(PXD) DQM Status

Björn Spruck, Uni Mainz

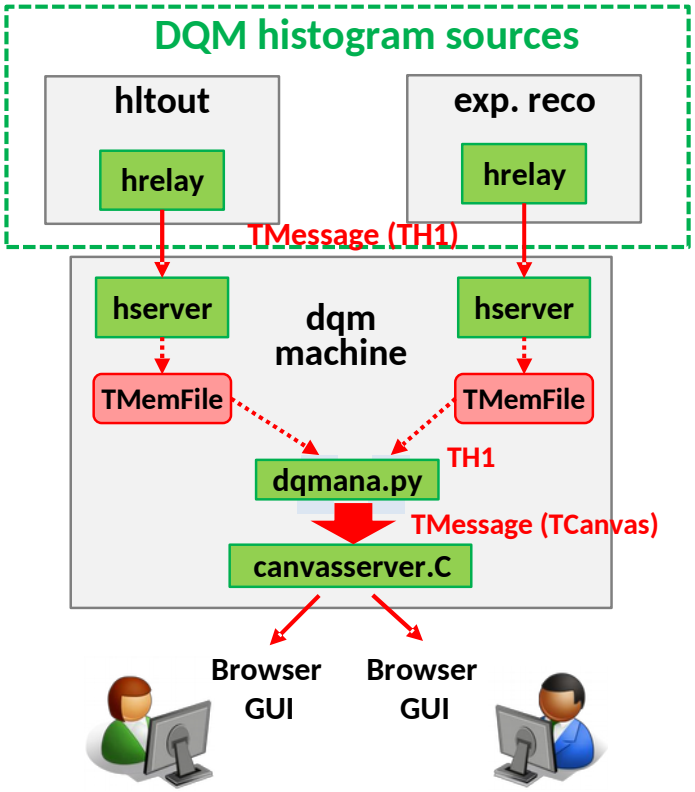
Belle2 PXD Workshop, DESY
23.1.2018

DQM viewer scheme

Tomoyuki Konno
Software workshop
2017/10/06, KEK, Tsukuba

JSROOT based DQM scheme is implemented

- jsroot: https://root.cern.ch/js/
- Histograms in TMemFile are packed into TCanvas
- Shown in web browser of CSS (internal browser) by THTTP



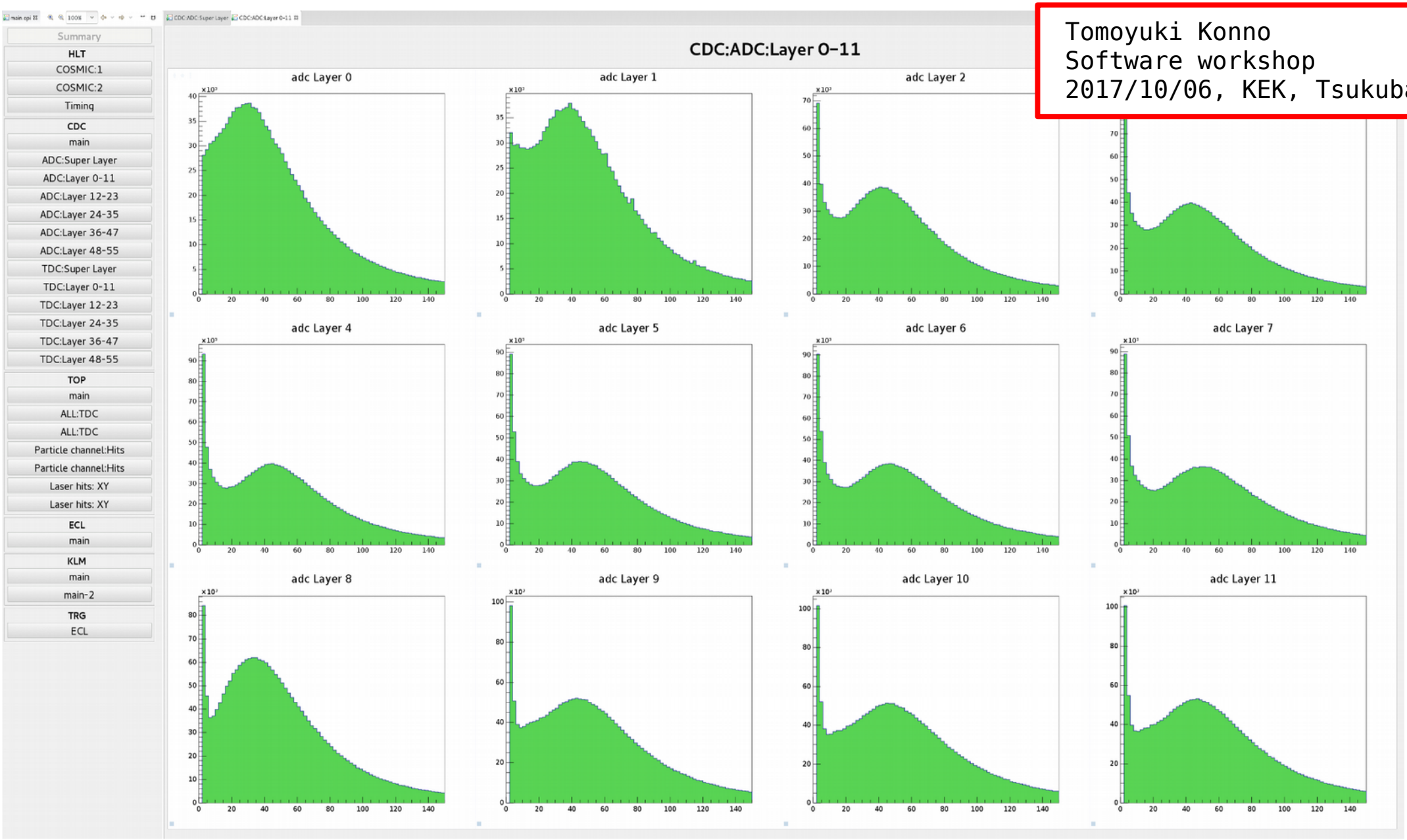
3 components working in the new scheme

- basf2 in HLT/expreco analyze online data togenerate histograms same as before
- dqmana.py (basf2 script) analyze histograms to pack into TCanvas
  - Adding new modules for dedicated analysis
  - Reference histogram can be added here
- canvasserver.C (root script) accepts TMessages with canvases and distributes them to browsers (THttpServer)
  - CSS and browser are available as viewer

→ My suggestion with Canvas-Server was happily accepted by DAQ group.

Belle2 DQM – Shifter Screen – Templates

Tomoyuki Konno
Software workshop
2017/10/06, KEK, Tsukuba



Histogram/Canvases accessed by their name → "just" change the names in template

Discussion Outcome from B2GM (Feb 2017) and Ringberg

- Because of ERECO (ExpReco) limitations we need a separate PXD DQM machine (for expert histograms!)
  - In addition to mainline expreco DQM
  - Allows for “unlimited” number of histograms and fine binning (even up to pixel level)
- Can use EPICS slow control interface (→ alarm system) directly
- Sample events from ExpReco, but do VXD only analysis with more and refined histograms as output (within basf2!)
- Have the possibility to cross-check (fit, compare) histograms from other sources (e.g. BonnDAQ, SVDDQM)
  - Import from EPICS and other sources
- Using same code (DQMHistAnalysis…) for synergies with other detectors

PXD DQM Machine

- PXD DQM machine installed (Sep/Oct 2017) and used since then.
  - Current location: next to control room → move to ERECO room
  - SL7, basf2 development and private branches, **not** release v01.00.00
  - hserver, jsroot server
- ERECO Event Server **not** tested yet.
  - ERECO only available since middle January, but without Event Server
- → PXD DQM, but without input data/events. Not too useful atm.

- dqm folder of basf2 moved out of daq folder.
  - Liberian Boqun Wang
  - Problem: still depends on daq and nsm2
  - → does not compile out of the box
  - requirement: nsm2 in external packets, daq folder compilable
    - work in progress
- Feature commits only by T. Konno and BS
- New features in master
  - automatic reference histogram comparison
    - coloring depending on result (“red frame”)
  - exporting fit results as PV, need local EPICS installation
    - request: EPICS 7 in external packages
  - importing Histogram PVs from other machines e.g. Bonndaq

PXD DQM Modules

- Two steps:
  - Filling histograms, analysing histograms → In separate tasks!
  - Analysis code only needed if special things have to be done. Common plotting/fitting/comparison tasks can be used otherwise.
- Current “main” DQM module incompatible with ERECO parallel task scheme
  - Stripped down version used at the moment.
  - Histogram analysis must be done on final histogram → separate histogram analysis task
- Low level DQM modules (working on “raw” pixels) work fine.
  - Histograms depending on Switcher/DCD combination created, fit result prepared in overview histogram
- Additional DQM modules (DAQ status, errors, event wise data reduction etc) in the pipeline

Other TODO Topics

- Histogram is updated every xxx events, but only reset on begin of run
  - We want different number of events per histogram, better not integrate a whole run with O(hour) length
- Itoh-san promised to provide infrastructure for “delta” histograms.
  - … no time until now
- How to handle “stale” histograms? Remove after xx minutes from histogram server? Add some “stale” text across?
- Current root/jsroot is missing some feature we want to use. (→ update externals, use private jsroot)
- Current PXDDQM borrowed from Itoh-san. Clarify when (if) we have to give it back.
- Some network issues not finally decided…

History, Histogram Archiver and Reference Histograms

- How and where to store reference histograms?
  - Common problem?
  - DB (Peter Kodys using this right now in PXD DQM)
  - (static) root files (as for software quality monitoring)
- Need easy way to access reference histograms (from last hour/day/week).
  - In CSS? Within the same Canvas?
    - JSROOT would allow for that
  - How to update them?

Backups

Analysis Example

- High level DQM provided by Prague group
  - 17*layers = 680 hists
  - Incl. 2d histos with pixel binning
  - Binning configurable
  - → too much for ERECO
- Histograms depending on DCD/Switcher combination
  - 40*6*4=960 histograms per type
  - Fit + fill 40 new 2d histogram with fit results
  - → Check with new histogram/canvas server → O.k.
  - (sorry, no screen shot)
- Guess: Not on ExpReco (at least not for 40 sensors)
- Simplify: using mean instead of fit → reduce to 80 histograms

Display in CSS

- Until TB 2017, we used DAQ home-brewed solution to display histograms in CSS, using some histogram proxy server
  - Supports only „raw“ histograms, no graphs, fit functions etc
- JSROOT (as part of root) has much better support for any kind of root objects. Using root's internal http server.
  - Display complete histogram+fit+values/labels/guidance lines etc
- Plugs seamless into existing histogram and fitting server scheme
  - One additional server task needed.
  - Full advantage only if fitting and drawing is done in DQM modules
- DAQ accepted → is/will be used now