.8.0 is installed
- Belle DIRAC v6r20p26 as [Minimal BelleDIRAC Grid System](#)
- [Monitoring and Accounting systems](#)* are activated.

* Monitoring System stores DIRAC's components statistics such as CPU, memory usages, threads and status. Accounting System stores information about current jobs. Both fully implemented into DIRAC.

Next step is to install everything on actual Belle DIRAC Development environment.



Another missing in MySQL usage of ES is a full text analyser. We can store whole Pilot and Payload logs into ES and analyse it, but it is impossible to automatically analyse undefined errors. List of predefined errors is on [confluence page](#)

Work-flow for storing logs is developed:

- 1 Download pilot or payload logs from the DIRAC server
- 2 Send jobs' attributes to uploading script
- 3 Create and upload JSON file to the ES via curl command

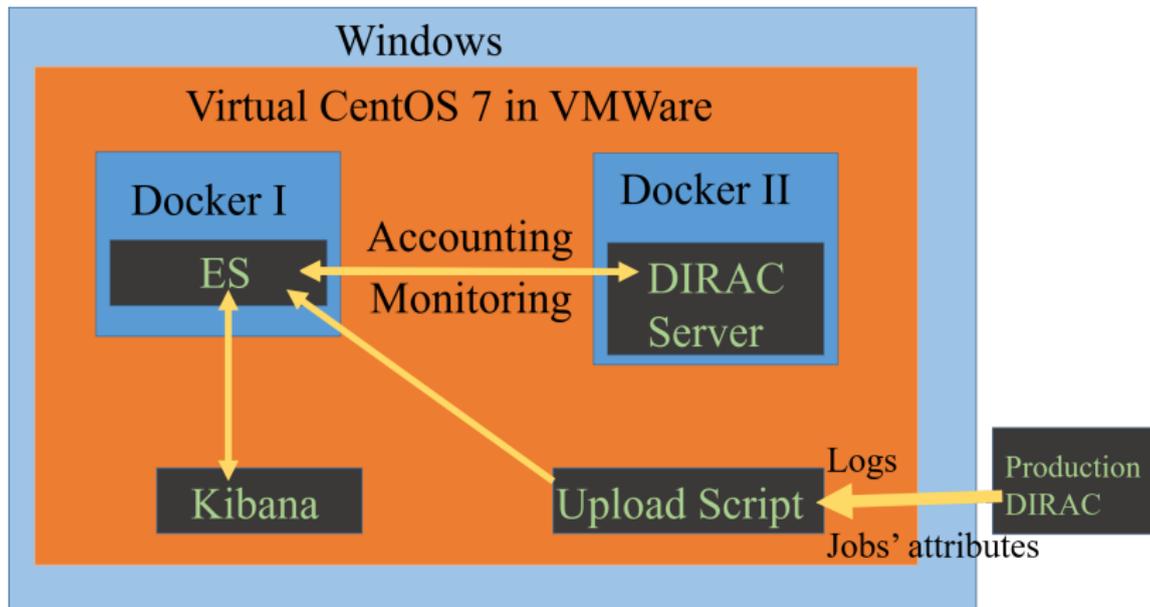


Illustration of the first test on the laptop



Test shows that Elasticsearch successfully handles logs up to 5 MB each, while Kibana interface allow to interactively search through data and visualise it on a dashboard. Unfortunately, full log text available only from the Discovery page.

There are some issues:

- Process is not automatised as soon as this is not DIRAC component, but a couple of scripts
- Sometimes log size is too big (up to 100 MB) and ruins Kibana; for now threshold is 5 MB – for larger log only its size is saved
- Connection to the ES is through `http://`



For now ES and Kibana are installed on actual Belle DIRAC Development environment. Way to access during tests (later will be accessible without port forwarding):

- Set up forwarding using kekcc account:
ssh -NL 15601:localhost:5601 user@cw01.cc.kek.jp
- Open Kibana in browser: localhost:15601
- user: guest password: guestguest

If something wrong with connection, contact Miyake-san.

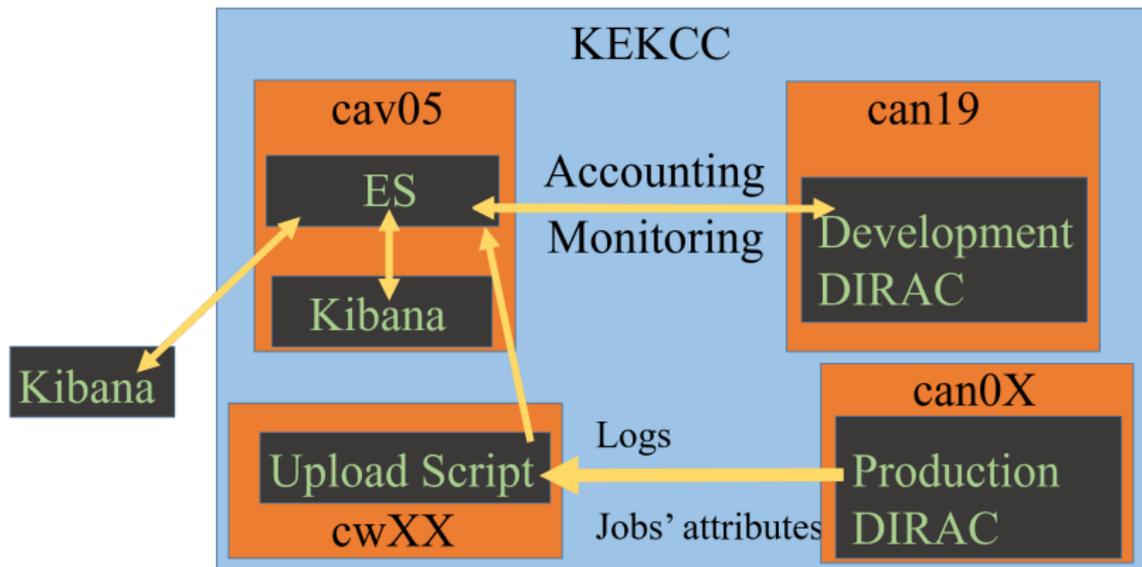
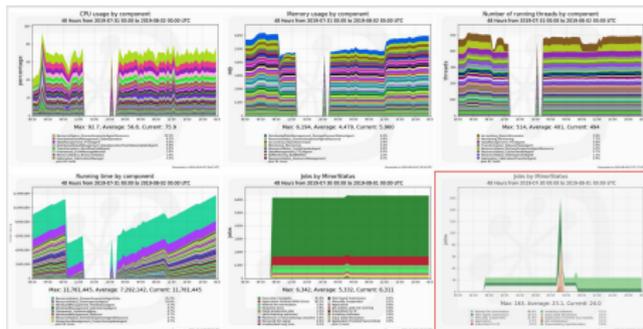
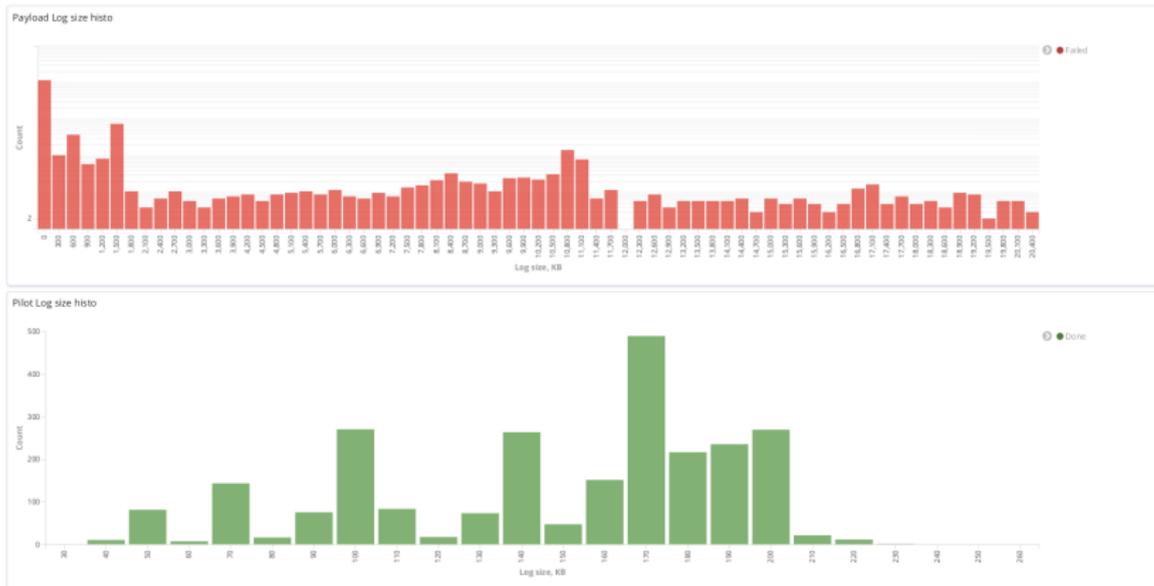


Illustration of the second test on the actual environment. Kibana could be installed anywhere.

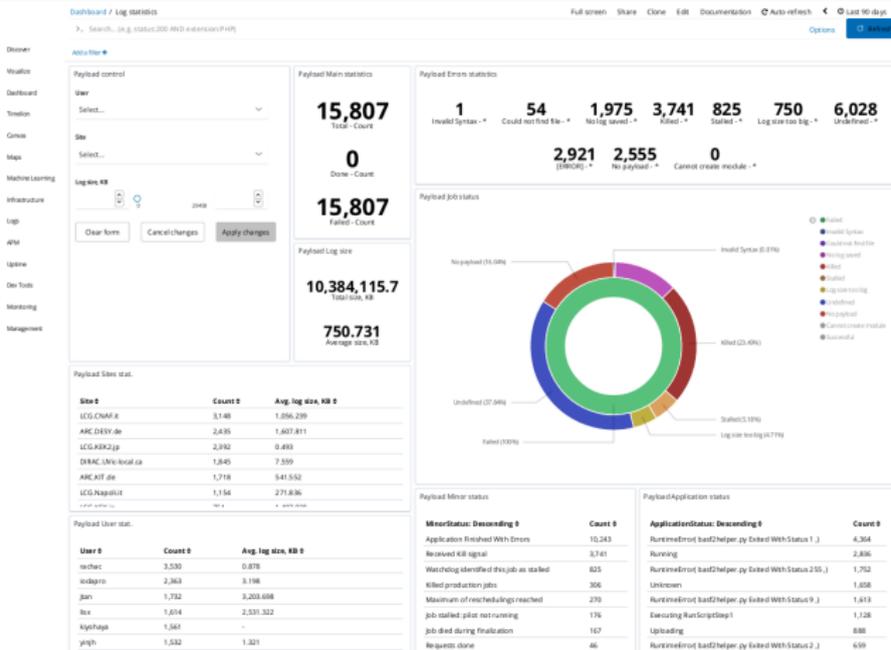


Example of Kibana dashboard for real-time monitoring (left). The same graphs could be created with DIRAC WebbApp (right).

- There is lack of settings for graphs in Kibana
- Looks like it is hard for Kibana to draw 70+ colours



Payload's (top, in log. scale) and pilot's (bottom) histograms of log size in KB



Example of a dashboard developed for log analysis. It is possible to filter by site, user or whatever in "Payload control" and even to use complicated filter by clicking corresponding value on graph/metrics. "Error statistics" based on keyword search through logs text.



- Elasticsearch and Kibana are installed on Development environment
- Monitoring and Accounting Systems are activated and stably works
- Manual work-flow for logs is established and dashboards for data visualisation are created

Next steps:

- Adjust list of predefined errors
- To store logs into ES automatically via DIRAC component