

14th Collaboration Meeting of the BM@N Experiment at the NICA Facility

# Development of new Event Display and auxiliary software for BM@N experiment

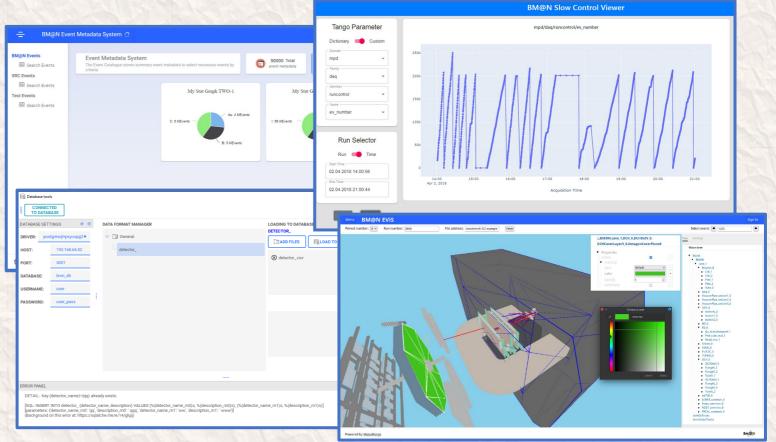
#### Igor Dunaev the MIPT team for the BM@N collaboration



## MIPT Software for BM@N – Team

#### Supervision: T. A.-Kh. Aushev Team members:

- P. Klimai
- A. Nozik
- I. Dunaev (student 6y)
- O. Nemova (student 6y)
- V. Kaplenko (student 6y)
- A. Degtyarev (PhD st. 2y)



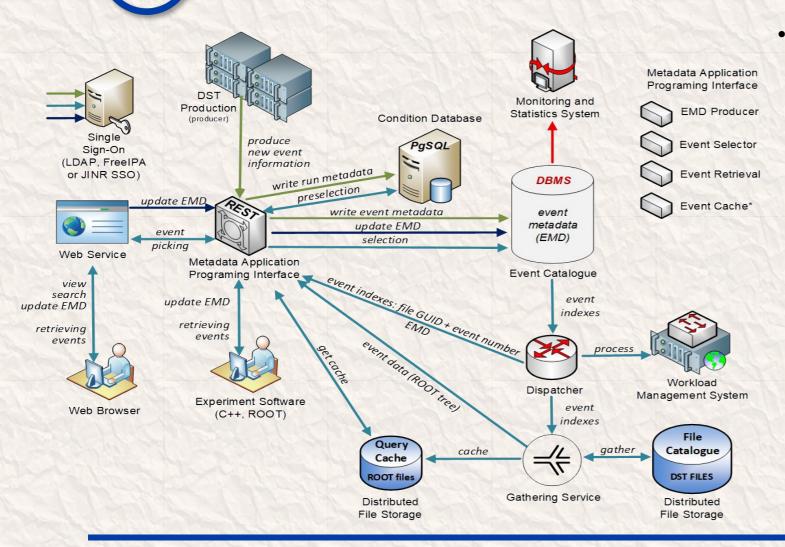
## Main Projects Summary

Project	URL
Event Metadata System	https://git.jinr.ru/nica_db/emd_ https://git.jinr.ru/pklimai/ems-deploy
Next-generation Event Display	https://git.jinr.ru/nica-visualization/visionforge https://git.jinr.ru/nica-visualization/visapi https://git.jinr.ru/nica-visualization/evis
Monitoring Service	https://git.jinr.ru/nica-computing/mon-service-deploy
REST API and Web interface for slow control system	https://git.jinr.ru/nica_db/tango_web https://git.jinr.ru/nica_db/tango-rest



## Event Metadata System (an update)

#### BM@N Event Metadata System



- Event Metadata System
  - Event Catalogue based on PostgreSQL
  - , Integration with BM@N Condition database
  - REST API and Web UI based on Kotlin Multiplatform
  - Configurable to support different metadata
  - ROOT macro to fill the catalogue
  - , Automatic deployment
  - High Availability solution
  - Statistics collection and display
  - 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).

## Recent Progress for the EMS

- EMS progress
  - New version deployed at https://bmn-event.jinr.int/ with about 700M records currently in it
  - Performance improved for bulk writing of new events
  - Improved error processing and display
  - Statistics collection script developed and included in the main repository

BM@N Events		Software Versi.
Search Events	Event Metadata System The Event Catalogue stores summary event metadata to select necessary events by criteria Total Events - 699233764	Software Version - 24.04.0
	Beam Xe (E = 3.8 GeV/n)         Beam Xe (E = 3.0 GeV/n)           Total: 637408951 events         Total: 61824813 events	
	Csi (2%): 586,227,739 events no target:	
	Empty 1444280	
	Gsi (2%): S86227739 Gsi (2%): S6237739 Gsi (2%):	
	events events	
合		BM@N

## **EMS Filtering Event Example**

#### Example obtaining events for run 6700, having 5+ tracks

• Pipe symbol is used for ranges, so filter is to be specified as "5|"

습

BM@N Event Met	adata System						pklimai
Events Search Events	BM@N Events	Storage	File path # Event	Software	Period	# Run	Total track number
	Software Version	ncx	/eos/nica/bmn/exp/dst/run8/2 721	24.04.0	8	6700	5
	Period Number	ncx	/eos/nica/bmn/exp/dst/run8/2 994	24.04.0	8	6700	5
	Run Number 6700	ncx	/eos/nica/bmn/exp/dst/run8/2 1050	24.04.0	8	6700	6
		ncx	/eos/nica/bmn/exp/dst/run8/2 1056	24.04.0	8	6700	6
	Beam Particle	ncx	/eos/nica/bmn/exp/dst/run8/2 1238	24.04.0	8	6700	5
	Target Particle	псх	/eos/nica/bmn/exp/dst/run8/2 1301	24.04.0	8	6700	5
	Energy, GeV	ncx	/eos/nica/bmn/exp/dst/run8/2 1360	24.04.0	8	6700	13
	5	ncx	/eos/nica/bmn/exp/dst/run8/2 2269	24.04.0	8	6700	7
	Limit [dfit=100]	ncx	/eos/nica/bmn/exp/dst/run8/2 2343	24.04.0	8	6700	5
	Offset	ncx	/eos/nica/bmn/exp/dst/run8/2 2555	24.04.0	8	6700	8
	Filter Reset				R	ows per page: 10 👻	1-10 of 100 < >

BM@N



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

## VisionForge Project Overview

VisionForge – platform for creating next-gen visualization systems

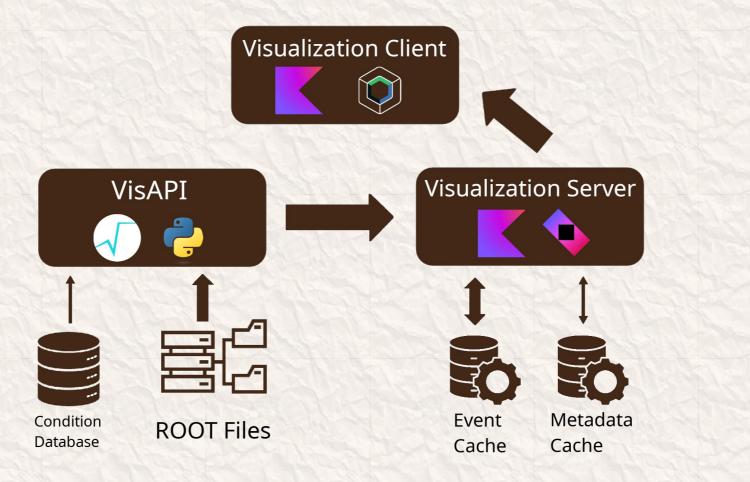
- Distributed dynamic system
  - Visualization model can be created on one node, transferred to another node and rendered there
  - Nodes can exchange updates to the model
  - Changing one element or attribute only requires sending this small change
- Performance and optimizations
  - BM@N geometry model 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

## VisionForge Project Overview

#### See also:

- VisionForge project
  - sciprog.center/projects/visionforge
- VisionForge source code
  - git.sciprog.center/kscience/visionforge/src/branch/dev/
- Alexander Nozik Unbearable lightness of data visualization in Kotlin full stack
  - https://www.youtube.com/watch?v=uT5j-xOXC3E&ab\_channel=JPoint%2CJo ker%D0%B8JUGru





NPM		Event		ater	ncy p	robl	em				500		
	1	12 Per		12-	127	12-25	1	12-51		175	13	1	122
Method	D	File	In	Туре	Transferred	Size	0 ms	1.28 s	2.56 s	3.84 s	5.12 s	6.40 s	7.68 s
GET	<b>%</b>	8000	● <u>e</u>	html	4.95 MB	4.95 MB							7806 ms
GET	<b>%</b>	1201?file=/eos/bmr	♠ <u>e</u>	html	118.53 kB	118.29 kB							7930 m

- Each new event/geometry request takes a significant time to process (approximately 8 seconds in the test setup)
- In case of geometry, the main reason is the JSON format and the size of geometry objects themselves
- Events occupy much fewer space, but they take significantly more time to be extracted from ROOT files.
- Long event load latency is the main distraction for the user looking several sequential events

## Event Cache

Simple caching algorithm (LRU with prefetching):

- Check if the requested event has previously been loaded (hit or miss);
  - If hit: return data from the cache;
  - If miss: send a request to the VisAPI, add results to the cache and return the data.
  - For prefetching: check neighbouring events as well and load them in case of misses;
- Disadvantages:
  - Multiple request to a single event produce multiple requests to the VisAPI (when serving several users);
  - With prefetching, multiple requests to sequential events may produce many requests to the VisAPI and block it (even when serving a single user)

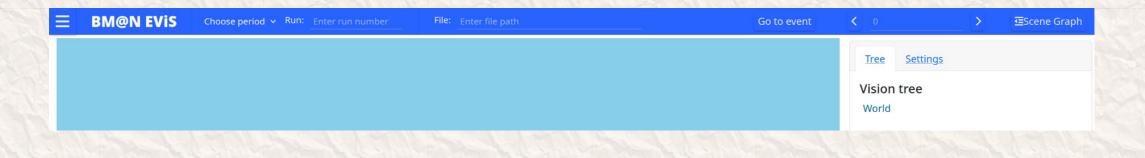
Event Cache

Event service uses an improved LRU cache with prefetching algorithm to reduce sequential event loading latency and improve user experience.

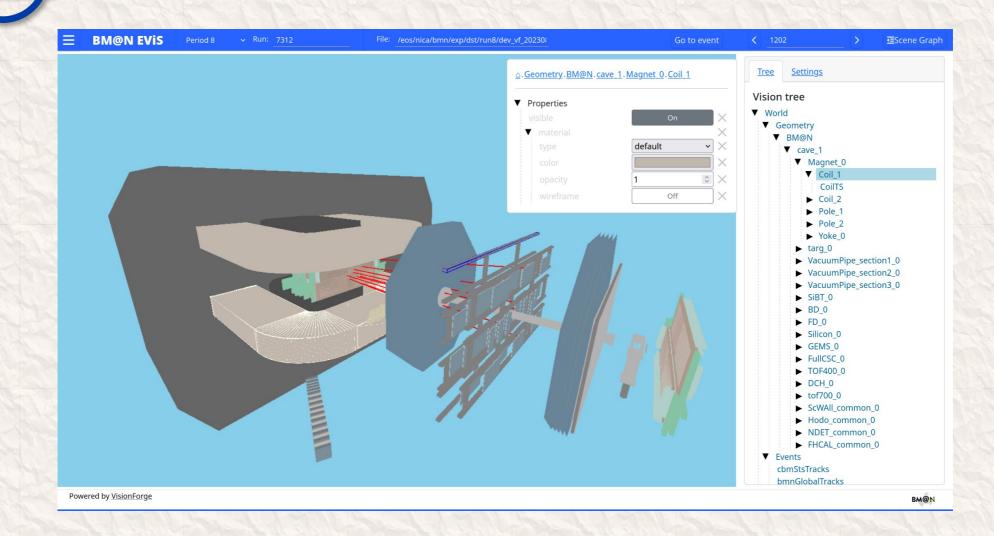
- Check if requested file has been loaded and its metadata are in the metadata cache;
- Atomically:
  - Check if requested event is in the cache;
  - If it is missing, or if the user is close to a non-loaded event, a request to the VisAPI is sent (asynchronously);
  - If the request has been sent, deferred values are added to the cache;
- At this point, a deferred object of the requested event is guaranteed to be present in the cache. Then the event object is awaited and returned.

## Available for test now!

- Available online at <u>http://10.220.16.81:8080/</u>
- Example input:
  - 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 build and run it on your own as well (source code available at <u>git.jinr.ru/nica-visualization/evis</u>)



#### Geometry, tracks, scene graph, tuning



## Available Features

- Visualization of geometry of the detector with a detail level of choice.
- Working with the scene: the ability to scale, shift and rotate the scene, section by plane.
- Saving scene configuration (JSON) to a file.
- Ability to show/hide geometric elements, set color, transparency. For a solid detector, we loaded from a prepared scheme (XML or JSON) to replace the default.
- Visualization of particle collision events: display of tracks and hits. The source is a ROOT file ROOT) stored on a server.
- Event objects and detector geometry are presented as a hierarchical tree, with tracks grouped by particle type. When an object is selected in the tree, the object is highlighted, and vice versa, when an object is selected in the view, its properties are opened.

## WIP Items

- Desktop application with a web UI based on the current client-server solution.
- Real time visualization soluiton: the present visulization solution using websockets to receive data for visualization and to update the data being visualized by web clients.
- Saving an image, saving scene parameters (without event or geometry data itself) to a file.
- Visualization hits and activated calorimeter towers. The source is either a file (initially ROOT), or a data stream from the socket for online monitoring.
- Selection of event objects with viewing of their properties, editing of color, visibility, marker, size/thickness. Selection/scrolling of transferred events in case of the source from a file.
- Filter of displayed event objects: particles by their code, energy range, only primary tracks. Show/hide separately simulated tracks/particles (before reconstruction), reconstructed tracks/particles
- Show general information: selected setup geometry, event number, number of events (if from file), number of displayed geometry objects.



# Development of REST API and Web interfaces for slow control system



#### BM@N slow control system database

- Updated version of Tango slow control database uses PostgreSQL
- Convenient REST API and Web access is required

Language: English v	PostgreSQL » 10.18.86.81:5000 » hdb » public » Select: att_conf
Adminer 4.8.1	Select: att_conf
DB: hdb Schema: public SQL command Import Export Create table	Select data       Show structure       Alter table       New item         Select       Search       Sort       Limit       Text length         name       > =       > temperature       10 0       Select !         (anywhere)       > =       >       Sort       Select !

select att\_array\_devboolean select att array devdouble select att array devencoded select att array devenum select att\_array\_devfloat select att array devlong select att\_array\_devlong64 select att\_array\_devshort select att array devstate select att array devstring select att\_array\_devuchar select att array devulong select att\_array\_devulong64 select att array devushort select att\_conf select att conf format select att conf type select att conf write select att error desc select att history select att history event

SELECT \* FROM "att\_conf" WHERE "name" = 'temperature' LIMIT 10 (0.498 s) Edit

(	Modify	att_conf_id	att_name	att_conf_type_id	att_conf_format_id	att_conf_write_id	table_name	cs_name	domain	family	member	name	tti	hide
(	edit	4	tango://bmn-sc-tangodb.he.jinr.ru:10000/bmn/env/pir230e_2/temperature	5	1	1	att_scalar_devdouble	bmn-sc-tangodb.he.jinr.ru:10000	bmn	env	pir230e_2	temperature	0	f
0	edit	3876	tango://bmn-sc-tangodb.he.jinr.ru:10000/bmn/env/pir230e_3/temperature	5	1	1	att_scalar_devdouble	bmn-sc-tangodb.he.jinr.ru:10000	bmn	env	pir230e_3	temperature	0	f
(	edit	2	tango://bmn-sc-tangodb.he.jinr.ru:10000/bmn/env/pir230e_1/temperature	5	1	1	att_scalar_devdouble	bmn-sc-tangodb.he.jinr.ru:10000	bmn	env	pir230e_1	temperature	0	f

Whole result	- Modify -	Selected (0)	Export (3)
□ 3 rows	Save	Edit Clone Delete	

Import

## API Service Development

- REST API was developed
  - Using FastAPI framework
  - Deployed in JINR network
  - Used by Web interface (see next slide)
  - Sources available at <u>https://git.jinr.ru/nica\_db/tango-rest</u>

FastAPI <sup>610</sup> (	DAS 3.1	
/openapi.json		
default		^
GET /tango_api/v	/1/ Root	~
GET /tango_api/v	/1/domains Get Domains	~
GET /tango_api/v	/1/families GetFamilies	~
GET /tango_api/v	1/members Get Members	~
GET /tango_api/v	/1/names Get Names	~
GET /tango_api/v	/1/attdata Get Graphdata	^
Parameters		Try it out
Name	Description	
domain * required	domain	
family * required (query)	family	
member * required (query)	member	
name * required (query)	name	
<pre>start_datetime * required (query)</pre>	start_datetime	
end_datetime * required (query)	end_datetime	
Responses		

## Web Interface Development

- Web-based viewer for SCS was updated to use the new API and, hence, the new database
- Available at <u>https://bmn-tango.jinr.int/</u>

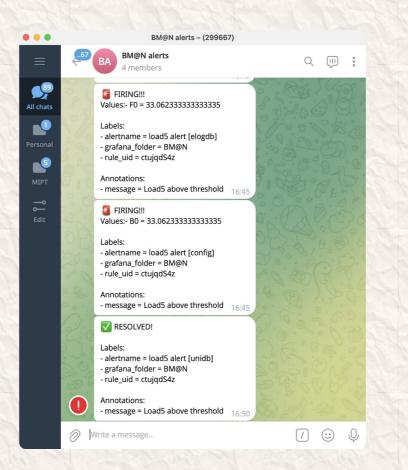




## Monitoring Service (an update)

## **New and Planned Features**

- Dasboard UI
  - Configurable Quick overview
  - Chart sorting
  - HTTP response code chart
- Telegram alerts formatting
  - [WIP] Quick alert summary
- Planned features
  - User Manual
  - Gateway Server, Log Collector Server, Keycloak and FreeIPA service healthchecks
  - Event Metadata system and other systems monitoring





• • • • •	< >			a mon-service.jinr.ru		9 <b>0</b>		⊕ ⊕ + €	
ansible.builtin.scr	G Gmail	J dashboard-gener	o Unified Condition	🧑 Edit panel - Teleg	G HTTP Response	C Telegraf sysstats	online Configurat	Complete list of g.	
BN@N - D	Home , Dashboards , BM@N , Unified	Condition Database (UniConDa)				Q Search or jump to		∞ X+k +- ⊙ A Q	
Home							Add - Settin	gs Exit edit Save dashboar	
Bookmarks - Starred -							Ø	Last 6 hours - Q C Refres	
Electronic LogBook Pl	Quick Overview								
Online Configuration	Uptime	CPU usage	LA medium	RAM usage	Disk usage	IOWait	HTTP Response 💿		
Dashboards -		$\cap$		$\cap$	$\cap$				
Playlists Snapshots	1.1 years	4.85%	4.9	ASA 0.461%		0.03%	Serv	er Error	
Library panels	,	4.85%	1 MAN WWW	0.461%	1 18.6%	- Amanan	MM		
Shared dashboards	- Host Statistics								
Explore -	bmn-unidb.jinr.ru: Memory usage O		bmn-unidb.jinr.ru: CPU usage 💿		bmn-unidb.jinr.ru: Disk usage 💿 💌		bmn-unidb.jinr.ru: Load average (past	5 minutes) O	
Alert rules	80%		SON THE AND	a construction of the second sec	100% 80% 60%		5% A A		
Contact points	60% 40%		1000 High Marked Weber	add margan during the	60% 40% 20%		in rul mulin	Mr. minum MM	
Notification policies	20% 0% 15:30 18:00 18:30 17:00 17:30 18:00 18	100 1000 1000 2000 2000 2000	50% 2% 15:30 W.00 16:30 17:50 17:50 18:00 16:5	a 19-00 19-30 19-00 20-30 19-00	0%	18.50 19.00 19.50 20.00 20.30 21.00	15.30 Ni.00 Ni.30 17.00 17.30 Ni.00	M-50 79-00 79-30 10:00 20:30 1	
Silences Active notifications	- used memory		- user mode - waiting for I/O - system		- used disk		- load5 bmn-unidb.jinr.ru: GET response time		
Settings	bmn-unidb.jinr.ru: Network activity		bmn-unidb.jinr.ru: Memory Swap 0		bmn-unidb.jinr.ru: Ping 💿				
Connections -	2.50 MB/s		100 m8 80 m0		0.4 ms		0.008/04		
Add new connection Data sources	2 MB/s 1.50 MB/s		ee aad 60 mb 40 mb 20 mb		6.3 mi 0.25 mi	descent rabine	0.000 ms		
Administration -	3 MB/s 512 KB/s 0 B/s				0.2 ms // Warrillan Juliv Warrily with	A Constration of the second of	to access which with a share with the state of the state		
General +	15:30 16:00 16:30 17:00 17:30 18:00 - bytes received - bytes sent	18:30 19:00 18:30 20:00 20:30 21:00	15:30 16:00 16:30 17:00 17:30 18:00 18:1 - swap total - swap used	0 10:00 10:30 20:00 20:30 21:00	15:30 16:00 16:30 17:00 17:30 16:00 - 10.18.25.24 (data source)	18:30 19:50 19:30 20:00 20:30 21:00	15:30 %:00 %:30 17:00 17:38 %:00 - response time	18:30 19:00 19:30 20:00 20:30 2	
Plugins and data + Users and access +			- anap total - anap used		- Interstation interest				
	Database Statistics     uni_db: Number of connections		the state of some formation and the		and the Westmanning and the second state		and the Product by Instantion and the		
	s		uni_db: Number of rows (per minute)		uni_db: Transaction count (per minute)	10	uni_db: Cache hit (per minute) O		
	* 11 A.		400000		20000		80000		
	" A.A.A.	<u>_</u> 1.	200000 A A M		10000		42000		
	10 15:50 16:00 16:30 17:00 17:30 16:00 16:1	30 19:00 19:30 20:00 20:30 21:00	15:30 16:00 16:30 17:00 17:30 18:00 18: - rows returned - rows fetched - rows		0 15:30 16:00 16:30 17:00 17:30 16:00	18.30 19.00 19.30 20.00 20.30 21.00	0 15:30 16:00 16:30 17:00 17:30 16:00	18:30 19:00 19:30 20:00 20:30 3	
	- numbackends		- rows deleted		- transactions committed - transacti	ions rollbacked	- disk blocks found in cache - disk l	blocks read from disk	
	- Details								
	bmn-unidb.jinr.ru: Network errors		bmn-unidb.jinr.ru: Number of processes	o	bmn-unidb: Number of buffers		bmn-unidb.jinr.ru: Available Entropy	•	
	100 60		100		15000		500 400		
	40 20		50 15		5000		900 200		
	0 15:30 16:00 16:30 17:00 17:30 18:00 18	30 19:00 19:30 29:00 20:30 21:00	0 15:30 16:00 16:30 17:30 17:30 18:00 18:30	19:00 19:30 20:00 20:30 21:00		18.30 19:00 18:30 20:00 20:30 21:00	100 0		
	- in packets dropped - receive errors	- out packets dropped	- total threads - total processes - runn	ing processes	- buffers allocated - buffers written	by a backend	15:30 %:00 16:30 17:00 17:30 16:00 - entropy avail	18.30 19:00 19:30 20:00 20:30 2	



## Thank You!