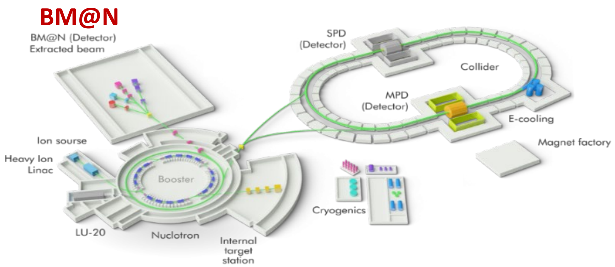
Software development for tracking detectors of the BM@N experiment for the first experimental run in 2022

Baranov Dmitry

for the BM@N collaboration

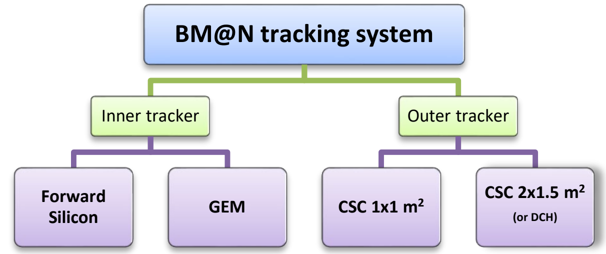
BM@N experimental setup

BM@N (**B**aryonic **M**atter at **N**uclotron) is a fixed target experiment at the NICA accelerator complex aimed to study dense baryonic matter on heavy-ion collisions



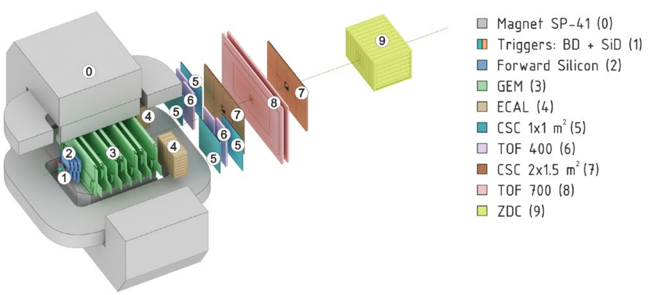
NICA (Nuclotron-based Ion Collider fAcility) is a collider complex being constructed by the Joint Institute for Nuclear Research in Dubna

The complete **tracking system** of the BM@N setup is divided into **inner** and **outer** trackers:



<u>**BM@N exprerimental setup** incudes:</u>

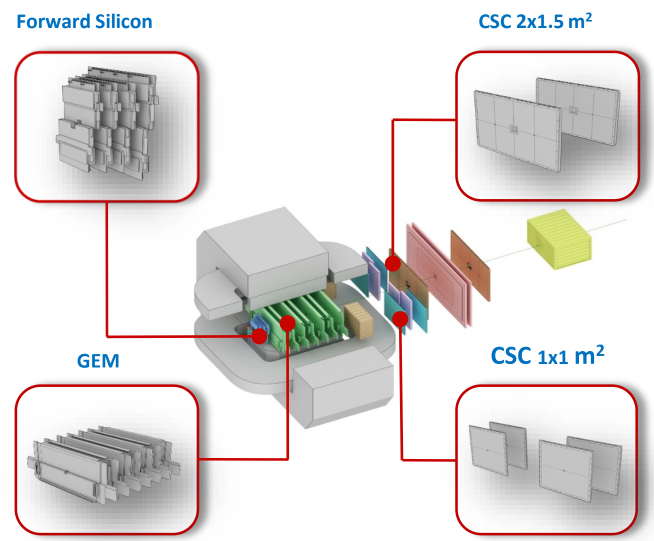
1. **Tracking system** :
   - Forward Silicon (*Semiconductor Silicon Modules*)
   - GEM (*Gas Electron Multipliers*)
   - CSC (*Cathode Strip Chambers*)
   - DCH (*Drift Chambers*)
2. **Particle identification system**:
   - TOF-400 (*First Time-of-Flight detector*)
   - TOF-700 (*Second Time-of-Flight detector*)
3. **ECAL and trigger systems**:
   - Trigger detectors
   - ECAL (*Electromagnetic Calorimeter*)
   - ZDC (*Zero Degree Calorimeter*)



Configuration of the BM@N setup for the first experimental run in 2022

<u>BM@N tracker consists of the following components:</u>

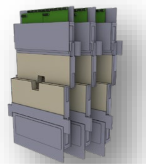
- ❑ **Forward Silicon** detector (4 planes)
- ❑ **GEM** detector (7 planes)
- ❑ **CSC** detector (4 planes of 1x1 m^2)
- ❑ **CSC** detector (2 planes of 2x1.5 m^2) *as the replacement of DCH*



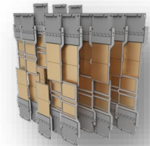
Tracking detectors: configuration

Forward Si is a silicon based semiconductor detector consisting of Si-modules. There are two types of these modules: standard with a sensor size of **63x126** mm^2 and long with a sensor size of **63x186** mm^2

1st configuration (minimal)

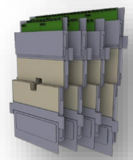


3 stations

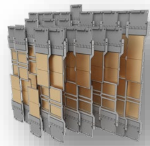


42 Si-modules

2nd configuration (maximal)



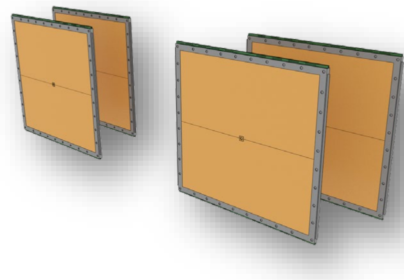
4 stations



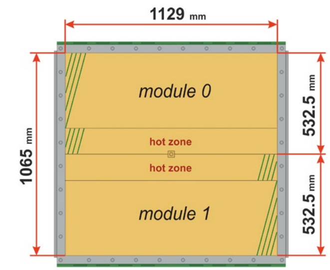
64 Si-modules

At this moment there are two versions of the configuration for the Forward Silicon detector for the future BM@N run

CSC (**C**athode **S**trip **C**hamber) is a gaseous detector with microstrip readout. It is comprised in the outer tracking system of the BM@N setup. The configuration of this detector for RUN-8 consists of four planes located behind the magnet

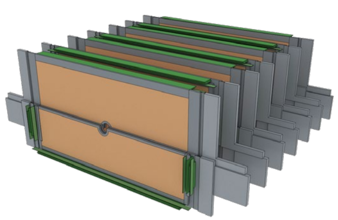


Configuration of the CSC detector for RUN-8

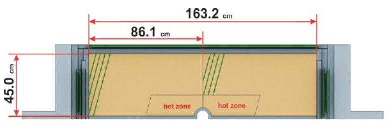


*Each plane has sensitive area with a size of **1129x1065** mm^2*

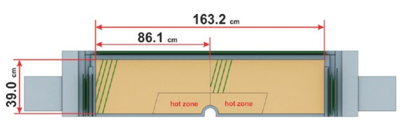
GEM (**G**as **E**lectron **M**ultiplier) is a gaseous detector with microstrip readout. Its configuration for RUN-8 comprises seven stations located inside the magnet along the beam axis



Configuration of the GEM detector for the first experimental run in 2022



upper half-plane

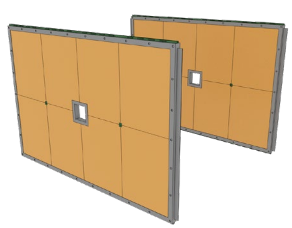


lower half-plane

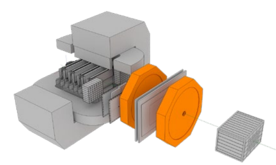
Each station in the configuration is combined from two half-planes: upper and lower. They have different sizes of their sensitive areas:

- Upper half-plane sensor: **163x45** cm
- Lower half-plane sensor: **163x39** cm

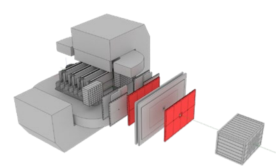
CSC detector consisting of two large chambers with a sensor area size of **2190x1453** mm^2 was also developed for the next RUN. As a part of the upgraded BM@N tracking system, this detector will replace the current DCH (drift chambers) detector with itself



CSC detector consisting of two large chambers for RUN-8



Previous BM@N tracking system includes two drift chambers (in orange color)



Upgraded BM@N tracker has two cathode strip chambers (in red color) instead of drift chambers

Tracking detectors: simulation

Stages of simulation for tracking detectors in BM@N

Using Geant3/4 transport engine for particle propagation — **Monte-Carlo simulation**

↓

MC points

↓

Using tool Garfield++ for detailed simulation of physics processes inside chambers — **Realistic simulation**

↓

Forming clusters on strip layers — **Digitization**

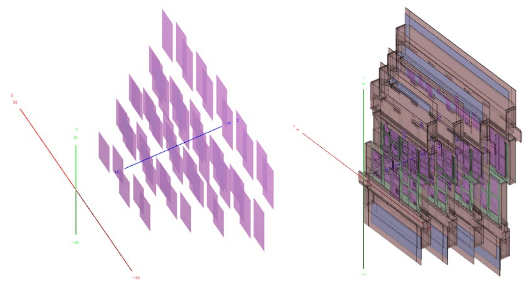
↓

A set of strips with their signals — **Digits**

Monte-Carlo simulation

To perform Monte-Carlo simulation we must prepare **ROOT geometry**, which describes the constructive features of detectors. Then, this geometry is used by a transport engine, such as Geant3 or Geant4, **to simulate the propagation of particles** through matter. The result of the procedure is a set of **Monte-Carlo points** (spatial coordinates), that describe particle trajectories.

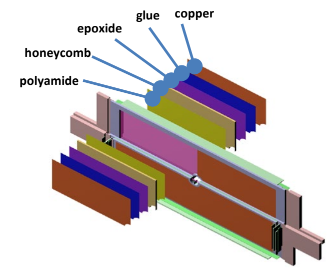
There are two versions of ROOT geometry designed for each tracking detector: simplified and detailed



Simplified: only sensors *Detailed: sensors and passive elements*

Example of ROOT geometry for the Forward Silicon detector

Detailed geometry completely describes the detector structure, which is needed to take into account the material budget of detector in MC-simulation

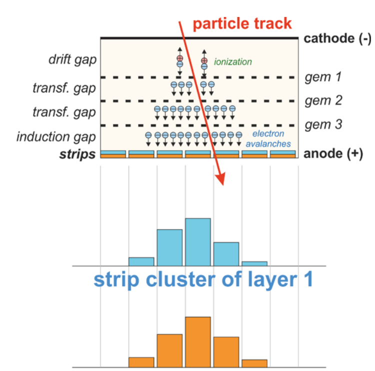


Example: structure of the detailed ROOT geometry for the GEM detector

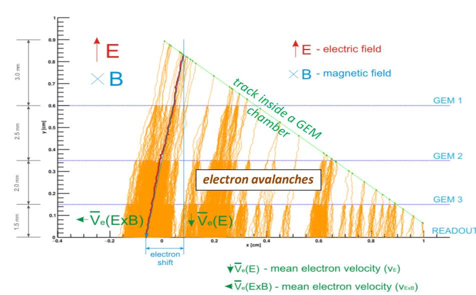
Realistic simulation

In addition to Monte-Carlo simulation, to achieve maximum conformity between simulated and experimental data we use **realistic simulation**, that takes into account the features of signal formation in our detectors.

A **digitization procedure** transforms Monte-Carlo points into "digits" (or "strips" in case of microstrip readout) by applying realistic simulation algorithms for a certain detector type.



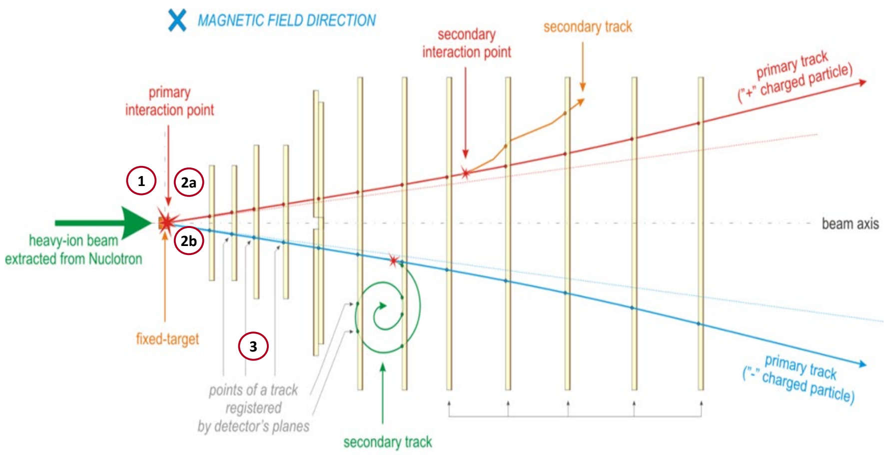
Example of signal formation in GEM chamber



*Example of detailed simulation of one particle inside GEM chamber (calculated by using **Garfield++**)*

Tracking detectors: hit-reconstruction

Scheme of registering particle trajectories by tracking detectors

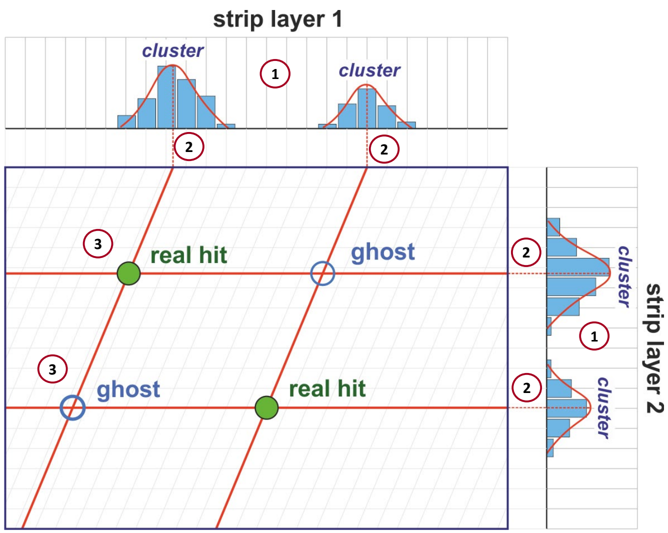


1. A heavy-ion beam, extracted from Nuclotron, collides with a fixed target

2. As a result of this primary interaction is various particles. Their flying directions depend on their charge and a magnetic field which the detector located in (due to the Lorentz force)

3. Passing through the detecting planes, a particle leaves a "trace" (signal) on each of them. Our purpose is to reconstruct a spatial point, called "hit", which this particle passed through. A set of these hits from one particle defines its trajectory

Hit reconstruction procedure for microstrip readout



1. There is a set of digits for one event (signals on strips for each layer). We find clusters of strips and estimate their parameters

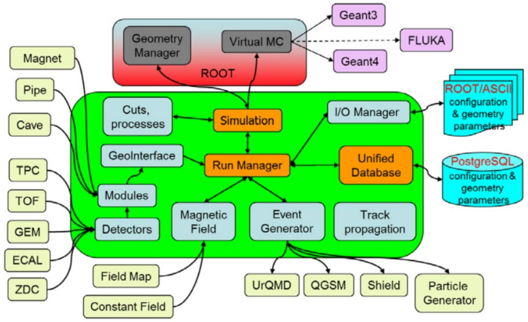
2. We find weighted position of each cluster to collapse lighted strips into one average-weighted strip

3. Crossing these strips of one layer with another, we get intersections, where one part of them are hits from real particles and another – "ghosts"

4. The obtained hits are used in the subsequent track finding procedures which find tracks and eliminate ghost hits

Tracking detectors: software implementation

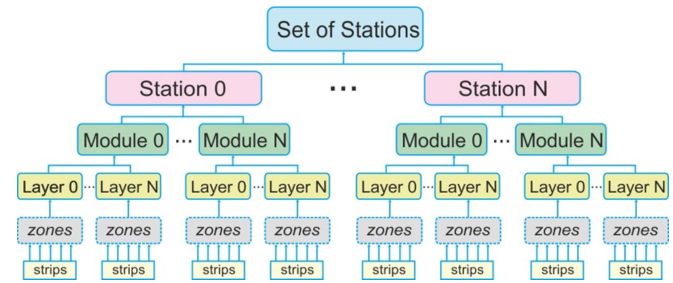
The official software using to support the BM@N experiment is based on the **BmnRoot** framework. It provides powerful tools for simulation, reconstruction and data analysis



Structure of the BmnRoot framework

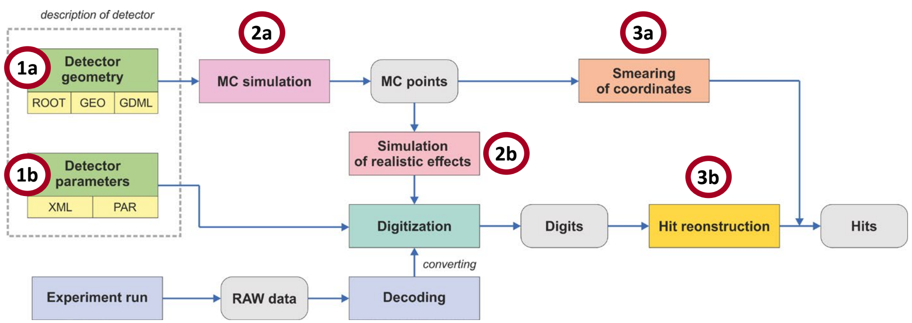
All the tracking detectors have the same hierarchical program structure:

Strips are integrated into a layer,
Layers – into a module,
Modules – into a stations,
Stations – into a set of stations



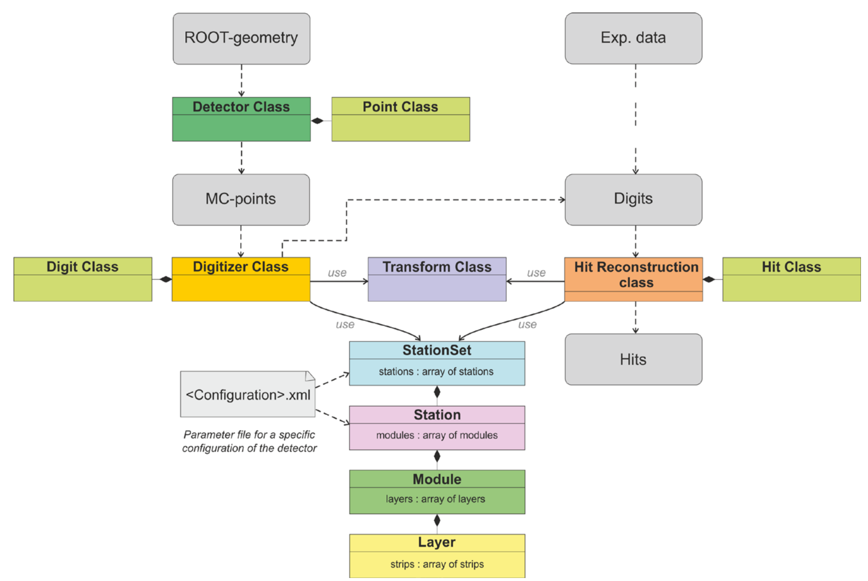
Structure of tracking detector in BmnRoot

Data processing cycle for tracking detectors in BmnRoot:



1. **Complete description of a detector**:
   a) Description of detector geometry (ROOT files)
   b) Description of detector parameters (XML files)
2. **Simulation**:
   a) Monte-Carlo simulation
   b) Simulation of realistic effects
3. **Procedures of getting "hits"**:
   a) Smearing Monte-Carlo points (hit producing)
   b) Hit reconstruction from "digits":
      - Realistic simulation + digitization
      - RAW experimental data + digitization

Software for the tracking detectors of the BM@N setup, including Monte-Carlo and hit reconstruction procedures have been implemented in BmnRoot and are ready to be used



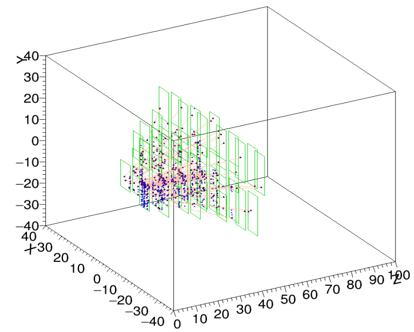
Software for the tracking detectors (class diagram)

Conclusion: what has been done

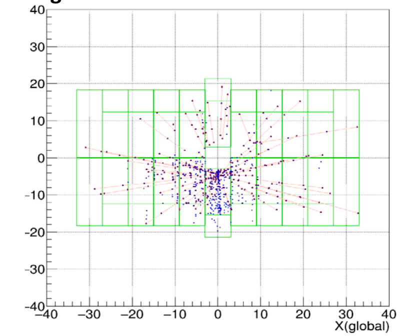
What we have done for the tracking detectors (RUN-8 configuration) at this moment:

- ✓ Complete geometry description (basic and detailed versions for each detector)
- ✓ Algorithm for realistic Monte-Carlo simulations
- ✓ Algorithm for the reconstruction of spatial coordinates from microstrip readout planes (hit reconstruction)

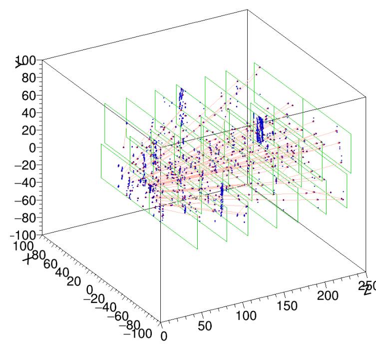
Example of a single event as a results of Monte-Carlo simulation and hit reconstruction for the three tracking detectors



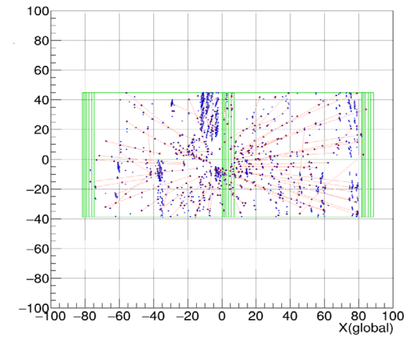
Forward Silicon *detector (3D view)*



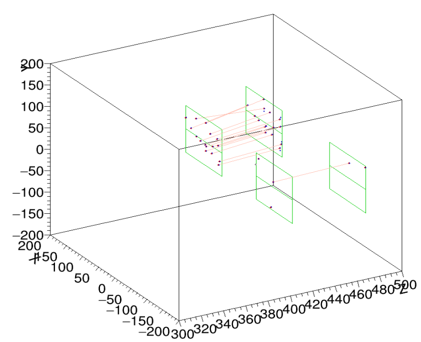
Forward Silicon *detector (XY view)*



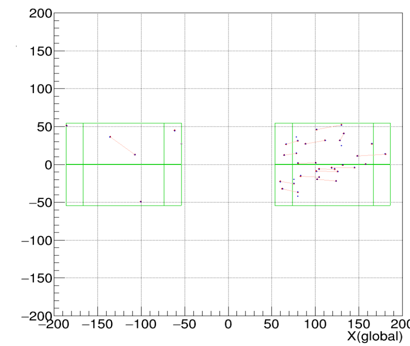
GEM *detector (3D view)*



GEM *detector (XY view)*



CSC *detector (3D view)*



CSC *detector (XY view)*

In the pictures:
Red marks are MC- points;
Blue marks are reconstructed hits;
Green rectangles are borders of sens. areas

Thank you for your attention…