Software contribution from MIPT: Development of software systems for BM@N

Olga Nemova < *olyanemova36@gmail.com* >

the MIPT team for the BM@N collaboration

Main Projects Summary

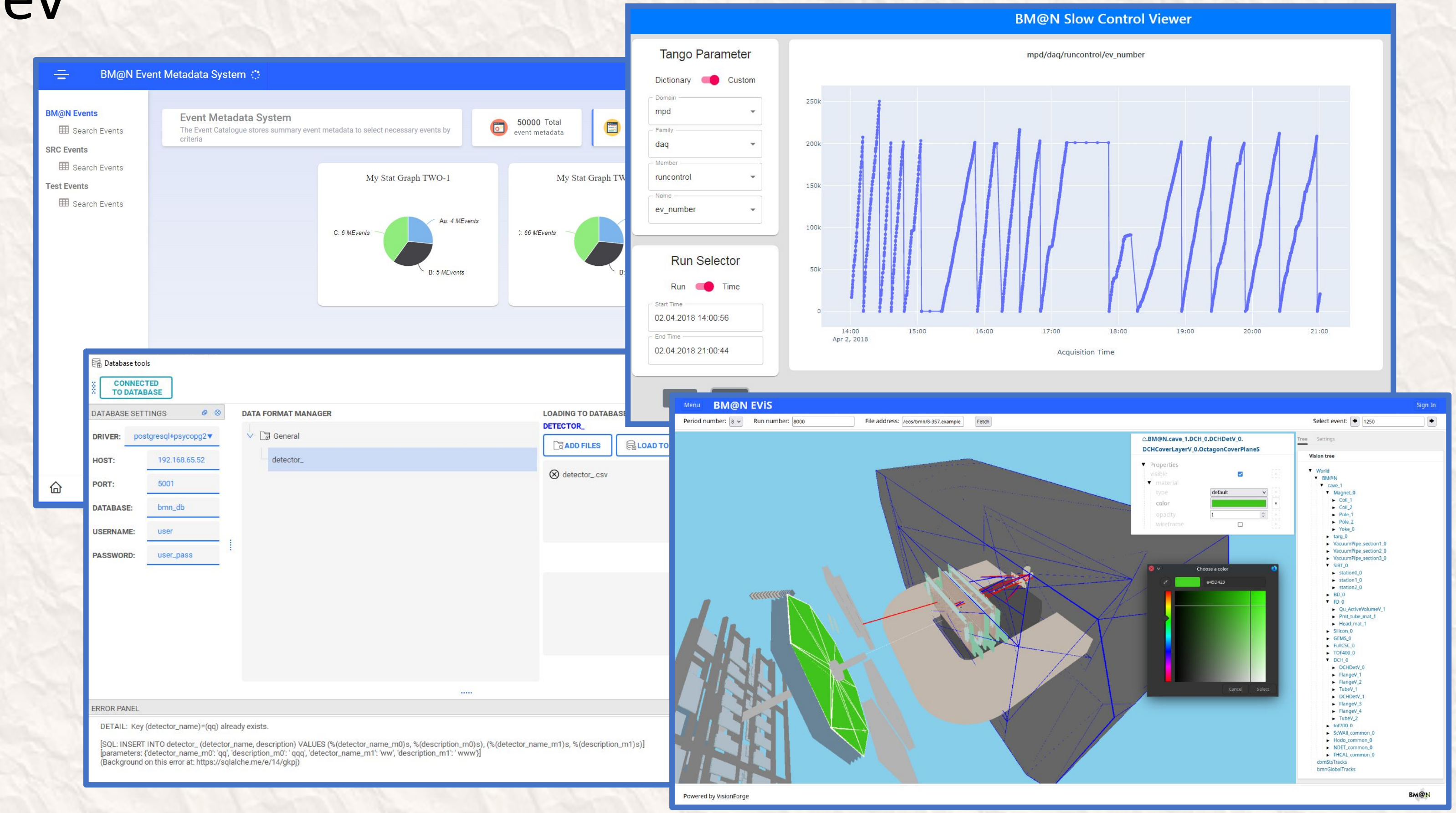
Project	URL
Event Metadata System	https://git.jinr.ru/nica_db/emd https://git.jinr.ru/pklimai/ems-stat-collector https://git.jinr.ru/pklimai/ems-deploy
Next-generation Event Display	https://git.jinr.ru/idunaev/visionforge https://git.jinr.ru/pklimai/visapi
Monitoring Service	https://git.jinr.ru/pklimai/mon-service-deploy
Development of REST API service for slow control system	WIP

MIPT Software for BM@N – Team

Supervision: T. A.-Kh. Aushev

Team members:

- P. Klimai
- A. Nozik
- O. Nemova (student 5y)
- I. Dunaev (student 5y)
- S. Efimov (student 6y)
- A. Degtyarev (PhD st. 1y)



The Event Catalogue stores summary event metadata to select necessary events by criteria

Domain: mpd
Family: daq
Member: runcontrol
Name: ev_number

Start Time: 02.04.2018 14:00:56
End Time: 02.04.2018 21:00:44

DRIVER: postgresql+psycopg2
HOST: 192.168.65.52
PORT: 5001
DATABASE: bmn_db
USERNAME: user
PASSWORD: user_pass

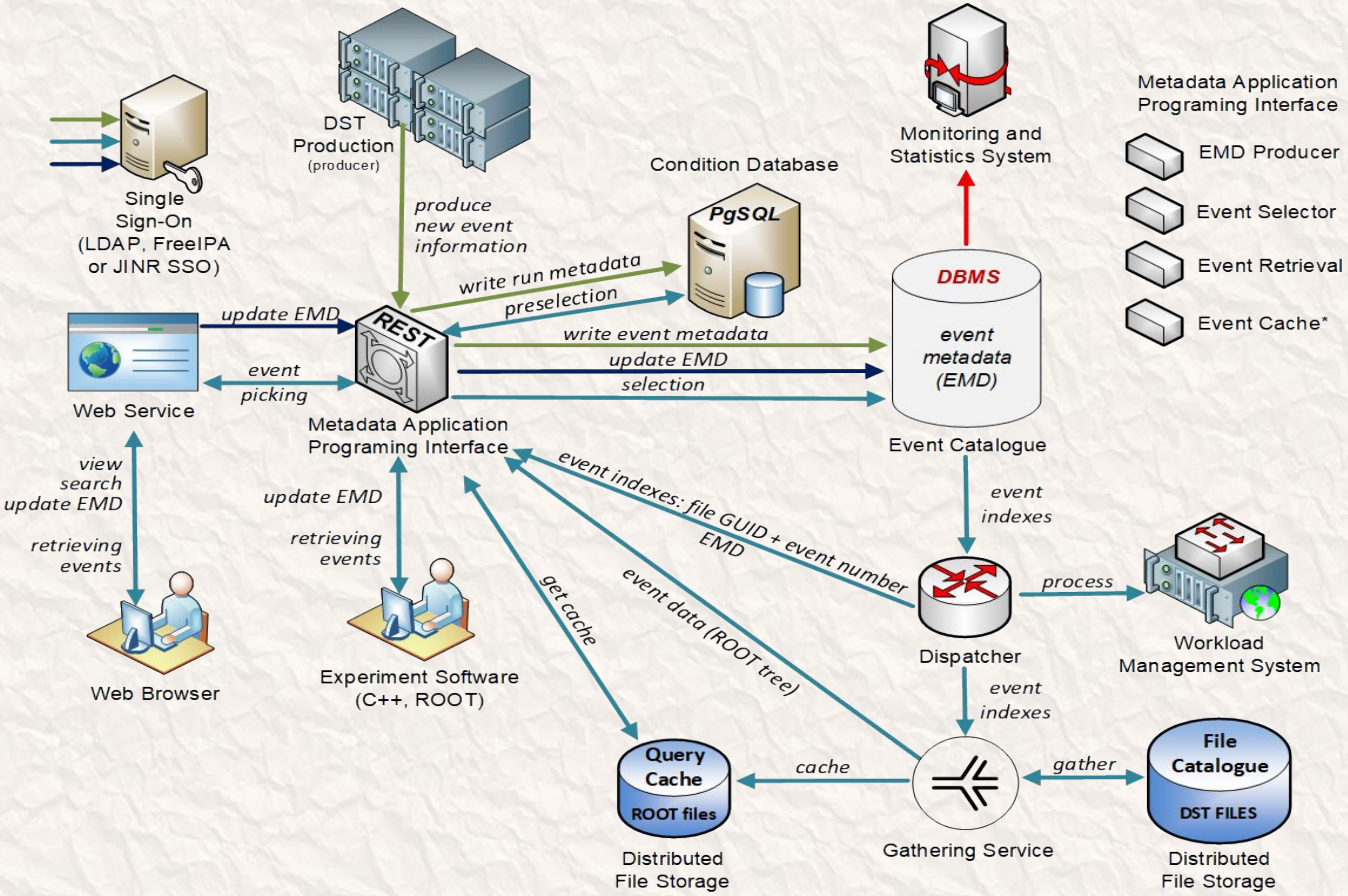
```
DETAIL:  Key (detector_name)=(qq) already exists.

[SQL: INSERT INTO detector_ (detector_name, description) VALUES (%(detector_name_m0)s, %(description_m0)s), (%(detector_name_m1)s, %(description_m1)s)]
[parameters: {'detector_name_m0': 'qq', 'description_m0': ' qqq', 'detector_name_m1': 'ww', 'description_m1': ' www'}]
(Background on this error at: https://sqlalche.me/e/14/gkpj)
```

Period number: 8
Run number: 8000
File address: /eos/bmn/8-357.example
Select event: 1250

BM@N.cave_1.DCH_0.DCHDetV_0.DCHCoverLayerV_0.OctagonCoverPlane5

Event Metadata System
(an update)

BM@N Event Metadata System



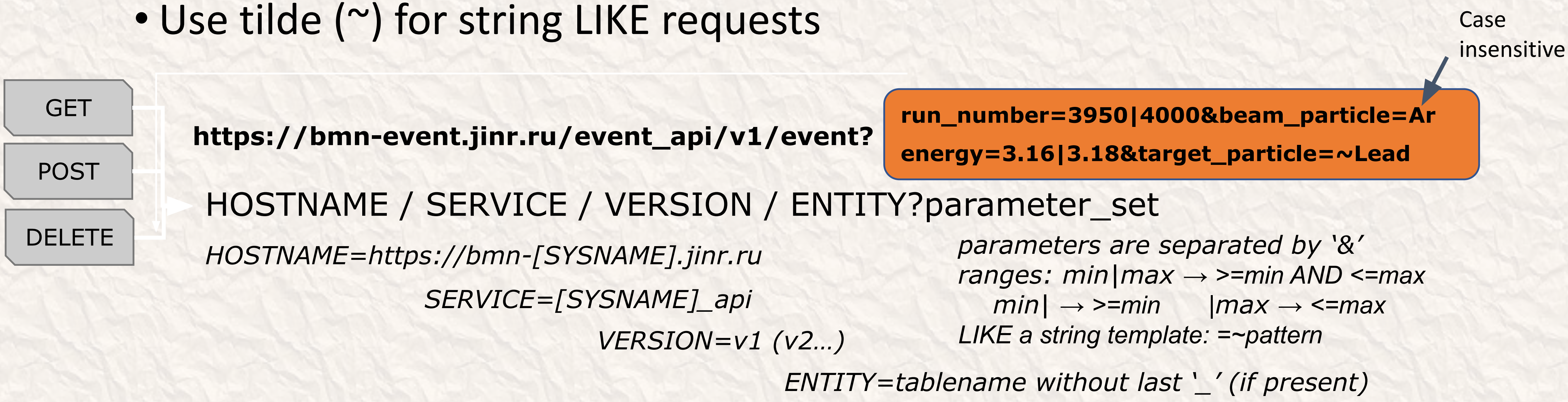
- Event Metadata System
  - Event Catalogue is based on PostgreSQL
  - Integrates with BM@N Condition database
  - REST API and Web UI developed based on Kotlin multiplatform
  - Configurable to support different metadata
  - ROOT macro to write BM@N events in the catalogue
  - Role-based access control implemented
  - Monitoring

For more details:

E. Alexandrov, I. Alexandrov, A. Chebotov, A. Degtyarev, I. Filozova, K. Gertsenberger, P. Klimai and A. Yakovlev, "Implementation of the Event Metadata System for physics analysis in the NICA experiments", J. Phys.: Conf. Ser. 2438, 012046 (2023).

Updated REST API scheme for EMS

- The new scheme is unified for different BM@N Information Systems
  - Use pipe (|) for ranges
  - Use tilde (~) for string LIKE requests



GET

POST

DELETE

Case insensitive

https://bmn-event.jinr.ru/event_api/v1/event?

run_number=3950|4000&beam_particle=Ar
energy=3.16|3.18&target_particle=~Lead

HOSTNAME / SERVICE / VERSION / ENTITY?parameter_set

HOSTNAME=https://bmn-[SYSNAME].jinr.ru

SERVICE=[SYSNAME]_api

VERSION=v1 (v2…)

ENTITY=tablename without last '_' (if present)

parameters are separated by '&'
ranges: min|max → >=min AND <=max
min| → >=min |max → <=max
LIKE a string template: =~pattern

For the Unified Condition Database (UniConDa), SYSNAME = uniconda

For the Event Metadata System (EMS), SYSNAME = event

For Geometry Database, SYSNAME = geo

KeyCloak Integration

- Authentication and authorization in EMS
  - FreeIPA / LDAP support has been dropped
  - KeyCloak token-based authentication and authorization is now supported
  - Database-based authentication is supported as before

```
keycloak_auth:
  server_url: "https://bmn-user.jinr.ru"
  realm: "BMN"
  client_id: "emd_api"
  client_secret: "*****"
  writer_group_name: "bmneventwriter"
  admin_group_name: "bmneventadmin"

# database_auth: True
```

Development of Next-Generation Event Visualization Platform for BM@N (an update)

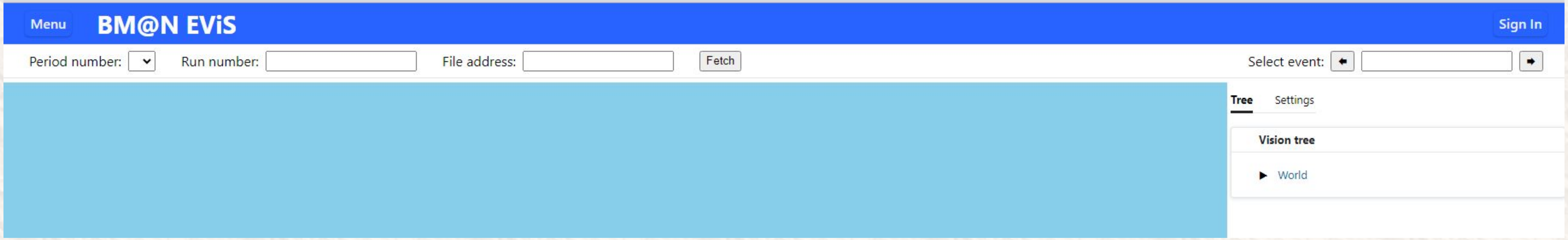
VisionForge Project Overview

- VisionForge – platform for creating next-gen visualization systems
  - Distributed dynamic system
    - Visualization model can be created on one node, transferred to another node and rendered there
    - Nodes can exchange **updates** to the model
    - Changing one element or attribute only requires sending this small change
  - Performance and optimizations
    - BM@N geometry model includes more than 400 000 elements
    - Geometry can be defined as **prototype** that is used by a set of objects, in this case rendering is simplified – only required properties can be changed if needed
  - Using Kotlin-Multiplatform

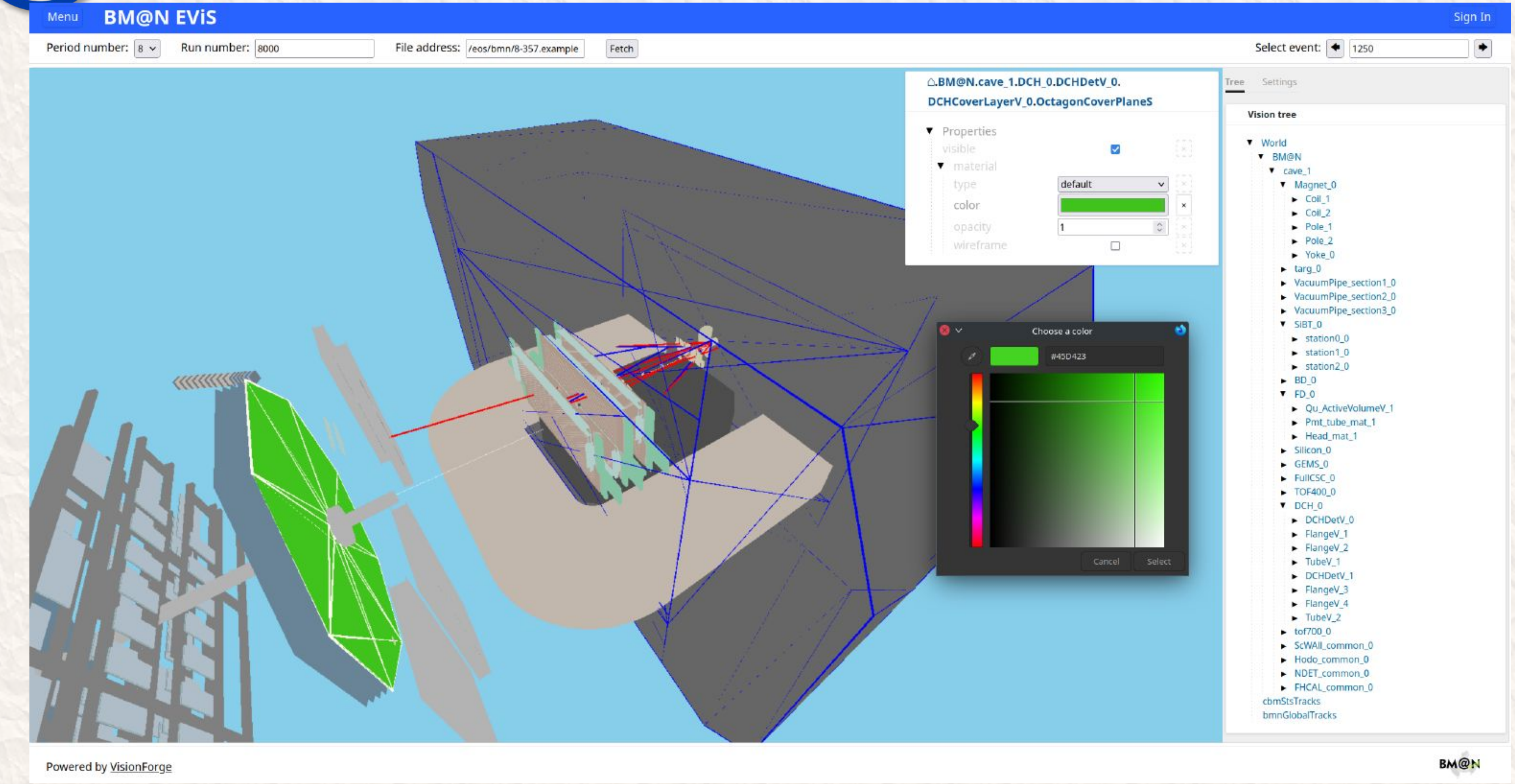
See also: Alexander Nozik — Unbearable lightness of data visualization in Kotlin full stack
https://www.youtube.com/watch?v=uT5j-xOXC3E&ab_channel=JPoint%2CJoker%D0%B8JUGru

Available for test now!

- Available online at http://10.220.16.81:8080/
- Example entry:
  - Period number: **8**
  - Run number: **8000**
  - File address: **/home/lab/events/mpd_run_Top_8000_ev1_p8.root**
  - Select event: **1, 2, 3,…**
- Possible to run it on your own as well (not so simple right now)
- Please send us feedback (contacts on the title slide)!



Geometry, tracks, scene graph, tuning



Period number: 8

Run number: 8000

File address: /eos/bmn/8-357.example

Select event: 1250

⌂.BM@N.cave_1.DCH_0.DCHDetV_0.DCHCoverLayerV_0.OctagonCoverPlaneS

- ▼ Properties
  - visible ☑
  - ▼ material
    - type: default
    - color: 🟩
    - opacity: 1
    - wireframe ☐

Choose a color

#45D423

Vision tree

- ▼ World
  - ▼ BM@N
    - ▼ cave_1
      - ▼ Magnet_0
        - ► Coil_1
        - ► Coil_2
        - ► Pole_1
        - ► Pole_2
        - ► Yoke_0
      - ► targ_0
      - ► VacuumPipe_section1_0
      - ► VacuumPipe_section2_0
      - ► VacuumPipe_section3_0
      - ▼ SiBT_0
        - ► station0_0
        - ► station1_0
        - ► station2_0
      - ► BD_0
      - ▼ FD_0
        - ► Qu_ActiveVolumeV_1
        - ► Pmt_tube_mat_1
        - ► Head_mat_1
      - ► Silicon_0
      - ► GEMS_0
      - ► FullCSC_0
      - ► TOF400_0
      - ▼ DCH_0
        - ► DCHDetV_0
        - ► FlangeV_1
        - ► FlangeV_2
        - ► TubeV_1
        - ► DCHDetV_1
        - ► FlangeV_3
        - ► FlangeV_4
        - ► TubeV_2
      - ► tof700_0
      - ► ScWAll_common_0
      - ► Hodo_common_0
      - ► NDET_common_0
      - ► FHCAL_common_0
  - cbmStsTracks
  - bmnGlobalTracks

Powered by VisionForge

Development of a service for monitoring software systems of the BM@N experiment

BM@N Information systems

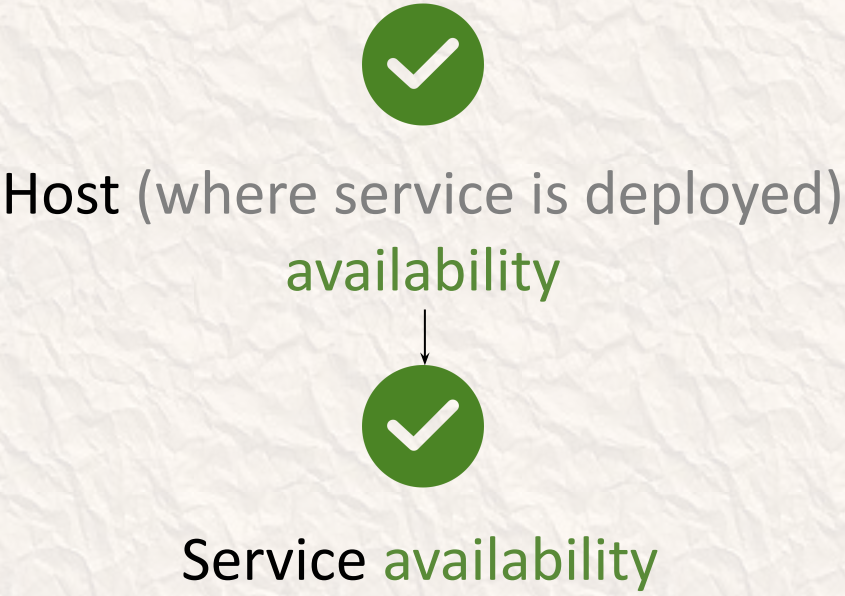
Include:

- Unified Condition database
- Configuration database
- Integrity Inspector
- Electronic LogBook
- and etc.

Monitored parameters

For checking stability and reliability:

- **Endpoints state**:
  - network interfaces,
  - memory,
  - disk,
  - CPU.
- **Database** (e.g. PostgreSQL):
  - latency.
- **Web interfaces**:
  - HTTP requests checks (e.g. GET-request).



✓

Host (where service is deployed) availability

↓

✓

Service availability

Using **TIG** (Telegraf + InfluxDB + Grafana) stack.

Key advantages of the implementation

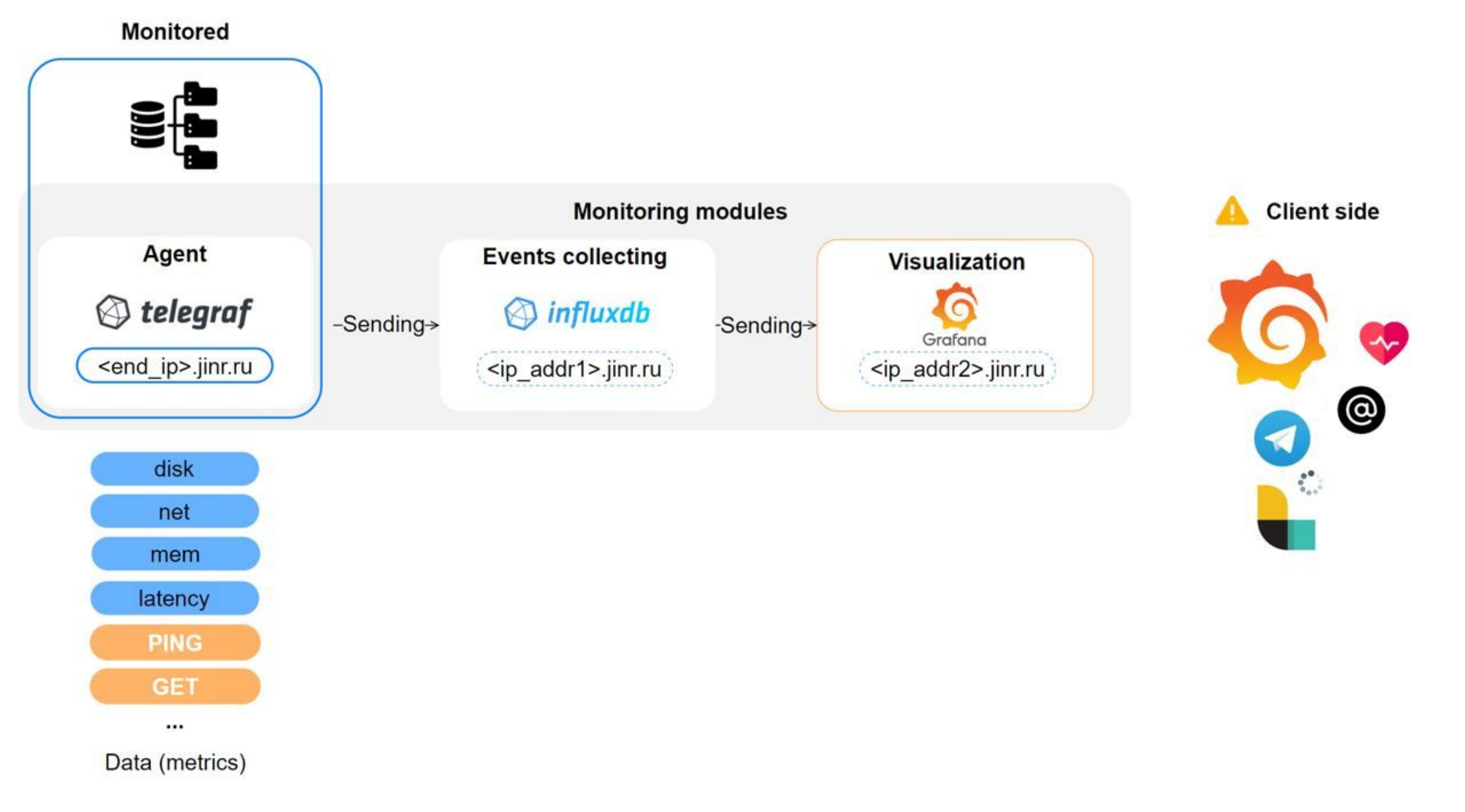
- **Automated** deployment of the **tools** (ex. telegraf) for the monitoring service with **Ansible playbooks**

- **Automated configuration** generation (Jinja2 + JSONs: Alerts and Dashboard)

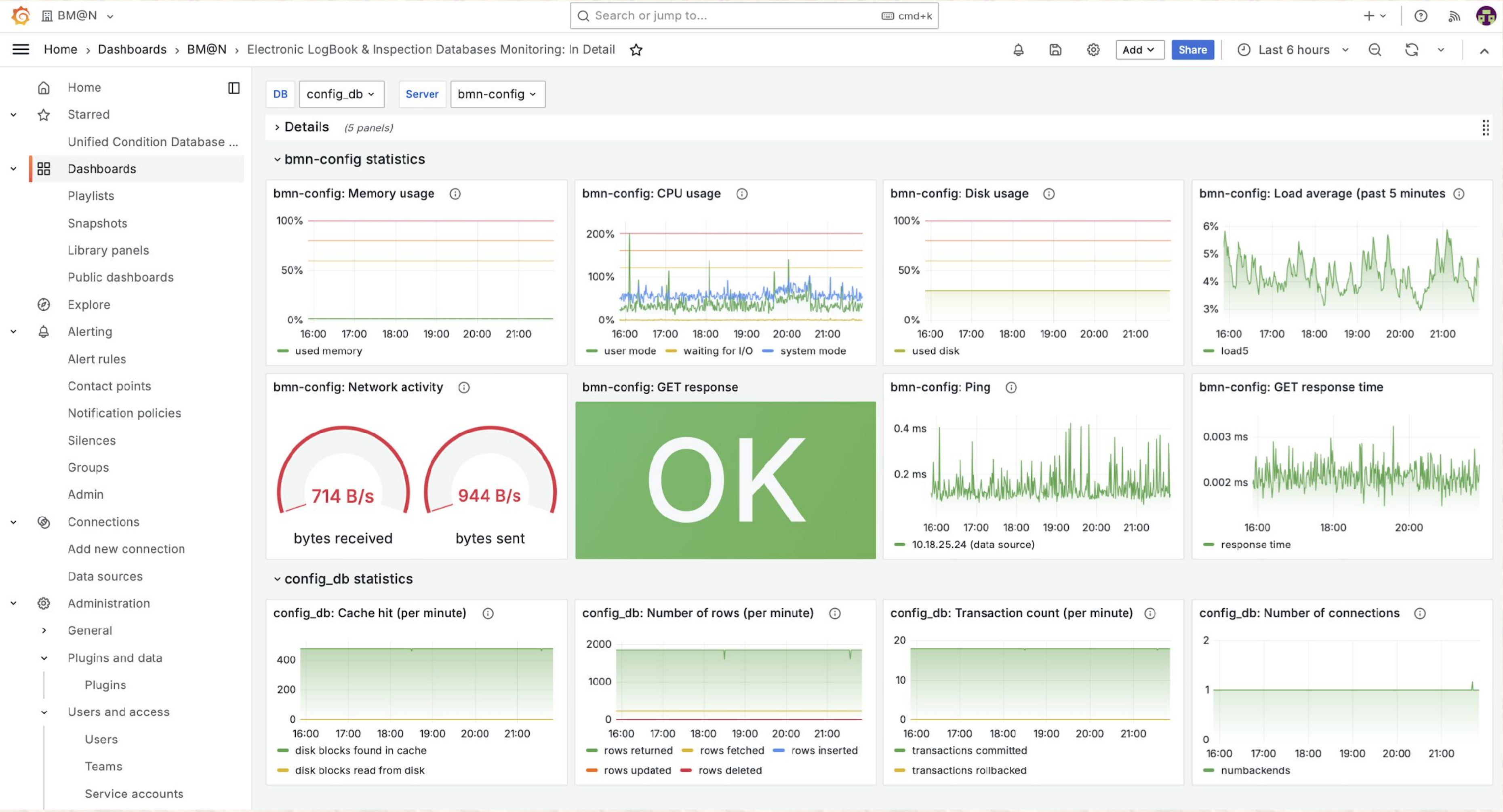
- Ease of scaling because of **module architecture**

- Failure **alerting** with Grafana

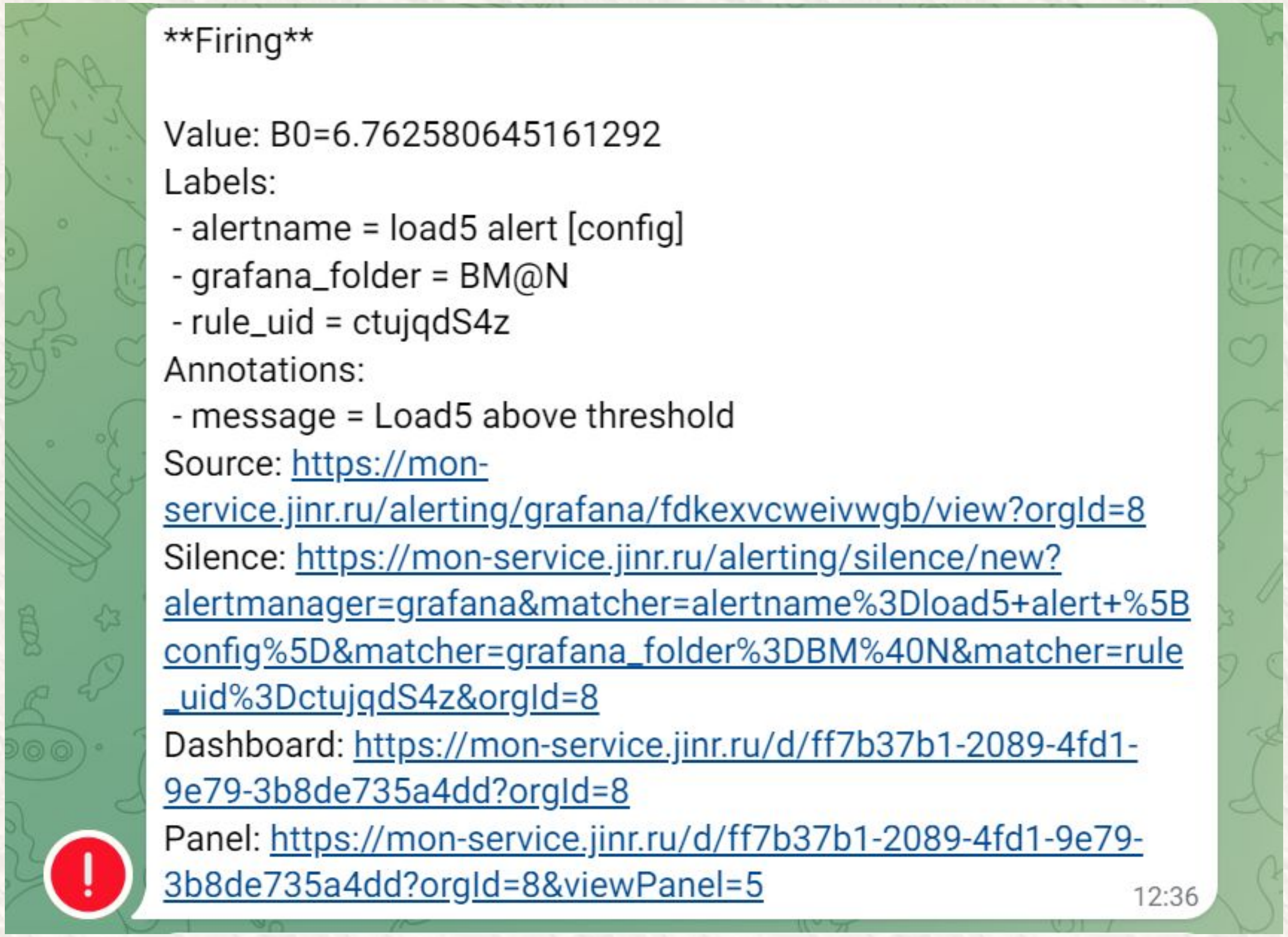
Architecture for **monitoring** of the software systems



BM@N monitoring **client’s view** (Dashboard)

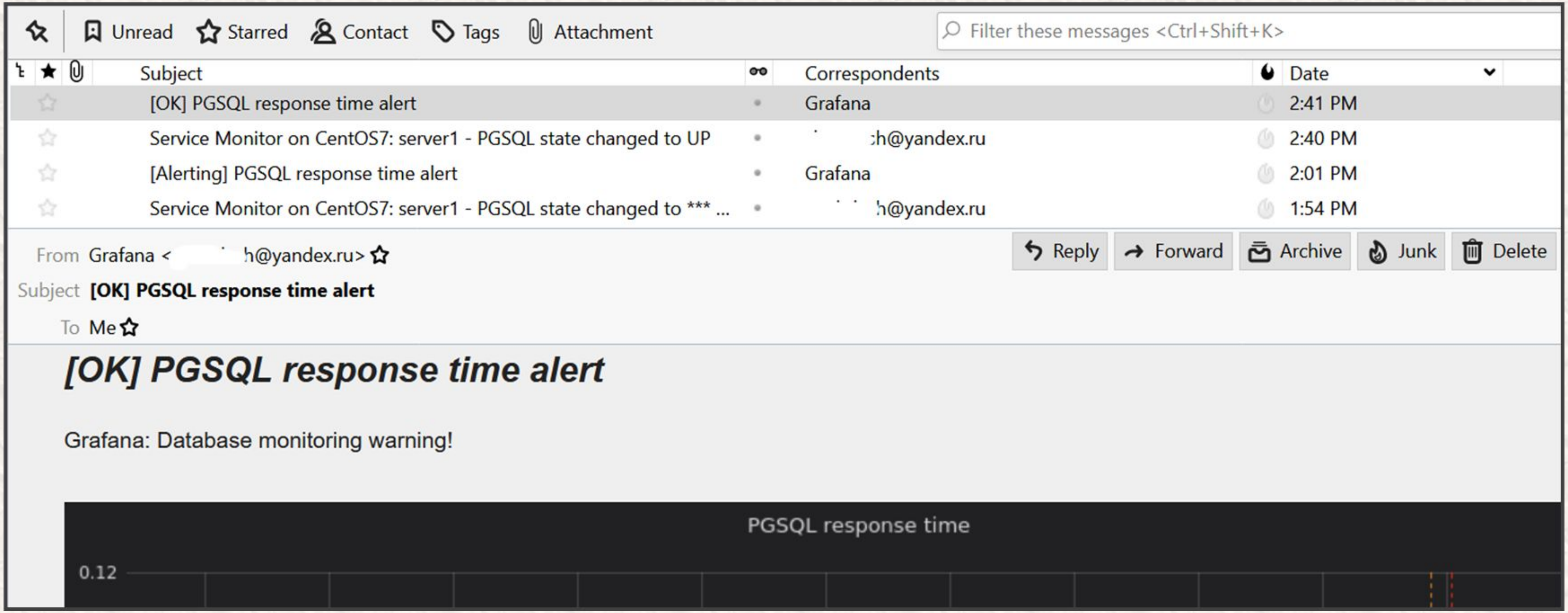


BM@N monitoring **alerting** (Client telegram message)



```
**Firing**

Value: B0=6.762580645161292
Labels:
 - alertname = load5 alert [config]
 - grafana_folder = BM@N
 - rule_uid = ctujqdS4z
Annotations:
 - message = Load5 above threshold
Source: https://mon-service.jinr.ru/alerting/grafana/fdkexvcweivwgb/view?orgId=8
Silence: https://mon-service.jinr.ru/alerting/silence/new?alertmanager=grafana&matcher=alertname%3Dload5+alert+%5Bconfig%5D&matcher=grafana_folder%3DBM%40N&matcher=rule_uid%3DctujqdS4z&orgId=8
Dashboard: https://mon-service.jinr.ru/d/ff7b37b1-2089-4fd1-9e79-3b8de735a4dd?orgId=8
Panel: https://mon-service.jinr.ru/d/ff7b37b1-2089-4fd1-9e79-3b8de735a4dd?orgId=8&viewPanel=5
12:36
```

BM@N monitoring **alerting** (Client email)



Subject	Correspondents	Date
[OK] PGSQL response time alert	Grafana	2:41 PM
Service Monitor on CentOS7: server1 - PGSQL state changed to UP	[redacted]:h@yandex.ru	2:40 PM
[Alerting] PGSQL response time alert	Grafana	2:01 PM
Service Monitor on CentOS7: server1 - PGSQL state changed to *** ...	[redacted]h@yandex.ru	1:54 PM

From Grafana <[redacted]h@yandex.ru>

Subject **[OK] PGSQL response time alert**

To Me

[OK] PGSQL response time alert

Grafana: Database monitoring warning!

Thank You!