PLR TDAQ Needs

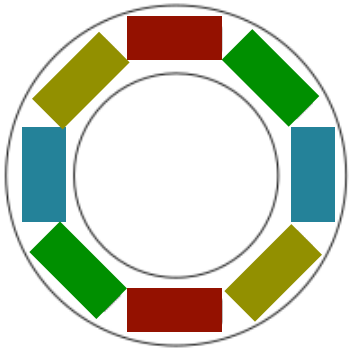
PLR Kickoff
31 October, 2024

Largely taken from IDR presentation
and discussions afterwards,
but heavily condensed…

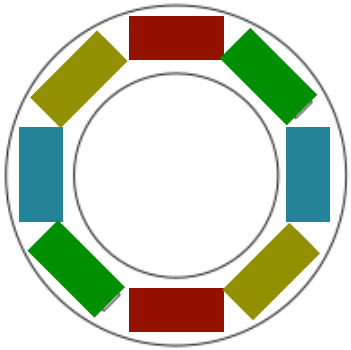
Eric Torrence
University of Oregon

PLR Readout

- One design goal is to have 4 ‘separate’ luminometers
  - Provides some systematic cross-checks
  - Redundancy in case of hardware problems
- Symmetry in ±η and ϕ important
  - Reduce first-order sensitivity to beamspot position, crossing angle, beta*, etc.
  - Clearly demonstrated with LUCID
- Save cluster counts per module (and possibly per sensor, or 2 regions in radius) to give more information
  - At least 16 x 3564 counters per lumi block, possibly more



A-side



C-side

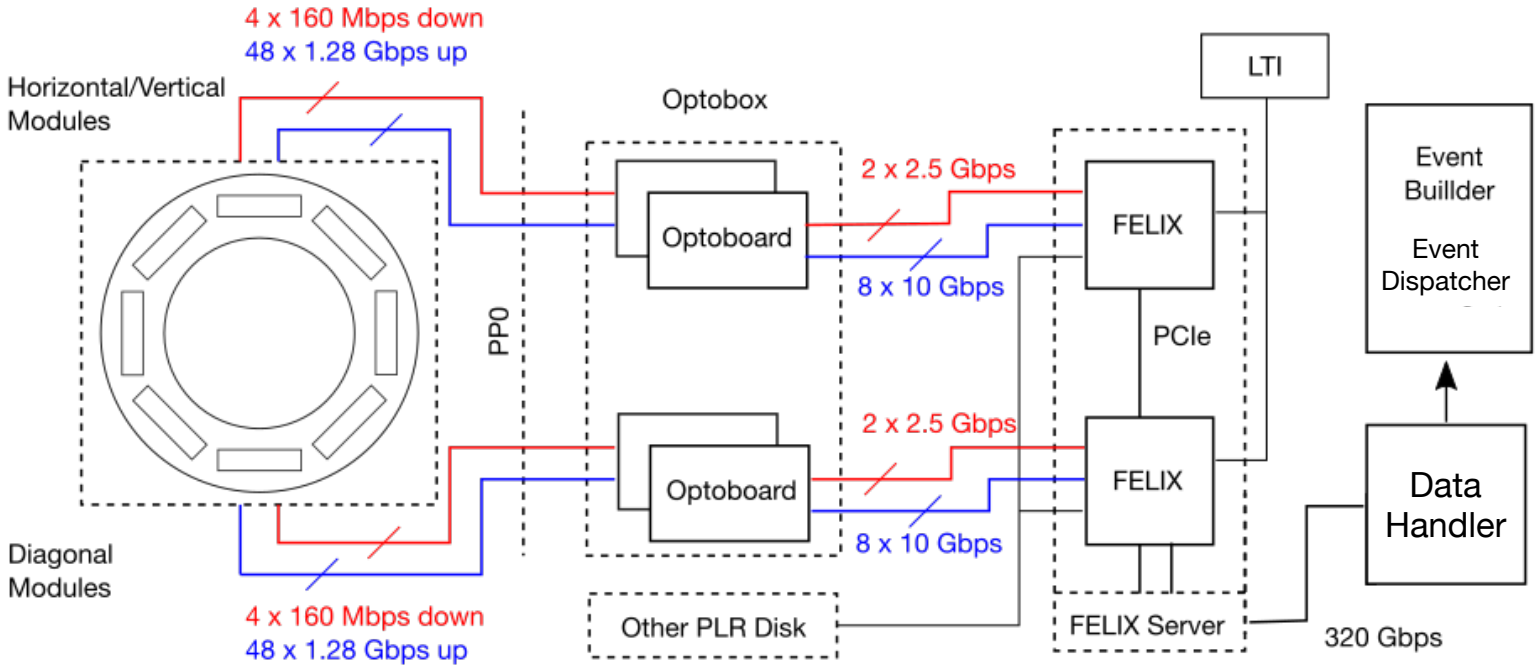
Horizontal
Vertical
Diagonal U
Diagonal V

PLR TDAQ Overview

- Independent readout using PLR LTI in “mini-CTP” mode
  - Randomly sampling filled bunches w/ 1 MHz readout
  - dL/L ~ 1% per sec per BCID @ μ=200
  - Sample fewer (~100) BCIDs at 11kHz at μ ~ 1 (vdM scans)
- Readout through FELIX is IBL standard
- Dedicated PLR workers in event filter
  - Unpack, perform pixel clustering, form sums
  - Highly prescaled output stream for monitoring/debugging
- Opportunistic Readout
  - For PLR triggers that coincide with ATLAS L0A, want to include PLR hits in ATLAS readout
  - Not to be included into track reco, but PLR hits can be matched to reco tracks
  - Important tool for monitoring PLR efficiency

Will concentrate on issues to keep this short

PLR Readout

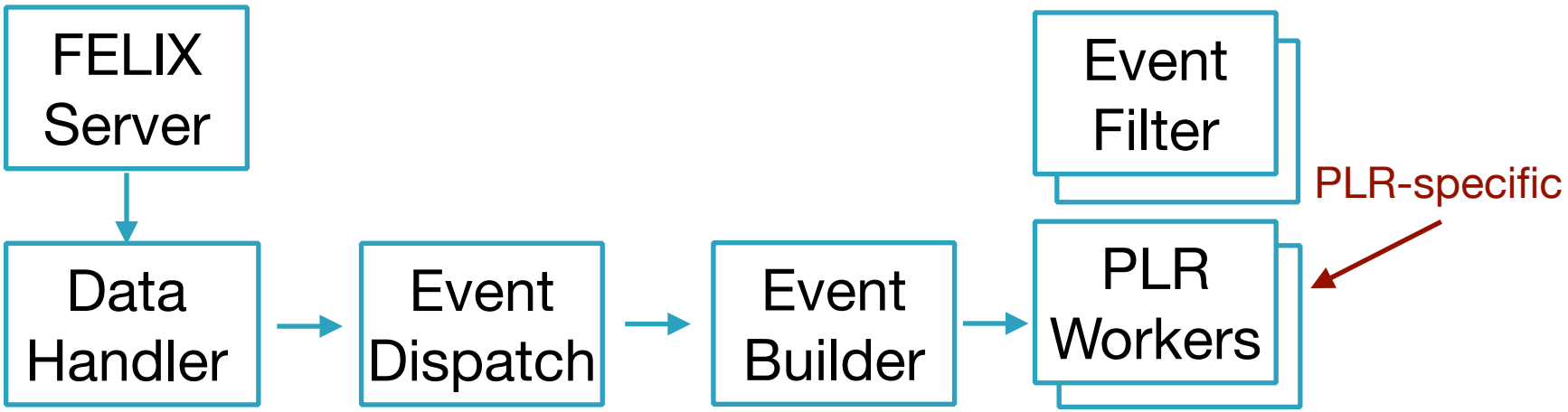


- Each FE can produce up to 5 Gbps (4 links)
  - 96 copper uplinks + 8 downlinks to 4 optoboards per ring
  - 16 fiber uplinks to FELIX per ring, 32 fibers in total
- Planning for 2 FELIX boards, one server
  - ITk pixel spec is 24 optical links at ~50% utilization per link

If decision is 3 links, limited to 1 MHz at high mu.

May still be able to run faster at low mu, desire to explore FELIX limits

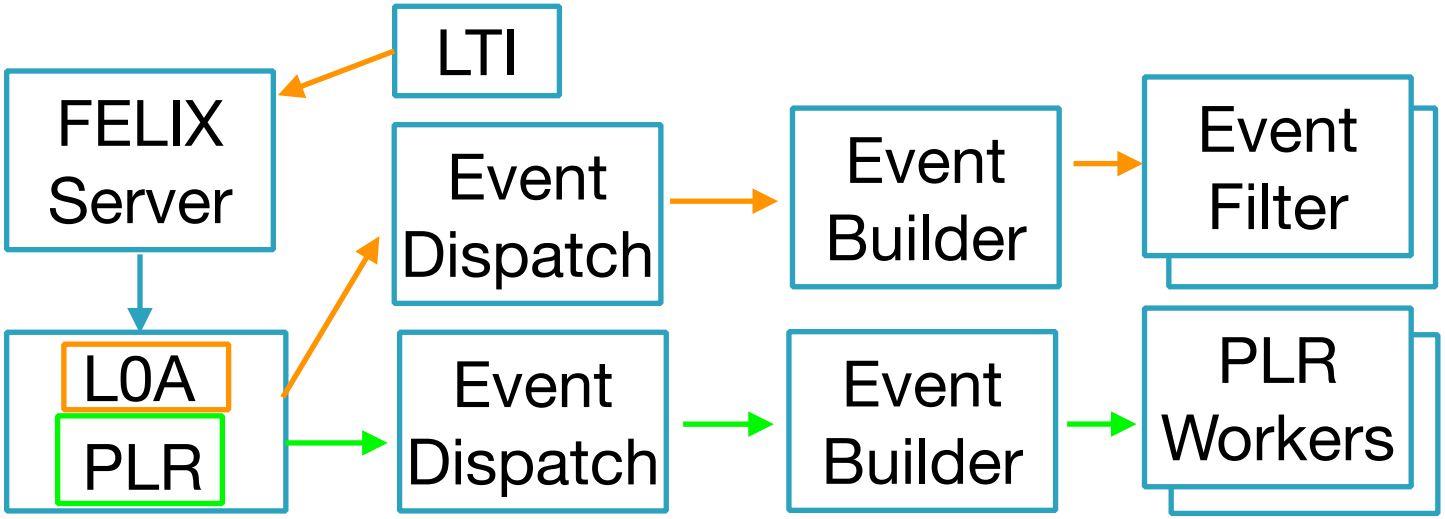
PLR Dataflow



- PLR-specific workers in Event Filter will receive data from PLR Data Handlers at 1 MHz+
  - Unpacking, clustering, counting, summing
  - EventFilter environment, many things shared with Pixel processing
  - Same dispatch rate as rest of Event Filter
- Limited data streaming (via Event Aggregator)
- Similar functionality as high-rate calibration data

EF resource needs for data distribution and processing needs some thought, eventually testing

Opportunistic Readout



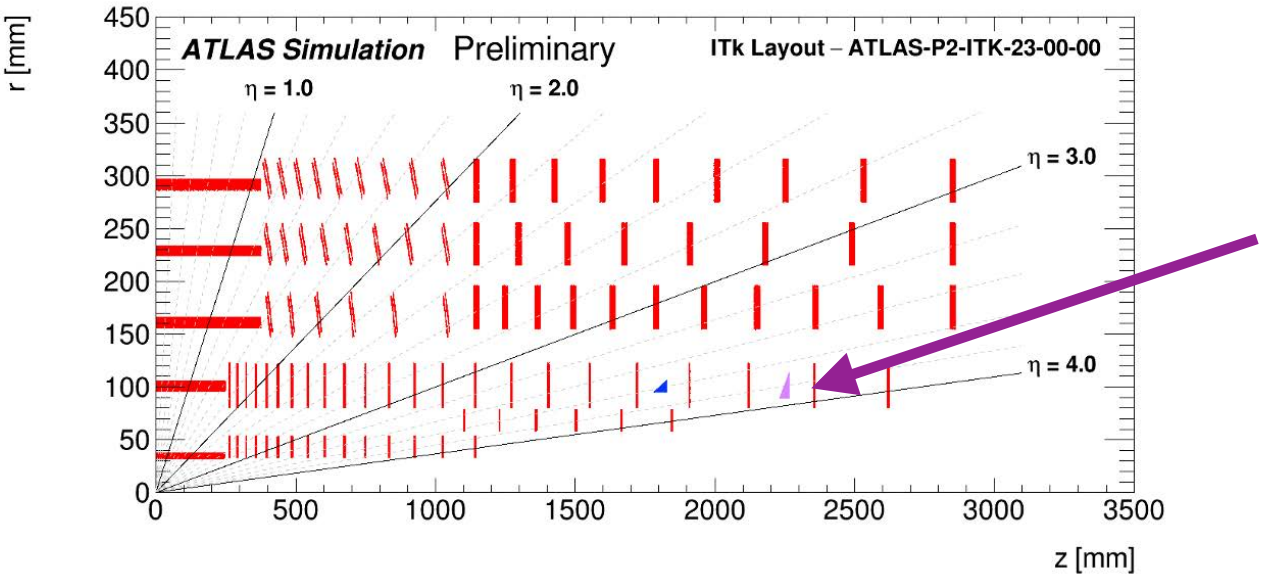
- EventDispatcher sorts events by L0A identifier
  - PLR in standalone mode generates its own identifiers
- LTI may be able to inject ATLAS L0A into PLR fragments
  - For PLR events that also have L0A on same BCID
- DH (or ED) could then make copies that look like a normal ITk fragment
  - Needs scheme to get L0ID into fragments
  - Needs addition to DataHandler (or Dispatcher) functionality
  - Approx ~(1 MHz / 40) ATLAS triggers would have PLR hits

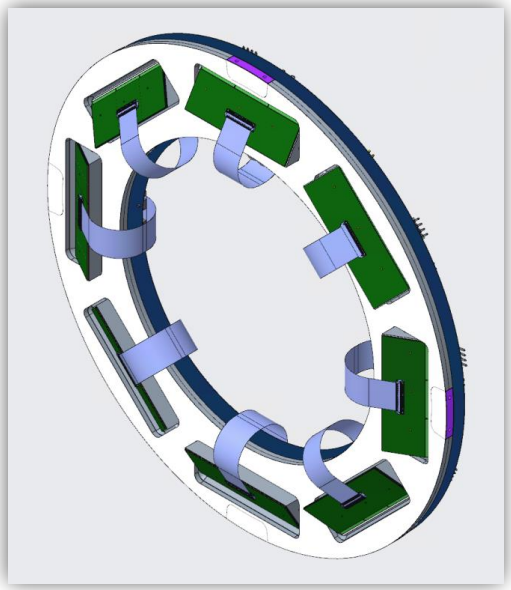
Comments

- Goal is to keep as much of PLR common with ITk pixels as possible (config, calibration, DCS, FELIX…)
- General DAQ layout and performance needs to be looked at again with benefit of 3 years more knowledge, descoping to 3 links, data from real readout tests
- Main PLR readout scheme similar to other high-rate calibration stream needs - is there a TDAQ plan for these?
- PLR Event Filter processing needs some thought
  - would welcome a new group to take this on
  - resource estimates from ITk pixels for unpacking?
- Opportunistic readout is challenging but interesting, need collaboration with CERN group
  - Had some discussions with Thilo and Will in the past, they believed this was possible, but far from solved…

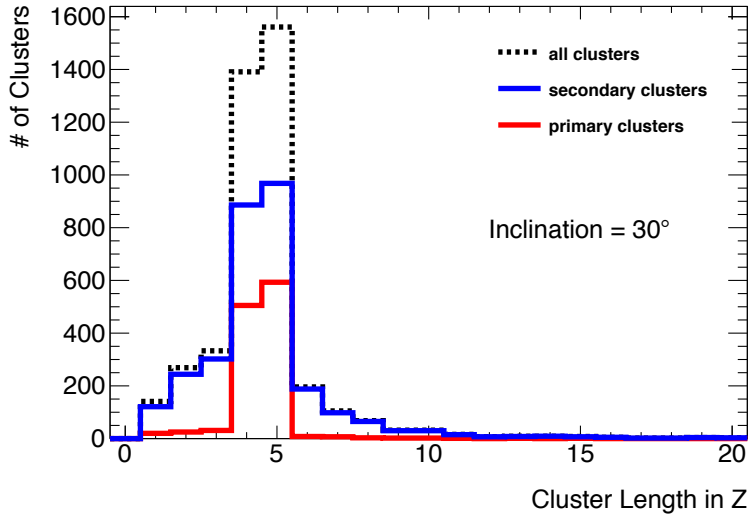
Backup

PLR in a nutshell





- ITK pixel ring at η ~ 3.9
- Use inner barrel ‘linear triplets’ (3D pixels, 25 x 100 μm^2)
- Incline sensors by 30 degrees (tracks from IP hit 4-5 pixels)
- Count clusters per event (PCC), linear to ~1% at μ = 200



PLR data products

- Count clusters per BCID per triplet for every triggered event
  - Input rate ~1 MHz, N x 3564 sums, output once per LB
  - N at least 16 (number of modules), could be more
- Write fraction of PLR raw data for monitoring and offline analysis
  - Highly pre-scaled, partial events, low rate (~ 100 Hz?)
  - Could be raw hits, or just cluster info (η, ϕ, TOT, size), histograms?
  - Probably save some amount of both, can't save all triggers
- Write PLR clusters when available for events with L0A "Opportunistic Readout"
  - PLR clusters on ITk tracks in forward rings allows powerful in-situ efficiency monitoring
  - Need ~200 Hz to measure 0.1% variation per "independent algorithm" over 5 hour fill
  - Output to partial-event stream - special low p_T track reco
- Provide fast (~ 1 Hz) BCID-blind sums for LHC
  - Lowest priority, not strictly necessary, but nice if easy

DCS/Calibrations

- Plan to run PLR like any other ITK disk
- Will need pixel calibrations
  - Running in separate partition, possibly separate tunings
  - Expect to be able to use same calibration software, or with only minor changes/different configuration
- DCS
  - Also plan to have this in common with ITK
  - One caveat is that we will want to turn PLR on once flattop is reached
  - Needs testbeam measurements and risk assessment (expectation is that this will be safe, to be confirmed)