Implementation of the ATLAS trigger within the multi-threaded AthenaMT framework

ATLAS EXPERIMENT

Kondoli woven sculpture, SA Maritime Museum
Image via ABC

UNIVERSITÉ DE GENÈVE

Teng Jian Khoo
on behalf of the ATLAS Collaboration

FNSNF

OVERVIEW

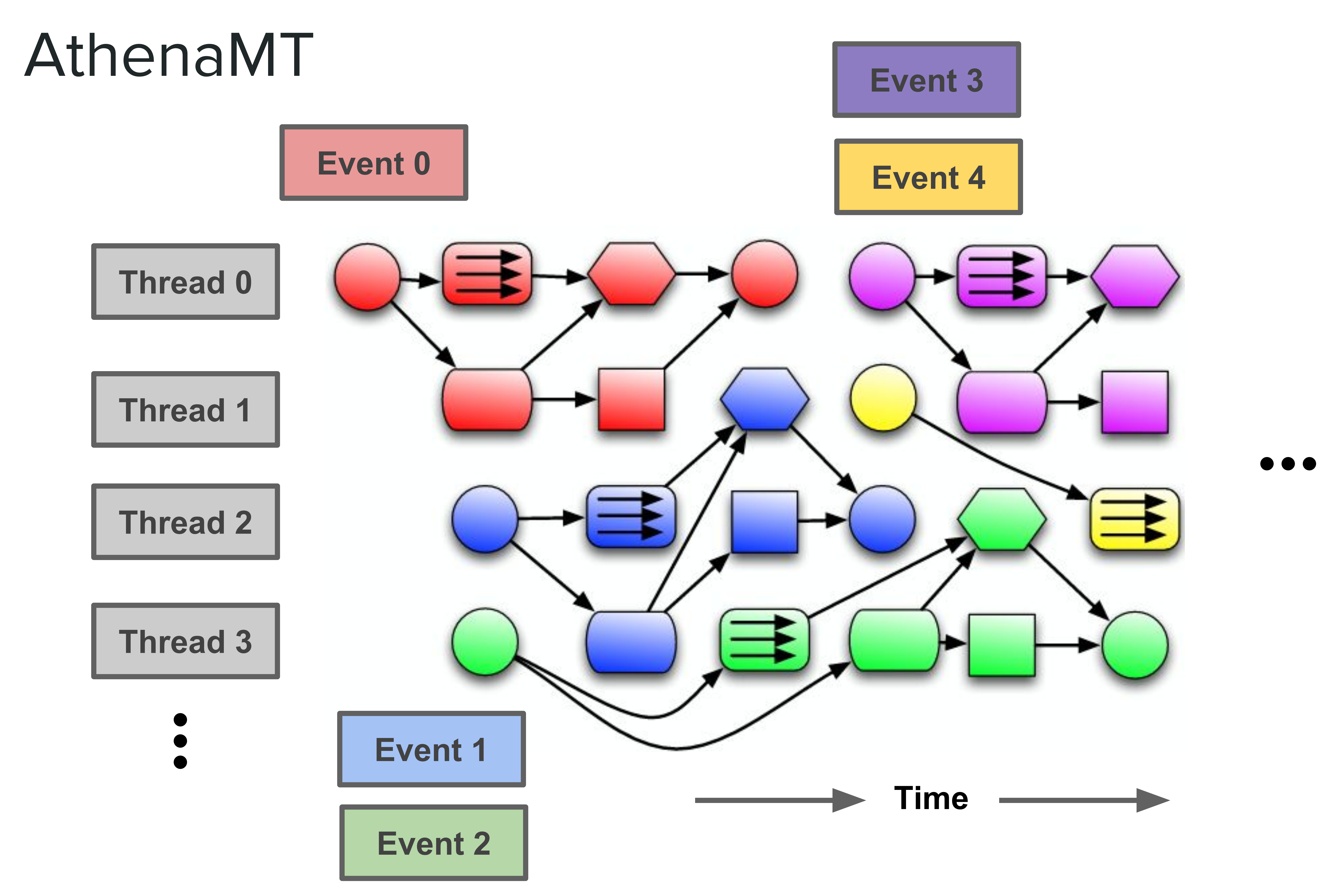
- Computing resources don't scale with challenge of growing LHC luminosity — SW evolution necessary
- Large project to implement GaudiHive & AthenaMT framework upgrades
  - Framework elements common with LHCb and other experiments [Gaudi webpage]
  - Multithreading uses Intel Threading Building Blocks (TBB)
- Here cover adaptation of ATLAS software trigger

WHY MULTITHREADING?

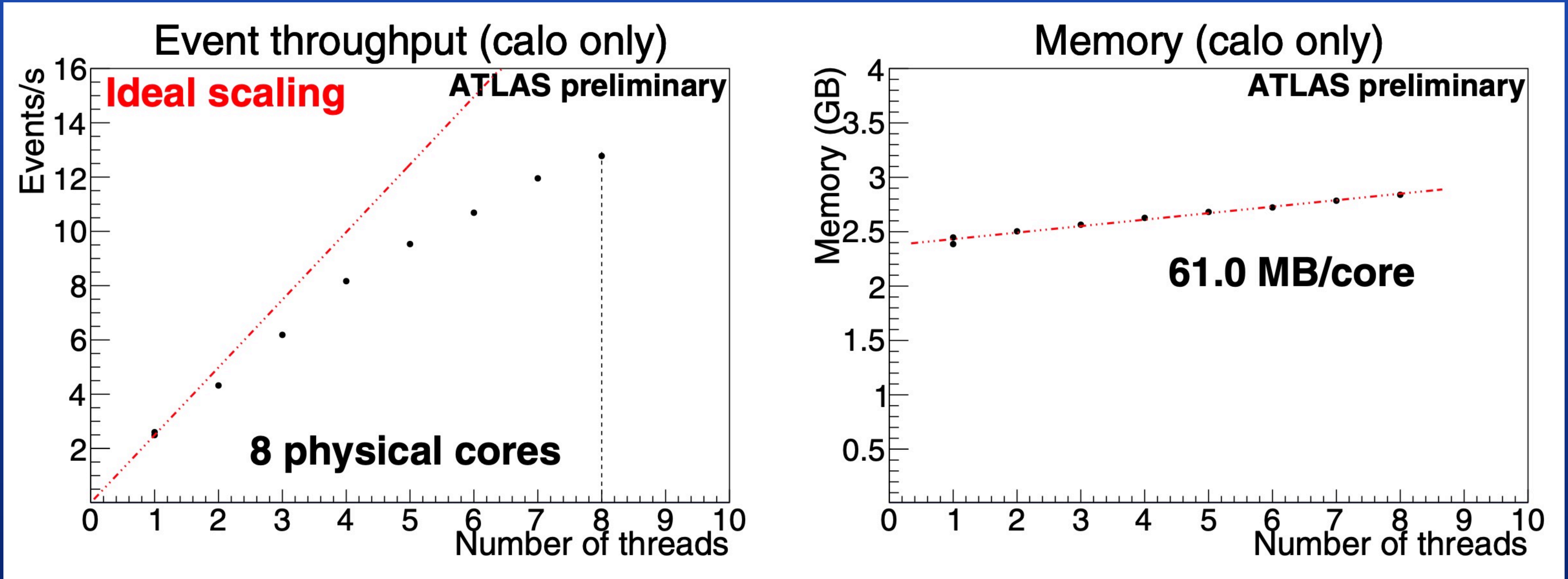
- “Embarrassingly parallel” no longer enough
  - Processor speed plateau —> growth of multicore
  - Memory price floor —> need for memory sharing
- Run 2 approach: multiprocess execution
  - Fork workers after initialisation (or first event)
  - Share large static data structures
  - Slow memory growth unavoidable
  - Intra-event parallelisation limited
- Pathway to future computing architectures
  - E.g. GPUs or FPGAs for hardware acceleration

PARALLELISM

Inter-event ✓
Intra-event ✓
Sub-algorithm **Possible**



PERFORMANCE SCALING

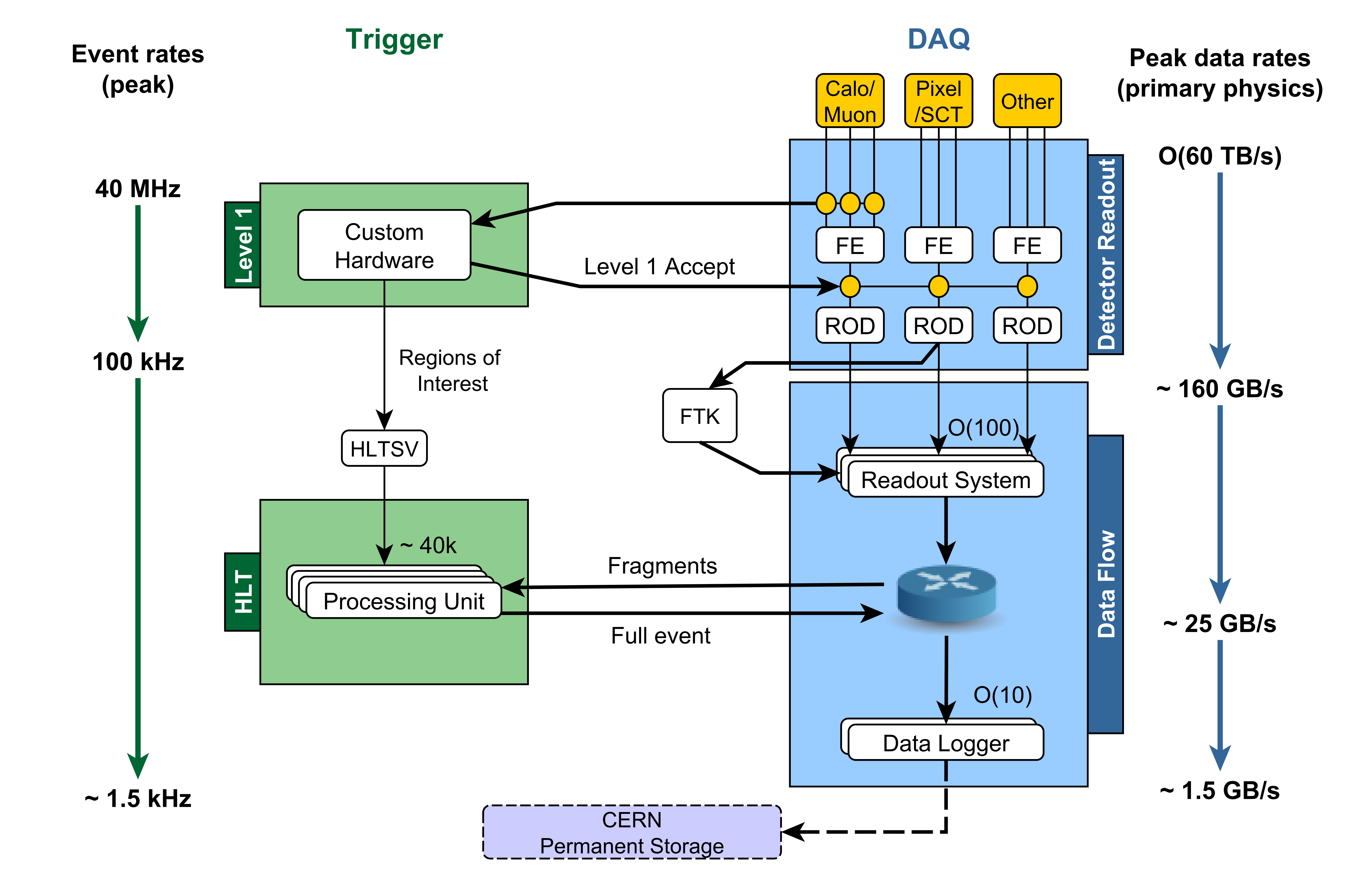


Close to ideal CPU scaling per core with minimal memory growth
Some bottlenecks still to be addressed

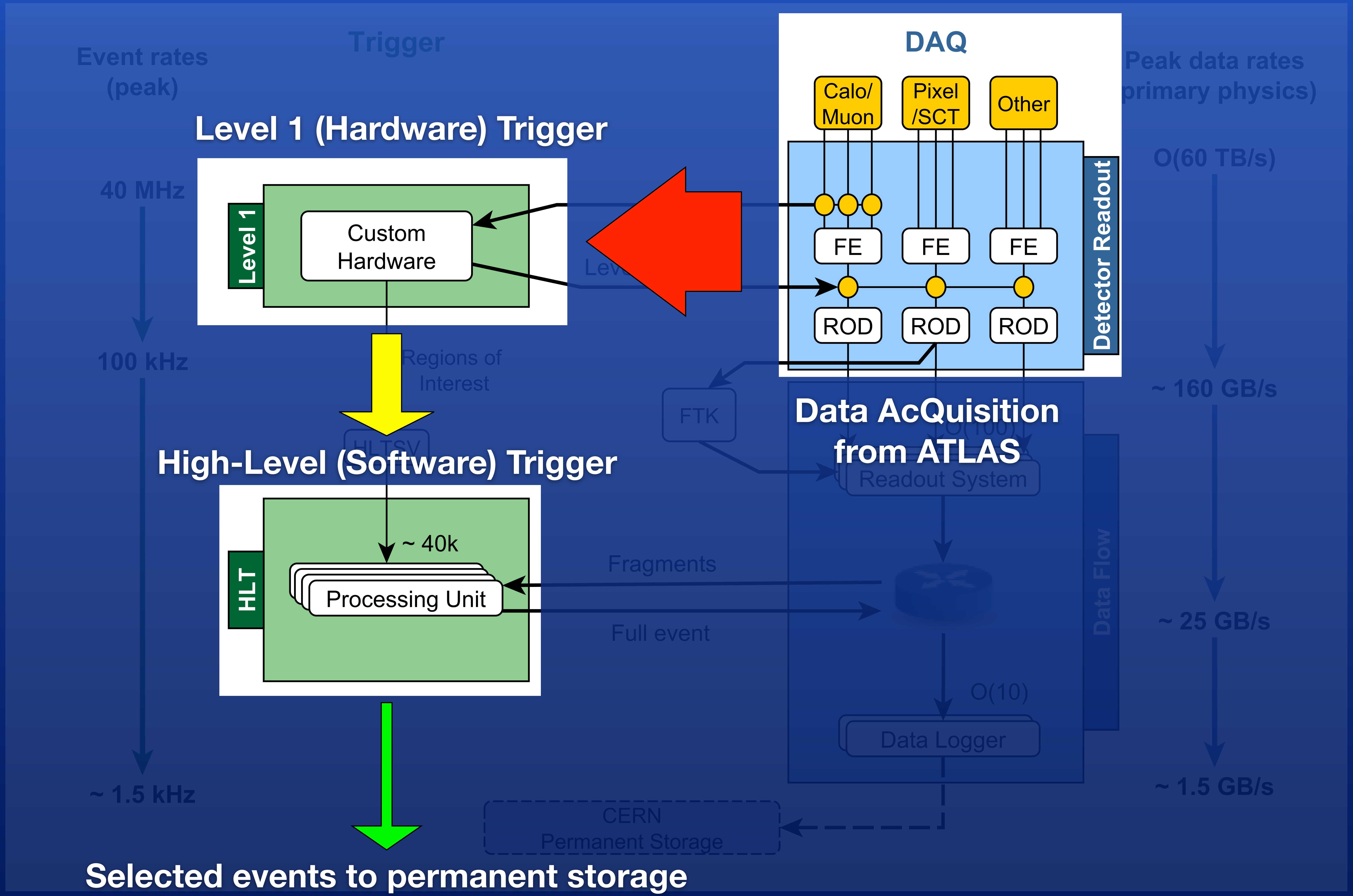
S Snyder et al 2019, EPJ Web Conf., 214 (2019) 05018

DISTILLING PHYSICS

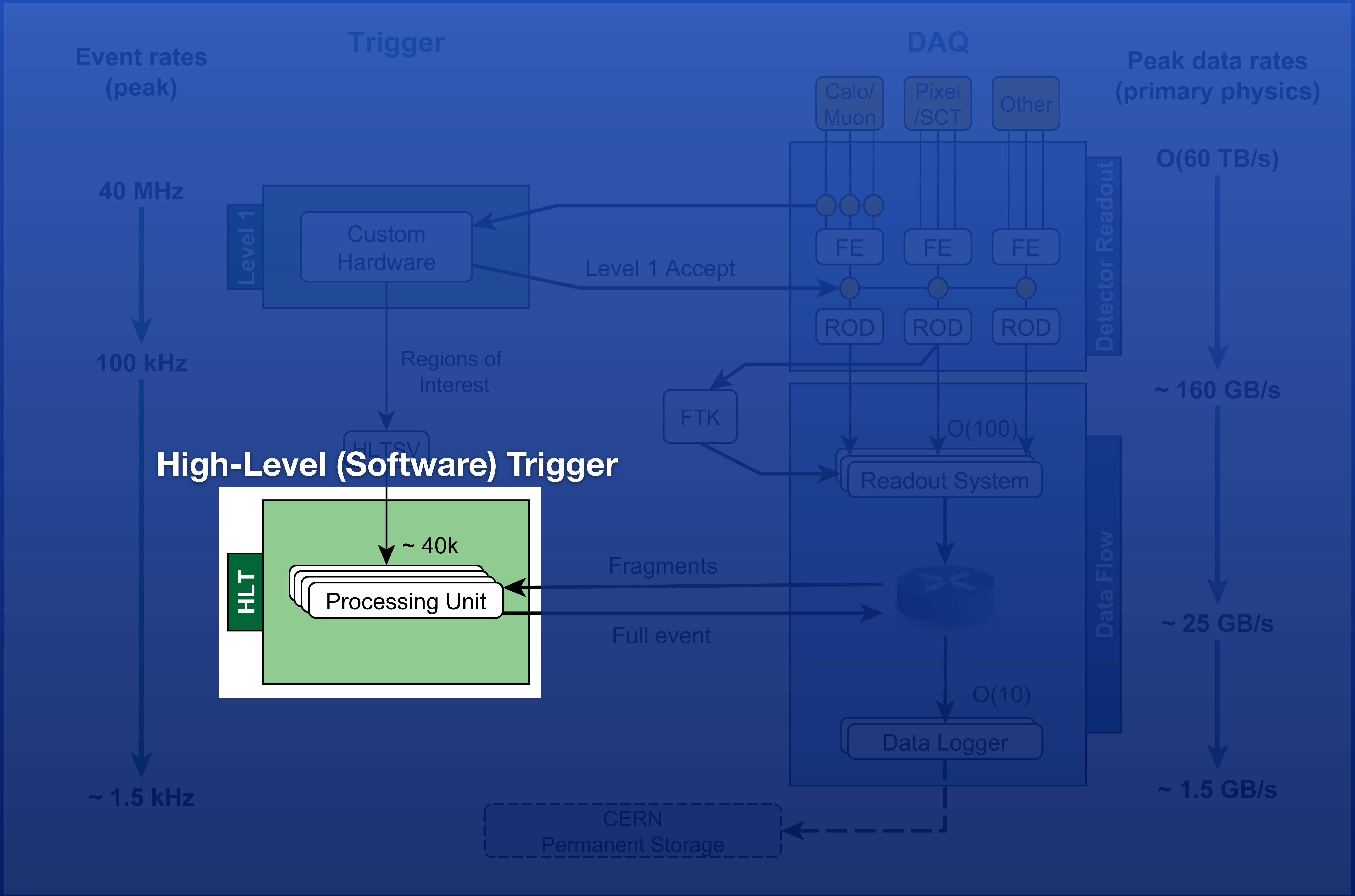
ATLAS TRIGGER ARCHITECTURE



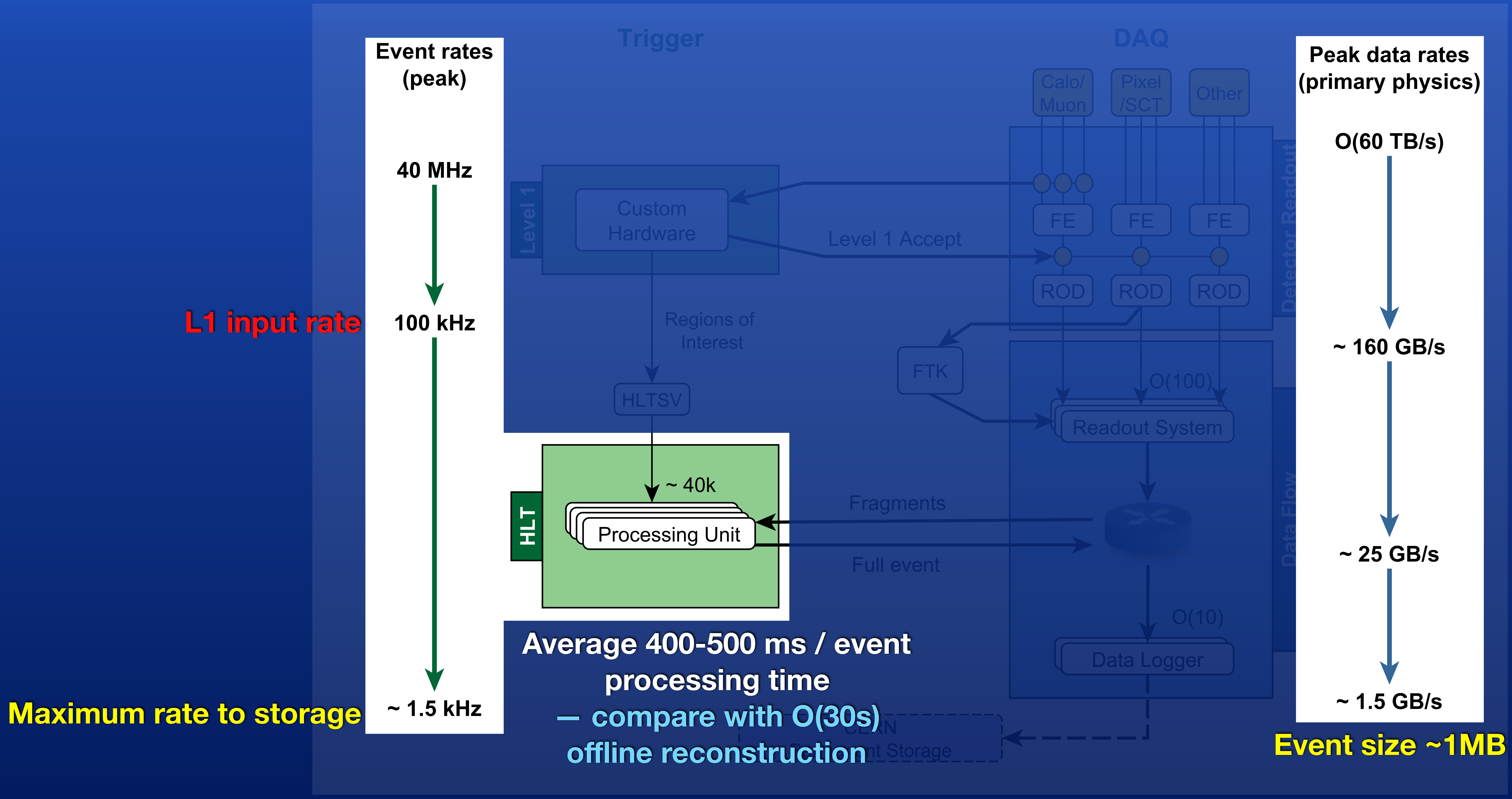
ATLAS TRIGGER ARCHITECTURE



TOPIC OF THIS TALK



HLT COMPUTING CONSTRAINTS



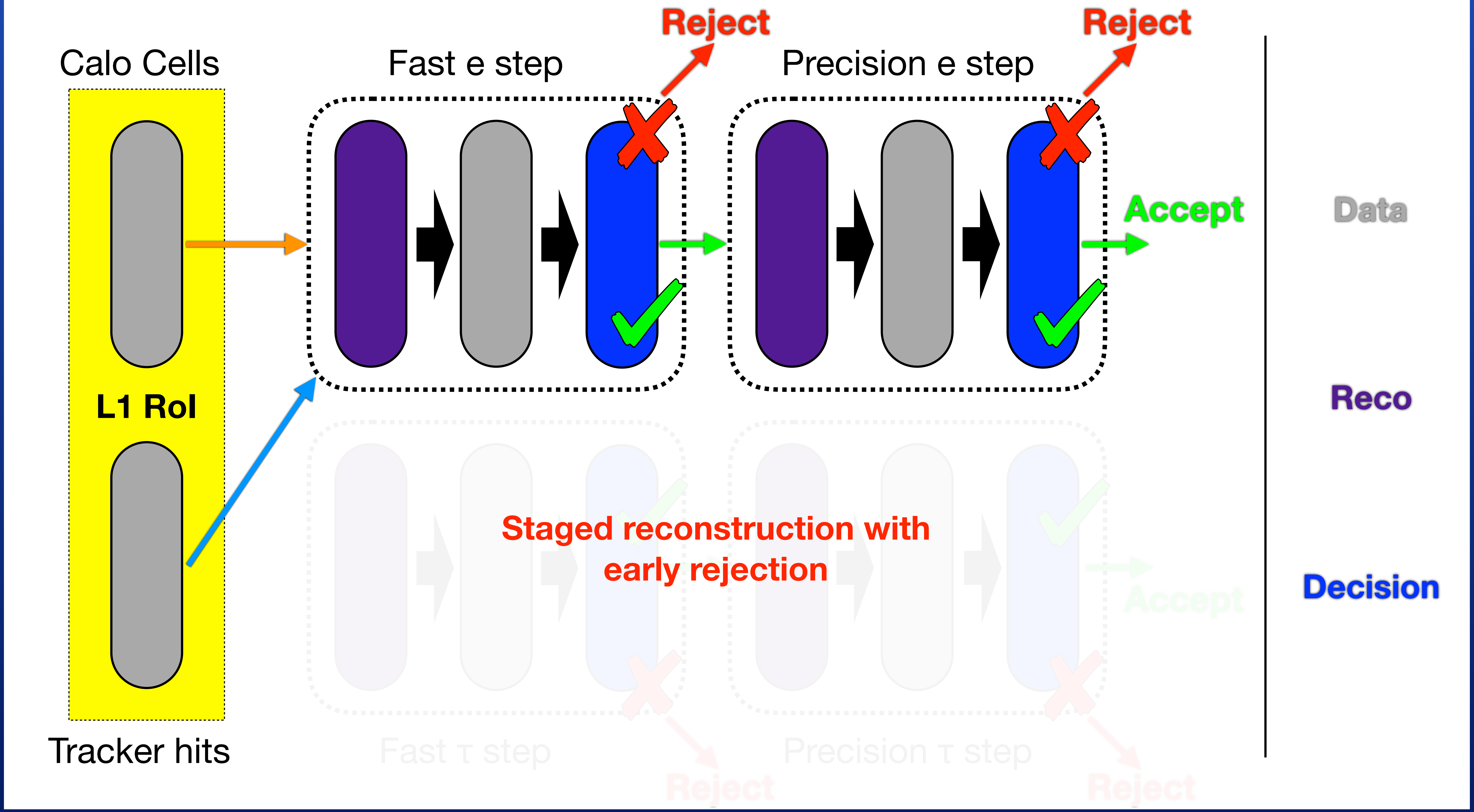
Average 400-500 ms / event processing time
— compare with O(30s) offline reconstruction

CORE HLT ELEMENTS

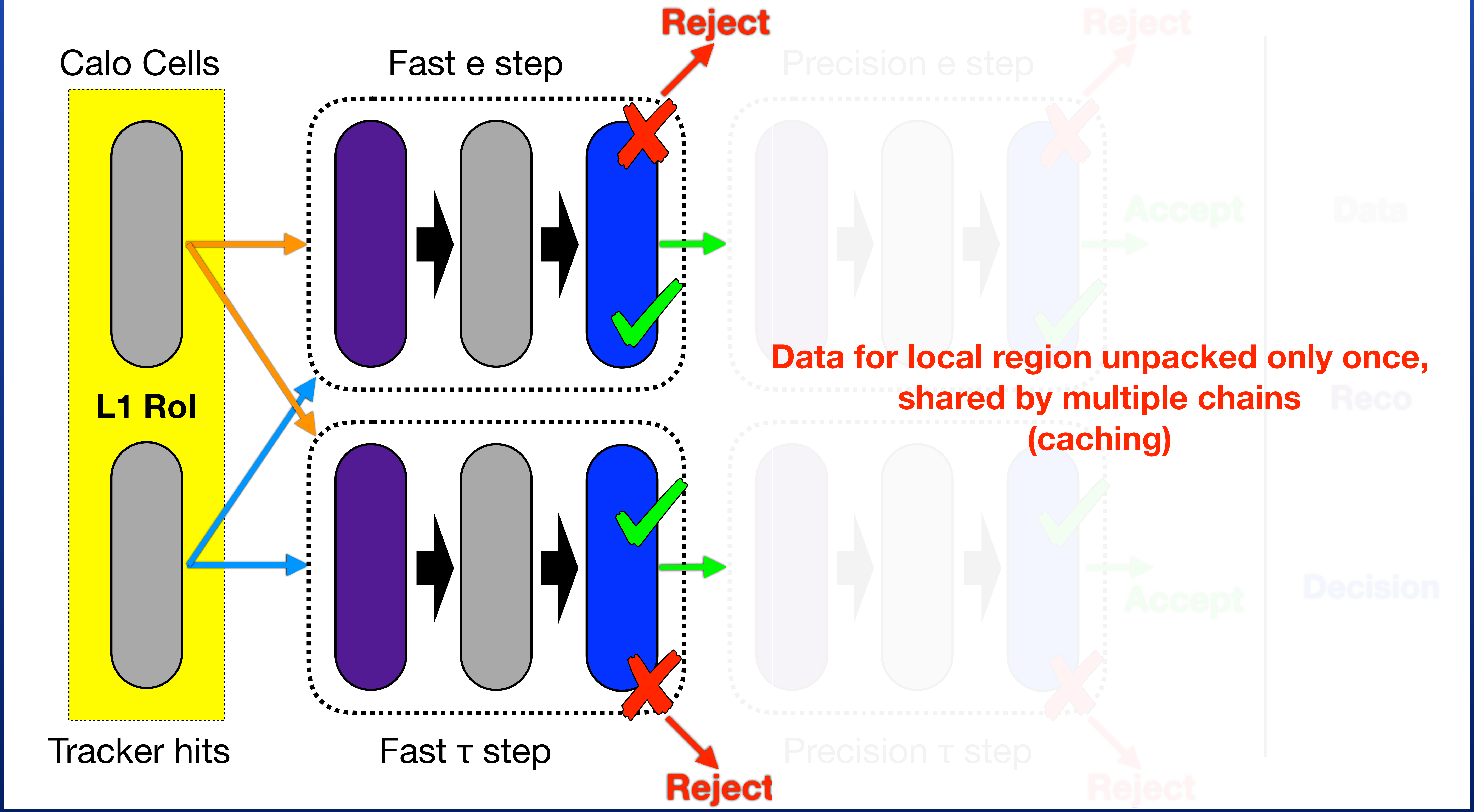
- Algorithmic code
  - Four-momentum reconstruction & Particle ID
  - Decision-making (“Hypothesis”)
- Regional reconstruction
  - Local detector readout where L1 trigger fired — “Regions of Interest” (all detector slices)
- Data flow & scheduling (“Control Flow”)
  - Reconstruct once, cache data for reuse
  - Early termination when event rejection is established
- Decisions recorded as event data

TRIGGER DECISION SCHEMATIC

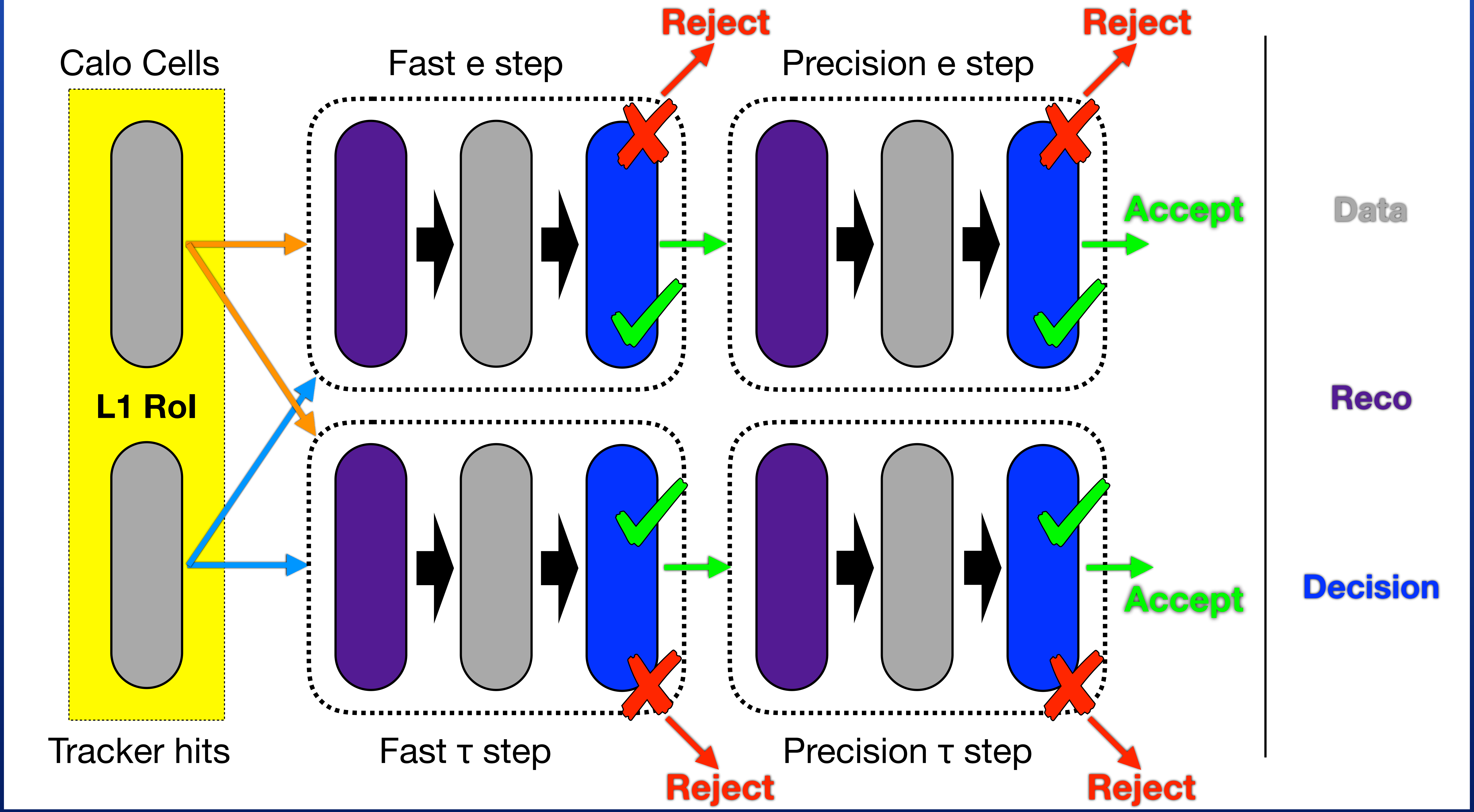
TRIGGER DECISION SCHEMATIC



TRIGGER DECISION SCHEMATIC



TRIGGER DECISION SCHEMATIC



TRIGGER DECISION SCHEMATIC

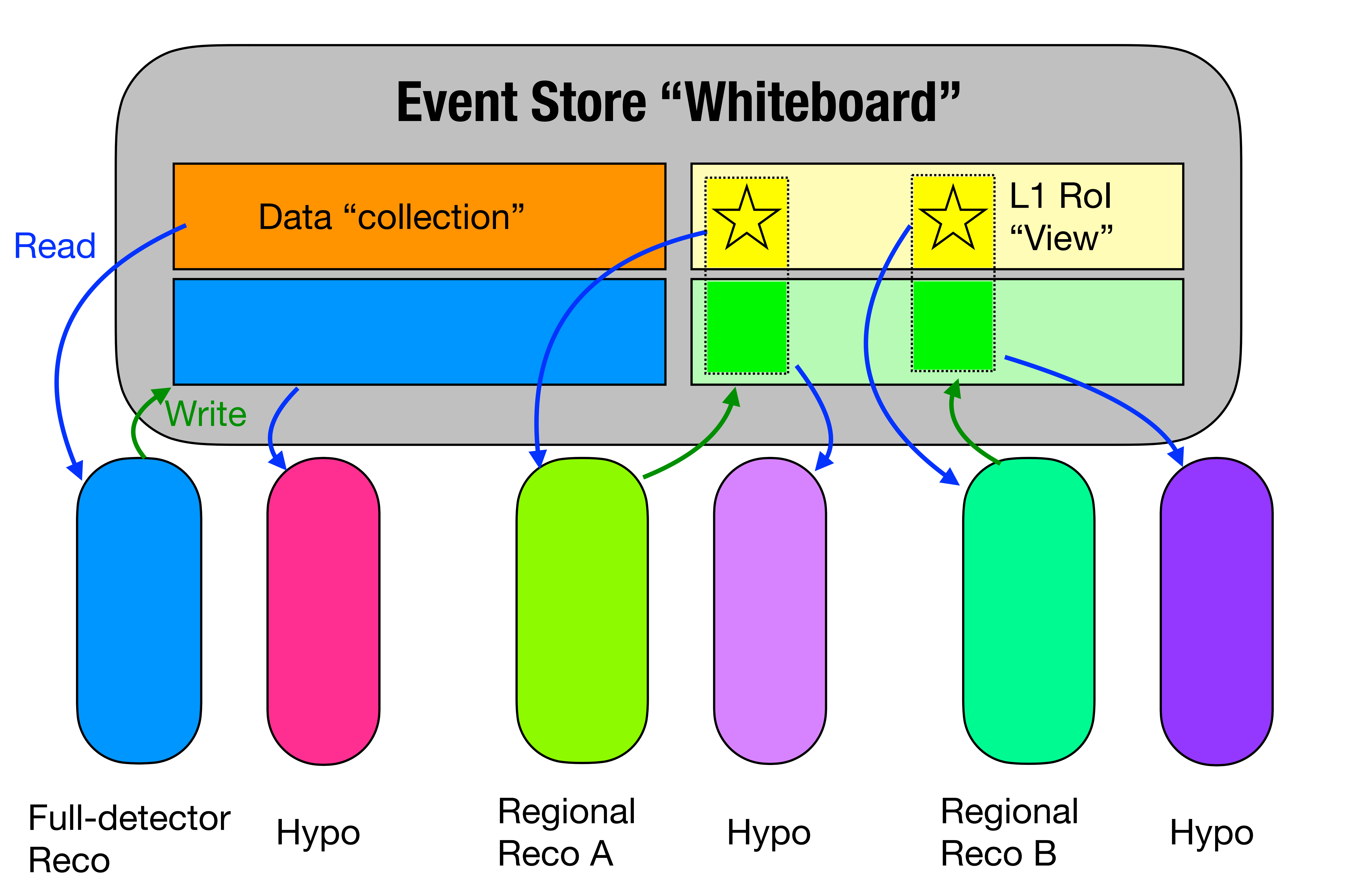
CONSTRAINTS & SOLUTIONS

- Ensure code is thread-friendly
  - Const data access
  - State-free execution
- Asynchronous conditions access
  - Conditions object containers in data store
  - Intervals Of Validity mapped to data objects
  - Conditions Algorithms populate store for new IOV
- Removal of trigger-specific steering wrapper
  - Integration of Control Flow elements into framework allowing execution to be stopped early

Core HLT Elements in AthenaMT

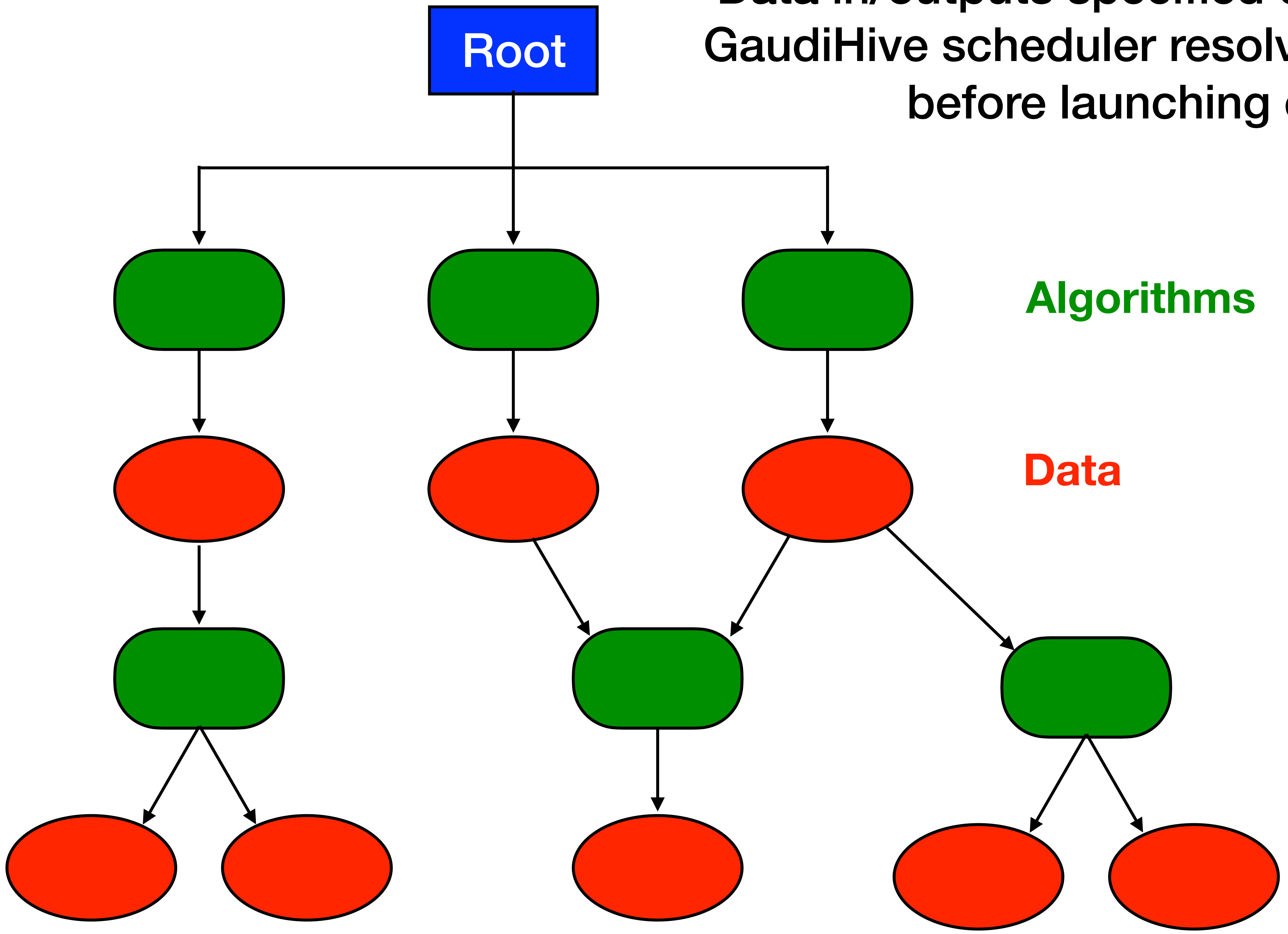
- Algorithmic code
  - Shared where possible with offline domain
- Regional reconstruction
  - Views in Event Store — restrict geometric acceptance transparently to algorithmic code
  - Parallel reconstruction of multiple Regions of Interest
- Data flow & scheduling
  - GaudiHive graph-based scheduler
    - Declarative data access
  - Control Flow sequences
- Decisions recorded as event data

AthenaMT Data Access



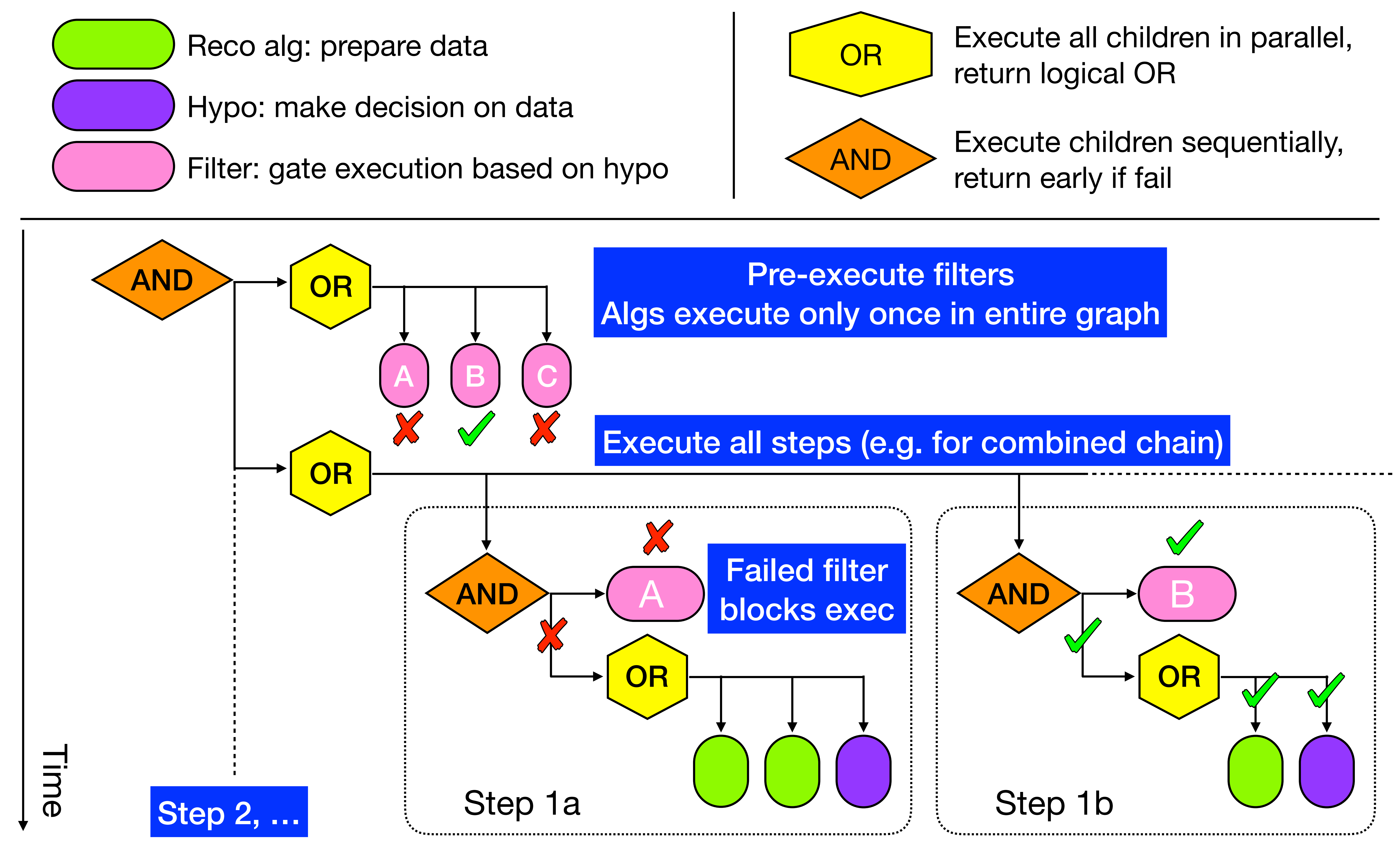
ATHENAMT SCHEDULING

Data in/outputs specified during initialisation. GaudiHive scheduler resolves execution graph before launching event loop



Sub-algorithm components (tools) propagate data inputs/outputs to parent

TRIGGER SCHEDULING



MIGRATION STATUS

- Core infrastructure largely in place
- “Mechanical” migration of reconstruction elements in progress
  - Jointly with offline code
- O(100) Run 2 chains in test trigger menu, some fraction of which fully implemented

SUMMARY

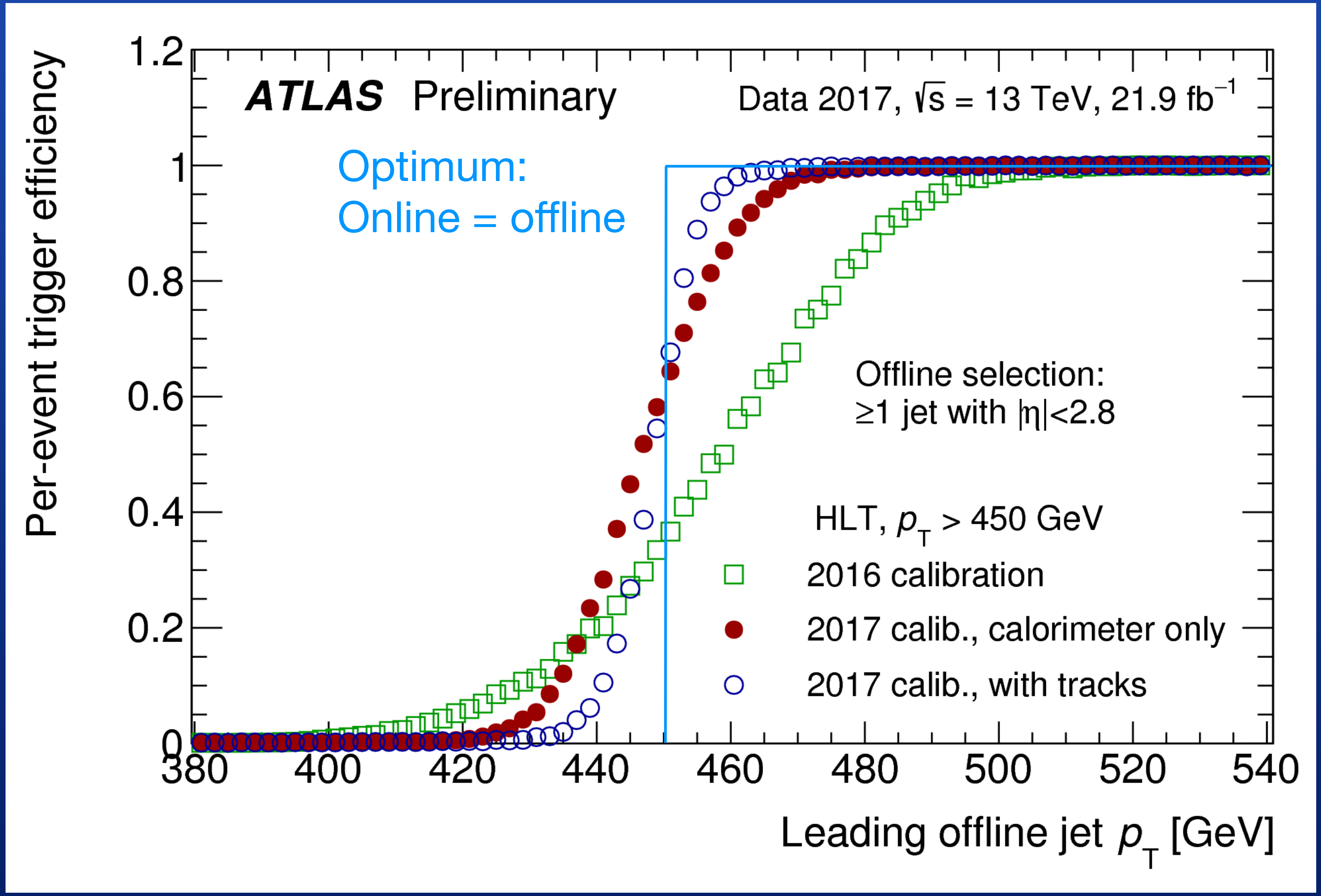
- Common Gaudi framework extended to support multithreading
- AthenaMT extensions permit ATLAS HLT operation for O(500ms) reconstruction and event filtering
  - Regional reconstruction with Event Views
  - Early rejection with Control Flow gate nodes
- Some Run 2 chains fully implemented
- Validation & performance assessments to come

References

- AthenaMT: ATL-SOFT-PROC-2017-019
- GaudiHive/Avalanche scheduler: http://concurrency.web.cern.ch/GaudiHive

BACKUPS

PHYSICS PERFORMANCE



Improvement from implementing offline corrections at cost of CPU

ATLAS-DAQ-PUB-2018-002

Overview of Athena architecture