DQM for 2021 Data Taking

Matt Graham & Jeremy McCormick

June 2021 Collab Meeting

June 25, 2021

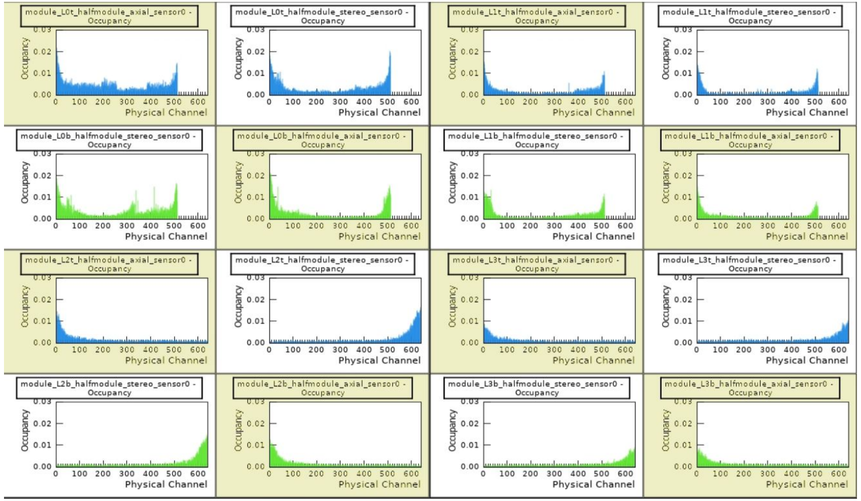
Overview

- DQM here really refers to two things:
  - online monitoring to pick up ~gross issues while running
  - offline monitoring (soon after run is on tape) to pick up finer issues and help monitor things over time run-by-run
- In 2019 run, these used separate drivers and tools (though both used hps-java to recon and that will continue)
- This talk, I'll go over the existing tools and a new scheme concocted by Jeremy (mostly) and me (a little)
  - Focus mostly on online monitoring

monitoring-app: existing online monitoring

- The online monitoring-app used for 2019 still works for latest hps-java (with a few bug fixes, not yet committed)
- The app connected to ET ring, ran set of drivers that made AIDA plots and displayed them...plots are interactive.
- We had separate scripts for SVT, ECal, Recon monitoring
- Shifter looked at these plots for every run and posted PDFs to eLog (there is a button to make pdf)

Occupancy: L0-L3



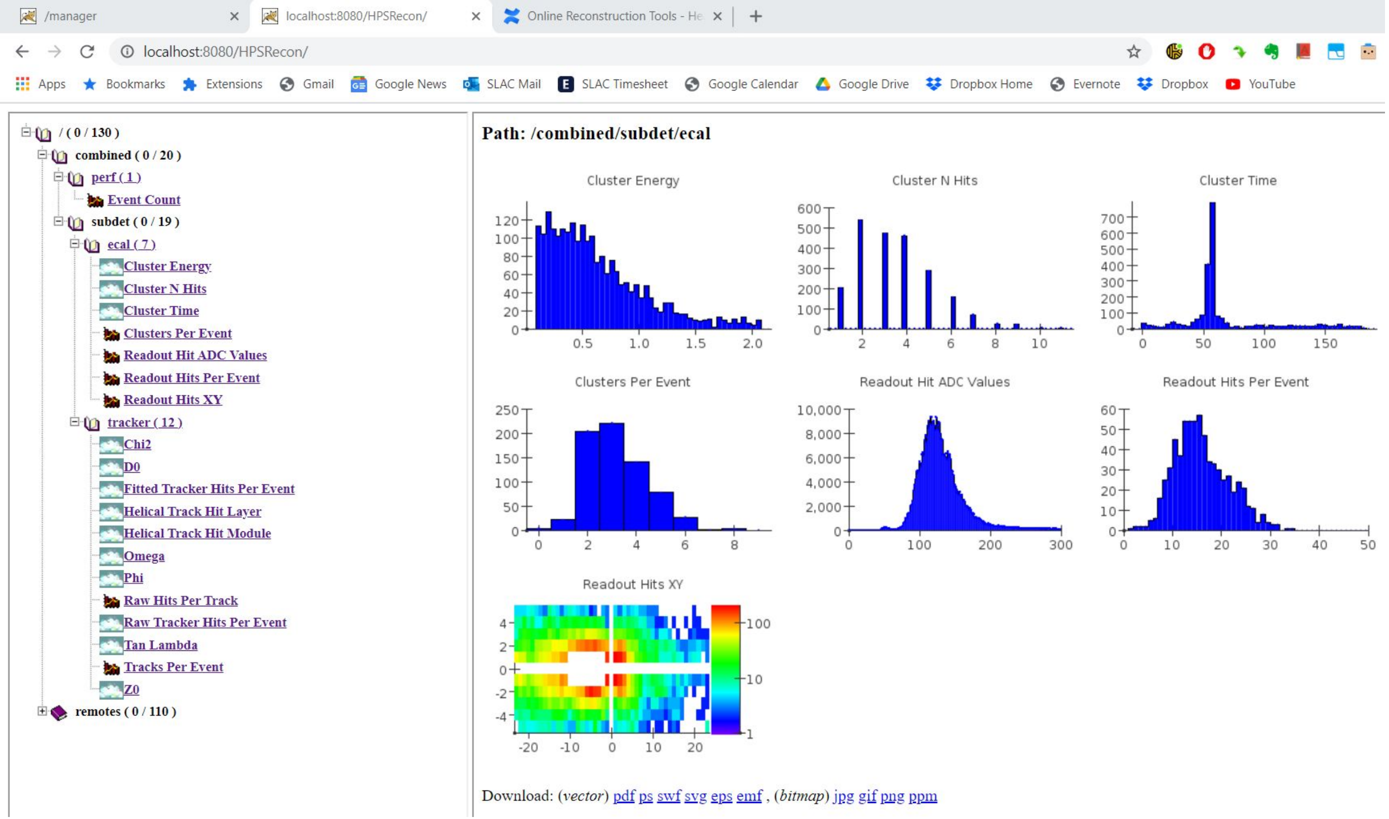
Issues with monitoring-app

- It’s fairly memory intensive
  - This is likely due to the actual plots being filled...IIRC SVT monitoring was the worst since some of the plots had 1 bin/channel.
- It’s slow...particularly tracking recon
  - Well, this is mainly due to seed tracker + large timing window we had to use…
  - Kalman is faster though with unaligned detector + timing maybe still an issue, likely to be fairly slow
- Too many plots, shifters didn’t know what they meant
  - This is on me...I like lots of plots. Working on paring this down to minimum
  - Documentation needs to be much better for anything we do (on me also)

Jeremy’s new online monitoring scheme

- Confluence page for online recon
- This spawns a number of reconstruction jobs connected to the ET ring, all making the same plots
    - Parallelizes recon → more throughput
- There is an aggregator the periodically gets the plots and sums them
- There is a thread that serves these plots to the web (or an AIDA client)
- If we can get correct permissions from JLAB, we could look at these plots in real time via web browser (see next slide)...would be great for remote workers
- Can also (now? soon?) save snapshot of aida/root tree which (again with permissions) be viewable/linkable on the web via JSROOT

Snapshot of web client



- / (0 / 130)
  - combined (0 / 20)
    - perf (1)
      - Event Count
    - subdet (0 / 19)
      - ecal (7)
        - Cluster Energy
        - Cluster N Hits
        - Cluster Time
        - Clusters Per Event
        - Readout Hit ADC Values
        - Readout Hits Per Event
        - Readout Hits XY
      - tracker (12)
        - Chi2
        - D0
        - Fitted Tracker Hits Per Event
        - Helical Track Hit Layer
        - Helical Track Hit Module
        - Omega
        - Phi
        - Raw Hits Per Track
        - Raw Tracker Hits Per Event
        - Tan Lambda
        - Tracks Per Event
        - Z0
  - remotes (0 / 110)

Path: /combined/subdet/ecal

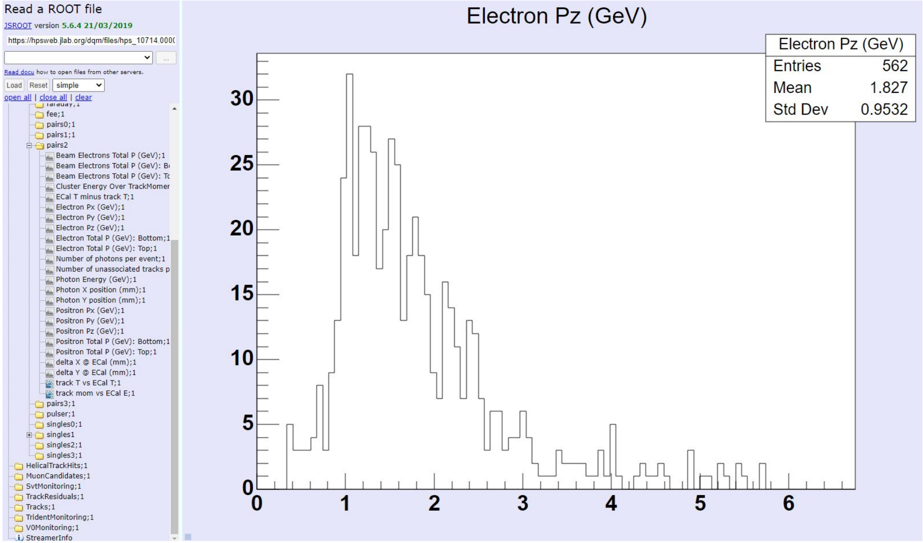
Download: (*vector*) pdf ps swf svg eps emf , (*bitmap*) jpg gif png ppm

Looking at online snapshot root file

Direct link to DQM File

We already have ability to look at aida->root files at JLAB via JSROOT.

The idea is that instead of dumping pdf files to the ELog for every run, we put a link to the file via JSROOT and people can browse



Read a ROOT file

JSROOT version 5.6.4 21/03/2019

https://hpsweb.jlab.org/dqm/files/hps_10714.000(

Read docu how to open files from other servers.

Load | Reset | simple

open all | close all | clear

```
        faraday;1
        fee;1
        pairs0;1
        pairs1;1
      ⊟ pairs2
          Beam Electrons Total P (GeV);1
          Beam Electrons Total P (GeV): B
          Beam Electrons Total P (GeV): To
          Cluster Energy Over TrackMomer
          ECal T minus track T;1
          Electron Px (GeV);1
          Electron Py (GeV);1
          Electron Pz (GeV);1
          Electron Total P (GeV): Bottom;1
          Electron Total P (GeV): Top;1
          Number of photons per event;1
          Number of unassociated tracks p
          Photon Energy (GeV);1
          Photon X position (mm);1
          Photon Y position (mm);1
          Positron Px (GeV);1
          Positron Py (GeV);1
          Positron Pz (GeV);1
          Positron Total P (GeV): Bottom;1
          Positron Total P (GeV): Top;1
          delta X @ ECal (mm);1
          delta Y @ ECal (mm);1
          track T vs ECal T;1
          track mom vs ECal E;1
        pairs3;1
        pulser;1
        singles0;1
      ⊞ singles1
        singles2;1
        singles3;1
  HelicalTrackHits;1
  MuonCandidates;1
  SvtMonitoring;1
  TrackResiduals;1
  Tracks;1
  TridentMonitoring;1
  V0Monitoring;1
  StreamerInfo
```

Electron Pz (GeV)

Electron Pz (GeV)	
Entries	562
Mean	1.827
Std Dev	0.9532

Plan of attack and work to be done

- The old monitoring-app is working locally and I will test it at JLAB soon
  - Changed/added drivers/steering files to use Kalman tracking
  - Will also make a minimal set of drivers/plots that are appropriate for shifters
  - Shifters will have to look/interact with these via VPN
  - This is the backup...safe and “pretty much” ready
- We’d like to shift to Jeremy’s plan
  - I think this would really help remote workers. Still need to log in to clone machines for some controls but looking at plots is all web.
  - I think Nathan has already tried out the guts of this on a clon machine and it worked ok (true Nathan?)
  - Jeremy is working on getting some more features that I requested
  - We need to start getting permissions set up, which means we need to know which machines/ports we would use
  - I also need to make my drivers to be compatible with this app...it’s not hard but it is work.
  - Requires much testing @ JLAB….needs to start soon