Upgrade of Online Storage and Express-Reconstruction System for the Belle II experiment

Seokhee Park[1,*], *Ryosuke* Itoh[1], *Soh Yamagata* Suzuki[1], *Daniel* Jacobi[2], *Satoru* Yamada[1], *Takuto* Kunigo[1], and *Dmytro* Levit[1]

[1]High Energy Accelerator Research Organization (KEK), 1-1 Oho, Tsukuba, Ibaraki 305-0801, Japan
[2]University of Bonn, 53113 Bonn, Germany

Abstract. The backend of the Belle II data acquisition system consists of a high-level trigger system, online storage, and an express-reconstruction system for online data processing. The high-level trigger system was updated to use the ZeroMQ networking library from the old ring buffer and TCP/IP socket, and the new system is successfully operated. However, the online storage and express-reconstruction system use the old type of data transportation. For future maintainability, we expand the same ZeroMQ library-based system to the online storage and express-reconstruction system. At the same time, we introduce two more updates in the backend system. First, online side raw data output becomes compressed ROOT format which is the official format of the Belle II data. The update helps to reduce the bandwidth of the online to offline data transfer and offline-side computing resource usage for data format conversion and compression. Second, high-level trigger output-based event selection is included in the online storage. The event selection allows more statistics of data quality monitoring from the express-reconstruction system. In the presentation, we show the description and test result of the upgrade before applying it to the beam operation and data taking.

1 Introduction

Belle II experiment is operated to take data of electron-positron collisions provided by the SuperKEKB accelerator [1] near the energy range of the $\Upsilon(4S)$. The collision data are collected by the Belle II detector [2], which consists of subdetectors - a pixel detector (PXD), a silicon-strip vertex detector (SVD), a central drift chamber (CDC), a time-of-propagation counter (TOP), an aerogel ring-imaging Cherenkov counter (ARICH), an electromagnetic calorimeter (ECL), and a K-long and muon detector (KLM). Each subdetector's data is processed by a unified method, with the exception of the PXD due to its large data size. The collected analog data is digitized by each subdtector's front end electronics and are read-in by the level 1 hardware trigger. The level 1 trigger is synchronized to the unified timing distribution system [3]. The accepted trigger rate is up to 30 kHz by design. The digitized data are transferred to the common readout systems named "common pipelined platform for electronics readout" (COPPER) [4] or its successor PCI-E 3.0-based readout boards [5]. Then, the readout servers merge the outputs of readout boards and send them to the next part, the backend part of the Belle II data acquisition system (DAQ).

*e-mail: seokhee.park@kek.jp

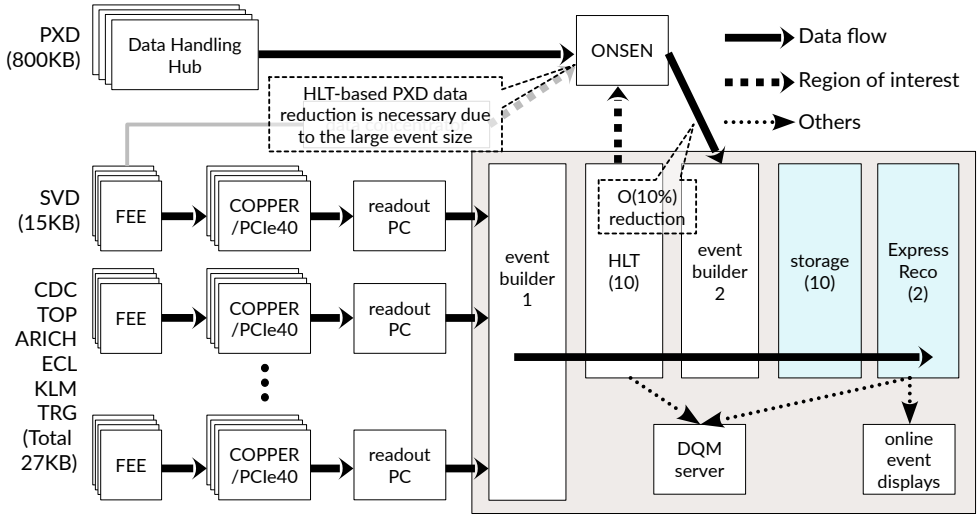

Figure 1. Schematic view of the Belle II DAQ data flow.

The backend of the DAQ consists of two network switch-based event builders [6], a high-level trigger system (HLT) [7], online data storage (STORE), and an express reconstruction system (ExpReco). The first event builder serializes the data from readout servers and sends them to the HLT. The HLT selects events by a software trigger algorithm with the full event reconstruction. The reconstructed track information, forming regions of interest, is also provided to the PXD readout to select the PXD hits associated with the tracks. As a result, the PXD data size is reduced by $\sim$ 10%. The events selected by HLT and reduced PXD data are combined by the second event builder, and the final data are transferred to the STORE. The data are saved in STORE disks and registered in an online database. The output files are regularly transferred to the KEK offline computing center (KEKCC). HLT and ExpReco generate data quality monitoring (DQM) histograms. The HLT DQM has higher statistics, but lack PXD information. On the other hand, ERECO DQM has more histograms including PXD and physics variables, but small statistics caused by the computing power. Till 2022, ERECO has $\sim$ 640 treads of CPU, 10 times smaller than HLT, $\sim$ 6400 threads. Figure 1 shows a schematic view of the Belle II DAQ dataflow [8].

HLT, STORE, and ERECO systems are built on the Belle II Analysis Software Framework (BASF2) [9]. BASF2 is written in C++ with a Python interface and has a module-and-path structure. For data input and output, the backend system uses several network applications for data transfer based on either the ZeroMQ library [10] or ring buffers and TCP/IP sockets. Currently, HLT uses the ZeroMQ library, and STORE and ERECO use ring buffers and TCP/IP sockets.

During the year 2022-2023 long shutdown period, ERECO and STORE systems were upgraded [11]. The main goal of the upgrade is the design unification. HLT was successfully operated with ZeroMQ library-based design, so we extended the same design to STORE and ERECO. The design unification improves the future maintainability and stability of the backend system. At the same time, we changed the output format in STORE to ROOT [12] from homemade sequential file format (SROOT). By unifying the online and offline output format and compression level, we can save offline computing resources and network

bandwidth for file transfer. For ERECO, a new skimming module and additional software ERECO unit (physics ERECO) were prepared for selecting physics-flagged events by HLT. The physics ERECO can dramatically increase the statistics of DQM histograms for physics-flagged events.

2 Online Storage Upgrade

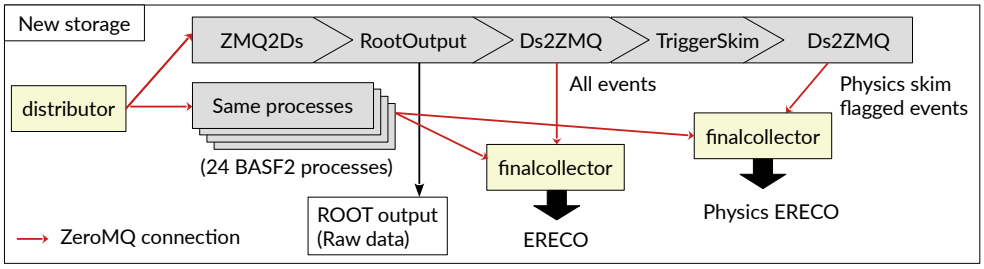

Figure 2. Structure of the new online storage system.

The new STORE consists of three parts: input application (*distributor*), many BASF2 processes, and output application (*finalcollector*). The *distributor* gets the data from readout servers via TCP/IP sockets and sends the data as ZeroMQ messages to BASF2 processes in a round-robin manner. A BASF2 process contains an input module (*ZMQ2Ds*), a ROOT output writing module (*RootOutput*), two output modules (*Ds2ZMQ*), and a skimming module (*TriggerSkim*). *ZMQ2Ds* module accepts and deserializes the ZeroMQ message from the distributor and saves them into the data object (DataStore) shared across the entire BASF2 process. A *RootOutput* module reads the DataStore and writes events into the ROOT file. The first *Ds2ZMQ* serializes the DataStore information, makes ZeroMQ messages, and sends the messages to ERECO via TCP/IP socket. After the first *Ds2ZMQ* module, the *TriggerSkim* module reads the HLT trigger selection information and discards events that do not have physics-related flags. The second *Ds2ZMQ* module serializes the physics events from the DataStore and sends them to the physics ERECO. Fig. 2 shows the structure of STORE.

One of the main differences between old and new systems is the output file format. The new STORE has several benefits compared to the old SROOT STORE. First, output files can be compressed while being written. In the general case, the Zstandard algorithm [12] is applied and the output file size will be reduced by $1/2 - 1/3$. Also, no further reprocessing is necessary from the offline side, so the computing resources of KEKCC are saved. A consequence of online compression is that large CPU power is required; we solved the issue by launching multiple BASF2 processes. The number of processes in a STORE is more than 24 and this is enough to store all physics data by design.

At every end of a run, more than 240 small files are generated in the entire DAQ backend. These many small files cause disk I/O performance degradation. Because of that, STORE also performs small file merging after the end of runs. To achieve this goal, we defined the post-processing chain based on Python scripts and databases as shown in Fig. 3. First, run start and stop are detected by the run information tool and written in the run information table. After the run starts, RootOutput modules write files and the file list is written in the 'output list' table. The run information tool detects a run stop, and the run information table is updated once again. Then, the post-processing tool catches the run stop by checking the run

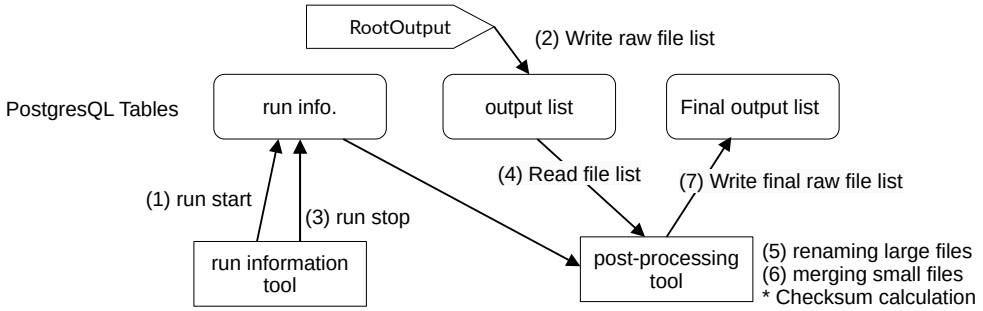

Figure 3. Schematic view of post-processing process and database tables.

information tool and doing small file merging and renaming. The file renaming is performed for large-sized files to unify the file naming convention. While performing file merging and renaming, the file checksum is calculated, too. The merged and renamed files are listed in the 'final file list' database table with the checksum. Finally, after all files are processed, the run information table is updated once again, showing that the run is ready to be sent to the KEKCC storage.

3 Express-Reconstruction System Upgrade

The HLT-selected data and PXD data are merged from STORE and transferred to ERECO. The ERECO reconstructs the full event and generates DQM histograms including PXD information and physics variables. Unlike HLT, ERECO used a TCP/IP socket and ring buffer-based framework. However, the system has several problems. For example, the DQM histogram transferred slowly and data flow corruption was not correctly monitored. The new ZMQ system can solve the instability of ERECO.

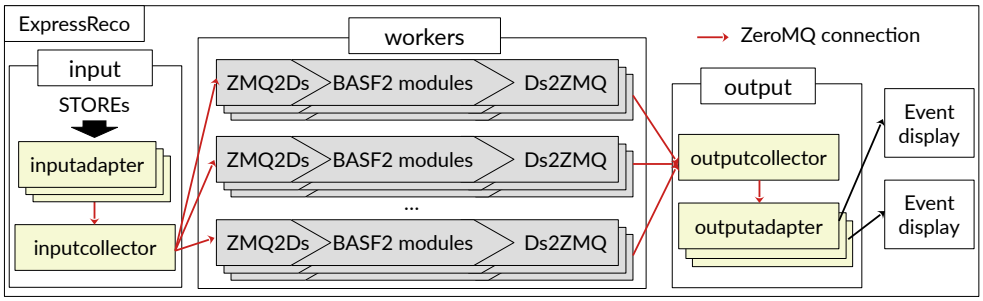

Figure 4. Structure of the new express-reconstruction system.

The new ERECO structure is shown in Fig. 4. The *inputadapter* gets the events from a STORE, and an *inputcollector* collects the data from many *inputadapters*. The number of *inputadapter*s is determined by the number of connected STORE. The *inputadapter* distributes the events to each BASF2 process operated on worker hosts in a round-robin manner. The BASF2 worker process reconstructs full events including PXD data and generates DQM

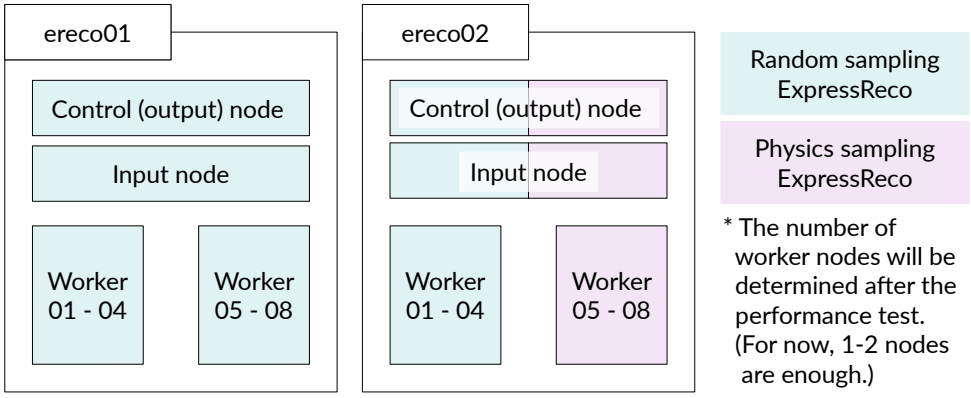

Figure 5. Implementation of express-reconstruction system for physics-flagged events.

histograms. The reconstructed data are sent to the event displays via an *outputcollector* and *outputadapter*s.

One additional feature is dedicated ERECO for physics-flagged events tagged by HLT (physics ERECO). Because the performance is ERECO is almost 1/10 of HLT, the events that are not accepted by ERECO are randomly discarded. However, this random selection is also applied to physics-flagged events. The ratio of physics-flagged events is already small, so additional random selection makes the total number of selected events minuscule. To avoid this situation, additional ERECO is prepared at the software level. Figure 5 shows the diagram of Belle II ERECO. The physics ERECO is set up in one of ERECO. So, input and output nodes are shared for both types of ERECO. A few worker nodes are dedicated to physics ERECO and the number of ERECO can be determined by the number of physics-flagged events. Finally, the physics ERECO can receive the events from STORE which have passed the *TriggerSkim* module shown in Fig 2. The number of physics-flagged events is less than 1% of the entire events. Therefore, the events can be processed without additional rejection caused by the processing power of the physics ERECO unit.

We studied how many worker nodes are needed for physics ERECO by checking the old runs. Figure 6 shows the required number of worker nodes for each combination of possible physics flags. In most cases, two worker nodes are enough to make high statistics DQM. If the situation is changed, we can easily increase or decrease the number of worker nodes by changing config files.

4 Conclusion

The Belle II experiment will resume operation at the end of 2023. For the operation, we prepared upgrades of STORE and ERECO systems. We unified the DAQ backend system design based on the ZeroMQ library and added new features. STORE can save the output data in ROOT format with online compression. This can save the computing resources of KEKCC and network bandwidth for file transfer. At the same time, it selects physics-flagged events and sends them to physics ERECO. This pre-selection from the STORE side with the dedicated ERECO unit can provide higher statistics for the DQM histograms of physics-flagged events.

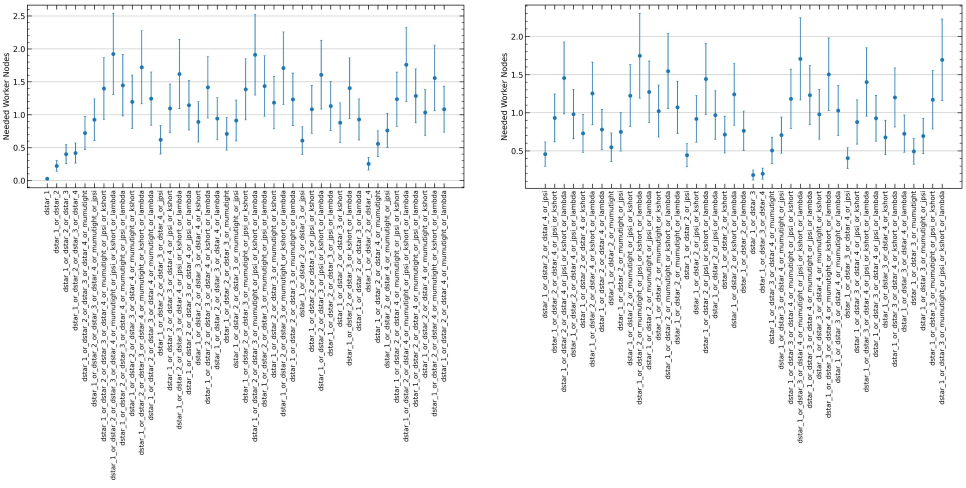

Figure 6. The number of required express-reconstruction worker nodes for each combination of physics flags.

References

[1] K. Akai, K. Furukawa, and H. Koiso, "SuperKEKB collider," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 907, pp. 188-199, Nov. 2018, doi: 10.1016/j.nima.2018.08.017.

[2] T. Abe, I. Adachi, K. Adamczyk, S. Ahn, H. Aihara, K. Akai *et al.*, "Belle II technical design report," 2010, *arXiv:1011.0352*. [Online]. Available: http://arxiv.org/abs/1011.0352.

[3] M. Nakao, "Timing distribution for the Belle II data acquistion system," *J. Instrum.*, vol. 7, no. 1, Jan. 2012, Art. no. C01028, doi: 10.1088/1748-0221/7/01/C01028.

[4] S. Yamada, R. Itoh, T. Konno, Z. Liu, M. Nakao, S. Y. Suzuki, and J. Zhao, "Common readout subsystem for the Belle II experiment and its performance measurement," *IEEE Trans. Nucl. Sci.*, vol. 64, no. 6, pp. 1415-1419, June. 2017, doi: 10.1109/TNS.2017.2693297.

[5] Q. D. Zhou, S. Yamada, P. Robbe, D. Charlet, R. Itoh, M. Nakao *et al.*, "PCI-Express Based High-Speed Readout for the Belle II DAQ Upgrade," *IEEE Trans. Nucl. Sci.*, vol. 68, no. 8, pp. 1818-1825, Aug. 2021, doi: 10.1109/TNS.2021.3086526.

[6] S. Y. Suzuki, S. Yamada, R. Itoh, M. Nakao, T. Konno, T. Higuchi, and K. Nakamura, "The Three-Level Event Building System for the Belle II Experiment," *IEEE Trans. Nucl. Sci.*, vol. 62, no. 3, pp. 1162-1168, June 2015, doi: 10.1109/TNS.2015.2422376.

[7] R. Itoh, T. Higuchi, M. Nakao, S. Y. Suzuki, and S. Lee, "Data Flow and High Level Trigger of Belle II DAQ System," *IEEE Trans. Nucl. Sci.*, vol. 60, no. 5, pp. 3720-3724, Oct. 2013, doi: 10.1109/TNS.2013.2273091.

[8] M. Nakao, T. Higuchi, R. Itoh, and S. Y. Suzuki, "Data acquisition system for Belle II," *J. Instrum.*, vol. 5, no. 12, Dec. 2010, Art. no. C12004, doi: 10.1088/1748-0221/5/12/C12004.

[9] T. Kuhr, C. Pulvermacher, M. Ritter, T. Hauth, and N. Braun, "The Belle II Core Software," *Comput. Softw. Big. Sci.* **3**, 1, 2019, doi: 10.1007/s41781-018-0017-9.

[10] *ZeroMQ*. [Online]. Available: https://zeromq.org/. Accessed on: Nov. 11, 2022.

[11] S.-H. Park *et al.*, "Upgrade of Online Storage and Express-Reconstruction System for the Belle II Experiment," *IEEE Trans. Nucl. Sci.*, vol. 70, no. 6, pp. 949-953, June 2023, doi: 10.1109/TNS.2023.3253517.

[12] R. Brun and F. Rademakers, "Root - an object oriented data analysis framework," *Nucl. Instr. Meth. A*, vol. 389, Issues 1-2, pp. 81-86, 1997, doi: 10.1016/S0168-9002(97)00048-X.

[13] *Zstandard - Real-time data compression algorithm*. [Online]. Available: https://facebook.github.io/zstd/. Accessed on: Nov. 11, 2022.