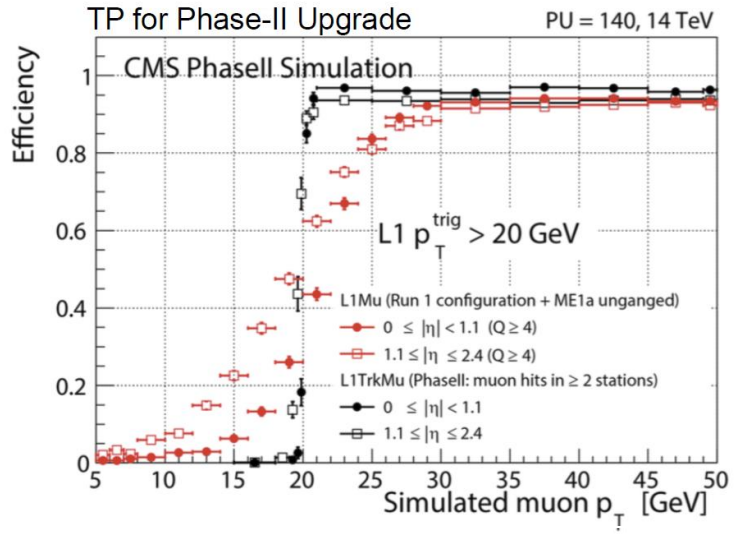
北京大学

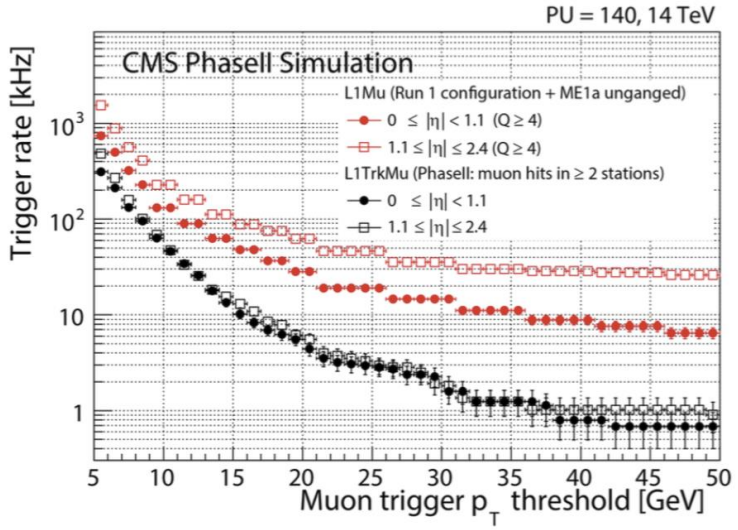
Level-1 Track Trigger R&D

Zijun Xu
Peking University
2016-12

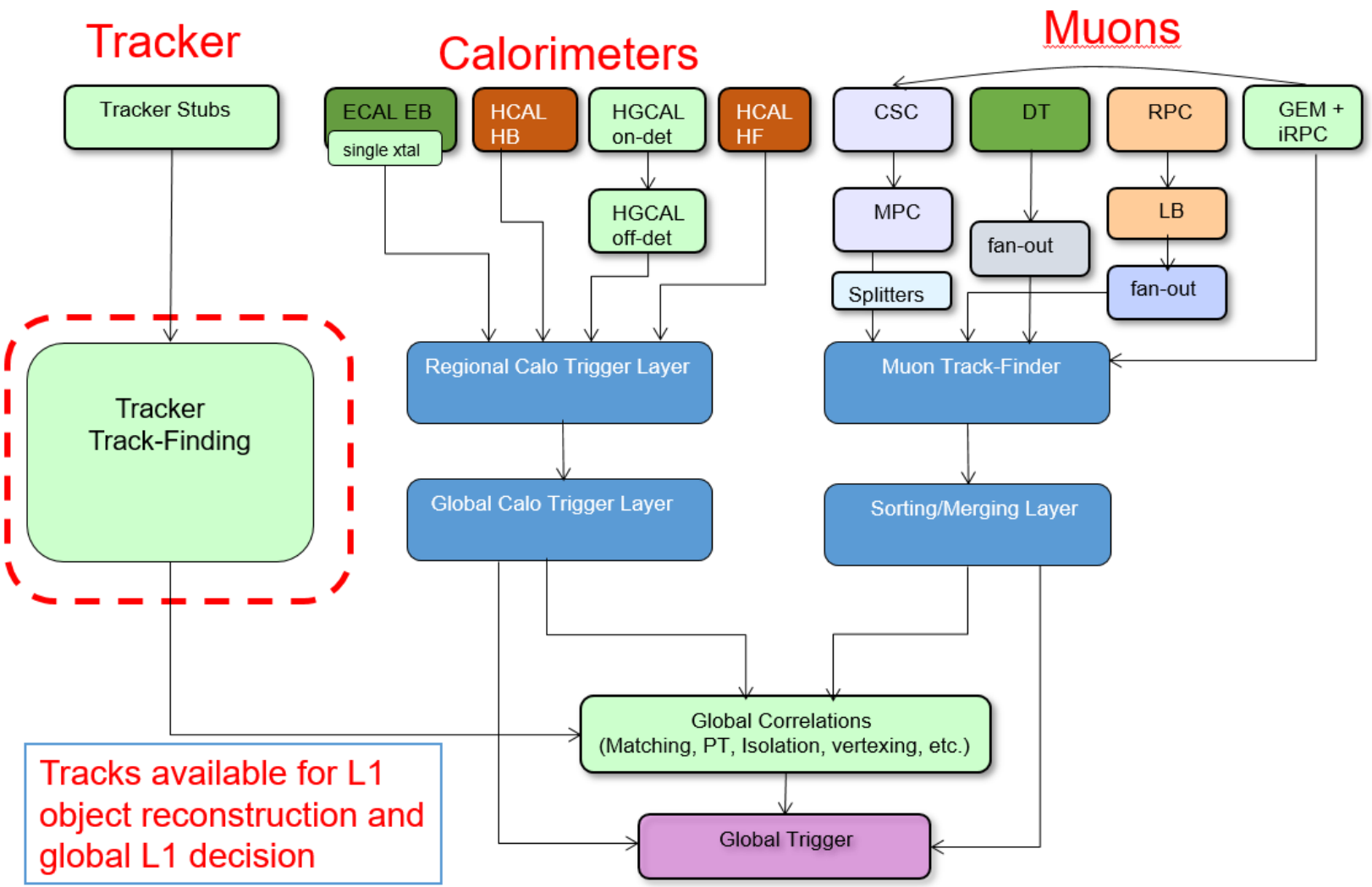
Level-1 Track Trigger for CMS Phase2 Upgrade

- HL-LHC, ~2025
  - Pileup 140 - 250
- Silicon based Level 1 Track Trigger
  - Be crucial for trigger objects reconstruction
    - Tracking is highly effective for pileup mitigation
  - Outer Tracker design will be optimized for Track Trigger
  - 40 MHz input
    - 100 Tbps raw data from Outer Tracker
  - Aiming for 4 μs latency
- For comparison: ATLAS Fast Tracker Trigger for Phase1
  - High Level Trigger
  - 100KHz input
  - 100 μs latency

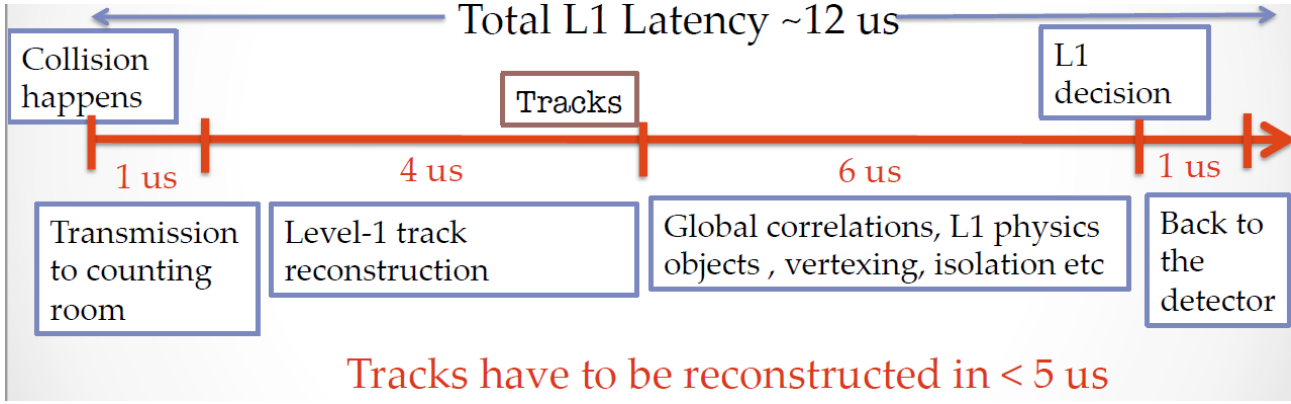




Proposed L1 Trigger Architecture for CMS Phase-2



L1 Track Trigger



Detector design for triggering

Data transfer

Partition detector into trigger towers/sectors

Data formatting

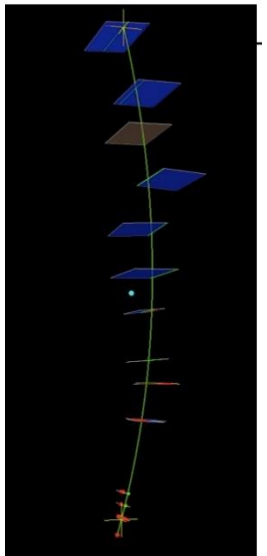
- AM Approach
  - proven by CDF/SVT
- Hough Transformation
- Tracklet-based
- …

Pattern Recognition

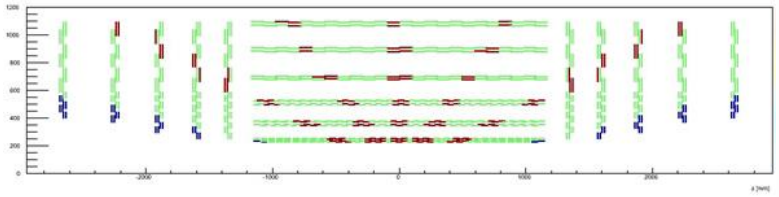
Finer pattern recognition

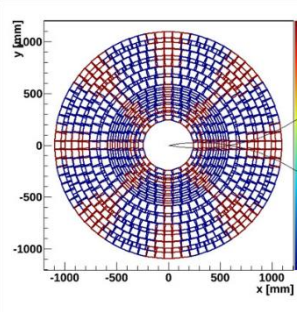
Track Fitting

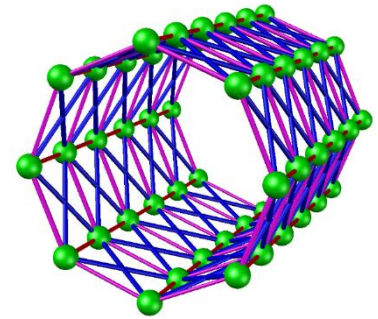
Goal 4μs



Track Trigger Architecture: Divide and Conquer



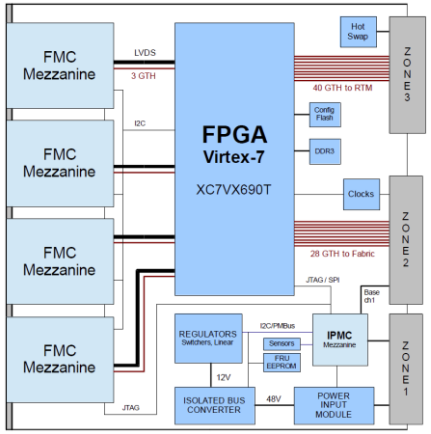


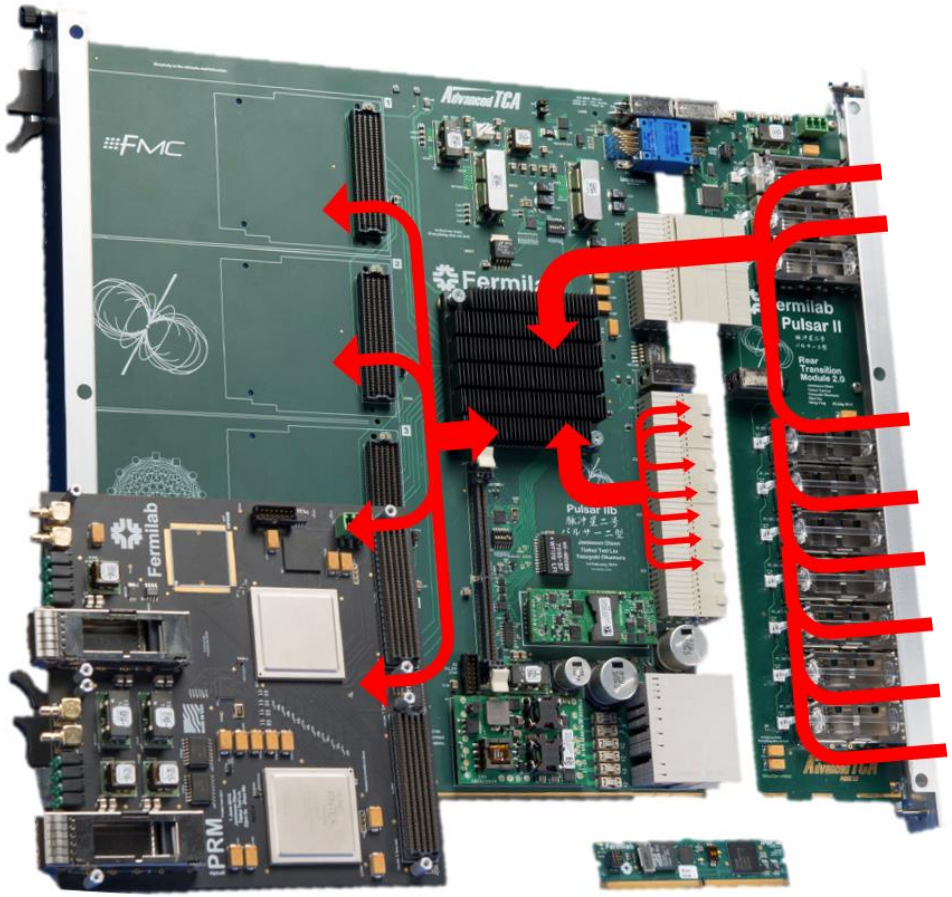


- 6x8=48 Trigger Towers
  - 48 Space multiplexing
  - 100 Tbps → ~2 Tbps per trigger tower
- One ATCA shelf per trigger tower
  - 10 blades for parallel processing
  - ~200 Gbps input per blade
  - 1 blade has up to 4 mezzanine cards (Tracking Engine)
  - time multiplexing up to 40
  - 40MHz → 1MHz processing per engine

ATCA platform
- I/O capability: Tb/s
- I/O interfaces Flexibility
- 99.999% Stability

Processing blade: Pulsar2b

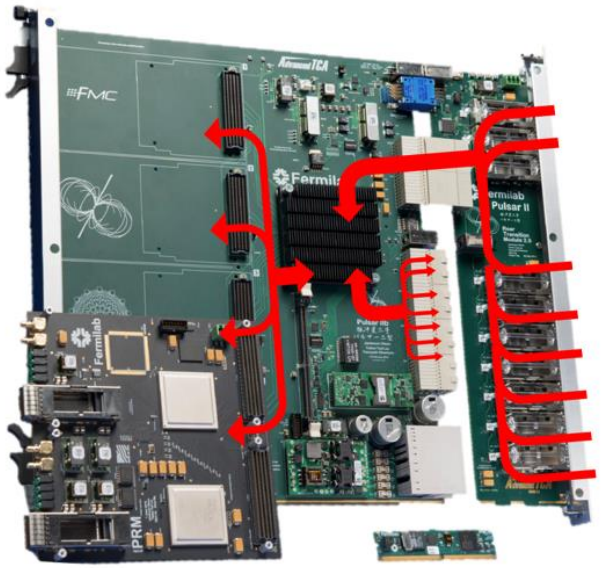


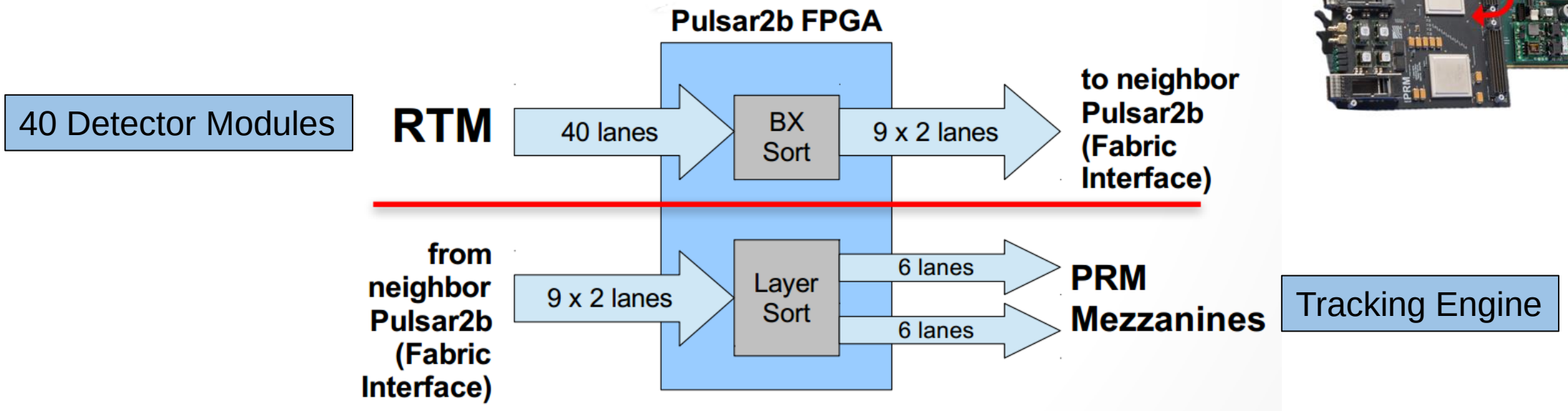


A general purpose designed ATCA blade

- Xilinx Virtex-7 FPGA
- 4 FMC mezzanine slots
- Pulsar2b I/O
  - Receiving raw data from detector by RTM
  - Receiving/Sending by full-mesh backplane for time multiplexing
  - whole data of one event sending to one PRM
  - Pattern Recognition and track Fitting is done inside one PRM

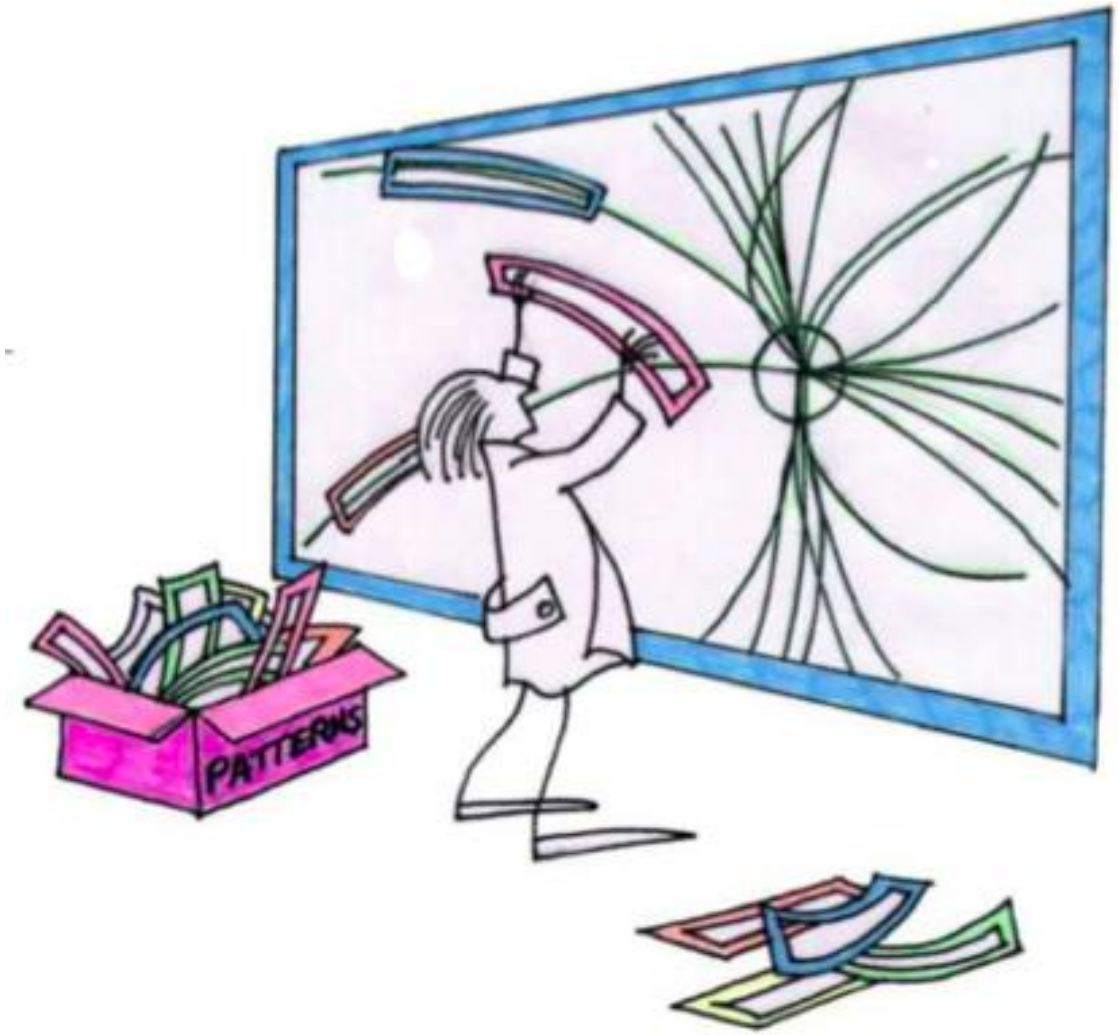
Data Formatting on Pulsar2b

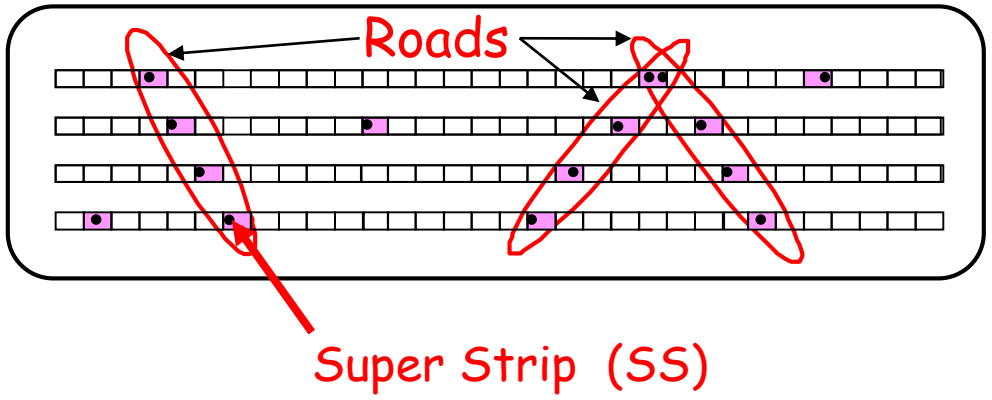


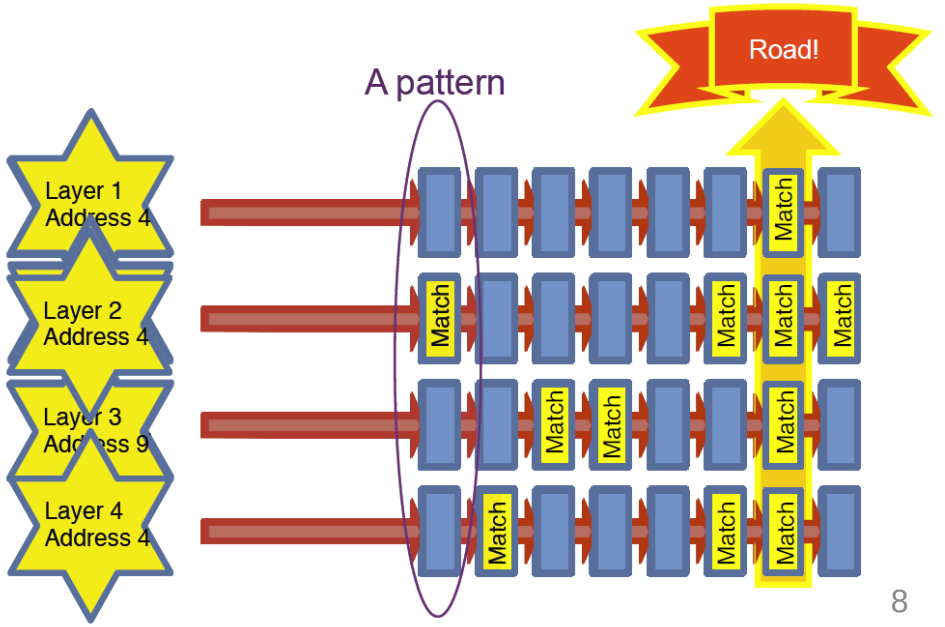


- One trigger tower has ~400 detector modules
- 10 Pulsar2b+RTMs receiving data from the 400 detector modules
- Data delivering latency : 1.2 μs
- Data transfer speed achieved 10 Gbps per GT channel

The Associative Memory Approach for Pattern Recognition

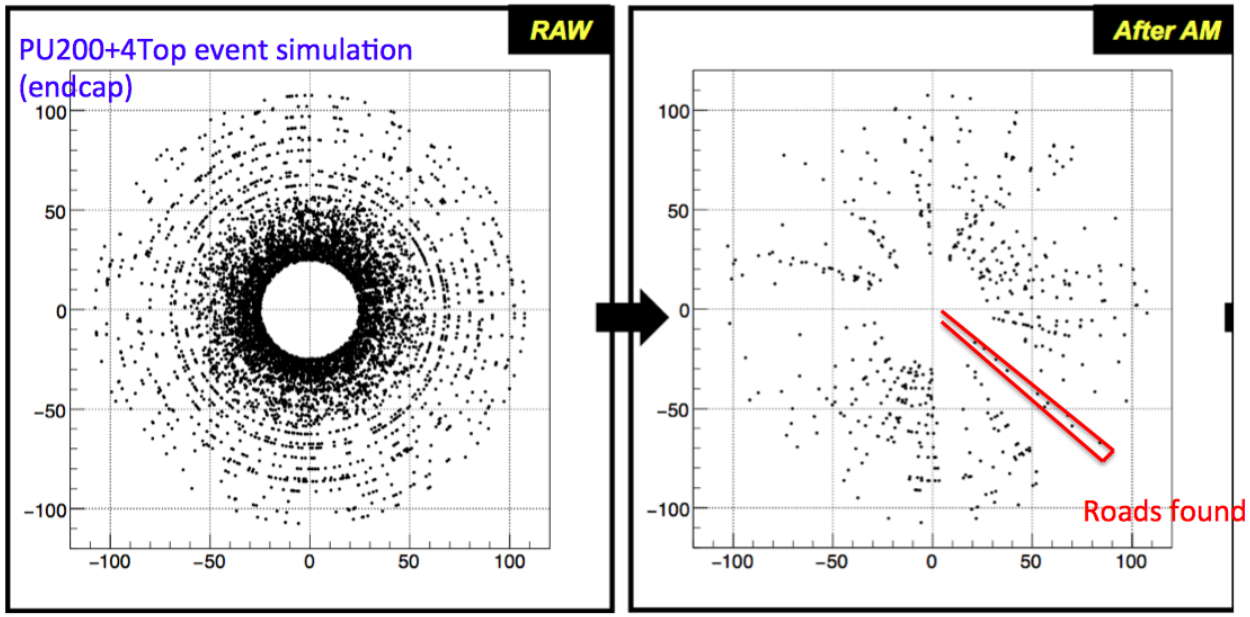






The Associative Memory Approach for Pattern Recognition

- Massive parallel processing to tackle the intrinsically complex combinatorics
  - Avoid the typical power law dependence of execution time on occupancy
  - Solving the pattern recognition in times roughly proportional to the number of hits
  - Two million patterns for each trigger tower
- Sorted Road output
  - high p_T road sent out first → keep high p_T track efficiency
- Roads already have rough track information

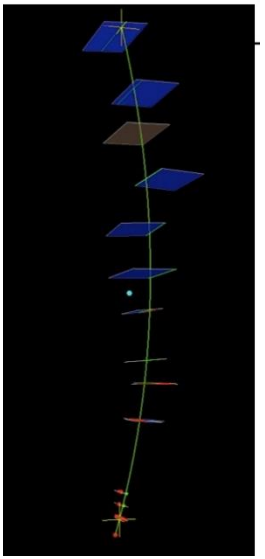


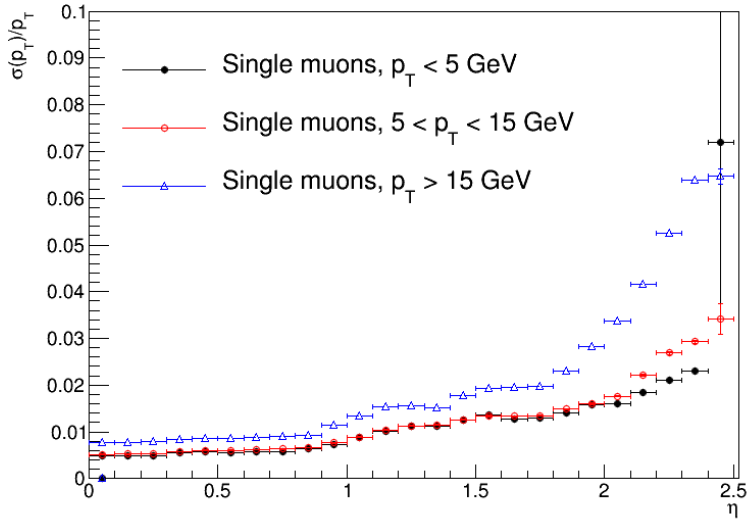
Track Fitting

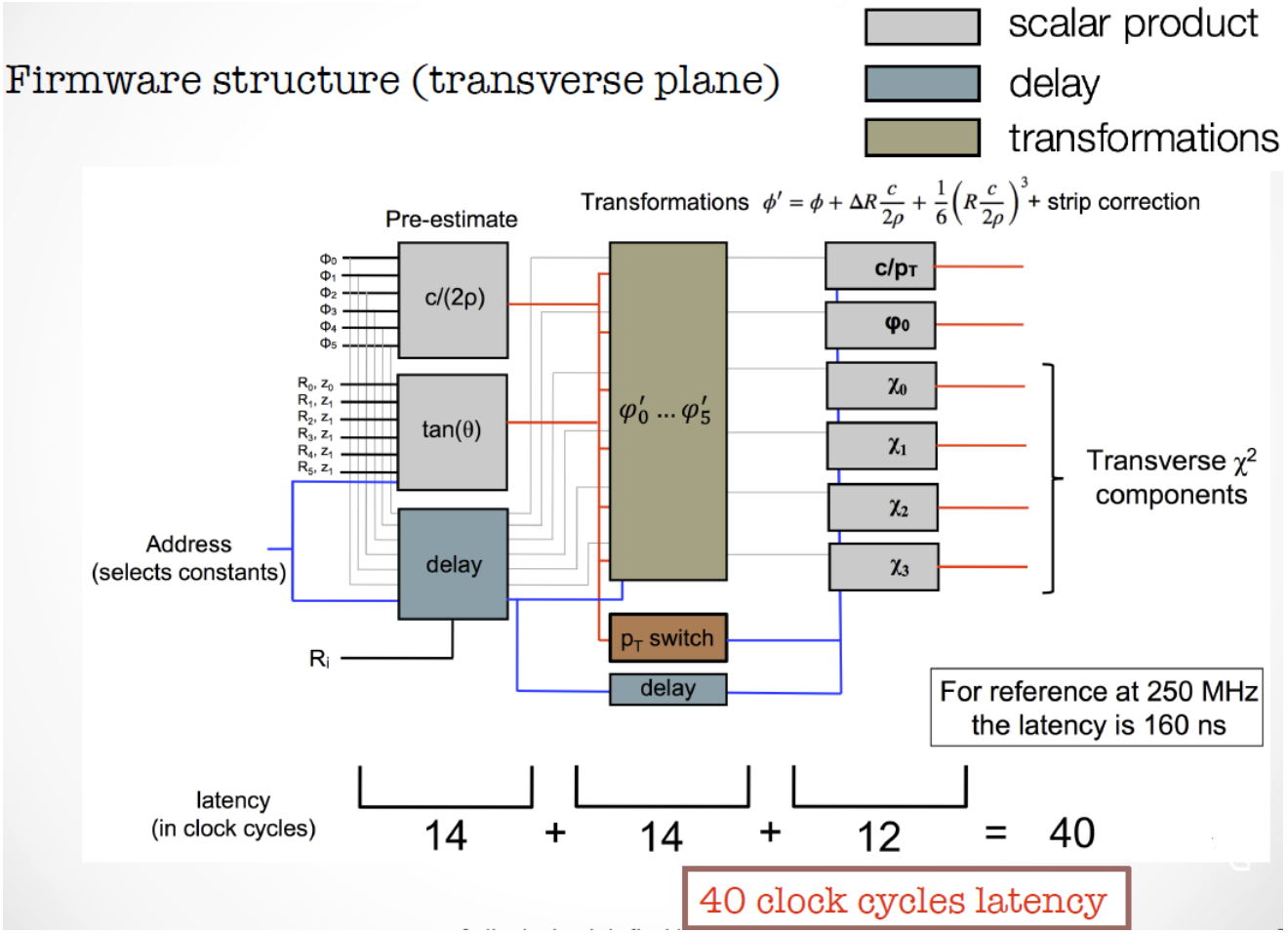
- Linear Track Fitting
  - Road is narrow enough for linear calculation
  - FPGA-friendly: LUT+DSP

Compatibility with a track: χ2/ndof

Track parameters: charge/p_T, ϕ_0, cot(θ), z_0, d_0

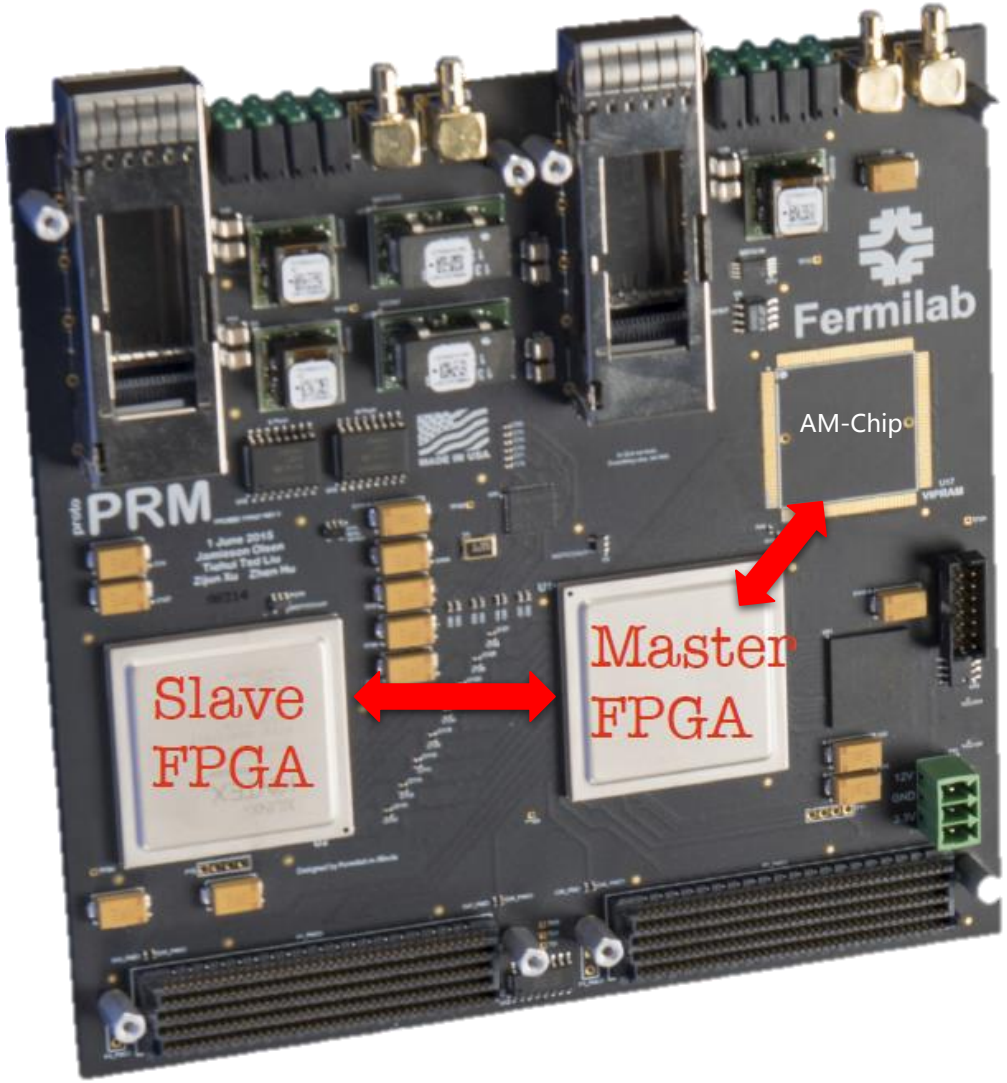




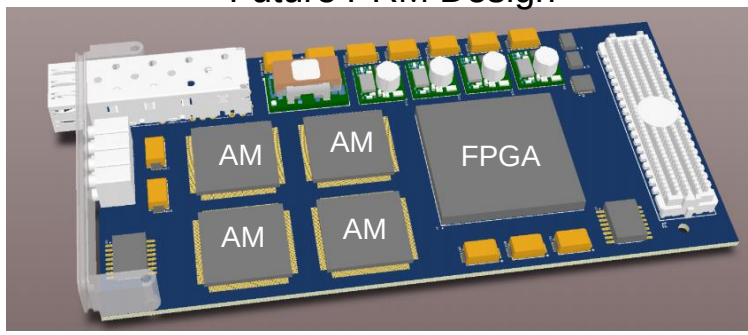


Latency: 0.16 μs

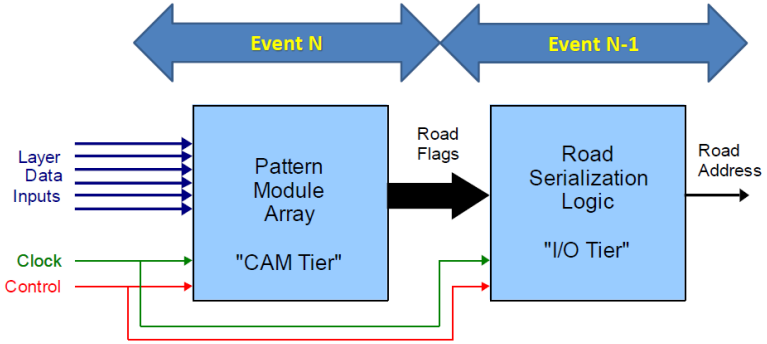
ProtoPRM: Tracking Engine



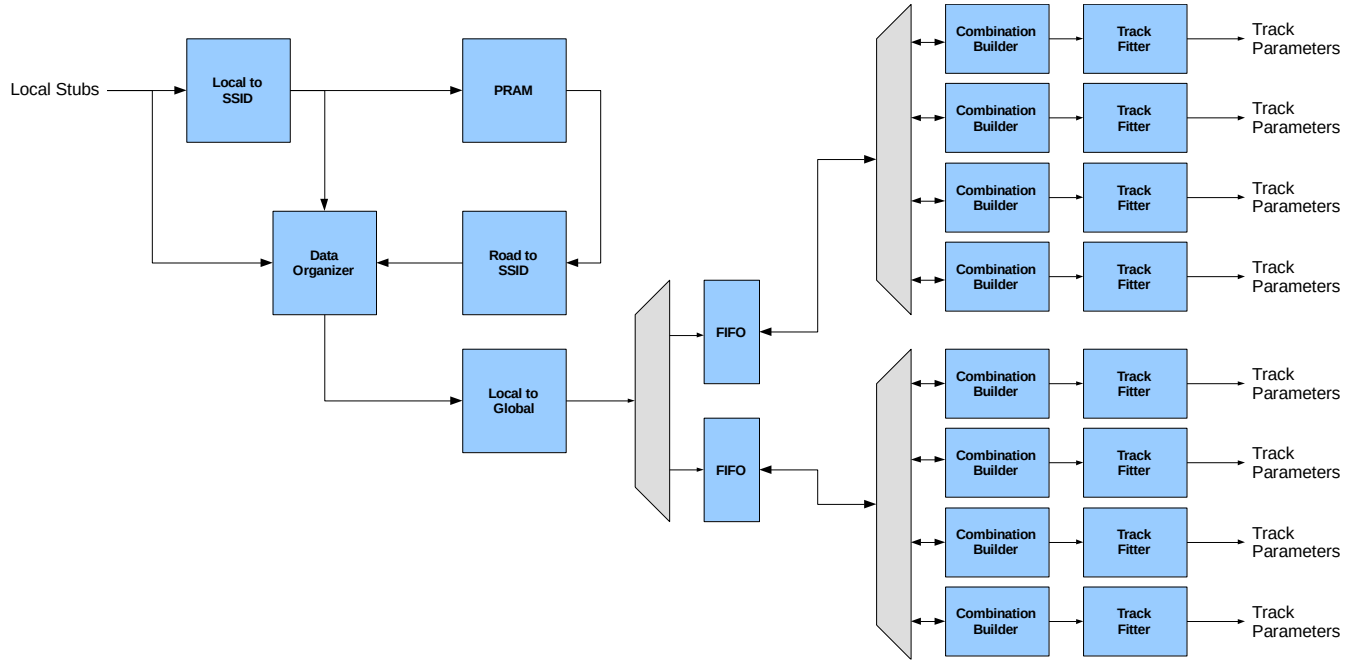
Future PRM Design

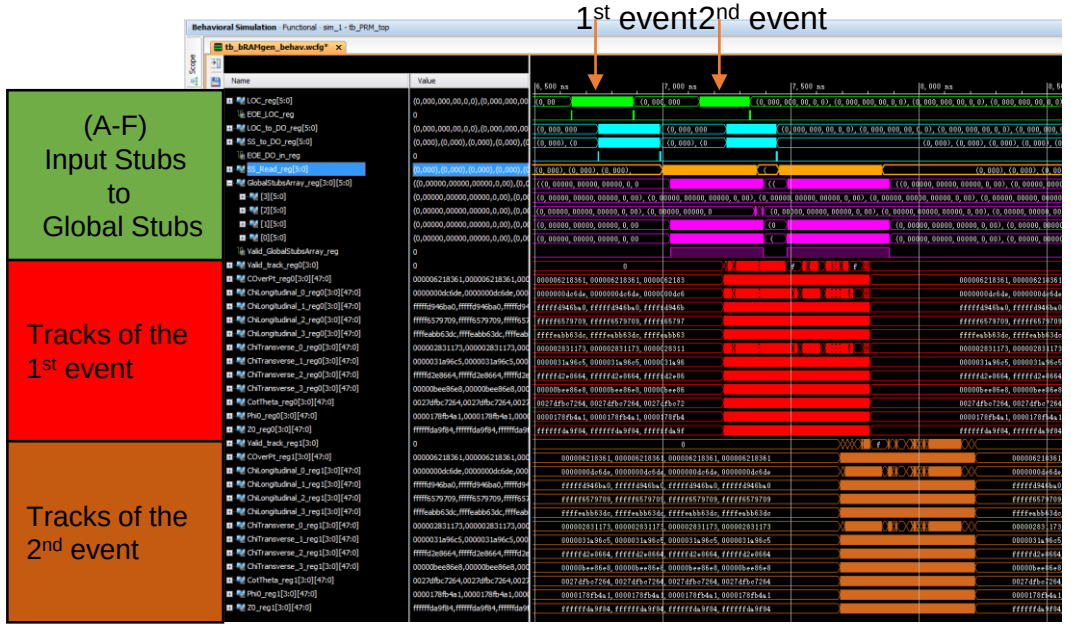


- Prototype tracking processing engine for demonstration
  - Kintex UltraScale KU060
- Data Organizer in the Master FPGA
  - Local Stubs from Pulsar2b
  - Super Strips out to AM
- AM in the Slave FPGA
  - FPGA implementation of AM ASIC

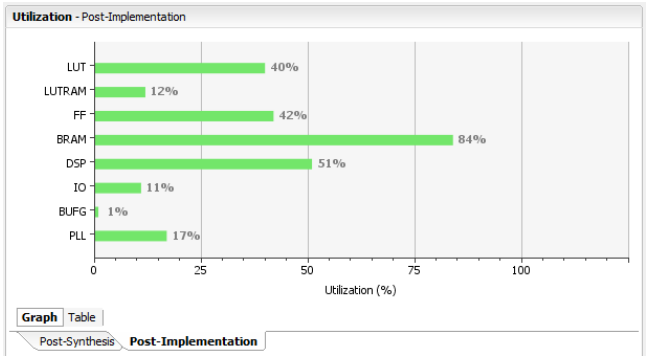


Pattern Recognition + Track Fitting Firmware





- Half of the FPGA resource is used
  - Kintex UltraScale KU060
- Latency
  - AM-Based Pattern Recognition: 0.6 μs
  - Linear Track Fitting: 0.2 μs



L1 Track Trigger Timing

Detector design for triggering

Data transfer

Partition detector into trigger towers/sectors

0 μs

Data Formatting

- AM Approach
  - proven by CDF/SVT
- Hough Transformation
- Tracklet-based
- …

1.2 μs

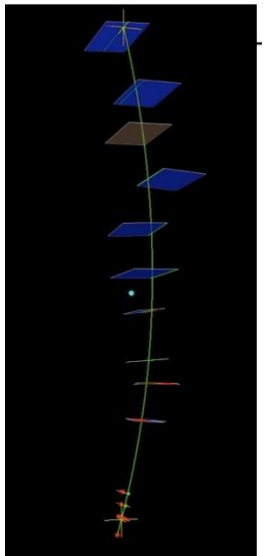
Pattern Recognition

Finer pattern recognition

1.8 μs

Track Fitting

2.0 μs



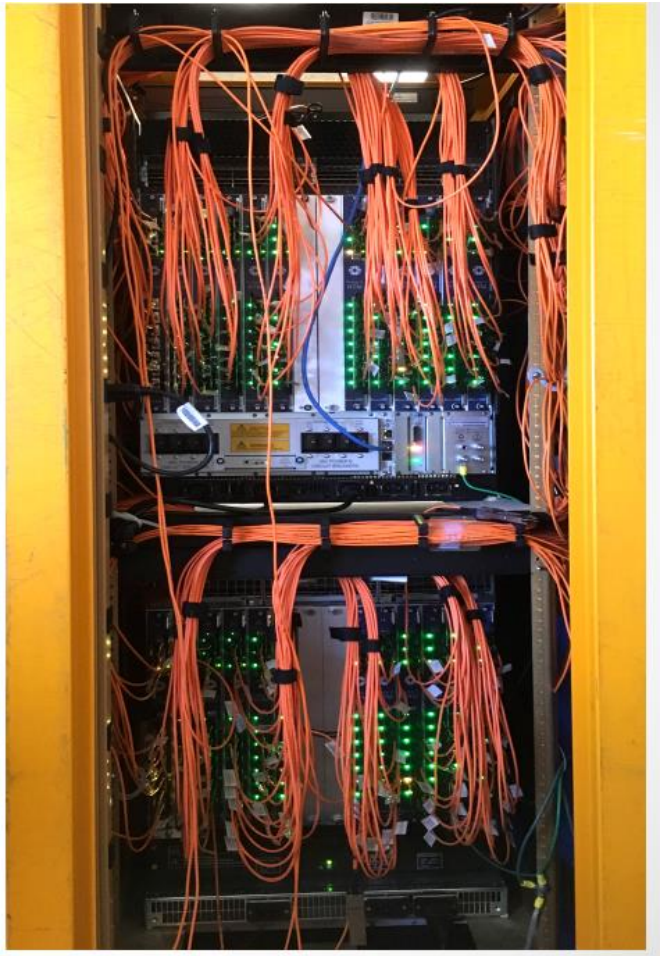
Within the target latency budget of 4 μs

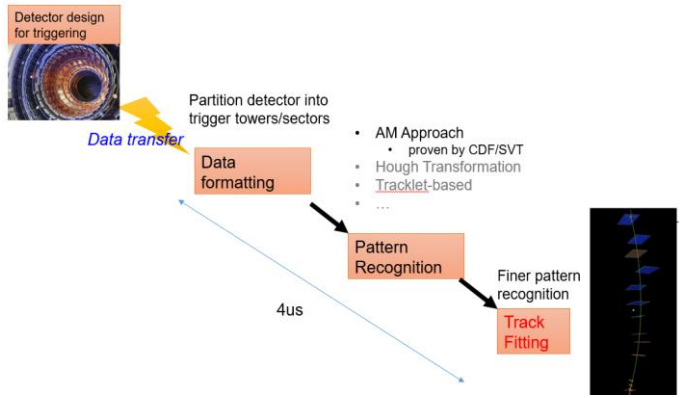
- After 2.0 μs: first track output
- 2 μs left to do more processing
  - high p_T Jets

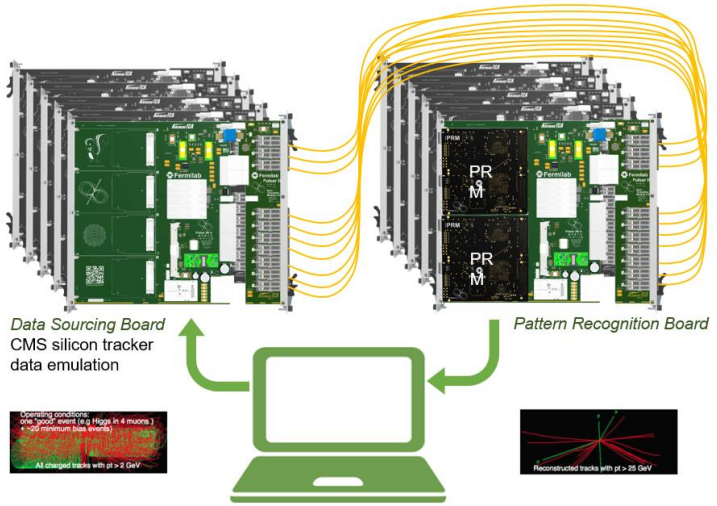
Track Trigger Demonstration

Front view

Back view





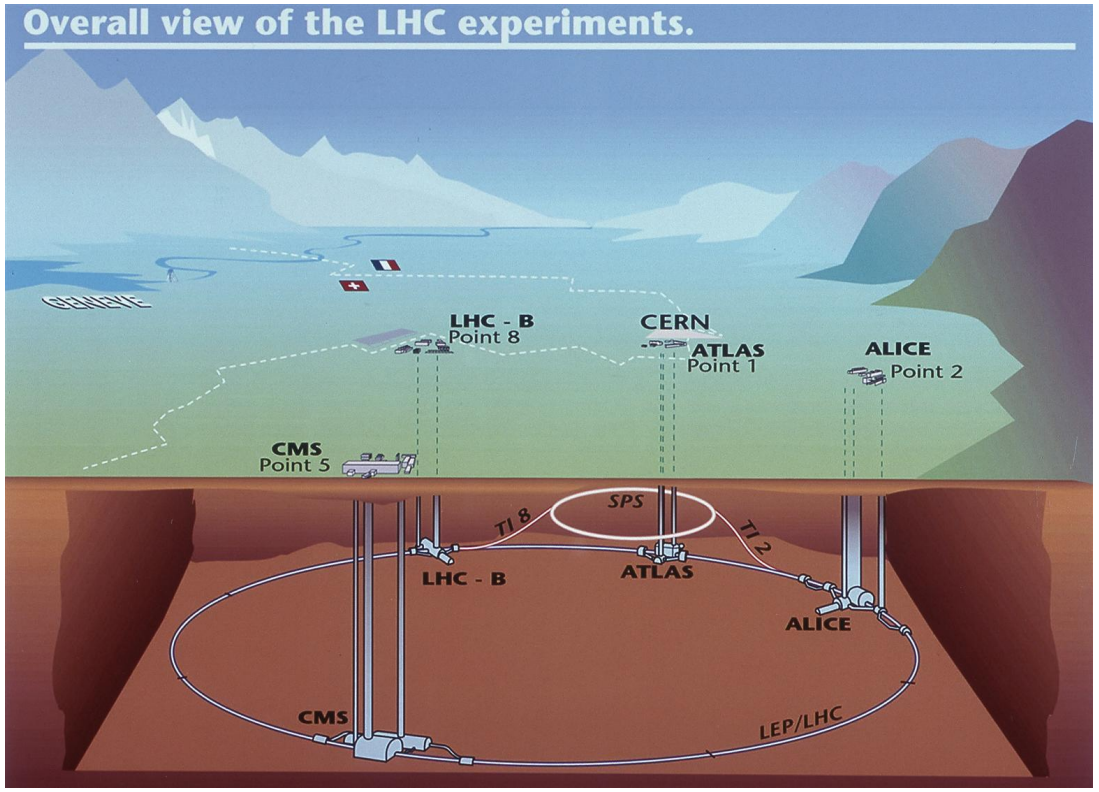


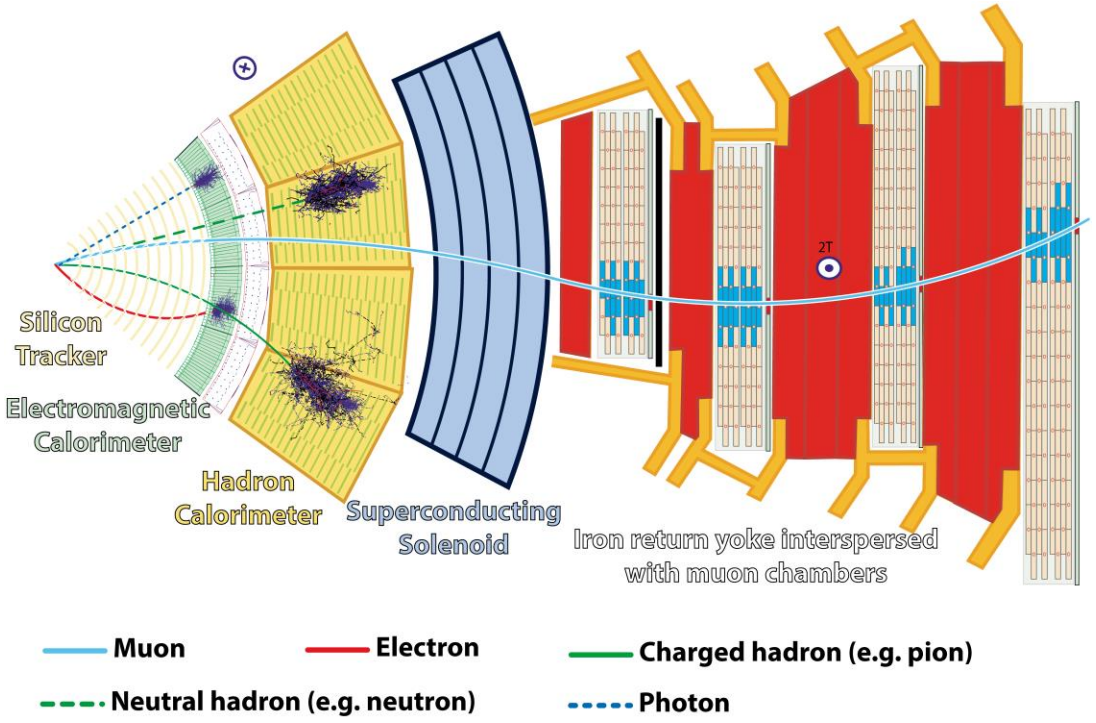
Conclusions

- Having Level-1 track trigger is crucial for success of CMS physics goals in HL-LHC
- Highly challenging as track triggering at this scale and speed has never been implemented before
- Track Trigger System is demonstrated with today's technology
  - Within the target latency budget of 4 μs

Backup

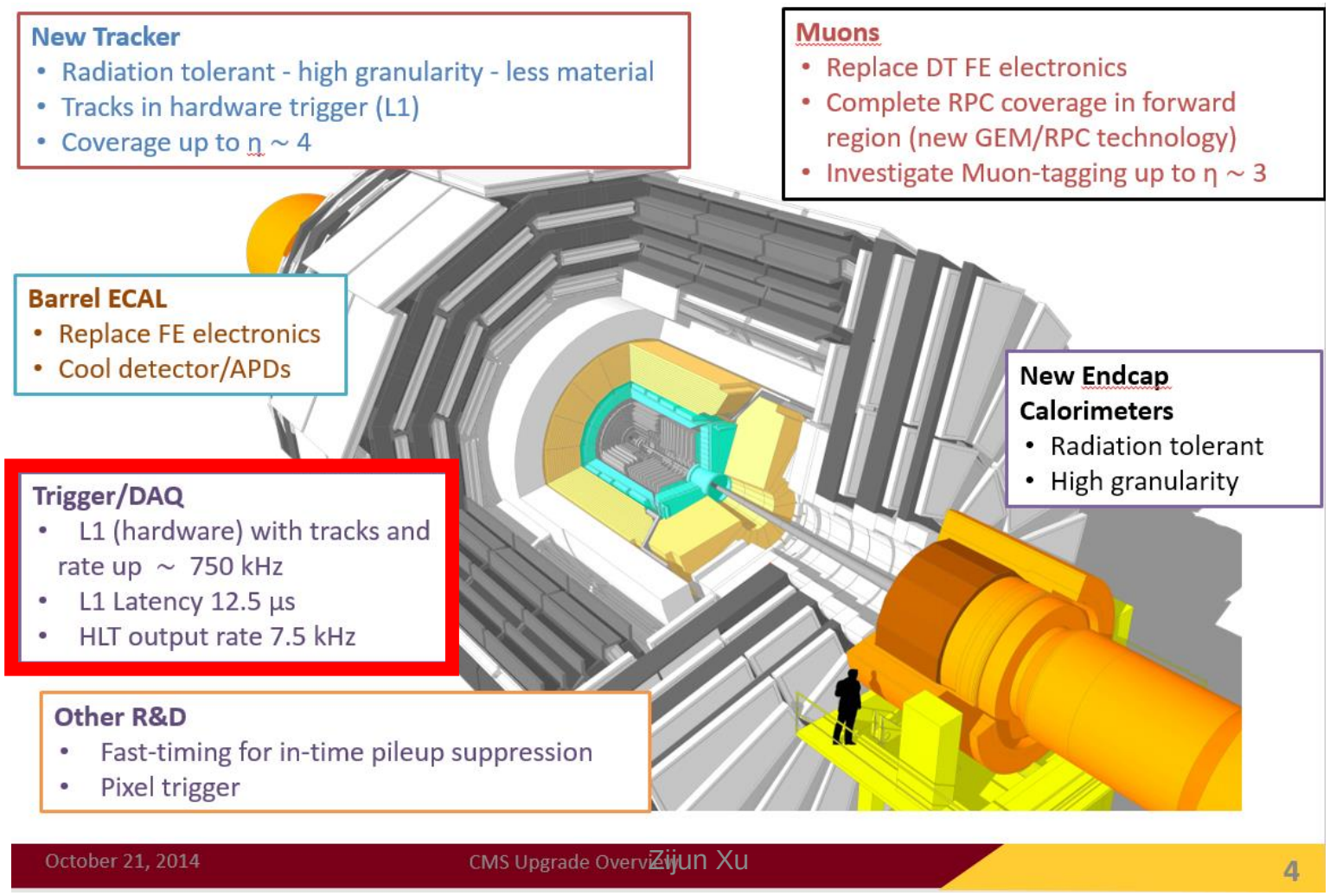
LHC and CMS



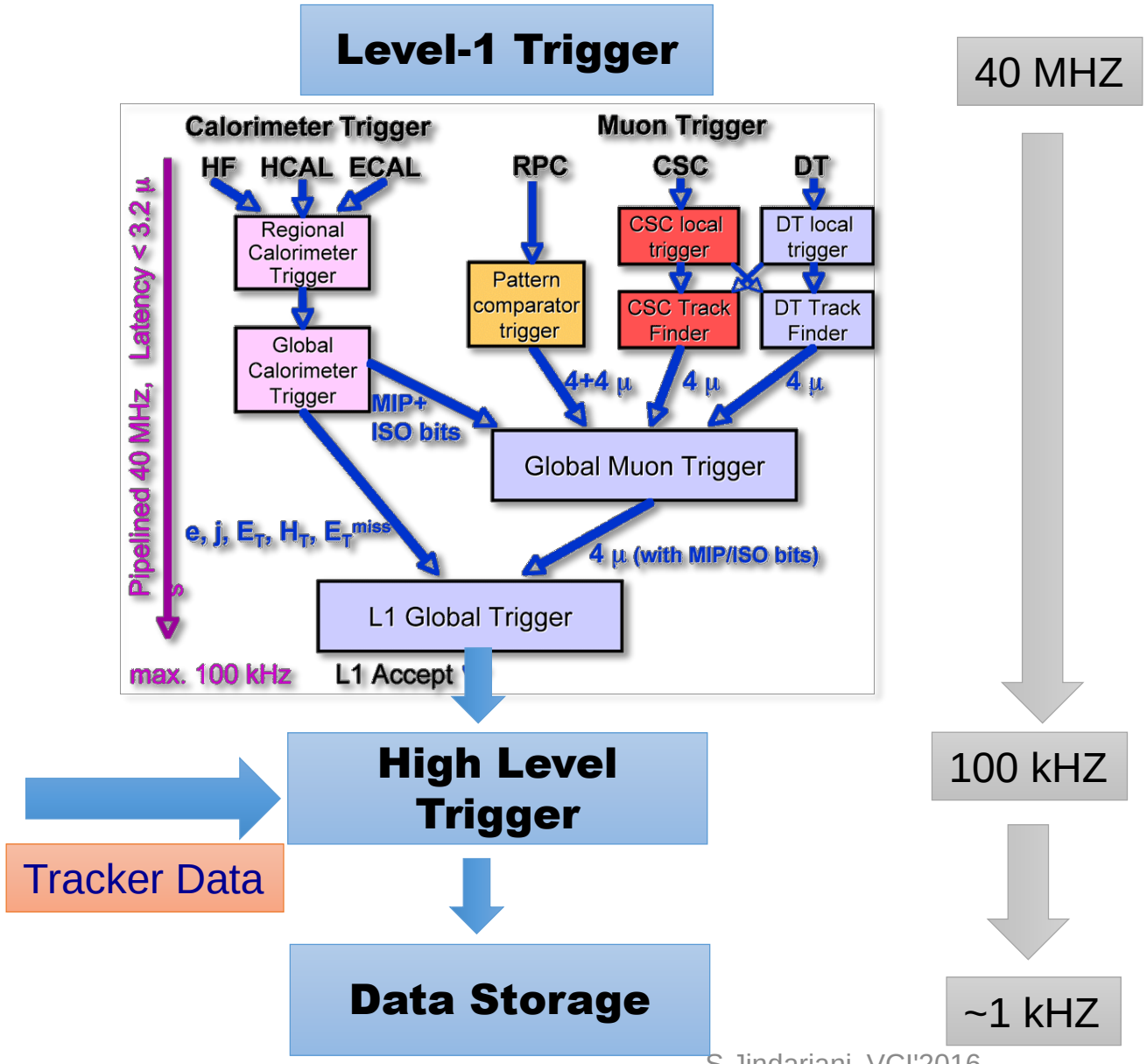


ATLAS, ALICE, **CMS**, LHCb

CMS Phase2 Upgrade



Current CMS trigger

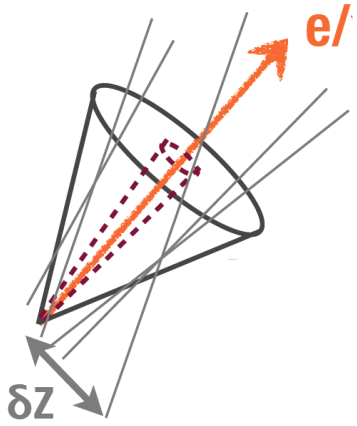


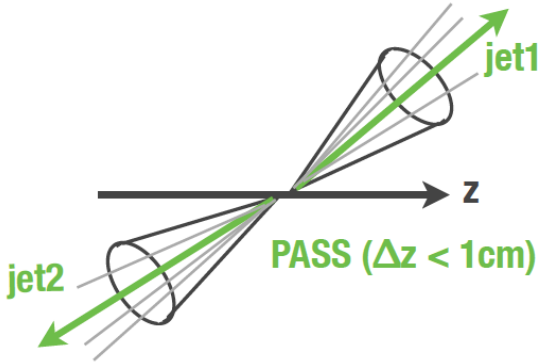
- L1 trigger system reduces event rate from 40 MHz down to 100 kHz
- Until HL-LHC, Level-1 decision is based solely on calorimeter and muon system information
- Tracker data available at the HLT level only

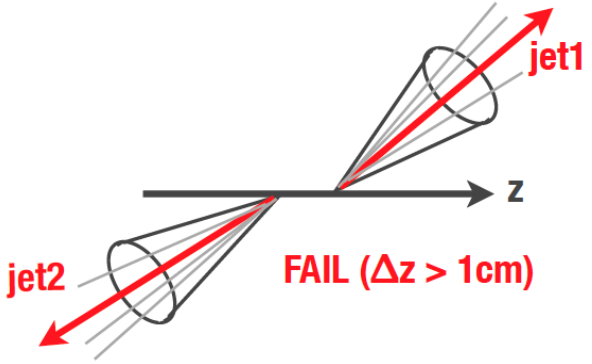
Tracking in L1 trigger:

Tracking is highly effective for pileup mitigation

- Electron/Photons
  - Extra measurement - Rate Reduction
  - Isolation
- Muons
  - Excellent Pt Resolution
  - Isolation
- Tau Triggers
  - Multiprong
- Separation of Interactions
  - Hadronic/Multi-object Triggers
  - Track-based Missing Energy



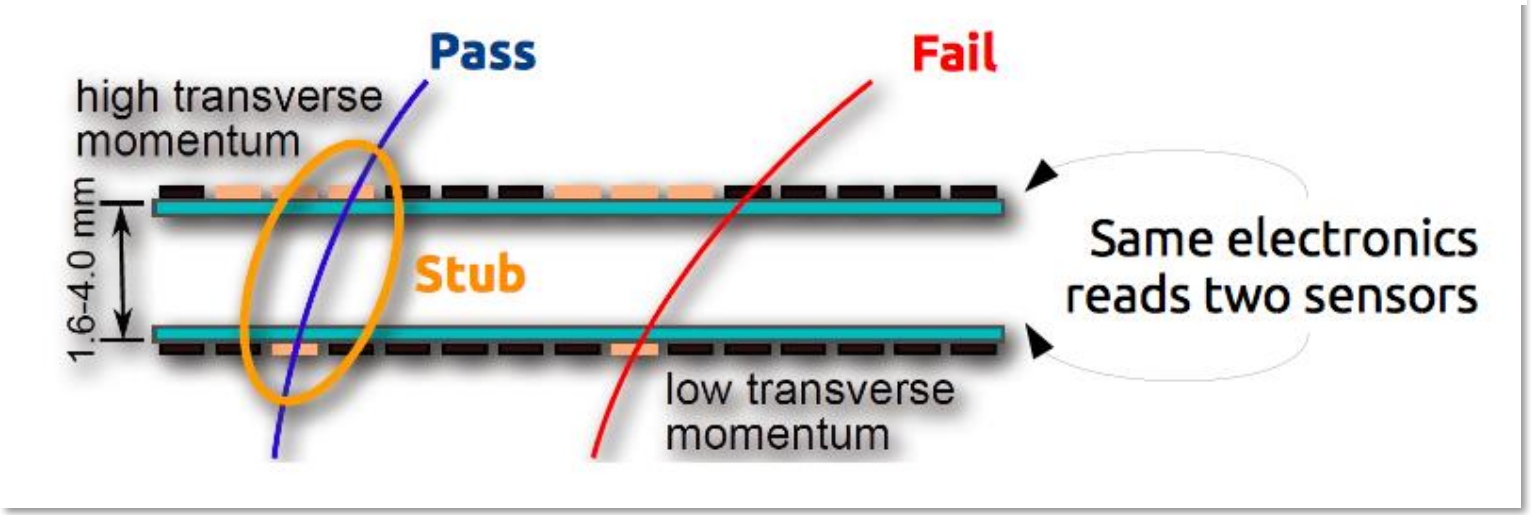




New CMS Tracker

More on the tracker in the talks by Giacomo SGUAZZONI and Axel KONIG (Wednesday)

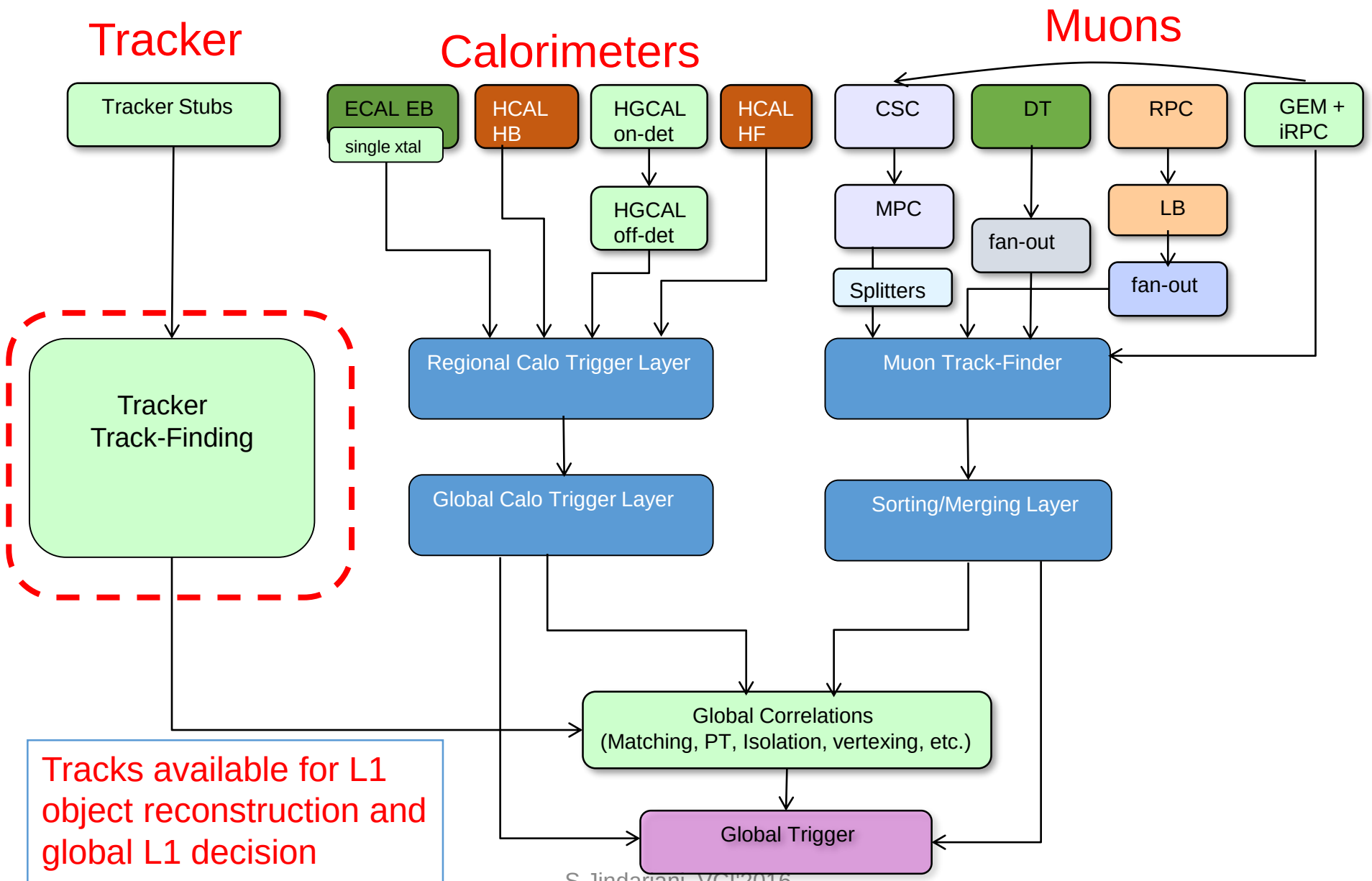
Tracker design is from the ground up done for triggering



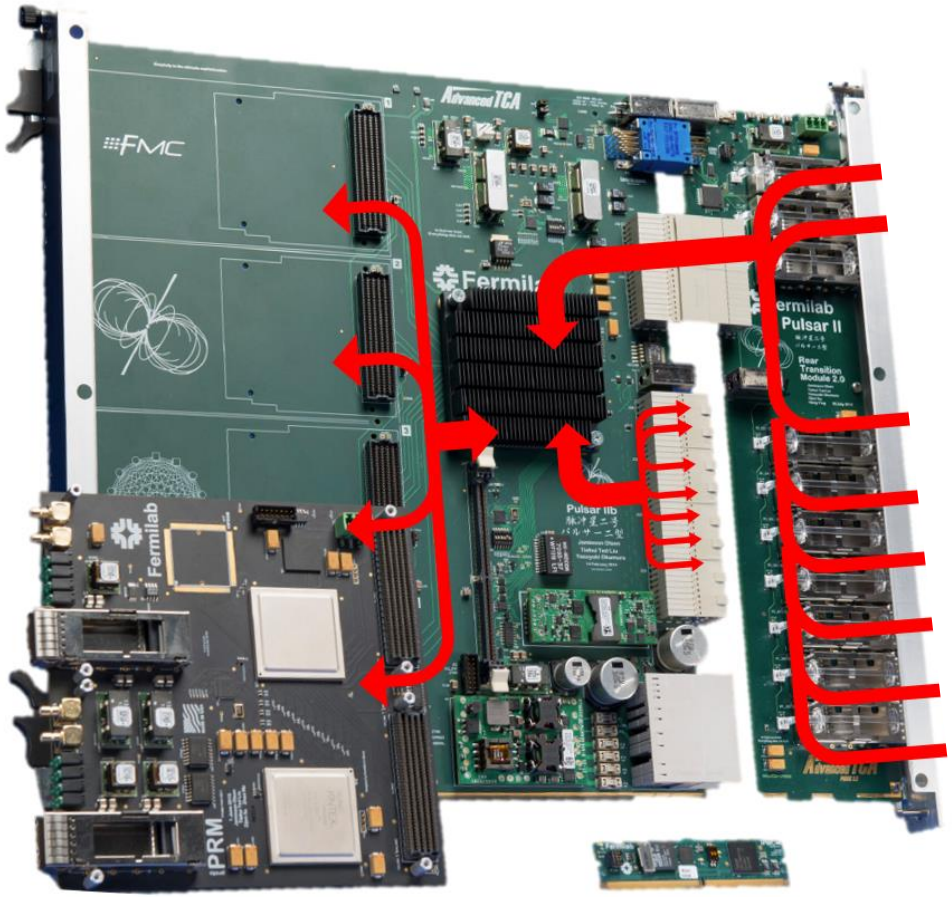
- Stub = pair of clusters in the 2 sensors of a module within a predefined strips window (enabling pT cut at the module level).
- Pass/Fail window is programmable (2 GeV default cut)
- Stubs drastically reduce (by a factor 10-20) the amount of data to extract from the tracker @40MHz
- Stubs allow L1 tracking possibility

- **~15000 modules transmitting**
  - **p_T-stubs to L1 trigger @ 40 MHz**
  - **Full tracker readout @ 750 kHz**

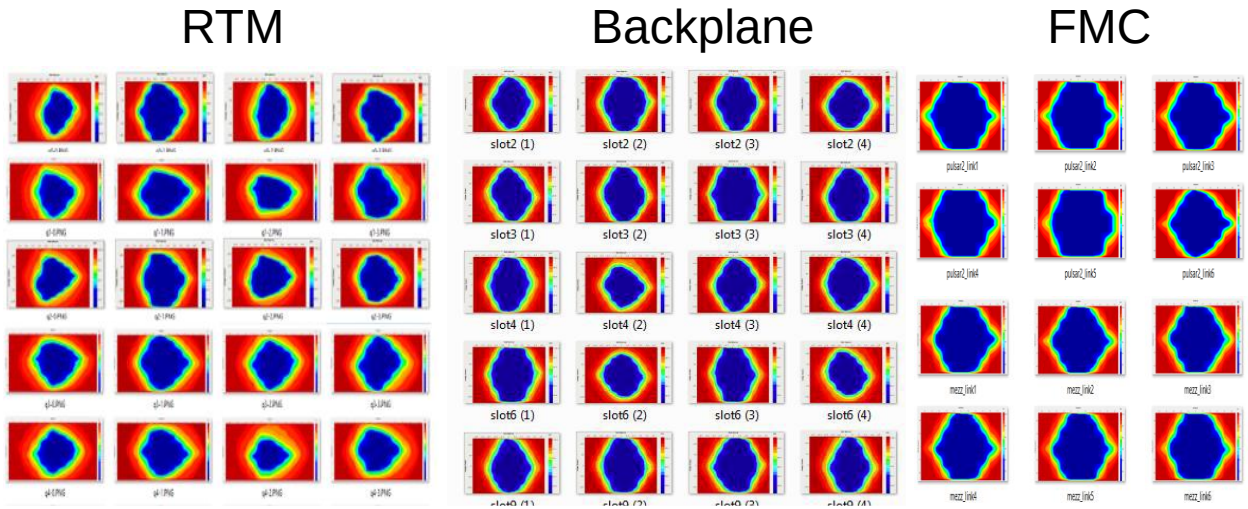
Proposed L1 Trigger Architecture



Processing blade: Pulsar2b



- IBERT Test for GT high Speed Link
  - 10 Gbps per link achieved
  - Total I/O bandwidth of one Pulsar2b up to 1.6 Tbps



Linearized track fitting

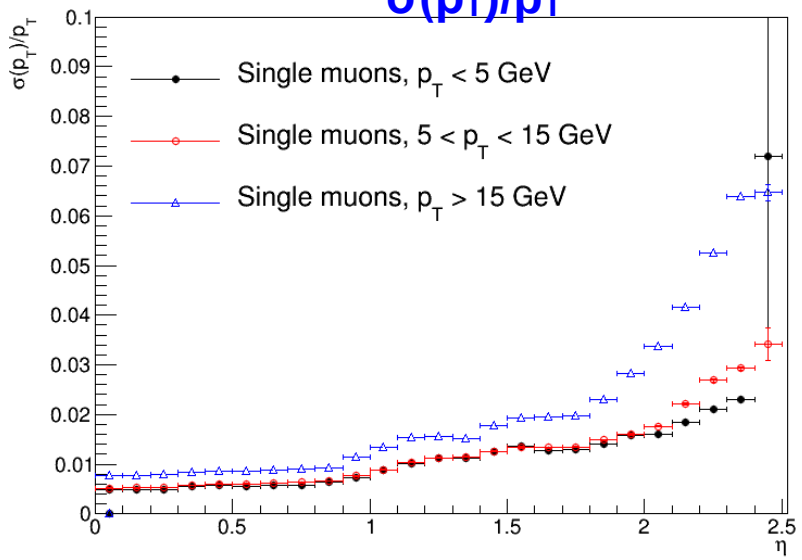
Given a set of stubs estimate:

- compatibility with a track: χ^2/ndof
- track parameters: charge/p_T, Φ_0, z_0, $\cot(\theta)$ and d_0

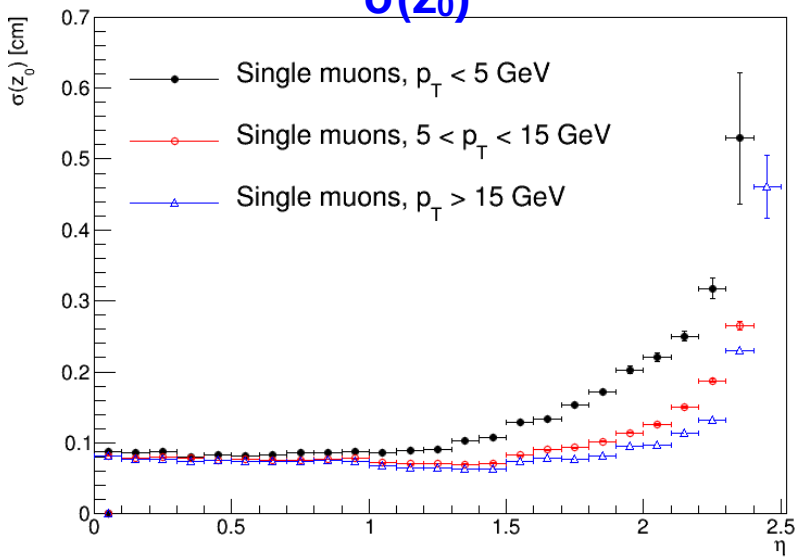
Method: Linearized Track Fit $\phi_0 = \sum_i A_i \Delta\phi_i + \bar{\phi}_0$ where $\Delta\phi_i = \phi_i - \bar{\phi}_i$

New Idea: To minimize number of constants transform the tracker into a smooth cylinder (only 20k constants for the entire tracker)

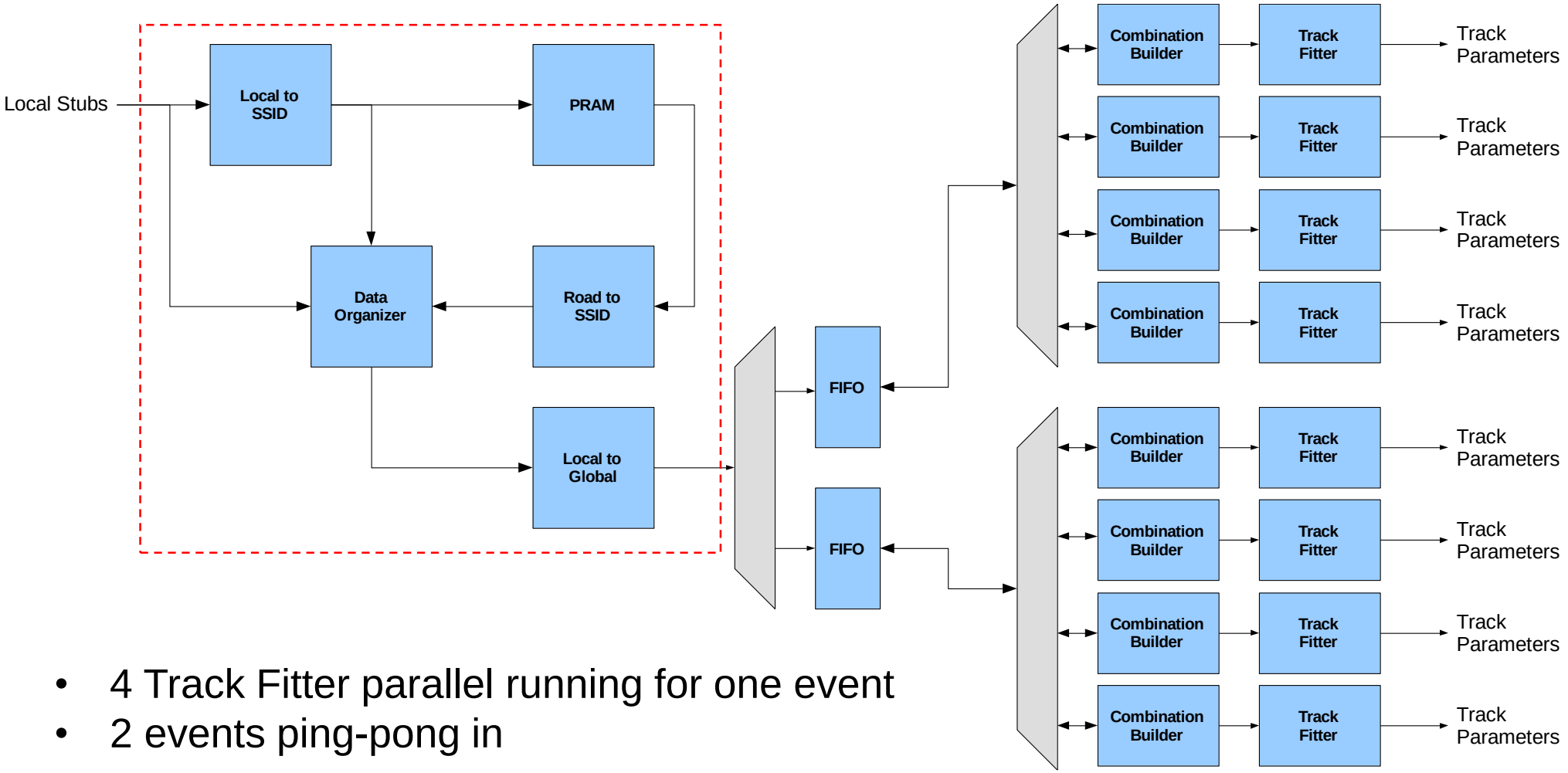
$\sigma(p_T)/p_T$



$\sigma(z_0)$



Pattern Recognition + Track Fitting

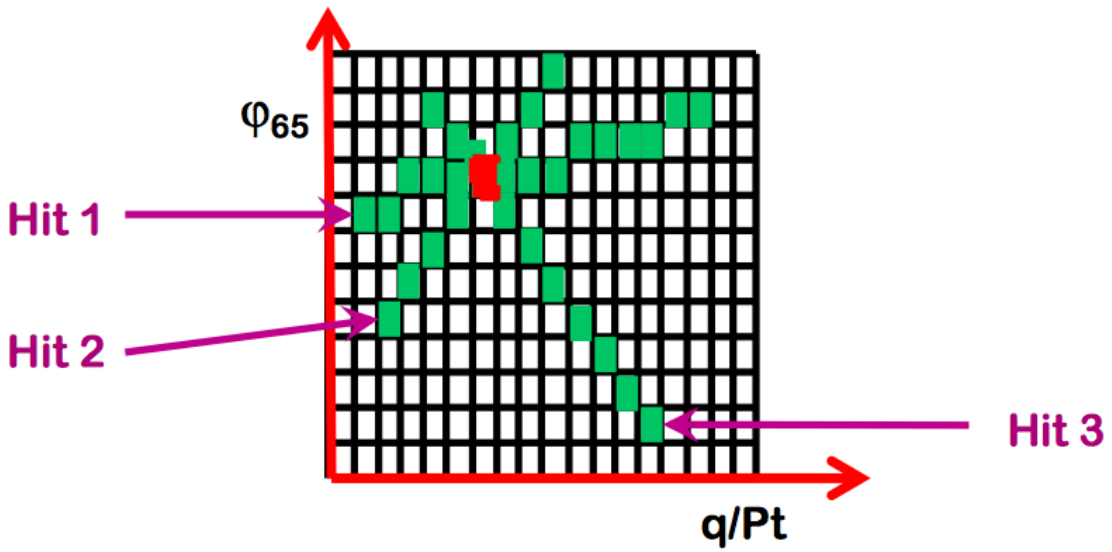


- 4 Track Fitter parallel running for one event
- 2 events ping-pong in

Review: The art of asking the right questions

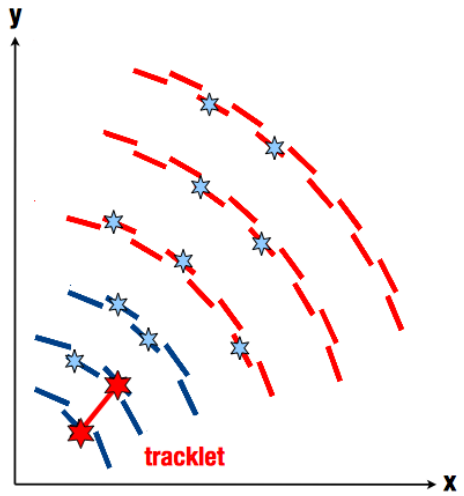
- The question is not whether we need ASIC or not
  - Given any HEP data/signal processing challenge, there are times in history when ASIC is really required, or preferred, and there are times when FPGA has reached the point to do the job comfortably (no more ASIC needed)
  - ***Where are we now with CMS phase 2 tracking trigger, & in a few years?***
- ASIC should be considered IF and ONLY IF (at least) one of the following is true:
  - ***(1) If it is the only way to solve the problem, or***
  - ***(2) If it provides enough performance safety margin and robustness,or***
  - ***(3) If it can reduce the overall system level cost in an effective way***
- To answer these three questions, we need to know:
  - Pure FPGA can handle it comfortably? (the two FPGA approaches)
  - What can ASIC really help and what ASIC required (this session)
  - What it takes to implement such an ASIC (next session)
- ***All three approaches/efforts under review will help answer these questions***
  - ***We have been all working together towards the same goal***

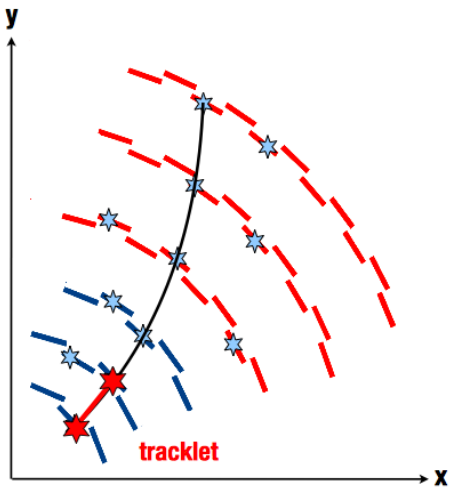
FM-TMT

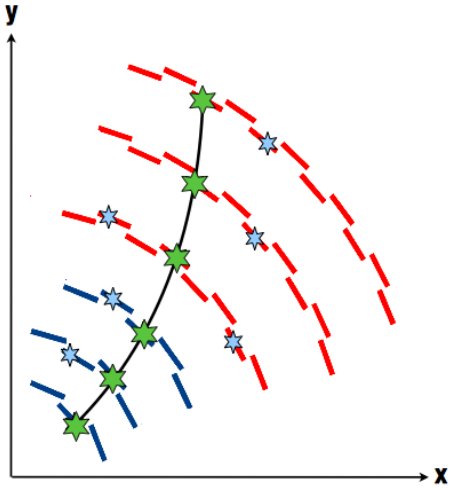


- Track finding done using Hough Transformation (HT)
- 36 or 64 (2 implementations) ϕ sectors. Processed processed by independent HT
- Currently, each MP7 processes all (or many) ϕ sectors within a single η sector.
- First tracks showing up in hardware. ~ agree with simulation

Tracklet based approach







Seeding:
- Form tracklets from pairs of stub in adjacent layers
- Use beamspot constraints
- Tracklet must be consistent with Pt and z0 requirements

Projecting:
- Project to other layers and disks
- search window derived from residuals b/w projected tracks and stubs
- In-out & Out-in

Fitting
- linearized track fit

Duplicate Removal:
Based on number of shared stubs