Analytical Platform for Distributed Data Production and Analysis in ATLAS Experiment at LHC

Aleksandr Alekseev
UNAB (Santiago, Chile), ISP RAS (Moscow, Russia)

IVANNIKOV ISP RAS OPEN CONFERENCE
MOSCOW, 5-6 DECEMBER, 2019

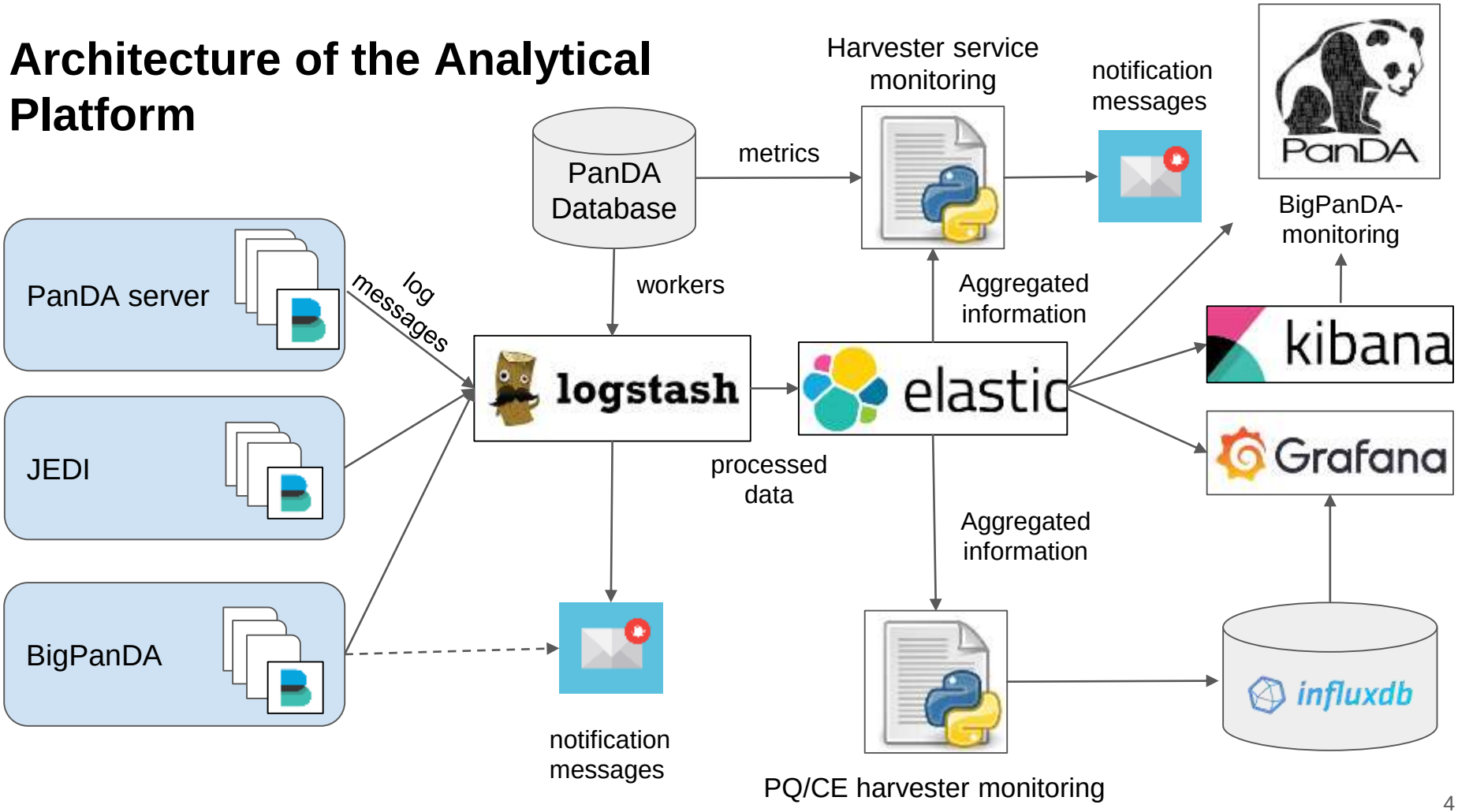
Outline

- Introduction
- Collection, processing, storage and analysis of log files of the PanDA components
- Harvester monitoring
  - via ELK-stack
  - Harvester service monitoring (HSM)
  - PQ/CE harvester monitoring
- Summary

Introduction

- **PanDA** (**P**roduction and **D**istributed **A**nalysis) is the workflow management system of the ATLAS experiment at the LHC which is responsible for generating, brokering and monitoring up to two million jobs per day across 170 computing centers in the Worldwide LHC Computing Grid, HPC, clouds, volunteer computing resources
- It consists of several components, including:
  - **JEDI** (**J**ob **E**xecution and **D**efinition **I**nterface)
  - **PanDA server**
  - **BigPanDA monitor**
  - **Harvester service**
- The components generate **~ 1 TB of metadata** per day in PanDA database and log files which are generated on distributed nodes
- **Analytical Platform** was developed to monitor and analyze metadata from different data sources given the specifics of these components

Architecture of the Analytical Platform



Collection, processing, storage and analysis of log files of PanDA components

Introduction

- Logs processing:
  - **PanDA server** is a core of the PanDA system that distributes and manages jobs among computing resources
  - **JEDI** (**J**ob **E**xecution and **D**efinition **I**nterface) is an intelligent component in the PanDA system to have capability for task-level workload management
  - **BigPanDA** is a monitoring system which provides a comprehensive and coherent view of the tasks and jobs executed by the system, from high level summaries to detailed drill-down job diagnostics
- The components are deployed on distributed nodes
- There are several types of log files, each of them may contain up to **2 million messages/day** including errors description and another useful for monitoring information. Each type of log has a unique metadata structure

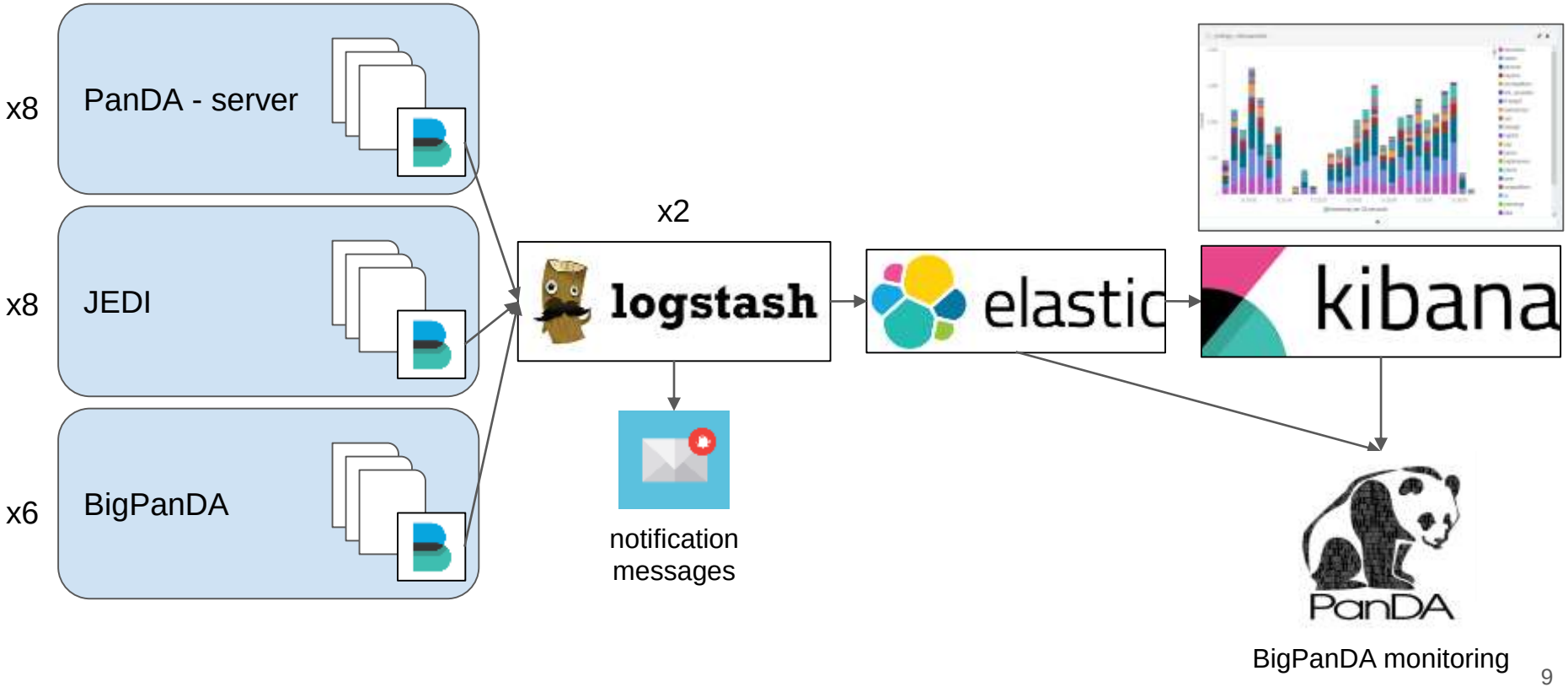
Logs Collection in PanDA Infrastructure

- **HTTP server and Oracle based log collection**
  - PanDA and JEDI loggers were instrumented with a HTTP handler, which sent selected messages to a dedicated HTTP server
  - Disadvantages:
    - **Coupling**
    - **Low scalability**
    - **Accessing the data**
    - **Representation**
- **Flume server and ElasticSearch based log collection**
  - This solution consisted in replacing the HTTP server by load-balanced Flume servers, which shipped the messages to ElasticSearch ATLAS cluster
  - Disadvantages:
    - **Coupling**
    - **Low scalability**
- **ElasticSearch + Logstash + Kibana (ELK) stack based log collection (mid 2016)**

ELK - stack

- **Filebeat** - data collecting from log files
- **Logstash** - filtering and normalizing logs
- **ElasticSearch** - storing and searching information
  - **Logstash** writes processed log messages into the ElasticSearch repository. It is also used by other services and systems: *Rucio, Harvester and etc.*
- **Kibana** - visualization system for ElasticSearch that provides a flexible mechanism for filtering, grouping in the dashboard and visual data analysis using interactive graphical representations

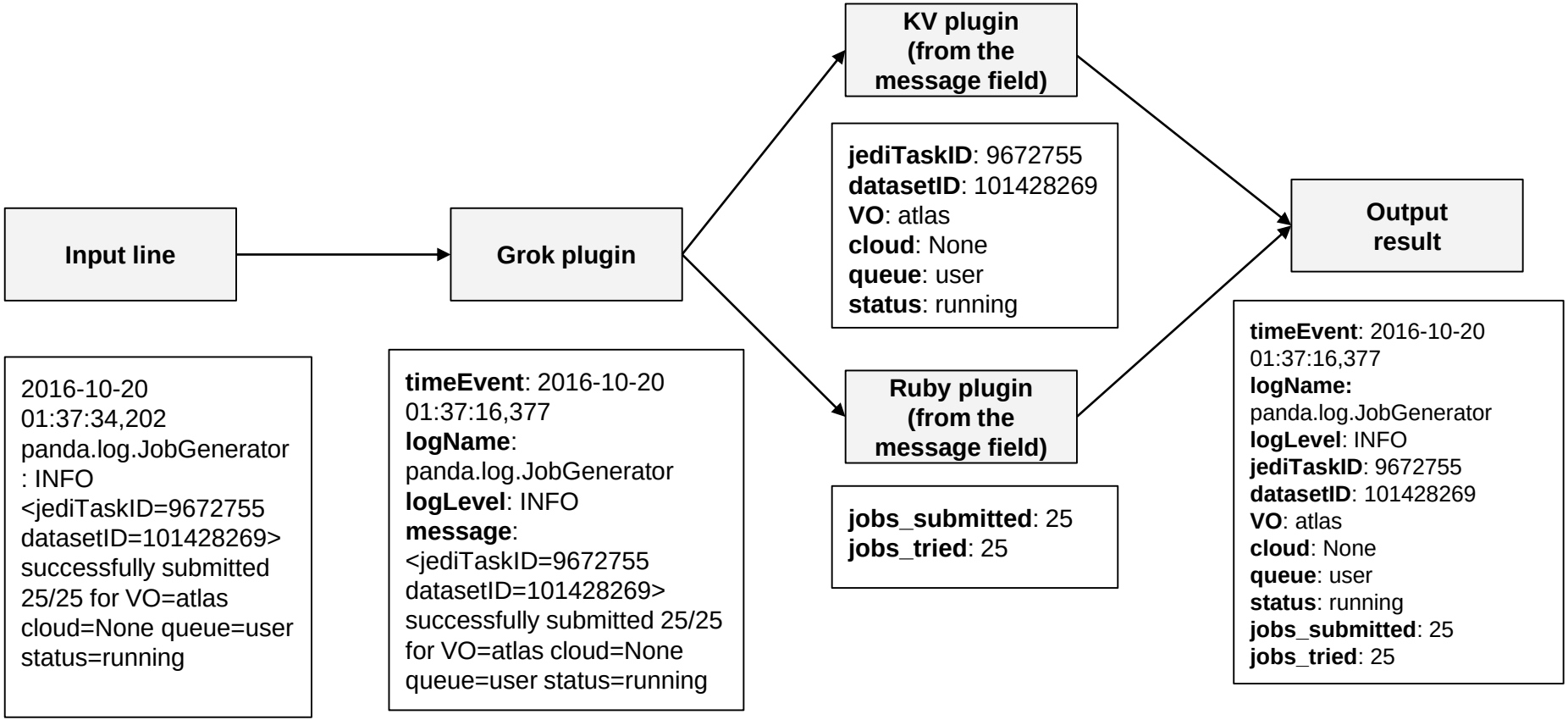
ELK-stack for Processing Logs in PanDA Infrastructure



Multithread Log Processing

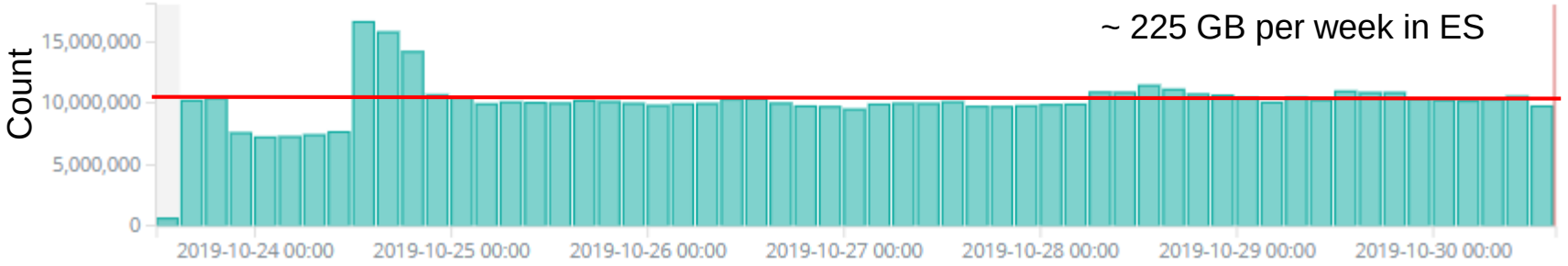
- 69 logstash configuration files
  - (27 for JEDI | 34 for PanDA | 6 for BigPanDA | 2 input&export)
- Each configuration file can contain several plugins:
  - **Grok**
    - uses a regular expression to split strings into separate fields
  - **KV**
    - extracts all values that match the pattern of key-value
  - **Ruby**
    - for processing strings and getting necessary values using special conditions

Example of Log Processing



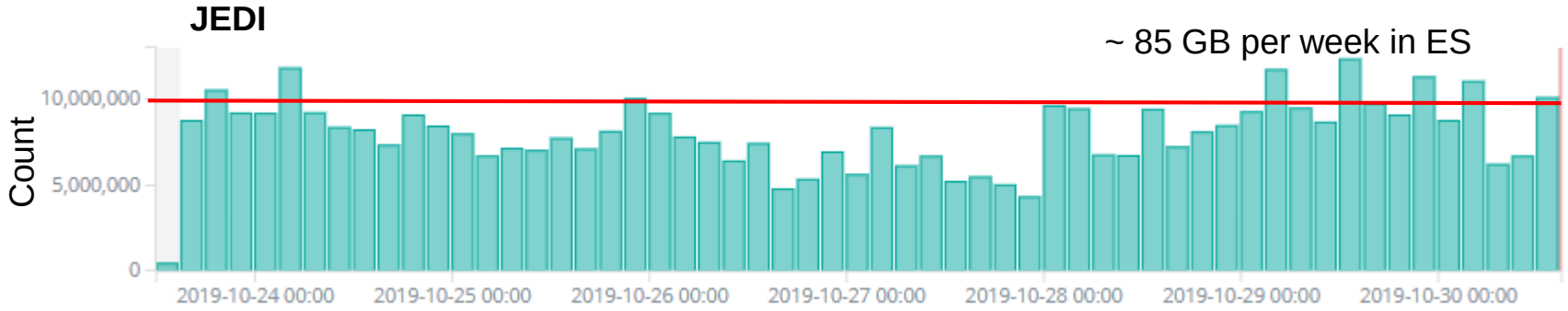
Scale of Processed Data in ES and Kibana

PanDA server



Timestamp per 3 hours

JEDI

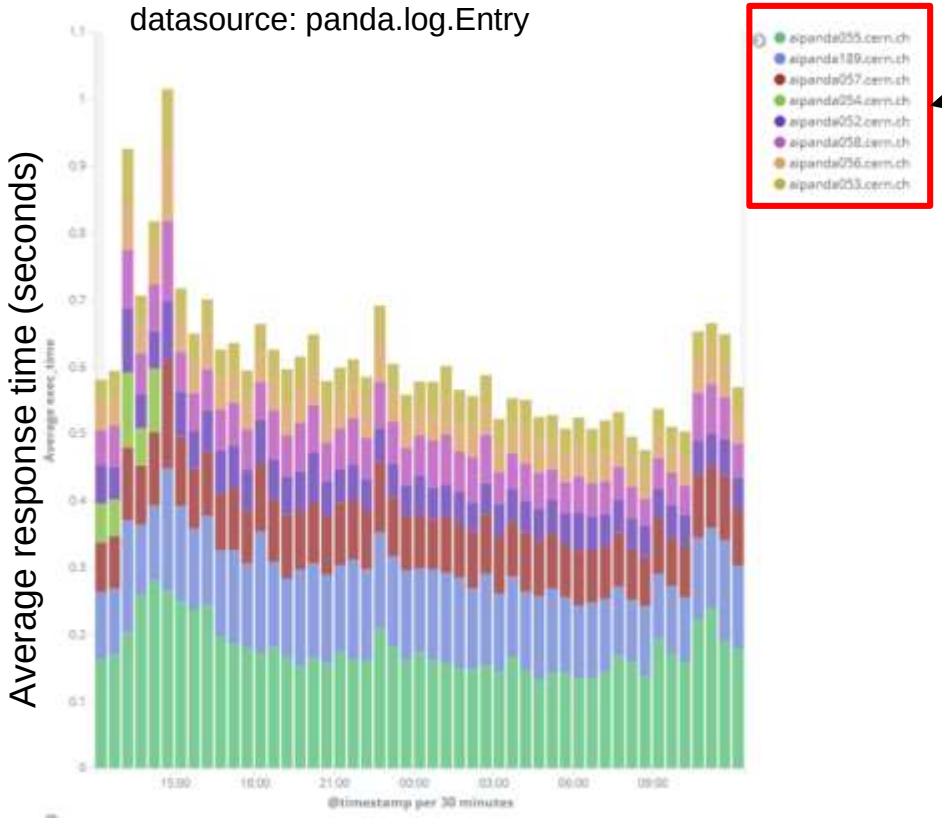


Timestamp per 3 hours

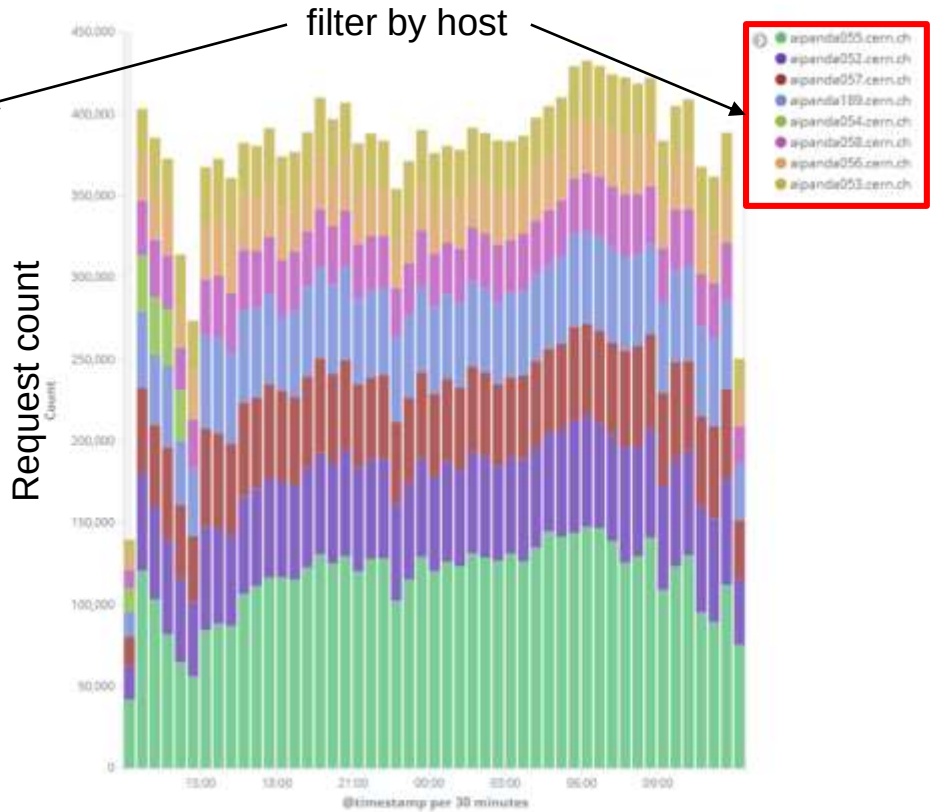
Panda Server Health Metrics Dashboard

datasource: panda.log.Entry

filter by host



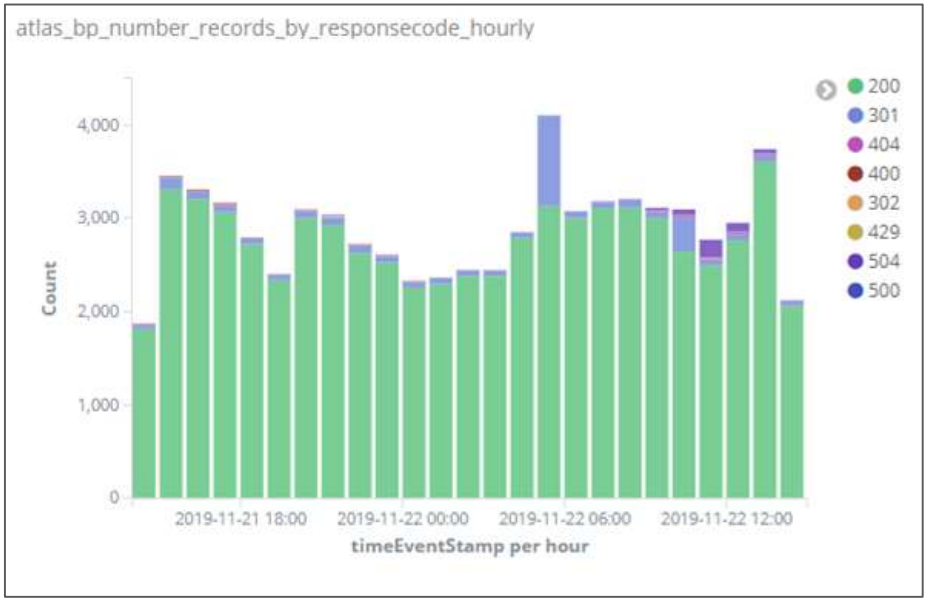
Timestamp per 30 minutes



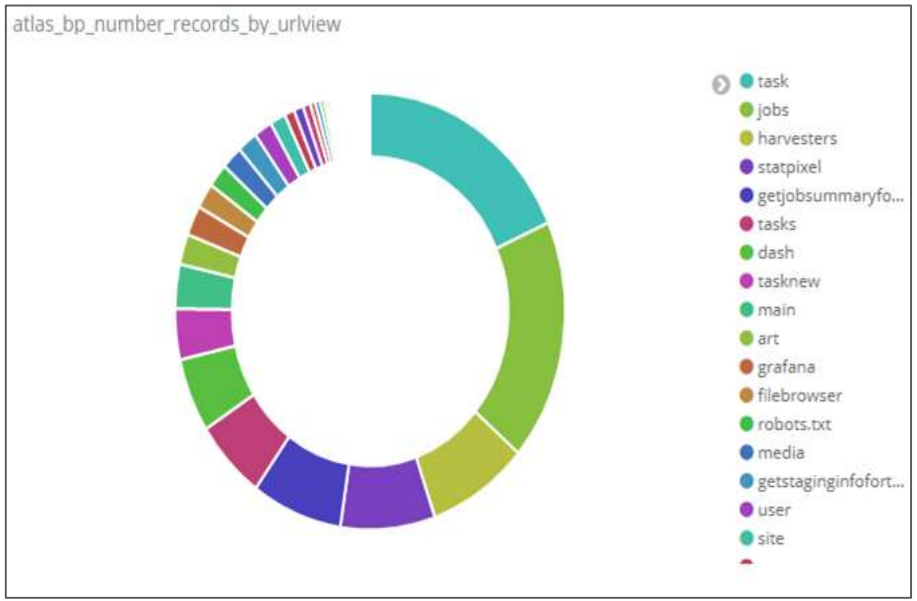
Timestamp per 30 minutes

Dashboard for monitoring of BigPanDA monitoring system

Number of requests grouped by response codes



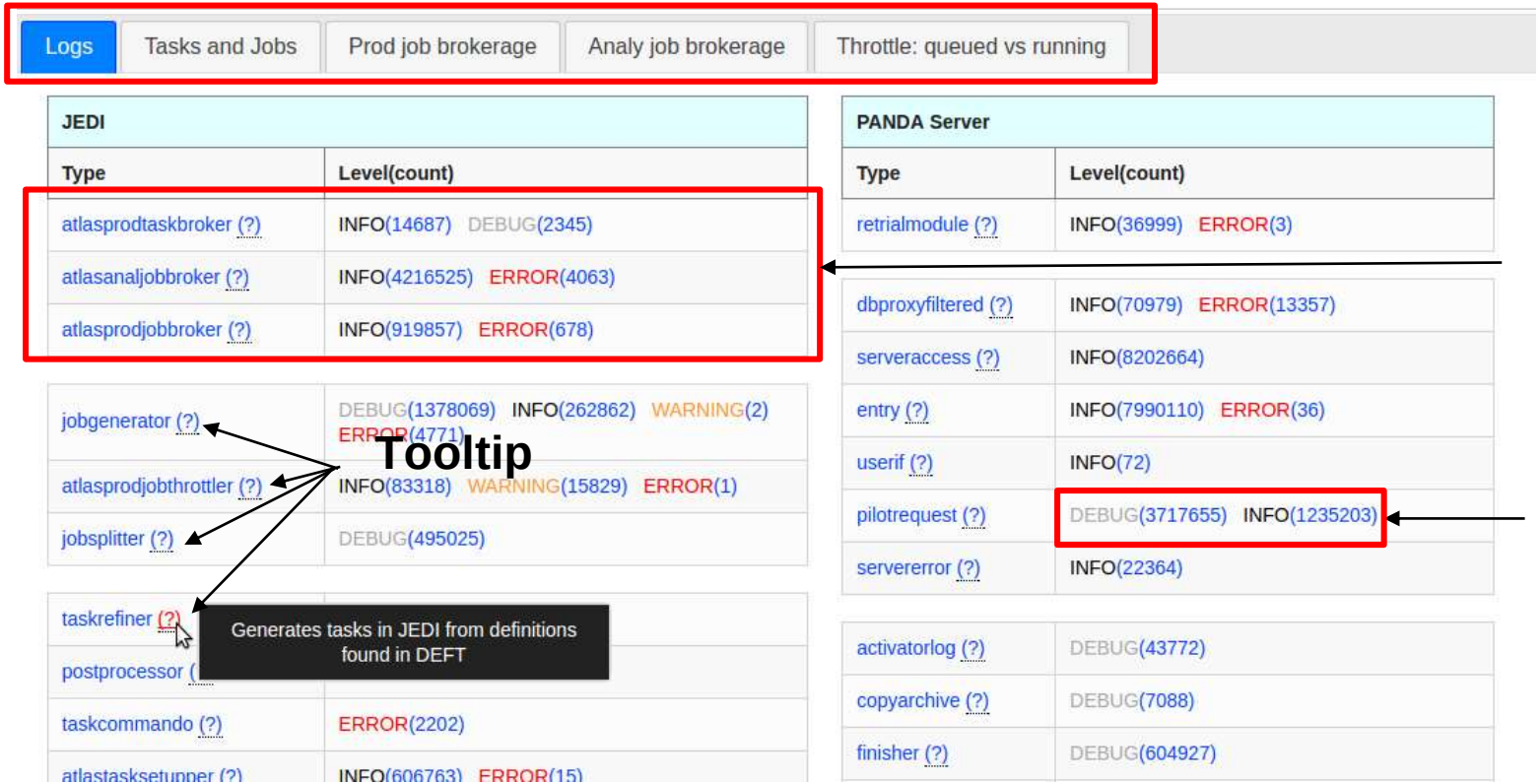
Number of requests grouped by URL view



* Based on Apache and Django application logs

Integration With BigPanDA Monitoring. Logs View

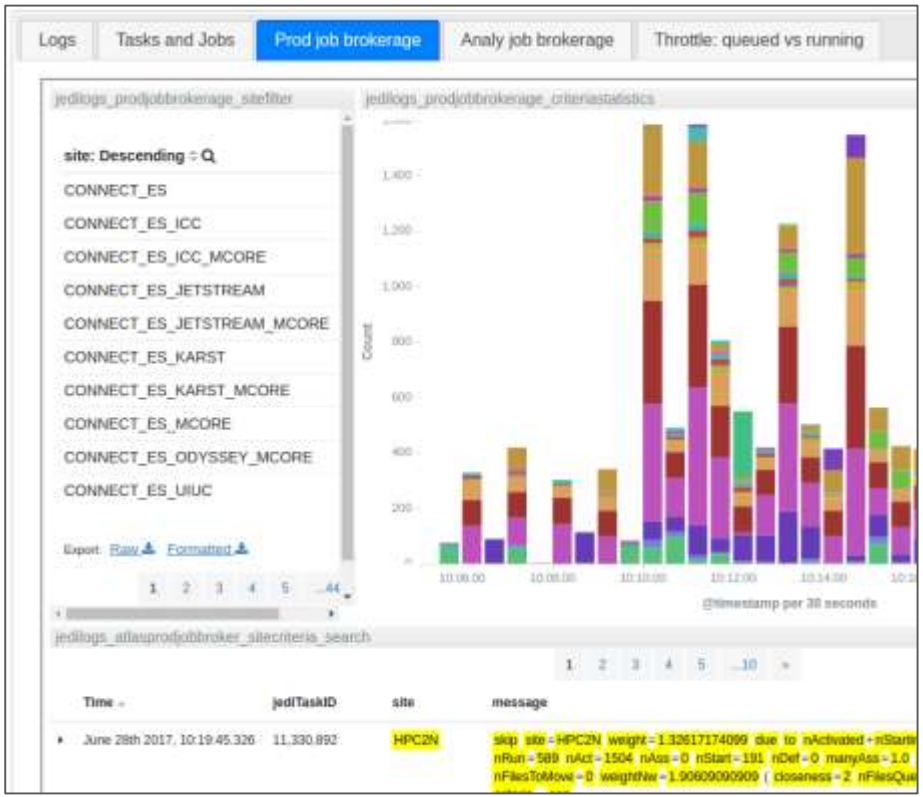
https://bigpanda.cern.ch/esatlaslogger/

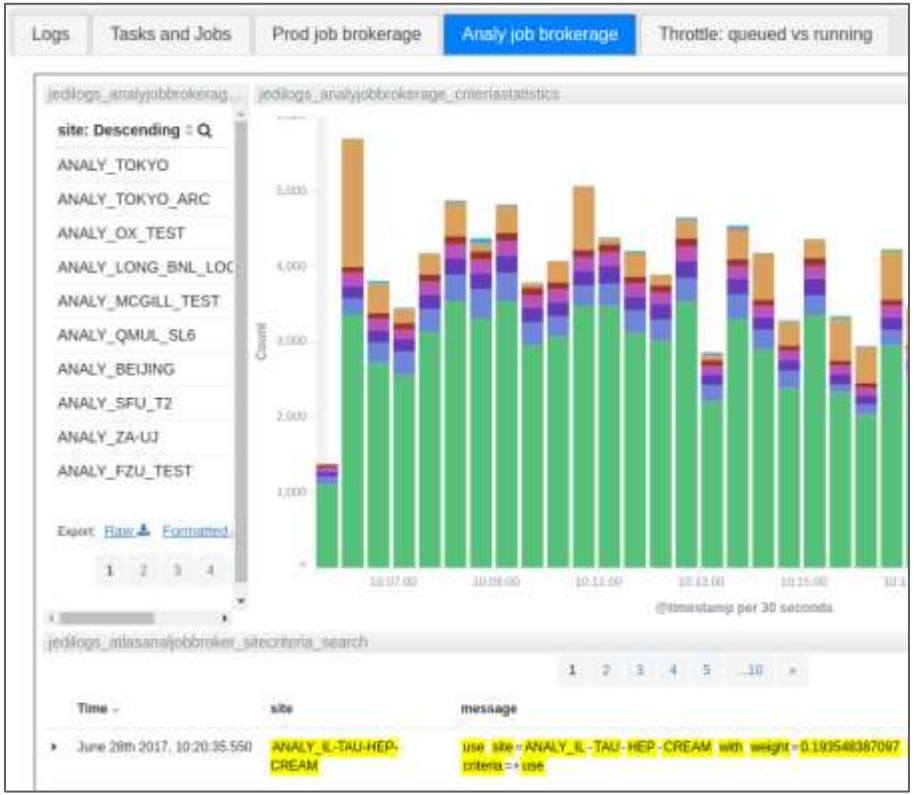


Log types ordered by interest

Log level and number of entries

Integration With BigPanDA Monitoring. Dashboards



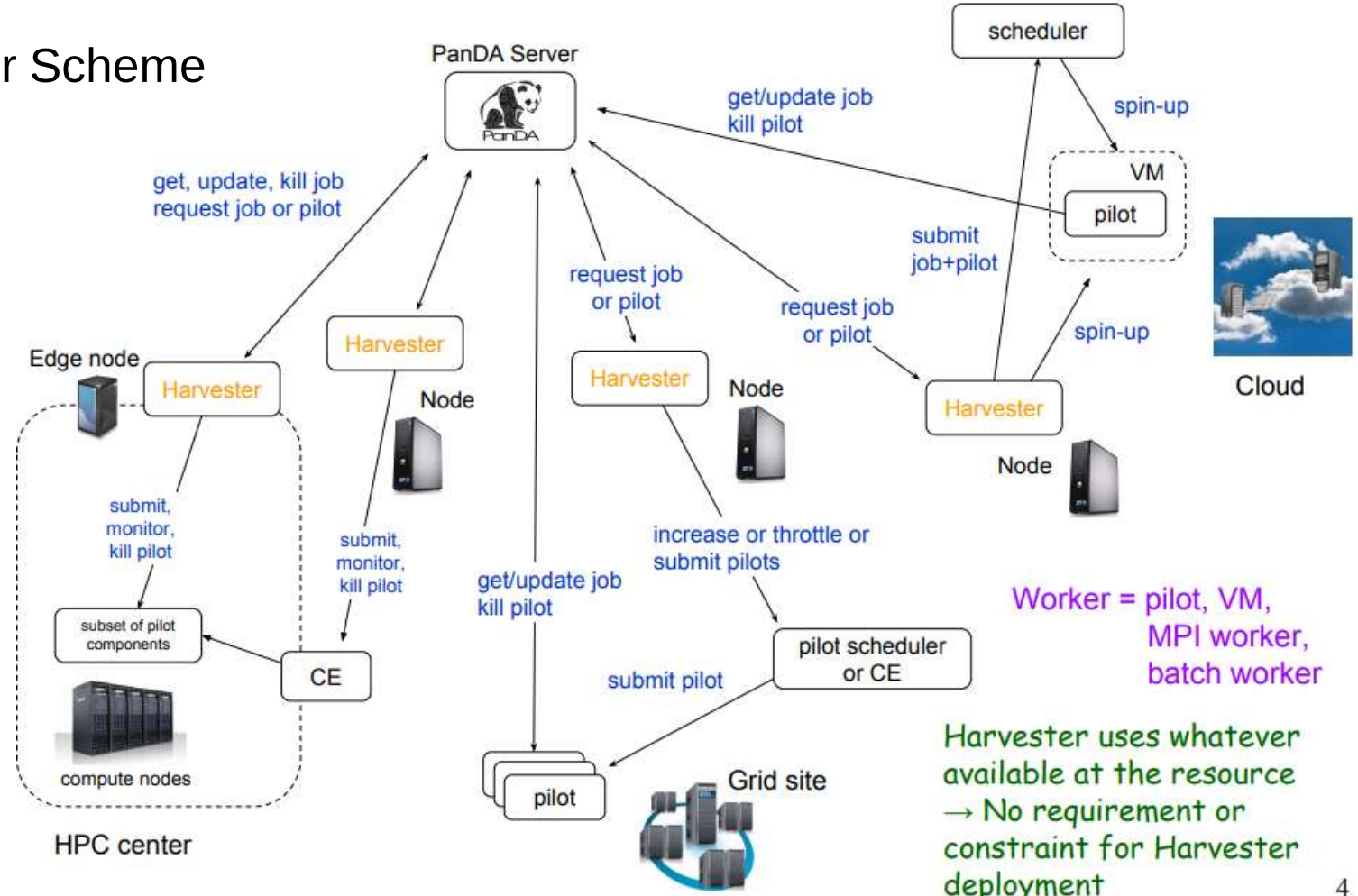


Harvester Monitoring via ELK-stack

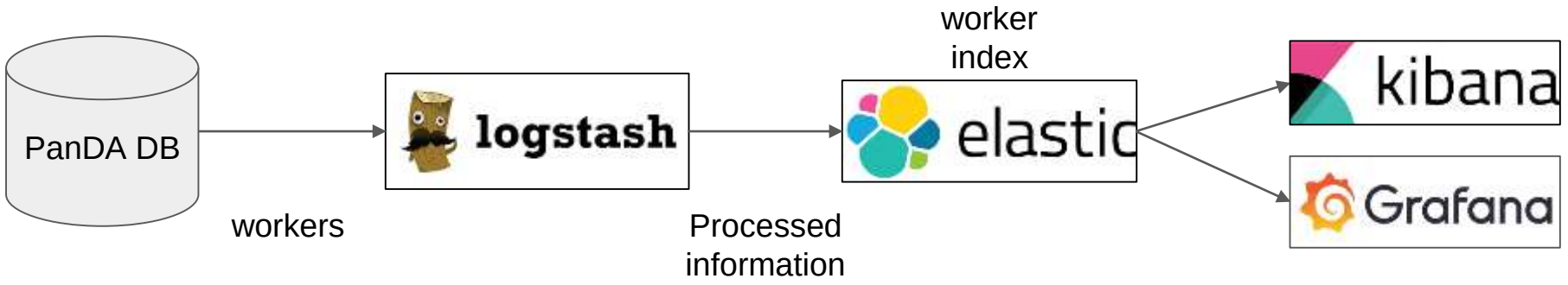
Introduction

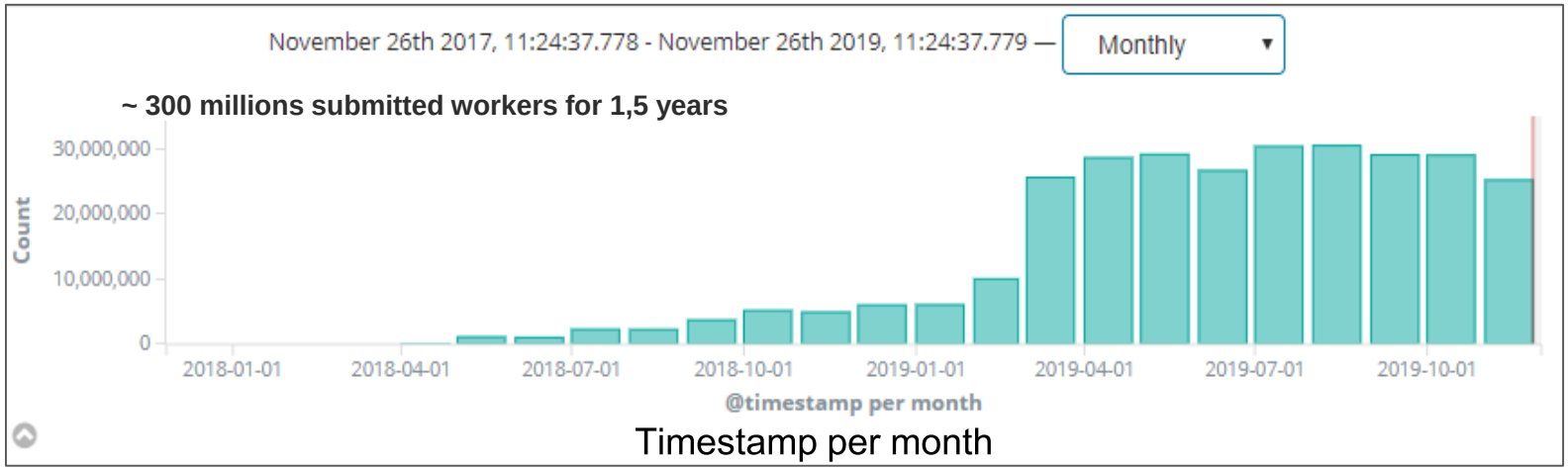
- **Harvester** is a resource-facing service between the PanDA server and collection of pilots. It is stateless with a modular design to work with different resource types (GRID, HPC, CLOUD) and workflows
- **Worker** is abstraction of Harvester service
- Harvester stores metadata in the PanDA database in separate tables:
  - Workers table. A lifetime of the table in PanDA DB is 6 months (Contains ~ 90 millions entries)
  - Metrics table
  - Diagmessages table
  - etc.
- Logstash is used to copy and process this metadata in ElasticSearch storage

Harvester Scheme



Harvester Monitoring via ELK-stack

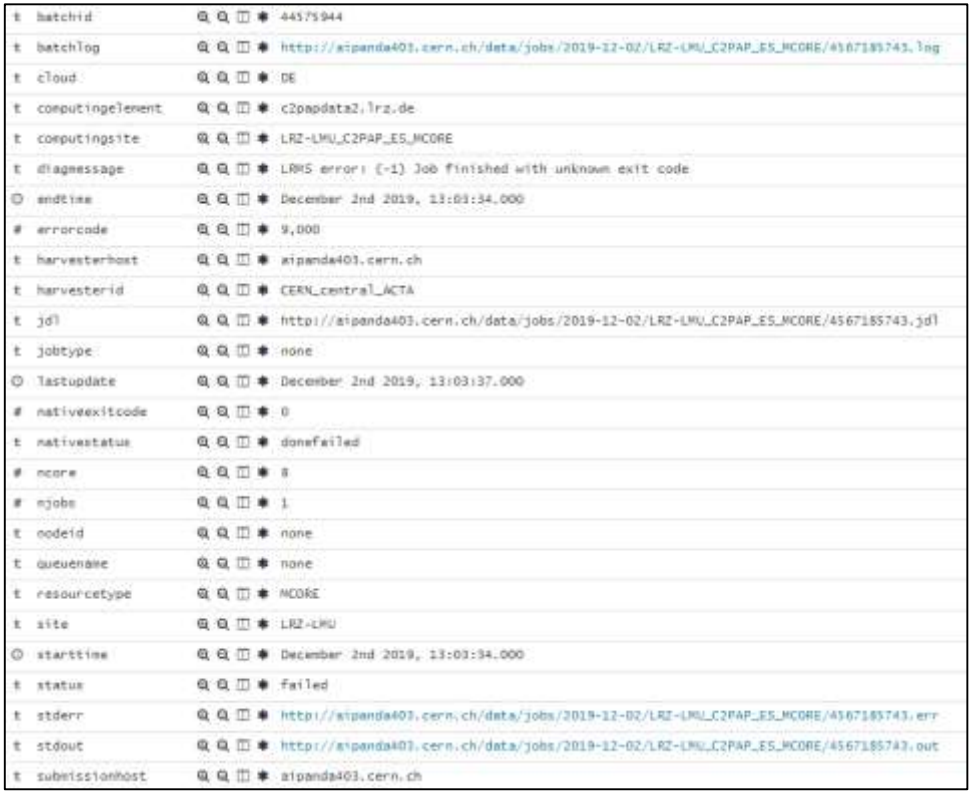




Timestamp per month

Harvester Worker data example in ES and Kibana

- Automatically expanding data in ES
- Workers fields:
  - computingelement
  - computingsite
  - status
  - diagmessage
  - etc.



t	batchid	44575944
t	batchlog	http://aipanda403.cern.ch/data/jobs/2019-12-02/LRZ-LMU_C2PAP_ES_MCORE/4567185743.log
t	cloud	DE
t	computingelement	c2papdata2.lrz.de
t	computingsite	LRZ-LMU_C2PAP_ES_MCORE
t	diagmessage	LRMS error: (-1) Job finished with unknown exit code
⏲	endtime	December 2nd 2019, 13:03:34.000
#	errorcode	9,000
t	harvesterhost	aipanda403.cern.ch
t	harvesterid	CERN_central_ACTA
t	jdl	http://aipanda403.cern.ch/data/jobs/2019-12-02/LRZ-LMU_C2PAP_ES_MCORE/4567185743.jdl
t	jobtype	none
⏲	lastupdate	December 2nd 2019, 13:03:37.000
#	nativeexitcode	0
t	nativestatus	donefailed
#	ncore	8
#	njobs	1
t	nodeid	none
t	queuename	none
t	resourcetype	MCORE
t	site	LRZ-LMU
⏲	starttime	December 2nd 2019, 13:03:34.000
t	status	failed
t	stderr	http://aipanda403.cern.ch/data/jobs/2019-12-02/LRZ-LMU_C2PAP_ES_MCORE/4567185743.err
t	stdout	http://aipanda403.cern.ch/data/jobs/2019-12-02/LRZ-LMU_C2PAP_ES_MCORE/4567185743.out
t	submissionhost	aipanda403.cern.ch

Kibana Worker Dashboard

Filters

JEDI

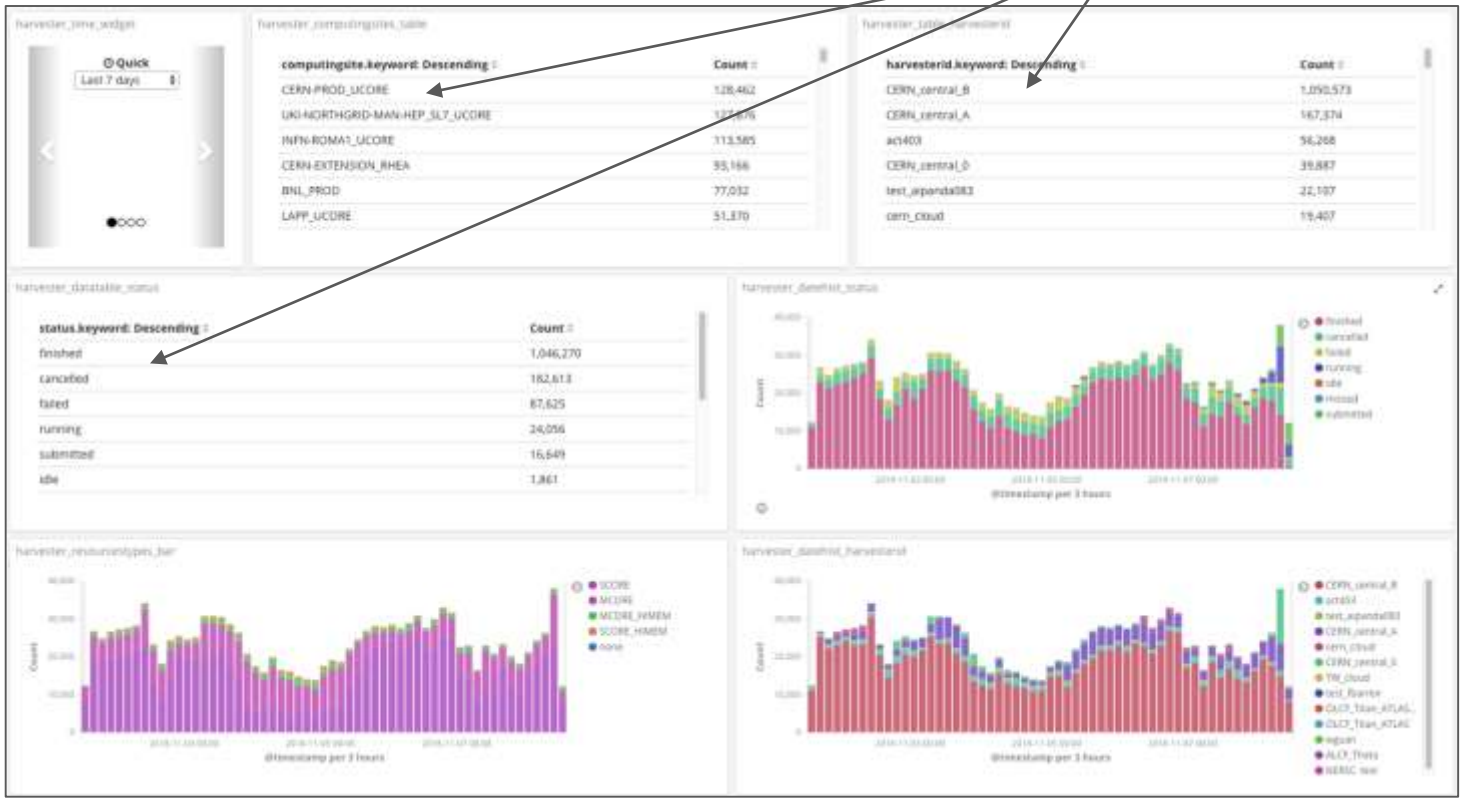
- Prod task brokerage
- Prod job brokerage
- Analy job brokerage
- Throttle: queued vs running
- Throttle: queued vs running per WorkQueue
- Job generator (WIP)
- Disk IO
- Rucio timings

PanDA server

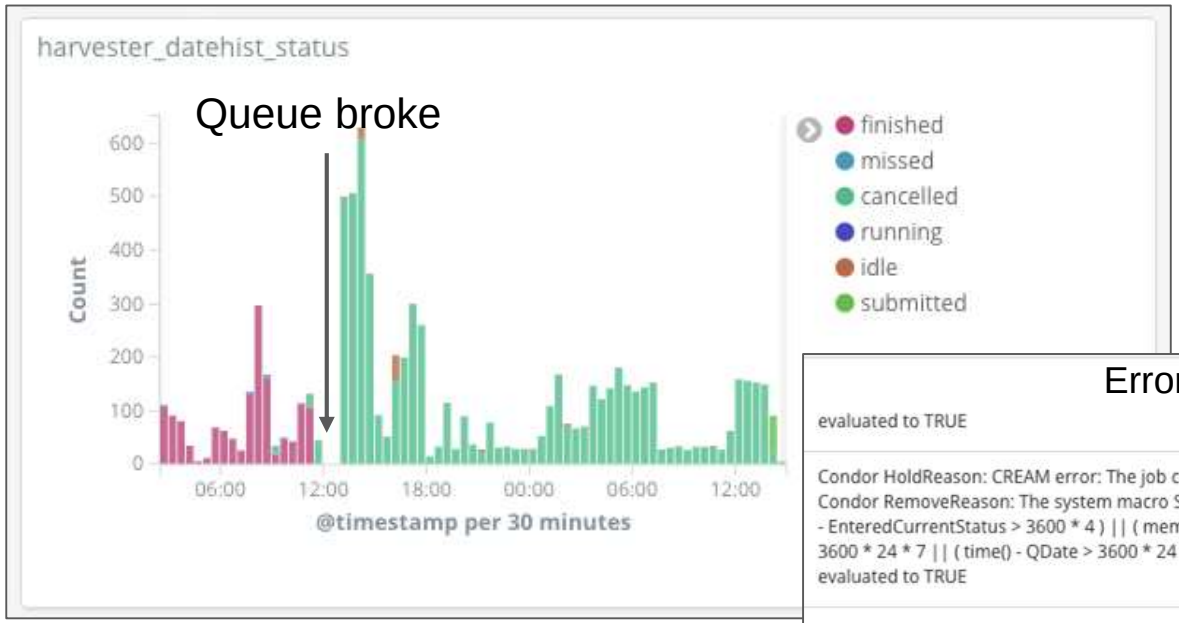
- Server metrics
- Retry module actions

Harvester

- Worker dashboard
- CE monitoring
- Job-worker monitoring [BETA]
- Harvester service metrics
- Condor service metrics



Kibana Worker Dashboard for a single PanDA queue



Errors description

evaluated to TRUE

Condor HoldReason: CREAM error: The job cannot be submitted because the blparser service is not alive ; Condor RemoveReason: The system macro SYSTEM_PERIODIC_REMOVE expression '(JobStatus == 5 && time() - EnteredCurrentStatus > 3600 * 4) || (member(JobStatus,{ 1,2,5,6,7 }) && (time() - EnteredCurrentStatus > 3600 * 24 * 7 || (time() - QDate > 3600 * 24 * 15) || time() - LastRemoteStatusUpdate > 3600 * 4))' evaluated to TRUE

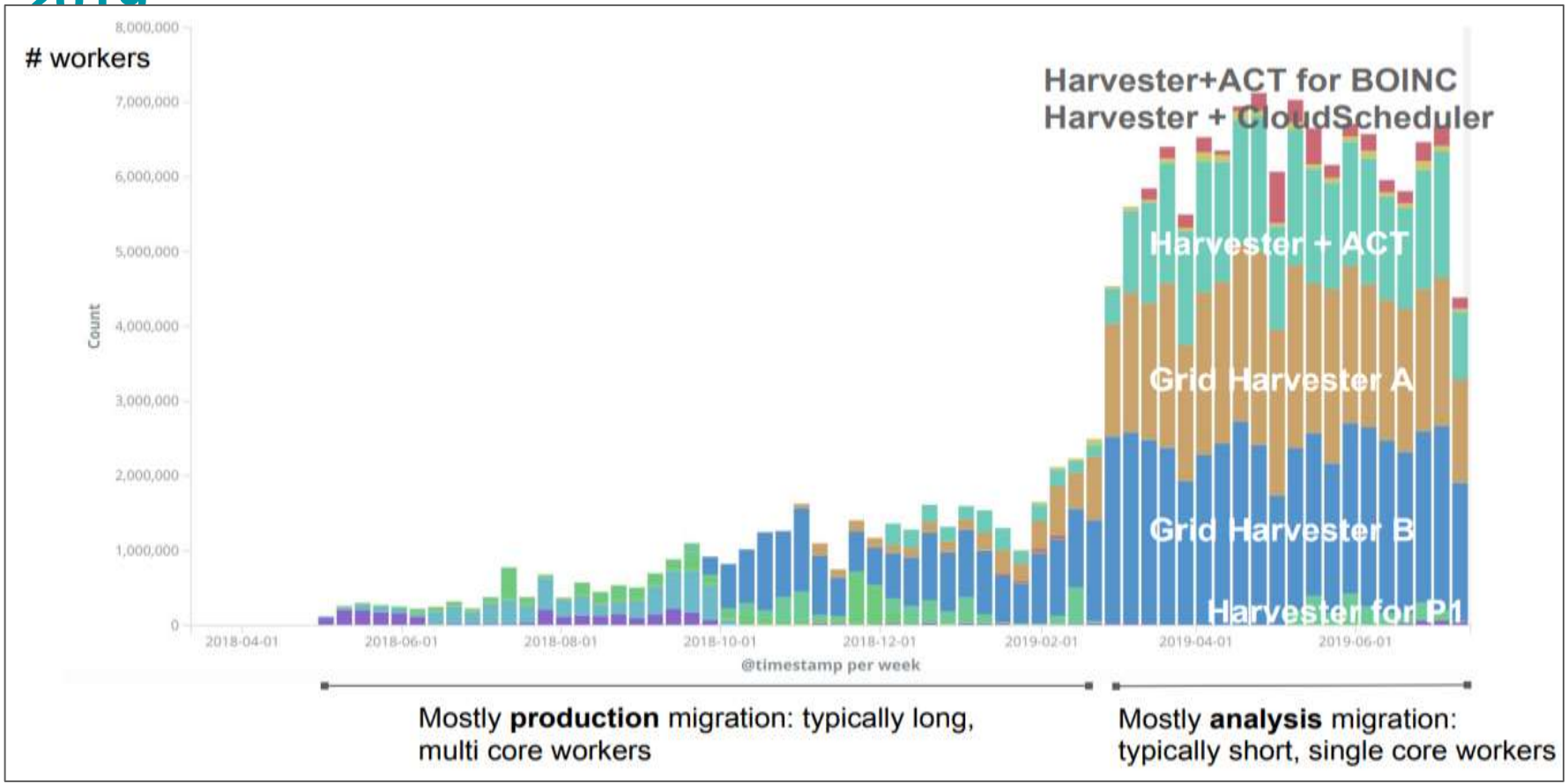
Condor HoldReason: CREAM error: The job cannot be submitted because the blparser service is not alive ; Condor RemoveReason: The system macro SYSTEM_PERIODIC_REMOVE expression '(JobStatus == 5 && time() - EnteredCurrentStatus > 3600 * 4) || (member(JobStatus,{ 1,2,5,6,7 }) && (time() - EnteredCurrentStatus > 3600 * 24 * 7 || (time() - QDate > 3600 * 24 * 15) || time() - LastRemoteStatusUpdate > 3600 * 4))' evaluated to TRUE

Harvester Monitoring in Grafana

- **Grafana** - visualization system in CERN for the different monitorings
- More intuitive GUI for general users
- Same visualization technology as **DDM** (**D**istributed **D**ata **M**anagement) accounting and **Job** Dashboards

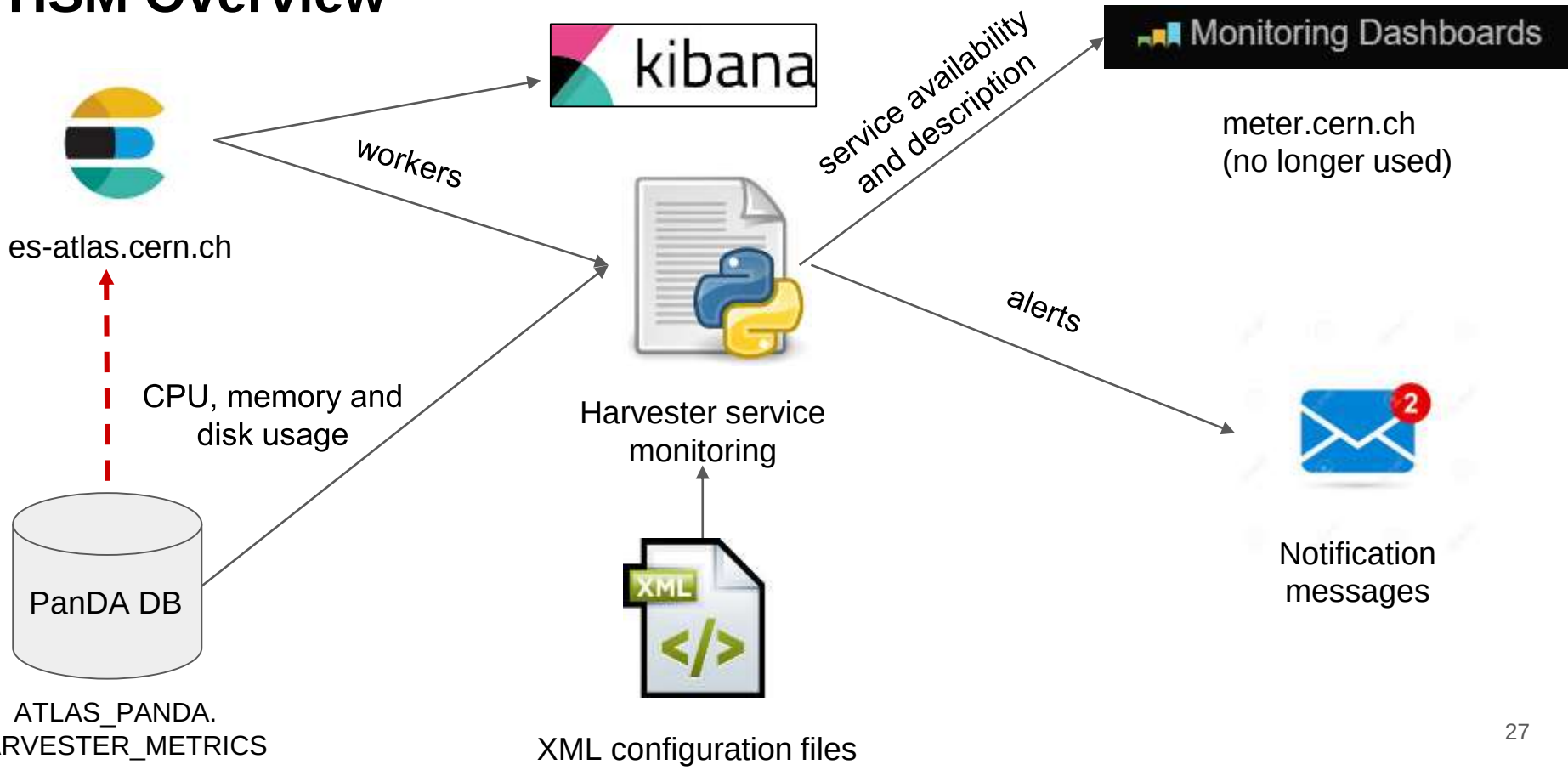
Harvester Commissioning Evolution

CHEP 2019



Harvester Service Monitoring (HSM)

HSM Overview



XML Configuration Files for HSM management

- One configuration file per harvester instance
- One instance may contain multiple tags of host
- Any instance/host/metric can be disabled, enabled or changed
- CPU, Memory and Disk tags contains two thresholds: Warning and Critical

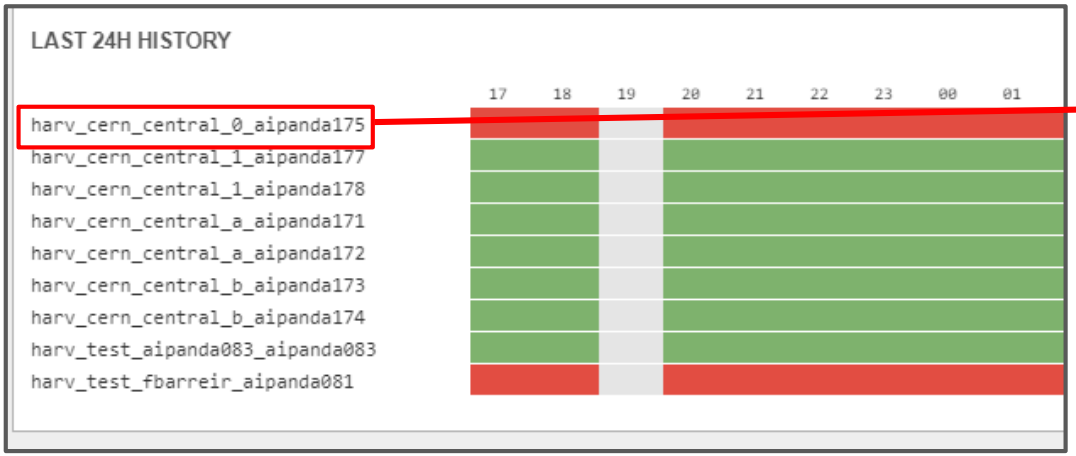
```
<?xml version="1.0"?>
<instances>
    <instance harvesterid="CERN_central_1" instanceisenable="True">
        <hostlist>
            <host hostname="aipanda177.cern.ch" hostisenable="True">
                <contacts>
                    <email>forcontact@mail.com</email>
                    <email>forcontact@mail.com</email>
                </contacts>
                <memory>6000</memory>
                <metrics>
                    <metric name="lastsubmittedworker" enable="True">
                        <value>30</value>
                    </metric>
                    <metric name="lastheartbeat" enable="True">
                        <value>30</value>
                    </metric>
                    <metric name="memory" enable="True">
                        <memory_warning>50</memory_warning>
                        <memory_critical>80</memory_critical>
                    </metric>
                    <metric name="cpu" enable="True">
                        <cpu_warning>50</cpu_warning>
                        <cpu_critical>80</cpu_critical>
                    </metric>
                    <metric name="disk" enable="True">
                        <disk_warning>70</disk_warning>
                        <disk_critical>80</disk_critical>
                    </metric>
                </metrics>
            </host>
            <host hostname="aipanda178.cern.ch" hostisenable="True">
```

HSM overview

Service issues on test_fbarreir aipanda081.cern.ch
atlpan@mail.cern.ch [atlpan@mail.cern.ch]

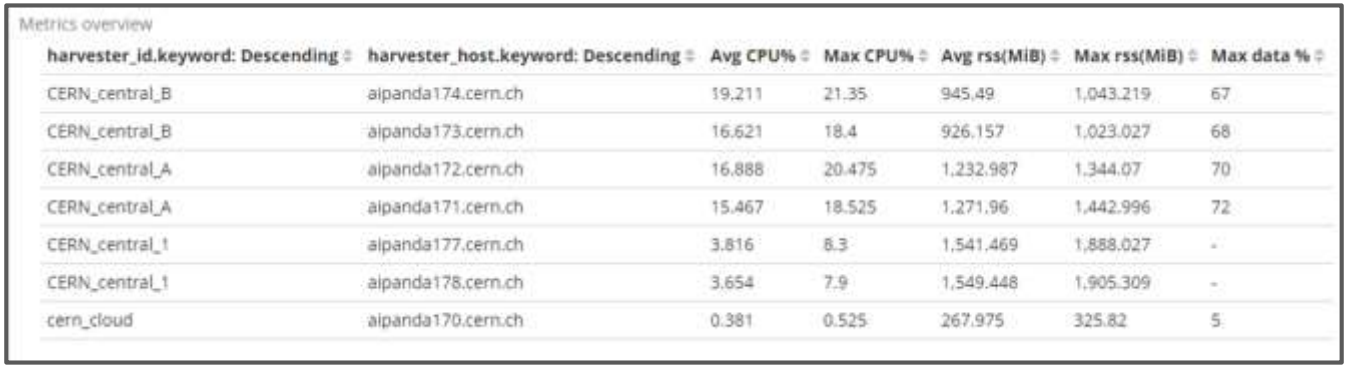
Sent: 20 March 2019 08:00
To: Aleksandr Alekseev; Fernando Harald Barreiro Megino

```
Last heartbeat was 2019-03-15 00:06:40
Last submitted worker was 2019-03-14 11:53:14
```



meter.cern.ch (no longer used)

HSM. Service metrics dashboard at Kibana

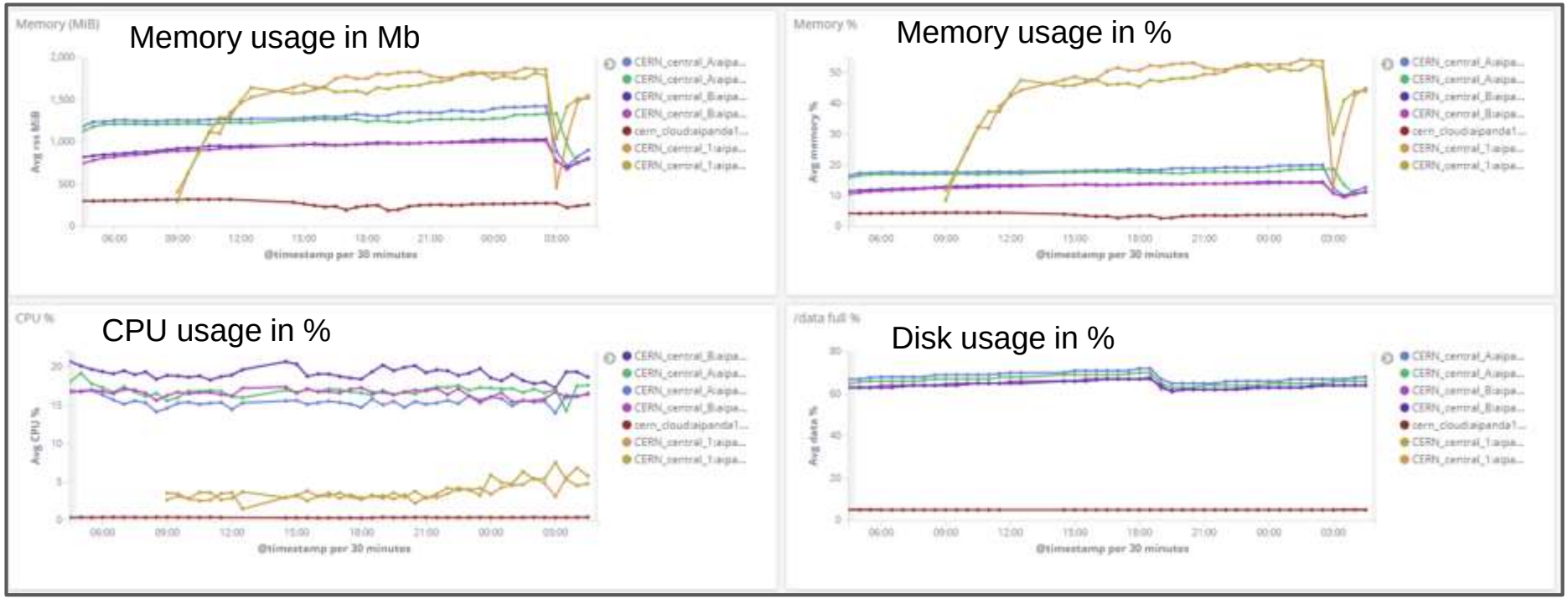


Metrics overview

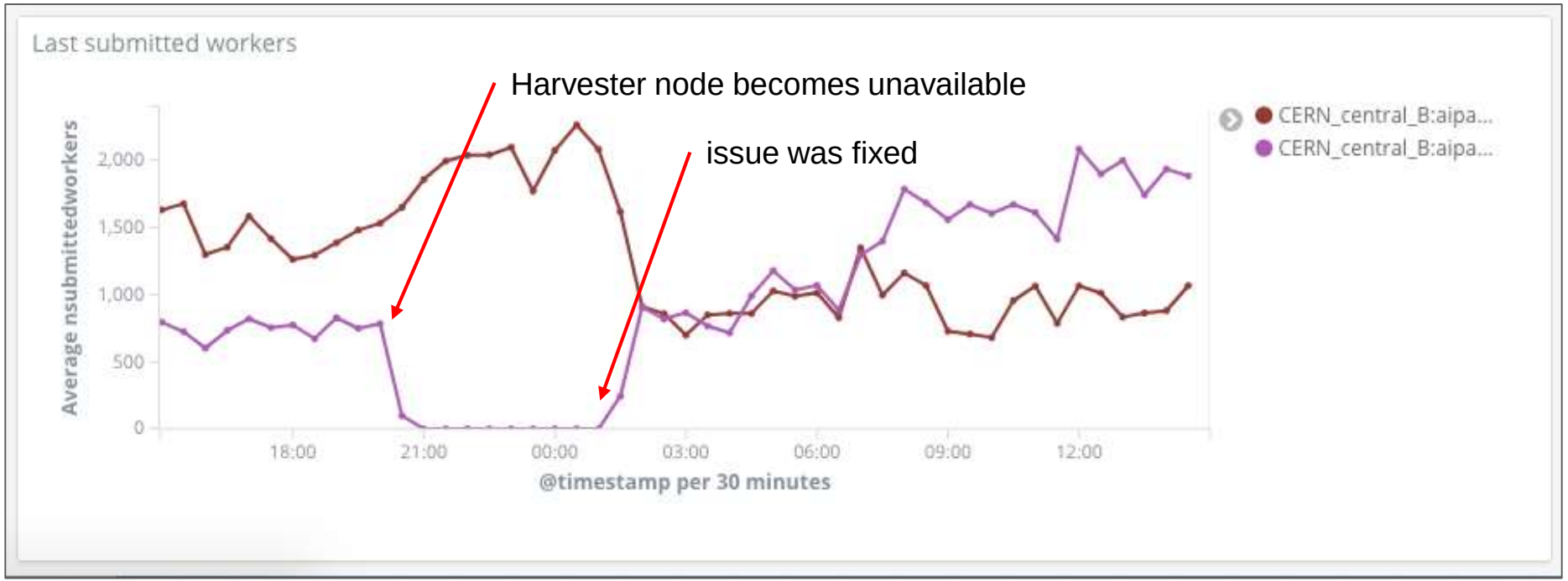
harvester_id.keyword: Descending	harvester_host.keyword: Descending	Avg CPU%	Max CPU%	Avg rss(MiB)	Max rss(MiB)	Max data %
CERN_central_B	aipanda174.cern.ch	19.211	21.35	945.49	1,043.219	67
CERN_central_B	aipanda173.cern.ch	16.621	18.4	926.157	1,023.027	68
CERN_central_A	aipanda172.cern.ch	16.888	20.475	1,232.987	1,344.07	70
CERN_central_A	aipanda171.cern.ch	15.467	18.525	1,271.96	1,442.996	72
CERN_central_1	aipanda177.cern.ch	3.816	8.3	1,541.469	1,888.027	-
CERN_central_1	aipanda178.cern.ch	3.654	7.9	1,549.448	1,905.309	-
cern_cloud	aipanda170.cern.ch	0.381	0.525	267.975	325.82	5

- Visualizations for service metrics:
  - Metrics overview table
  - Memory usage in Mb
  - Memory usage in %
  - CPU usage in %
  - Disk usage in %
- Visualizations for monitoring of workers (for last 30 minutes):
  - Last submitted workers
  - Active workers
  - Last updated workers
  - Completed workers

HSM. Visualizations for service metrics in Kibana



HSM. Example of detected issue in the past



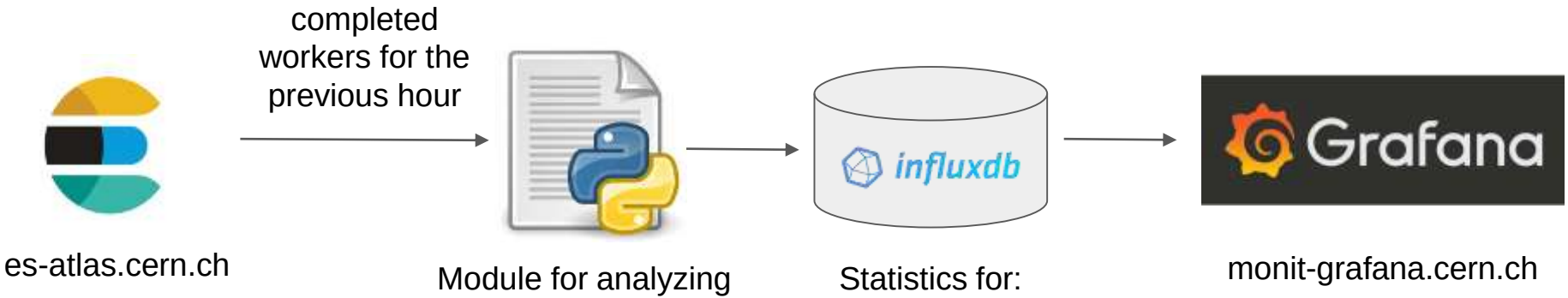
PQ/CE harvester monitoring

(Panda Queue/Computing Element)

PQ/CE harvester monitoring. Introduction

- PQ = PanDA queue (computing site), CE = computing element
- Monitors Harvester worker submission issues at PanDA queues and CEs
- Four components:
  - **ElasticSearch**: repository with worker information
  - **Python module** for data extraction and analysis
  - **InfluxDB**: storage of analyzed data
  - **Grafana**: visualization system

PQ/CE harvester monitoring architecture



es-atlas.cern.ch

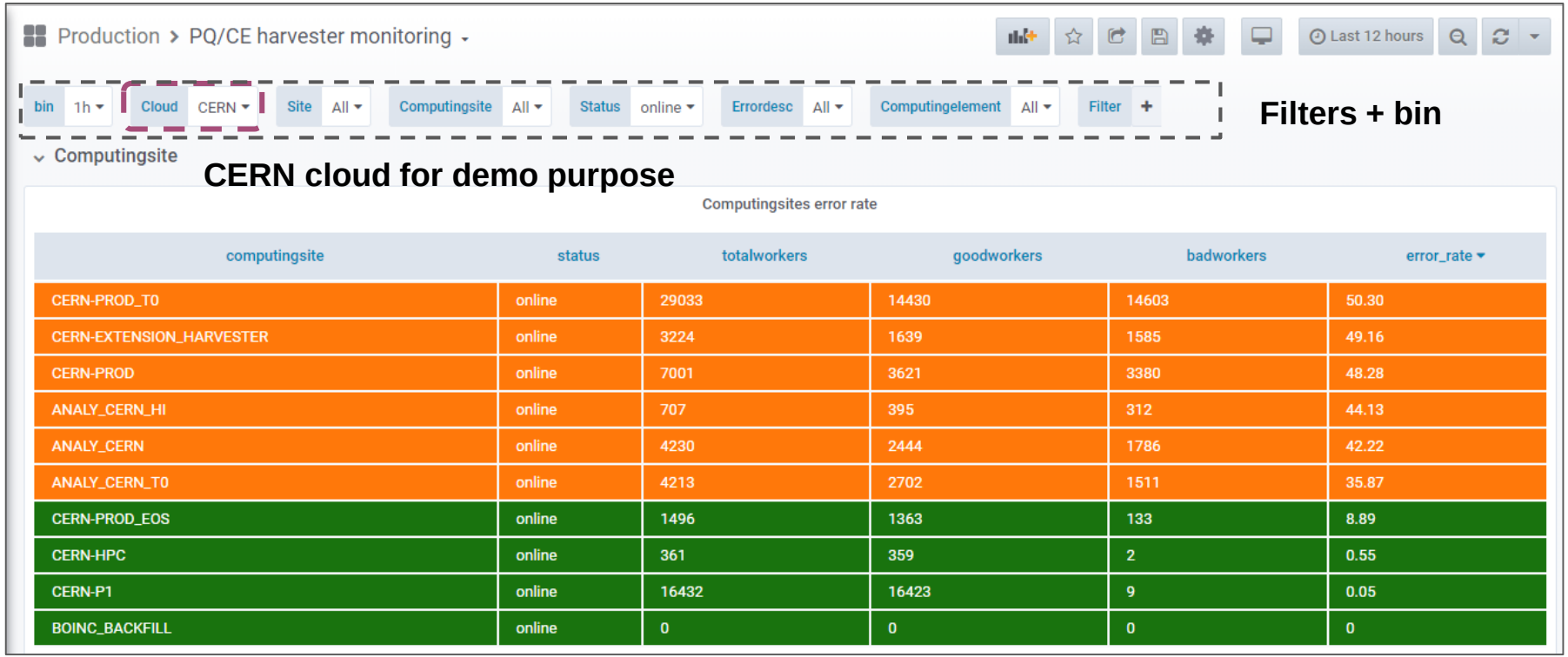
Module for analyzing workers

Statistics for:
PanDA queues
CEs
Error messages

monit-grafana.cern.ch

PQ/CE harvester monitoring dashboard. Submission stats table

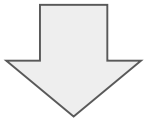
- “Good workers” in finished status
- “Bad workers” in failed, cancelled, missed statuses



Computingsites error rate

computingsite	status	totalworkers	goodworkers	badworkers	error_rate
CERN-PROD_T0	online	29033	14430	14603	50.30
CERN-EXTENSION_HARVESTER	online	3224	1639	1585	49.16
CERN-PROD	online	7001	3621	3380	48.28
ANALY_CERN_HI	online	707	395	312	44.13
ANALY_CERN	online	4230	2444	1786	42.22
ANALY_CERN_T0	online	4213	2702	1511	35.87
CERN-PROD_EOS	online	1496	1363	133	8.89
CERN-HPC	online	361	359	2	0.55
CERN-P1	online	16432	16423	9	0.05
BOINC_BACKFILL	online	0	0	0	0

PQ/CE harvester monitoring. Error messages



Computingsites errors list

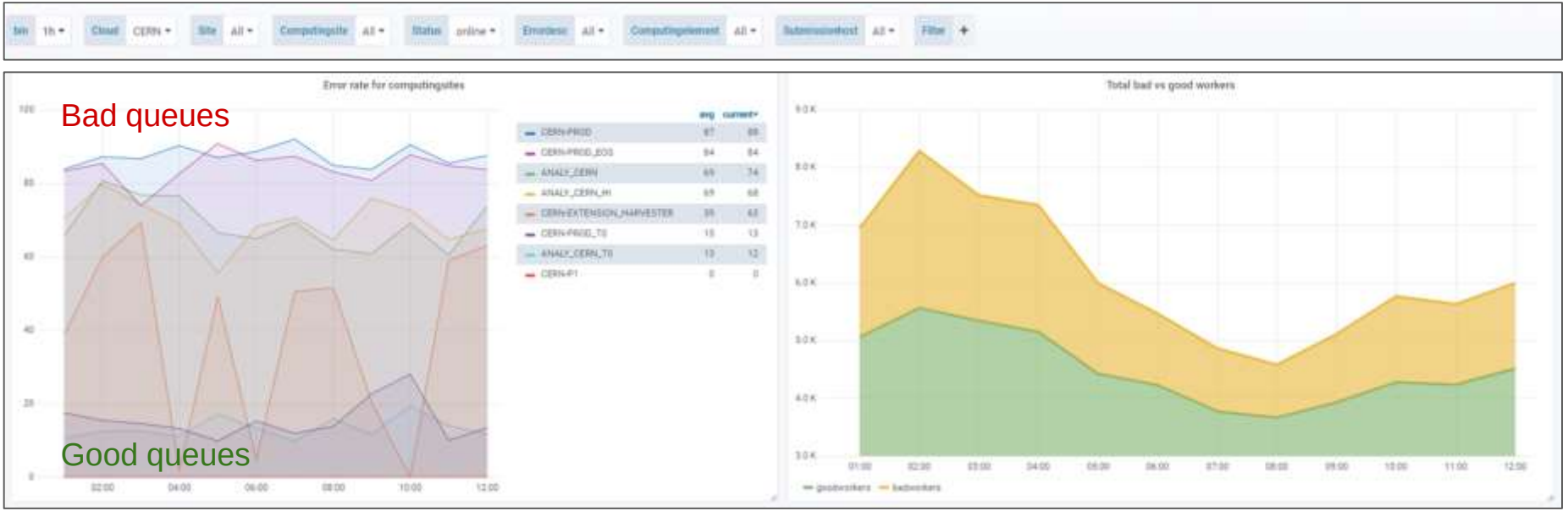
computingsite	errordiag	ratio_computingsite	count	ratio_error
ANALY_CERN	Condor HoldReason: None ; Condor RemoveReason: The system macro SYSTEM_PERIODIC_REMOVE expression '((NumJobStarts >= 1 && JobStatus == 1) || (NumJobStarts > 1 && JobStatus == 2)) || ((JobRunCount >= 1 && JobStatus == 1) || (JobRunCount > 1 && JobStatus ==	73.02	5782	99.16
CERN-PROD	Condor HoldReason: None ; Condor RemoveReason: The system macro SYSTEM_PERIODIC_REMOVE expression '((NumJobStarts >= 1 && JobStatus == 1) || (NumJobStarts > 1 && JobStatus == 2)) || ((JobRunCount >= 1 && JobStatus == 1) || (JobRunCount > 1 && JobStatus ==	87.35	5289	99.83
CERN-PROD_TO	Condor HoldReason: None ; Condor RemoveReason: The system macro SYSTEM_PERIODIC_REMOVE expression '((NumJobStarts >= 1 && JobStatus == 1) || (NumJobStarts > 1 && JobStatus == 2)) || ((JobRunCount >= 1 && JobStatus == 1) || (JobRunCount > 1 && JobStatus ==	15.09	2478	89.46
CERN-PROD_EOS	Condor HoldReason: None ; Condor RemoveReason: The system macro SYSTEM_PERIODIC_REMOVE expression '((NumJobStarts >= 1 && JobStatus == 1) || (NumJobStarts > 1 && JobStatus == 2)) || ((JobRunCount >= 1 && JobStatus == 1) || (JobRunCount > 1 && JobStatus ==	84.82	1887	99.89
ANALY_CERN_TO	Condor HoldReason: None ; Condor RemoveReason: The system macro SYSTEM_PERIODIC_REMOVE expression '((NumJobStarts >= 1 && JobStatus == 1) || (NumJobStarts > 1 && JobStatus == 2)) || ((JobRunCount >= 1 && JobStatus == 1) || (JobRunCount > 1 && JobStatus ==	14.16	1307	96.74
CERN-EXTENSION_HARVESTER	submission failed: Exception OSError: [Errno 28] No space left on device		789	99.12
ANALY_CERN_HI	Condor HoldReason: None ; Condor RemoveReason: The system macro SYSTEM_PERIODIC_REMOVE expression '((NumJobStarts >= 1 && JobStatus == 1) || (NumJobStarts > 1 && JobStatus == 2)) || ((JobRunCount >= 1 && JobStatus == 1) || (JobRunCount > 1 && JobStatus ==	68.16	730	100.00
CERN-PROD_TO	Condor HoldReason: HTCondor-CE held job due to no matching routes, route job limit, or route failure threshold; see 'HTCondor-CE Troubleshooting Guide' ; Worker canceled by harvester due to held too long or not found	21.26	291	33.37
ANALY_CERN	Condor HoldReason: Network error talking to schedd, probably an authorization failure ; Worker canceled by harvester due to held too long or not found	73.66	47	0.86

CERN-EXTENSION_HARVESTER: No space left on device

PQ/CE harvester monitoring. Submission rate history

Error rate for computing sites

Total bad vs good workers



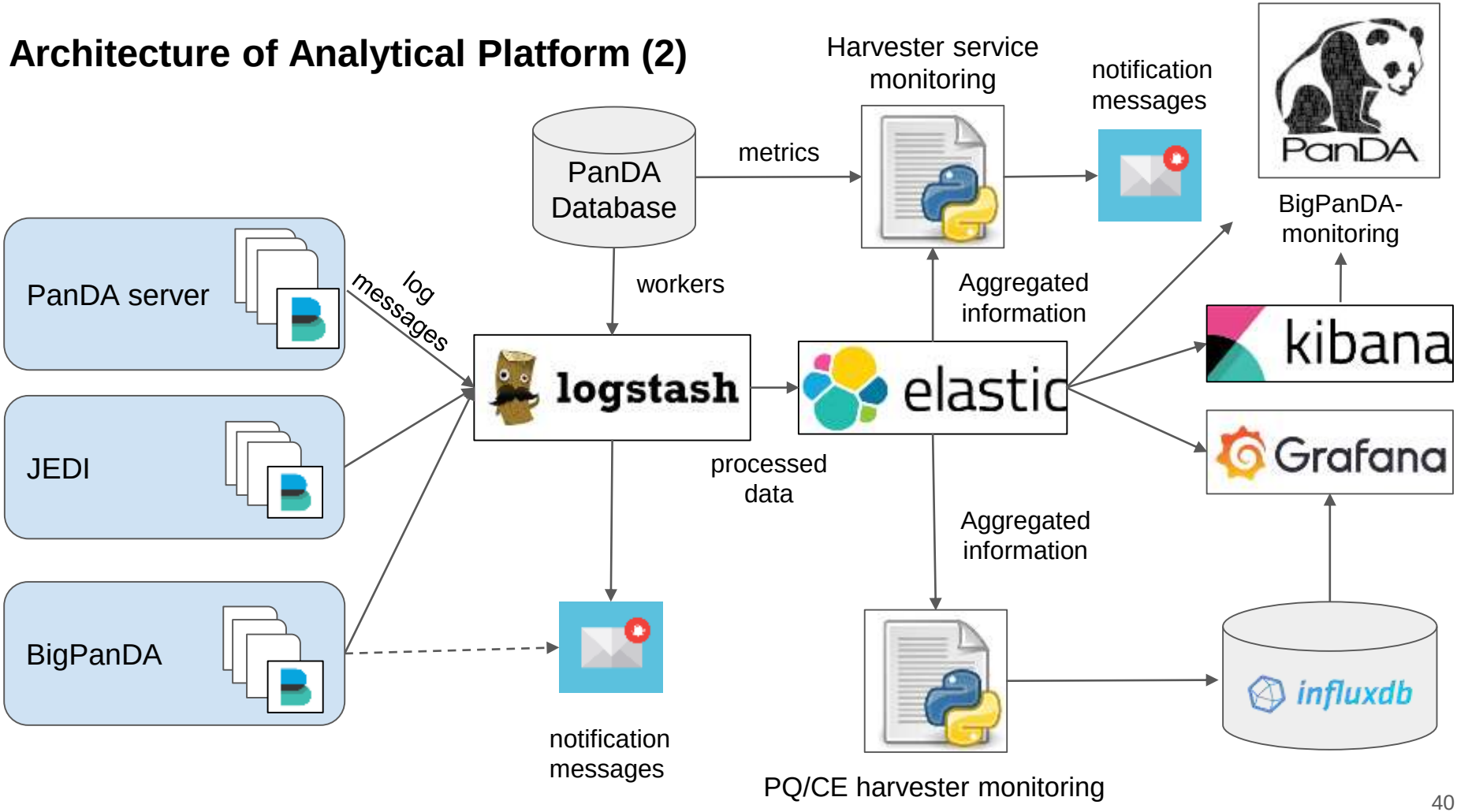
PQ/CE harvester monitoring. Errors patterns

- Regex-based patterns
- InfluxDB statistics contain pattern for better grouping
- Clustering of these errors is planned

August 15th 2019, 10:44:28.000	(74449692) not submitted due to incomplete data of the worker
August 15th 2019, 10:44:28.000	(74449688) not submitted due to incomplete data of the worker
August 15th 2019, 10:44:28.000	(74449693) not submitted due to incomplete data of the worker
August 15th 2019, 10:44:28.000	(74449691) not submitted due to incomplete data of the worker
August 15th 2019, 10:44:28.000	(74449690) not submitted due to incomplete data of the worker
August 15th 2019, 10:44:28.000	(74449687) not submitted due to incomplete data of the worker
August 15th 2019, 10:44:28.000	(74449694) not submitted due to incomplete data of the worker

computingsite	errordesc	ratio_computingsite	count ▾	ratio_error
RRC-KI-T1_TEST	(.*?) not submitted due to incomplete data of the worker	100.00	1958	100.00

Architecture of Analytical Platform (2)



Summary

- Work on the analytical platform based on ELK-stack was launched in 2016
- The platform collects and processes ~ 1 TB of metadata per day from two data sources:
  - PanDA database
  - Log files
- Processed metadata is stored in ElasticSearch cluster. This information is used for the visualizations and dashboards in Kibana and Grafana which provide only useful for analyzing information for different groups of users (PanDA developers, Shifters and etc.)
- Now this platform is used to monitor a critical for ATLAS collaboration components:
  - Panda server
  - JEDI
  - BigPanDA monitor
  - Harvester

Acknowledgements:

This work was partially funded by the Russian Science Foundation under contract No.19-71-30008 (research is conducted in Plekhanov Russian University of Economics)