BM@N Software Summary

Konstantin Gertsenberger

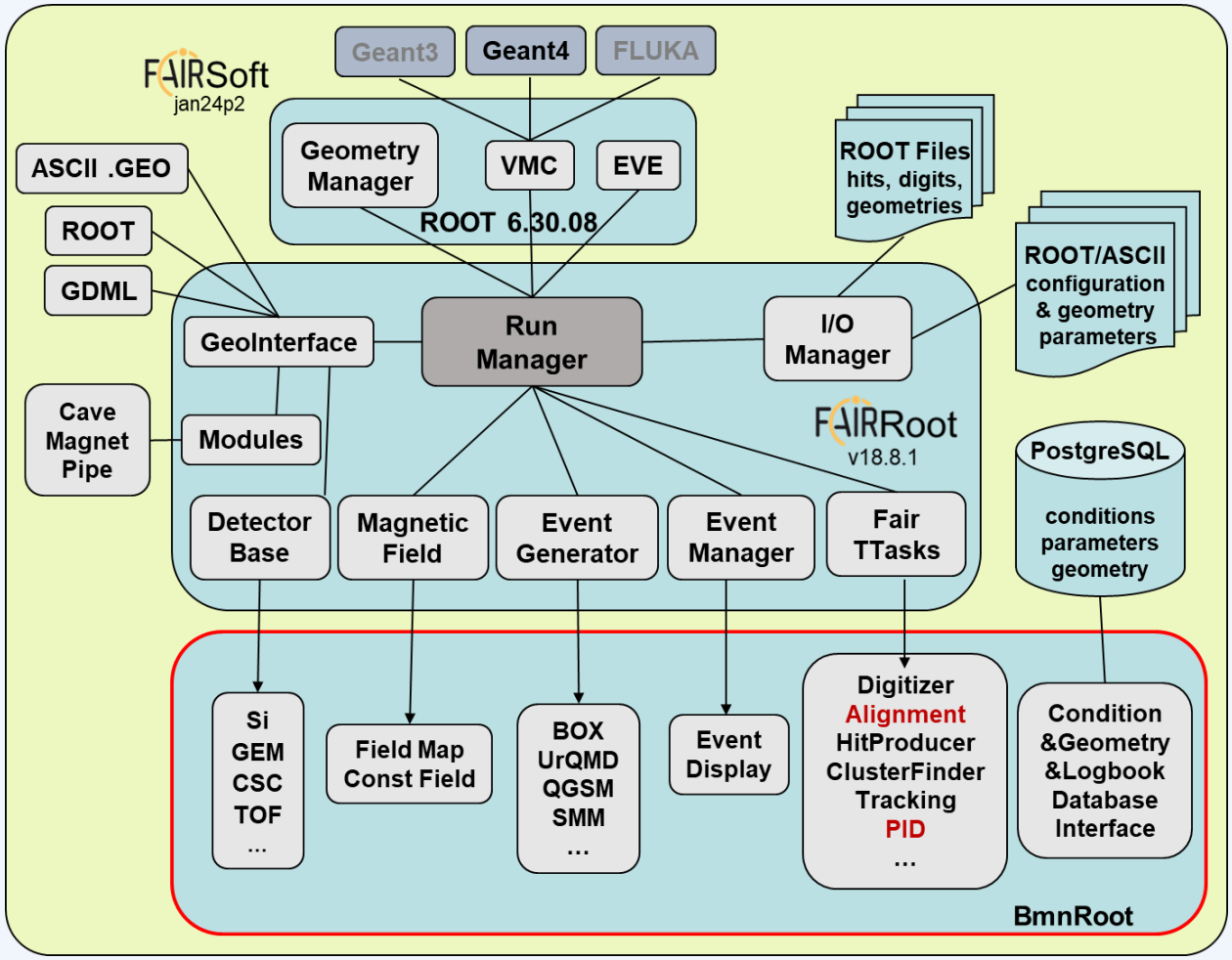
V. Veksler and A. Baldin Laboratory of High Energy Physics
Joint Institute for Nuclear Research

8–10 October 2024

BmnRoot. *Production Tag 24.09.0*

FAIRSoft jan24p2 FAIRRoot v18.8.1

- FairSoft jan24p2 is used now (with ROOT 6.32.08) instead of jan24 version
- FairRoot v19.0.0 is available but requires new compilation approach in BmnRoot



- Issue with the current FairSoft, FairRoot directory (…/bmn/) was corrected on the clusters
- The previous BmnRoot tag 24.04.0 has a serious bug in simulation (with DCH)
- New BmnRoot tag 24.09.0 has been created and is being tested with 10M data production
- *MCInfo* branch is a part of the DST$_{sim}$ format for using main MC data in physics analyses
- Most detector geometries were updated including the GEM geometry that crashes BM@N simulation
- VSP (Vertex Silicon Plane) was added for the next run

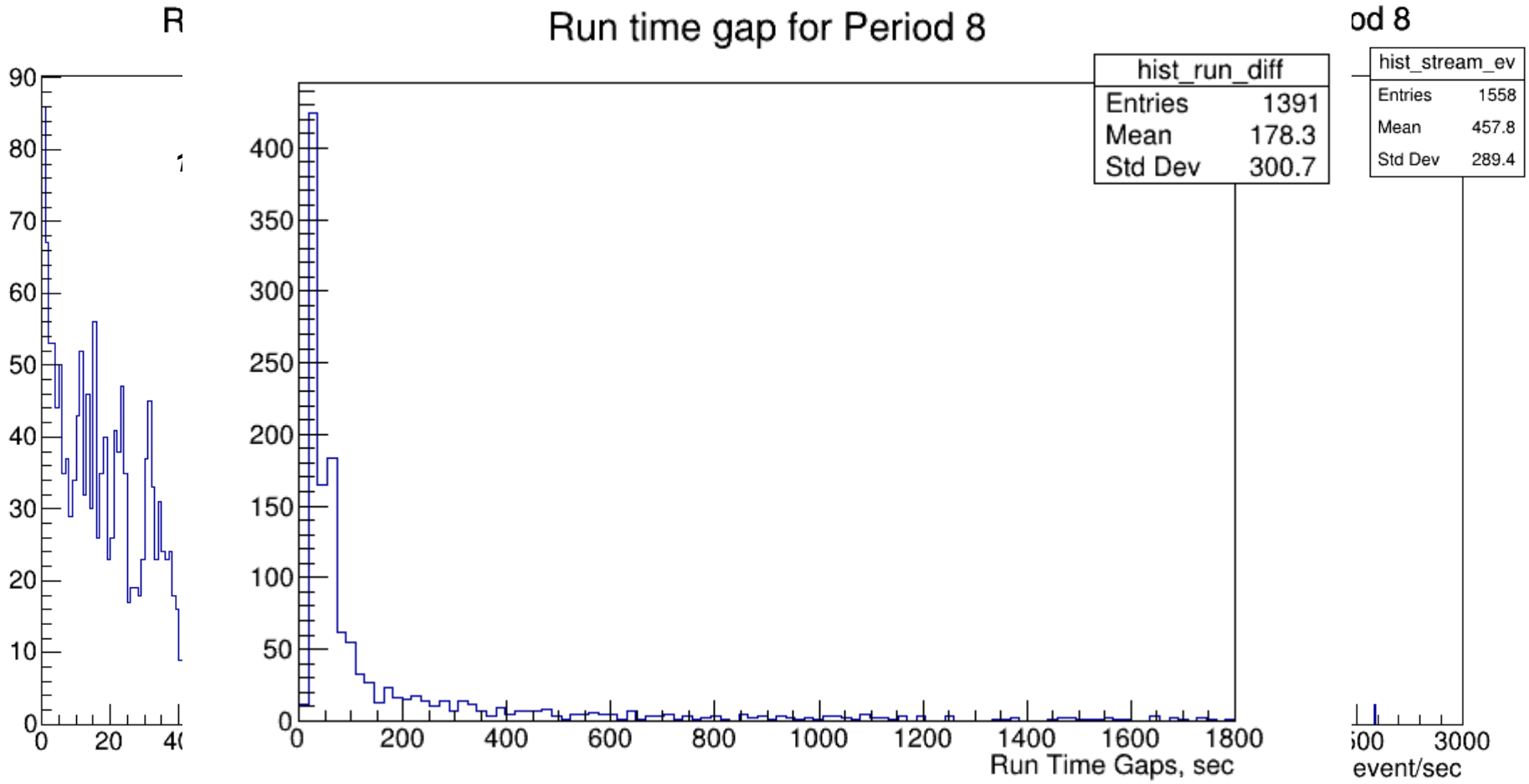
BmnRoot. *Spring Issues*

- ✓ The current BM@N geometry is not passing standard ROOT tests (e.g. there are a lot of overlaps) and breaking both the simulation and reconstruction macro → several overlaps
- ✓ BmnRoot macros must contain common logic and parameters without local input parameters, user paths, hardcoded global parameters:
  - No default input parameters for a concrete user (but for production)
  - No inner logic for own local machine
  - No own local paths and local parameters

  Simple check of all the macros was integrated to the BM@N pipeline (night checks)
- ✓ Strict separation of short summary output (by default) from debug output for individuals (macro output should be adapted for mass production) → solved, but new debug messages *FairLoggger::Severity* (debug level) or *fVerbose* flag (*SetVerbose* function) must be used
- ✓ A lot of compilation warnings (*after the update*) must be corrected → solved, almost all
- ✓ ClassImp() should be removed in the cxx files as obsolete → solved, but new lines (TOF700)
- ✓ SRC information will be removed from the BM@N databases → not done yet
- ✓ Raw Data Decoder should be fixed to write correct run and raw file metadata. New synchronization of the metadata is needed after → metadata reprocessed and synced (e-Log)

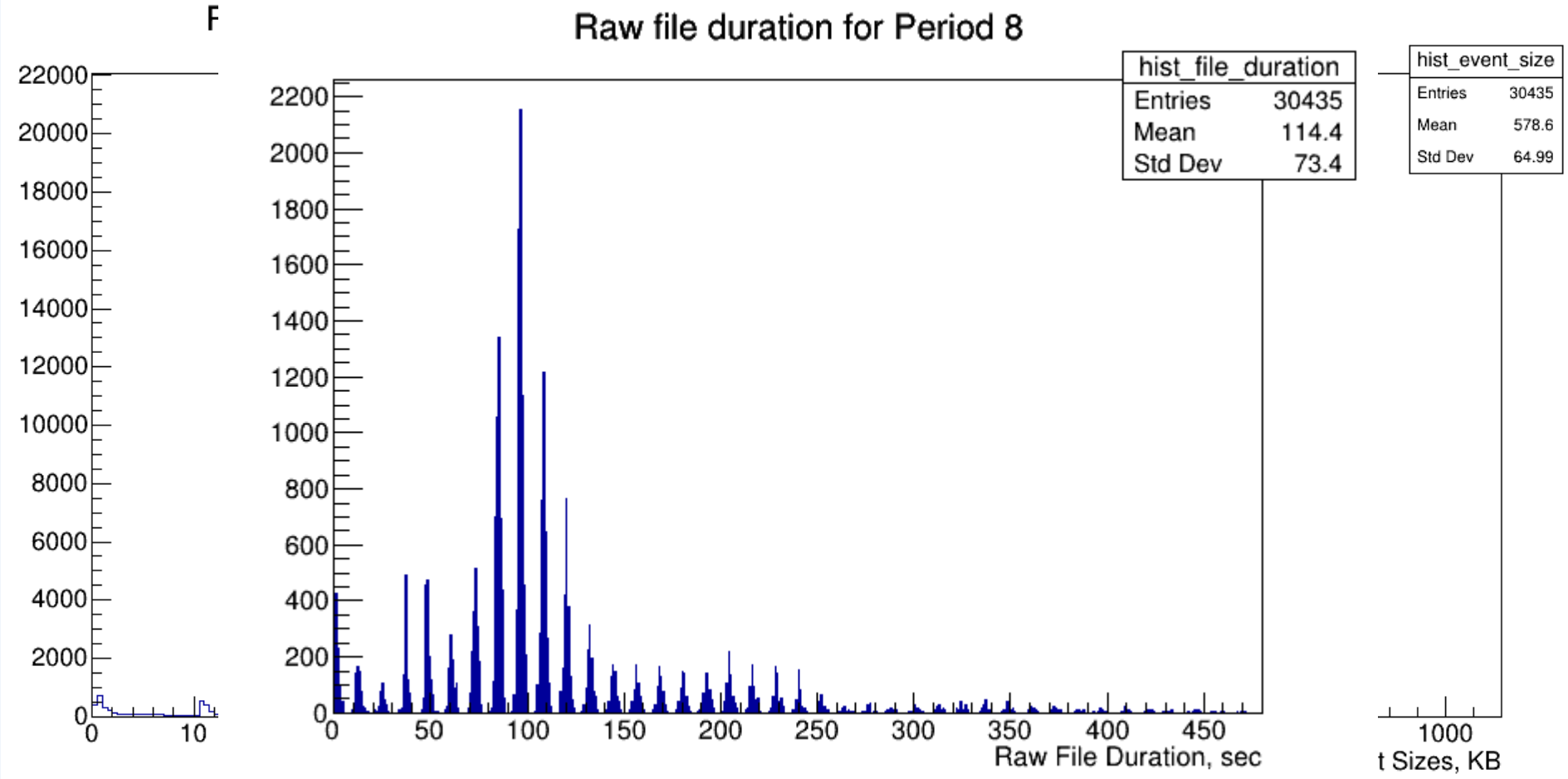
Reprocessed Run Statistics

*root bmnroot/database/uni_db/examples/run/**show_run_stats.C***



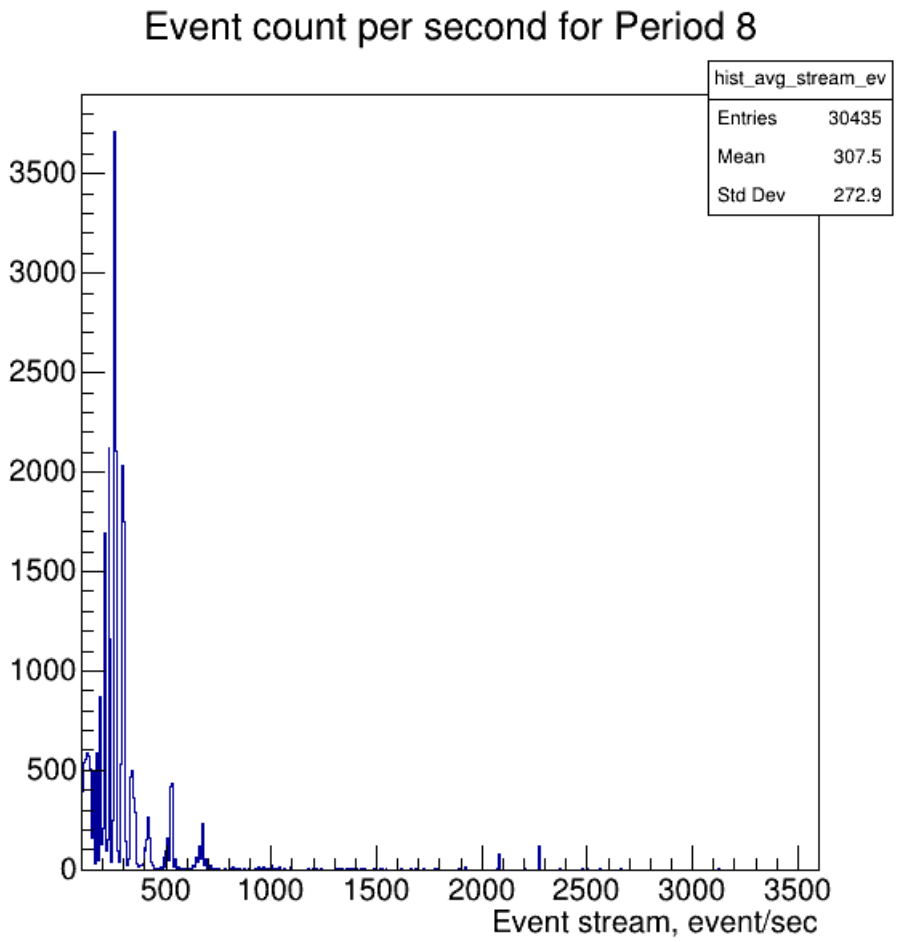
Reprocessed Raw Statistics

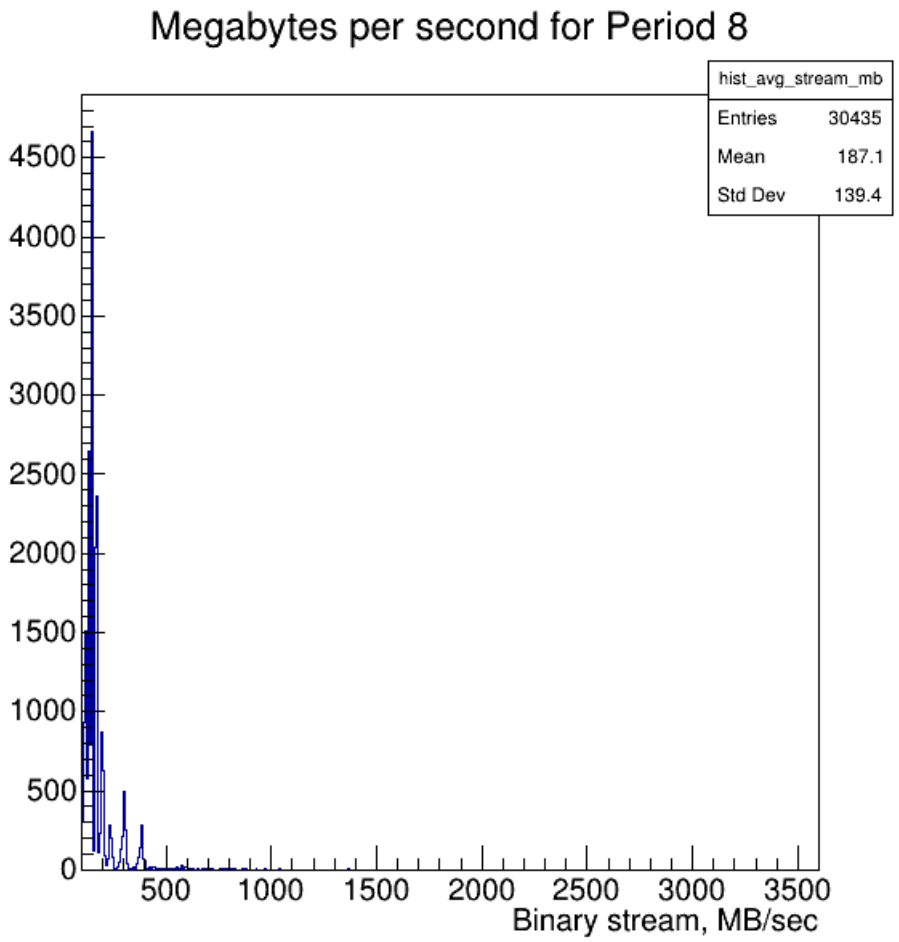
*root bmnroot/database/uni_db/examples/raw/**show_raw_stats.C***



Reprocessed Raw Statistics

*root bmnroot/database/uni_db/examples/raw/**show_raw_stats.C***

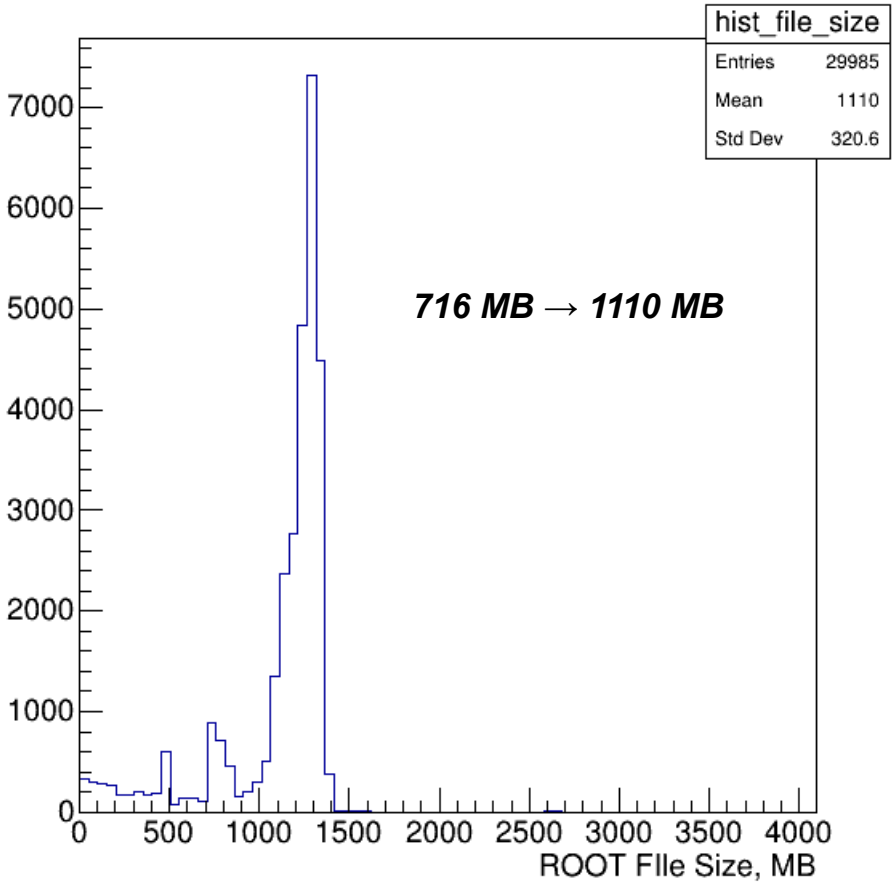




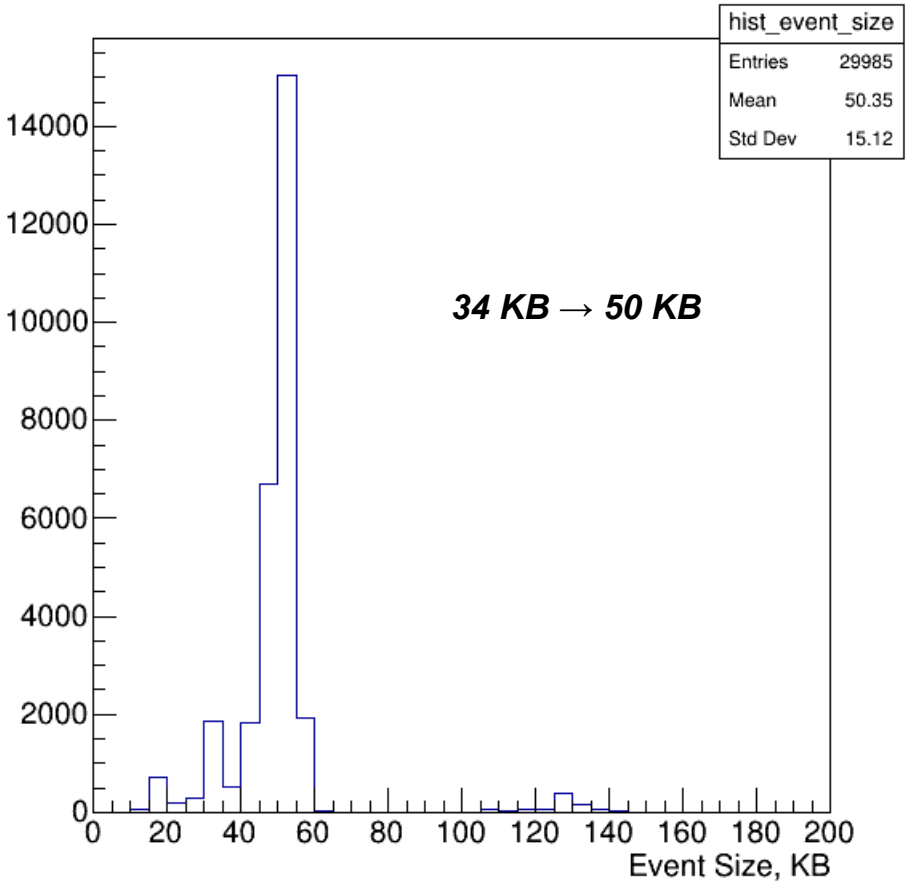
Reprocessed DIGIT Statistics (*24.04.0*)

*root bmnroot/database/uni_db/examples/root/**show_root_stats.C***

DIGIT file size



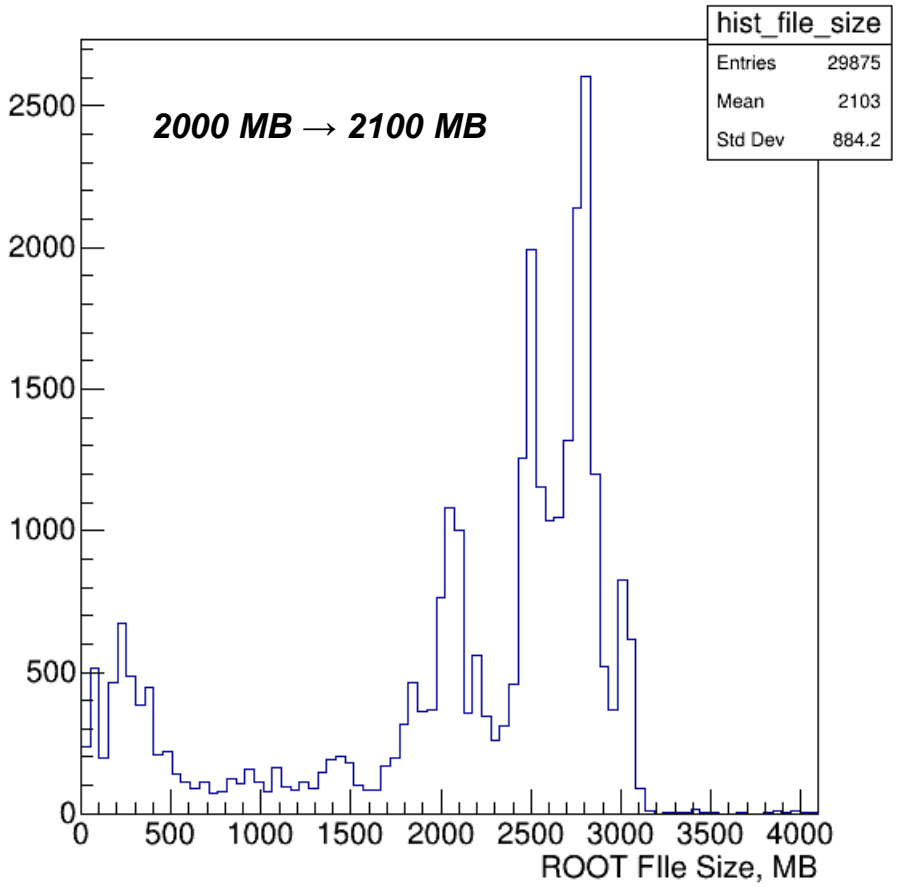
DIGIT event size



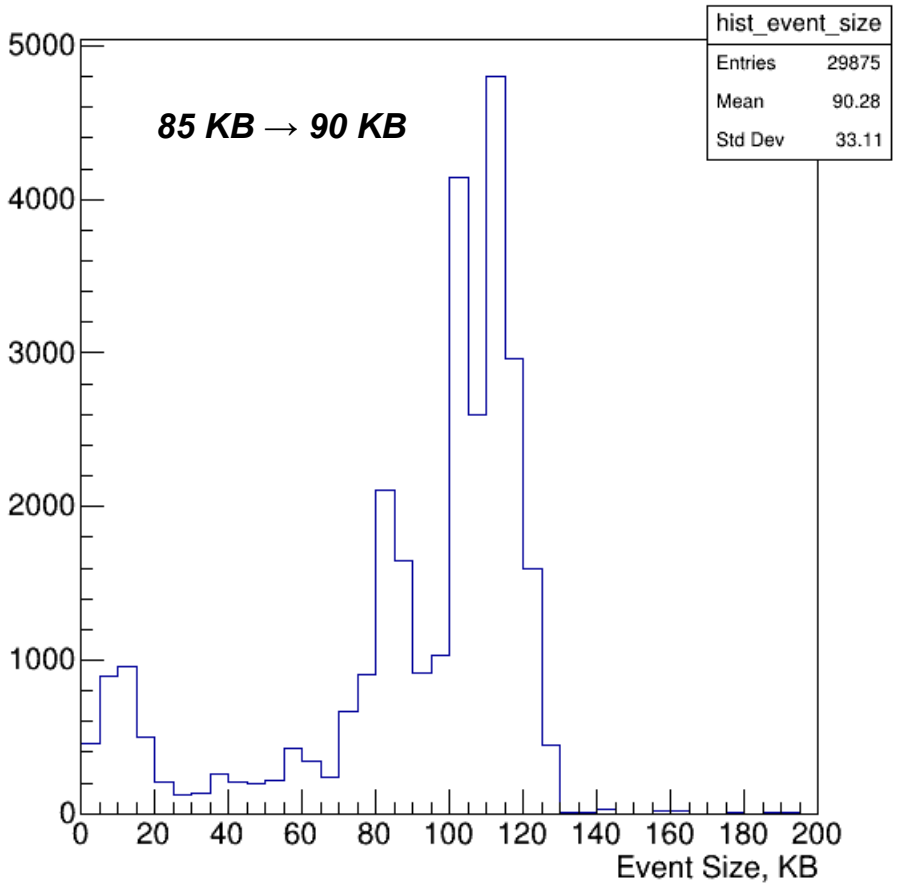
Reprocessed DST Statistics (*24.04.0*)

*root bmnroot/database/uni_db/examples/root/**show_root_stats.C***

DST file size



DST event size



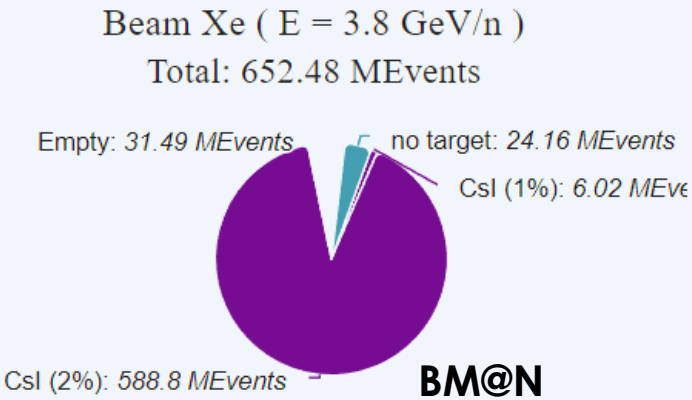
Data Production in BM@N Physics Run

1st Physics BM@N Run

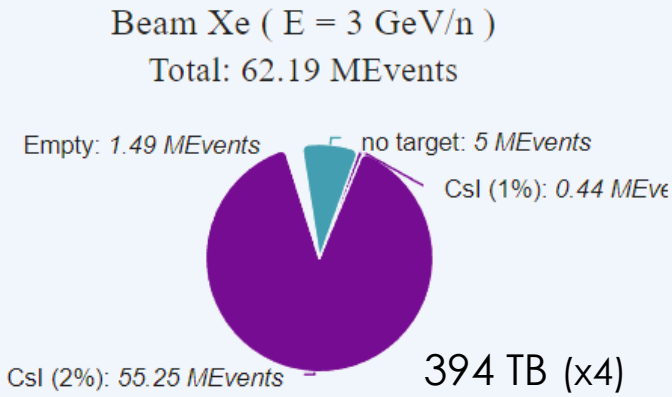
Two beam energy available for *Xe*-beam

CsI target is used as more similar to *Xe*

More than 600M events were collected



BM@N Run 8



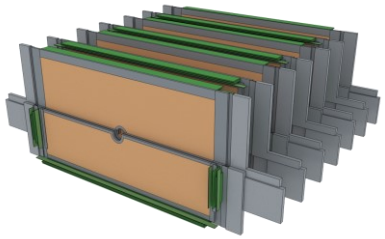
394 TB (x4)

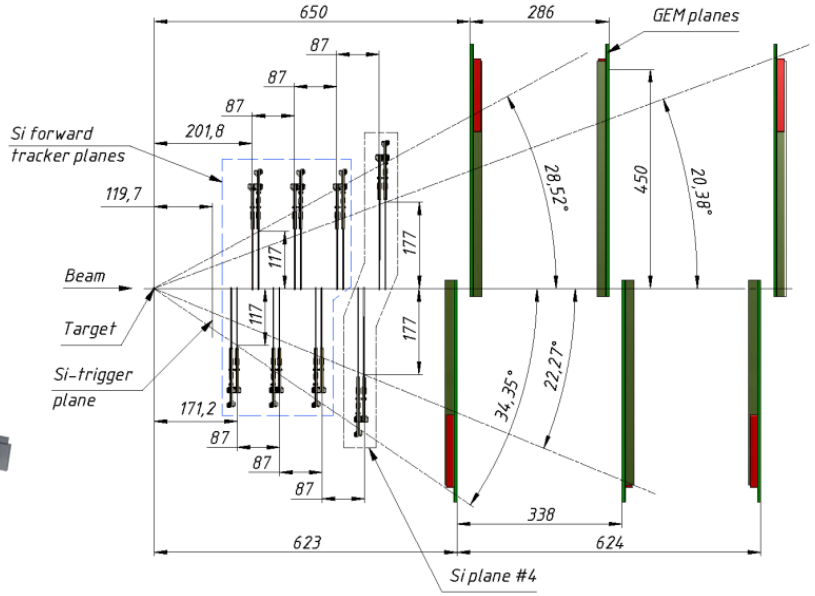
Parameter	Value (approx.)
Data acquisition time	**720 hours**
Average run duration	20 minutes
Average run time break	3 minutes
Beam intensity (3.8 AGeV)	up to 900k/2.2 Xe^+/sec up to 900k/12 Xe^+/sec
Trigger rate	**8 000/2.2 event/sec**
Average event size	**0,57 МБ**
Data rate	**up to 3 GB/sec**
Raw file size	**15 GB (avg.13)**
Event count per file (with pedestals?)	25 000
Total event count (+test, calibration, physics)	**710 M**
Total file count (with tails)	30 753
Total run count	1 706
Total raw data size	**394 TB**
Total replicated raw data	1.6 PB
Avg digit file size	**1 100 MB**
Avg DST file size	**2 100 MB**

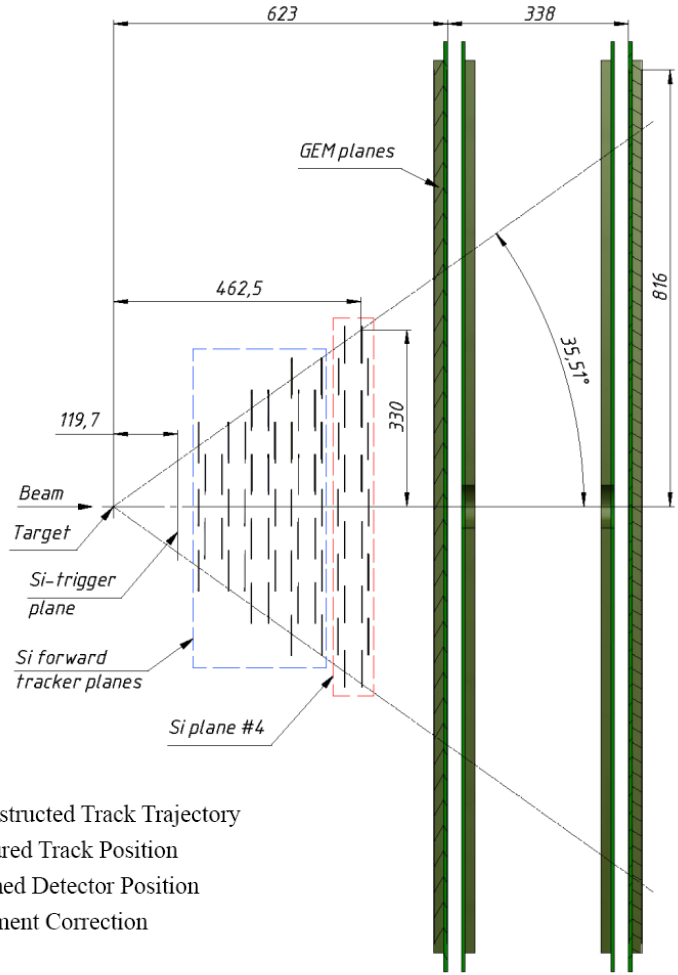
BmnRoot. *Detector Alignment*

Alignment Tools

- IMSL Fortran Library
- Eigen
- Millepede-II





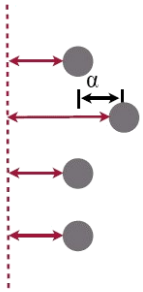


$$\begin{pmatrix} S_2 & 0 & 0 & 0 & 0 \\ 0 & S_2 & 0 & 0 & 0 \\ 0 & 0 & S_2 & 0 & 0 \\ 0 & 0 & 0 & S_2 & 0 \\ 0 & 0 & 0 & 0 & S_2 \end{pmatrix} \begin{pmatrix} S_1 & 0 & 0 & 0 & 0 \\ 0 & S_1 & 0 & 0 & 0 \\ 0 & 0 & S_1 & 0 & 0 \\ 0 & 0 & 0 & S_1 & 0 \\ 0 & 0 & 0 & 0 & S_1 \end{pmatrix} \begin{pmatrix} Z_2 & Z_3 & Z_4 & Z_5 \\ Z_2 & Z_3 & Z_4 & Z_5 \\ Z_2 & Z_3 & Z_4 & Z_5 \\ Z_2 & Z_3 & Z_4 & Z_5 \\ Z_2 & Z_3 & Z_4 & Z_5 \end{pmatrix}$$

$$\begin{pmatrix} S_1 & 0 & 0 & 0 & 0 \\ 0 & S_1 & 0 & 0 & 0 \\ 0 & 0 & S_1 & 0 & 0 \\ 0 & 0 & 0 & S_1 & 0 \\ 0 & 0 & 0 & 0 & S_1 \end{pmatrix} \begin{pmatrix} N_d & 0 & 0 & 0 & 0 \\ 0 & N_d & 0 & 0 & 0 \\ 0 & 0 & N_d & 0 & 0 \\ 0 & 0 & 0 & N_d & 0 \\ 0 & 0 & 0 & 0 & N_d \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

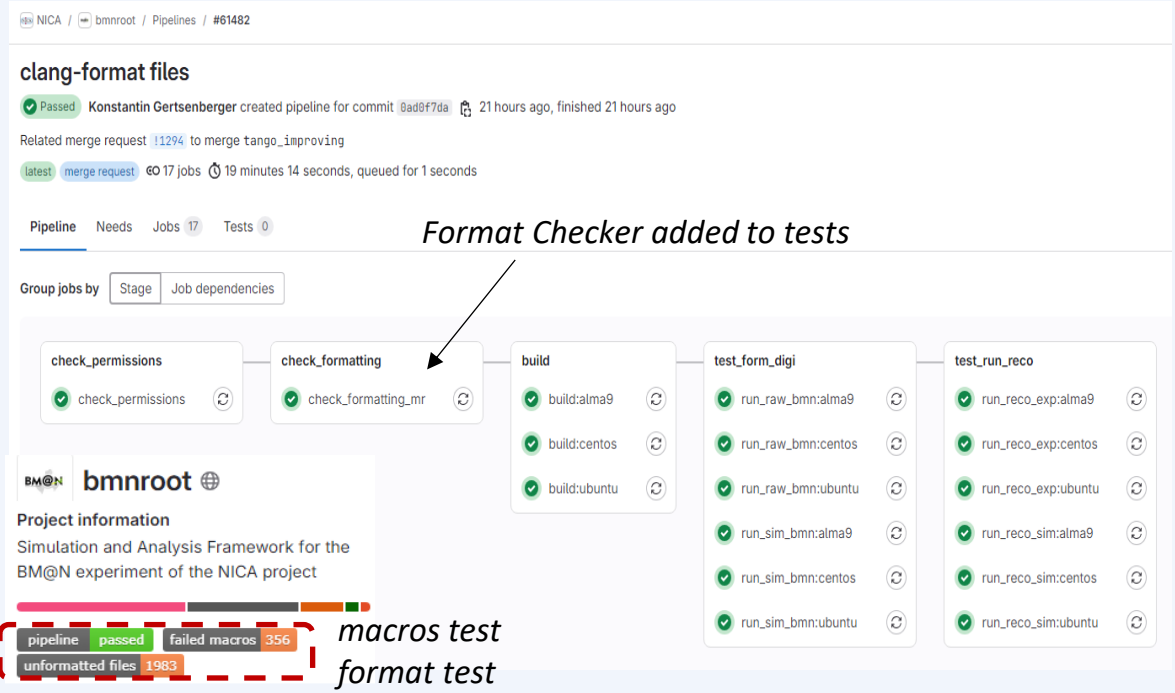
$$\begin{pmatrix} Z_2 & Z_2 & Z_2 & Z_2 & Z_2 \\ Z_3 & Z_3 & Z_3 & Z_3 & Z_3 \\ Z_4 & Z_4 & Z_4 & Z_4 & Z_4 \\ Z_5 & Z_5 & Z_5 & Z_5 & Z_5 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} N_t & 0 & 0 & 0 \\ 0 & N_t & 0 & 0 \\ 0 & 0 & N_t & 0 \\ 0 & 0 & 0 & N_t \end{pmatrix}$$

$$\begin{pmatrix} Ax_1^0 z_2 & Ax_1^0 z_3 & Ax_1^0 z_4 & \dots \\ Ax_2^0 z_2 & Ax_2^0 z_3 & Ax_2^0 z_4 & \dots \\ \dots & \dots & \dots & \dots \end{pmatrix} \begin{pmatrix} Ax_1^0 & Ax_1^0 & Ax_1^0 & \dots \\ Ax_2^0 & Ax_2^0 & Ax_2^0 & \dots \\ \dots & \dots & \dots & \dots \end{pmatrix} \begin{pmatrix} \sum_i Ax_i^0 & 0 & \dots \\ 0 & \sum_i Ax_i^0 & \dots \\ \dots & \dots & \sum_i Ax_i^0 \end{pmatrix}$$



- Reconstructed Track Trajectory
- Measured Track Position
- Assumed Detector Position
- α - Alignment Correction

Software Management & Distribution System



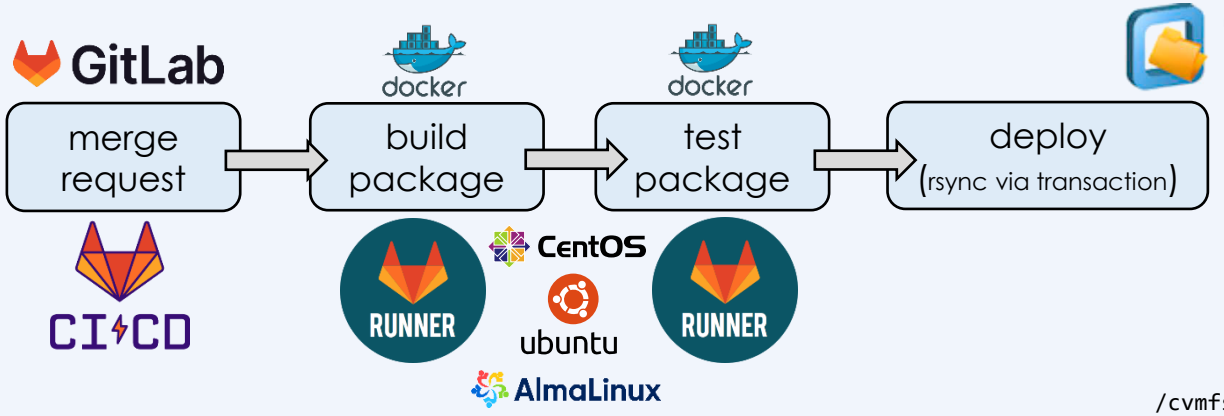
GIT Pipelines on Merge Requests

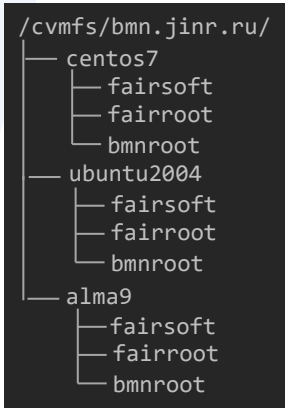
- checking compilation and main macros
  → stable dev & pro branches, all tags
- deploy BmnRoot to the CVMFS storage
- deploy BmnRoot containers to the registry
- checking Clang code formatting

GIT Night Tests

- test all BmnRoot macros
- test all C++ files for correct code format
- generate Doxygen documentation (http://nica.pages.jinr.ru/bmnroot)

Software Distribution via CernVM File System





Code Style (Format) Checker

clang-format files

Failed **Konstantin Gertsenberger** created pipeline for commit `d77e16c3`

Related merge request `!1294` to merge `tango_improving`

merge request · 17 jobs · 30 minutes 8 seconds, queued for 1 seconds

Pipeline | Needs | Jobs 17 | Failed Jobs 1 | Tests 0

Group jobs by: Stage | Job dependencies

Code Checker added Unformatted label 1 day ago

Code Checker @project_25_bot_7b91abf2d5f15e4b78377bd261dd93b8 · 1 day ago — Developer

Dear @gertsen, you have submitted unformatted files. To format them, please run the following command in your bmnroot directory:

clang-format -i **database/examples/uni_db/run/postrun_update.C database/tango/TangoData.cxx database/tango/TangoData.h database/tango/examples/show_avg_field.C database/tango/examples/tango_gem_trip.C database/tango/examples/tango_gem_u.C decoder/BmnConverter.cxx decoder/BmnDecoder.cxx decoder/BmnRawDataDecoder.cxx**

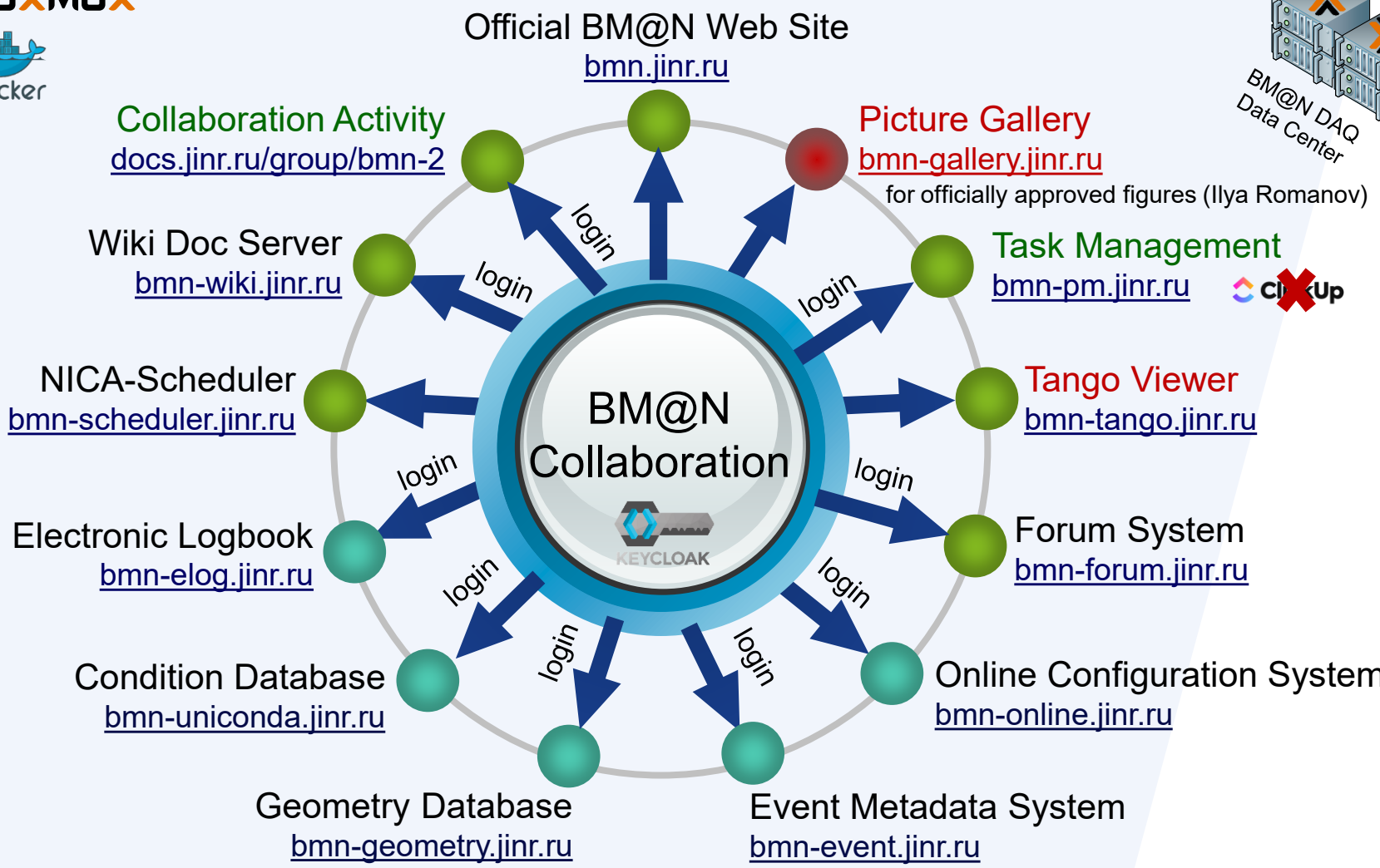
You can check the required changes in the job log or the full clang-format log.

check_permissions	check_formatting	build	test_form_digi	test_run_reco
check_permissions	check_formatting_mr	build:alma9	run_raw_bmn:alma9	run_reco_exp:alma9
		build:centos	run_raw_bmn:centos	run_reco_exp:centos
		build:ubuntu	run_raw_bmn:ubuntu	run_reco_exp:ubuntu
			run_sim_bmn:alma9	run_reco_sim:alma9
			run_sim_bmn:centos	run_reco_sim:centos
			run_sim_bmn:ubuntu	run_reco_sim:ubuntu

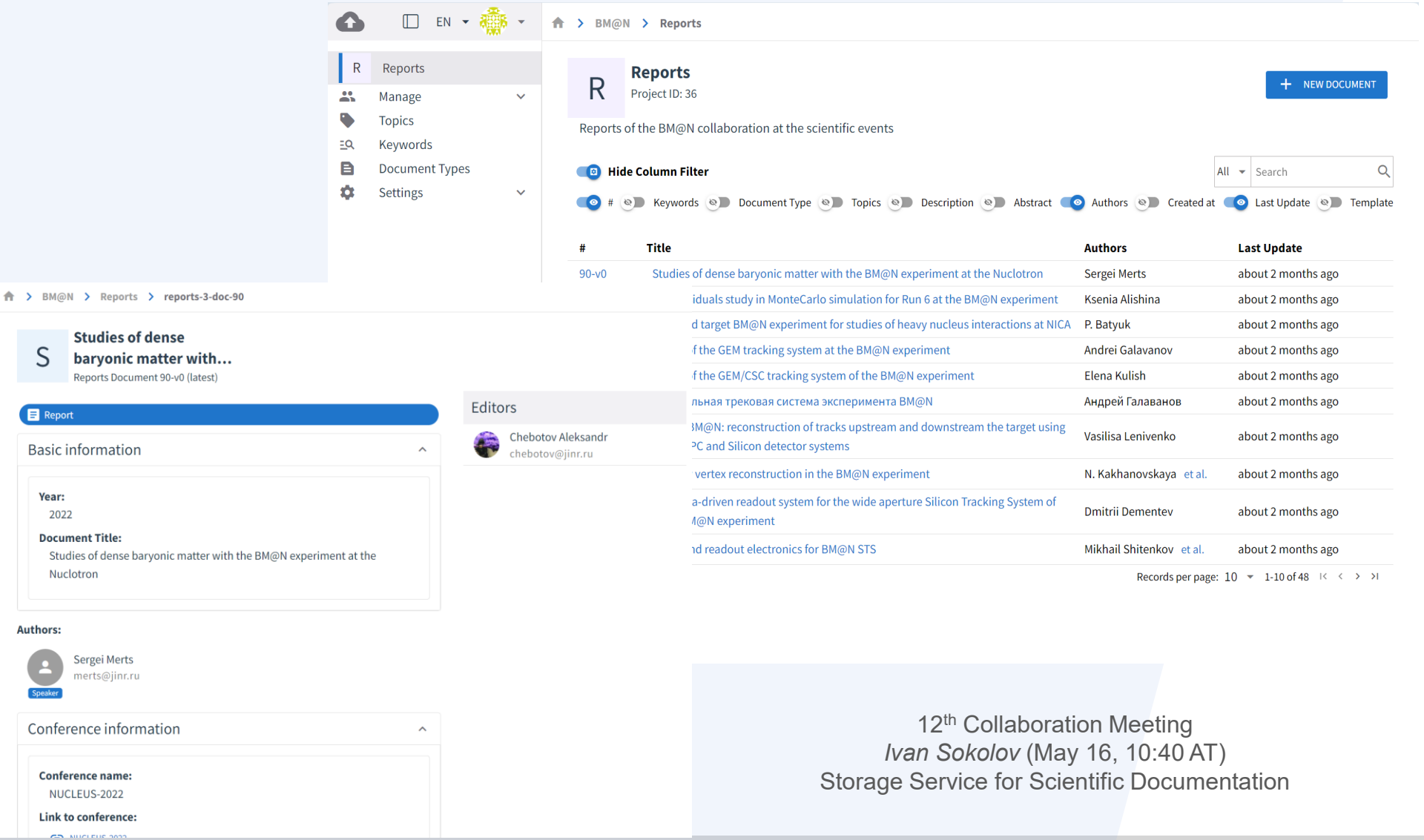
```
The unformatted files are:
 database/examples/uni_db/run/postrun_update.C database/tango/TangoData.cxx decoder/BmnDecoder.cxx
--- database/examples/uni_db/run/postrun_update.C       (original)
+++ database/examples/uni_db/run/postrun_update.C       (reformatted)
@@ -241,7 +241,7 @@
                 pCurrentRun->SetRunType(elog_type);
             }
         }   // if ((run_beam == "") || (run_target == nullptr) || (run_energy == nullptr) || (run_field == nullptr))
-    }   // for (int i = 0; i < run_count; i++)
+    }       // for (int i = 0; i < run_count; i++)

     return 0;
 }
--- database/tango/TangoData.cxx        (original)
+++ database/tango/TangoData.cxx        (reformatted)
@@ -810,7 +810,7 @@
                 }
             }
         }   // for (int j = 0; j < pParameter->bool_parameter_value.size(); j++)
-    }   // for (int i = 0; i < arrTangoData->GetEntriesFast(); i++)
+    }       // for (int i = 0; i < arrTangoData->GetEntriesFast(); i++)

```

12th Collaboration Meeting
Nikita Balashov (May 16, 10:25 AT)
Automated BM@N Software Distribution and Testing: Current Status

BM@N Software Ecosystem

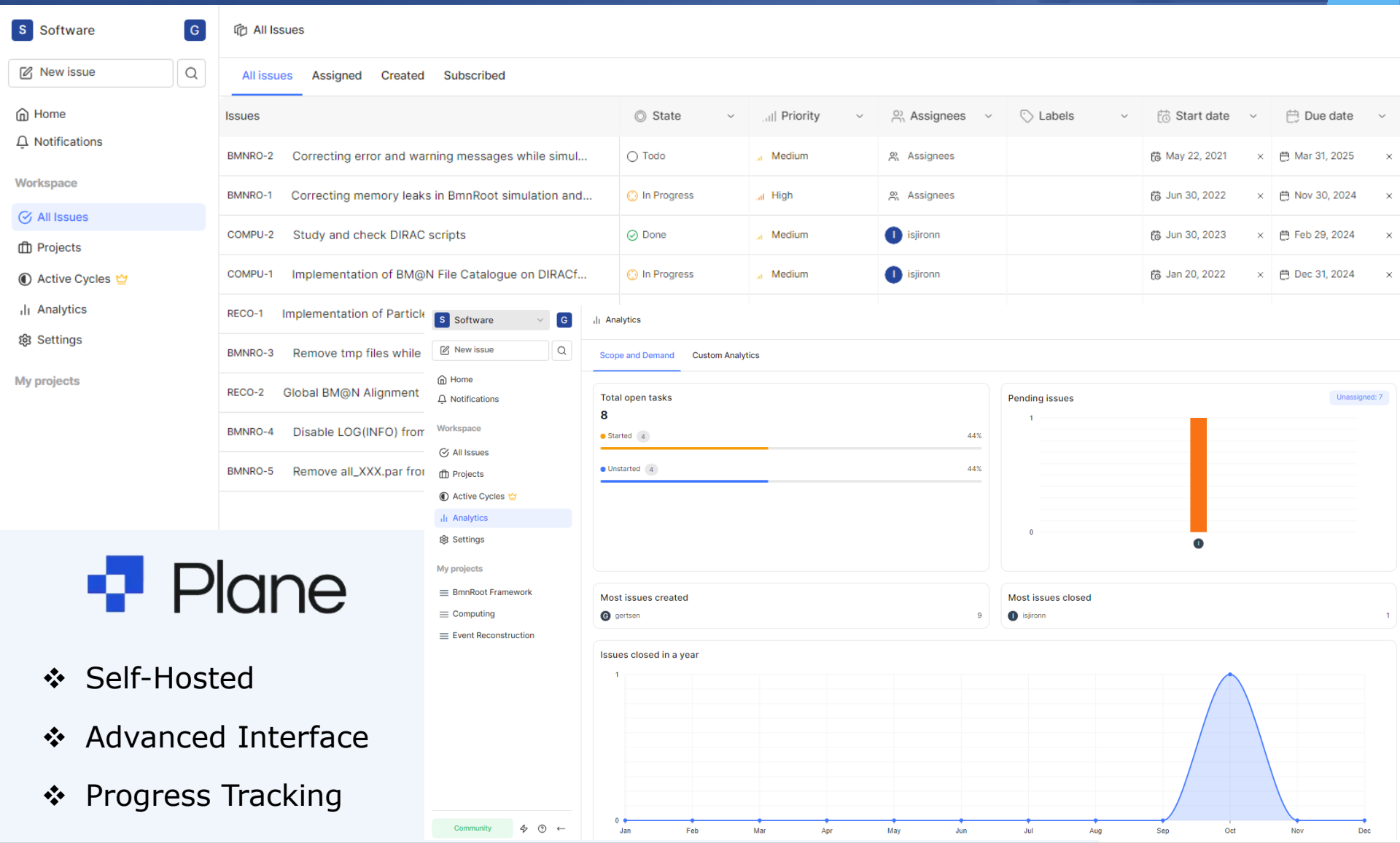


Presentation and Report (DocsDB) Server



12th Collaboration Meeting
Ivan Sokolov (May 16, 10:40 AT)
Storage Service for Scientific Documentation

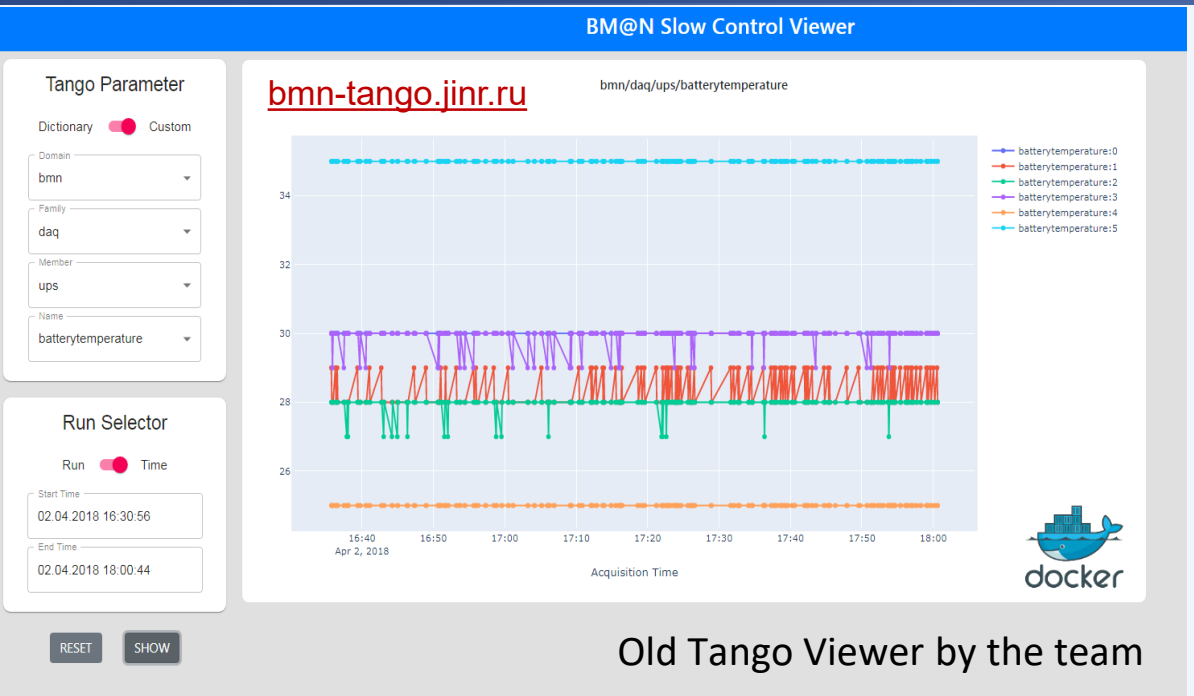
Project Management System for BM@N Software



Plane

- Self-Hosted
- Advanced Interface
- Progress Tracking

TANGO (Slow Control System) Viewer



Old Tango Viewer by the team

- Web-based Viewer for SCS hardware parameters
- Shows sensor data graph based on run number or time interval, and parameter name (dictionary or custom set)
- If a parameter is 1D array, in this case a multigraph is displayed
- Uses Dash framework and packed in Docker container

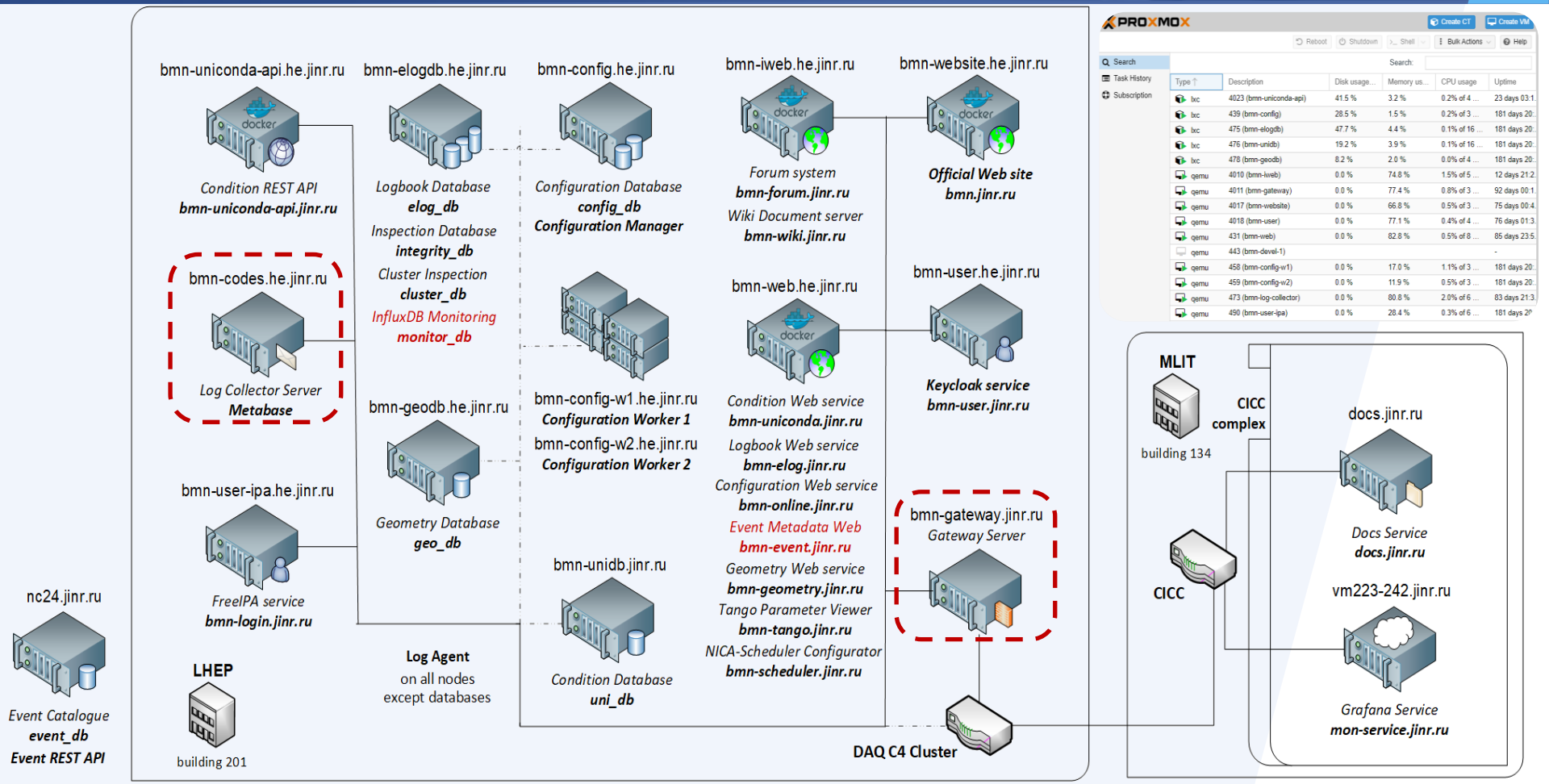
Updated version of Tango slow control database uses PostgreSQL

Convenient REST API service has been developed (supports gRPC)

REST API request example

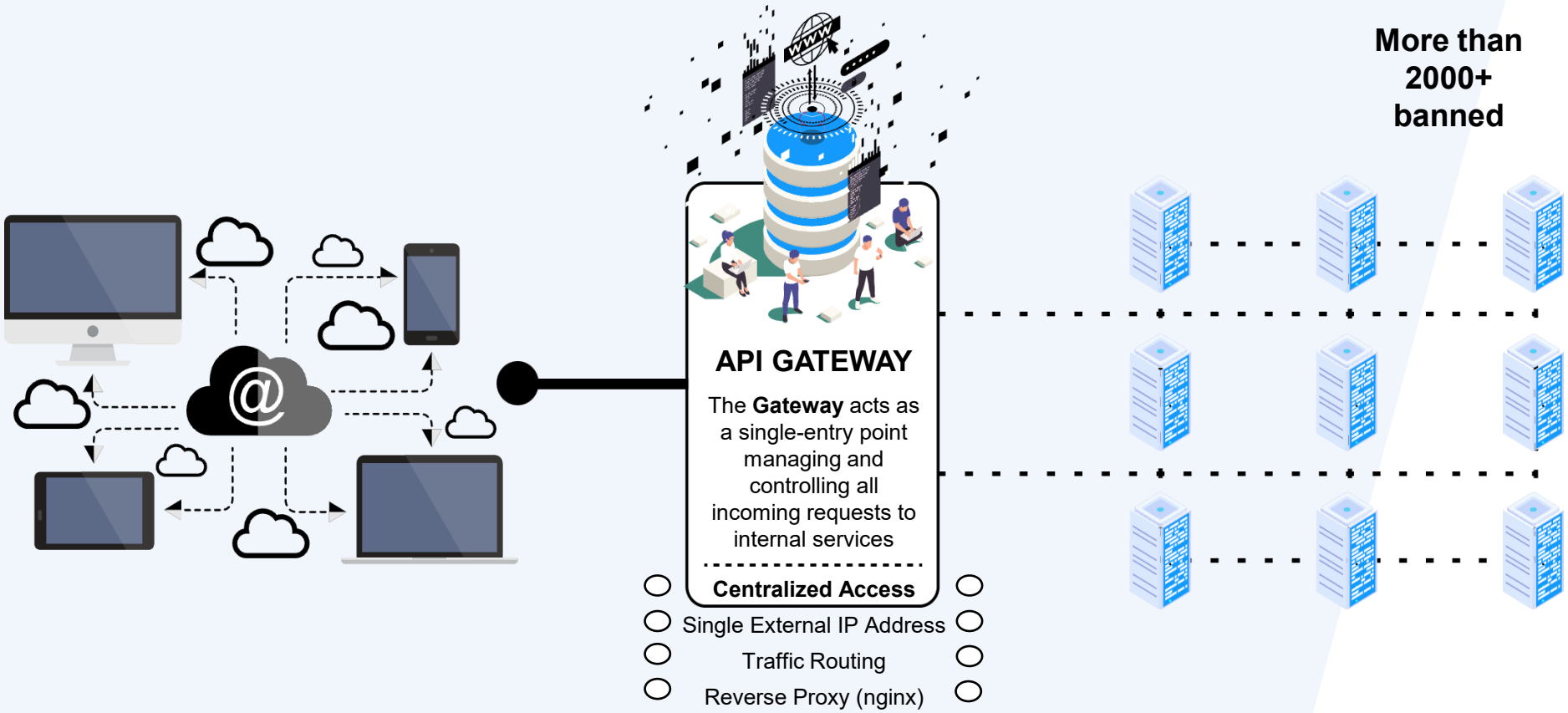
https://<host>:7000/tango-api/v1/parameter?**system_name=**bmn&**parameter_name=**temperature&**member_name=**pir230e_1&**start_time=**2021-11-26&**end_time=**2021-11-27

BM@N Services. *Migration to BM@N Online Farm*



DAQ C4 Cluster (*administrators: Ilya Slepnev and DAQ team)* uses Proxmox, virtualization and resource management platform that allows one to create VMs and containers using virtualization technologies such as KVM for VMs and LXC (Linux Containers) for containers.

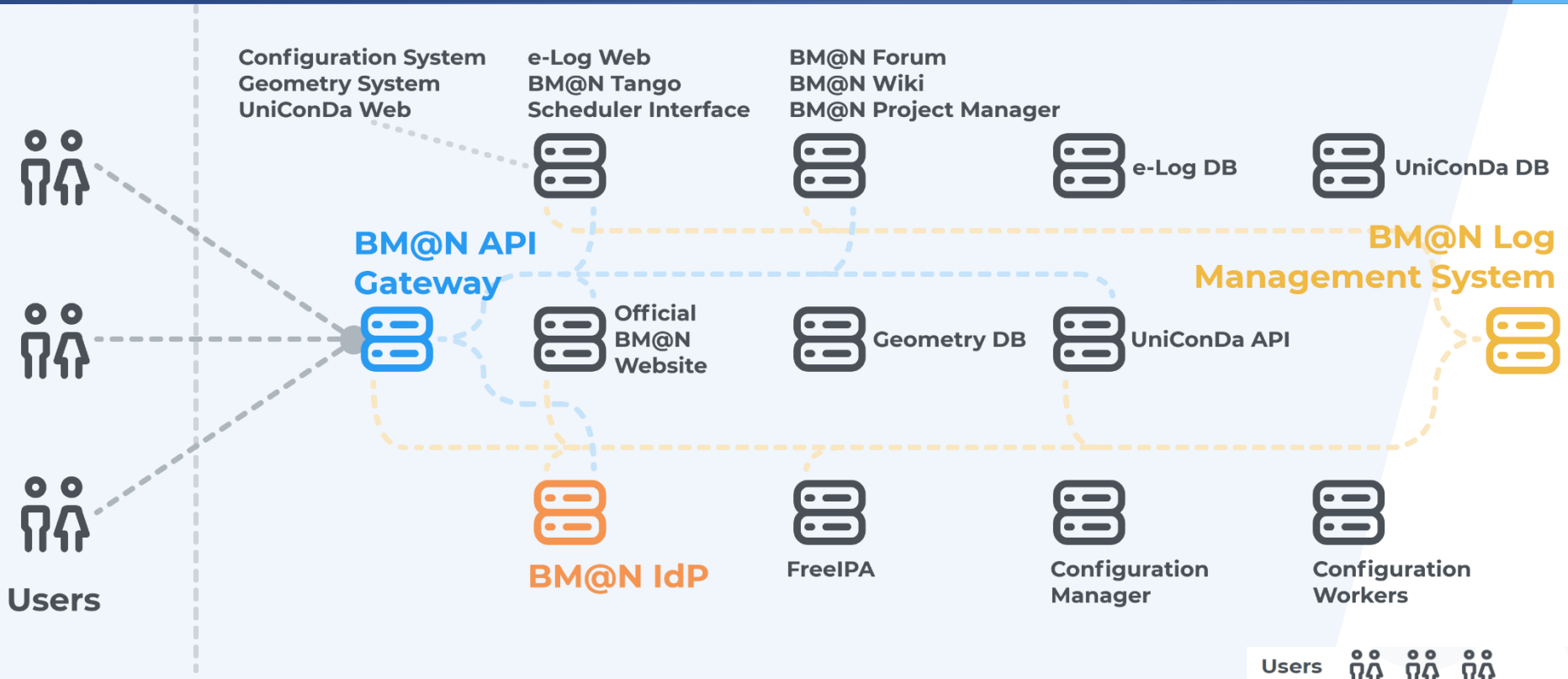
BM@N Gateway as a new security protection



- Enhanced Security
- Traffic Protection and Filtering
- Simplified Traffic Management
- Centralized Logging and Monitoring

- IPTABLES: Traffic filtering at the Linux kernel level
- Fail2Ban: Automated IP blocking on suspicious activity

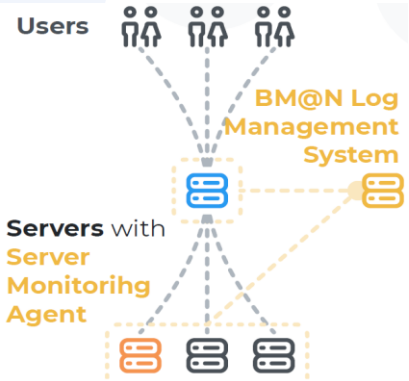
BM@N Log Management System



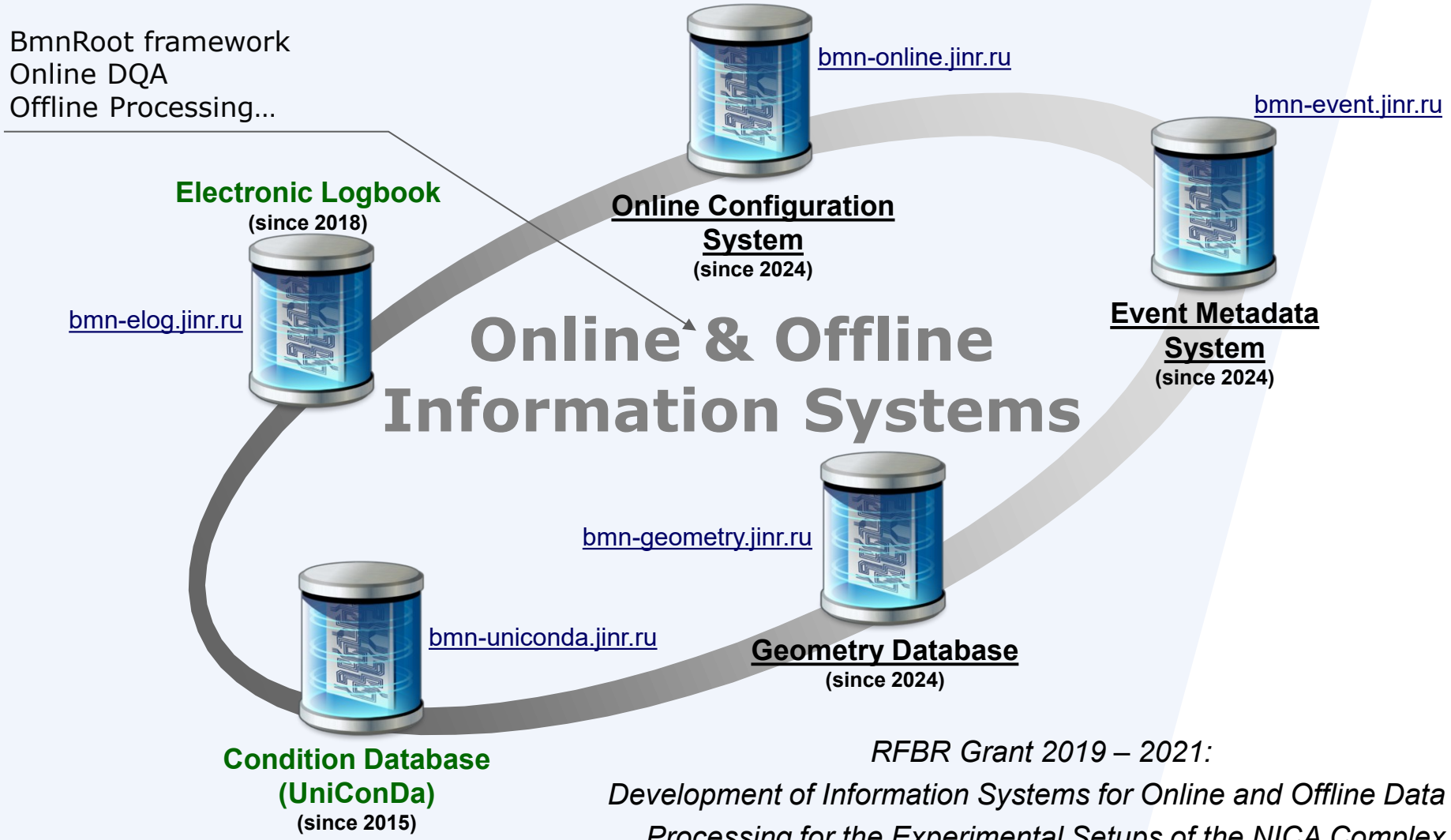
The utilization of the BM@N API Gateway as a single-entry point enables the collection of access logs for all systems in a single location.

The Server Monitoring Agent has been developed to monitor the operation of the BM@N servers and collects data from the auth.log file

The Log Management System helps to quickly identify and fix security issues, making the BM@N infrastructure more resilient.

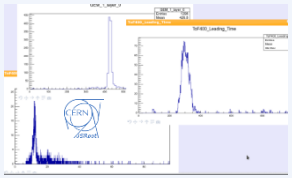


Complex of BM@N Information Systems

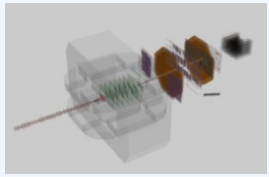


RFBR Grant 2019 – 2021:
Development of Information Systems for Online and Offline Data Processing for the Experimental Setups of the NICA Complex

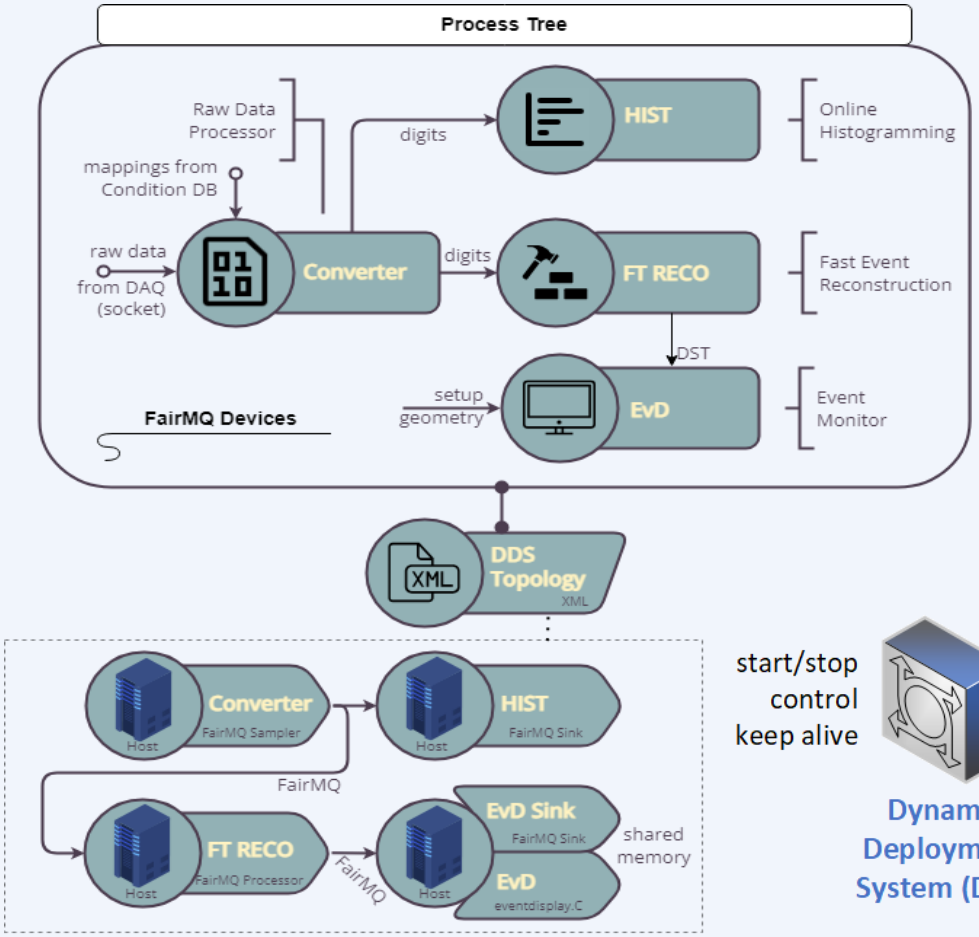
BM@N Online Configuration Platform

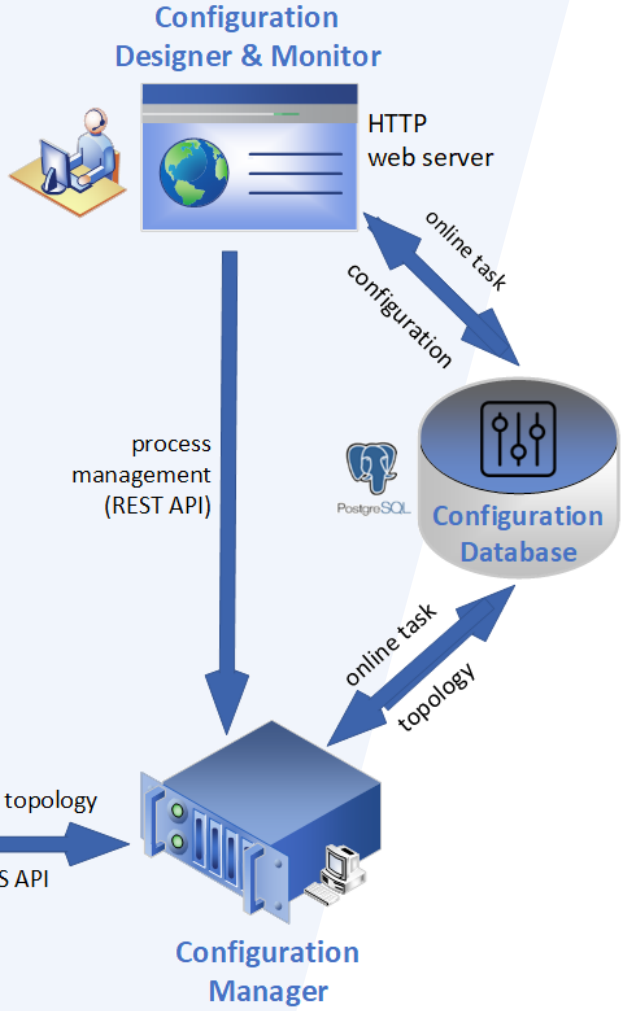


Online Histogramming System

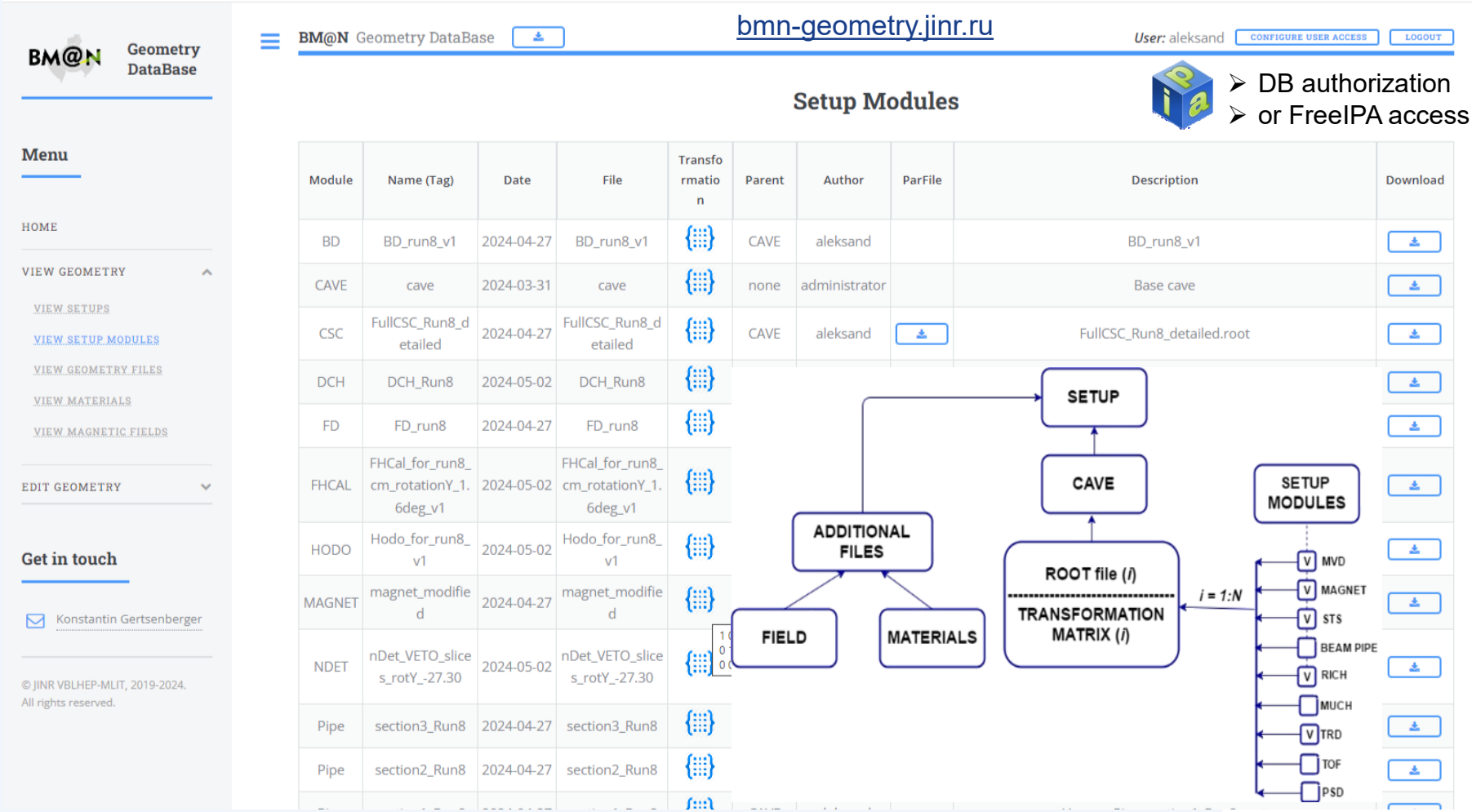


Event Monitor





Geometry Database



- DB authorization
- or FreeIPA access

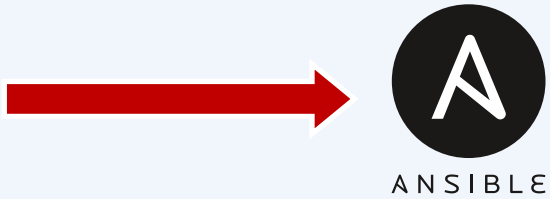
BM@N Geometry Database has filled with the setup geometries for the last Run 8

User Interface Functions:

View Add Edit Approve Download

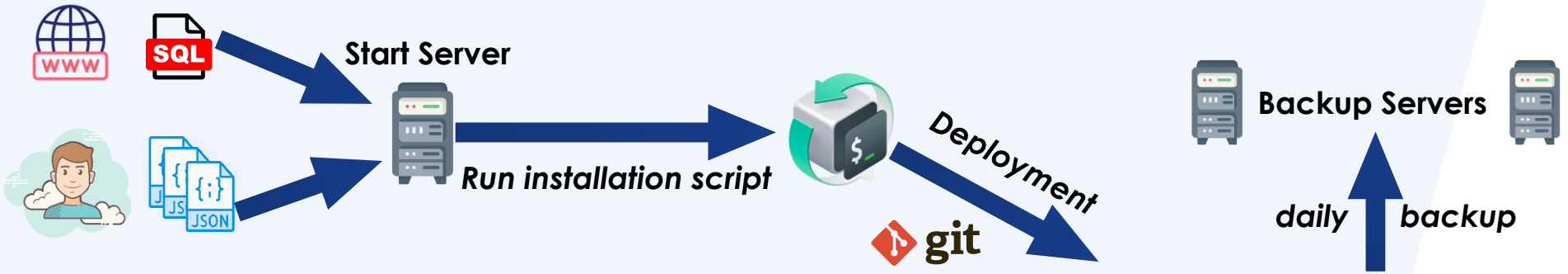
Automated Deployment for Information Systems

- *Why automated deployment?*
  - Manual deployment of a distributed system is slow and error-prone
  - Automation increases speed and predictability
  - Avoids issue of “forgotten step” in documentation
  - Information Systems may be deployed by other NICA experiments
- *Main components of solution*
  - Dockers
  - Python script



- *Inputs*
  - User Configuration of Information System as JSON → YAML template
  - Account configuration for the Information System as JSON file

Common Deployment System (CoDeS)



accounts4db.json
accounts4extauth.json

config.json

```
"dbHost"   : "db_host.jinr.ru",
"dbName"   : "elog_db",
"dbPort"   : 5432,
"dbBackup" :
"remote_user@backup_server.jinr.ru",

"authExt" : false,
"authHost" : ""…

"expName" : "NICA",
"expLogo" : "login/images/nica.png",
"expUrl"  : "https://nica.jinr.ru",

"notifySend"   : true,
"contactEmail" : "person@jinr.ru", …
```

rsync

Crontab

Tracking updates

```
~]# docker ps -a
IMAGE          COMMAND                  CREATED       STATUS       PORTS                                  NAMES
postgres:12.4  "docker-entrypoint.s…"   2 hours ago   Up 2 hours   0.0.0.0:5432->5432/tcp, :::5432->5432/tcp   elog_db
```

Database Server

docker

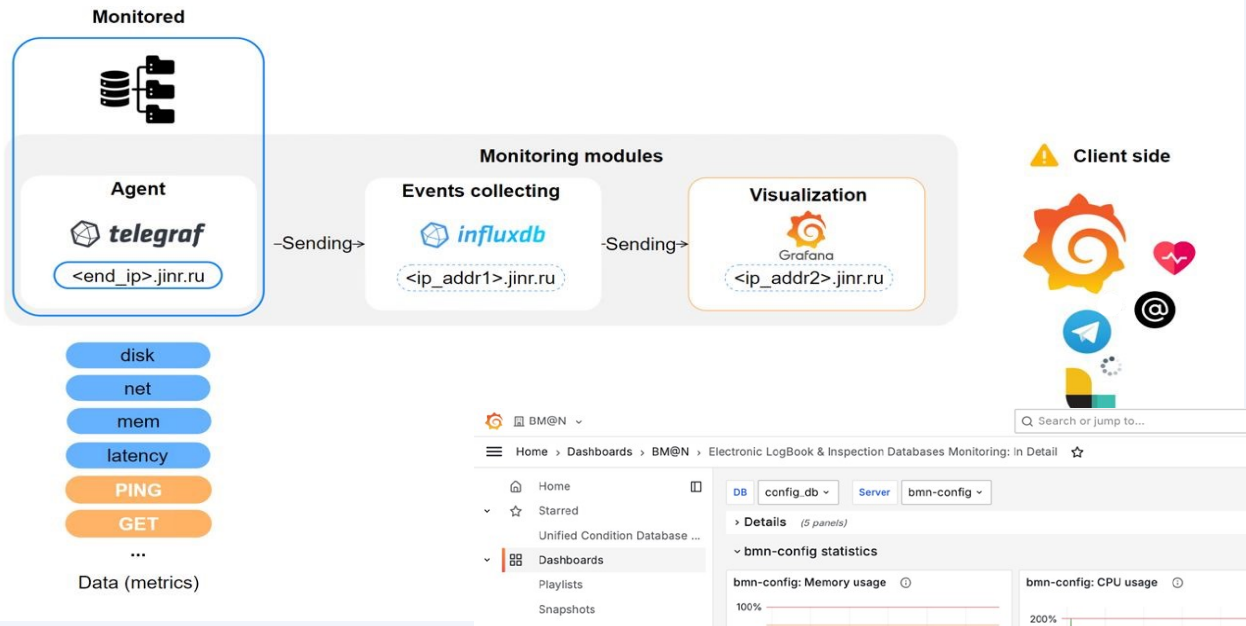
Web Server

docker

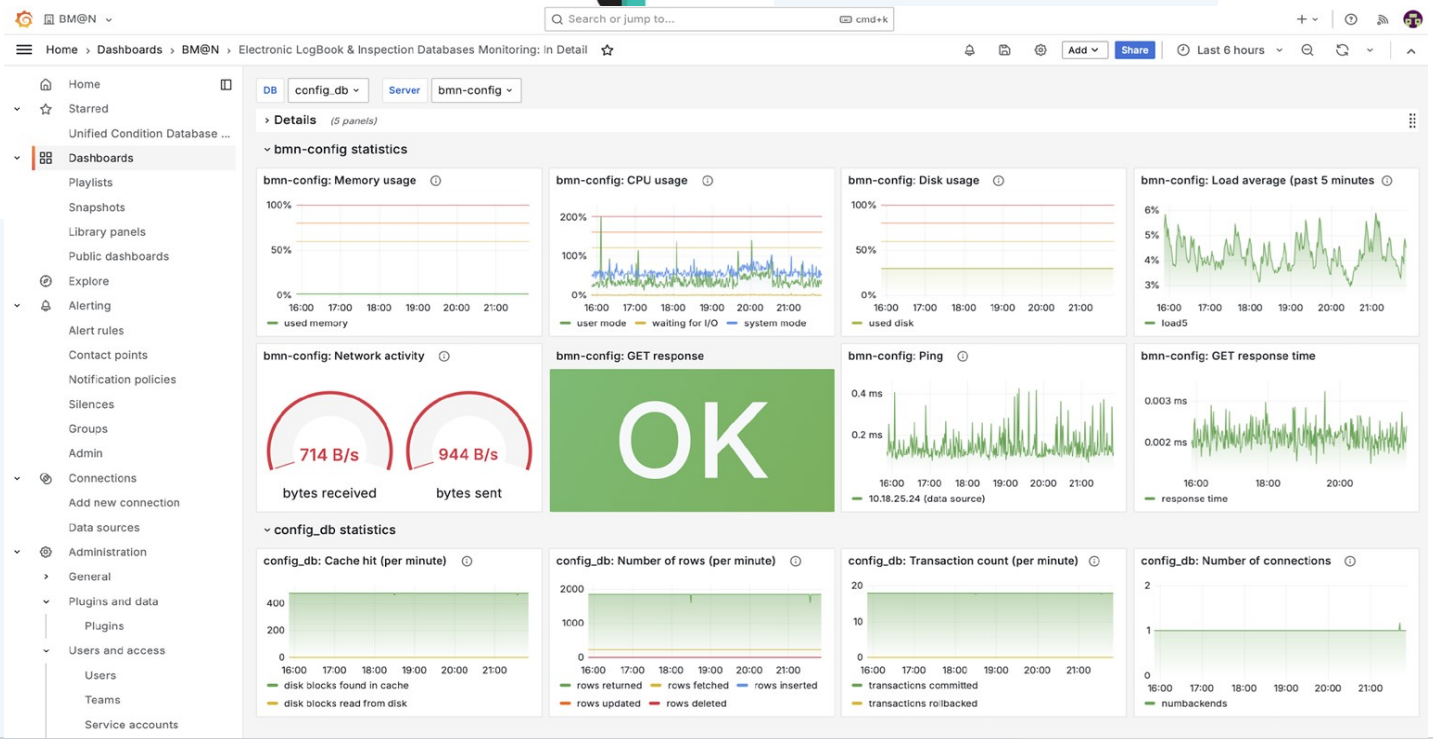
```
elog_platform]# docker ps -a
IMAGE     COMMAND                  CREATED             STATUS             PORTS                                  NAMES
nginx     "/docker-entrypoint.…"   About an hour ago   Up About an hour   0.0.0.0:80->80/tcp, :::80->80/tcp   elog_web
web_php   "/entry-elog-php.sh …"   About an hour ago   Up About an hour   9000/tcp                              elog_php
```

https://git.jinr.ru/nica_db/codes
https://git.jinr.ru/nica_db/elog_platform
https://git.jinr.ru/nica_db/uniconda_platform
https://git.jinr.ru/nica_db/geo_platform

BM@N Monitoring System



- Ping servers, HTTP request and SQL latency to check current status
- Monitor server parameters, such as *Disk*, *CPU*, *Memory*, etc.
- Response time stored in InfluxDB
- Use Grafana for visualization and failure alerting



- Module architecture
- Configurable via JSON file
- Deploying with **Ansible playbooks**
- Email and **Telegram** notifications

BM@N Computing Platforms

/bmn-daq-computing-center/

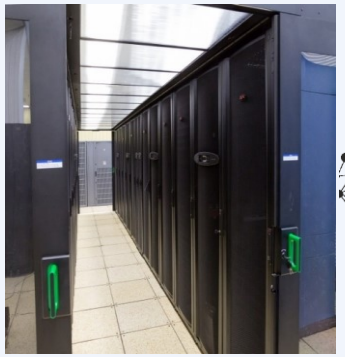
BM@N Online Cluster
ddc.jinr.ru
(LHEP, b.205)

bmn.jinr.ru/nica-cluster/

NICA Cluster
ncx[101-106].jinr.ru
(LHEP, b.216)

bmn.jinr.ru/micc-complex/

GRID Tier1&2 Centres
lxui.jinr.ru (CICC)
(MLIT, b.134)



bmn.jinr.ru/hybrilit-govorun/

HybriLIT platform
(«Govorun» SC)
hydra.jinr.ru
(MLIT, b.134)

OS: AlmaLinux 9

OS: CentOS / Scientific Linux 7.9 (EOL on June 30, 2024)

Central Software Repository based on ***CVMFS*** for the experiment

CEPH: 2.8 PB (*replica*)
CEPH (hot): 100 TB_{ssd}
SLURM: ≈**1000** cores

EOS: 1.2 PB (*replica*)
NFS: **300 TB** (*for NICA*)
SLURM: 1800 cores
(*for all NICA users*)

EOS: 1.2 PB (*replica*)
EOS CTA: **500** TB_{tapes}
SLURM: 2500 cores
(*for all NICA users*)

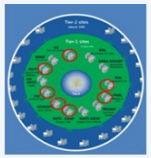
ZFS: 200 TB
Lustre: **300** TB_{ssd} (*for NICA*)
SLURM: *bmn* – **192** cores

BM@N software has been deployed on JINR CVMFS for Centos 7 and AlmaLinux 9
AlmaLinux 9 does not require cluster configuration script (but different versions of GCC)

DIRAC jobs on BM@N Computing Resources

BM@N
Online Cluster

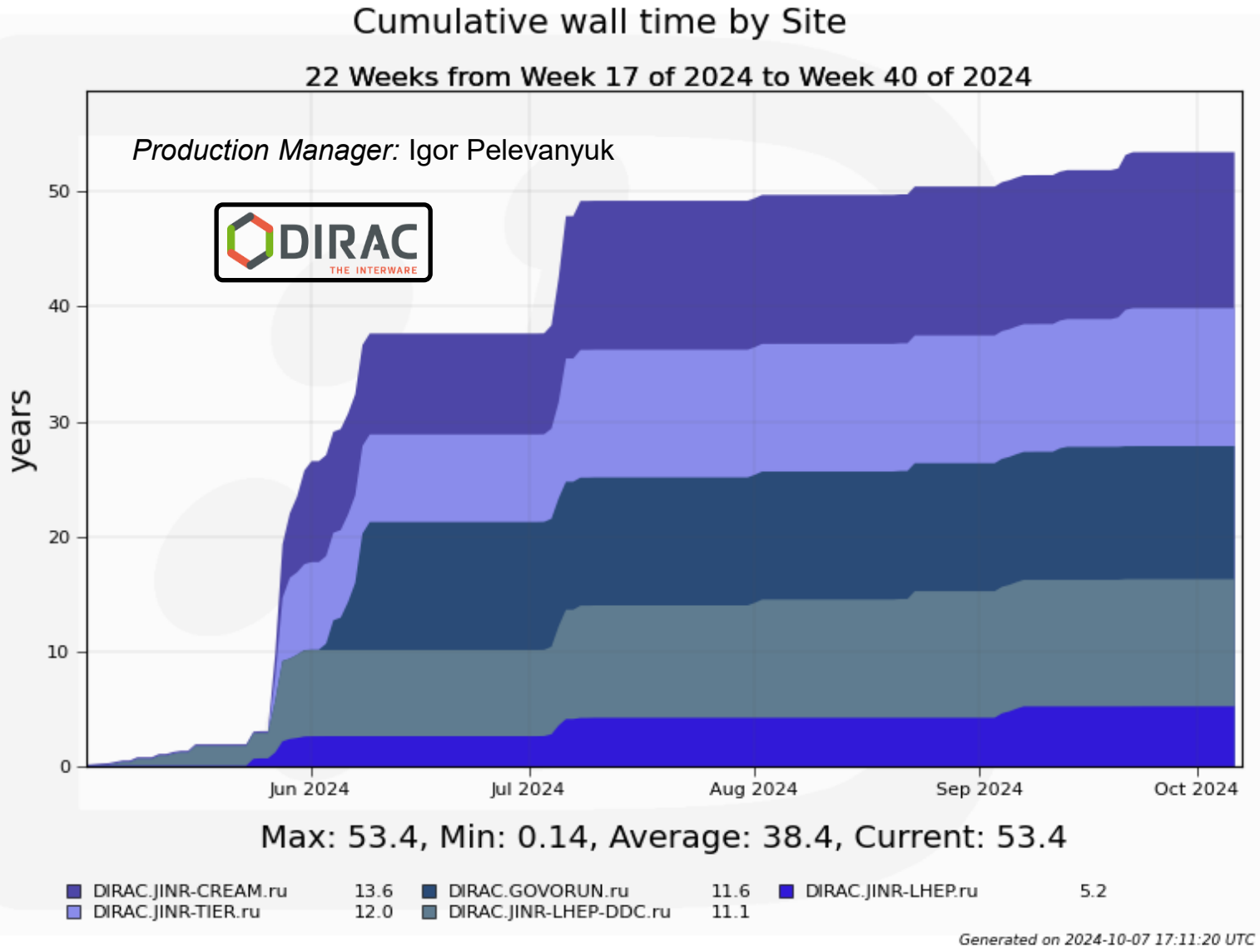
NICA Cluster



CICC
Tier-1

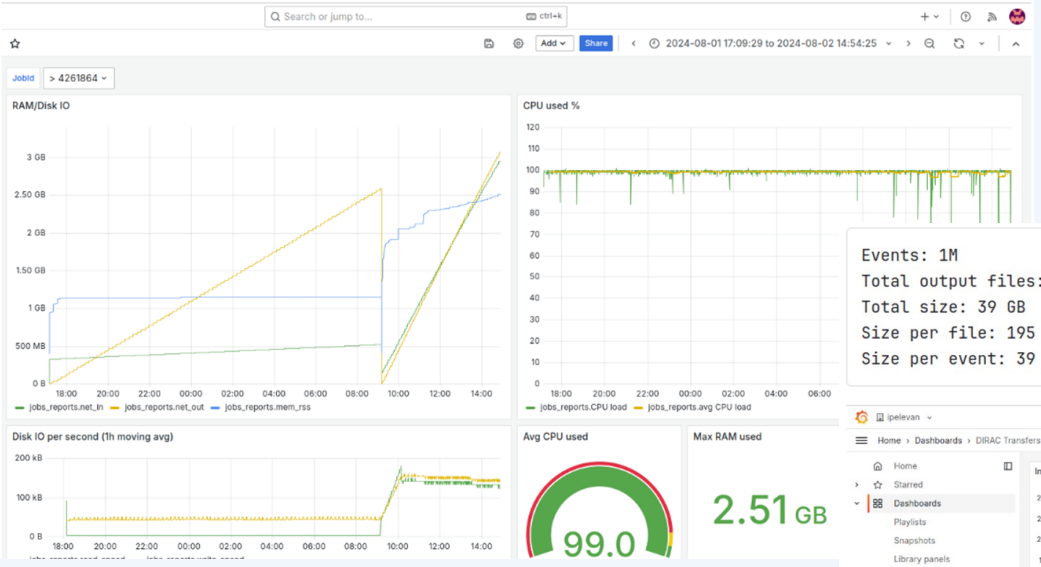
CICC
Tier-2

Govorun



Production Reports via DIRAC

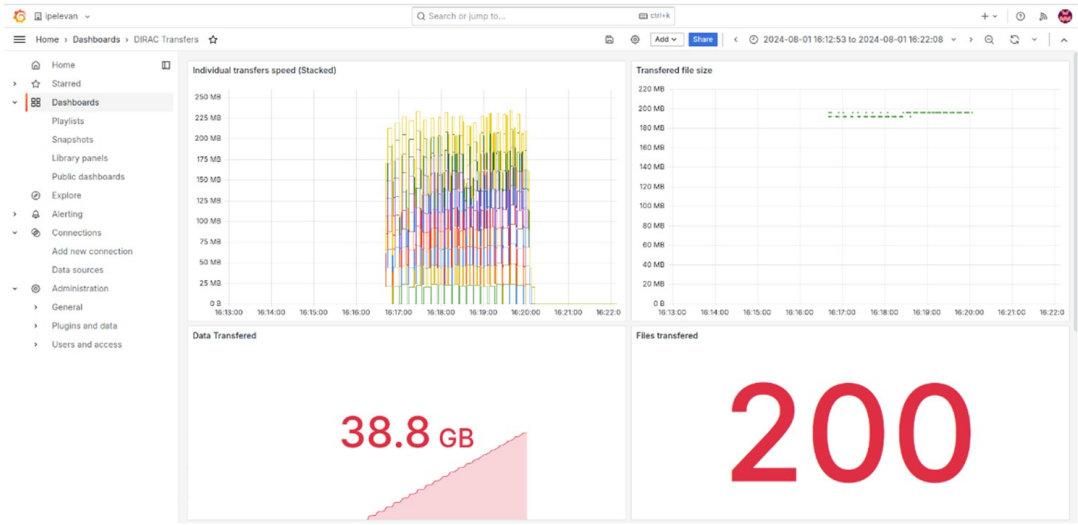
```
Avg execution time: 22 hours
GenToSim: 16 hours
SimToDst: 6 hours
Max RAM: ~ 2.6 GB
Disk usage: ~ 6 GB
CPU Usage: ~ 99%
CPU time per event: 16 s
```



utilization profile of processed jobs

profile of processed and transferred data

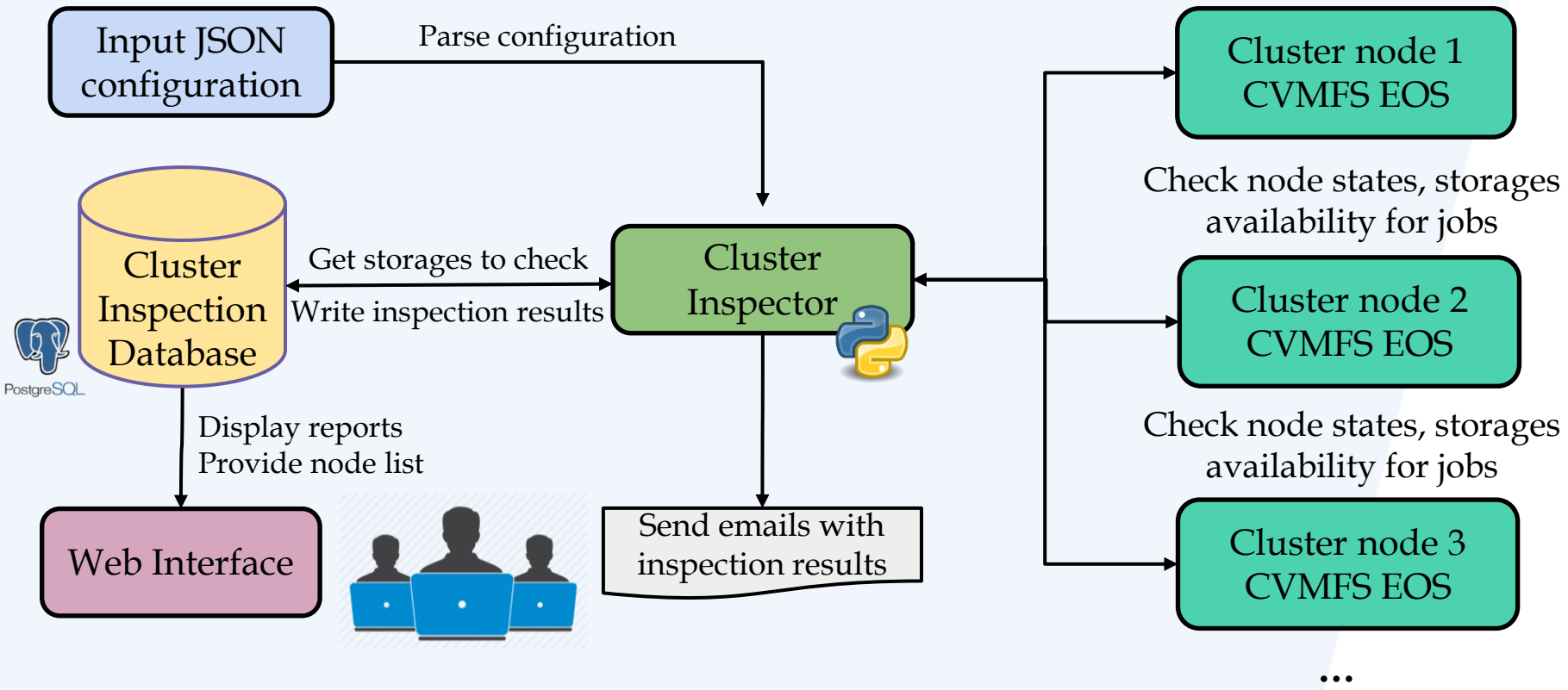
```
Events: 1M
Total output files: 200
Total size: 39 GB
Size per file: 195 MB
Size per event: 39 KB
```



Cluster Inspection Service. *Motivation*

- Not enough reliability of the NICA cluster including regular failure of its computing and storage elements
- The need to constantly have an up-to-date list of non-failed nodes with available data storages to run data processing via the batch system
- To store and visualize the history of the NICA cluster health
- As a result, a tool for tracking and analyzing the status of necessary cluster elements is required, which provides users with a list of successfully tested nodes

Cluster Inspector (Task Crawler). *Overview*

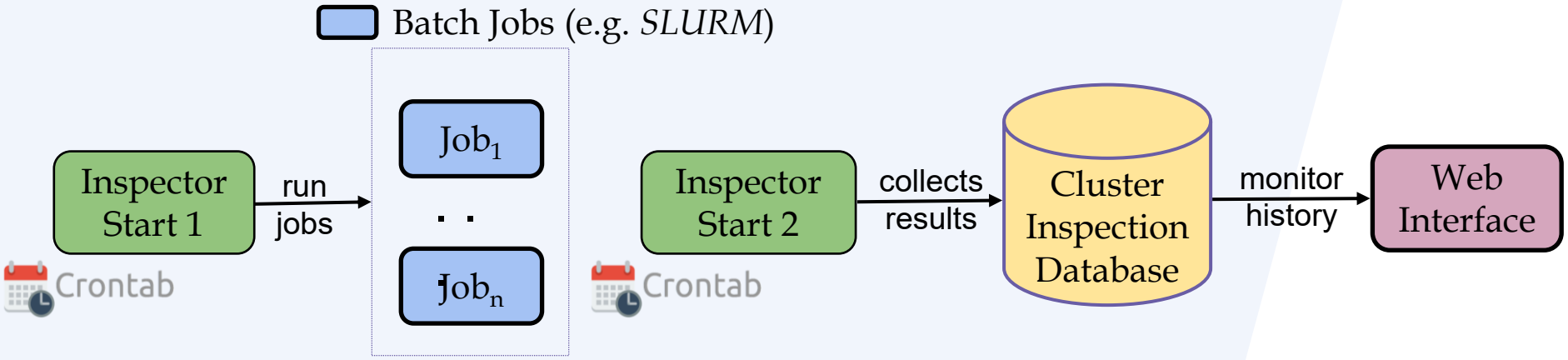


Input JSON configuration contains only the cluster name, its batch system type and email information to send the reports

Cluster Inspector. *Execution on the NICA cluster*

Periodically launches inspection jobs to a batch system of the specified cluster

- A list of nodes of the cluster is defined automatically via the batch system commands
- It checks states of the cluster nodes (*ping* & batch system commands), possibility to start jobs on the nodes (waiting one day) and access to specified (in DB) storage elements
- In 24 hours, the Cluster Inspector collects and analyzes the results, identifies failed nodes, writes the results to the database and send a summary report to specified emails
- The results are displayed on the deployed Web interface



Cluster Inspector. *Web Interface*

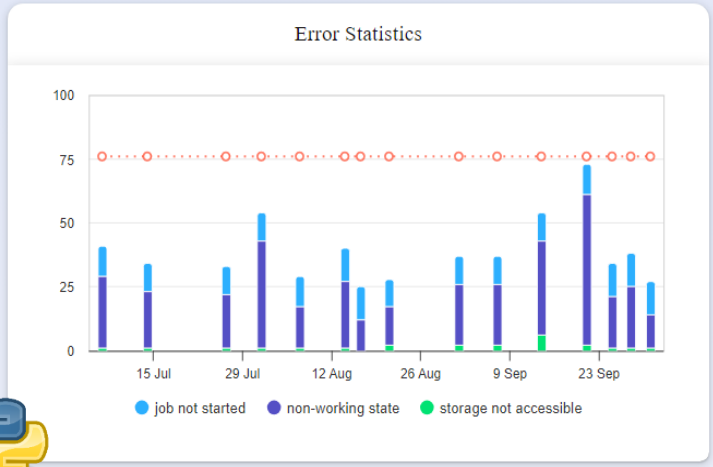
Cluster Inspection

save a list of fully checked cluster nodes to clipboard or text file

Save the last successful hostlist to Clipboard or File

Cluster name	Check date	Complete date	All nodes	Faulty nodes
NCX	2024-10-01	2024-10-02	76	27
NCX	2024-09-28	2024-09-29	76	38

Items per page: 2 1 – 2 of 92



Error Node	Error name	Error Details
ncx112	non-working state	State: down*
ncx115	non-working state	State: unk*
ncx121	non-working state	State: inval
ncx144	non-working state	State: drng
ncx157	non-working state	State: drng
ncx171	non-working state	State: unk*
ncx181	non-working state	State: unk*
ncx188	non-working state	State: drain
ncx214	non-working state	State: down*
ncx216	non-working state	State: unk*
ncx223	non-working state	State: drain
ncx225	non-working state	State: down*
ncx227	non-working state	State: unk*
ncx111	job not started	State: comp
ncx114	job not started	State: comp

contains history of all the conducted NICA cluster checks with statistics on different types of found errors and detailed information on each inspection

Crontab

https://bmn.jinr.ru/cluster_inspector/

BM@N Software Contribution

BM@N Software Contribution

Tagir AUSHEV, *Peter KLIMAI, Alexander NOZIK, PhD student, 1 x graduated , 3 x students (6y)*
Event Metadata System, Event Display, Monitoring Service, Services for BM@N Databases…

Sergei NEMNYUGIN, *Anatoly ALEXANDROV, Rinat NIZAMOV, Anastasiya IUSUPOVA*
RUCIO File Catalogue, Docker Containers for BmnRoot, BM@N Event Visualization

Arkadiy TARANENKO, *Peter PARFENOV, Anton TRUTSE*
Software corrections of the BmnRoot framework

***Director*: S. V. SHMATOV. *Scientific Leader*: V. V. KORENKOV**

Igor ALEXANDROV, Evgeniy ALEXANDROV, Irina FILOZOVA, et alia
Development of the Geometry Database and Online Configuration Systems

Zarif SHARIPOV, Zafar TUKHLIEV. ***Automation of BM@N Alignment***

Alexander AYRIYAN, Vladimir PAPOYAN
Implementation of BM@N Particle IDentification based on ML

Spokesperson: Mikhail KAPISHIN

BM@N Software "Group" (3 FTE)

Konstantin GERTSENBERGER
Alexander CHEBOTOV, Ilya ROMANOV

BM@N Computing and Technical Contribution

BM@N Computing & Technical Contribution

***NICA Computing Leader*: Andrey DOLBILOV**

Ilia SLEPNEV (LHEP Deputy Director for Computing)
BM@N DAQ & online farm support

Ivan SLEPOV
NICA Cluster support

Igor ZIRONKIN
BM@N distributed data processing with DIRAC File Catalogue

***Director*: S. V. SHMATOV. *Scientific Leader*: V. V. KORENKOV**

Nikita BALASHOV: ***CVMFS Deployment, GitLab Services, Docker Containers***

Igor PELEVANYUK: ***DIRAC workload management system and BM@N mass production***

Dmitriy PODGAYNY, Oksana STRELTSOVA
HybriLIT and SC Govorun support

Daria PRIAKHINA, Vladimir TROFIMOV
Modelling System for BM@N computing infrastructure

Software Strategy Risks

Software Fund

- *no financial fund*
- *no support of the external software participants*

Staff

- *no full-fledged software group (permission to hire 3rd employee)*

Computing Resources

- *not enough guaranteed resources for BM@N (only 192 cores on Govorun, online cluster)*
  2024→2027: 6 000 cores required
- *not stable work of the NICA cluster*
- *unrealizable procedure of obtaining VPN for new external participants*