

Development of the BM@N Web Monitoring

Ilnur R. Gabdrakhmanov^{1,*} and Sergei P. Merts¹

¹ Veksler and Baldin Laboratory of High Energy Physics, Joint Institute for Nuclear Research,
141980 Dubna, Moscow Region, Russia

Abstract. The BM@N experiment is a fixed target experiment and the first stage of the NICA project. Its development is of great importance for the NICA experiment as a whole. In order to effectively conduct the experiment, a convenient and unified monitoring tool is needed. The monitoring system backend is based on the *FairRoot* package while its frontend uses the CERN *jsROOT* library. The user is able to monitor any detector subsystem, select specific detector station, plane, time or strip profile histograms in 1/2/3D view. The QA functions currently are presented by reference run auto-selecting and consequent overlaying histograms.

1 Introduction

The state of nuclear matter at moderate energies became the central point of interest of modern theoretical and experimental studies in particle physics. On the one hand, it is the kinematical region of the nonperturbative QCD. On the other hand it is the physics taking place inside neutron stars. The matter being compressed to high densities turns into quark-gluon plasma (QGP) – the least studied state of matter. That is why much effort has been put into the systematic study of the heavy ion collisions.

The Nuclotron based Ion Collider fAcility (NICA) and the experiments to be done on it are being built at the Veksler and Baldin Laboratory of High Energy Physics (VBLHEP) of the Joint Institute for Nuclear Research (JINR). They are dedicated to the study of the properties of the baryonic matter under extreme density. Baryonic Matter at Nuclotron (BM@N) [1, 2] is a fixed target experiment which utilizes the beam produced by the Nuclotron-M. It is the first experiment at the NICA complex. Its setup consists of a dipole analyzing magnet with a high precision tracking system which includes several consequent silicon, GEM and CSC planes, two time-of-flight detectors, drift chambers, proportional chambers and calorimeters. For the short range correlation experiment (SRC), the setup was also equipped with the LAND – neutron detector.

In order to conduct studies on this multidetector facility a fast and flexible online monitoring system is being developed. Thus the main objectives of such systems are performance, flexibility in order to be able of implementing new detectors as the facility setup changes and the possibility of fine grain data selection and comparison. In case of the BM@N experiment it is utterly important since it will investigate high multiplicity gold-gold collisions which will impose hard requirements on system's performance and presentation capabilities. Similar data quality monitoring systems are developed on all modern particle experiments. For example there are complex multipurpose data

*e-mail: ilnur@jinr.ru

analysis and monitoring systems such as Go4 [3] and ATLAS TDAQ [4]. Although these systems provide wide variety of options the task on seamlessly embedding each of them would be nontrivial.

The system is implemented as part of the *BmnRoot* framework, a software package for simulation, reconstruction and analysis of the BM@N experiment data. The *BmnRoot* in turn is built on top of the *FairRoot* framework developed for the FAIR experiment facility based in GSI.

2 Data flow

The raw data on every event is collected and aggregated by the Data Acquisition (DAQ) system, it is written in data files and also mirrored via the TCP stream. The monitoring system is capable of using both input channels. The system is functionally organized as follows (see Fig. 1). The raw data decoder and the web monitoring were implemented as separate processes for the purpose of flexibility. The single raw data decoder can produce data for several monitoring processes which can run either on a same machine or on a different ones.

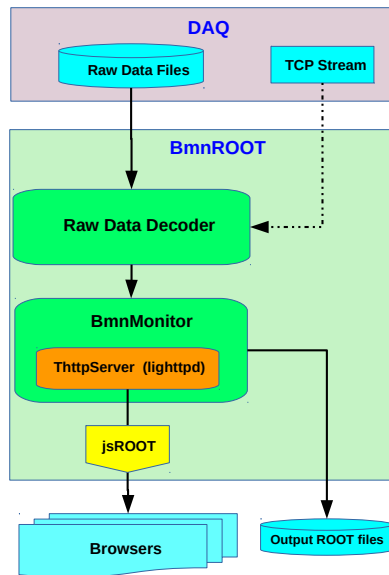


Figure 1. Data processing workflow

The raw data decoder includes two functional elements implementing two subsequent data operations: the raw data converter – parsing raw binary data into *ROOT* format so-called DAQ-digits (ADC, TDC, TQDC, HRB, ...) containing information in terms of electronic signals, and the data decoder – decoding “DAQ-digits” into the detector digits. The decoding workflow includes filtration, executing noise reduction, applying the channel-strip mapping and making preparation for a subsequent physical analysis. The processed data of each event is then sent to the monitoring process via the *ZeroMQ* [5, 6] pub socket. The *ZeroMQ* library adds a level of abstraction above the Unix sockets. Namely it implements implicit caching and message queue managing and also automatically reconnects in case when clients come online and offline. These advantages significantly simplify the system development and make it more flexible, allow system scaling and also increase the speed in some cases.

The data decoding currently implemented as single threaded. In case of the input buffer overflow the decoder is designed to drop the incoming events. During the Spring 2018 Run the decoding rate was ~ 300 Events/s varying depending on different beam particles and target material. In the future runs when Au-Au collisions will be investigated and the new inner tracking planes will be installed the data rate will grow by an order of magnitude. Thus we plan to parallelize the decoder in the nearest future.

3 Frontend

The monitoring process in turn receives the data from the *ZeroMQ* sub socket and fills the histogram sets for each detector subsystem. The histogram sets are registered on the *lighttp* [7] server by the *ROOT THttpServer* class [8] in order to be available on network http request. But still they are just *ROOT* objects and cannot be displayed on the user's computers without *ROOT* framework installed. That is why the *jsROOT* [9] library is used which is capable of converting *ROOT* objects into html ones. This scheme works as follows: the js script inside the user's browser requests the *ROOT* object from the monitoring process, receives it and draws it using the *jsROOT* library, repeats this cycle instantly as the new events come.

The system quality assurance (QA) functionality includes the reference run autoselection with superimposing the corresponding histograms in order to be able detect discrepancies and detector failures. The list of runs similar to the current ones is taken from the ELOG database considering beam energy, target material and thickness, trigger setup and magnetic field inside the analyzing magnet.

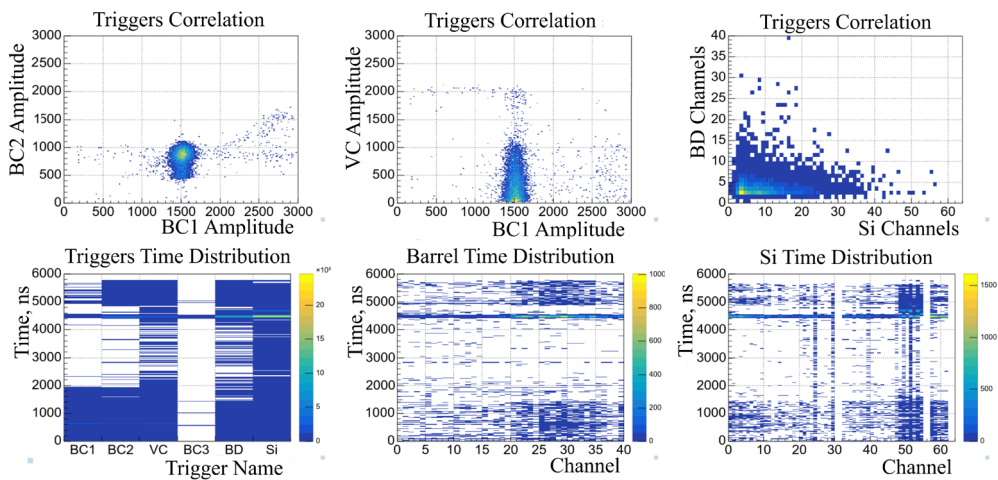


Figure 2. Example of the monitoring histograms. On the first line one can see the correlation plots of the corresponding beam counters. On the second line there are time distribution plots: several triggers vs time, Barrel Detector by channels vs time and Si Trigger by channels vs time respectively.

The users are able to select the reference run from the dropdown list on top of the page and then to superpose the current run with the reference one in order to find out the difference in a working mode or a malfunction. It is also possible to select a specific station, a plane or a strip histogram as well

as the time distribution for all channels or for a specific one. The *jsROOT* interface inherits most of the *ROOT* histogram interface capabilities. The histogram axes can be switched to log scales, 2D ones can be switched to 3D view. It is possible to select and zoom a specific histogram area and save it as a picture. All detector subsystems are included in the monitoring. Namely: triggers (beam counters, Silicon Forward Trigger, Barrel Detector), inner tracker (silicon, GEM and CSC planes), time-of-flight detectors (ToF400, ToF700), drift chambers (DCH1, DCH2), calorimeters (ZDC, ECAL).

Examples of output histograms are shown in the Fig. 2. The trigger subsystem is selected as the most illustrative one being at the same time quite simple in terms of logical structure and decoding. The correlation histograms placed on the first line allow observers to make assumptions about event multiplicity and be sure that the triggers work correctly. The time distribution plots placed on the second line show in the aggregated form that all trigger channels are synchronized and the decoding is done correctly. For example at the Si Trigger time distribution plot several channels are switched off and received no signals as expected. Also the narrow band about 4500 ns is visible on all working channels so there are no time shifts.

4 Conclusion

In the present short article we reviewed the software framework dedicated to the physical analysis and monitoring of the NICA-BM@N experiment. The web monitoring system, which is currently under development, was described in detail with the corresponding programming tools and libraries. The system is flexible by design and capable of providing online relevant and detailed information about each detector subsystem and the data quality checking. Future plans for the system development include the data decoding parallelization and the implementation of a full reconstruction chain with the online *Event Display* together with future QA improvements.

Acknowledgements

We would like to thank I.V. Slepnev, I.A. Filippov and A.V. Shchipunov for technical assistance and consulting. The work was supported by RFBR grant 18-02-40036.

References

- [1] “*BM@N–Baryonic Matter at Nuclotron*,” *Conceptual Design Report*. (2013), nica.jinr.ru/files/BM@N/BMN_CDR.pdf
- [2] D. Baranov, M. Kapishin, T. Mamontova, G. Pokatashkin, I. Rufanov, V. Vasendina, A. Zinchenko, *KnE Energ. Phys.* **3**, 291 (2018)
- [3] J. Adamczewski, M. Al-Turany, D. Bertini, H.G. Essel, S. Linev, *eConf C0303241*, MOLT003 (2003), physics/0306022
- [4] C. Haerberli et al., *IEEE Trans. Nucl. Sci.* **51**, 585 (2004)
- [5] www.zeromq.org
- [6] P. Hintjens, *ZeroMQ: Messaging for Many Applications* (O’Reilly Media, 2013), ISBN 978-1782161042
- [7] www.lighttpd.net
- [8] J. Adamczewski-Musch, S. Linev, *J. Phys. Conf. Ser.* **664**, 062032 (2015)
- [9] B. Bellenot, S. Linev, *J. Phys. Conf. Ser.* **664**, 062033 (2015)