Data Transportation with ZeroMQ at the Belle II High Level Trigger

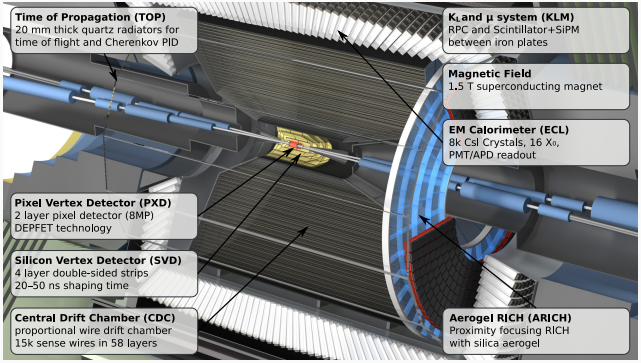
IEEE - Data Acquisition System Architectures

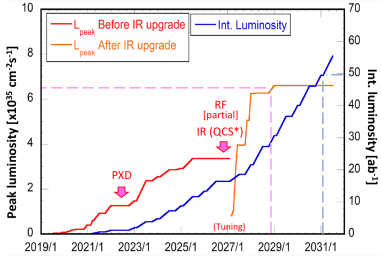
Markus T. Prim for the Belle II DAQ Group

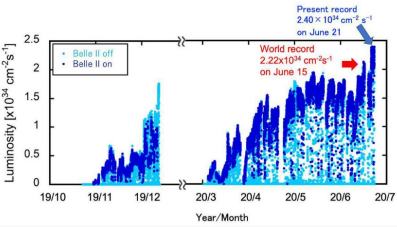
10th October 2020

Universität Bonn - Physikalisches Institut

Belle II Experiment



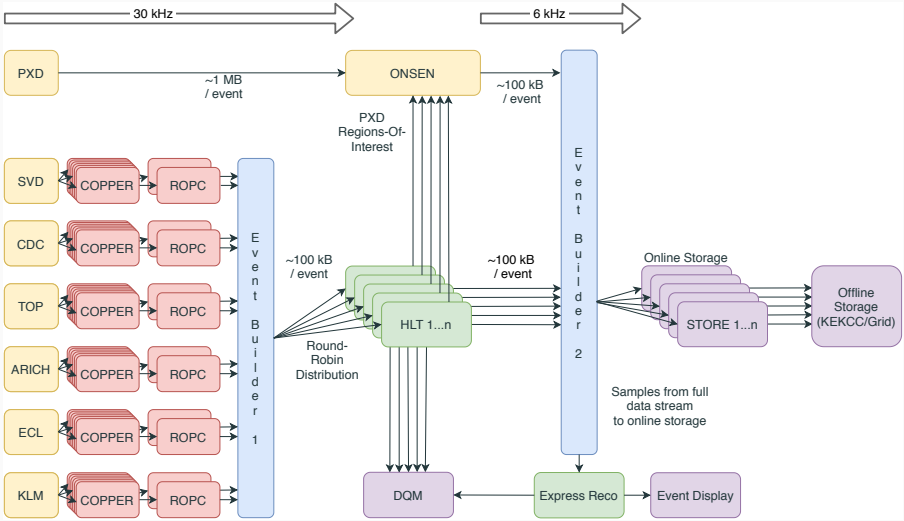




- Accumulated approx. $74\,\mathrm{fb}^{-1}$.
- SuperKEKB world record June 21st, 2020.
- Aiming for $50\,\mathrm{ab}^{-1}$ in the next 10 years.

Belle II Data Acquisition System (DAQ)

How to get the data from the detector to the storage?



Design Parameters

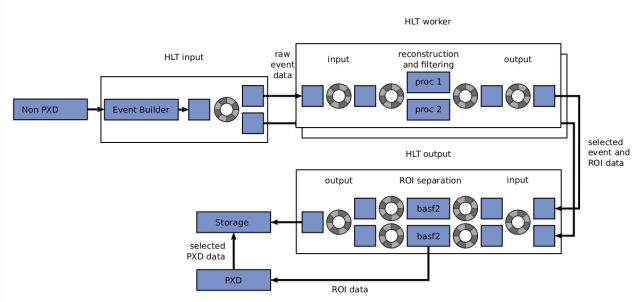
- HLT has to reconstruct the full event and perform the trigger decision.
- Keep/discard detector data but always keep meta data.
- No dedicated "fast" reconstruction software, identical to offline software.
- Average time per event reconstruction is $0.3 - 0.5\,\mathrm{s}$.
- 20 kHz input rate, with peaks up to 30 kHz.
- Round-robin distribution to individual HLT nodes, load-balancing within individual HLT nodes among workers.
- Average event data size of 100 kB without PXD data.
- PXD readout buffer: 5 s (time until region-of-interest has to be provided).
- Data Quality Monitoring has to be available life during operations.

Infrastructure and Environment

General Setup

- $\mathcal{O}(20)$ independent HLT units, each containing 12-20 worker nodes.
- Each HLT unit comes with a single dedicated input, output, and storage and n worker nodes with fast local interconnection.
- Raw data w/o PXD data is streamed by the event builder via TCP connections to the HLT.
- Data in form of ROOT objects is streamed to PXD and Storage system.

Evolution of Data Transport on the HLT



N. Braun, T. Kuhr, 2003.02552

- Shared memory-based ring buffers decoupled from specific processes $\rightarrow$ non-trivial clean-up.
- The rigid structure did not allow for non-data communication, e.g. advanced control features and monitoring of the connections.
- Custom ring-buffers and TCP implementation are difficult to maintain.

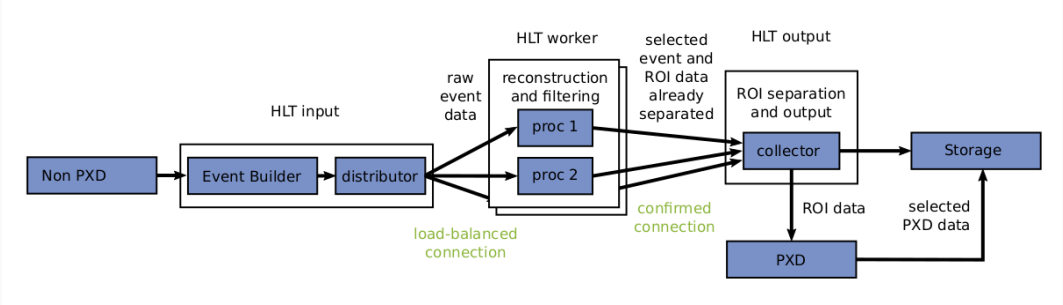
Moving to ZMQ

Why ZMQ?

- To increase the maintainability, core components were built on top of the well maintained, open-source library, ZMQ.
- Industry-standard for high-performance broker-less asynchronous messaging in distributed applications.

Data Transport on the HLT with ZMQ

- Inter- and intra-node communication handled by ZMQ-based TCP connections.
- Buffering via TCP message queue (removing the need for error-prone ring-buffers).
- Allows to pass either event data or control messages, opening up the option for advanced features.
- Initialization and cleanup of connections done automatically.



N. Braun, T. Kuhr, 2003.02552

Connection Types

Raw Connection

- Standard TCP connection.

Load Balanced Connections

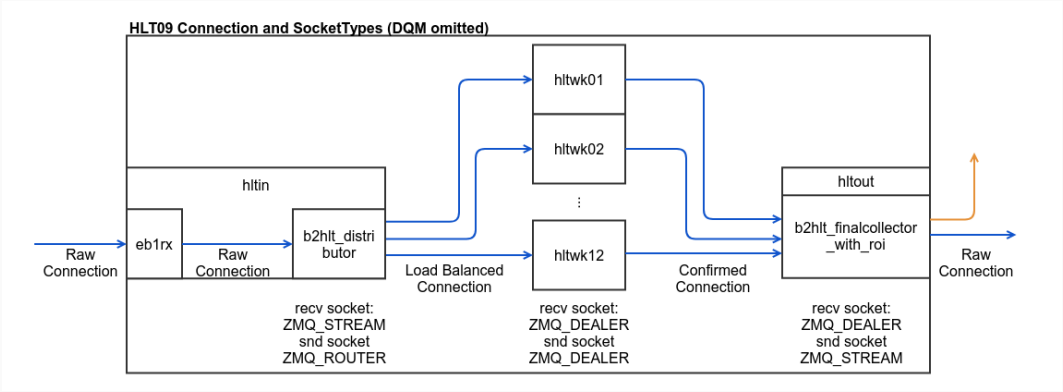
- Sender keeps list of ready receivers, receivers sends "Ready" message after each received message.
- Sender looks for incoming messages when at least one receiver is ready, until then messages are blocked (not dismissed).
- Sender transmits every message to exactly one receiver (except for stop/terminate).

Connection Types

Confirmed Connections

- Sender blocks until it receives "Accept" message (messages from sender to receiver are ensured or block).
- Receiver has list of all possible senders.
- Messages from receiver offline sender are dropped without notice.

Data Transportation Layout



If a worker process dies, $\mathcal{O}(3)$ events are lost, which corresponds to the message buffer.

Control Features

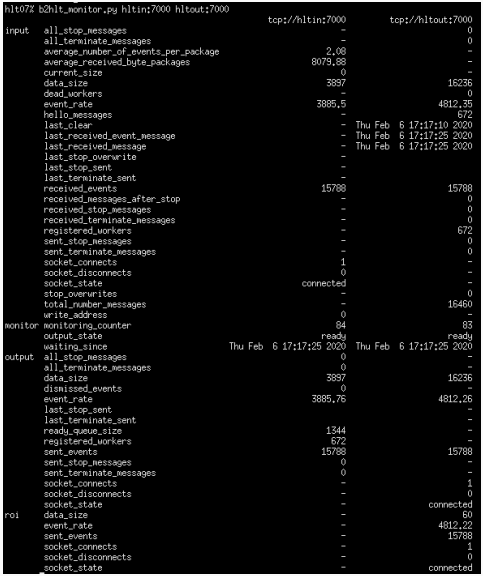
Run Start - SALS

- Restarting the run, e.g. due to sub-system failure, follows the Stop-Abort-Load-Start sequence.
- Loading HLT takes a significant amount of time $\mathcal{O}(60s)$ due to initialization of the geometry in the reconstruction software.
- New data flow system allows for simplified Stop-Start sequence, because control messages allow to e.g. update the run number.

Monitoring Features

Socket Monitoring

- ZMQ applications answer with a JSON encoded dump of their internal state and counters via a specific TCP message.
- Currently only used for manual debugging.
- ZMQ available as a python library, simple monitoring and data collection pipelines can be built and executed.



Experience

- Changing to a new data flow implementation is not free from teething problems, but ...
- switching from raw TCP sockets and custom ring buffers to ZMQ has allowed for rapid development and bug fixes.
- the entry level for new developers has decreases significantly.
- the advanced monitoring features have already proven helpful for debugging.

Summary

Results

- **New data transportation scheme handles rates up to 10 kHz per HLT unit.**
- Comparison: Previous implementation was able to cope with 1.5 kHz.
- ZMQ based setup allows to omit Abort/Load in restart sequence which reduces downtime (Loading geometry is time intensive).
- Current setup of the data transport is future proof for the anticipated instantaneous luminosity of SuperKEKB.

Outlook

- Not all data flow is migrated to ZMQ yet, but conceptually no show-stopper.
- Socket monitoring can be included into the elastic stack (see Real-time monitoring of operational data in the Belle II experiment presentation).