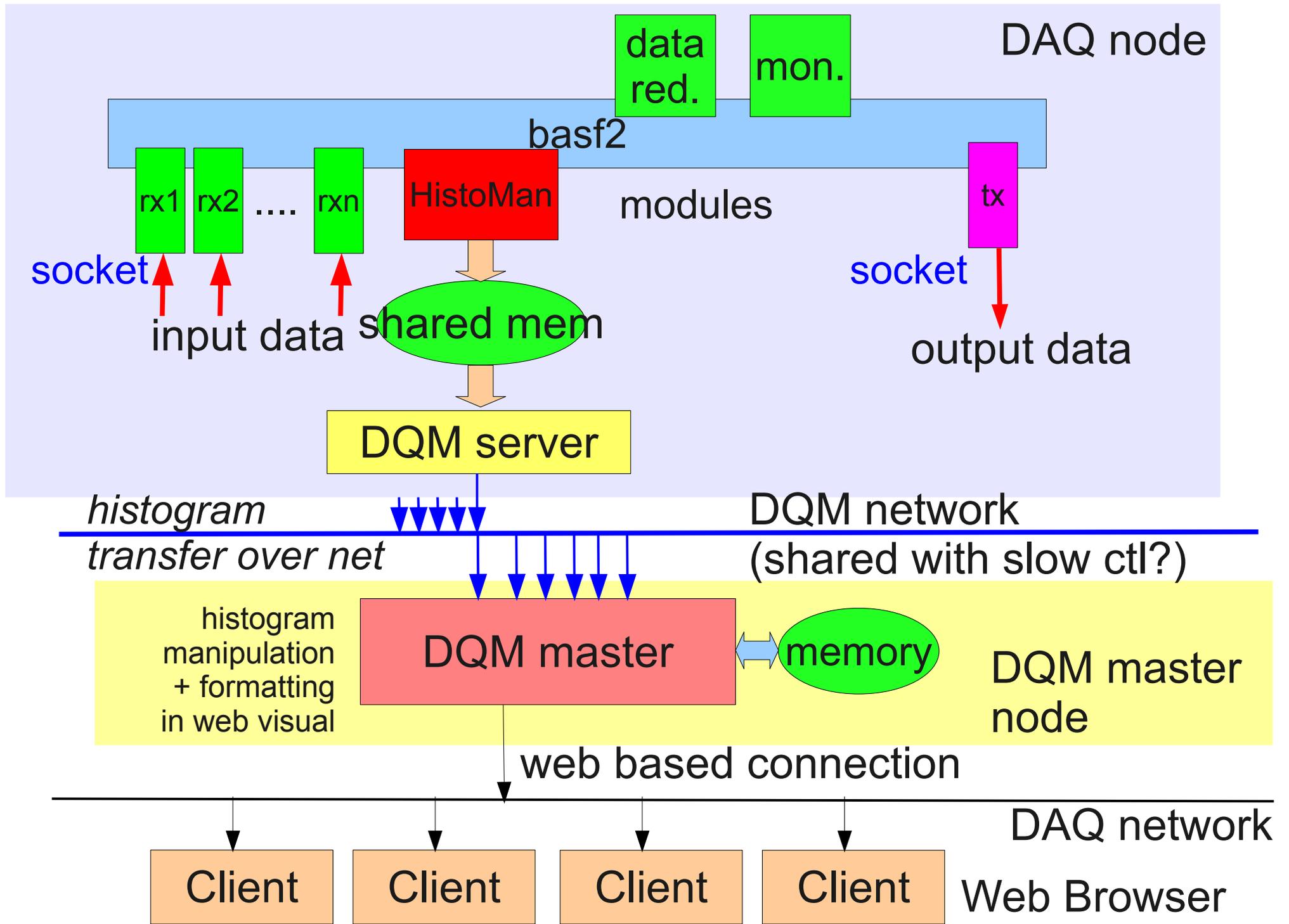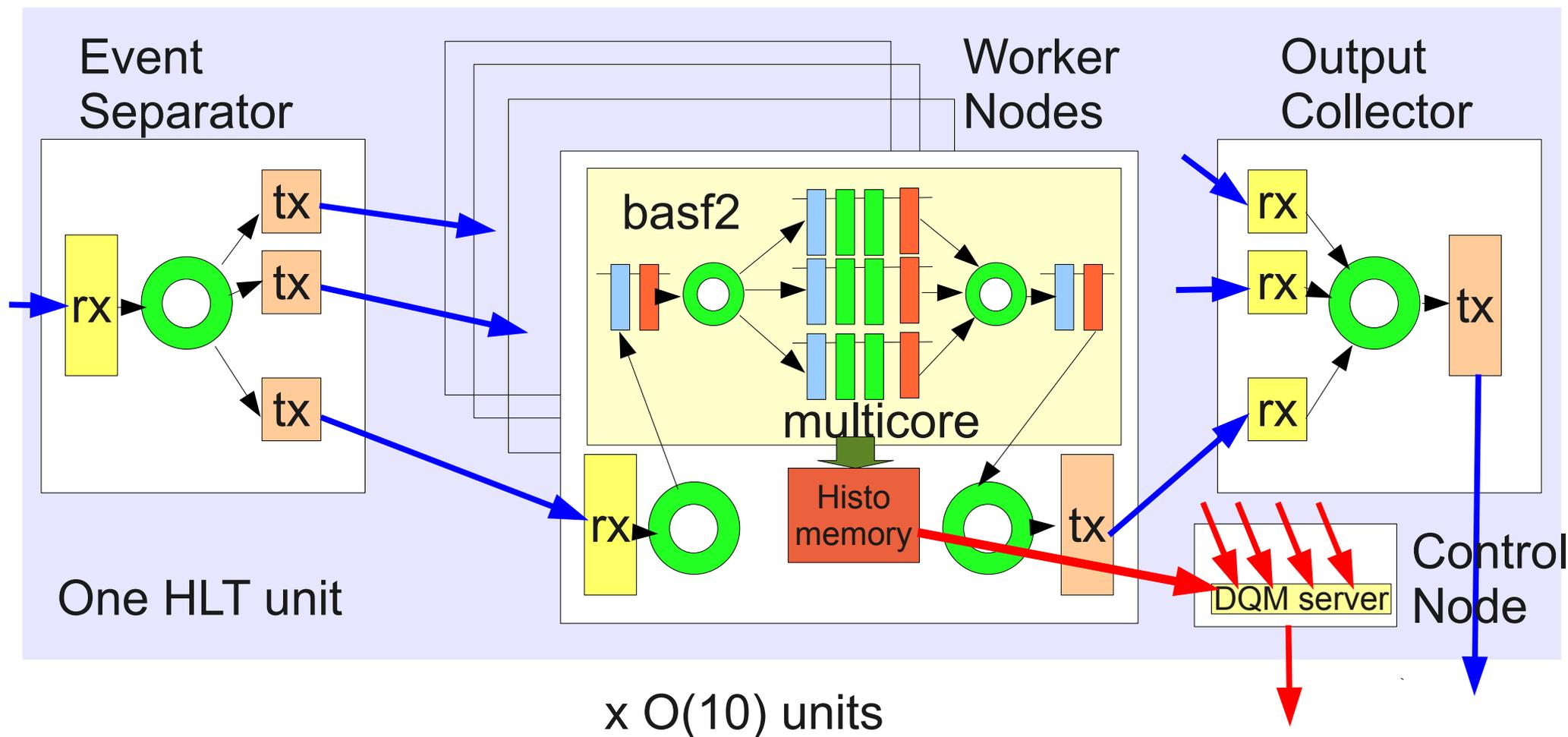# DQM and Prompt/Express Reco

R.Itoh, KEK

1. Design of DQM in Belle II DAQ data flow

- HLT is the main DQM histogram producer.

- Histograms accumulated on each node are first collected by control node and added.
- The added histograms are sent to DQM master through DQM server.

Belle II DQM consists of

1. Histogram manager module for basf2 (with DQM mods)
   - dumps the live histogram contents to
     the shared memory (or a memory file) periodically
     during the real-time data processing.

2. Histogram server
   - runs on each node
   - sends the histogram contents in the shared memory
     to network on request

3. DQM master
   - runs on DQM monitoring node
   - collects histograms from DQM nodes and format them
     into graphical objects on web.

4. DQM client
   - runs on console in control room + other user nodes.
   - browses histograms using web browser

DQM code

* DQM codes are supposed to be provided by each detector group / physics group

- DQM codes are basf2 modules with histogram accumulation using ROOT under HistoManager.

- Each detector group is supposed to provide 1-D histograms at least
    * Hit map
    * Gain map
  etc.

- High-level monitoring (tracking efficiency, R2, etc.) using HLT processing results is also possible.

- The codes are executed on HLT under full Belle2 software environment.

- "Reference" histograms for them have to be provided together.

In addition,

- Simple DQM codes can be run on COPPERs and Readout PCs.

- The purpose is mainly to issue warning or error to slow control
  (NSM) if bad data are detected.
  (ex. detection of AMT's L1 buffer overflow at Belle)

- Histograms can be accumulated, but they are turned on only
  when required for debugging purpose to reduce the CPU
  consumption. Their real-time monitoring is optional and limited
  to DAQ experts.

However, it is too early to prepare DQM codes now.
  - I will ask relevant people to provide the code when data flow
    scheme is established. (within ~2 years)
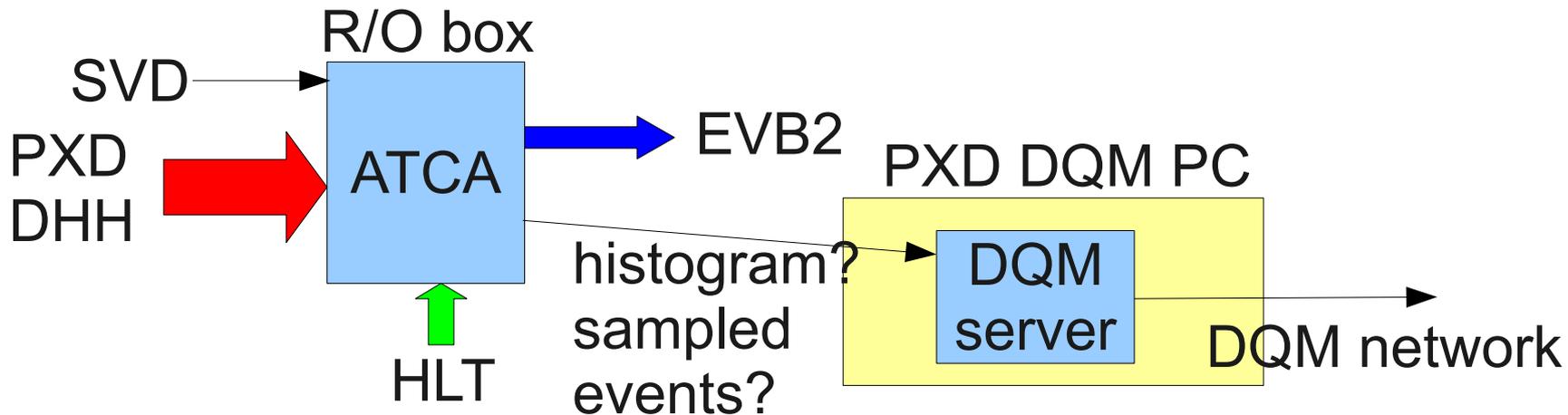
How to monitor histograms to find problems

- Data quality is primarily supposed to be monitored by exp shifters
  by comparing live histograms with references.
  * As done in Belle.
  * The number of histograms to be monitored is limited
    considering shifter's capability.

- Any automated method to compare live histograms with
  references and to find the problem?
  * LHC experiments should have some experience, but I
    don't know the detail (there were some presentations in
    CHEP2010, I remember).
  * I need help especially on this!
        -> "DQM Task Force Issue!!!!"

DQM for PXD

- Since the data from PXD are merged AFTER HLT
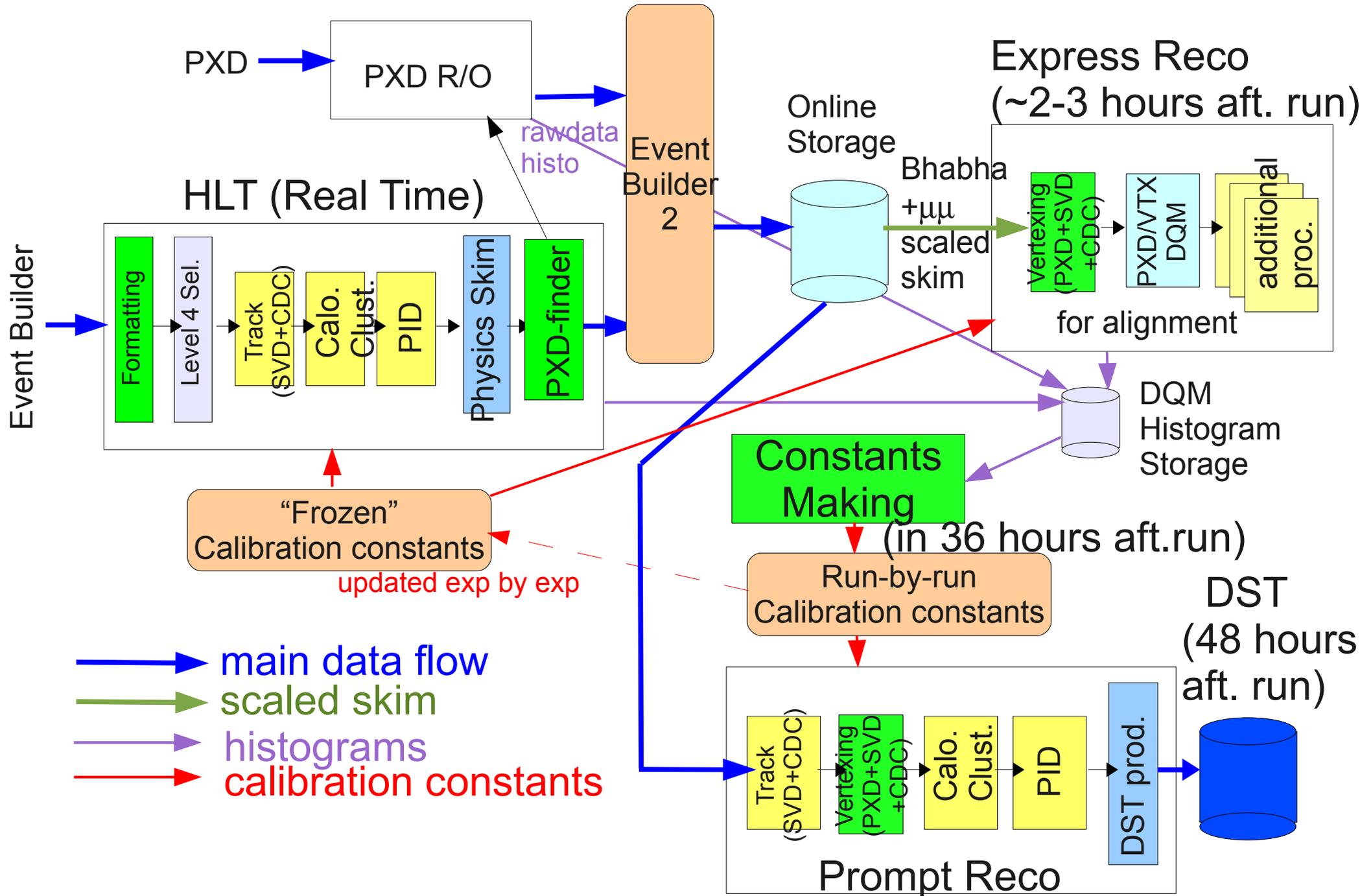 processing, there is no place to monitor the data.......

How to monitor PXD data quality????

- Simple "hit histograms" could be accumulated in
  PXD Readout Box (ATCA or PC). But PXD people are
  thinking about complicated processing in ATCA and the
  scheme to monitor the processing is quite unclear.....

R/O box

SVD ⟶

PXD
DHH ⟹ ATCA ⟹ EVB2

histogram?
sampled
events?

HLT

PXD DQM PC

DQM
server

DQM network

- For the detailed monitoring of PXD data, a full tracking
  with SVD and CDC data combined is required.
  -> Prompt Reco

Prompt Reco Scheme (Original)
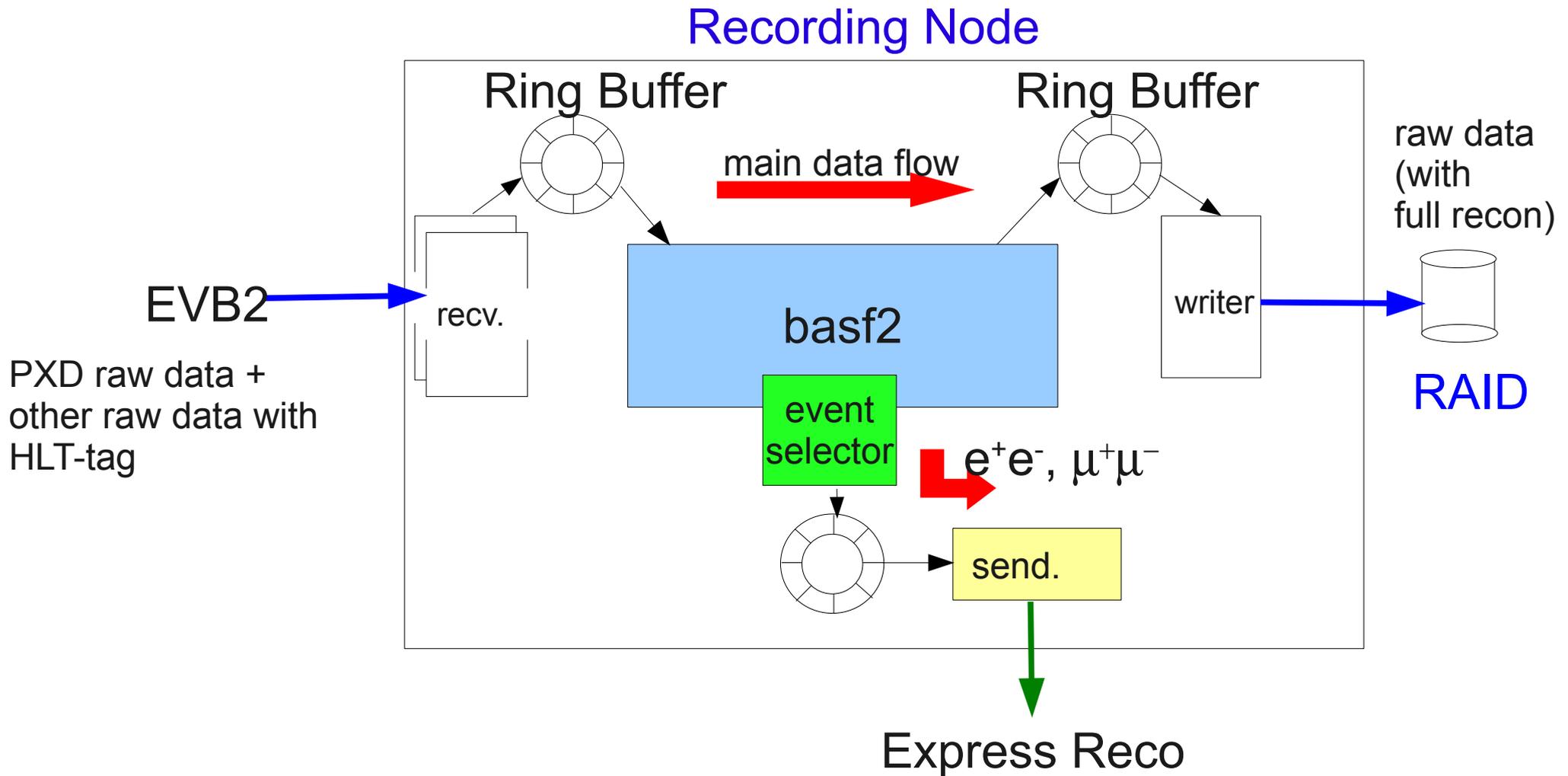
Express Reco: Previous Idea

- The raw data including PXD output + HLT results are written to the online storage.

- In the previous "Prompt Reco" design, we have "Express Reco" right after the online storage.

- Express Reco reads the recorded data in the storage and a full tracking using PXD + CDC + SVD is supposed to be performed utllizing the HLT processing results to generate PXD-DQM.
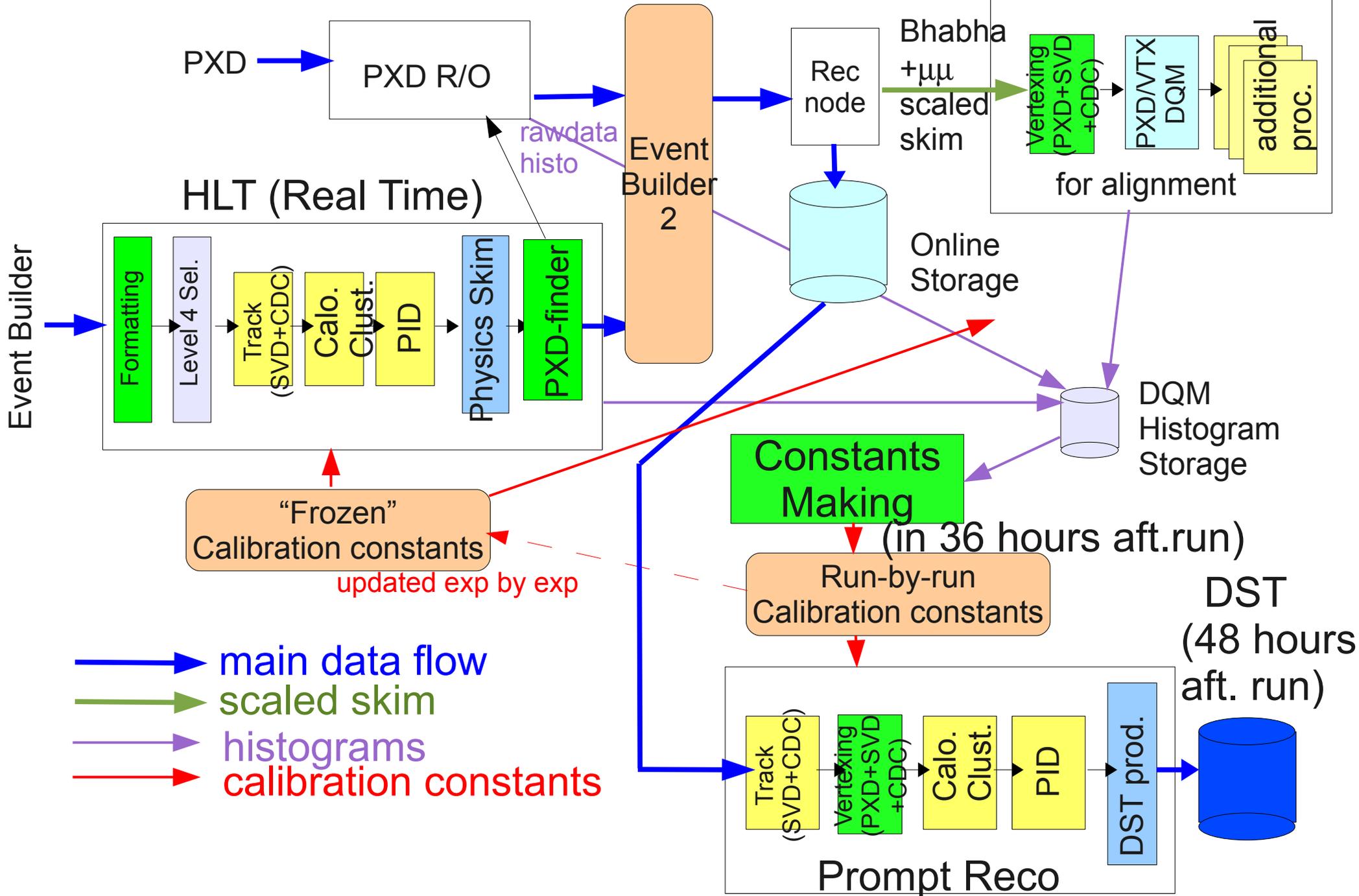
Delay in the data processing
- Original idea: to process the data after a run is taken
  -> a few hours after the run end, too late.......
- Improved idea: "Blocked file" scheme
  -> a few minutes delay, too complicated......

- It is turned out that Express Reco can be fully real-time by
  connecting to recording node through network directly
  (suggested by Christian) instead of reading from disk.

ExpressReco : Realistic Sketch

Express Reco [fully real time processing!]
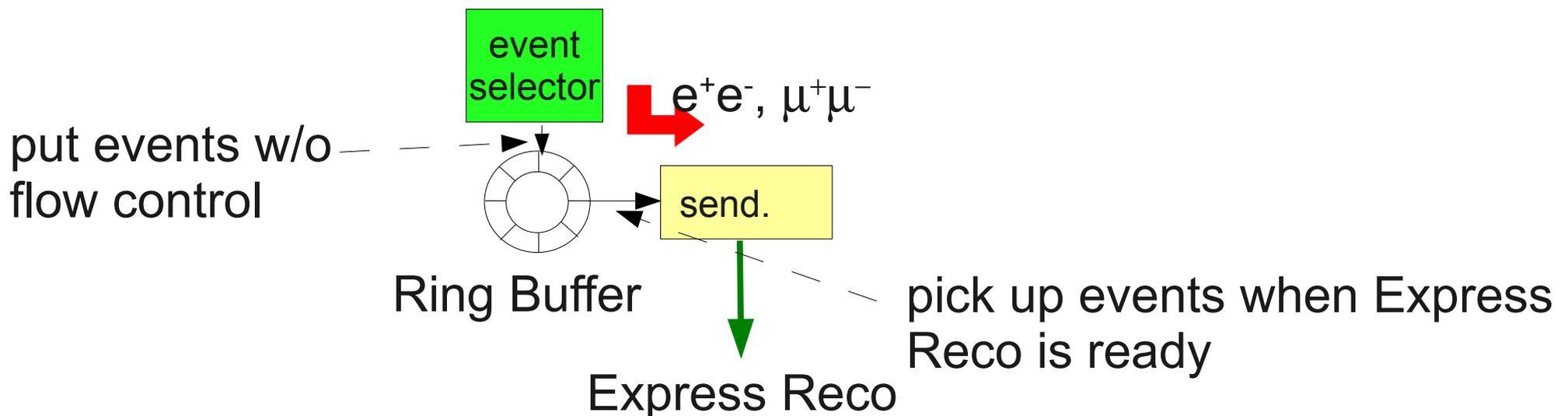(2-3 units in total)

Inputs to Express Reco

- The inputs to express reco are scaled HLT-accepted events such as:

  - Bhabha + $\mu^+\mu^-$
  - Random trigger
  - Scaled HadronBJ
  - Anything else?

- The inputs are scaled "as much as possible to process" basis.
  <- "Free runnning" ring buffer

event selector

e^+e^-, $\mu^+\mu^-$

put events w/o flow control

send.

Ring Buffer

Express Reco

pick up events when Express Reco is ready

Architecture of Express Reco

- The same HLT framework is supposed to be recycled for Express Reco.

- Supposed to have 2 "HLT-equivalent units" for Express Reco at t=0.

- We will run full trackiing/vertexing with PXD+SVD+CDC combined on Express Reco.

- Full DQM code for PXD is supposed to run.

- Detailed "physics monitor" can also be implemented using the tracking result. (ex. Hadron A selection performance with full tracking)

- What else?

Summary

- The real time histogram collection mechanism in Belle II data flow is being designed and developed.

- The scheme to check the collected histograms is not yet fixed.
    * "Manual" histogram check by exp. shifts may be not enough.
    * Automated checking scheme is required.
        -> DQM task force issue. Supposed to be formed right after DQM collection scheme is fixed (in a half year).

- Express Reco is introduced for the DQM of PXD. It is now fully real-time processing.

- Sophisticated DQM could be implemented on Express Reco utilizing full tracking results with PXD+SVD+CDC.