The Online DQM of BESIII

Xiaobin Ji[1,2,*], *Yanjia* Xiao[1,], *Jiada* Lu[1,], and *Fei* Li[1,3]

[1]Institute of High Energy Physics, Chinese Academy of Sciences, Beijing, China
[2]University of Chinese Academy of Sciences, Beijing, China
[3]State Key Laboratory of Particle Detection and Electronics, Beijing, China

Abstract.
The BESIII is a general-purpose experiment for studying electron-positron collisions at BEPCII which is located at IHEP, Beijing, China. It works in the τ-charm region mainly. Several world's largest samples in this region had been collected. The BESIII DQM is a lightweight online data quality monitoring (DQM) solution at BESIII. It uses the full offline reconstruction software to reconstruct a part of data for real-time monitoring the data quality. The document gives an overview of the BESIII DQM system, including the framework, main components and data flow. The DQM system separates online DAQ and offline software environment as much as possible and is easy to expand.

1 Introduction

The BESIII detector is a magnetic spectrometer [1] located at the Beijing Electron Positron Collider (BEPCII) [2]. The cylindrical core of the BESIII detector consists of a helium-based multi-layer drift chamber (MDC), a plastic scintillator time-of-flight system (TOF), and a CsI(Tl) electromagnetic calorimeter (EMC), which are all enclosed in a superconducting solenoidal magnet providing a 1.0 T (0.9 T in 2012) magnetic field. The solenoid is supported by an octagonal flux-return yoke with resistive plate counter muon identifier modules interleaved with steel. The acceptance of charged particles and photons is 93% over 4π solid angle. The charged-particle momentum resolution at 1 GeV/c is 0.5%, and the dE/dx resolution is 6% for the electrons from Bhabha scattering. The EMC measures photon energies with a resolution of 2.5% (5%) at 1 GeV in the barrel (end cap) region. The time resolution of the TOF barrel part is 68 ps, while that of the end cap part is 110 ps. The end cap TOF system is upgraded in 2015 with multi-gap resistive plate chamber technology, providing a time resolution of 60 ps [3].

BESIII works in the center of mass energy between 2.0 and 4.6 GeV. The maximum luminosity of BEPCII is 1.0×10^{33} cm^{-2} s^{-1}. The trigger rate can reach up to 4 KHz, and the peak data rate is about 40 MB/s. BESIII started data taking from 2009. It has already collected a huge amount of data in the $\tau-$ charm region, including 1.3 billion J/ψ, 0.5 billion $\psi(3686)$, $\psi(3770)$ data corresponding to an integrated luminosity of 2.9 fb^{-1}, data for XYZ particles study and R scan, and so on.

The physics at BESIII [4] includes spectroscopy studies as well as search for new physics and high precision measurement. High-quality experimental data is very important in order to achieve such goals.

---

*e-mail: jixb@ihep.ac.cn

2 Data quality at BESIII

Data quality monitoring (DQM) is an important aspect of every modern high energy physics experiment. It is used to monitor the detector performance and the quality of data. It helps shifters and experts to find the problems quickly.

At BESIII, data quality is monitored in three steps and operated in three different subsystems: data acquisition system (DAQ) [1], DQM, and data quality assurance (DQA). During data taking, events are assembled in raw data format and then flushed into persistent storage in the online environment. This is the work of the DAQ system. The DAQ also provides the hit-maps of each sub-detector and electronics. It uses all the acquired raw data with no reconstruction. It is the fastest method and can find serious problems in real time. But the DAQ system only is insufficient to monitor the data quality, BESIII DQM [5, 6] is developed for high-level data quality monitoring. Different from the DAQ system, the DQM system only uses part of the data for monitoring, but it takes advantage of the full offline reconstruction algorithms to reconstruct the data. It can provide more information about the data quality in real time. Since the DQM system only uses part of data and uses recent calibration constants to reconstruct data, it can only be used for the fast monitoring. However, more accurate data quality results are needed. After the data is calibrated and reconstructed in the offline environment, the final data quality monitoring results are obtained with the DQA system. The document will focus on the DQM system at BESIII.

3 DQM Systems

The hardware of the BESIII DQM system includes one server machine, five computing nodes, and two PC clients. The server is an IBM x3650 M4 machine with Intel Xeon E5-2630, 2.3 GHz, which is used for the DQM system management and virtual machine host. Computing nodes are IBM Flex System x240 machine with Intel Xeon E5-2620, 2.0 GHz. Each node has 24 CPU cores, which are used for the data reconstruction and analysis. PC clients are used for the results display. Data communications of the DQM machines are controlled by Cisco Gigabit Ethernet switches via high-speed Ethernet cable. Ganglia [7] is used to monitor the whole DQM system. The operating system is Scientific Linux CERN 6 (SLC6) mainly. The development languages are C++, python, and bash.

The BESIII DQM software framework was developed based on the framework of the ATLAS DQMF [10] and the BESIII offline software system (BOSS) [8]. The main components include a DQM server, DQM clients, histogram handling, database, and a results display. The DQM server and clients are the underlying components, and they are designed with a Client/Server structure.

3.1 DQM Server

The DQM server is a TCP server. The aim of it is to provide data to all DQM clients. It is running on the DAQ environment and controlled by DAQ. When a new run started, the DQM server is started as well. It sampled raw event data from DAQ data flow and provides events continuously. It is implemented with multi-threaded programming. Each thread processes a request from a DQM client.

3.2 DQM clients

DQM clients are the main component of the DQM system. They receive raw event data from the DQM server, use offline reconstruction algorithms to reconstruct the data, analyse them,

produce histograms in ROOT format, and publish these histograms to the histogram server. This is the most time-consuming component in the DQM system. The structure and the data processing flow of the DQM client is shown in Figure 1.
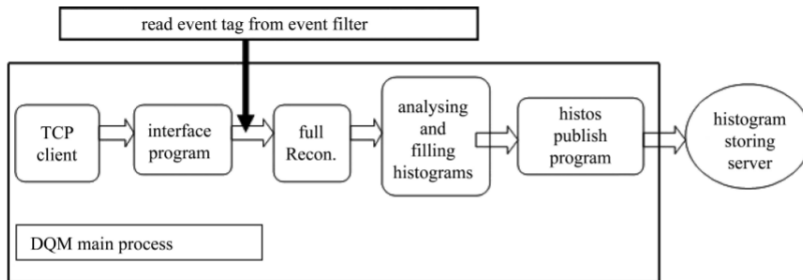
Figure 1. Data processing flow of the DQM main process – the DQM client. The DQM client reads data from the DQM data server through TCP client. The interface program invokes the offline reconstruction algorithms to do the full reconstruction. The event tag from event filter can be used before the full reconstruction to control which kind of events should be reconstructed. The reconstructed events are analyzed and the monitoring variables are filled in the histograms. Finally, the histogram publish program sends all the histograms to the histogram storing server.

The DQM client is also a multi-threaded program and developed under the offline software environment. One thread is used to receive data from the DQM server via the established TCP/IP connection. One thread calls BOSS environment in which the data is reconstructed and analyzed, just as in the offline environment. And in order to enhance the robustness of the program, an extra thread manager is introduced to recover the broken connection. Furthermore, a shared queue is used to store and buffer events received from the server.

Many DQM clients can be run simultaneously, typically one CPU core runs one DQM client job. In the current setting of BESIII, there are 120 CPU cores totally in the computation nodes, so 120 DQM client jobs can be run simultaneously.
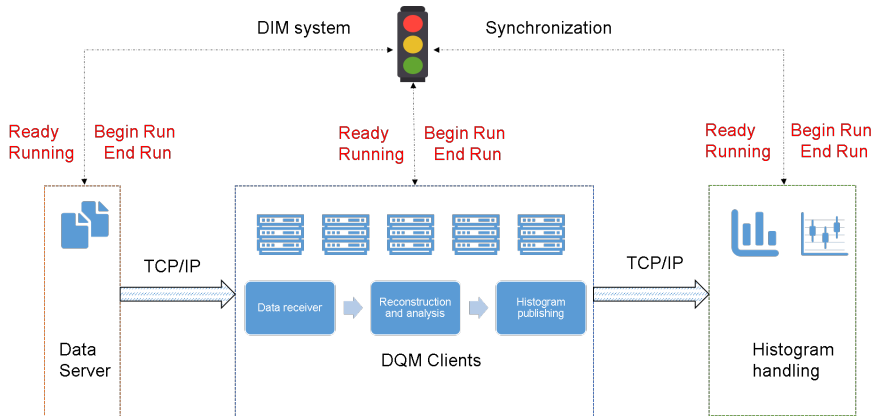
3.3 Histogram handling

The histogram handling component includes the histogram server, the histogram receiver, the histogram merger, and the histogram display. They are developed based on ROOT [9] and part of the ATLAS DQMF package [10]. The histogram server stores all histograms generated in the DQM system and saves histograms into a root file when a run is finished. The histogram receiver is used to receive histograms sent by the DQM client and to publish them to the histogram server. One histogram receiver is responsible for one DQM client. The histogram merger merges histograms with the same name from all DQM clients into a new one and publishes the merged histogram to the histogram server. The merge interval can be defined by a user. Finally, the histogram display program fetches the merged histograms from the server and displays a few selected ones to shifters.

If too many processes send histograms to the histogram server at the same time, I/O is a potential problem. In this case, the multi histogram server is necessary. One server is responsible for part of the processes. Although the DQM client can publish histograms to the histogram server directly, there is still some delay when the histogram server is busy.

Introducing the histogram receiver can solve the problem and improve the performance of the DQM client.

3.4 Job control

DQM is designed to run in standby mode. When the data taking is started, it starts running automatically. The Distributed Information Management System (DIM) [11] is used to synchronize all DQM components. It synchronizes the state of data taking (Begin Run, Run, End Run, and Wait) from the DAQ DIM server and sends the commands to related DQM components.

Figure 2. Overall structure of the DQM system. It includes three parts: the DQM (data) server, the DQM clients, and the histogram handling program. They are synchronized with DIM. Data transfer between different components uses the TCP/IP protocol only.

The overall structure of the DQM system is shown in Figure 2. DQM clients start/stop automatically when a run start/stop. After a run is finished, all merged histograms are stored in a root file. A separate job will deal with the root file, obtains useful information of the run, and put them into the database (MySQL).

3.5 Information display

DQM results are displayed using different methods. Using the offline event display package BesVis, the reconstructed events can be shown in real-time. The Online Histogram Present (OHP) is used for the real-time histogram monitoring. Run information, such as the center-of-mass energy, cross section, Interaction Point (IP) position, and so on which are calculated after the end of a run, can be checked from the web page. Real-time IP information is sent to DIM too so BEPC monitoring system can easily retrieve it.

3.6 Job management

After the DQM system is started, the jobs are running automatically. It is important for the system to be robust. All TCP connections are protected by a time-out mechanism. Several daemon processes are running in order to recover the process from unexpected errors. A crashed job can be restart automatically. Sometimes, some jobs have no response and do not

update the histograms any more, but they do not crash. Different methods are used to find them in time. The CPU usage of the machine with dead jobs is lower than the normal value. Additionally, job statistics and monitoring of the run number are added. Job statistics shows the number of processed events for each job while the run number monitoring shows the processing run of each job. The dead job can be found in a short time using these methods.

4 Summary

BESIII DQM is a lightweight online DQM solution. The design concept of DQM is expandable and easy maintenance. The framework adopts the Client/Server structure. It samples data from DAQ data flow. The offline software environment is integrated into the system. Offline algorithms can be used directly or with just minor modification. So the DQM system can be easily used to monitor the detector and data quality with full reconstructed events. The three components, DQM server, clients, and histogram handling are stand-alone programs and can be run in the different computers. They are synchronized with DIM. Data transfer between different components uses the TCP/IP protocol only. The system has been running successfully for several years at BESIII.

References

[1] M. Ablikim *et al.* [BESIII Collaboration], Nucl. Instrum. Meth. A **614**, 345 (2010).

[2] C. H. Yu *et al.*, Proceedings of IPAC2016, Busan, Korea, 2016, doi:10.18429/JACoW-IPAC2016-TUYA01.

[3] X. Li *et al.*, Radiat. Detect. Technol. Methods **1**, 13 (2017); Y. X. Guo *et al.*, Radiat. Detect. Technol. Methods **1**, 15 (2017).

[4] D.M. Asner *et al.*, International Journal of Modern Physics A Volume: **24**, Issue: 1 supp (2009)

[5] J.F. Hu, Y.H. Zheng, X.D. Sun, X.B. Ji, Chin. Phys. **C36**, 62 (2012)

[6] X.D. Sun, J.F. Hu, H.S. Zhao, X.B. Ji, Y.F. Wang, Y.H. Zheng, B.J. Liu, Chin. Phys. **C36**, 622 (2012), `1109.3487`

[7] Ganglia project, Available from http://ganglia.sourceforge.net.

[8] W.D. Li, H.M. Liu *et al.*, in proceeding of CHEP06, Mumbai, India, 2006.

[9] ROOT project, Available from https://root.cern.ch.

[10] S. Kolos, A. Corso-Radu, H. Hadavand, M. Hauschild, R. Kehoe, Journal of Physics: Conference Series **119**, 022033 (2008)

[11] DIM project, Available from http://dim.web.cern.ch/dim/.