Online data quality monitoring system at BESIII *

SUN Xiao-Dong(孙晓东)[1,2;1)] HU Ji-Feng(胡继峰)[1,2;2)] ZHAO Hai-Sheng(赵海升)[1,2)]
JI Xiao-Bin(季晓斌)[1] WANG Yi-Fang(王贻芳)[1] ZHENG Yang-Heng(郑阳恒)[2]
LIU Bei-Jiang(刘北江)[1]

[1] Institute of High Energy Physics, Chinese Academy of Sciences, Beijing 100049, China
[2] Graduate University of Chinese Academy of Sciences, Beijing 100049, China

Abstract: The online Data Quality Monitoring (DQM) tool plays an important role in the data recording process of HEP experiments. The BESIII DQM collects data from the online data flow, reconstructs them with offline reconstruction software and automatically analyzes the reconstructed data with user-defined algorithms. The DQM software is a scalable distributed system. The monitored results are gathered and displayed in various formats, which provides the shifter with current run information that can be used to identify problems quickly. This paper gives an overview of the DQM system at BESIII.

Key words: BESIII, DQM, sampling, histogram

PACS: 29.85.-c, 29.85.Ca **DOI:** 10.1088/1674-1137/36/7/010

1 Introduction

BESIII is a detector operating on the Beijing Electron Positron Collider (BEPCII) at the Institute of High Energy Physics (IHEP) of the Chinese Academy of Sciences in Beijing. With a BEPCII design luminosity of 10^{33} cm$^{-2}\cdot$s^{-1}, BESIII will collect a large dataset so that τ-charm physics can be studied with high precision [1].

The peak luminosity recently exceeded 6×10^{32} cm$^{-2}\cdot$s^{-1}. At such a high luminosity, it is essential to monitor the status of the BESIII hardware and to assure the quality of acquired data in real time. In an online environment, the DAQ (Data AcQuisition) system uses all the acquired data to get preliminary information about sub-detectors. However because of multi-beam bunches and a pipelined readout electronics system, some information, such as event start time, can not be obtained by the DAQ system. An offline reconstruction can give much more information about the detector performance and data quality, but its results will not be available until several days after the data has been taken. To monitor data quality both accurately and in real time, an online Data Quality Monitoring system (DQM) has been developed. The DQM fully reconstructs a portion of the acquired data, which is sampled randomly from the online data flow, using the offline full reconstruction software. The monitored results showing the detector status and data quality will be available only a few minutes after one run begins.

2 Features and operating environment

The BESIII online DQM system consists of 8 nodes: 5 IBM eServerBlade HS20, each with dual 3.0 GHz Xeon CPUs, and 3 PCs. All the nodes have a SLC 4.6 operating system installed. The five HS20 nodes process the event data, including the event reconstruction and analysis. One of the PCs is used as a DQM system server to provide system management and DQM services, such as histogram merging and display; the second PC deals with event display; the last one is a backup machine.

The communication between the DQM and DAQ is only handled by a high speed network connection. Events copied from the online data stream are transferred one by one to the DQM machines through

Received 15 September 2011, Revised 18 November 2011
∗ Supported by Knowledge Innovation Project of CAS (KJCX2-YW-N29)
1) E-mail: sunxd@mail.ihep.ac.cn
2) E-mail: hujf@mail.ihep.ac.cn

the network without influencing the online data flow. Only partial event data is sampled. The sampling rate depends on the processing capacity of the DQM programs. The most time consuming process is the data reconstruction. Currently, the maximum sampling rate can reach 360 Hz, among which the ratio of physics events (Bhabha, Dimuon and Hadron events) is about 40–60 Hz.

The BESIII DQM software framework was developed based on the framework of the ATLAS DQMF [2] and BESIII offline software system (BOSS). The work flow is redesigned to ensure the separation from the DAQ software system and the automatic control of data processing in the BESIII environment. Only one data server is required to run the online DAQ system. The data server samples data from the online data flow and then sends the data to the DQM for processing by TCP connections. This design ensures the safety of the DAQ system, additionally, the DQM system integrates with BOSS fully. Almost all of the offline algorithms can run in a DQM environment with minor or no modifications. The DQM system is a distributed system, in which there are 20 DQM main processes in total running on the HS20 nodes. The histograms generated by each main process are merged by the histogram merger process in real time. The merged histograms can then be checked during data collection and are stored in ROOT [3] files after a run ends. Detector properties such as the TOF time resolution and MDC momentum resolution obtained during each run are stored in the DQM database for use in monitoring the detector performance.

3 The architecture of DQM

The framework of the DQM [4] is shown in Fig. 1. The DQM system mainly consists of 6 parts, i.e. a DQM server, DQM clients, a merger, a histogram storing server, a DQM database and information display programs. The DQM server is used to get data from the online data flow. The DQM clients invoke reconstruction algorithms and data analysis algorithms to reconstruct events and fill histograms. The merger is used to merge histograms from different DQM clients. The histogram storing server is used to store all histograms. The DQM database is used to store parameters that Reflect the detector properties and the data quality of each run. Information display programs are used to display information that is useful for monitoring data quality.

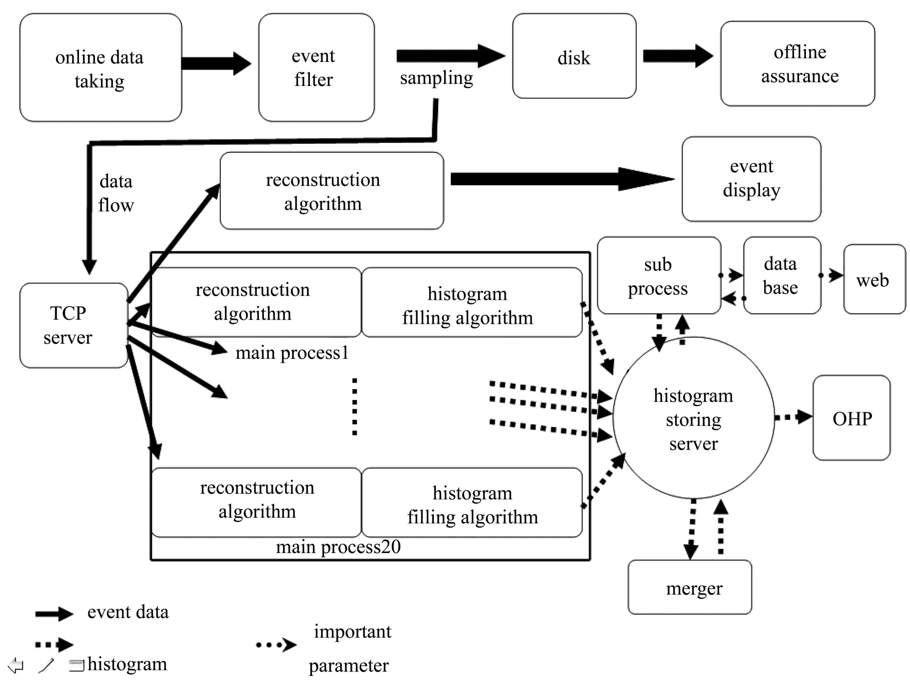

Fig. 1. The framework of the DQM.

3.1 Basic work flow of the DQM

The event fragments received from different sub-detectors are assembled into full events before classifying them by the event filter [5]. A TCP server, which is called the DQM server, is used to sample events passing through the event filter and then to deliver them to different DQM clients.

The DQM client reconstructs events to get basic status information for each sub-detector. A small part of the reconstructed events are displayed directly by the event display program, while the other reconstructed events are used to fill the first-level histograms (histograms filled with information on the event level). The first-level histograms from different DQM main processes are merged by a histogram merger after they are published to the histogram storing server. Some of the merged histograms will be displayed by the online histogram presentation program (OHP) directly and others will be used to generate the second-level histograms (histograms filled with information extracted from the first-level histograms) to be displayed by the OHP. Some global parameters of sub-detectors, such as momentum resolution, time resolution, event vertex and so on, are extracted from the histograms at the end of the run and are stored in the DQM database.

3.2 The DQM main processes

There are 22 DQM clients currently, running either the main processes or the event display. Twenty of them are DQM main processes which fully reconstruct the events and then fill the first-level histograms using the information in the reconstructed events. A DQM sub-process is used to handle the first-level histograms at the end of a run. The event display program fully reconstructs the events and displays it visually in real time, as shown in Fig. 2.

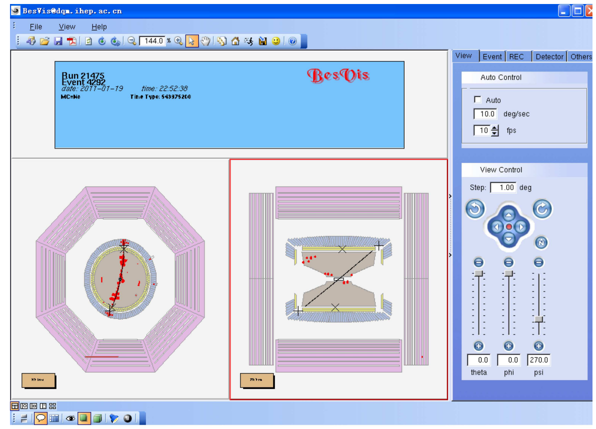

Fig. 2. A Bhabha event displayed by Event Display (color online).

The framework of the DQM main process is shown in Fig. 3. The DQM main process can be divided into 5 functional units. The TCP client is used to fetch the events from the DQM server. Then the interface program unpacks the data and supplies events in the format suitable for the reconstruction algorithms. The DQM uses the offline reconstruction algorithms to fully reconstruct the events. Histogram-filling algorithms (user defined algorithms) use information from the reconstructed events to fill histograms. The histogram publishing program publishes histograms to the histogram storing server.

To improve the processing speed, the KalFitAlg algorithm [6] used for track fitting is not used in the DQM online reconstructions, which is different from the offline reconstructions. As a result, the time spent on event recontruction decreases by almost 50% while the precision of the DQM result is slightly worse than the offline one. Another significant difference between the DQM reconstruction and the offline reconstruction is that the DQM reconstruction uses the calibration constants obtained from recent runs.

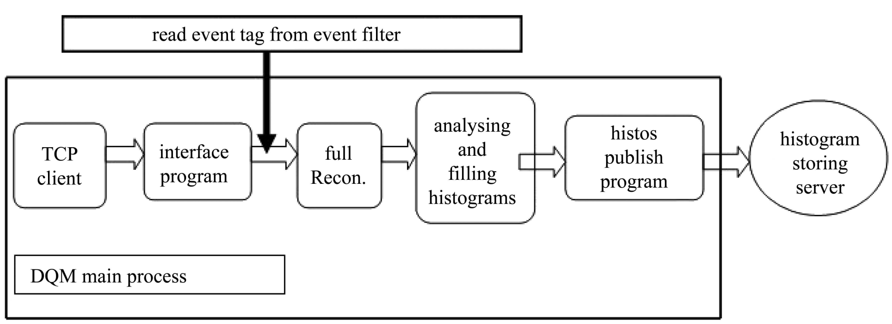

Fig. 3. Data processing flow of DQM main process.

In order to further improve the process capacity of the DQM system, event tags given by the event filter are checked by a specific algorithm, which is executed before the event reconstruction and only events with specific event filter tag are reconstructed. The processing rate of the DQM is improved by about 44% under the current configuration.

3.3 Histogram-filling algorithms (user defined algorithms)

The DQM main process invokes user-defined histogram-filling algorithms to fill histograms. The algorithms are flexible and independent of each other. Each of the histogram-filling algorithms can be added or removed easily, which ensures the extensibility of DQM system.

Many histogram-filling algorithms have been developed to fulfil the task of monitoring. A special event tag algorithm is used for event classification using the reconstructed information. Each event processed by the QM is tagged as a specific type such as Bhabha, Dimuon, Hadron, Cosmic Ray and so on, for future use.

Main histogram-filling algorithms are used to monitor the sub-system of BESIII, including the MDC (Main Drift Chamber), TOF (Time-of-Flight counters), EMC (Electro-Magnetic Calorimeter), MUC (Muon Counter), and trigger systems. All these algorithms can use event tags from the event tag algorithm to fill histograms with selected event types only. Furthermore, several histogram-filling algorithms related to special physics channels, such as inclusive K_S, inclusive J/ψ in ψ' data, D mesons in ψ'' data, and so on, are applied in order to use more physics variables for data quality monitoring.

Each of the detector-related alogithms has a sub-algorithm to fill the second-level histograms using the first-level histograms it created. The DQM sub-process invokes sub-algorithms to extract information in the merged first-level histograms and to get parameters such as detecting efficiency, noise ratio, energy resolution, etc. It then fills the second-level histograms with these parameters. Some important parameters are further stored into the DQM database which is a subset of the BESIII offline database [7]. These parameters are used to draw a historical curve in order to make comparisons.

Integral luminosity, event start time, event vertex etc. are also monitored in the DQM system, which are sensitive to the beam condition. The accelerator monitoring system can get them from the DIM [8](Distributed Information Management System). Fig. 4 shows the distribution of event start time with different beam intervals (8 ns left, 4 ns right). The shifter can find the change in beam interval quickly during data collection.

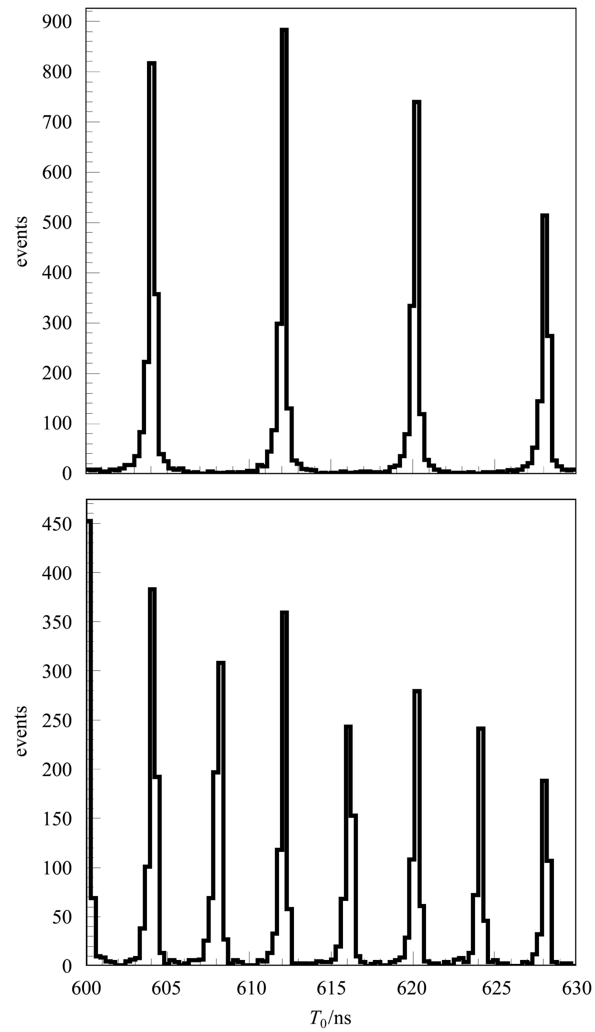

Fig. 4. The distributions of event start time with different beam intervals.

4 Cooperations between DQM processes

The DQM system is an automatic data collection system. The flow control system, based on TCP/IP sockets, controls different components of the DQM system to ensure that they all take the right action at any time.

When taking data during a run, the DQM server will automatically send event information to the DQM main processes by the TCP connections. Once receiving event information, the DQM main processes will reconstruct them and generate first-level histograms.

At the end of a run, the DQM main processes will publish all histograms of this run into the histogram storing server and then send a TCP signal to the merger. After receiving the TCP signals from all 20 DQM main processes, the merger will merge the histograms in the histogram storing server, and then send a TCP signal to the DQM sub-process. After that, the DQM sub-process will fetch the first-level histogram from the histogram storing server, generate the second-level histogram, publish the second-level histogram to the histogram storing server and then send a TCP signal to the DQM histogram storing server. Lastly the DQM histogram storing server will store all the histograms from the run into a ROOT file after receiving the TCP signal from the DQM sub-process.

All TCP connections are protected by a time-out mechanism. The DQM processes with TCP connections will wait for TCP signals before taking any further steps. However, if a TCP signal is not received after a fixed time, they will act without a TCP signal so that although some DQM processes are broken; the other processes will still work.

Several daemon processes are running in order for the DQM system to recover from unexpected errors. Once a DQM process crashes, it will be restarted automatically.

5 Information display

The DQM results are viewed in three ways: by event display, histogram display and web display.

The event display program is modified according to the offline version in BOSS. The reconstructed events can be displayed automatically in real time. Fig. 2 shows a Bhabha event displayed by the event display program. Two tracks can be seen clearly.

Histograms from different user-defined algorithms can be viewed easily using OHP and OHD tools. Important histograms for both sub-detectors and typical physics processes are displayed on the OHP. All the physical variables filled into these histograms are listed in Table 1.

Most of the histograms shown on the OHP are displayed with reference histograms, as shown in Fig. 5. The reference histograms come from a recent good run. The shifter can find the problem easily from the OHP when it occurs. For example, the first two histograms in Fig. 5 do not agree with their references. Further checking points out that this is caused by a problem with the accelerator.

The OHD is a tool which can be used to check all the histograms of a current run by sub-detector experts and those people who are authorized. Various ROOT tools can also be used to check the historical histograms stored in ROOT files.

Important parameters of detectors in each run stored in the DQM database can be checked in tables or by histograms on the web. People can check the momentum resolution and space resolution of the MDC, time resolution of the TOF, energy resolution of the EMC, peak value of shower energy deposited in the EMC for e^+ and e^- and the mean value of x, y, z of the event vertex. The integral luminosity of each run is calculated with Bhabha events and Diphoton events separately, and can also be checked on the web.

Table 1. The histograms produced by the OHP for data quality monitoring.

separated parts	histograms (most given X axis only, Y is the number of Bhabha events by default)
MDC	momentum of e^+ and e^-, residual [9] distribution, event start time, dE/dx, ϕ and $\cos\theta$ of e^+ and e^-, drift time in inner/outer chamber
TOF	ΔT of barrel/endcap, east barrel/west barrel/endcap hit map, barrel z of the hit position, time resolution of endcap, time difference between upper and lower TOF
EMC	shower energy deposited in barrel/endcap EMC, shower ϕ in barrel/endcap, shower ID(θ)
MUC	ϕ vs. $\cos\theta$ for all events, fired layers in MUC of tracks, event No. vs. number of MUC hits, acollinear angle distribution of the momentum of dimu tracks
Trigger	fired ADC number for barrel/endcap, scintillator ID for barrel/endcap, long track hit map, trigger channel, trigger condition
Physics	ϕ of e^+ and e^- for Barrel, $\cos\theta$ of e^+ and e^-

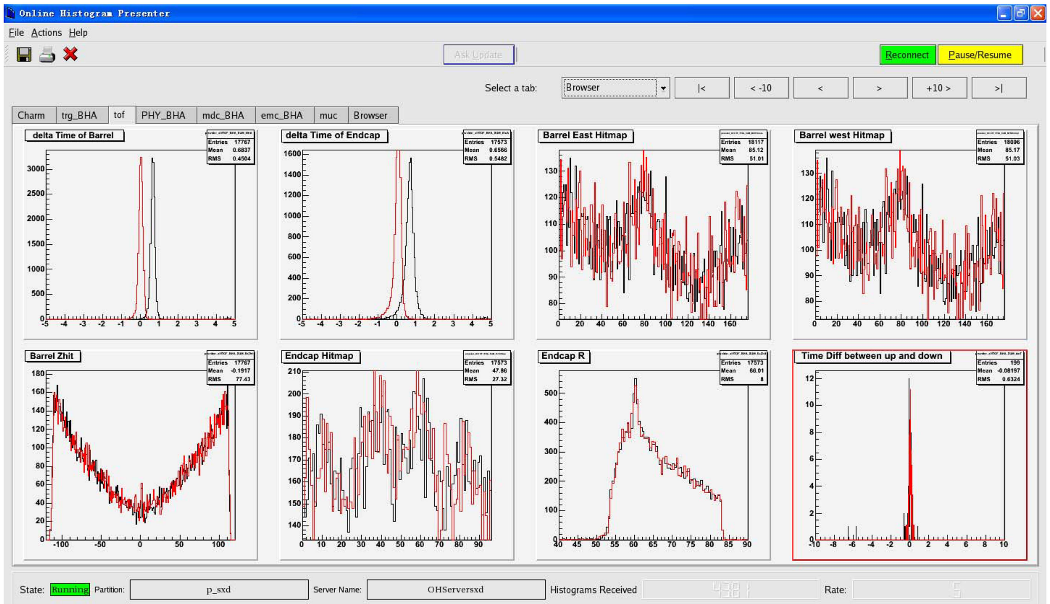

Fig. 5. (color online) A snapshot of the histograms displayed by the OHP.

6 Summary

The DQM system has been developed and implemented in BESIII. After careful tuning and updating for several months, the DQM has been running continuously and steadily during the BESIII data taking process for more than three years, since July 2008. The DQM system can monitor the sub-detector and trigger system in a much more physical and accurate way than the other online monitoring systems. DQM has become an essential part of the whole BESIII data quality monitoring system. Together with the online DAQ monitoring system, DQM ensures the successful and robust data collection and physics analysis on BESIII.

The authors would like to thank Zhu Yong-Sheng and He Kang-Lin for helpful discussion and suggestions, and acknowledge Tian Hao-Lai, Zou Jia-Heng, Wu Ling-Hui, Sun Sheng-Sen, Liu Chun-Xiu, Xie Yu-Guang, Cao Guo-Fu for their contributions to part of the DQM algorithms.

References

1 Ablikim M et al. (BES Collab.) Nucl. Instrum. Methods in Physics Research, A, 2010, **614**(3): 345–399
2 Kolos S et al. A Software Framework for Data Quality Monitoring in ATLAS. International Conference on Computing in High Energy and Nuclear Physics(CHEP' 07), 2008
3 http://root.cern.ch/
4 HU Ji-Feng et al. Chin. Phys. C (HEP & NP), 2012, **36**(1): 62
5 FU Cheng-Dong et al. Chin. Phys. C (HEP & NP), 2008, **32**(5): 329–337
6 WANG Ji-Ke et al. Chin. Phys. C (HEP & NP), 2009, **33**(10): 870–879
7 CHU Yuan-Ping et al. The Architecture of BESIII Offline Database. In Proc. IEEE NSS/MIC Nuclear Science Symp. Conf. Rec., 2009. 484–486
8 http://dim.web.cern.ch/dim/
9 WU Ling-Hui et al. Chin. Phys. C (HEP & NP), 2008, **32**(4): 265