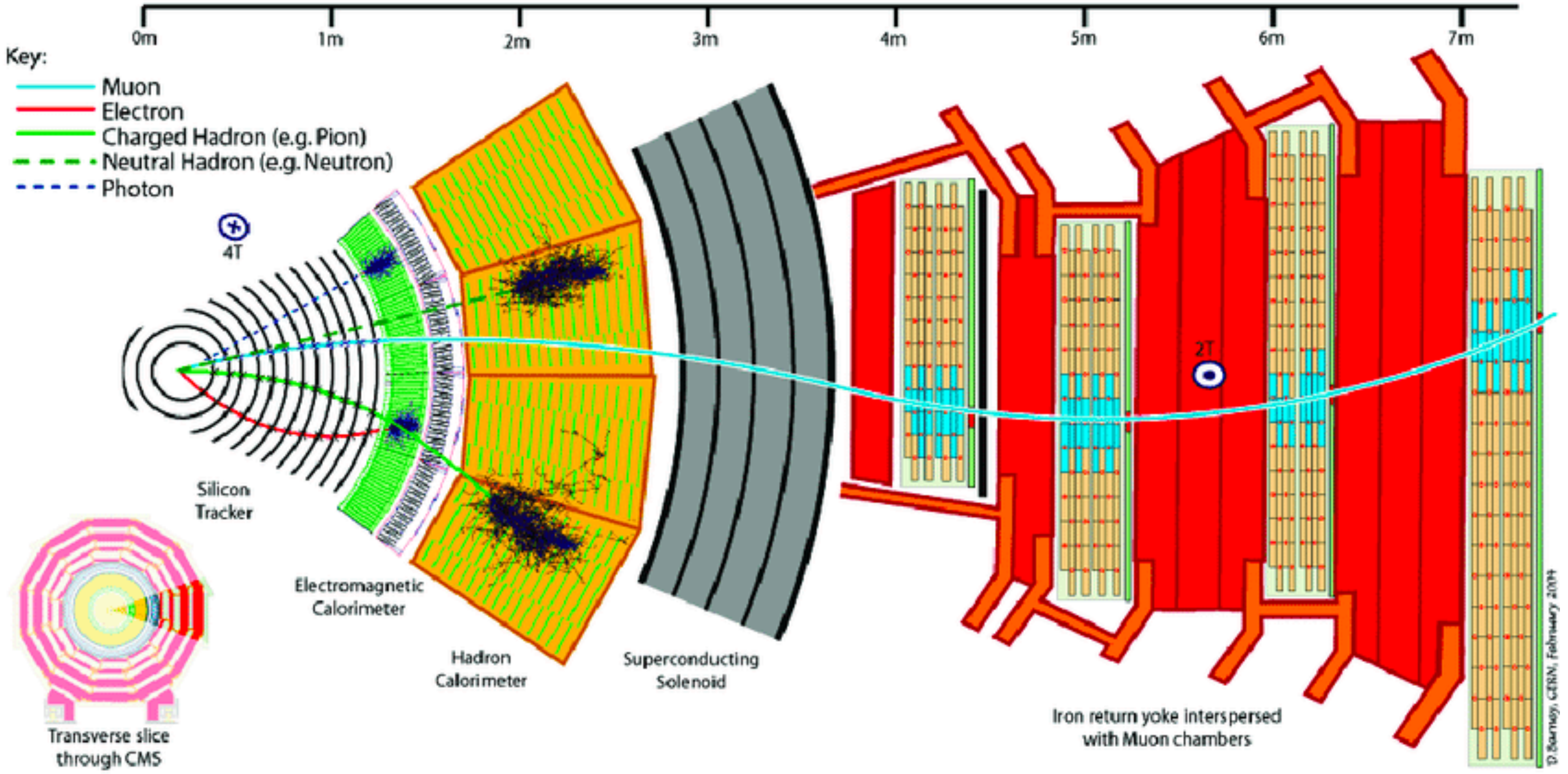
The Particle Flow Algorithm in the Phase II Upgrade of the CMS Level-1 Trigger

Sioni Summers for the CMS Collaboration

8th May 2023
26th International Conference on Computing in High Energy & Nuclear Physics

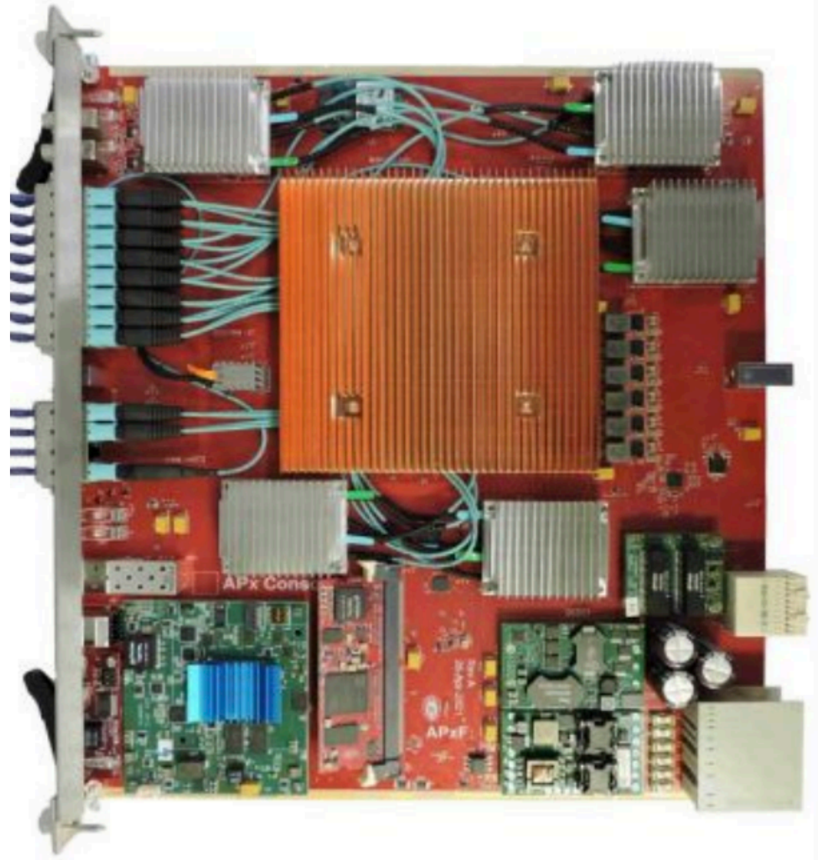
Introduction

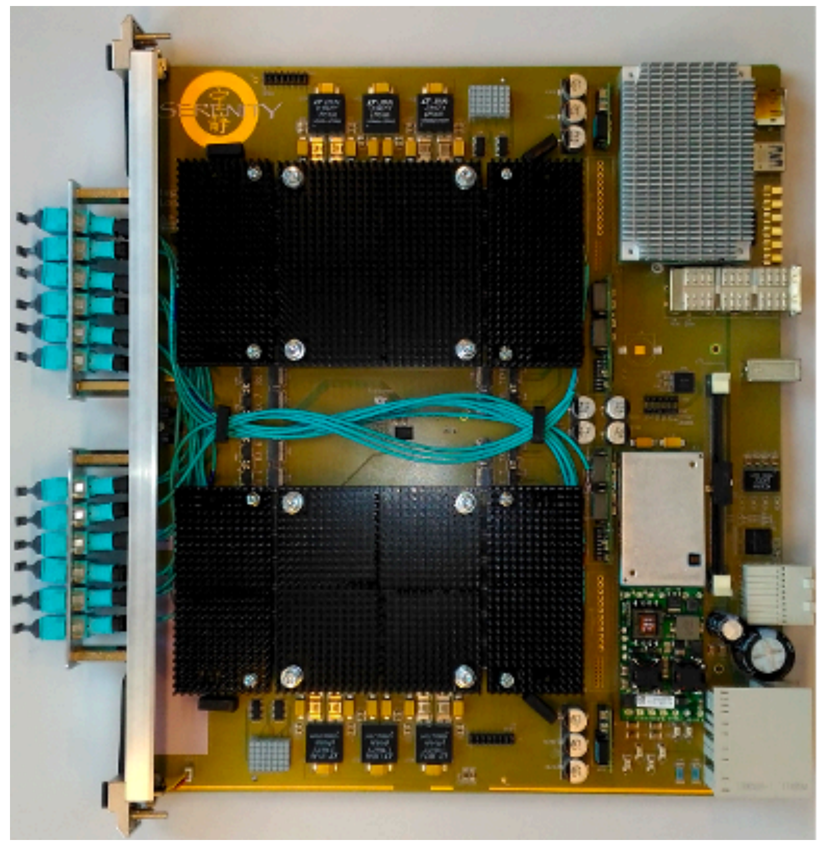
- CMS uses Particle Flow (PF) algorithm offline to best combine measurements from different sub-detectors to reconstruct final state particles
- Phase II Upgrade of CMS will equip Level 1 Trigger with significantly more precise inputs
  - Tracks of charged particles with p_T above 2 GeV
  - High Granularity Calorimeter in endcaps
  - More precision from barrel calorimeter to L1T
  - Muon systems
- Particle Flow will be brought to L1T for the first time
- Pile-Up Per Particle Identification (PUPPI) for mitigation of extremely high pileup environment



CMS Phase 2 Level 1 Trigger

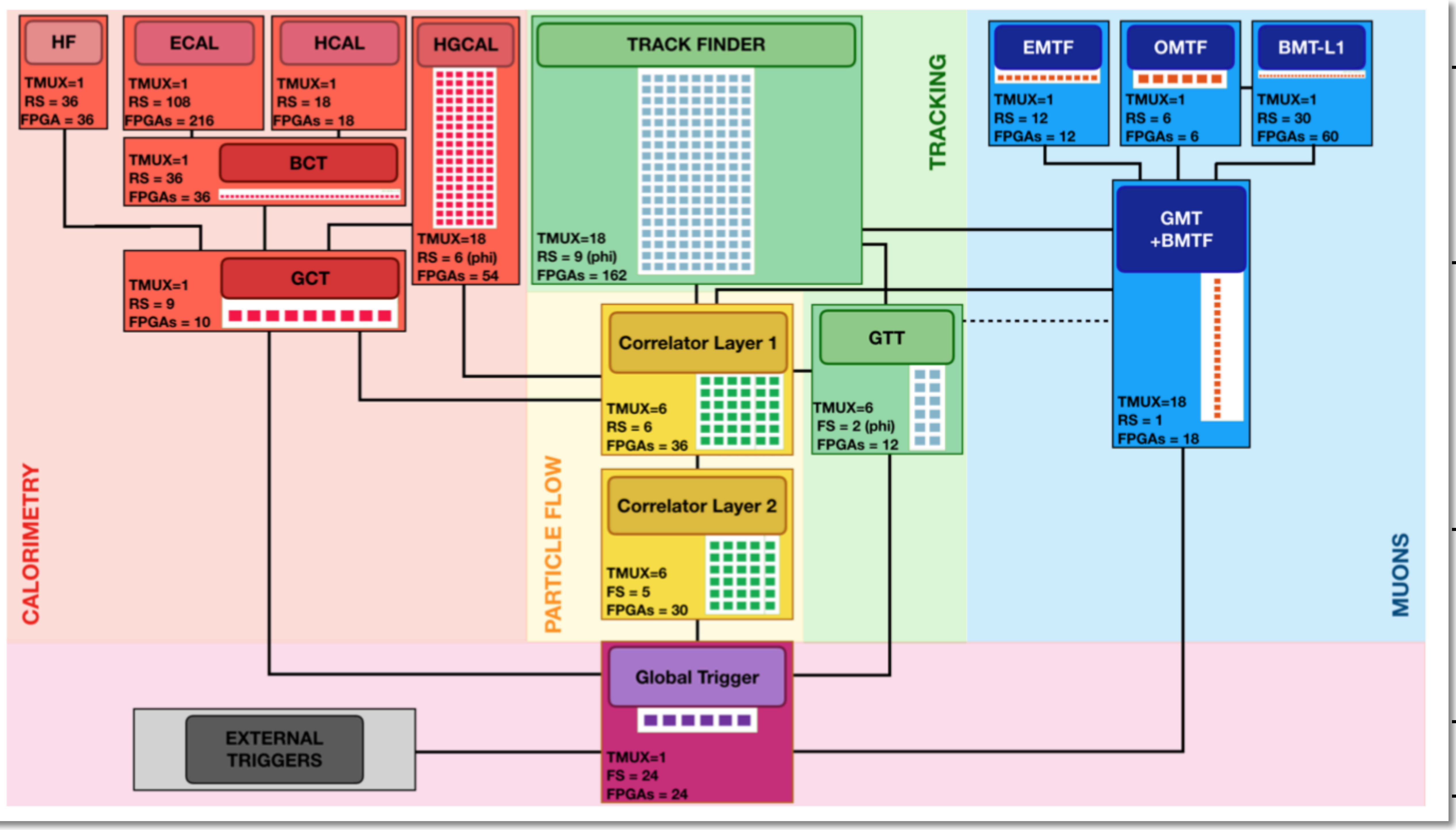
- System will comprise 100s of FPGA boards reconstructing and reducing data in stages
- Correlator Layer 1 will use Serenity and APx ATCA boards with Xilinx Ultrascale+ FPGAs and multi-Tb/s data IO over optical fibres





Each small box is one Xilinx Ultrascale+ FPGA

t = 0



t ≤ 12.5 μs

Detector hits

Clusters & Tracks

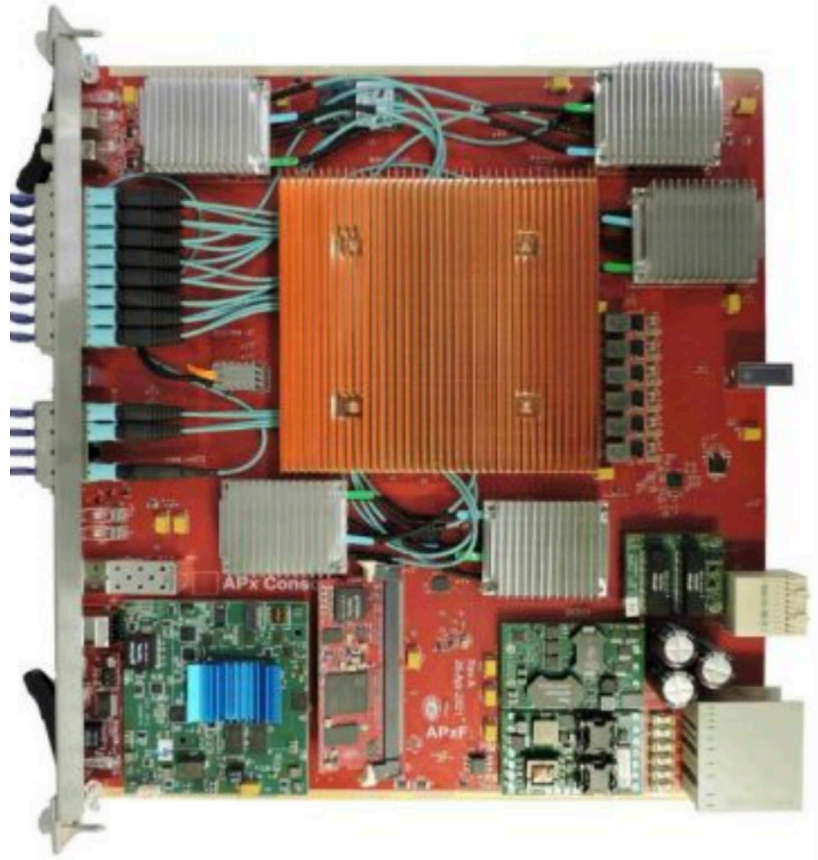
Particles

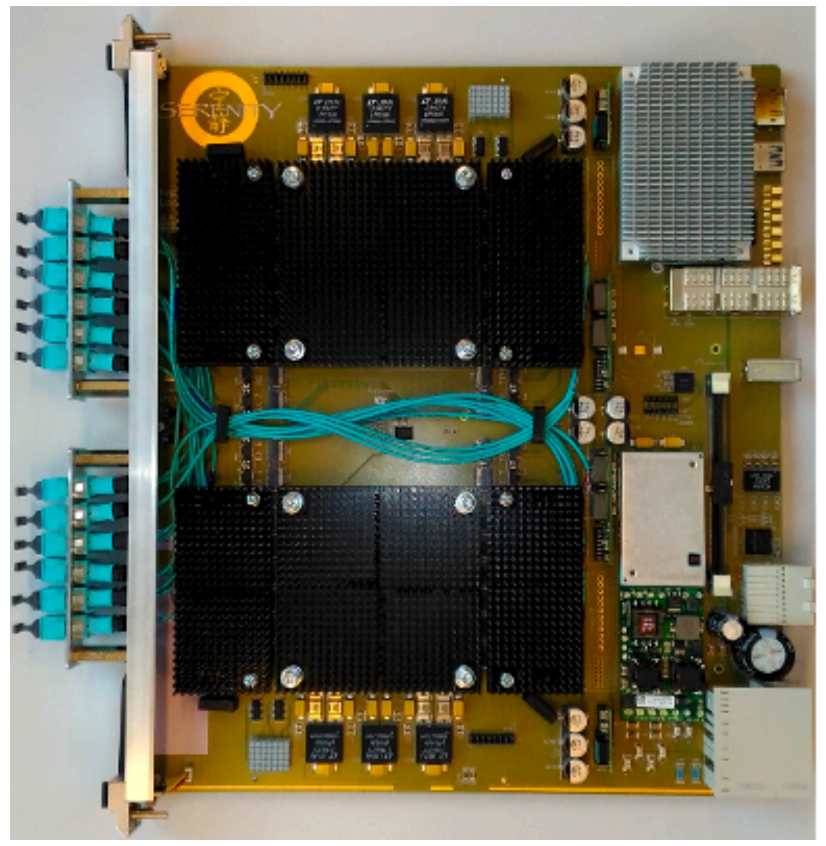
Event Categorisation

1 bit: keep / discard

CMS Phase 2 Level 1 Trigger

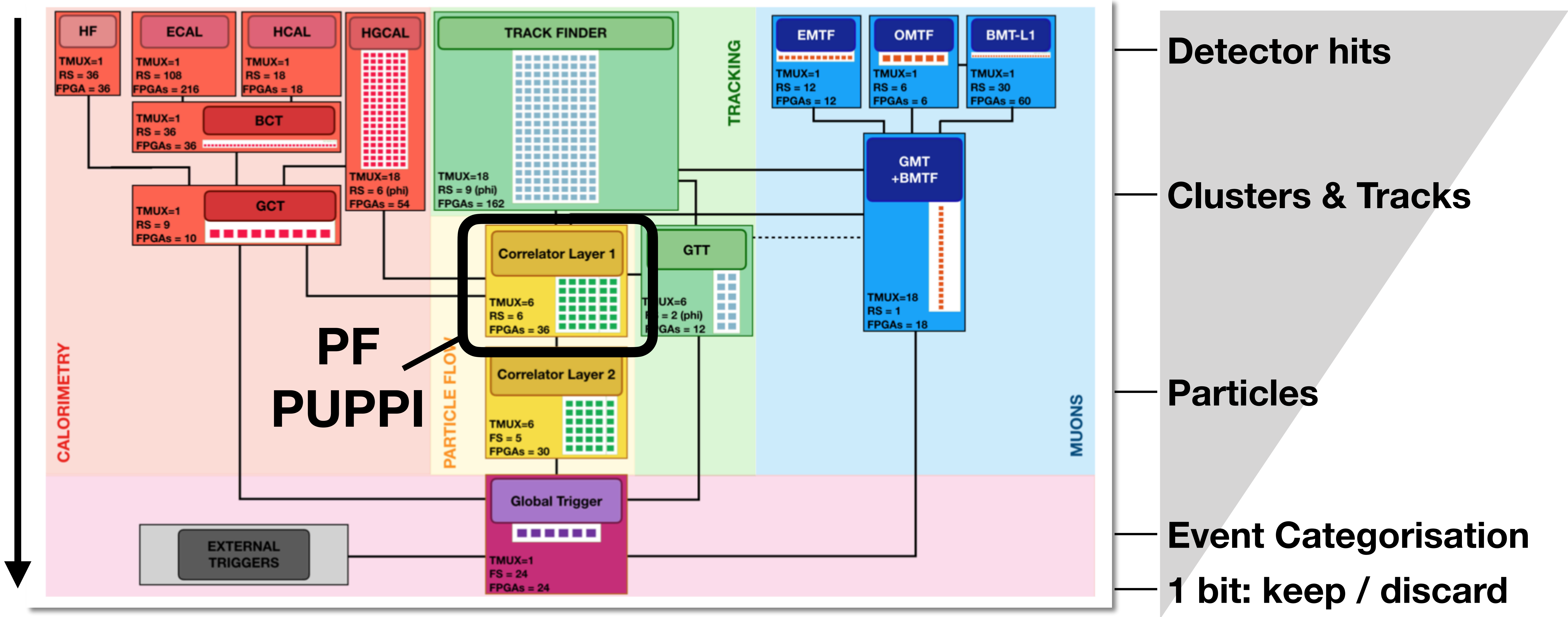
- System will comprise 100s of FPGA boards reconstructing and reducing data in stages
- Correlator Layer 1 will use Serenity and APx ATCA boards with Xilinx Ultrascale+ FPGAs and multi-Tb/s data IO over optical fibres





Each small box is one Xilinx Ultrascale+ FPGA

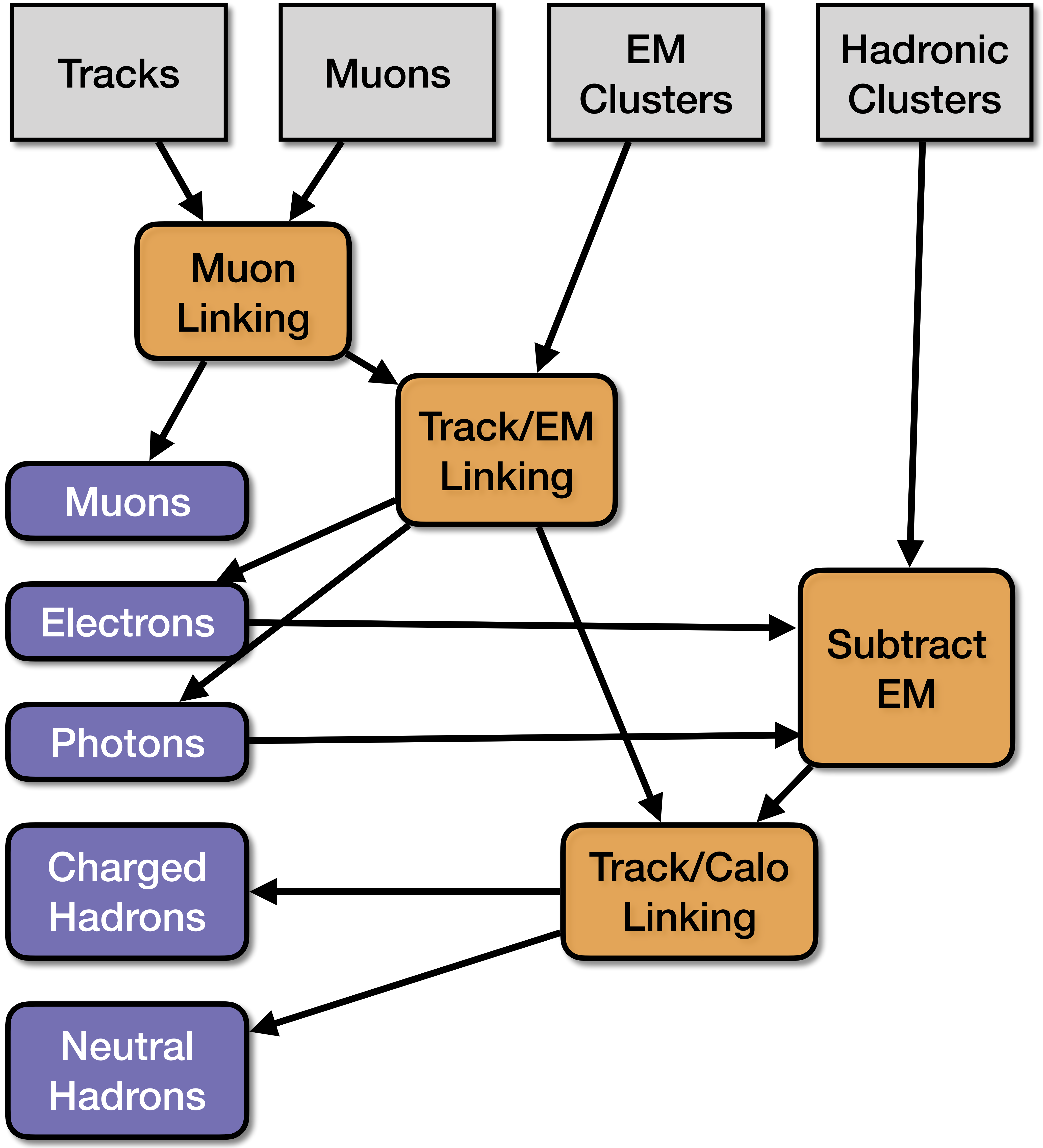
t = 0



t ≤ 12.5 μs

L1T Particle Flow Algorithm

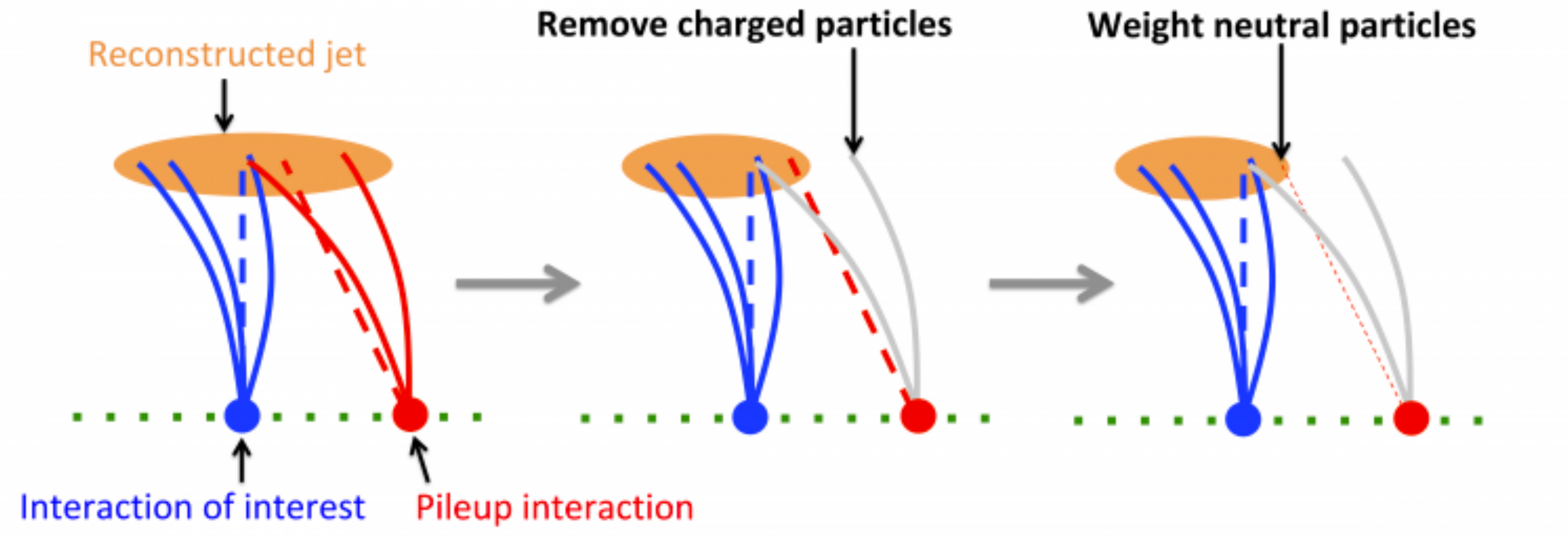
- Link elements from different sub-detectors to reconstruct final state particles
  - Relies on precise reconstruction of charged particle tracks
- Match tracks to muons by ΔR in (η, ϕ) and p_T
  - Exclude matched tracks from next steps
- Match tracks to electromagnetic clusters by ΔR, sum p_T of tracks around cluster
  - Classify as photon, electron, or hadron (discard)
- Match electromagnetic clusters to hadronic clusters by ΔR
  - Subtract clustered EM energy from hadronic energy
- Match tracks to hadronic clusters by ΔR and p_T
  - Classify as charged or neutral hadron
- Produces: muons, electrons, photons, charged and neutral hadrons



L1T PUPPI Algorithm

- Particle Flow reconstructs all particles, including from pileup interactions
- Pile Up Per Particle Identification (PUPPI) isolates the particles originating from leading interaction
  - Using Primary Vertex Finding and particle weighting
  - Vertexing performed externally to Correlator (need all tracks in one board), sent over optical fibre
- Charged particles selected if track z coordinate compatible with found vertex
- Neutral particles are weighted according to local neighbourhood of primary-vertex-associated charged particles
  - Weight is a function of $\Sigma\, p_T^2 / \Delta R^2$ over selected particles within $\Delta R < 0.4$
  - Outside tracker, all particles are used

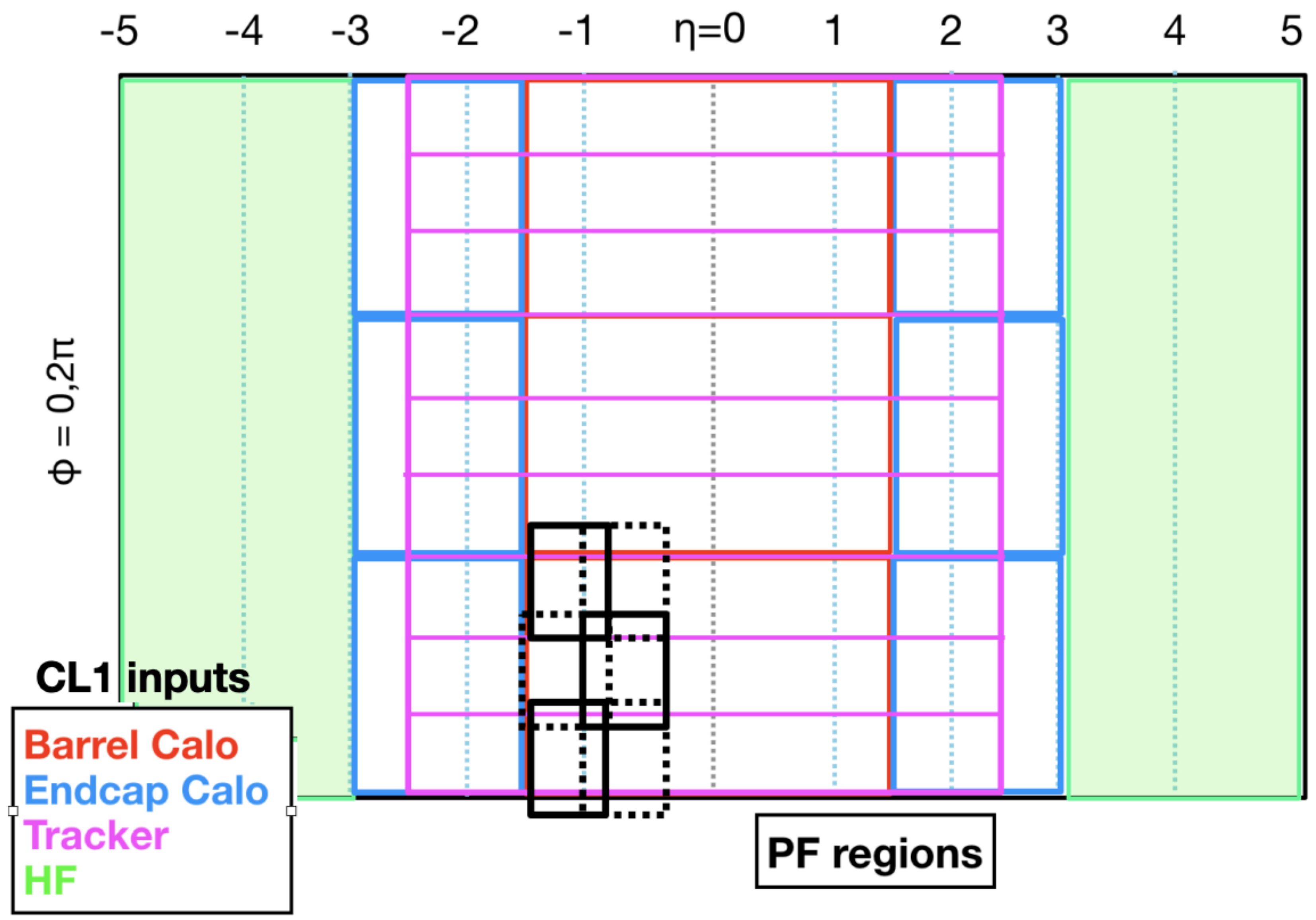
Particle Flow candidates

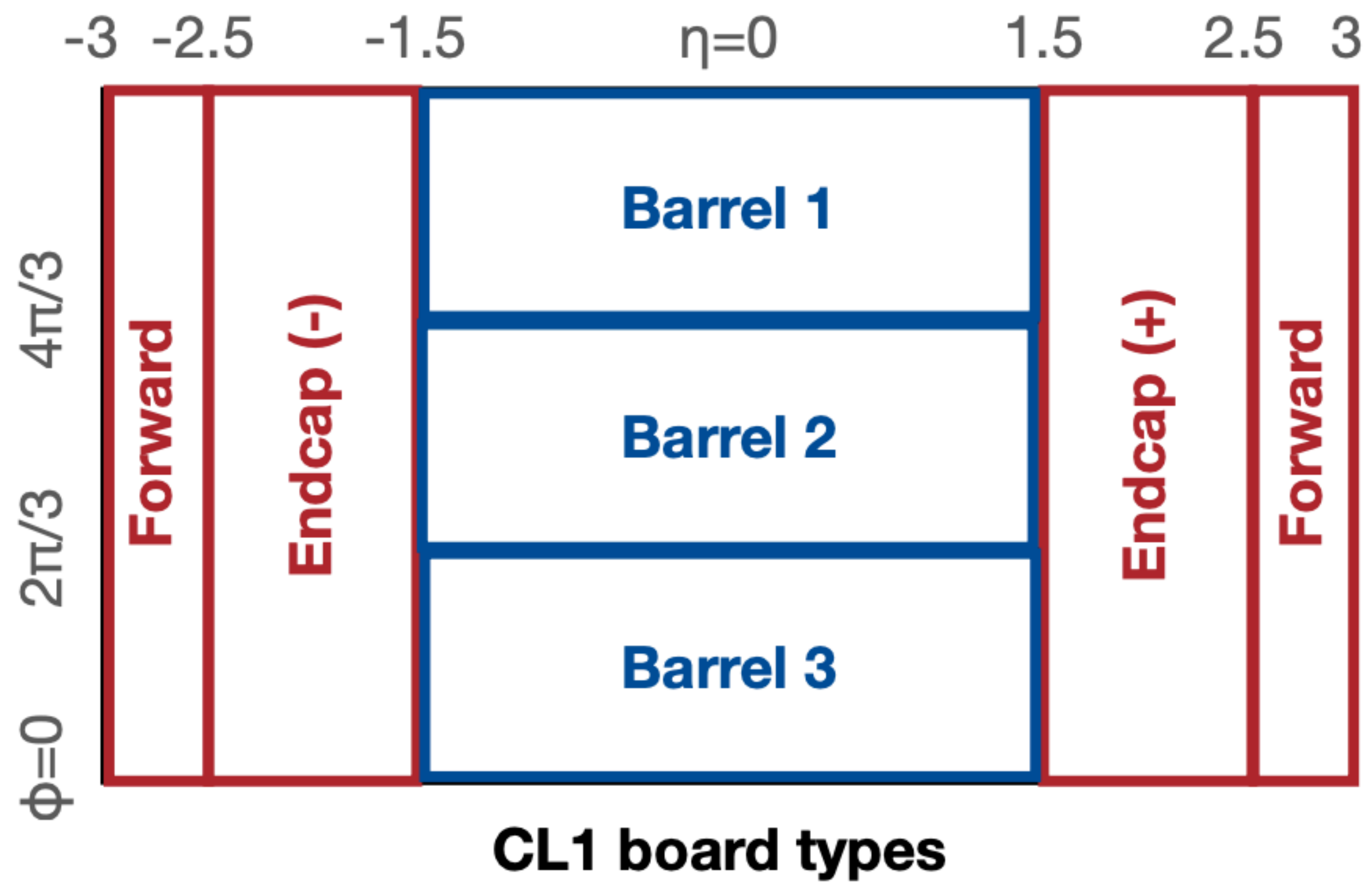


PUPPI candidates

Regionizer

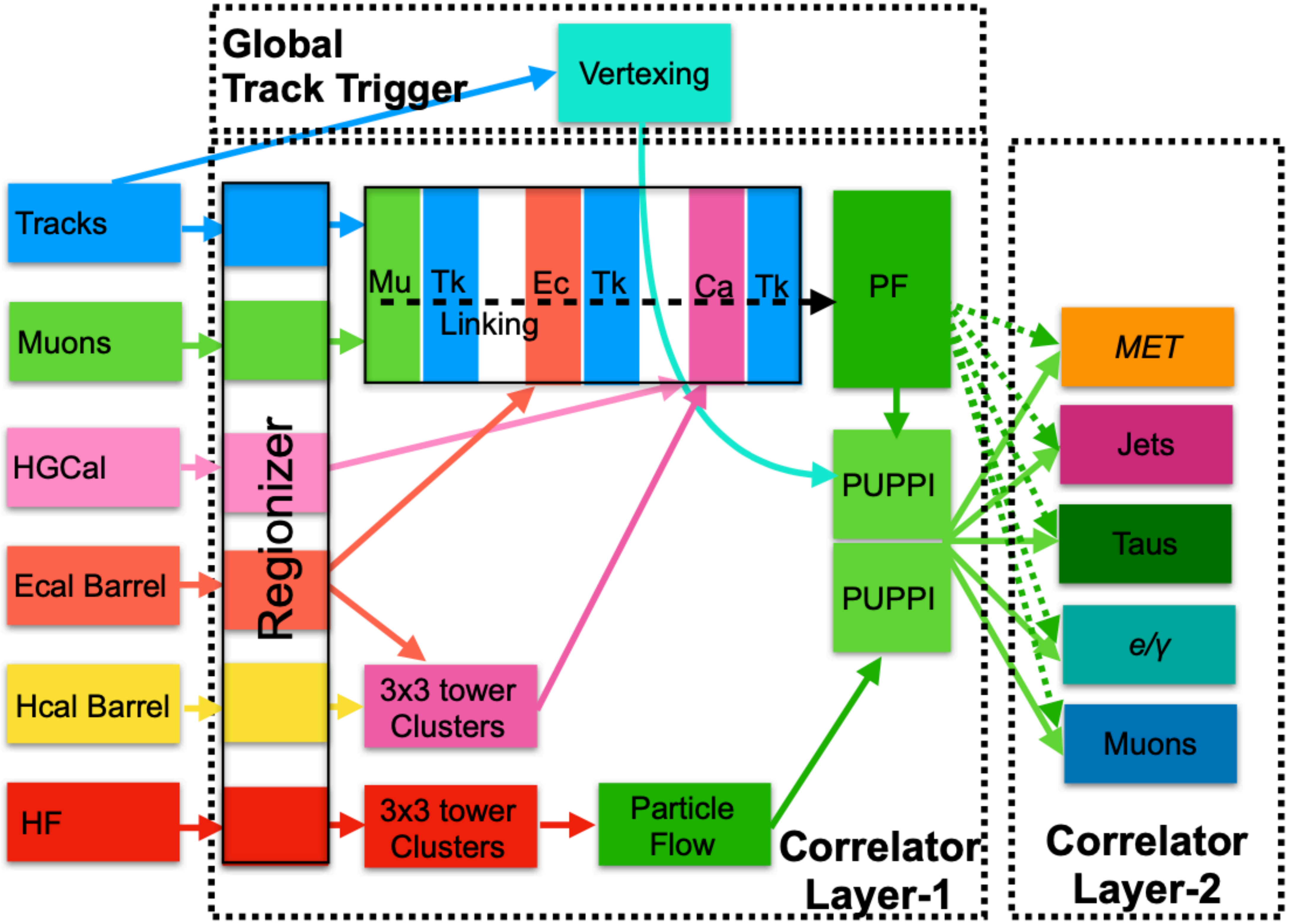
- 6 FPGA boards per event split detector into macro-regions in (η, ϕ)
  - Barrel in 3 slices, Endcap in 2 slices, Forward in 1 board (2 slices)
  - Constrained by number of input fibres into each board
- To break the processing down further into manageable pieces, PF is performed regionally in (η, ϕ)
  - Regions sized roughly 0.5 x 0.5, chosen such that multiplicity of inputs is reasonable (around 20-30)
  - However, inputs from upstream systems do not arrive conveniently for PF regions
- Regionizer modules aggregate and shuffle around inputs objects into memories (Block RAMs) for later consumption by PF algorithm
- Significantly challenging part of the design
  - Need to maintain high internal data throughput to keep overall latency low
- Implemented in VHDL, using many small memories (BRAM, URAM)





Particle Flow & PUPPI Implementation

- Within the PF Region objects are linked to tracks sequentially
  - Bit-vectors propagate linking information through algorithm
  - Tracks to muons, electromagnetic clusters to tracks, hadronic clusters to tracks
- PF implemented using Xilinx High Level Synthesis (HLS)
  - FPGA optimized C++ with annotations for guided synthesis
- Resource parallelism: compute all ΔR matches simultaneously
  - The major resource consumer of the module
- Pipeline parallelism: start processing next region `i+1` before finishing region `i`
- Regions are processed serially
  - Up to 18 regions per board - depending on the macro region
  - Particles found in region are sorted by p_T before output
- PUPPI particles are sent on 25 Gb/s optical fibres to Layer 2 (5 boards per event)
  - Jets (prev. talk), taus, e/γ, muons, MET

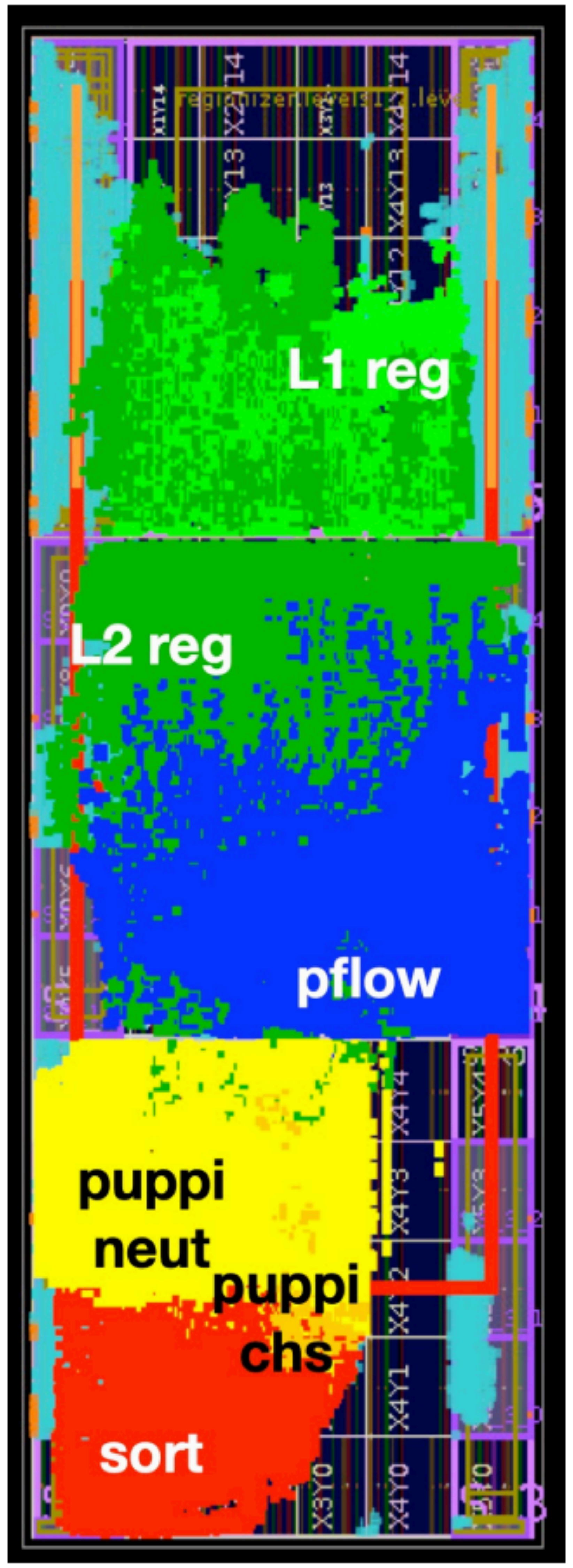


Floorplans, Resources

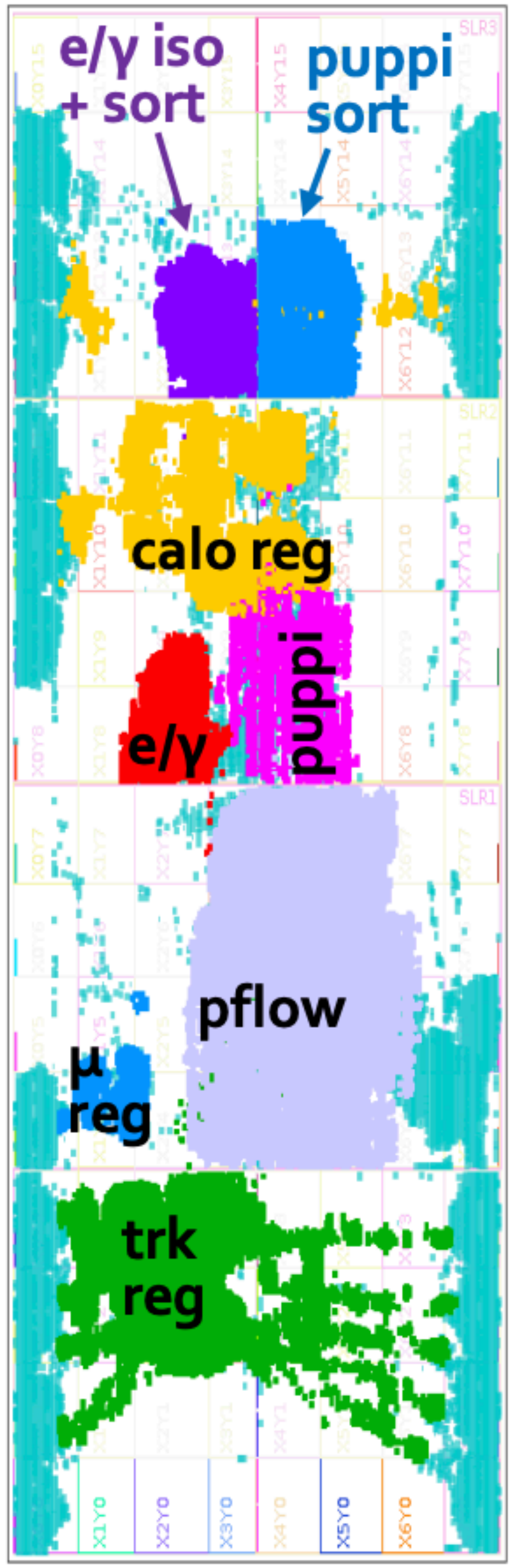
- All components deployed on prototype CMS Phase 2 L1T hardware
  - APx and Serenity ATCA cards with Xilinx Ultrascale+ FPGAs
- Tested loading data into input link buffer memories, ‘playing’ through algorithm and capturing at output memories
  - Compared against C++ software emulation: 100% agreement
- Tested using optical connections to neighbouring boards
  - Jets and electrons/photons in Layer 2, vertexing system

Resources	LUT	FF	BRAM	DSP
Barrel VU9P	36%	27%	15%	33%
Endcap VU13P	20%	16%	26%	14%
Forward VU13P	10%	8%	23%	2%

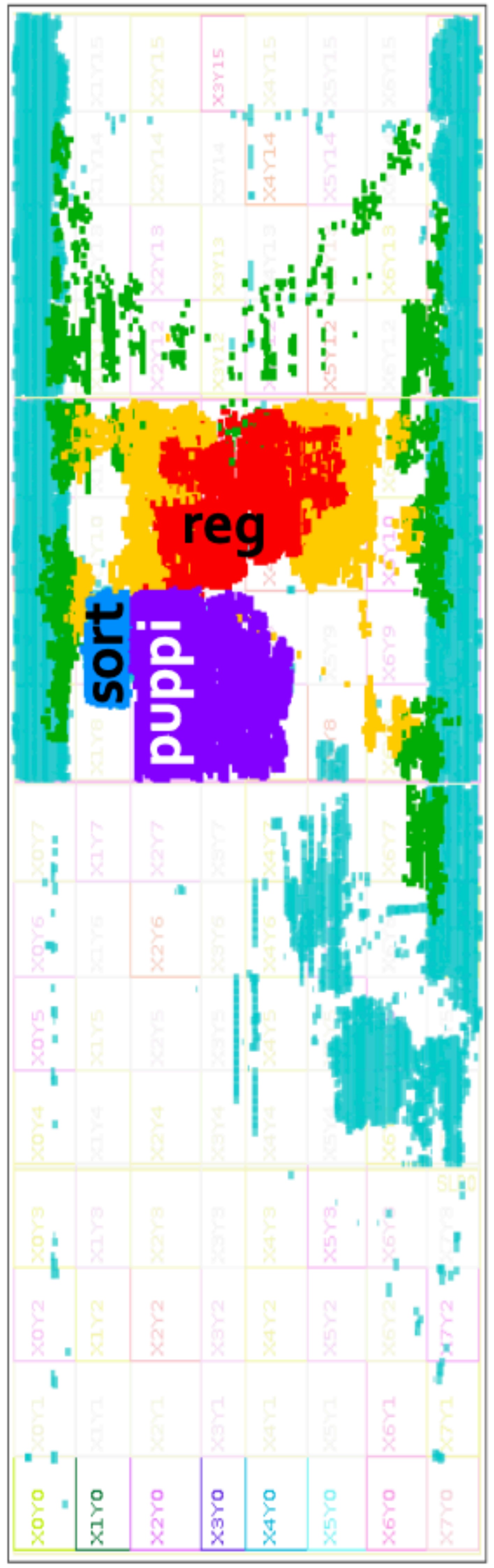
Barrel
APx VU9P



Endcap
Serenity VU13P

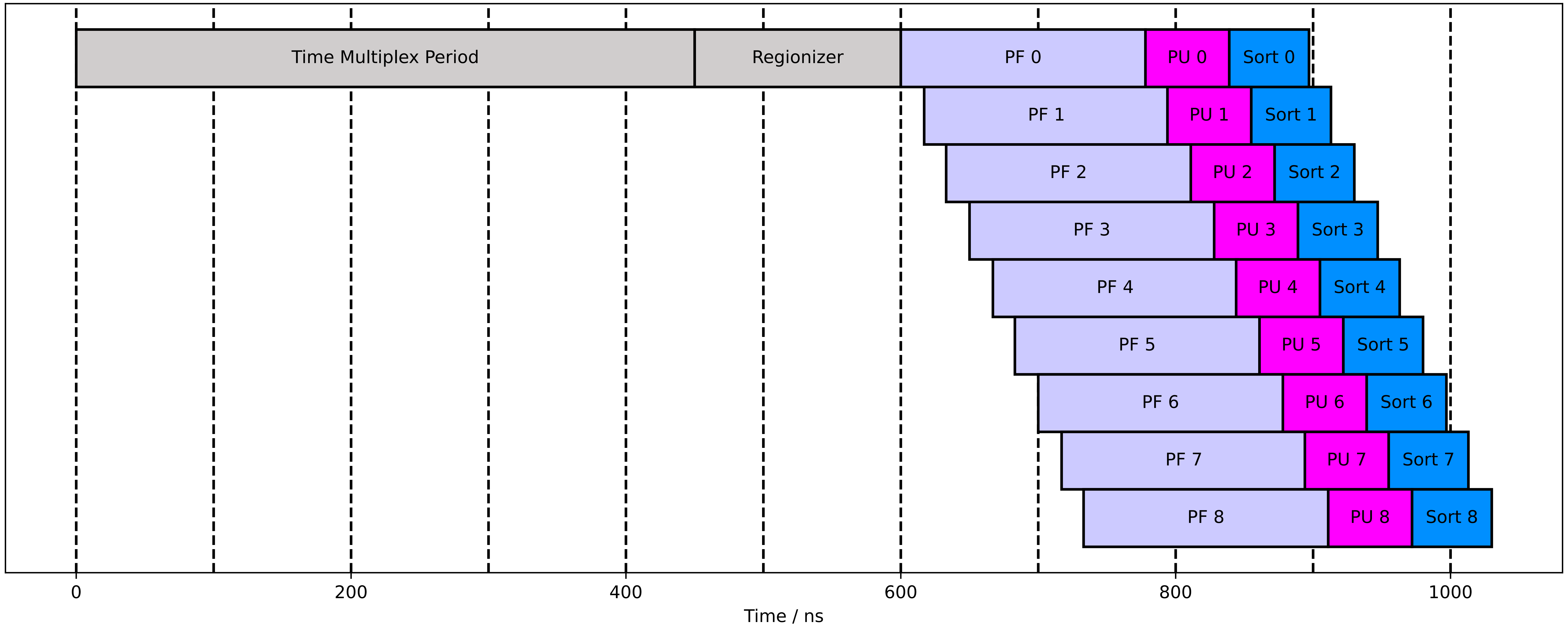


Forward
Serenity VU13P

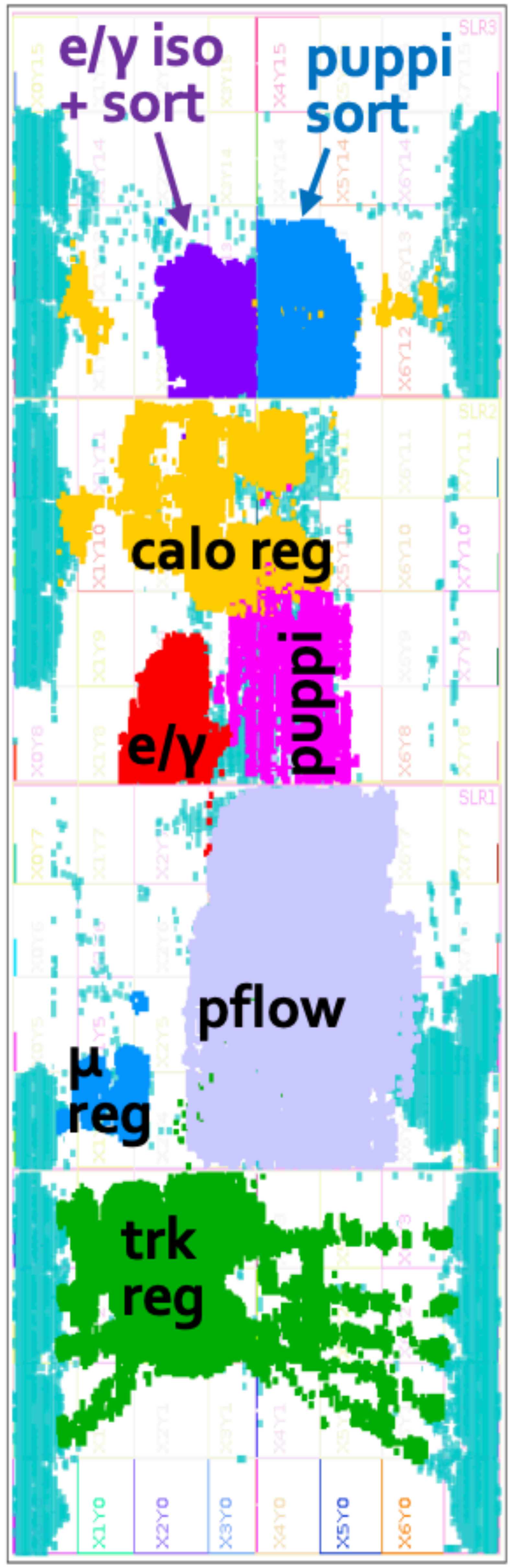


Latency

- Processing timeline for Correlator Layer 1 Endcap: 9 regions in (η, ϕ)
- Regionizer starts organising objects immediately at t=0
  - PF begins after Time Multiplex Period (all data arrival time) + Regionizer latency
- PF, PUPPI, and Sorter modules are pipelined: start processing iteration I+1 before finishing iteration I
- 1.03 μs First In to Last Out latency, processing up to 1 billion particles per second (real occupancy is lower)

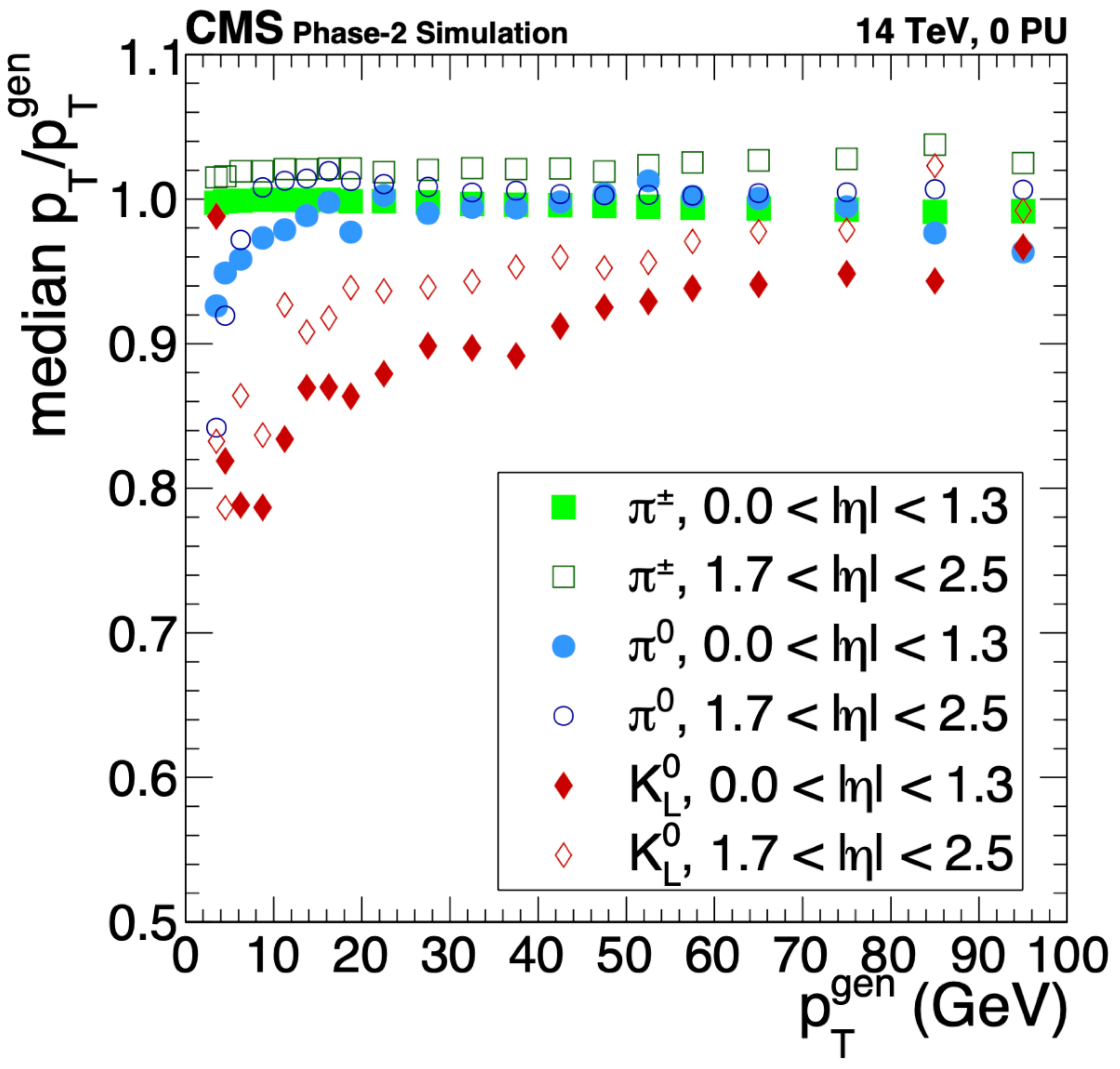


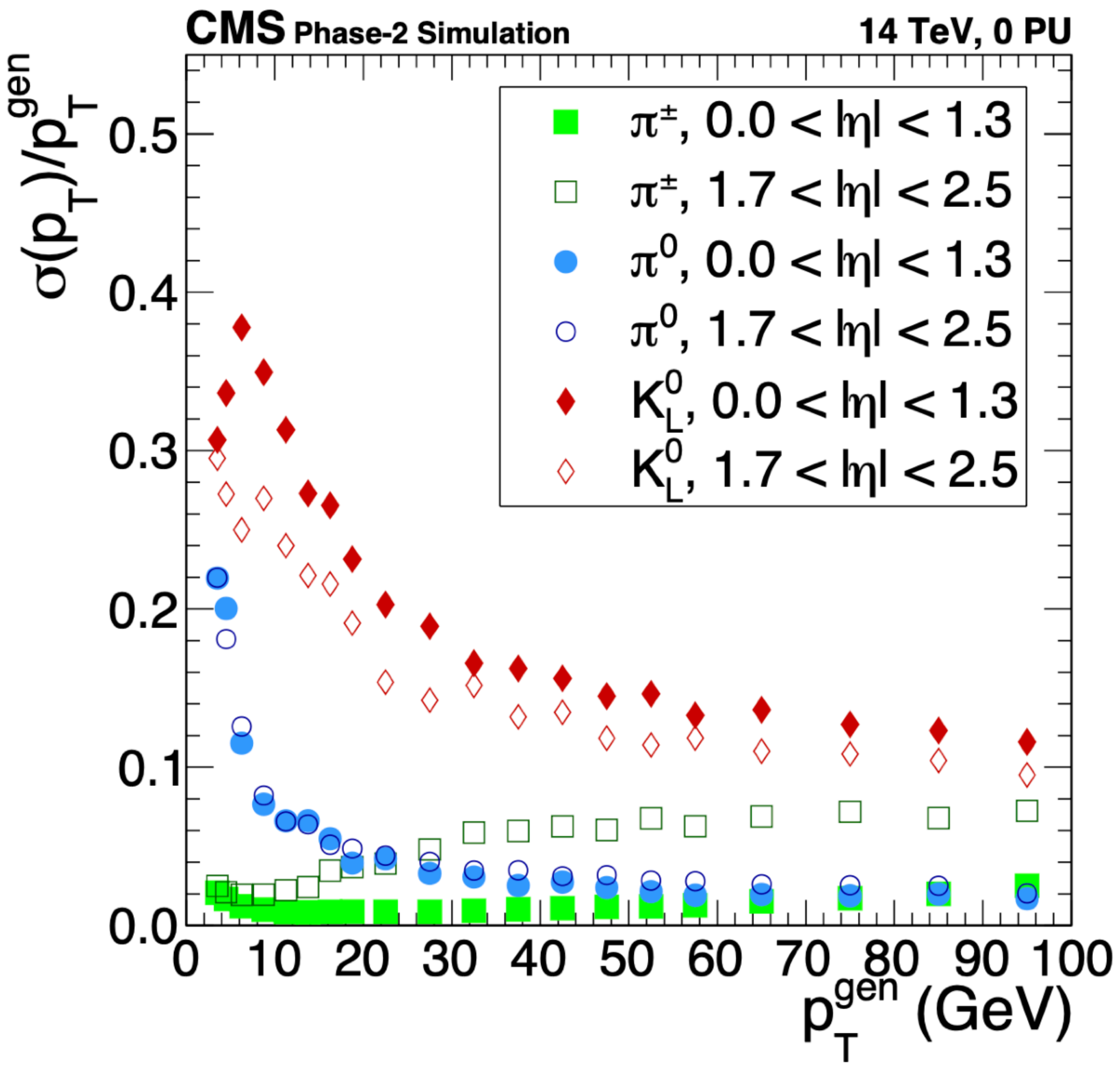
Endcap
Serenity VU13P

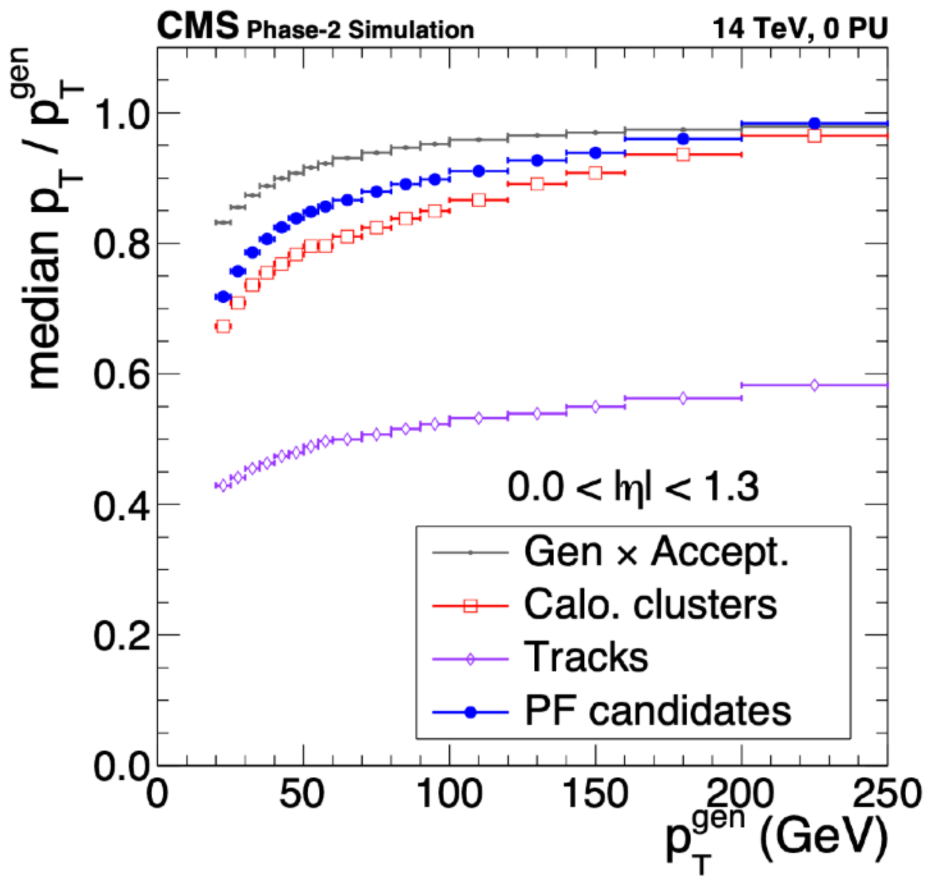


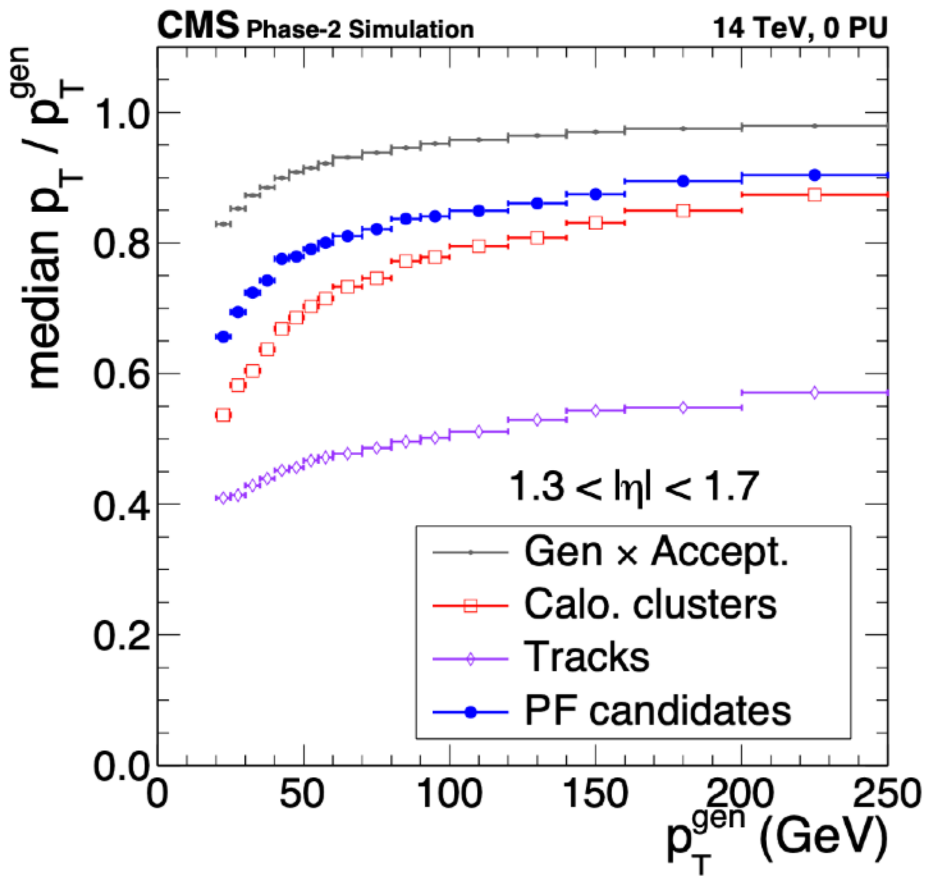
Physics Performance

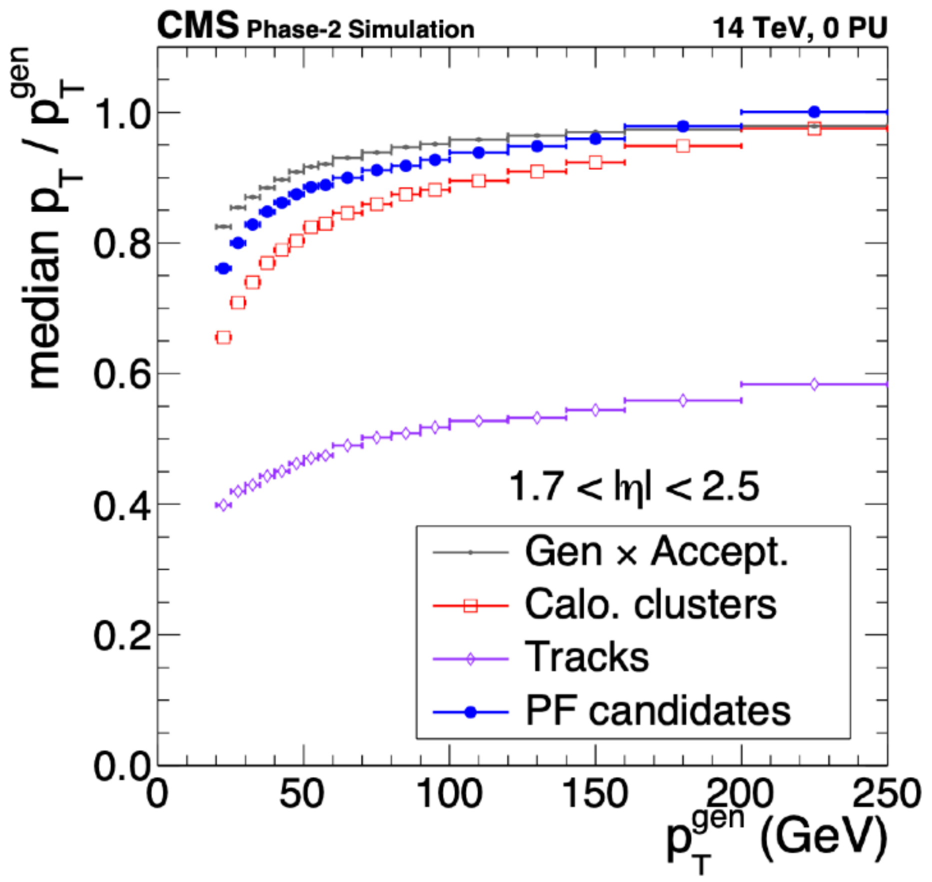
- Below: response (left) and resolution (right) of Particle Plow reconstructed charged pions, neutral pions, long-lived neutral kaons in different detector regions
- Right: response of jets reconstructed with either tracks alone, calorimeter clusters alone, or PF particles
  - Particle Flow achieves better performance than sub-detectors used individually











Conclusion

- For the Phase 2 Upgrade, CMS Level 1 Trigger will perform Particle Flow and PUPPI reconstruction for the first time
  - To best utilise new detectors: tracks above 2 GeV, High Granularity endcap calorimeter, increased barrel calorimeter precision
- PF & PUPPI bring improved performance to L1T over using individual sub-detectors in isolation
- The algorithms have been simplified and implemented in Xilinx Ultrascale+ FPGAs on prototype trigger boards
- Detector split into (η, ϕ) regions for parallel processing
  - Macro-regions processed simultaneously on different boards
  - PF regions process sequentially within each board with pipeline parallelism
- PUPPI particles are sent from Correlator Layer 1 to Correlator Layer 2 for reconstruction of high level objects such as jets, taus, MET, and isolation computations
- Processing latency around 1 μs, FPGA resources fitting within the constraints of the target devices

Backup

L1T PUPPI

$$\alpha_C = \log \sum_{i \in \mathrm{PV}, \Delta R < 0.3} \frac{\min({p_\mathrm{T}}^i, {p_\mathrm{T}}^{\max})^2}{\max(\Delta R, \Delta R^{\min})^2},$$

- Neutral particle p_T weighted as $p_T' = w_i\, p_T$
- Weight computed per particle as:

$$\begin{aligned} w_i &= \frac{1}{1 + e^{-x_\mathrm{tot}}} \\ x_\mathrm{tot} &= x_\alpha + x_{p_\mathrm{T}} - x_\mathrm{PU} \\ x_\alpha &= \min(\max(c_\alpha \cdot (\alpha - \alpha^0), -x_\alpha^{\max}), +x_\alpha^{\max}) \\ x_{p_\mathrm{T}} &= c_{p_\mathrm{T}} \cdot \left(p_\mathrm{T} - p_\mathrm{T}^0\right) \\ x_\mathrm{PU} &= \log\left(N_\mathrm{PU}/200\right) + c_0. \end{aligned}$$

- Constant parameters tuned to ensure good response and resolution for hadronic jets and MET