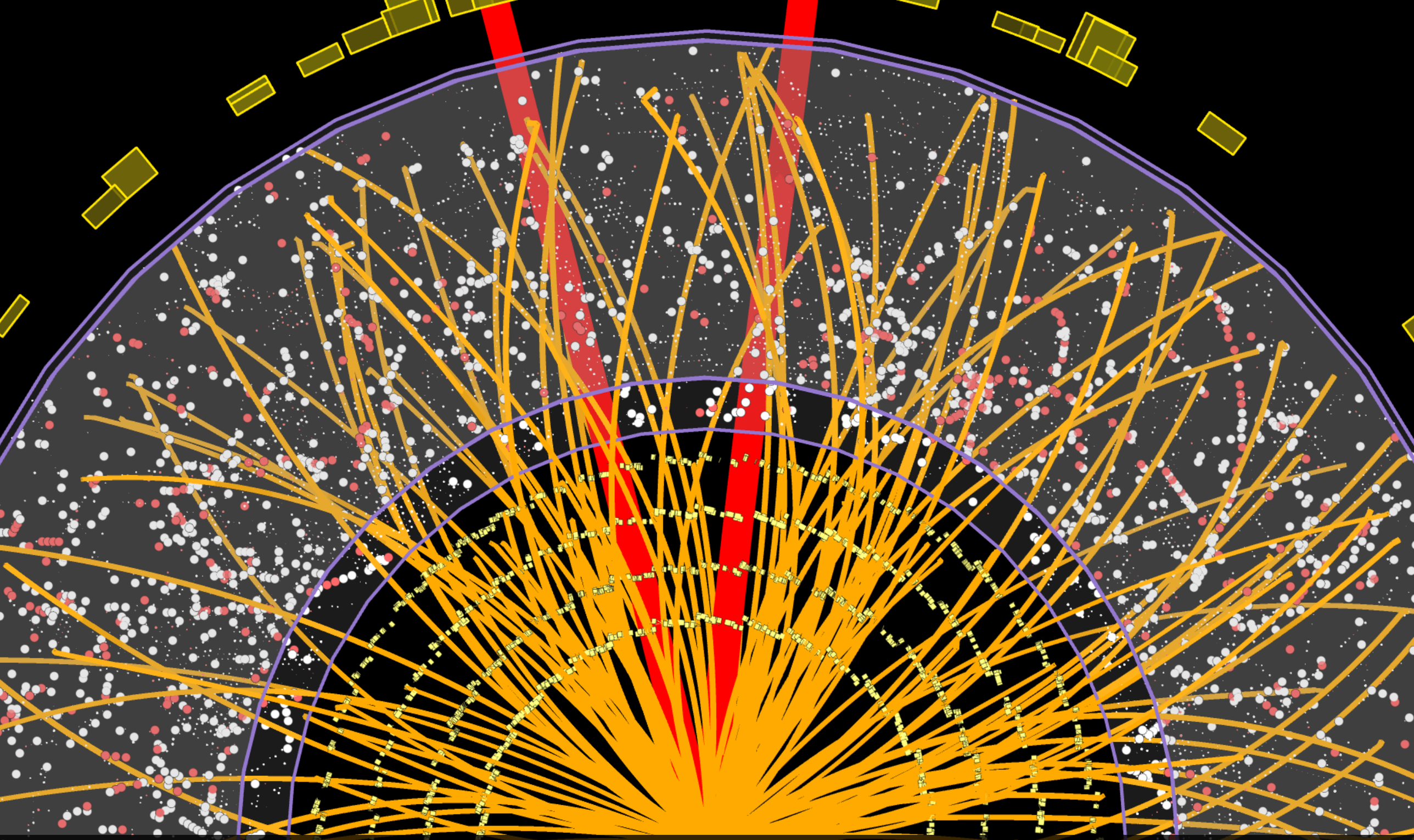


The ATLAS FastTracKer

Pioneering the next era of hardware track triggers

Tova Holmes, on behalf of the ATLAS Collaboration

ICHEP2018 Seoul
July 5, 2018

OVERVIEW

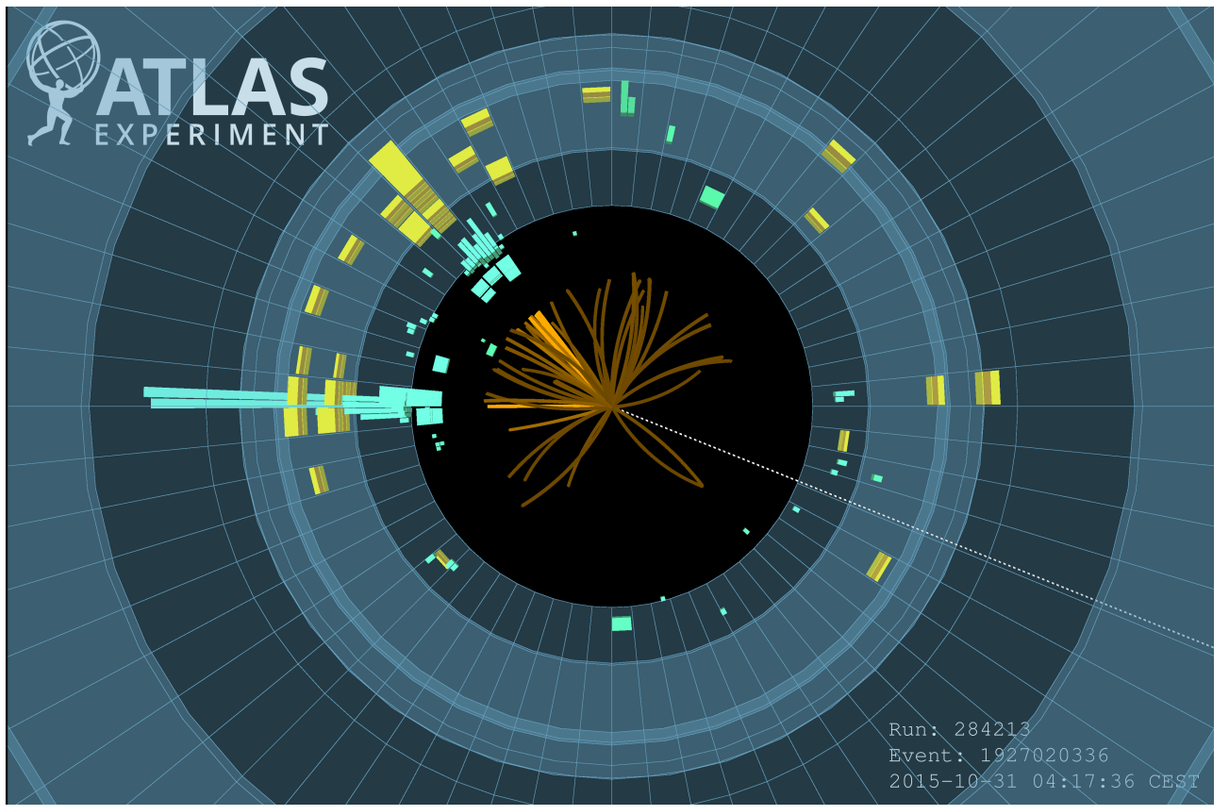
- Triggering in the ATLAS Detector
- FastTracKer Design
- FastTracKer Comissioning

THE ATLAS TRIGGER

Throwing away data

ATLAS rejects more than 99.99% of collisions

Final decision on what to keep is made in around 250 ms



ATLAS public event displays

How do we decide if this event is worth keeping?

(image of an event with analysis-level “offline” reconstruction)

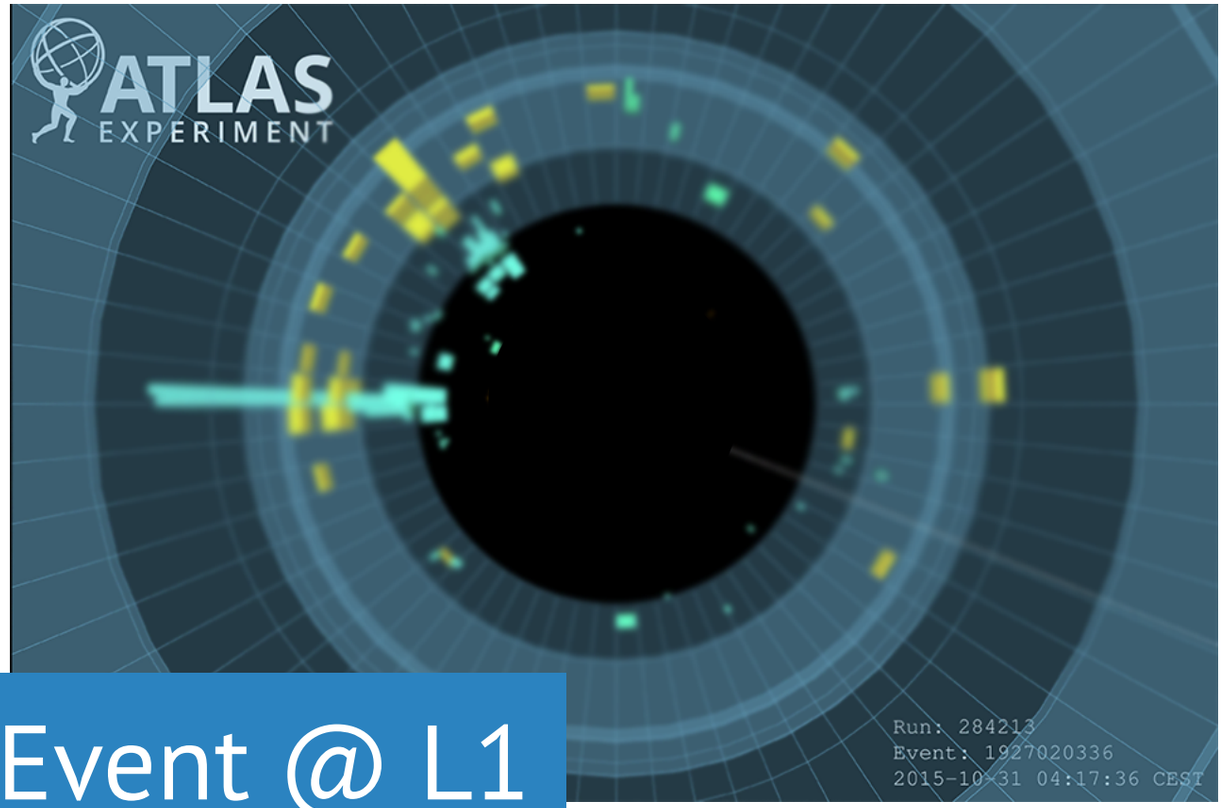
Throwing away data

Level 1 trigger decisions are made with rough calorimeter and muon information

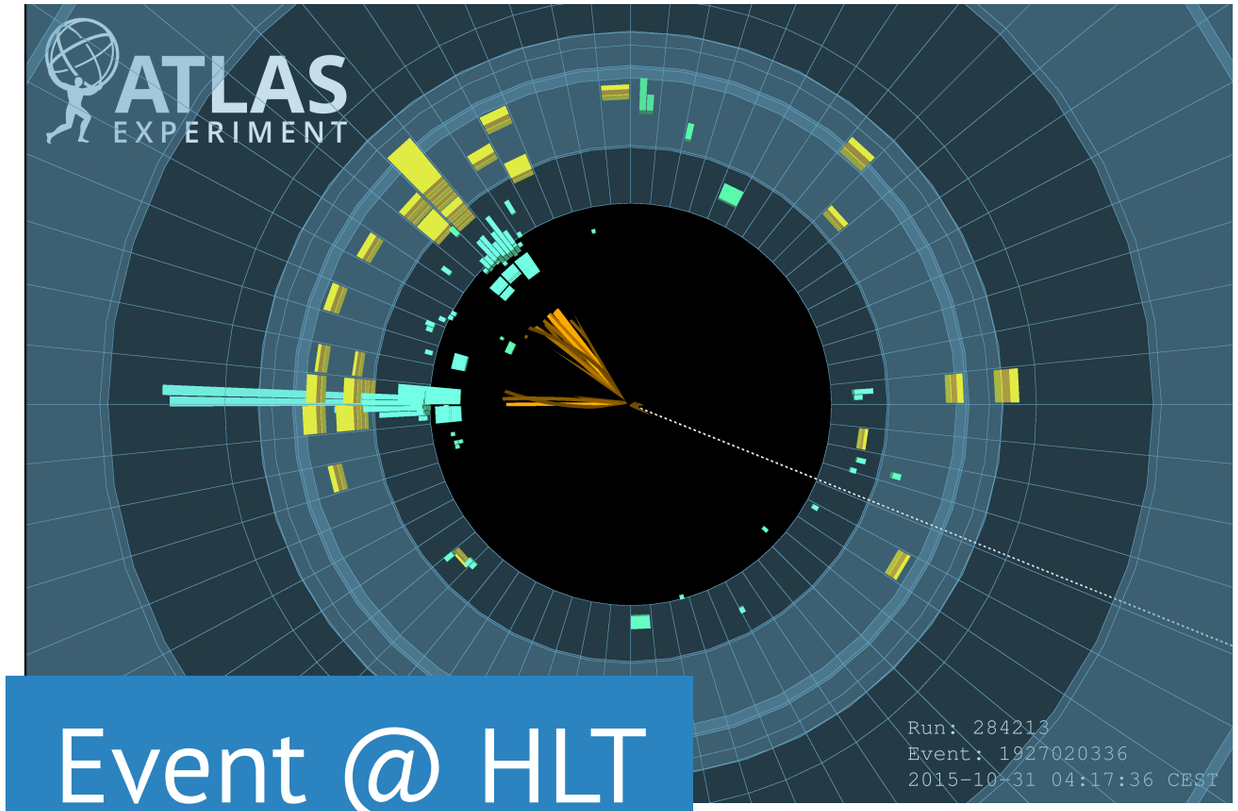
40 MHz → 100 kHz

High Level Trigger uses full precision information around objects identified at L1

100 kHz → 1 kHz



Event @ L1



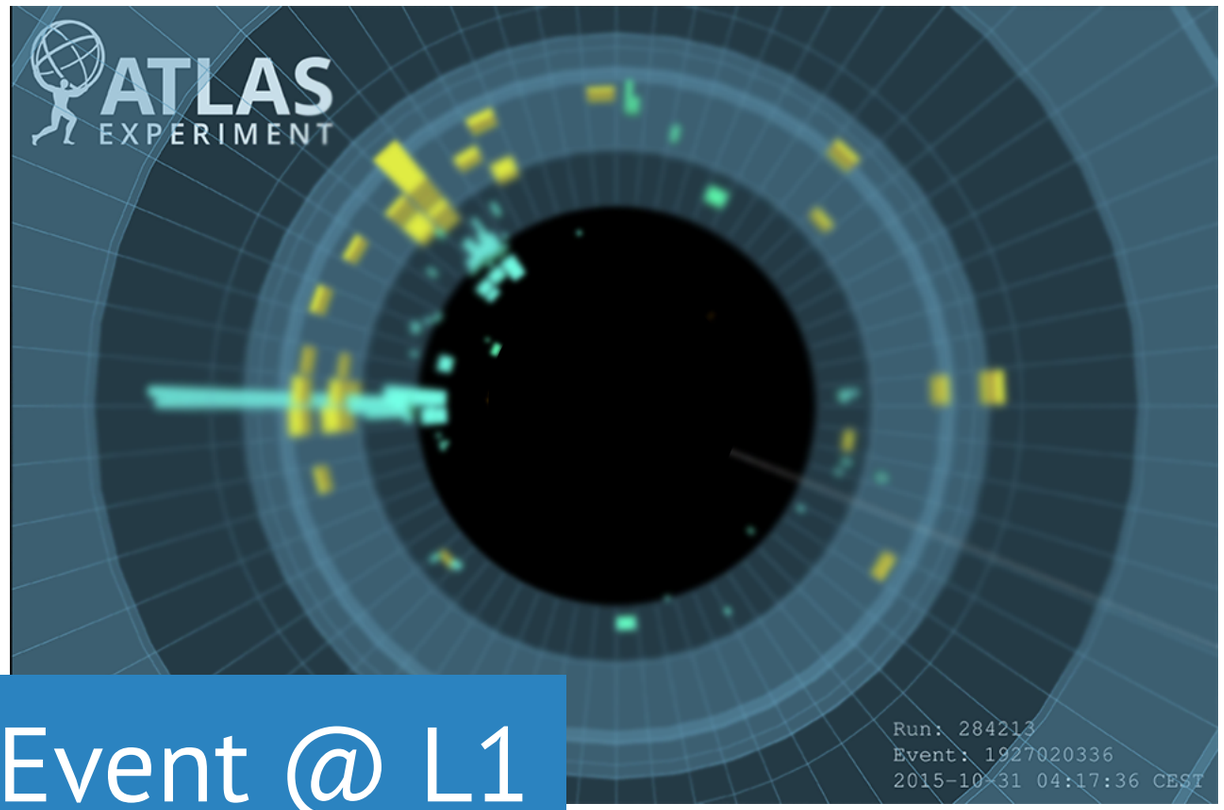
Event @ HLT

Throwing away data

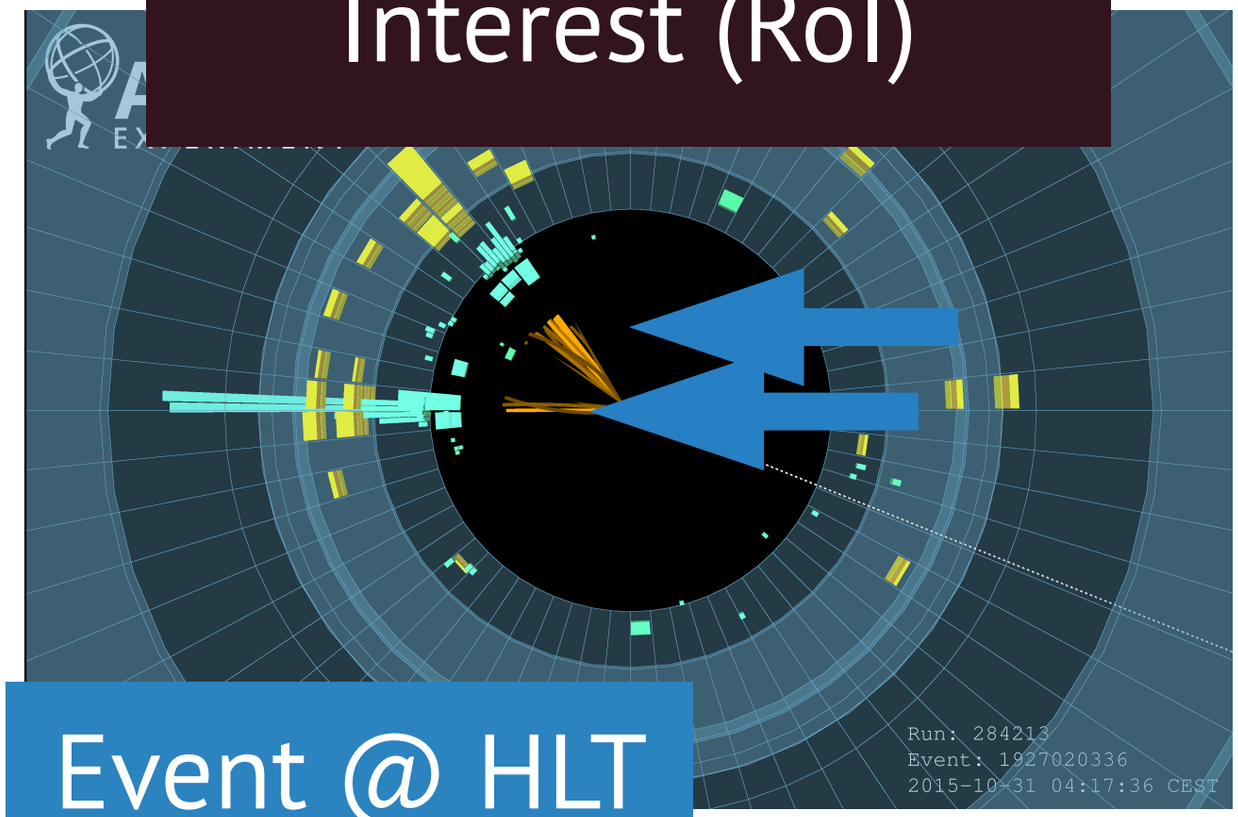
Level 1 trigger decisions are made with rough calorimeter and muon information

High Level Trigger uses full pre information around objects identi

tracking performed mainly in Regions of Interest (RoI)



Event @ L1



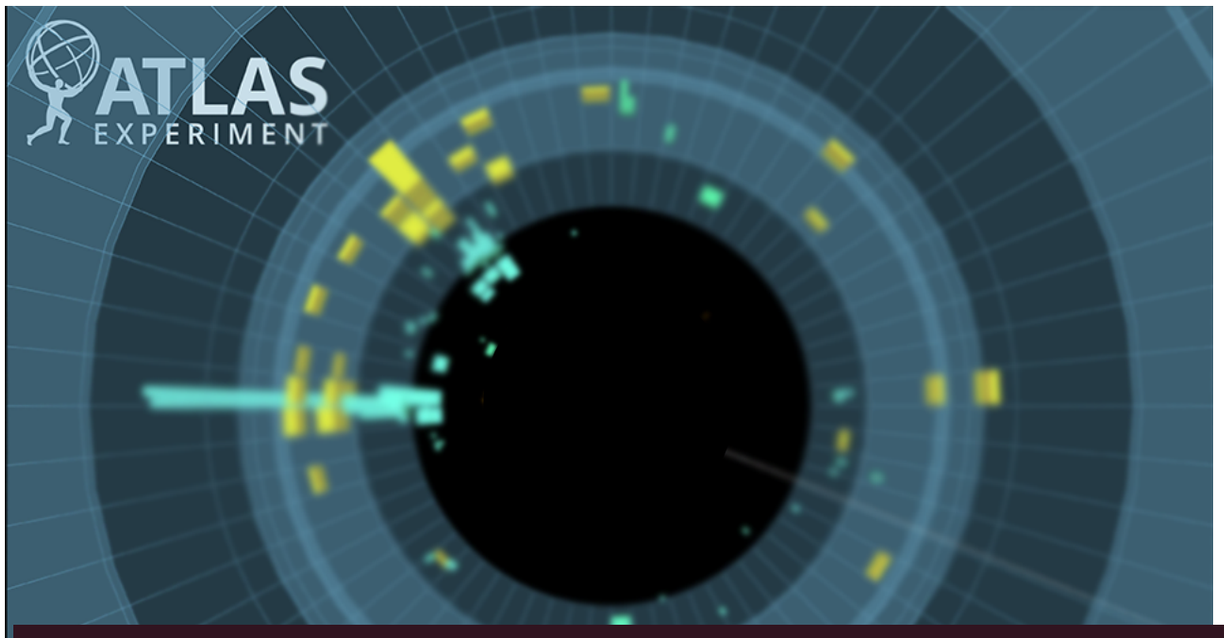
Event @ HLT

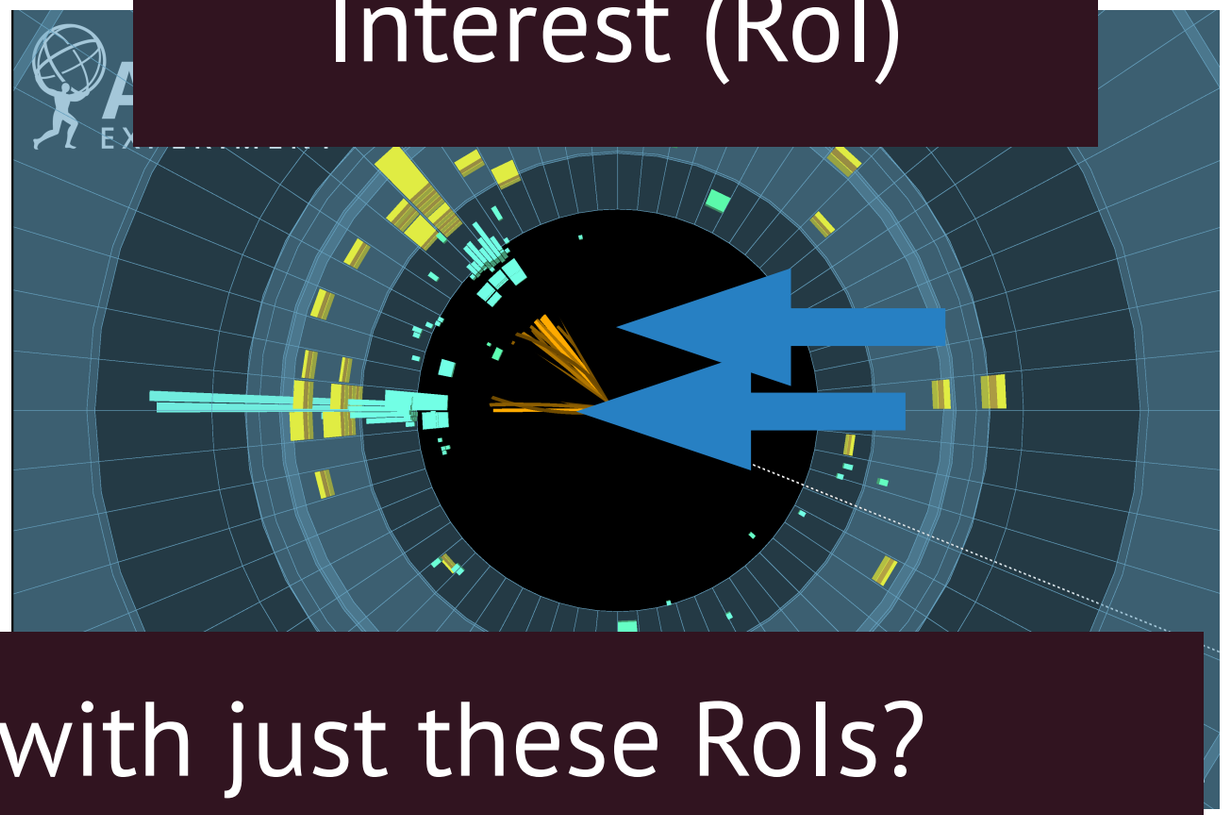
Throwing away data

Level 1 trigger decisions are made with rough calorimeter and muon information

High Level Trigger uses full pre information around objects identi

tracking performed mainly in Regions of Interest (RoI)





How well can we do with just these RoIs?

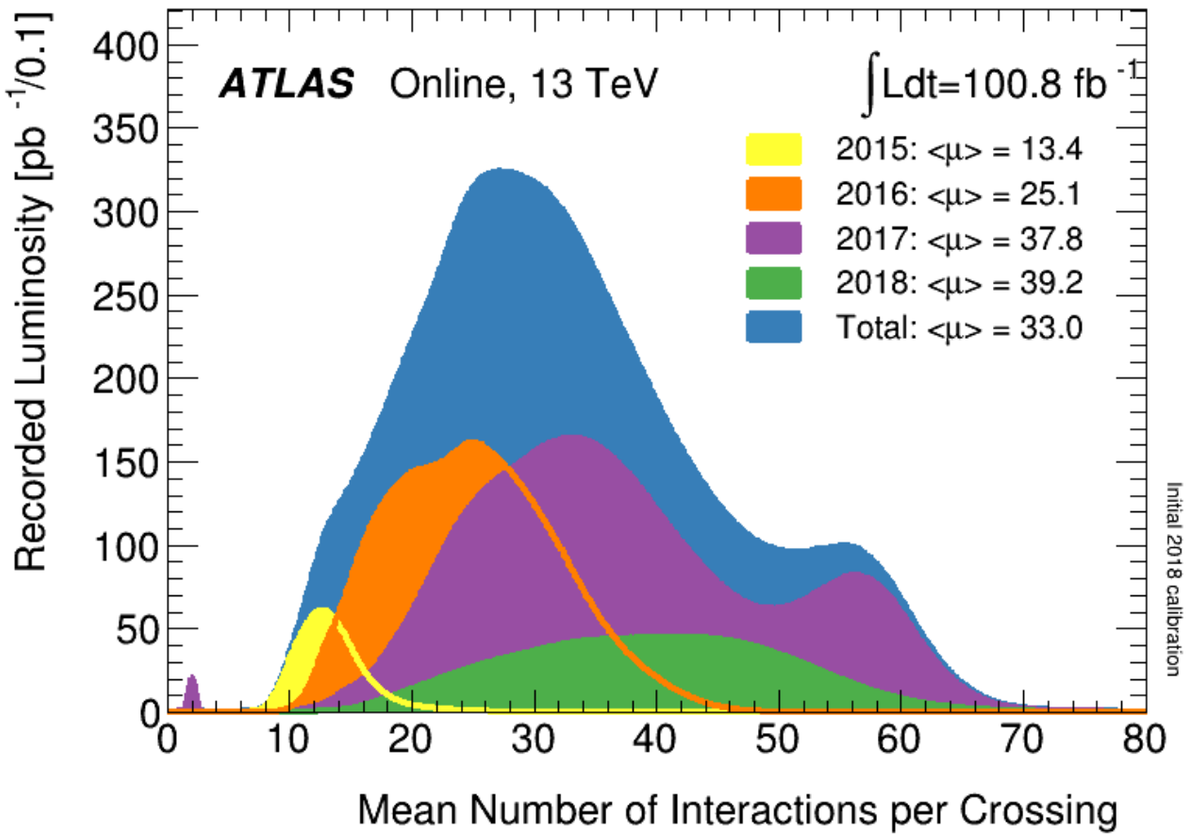
Pileup @ the LHC

~40 simultaneous *pp* interactions per event in 2018

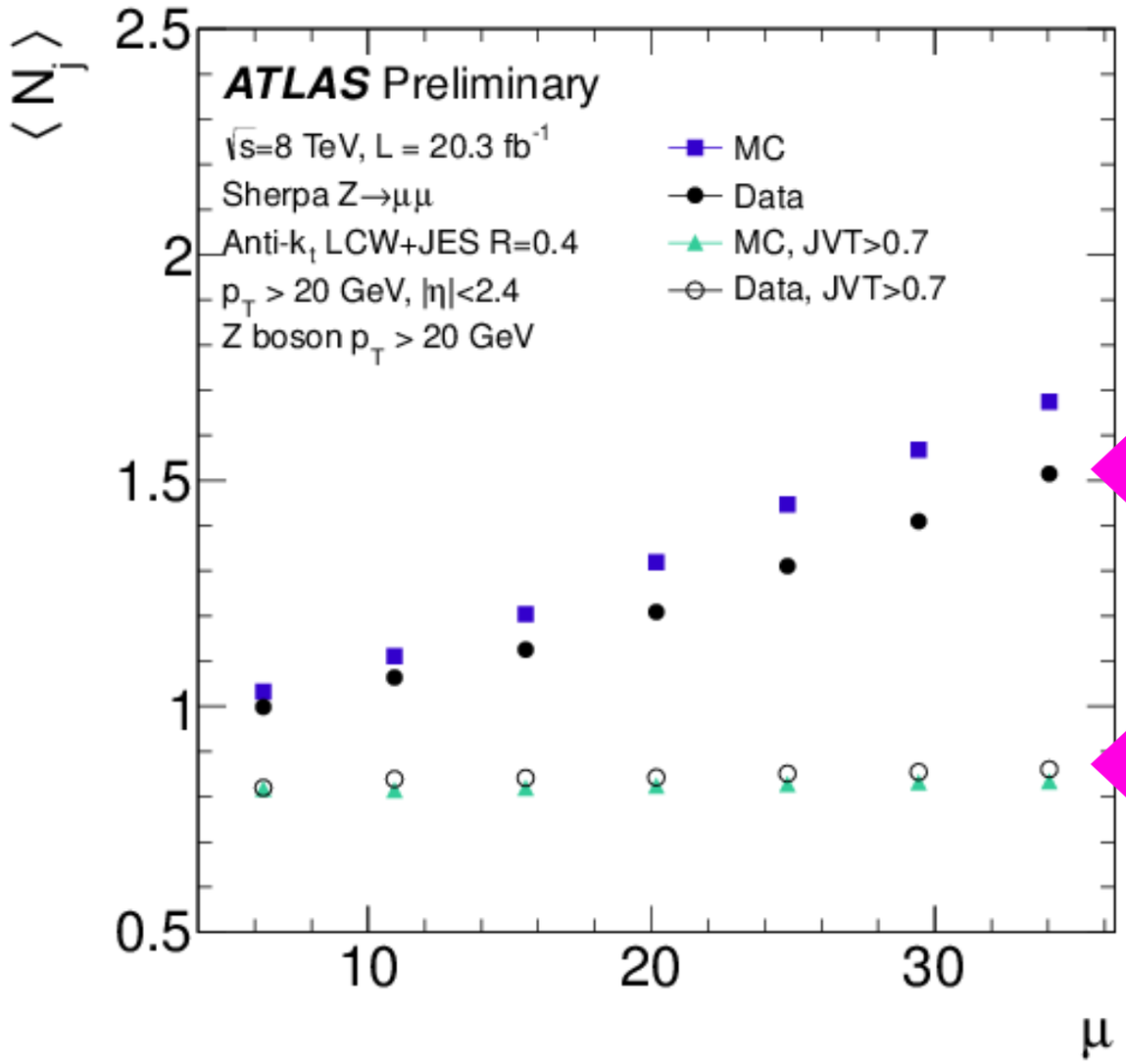
tracks let us identify objects from the primary vertex

(and ignore everything else)

need global tracking to do this for the full event!



Pileup @ the LHC



Tracking at the trigger level is essential to maintaining low trigger thresholds

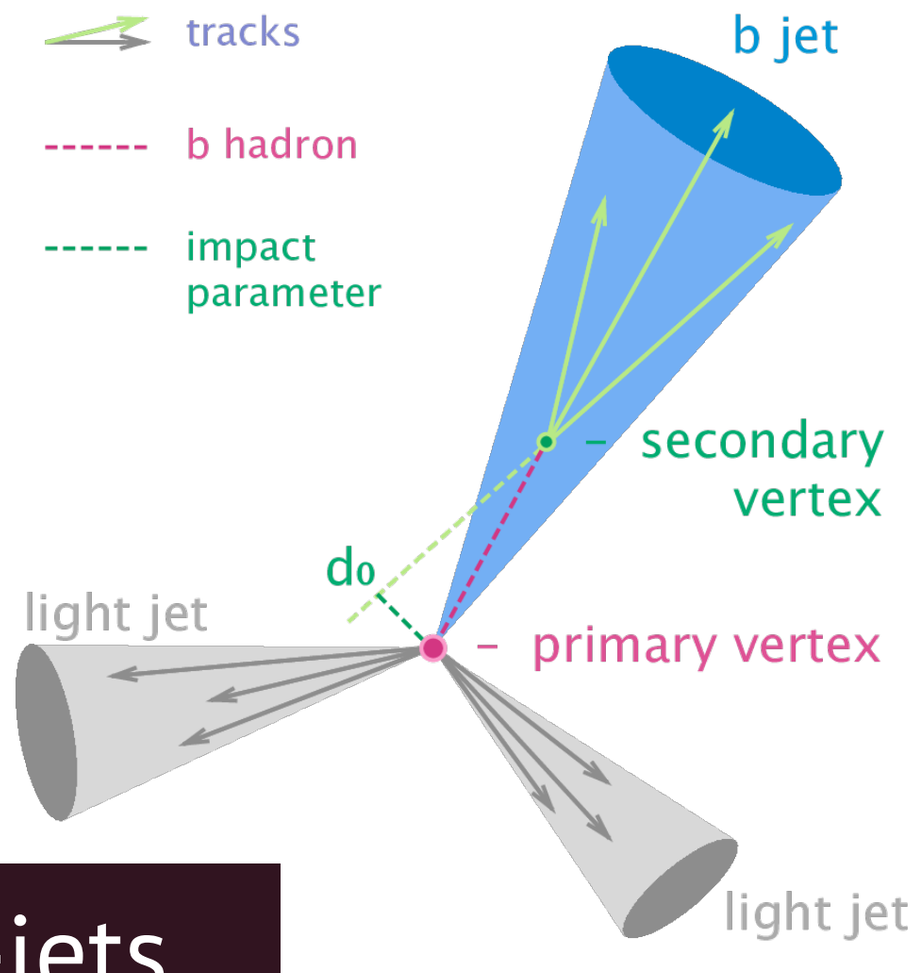
njets selected without tracking

njets selected with tracking

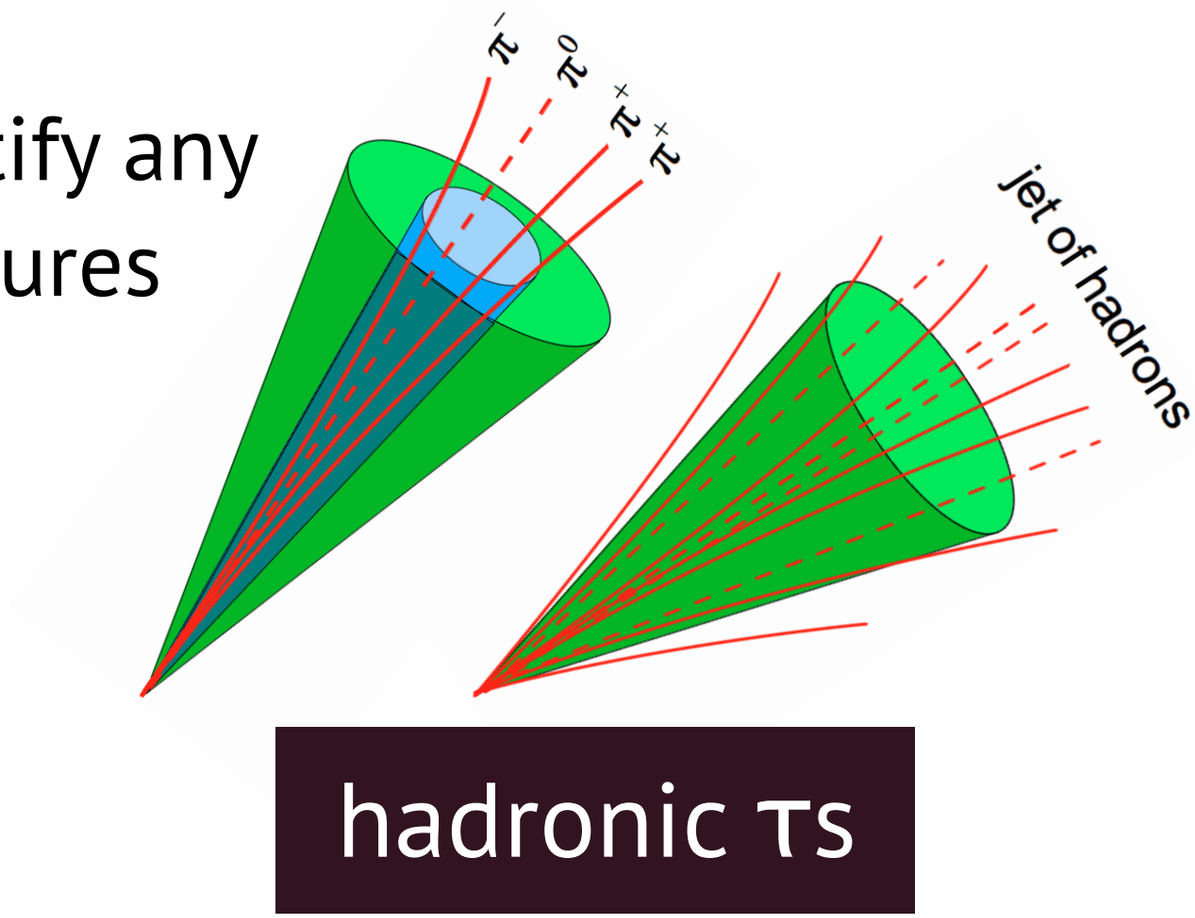
ATLAS-CONF-2014-018

More uses for tracking

Full-scan tracking can help identify any object with track-based signatures



b-jets



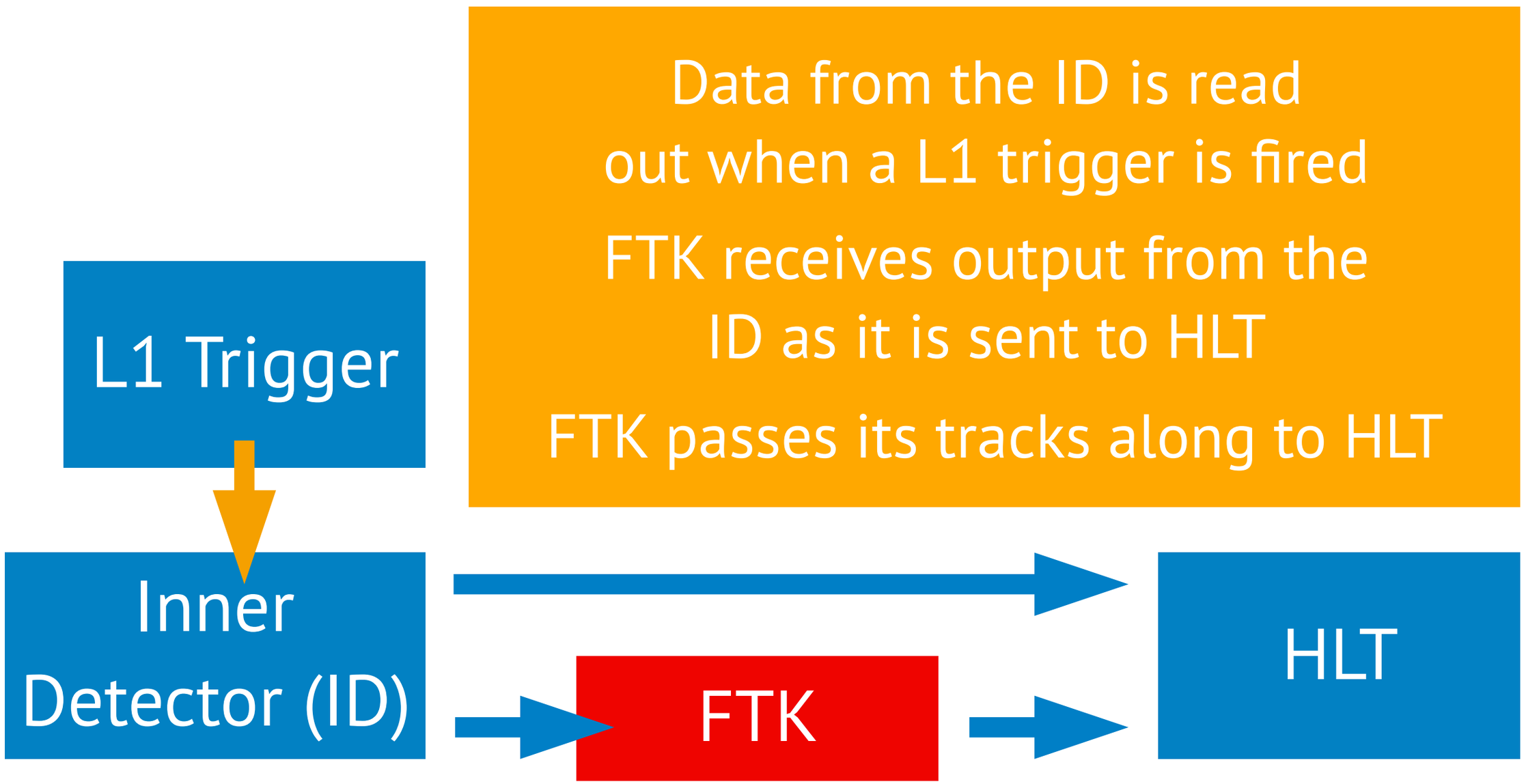
hadronic τs

If tracking is already available, frees up CPU at HLT for other tasks

THE ATLAS FASTTRACKER

Adding the FastTracKer (FTK) to the trigger

- FTK performs hardware-based tracking on silicon hits
  - provides HLT with >1 GeV tracks ID acceptance (|η|<2.5)



Time constraints

Offline track reconstruction for the full tracking volume requires about 10 s / event

To keep up with L1 rates, FTK must do tracking for the full event in ~.1 ms

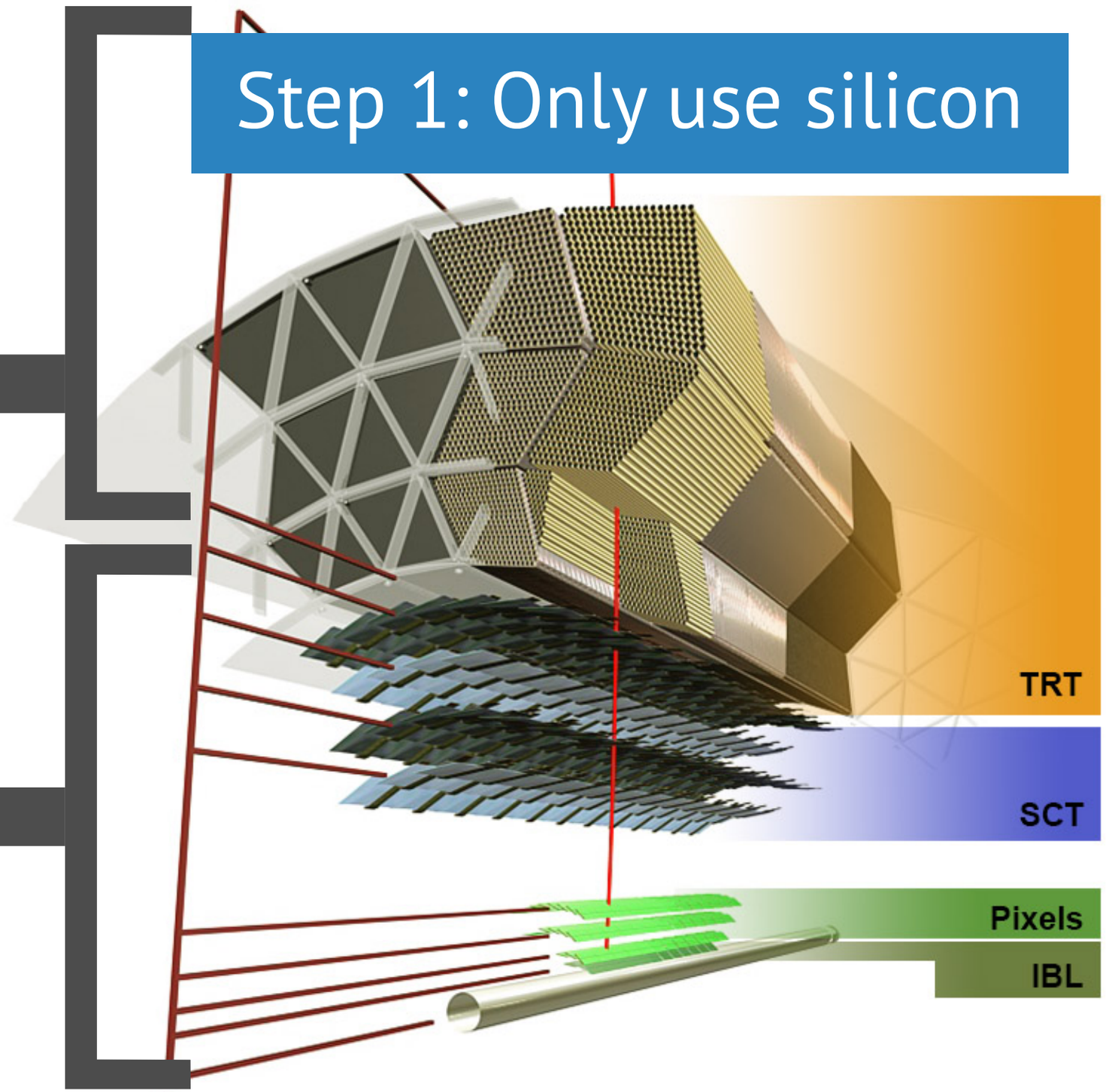
Requires time reduction of ~5 orders of magnitude

How can we track so fast?

Making Tracking a Simpler Problem

Straight to HLT

Split signals go to HLT and FTK

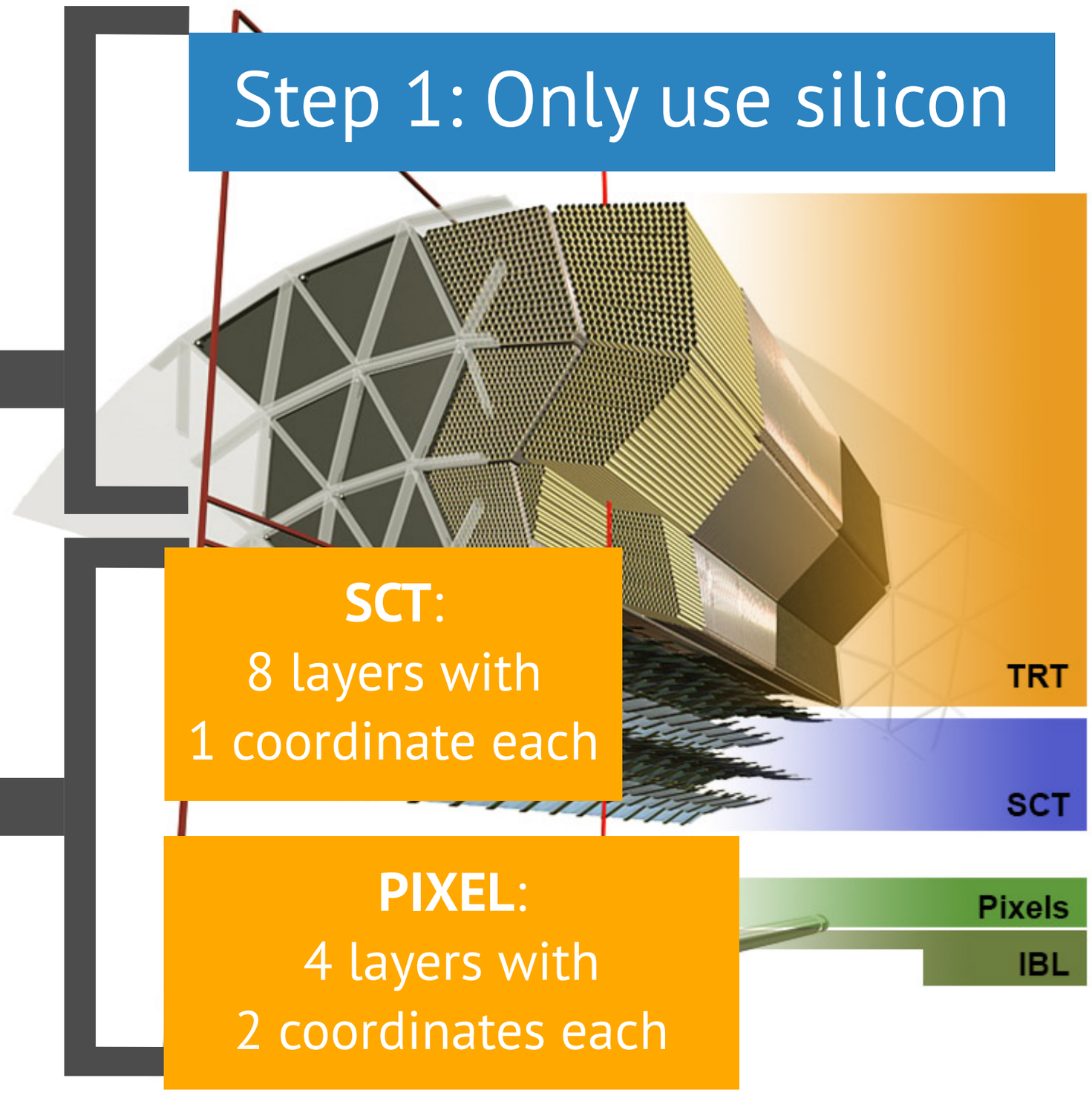


Making Tracking a Simpler Problem

Straight to HLT

Split signals go to HLT and FTK

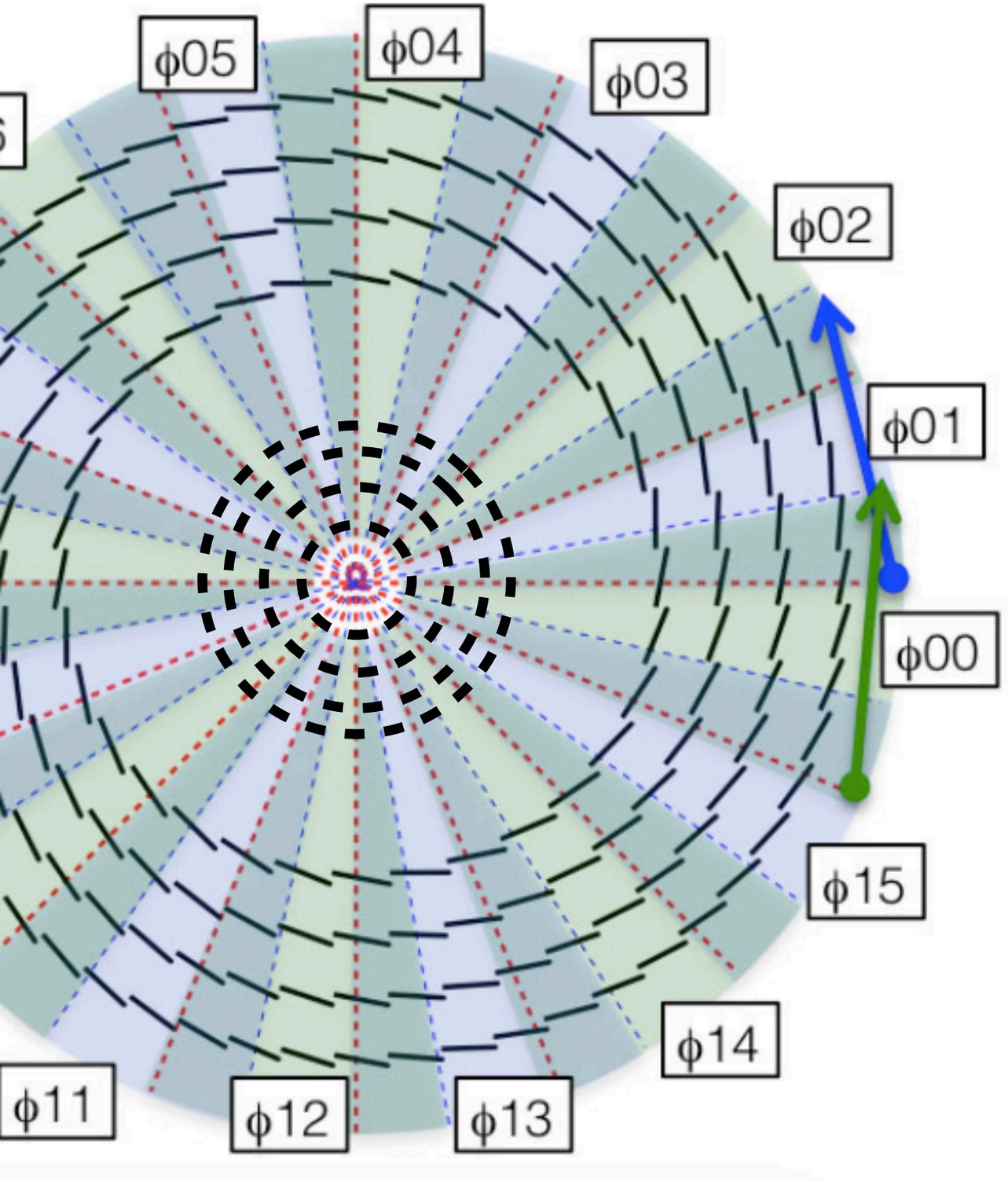
Step 1: Only use silicon



SCT:
8 layers with
1 coordinate each

PIXEL:
4 layers with
2 coordinates each

Making Tracking a Simpler Problem



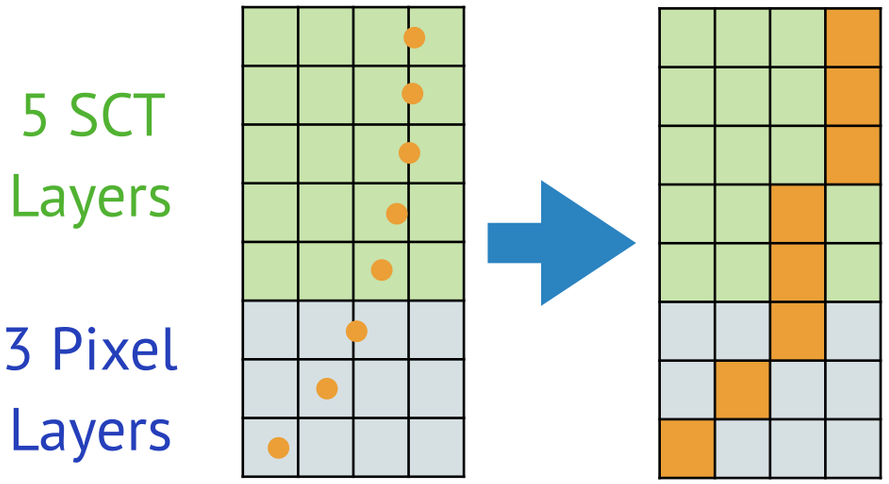
Step 2: Parallelize

Divide the detector into 64 overlapping towers

Send data from each tower to separate processing units

Making Tracking a Simpler Problem

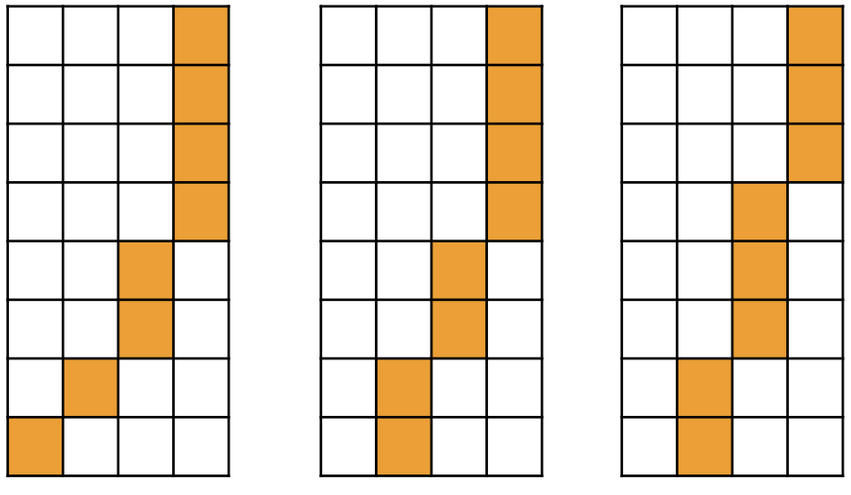
Start with 8/12 silicon layers of ATLAS

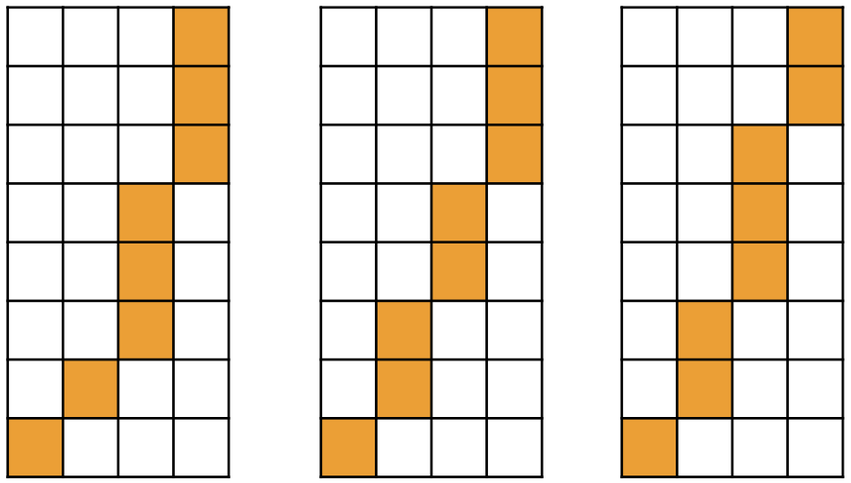


Step 3: Pattern Match

Divide each layer into coarse chunks

Making Tracking a Simpler Problem



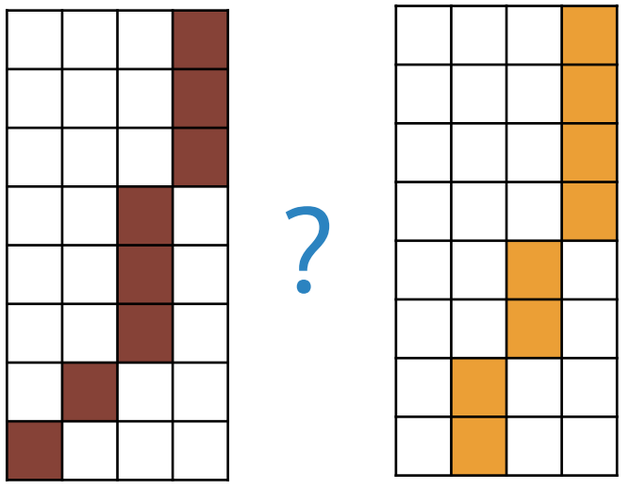


Step 3: Pattern Match

Divide each layer into coarse chunks

Define patterns of these chunks that correspond to tracks

Making Tracking a Simpler Problem



?

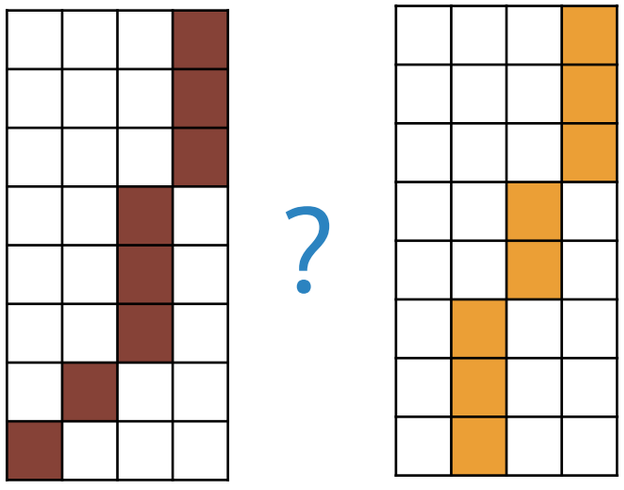
Step 3: Pattern Match

Divide each layer into coarse chunks

Define patterns of these chunks that correspond to tracks

Compare fired patterns to a stored bank of track-like patterns

Making Tracking a Simpler Problem



?

Step 3: Pattern Match

Divide each layer into coarse chunks

Define patterns of these chunks that correspond to tracks

Compare fired patterns to a stored bank of track-like patterns

Making Tracking a Simpler Problem

?

Step 3: Pattern Match

Divide each layer into coarse chunks

Define patterns of these chunks that correspond to tracks

Compare fired patterns to a stored bank of track-like patterns

Making Tracking a Simpler Problem

?

Step 3: Pattern Match

Divide each layer into coarse chunks

Define patterns of these chunks that correspond to tracks

Compare fired patterns to a stored bank of track-like patterns

Making Tracking a Simpler Problem

?

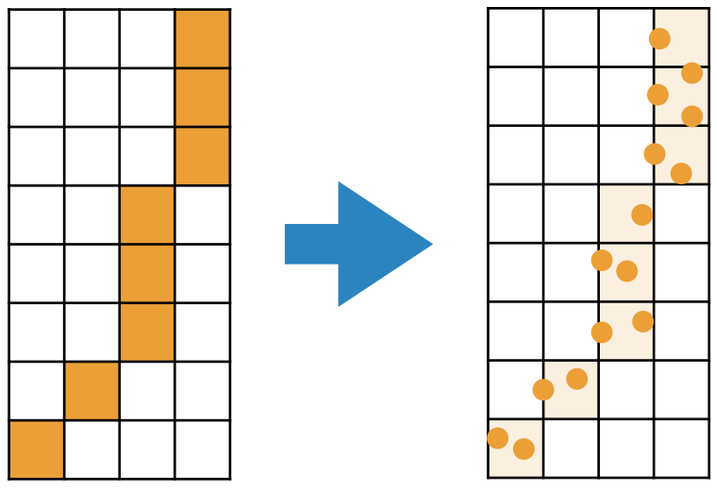
Step 3: Pattern Match

Divide each layer into coarse chunks

Define patterns of these chunks that correspond to tracks

Compare fired patterns to a stored bank of track-like patterns

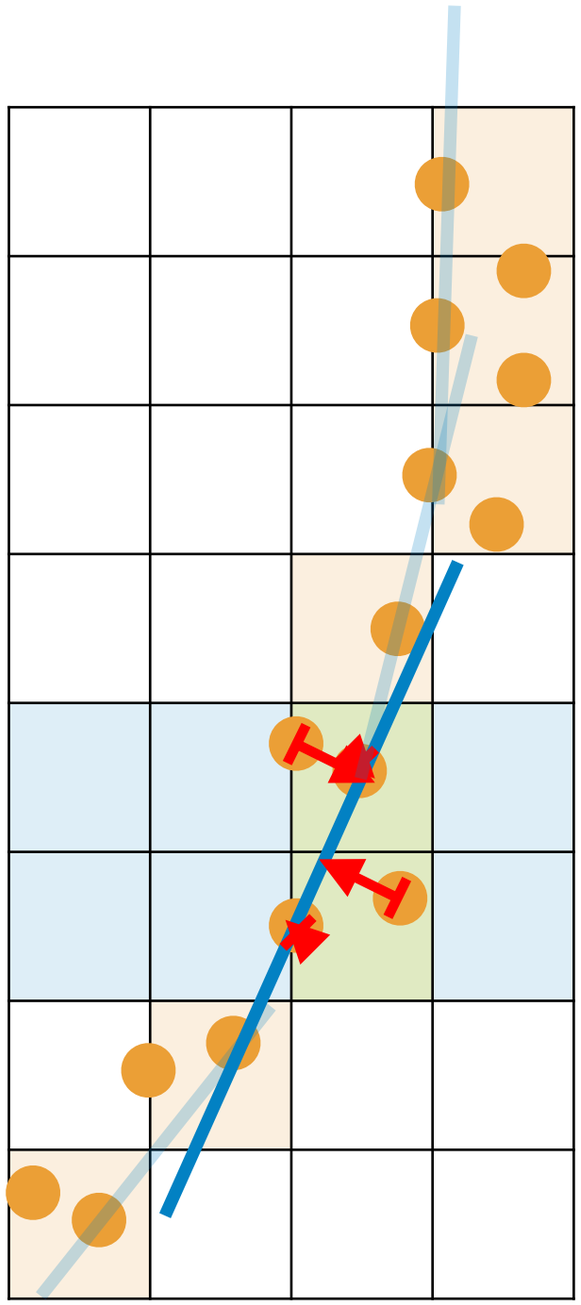
Making Tracking a Simpler Problem



Step 4: Fit a Subset

For matched patterns,
retrieve all full resolution hits

Making Tracking a Simpler Problem



Step 4: Fit a Subset

For matched patterns,
retrieve all full resolution hits

Perform a linearized fit
on the hits in 8 layers

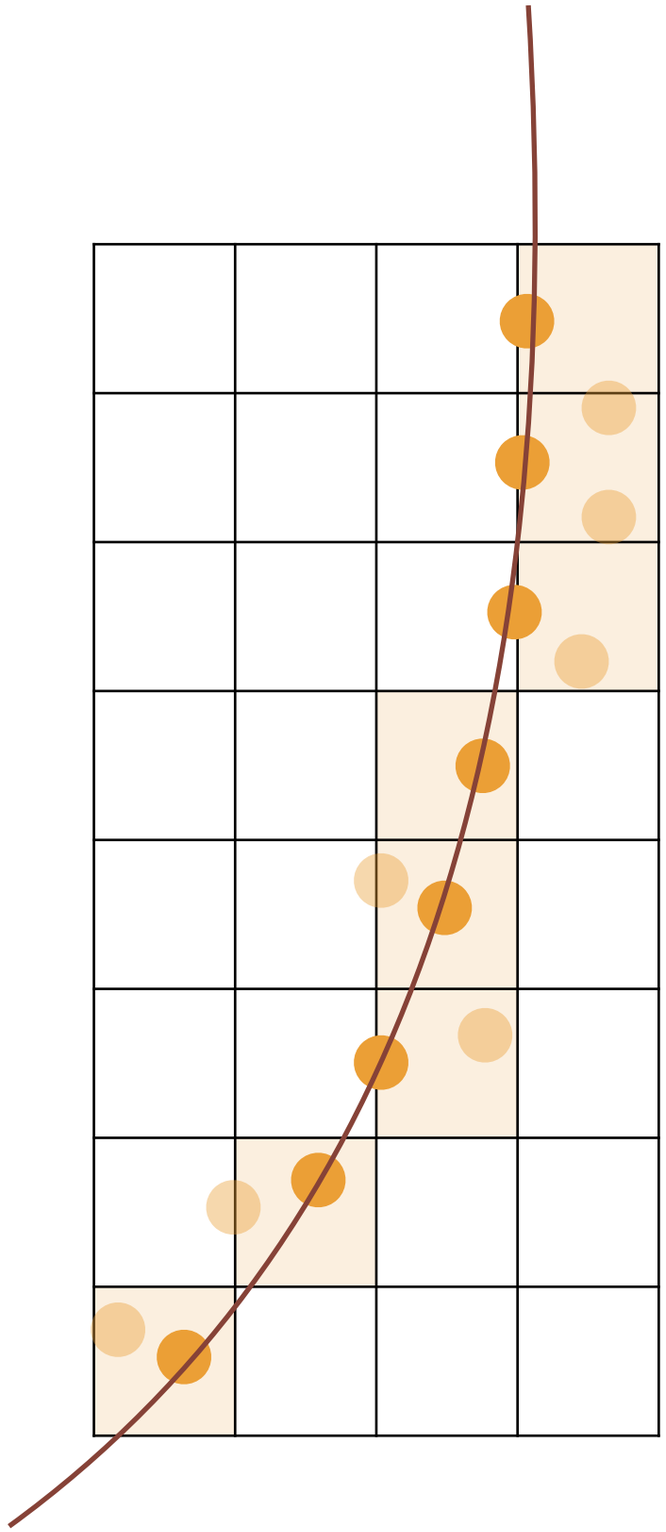
constants pre-defined per detector region

line: $y = mx + b$

each hit has a distance from the line: $\Delta x, \Delta y$

X² of fit: $\chi^2 = \sum_i^8 \sqrt{\Delta x_i^2 + \Delta y_i^2}$

Making Tracking a Simpler Problem



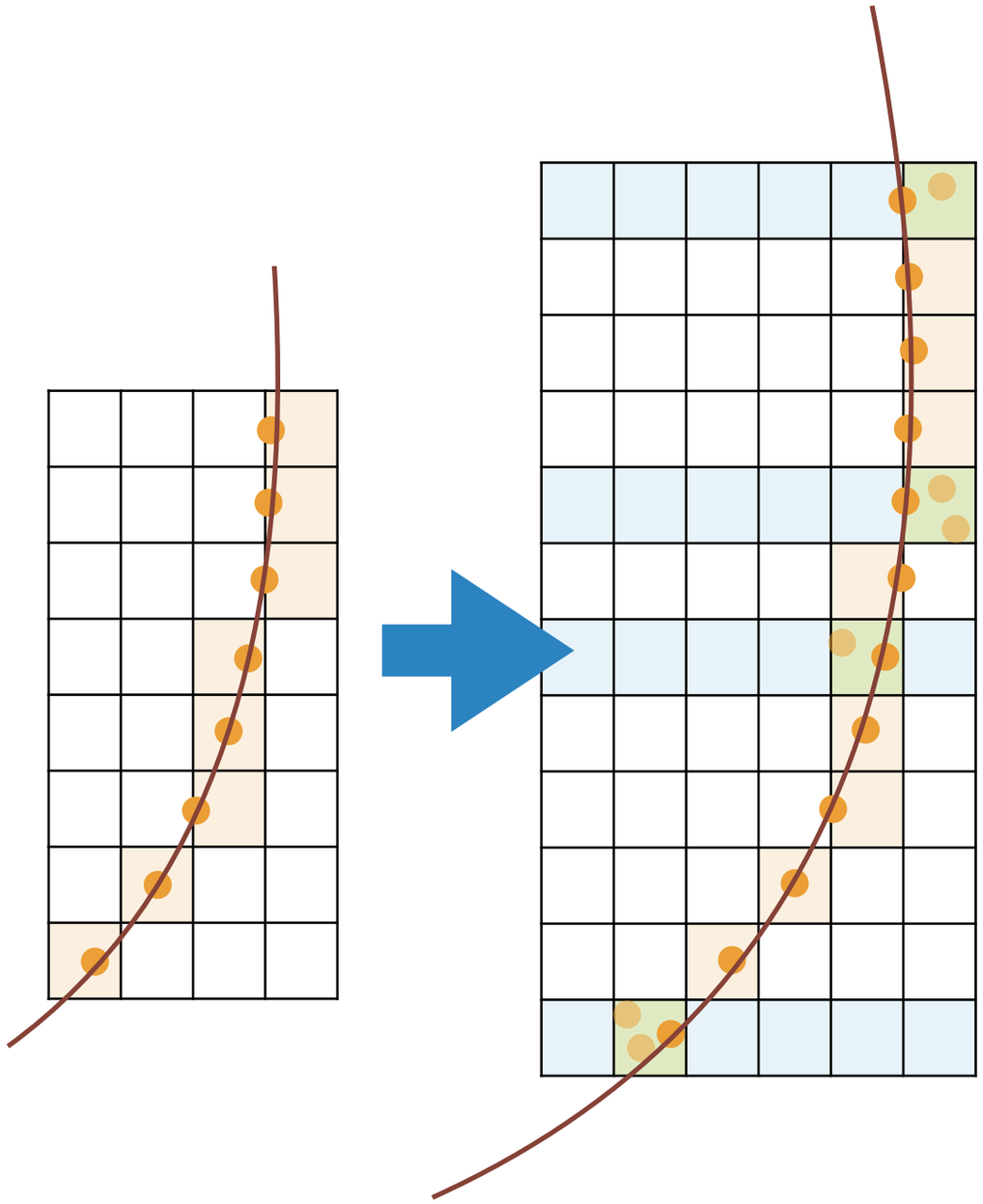
Step 4: Fit a Subset

For matched patterns,
retrieve all full resolution hits

Perform a linearized fit
on the hits in 8 layers

Keep tracks passing a χ^2 cut

Making Tracking a Simpler Problem



Step 4: Final Fit

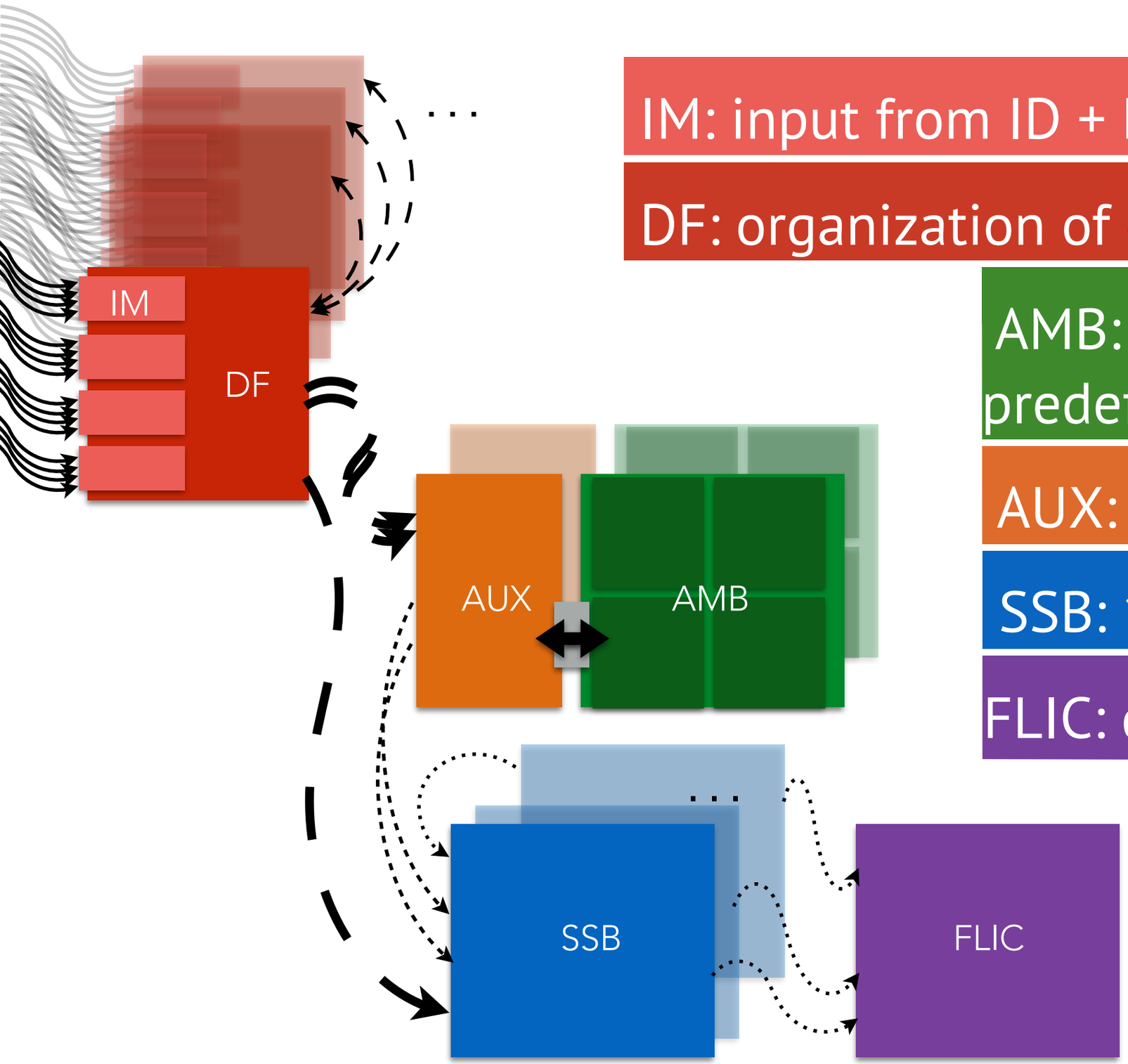
Look for nearby hits in remaining 4 silicon layers

Refit in all 12 layers

Send tracks* passing a χ^2 cut to HLT

*fit parameters also calculated linearly

FTK Boards



IM: input from ID + hit clustering

DF: organization of data into towers

AMB: matching clusters to predefined patterns

AUX: 8-layer track fitting

SSB: 12-layer track fitting

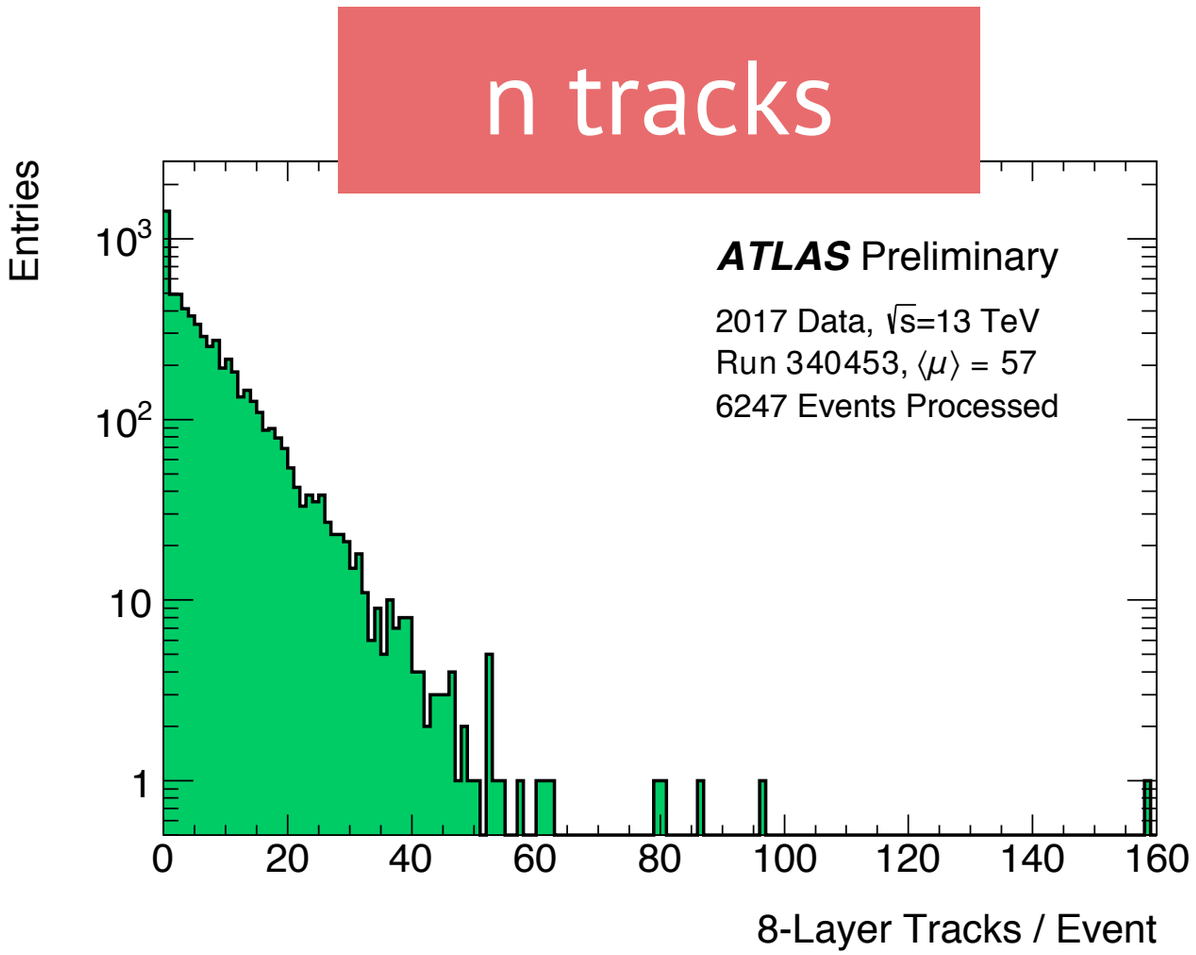
FLIC: data formatting for HLT

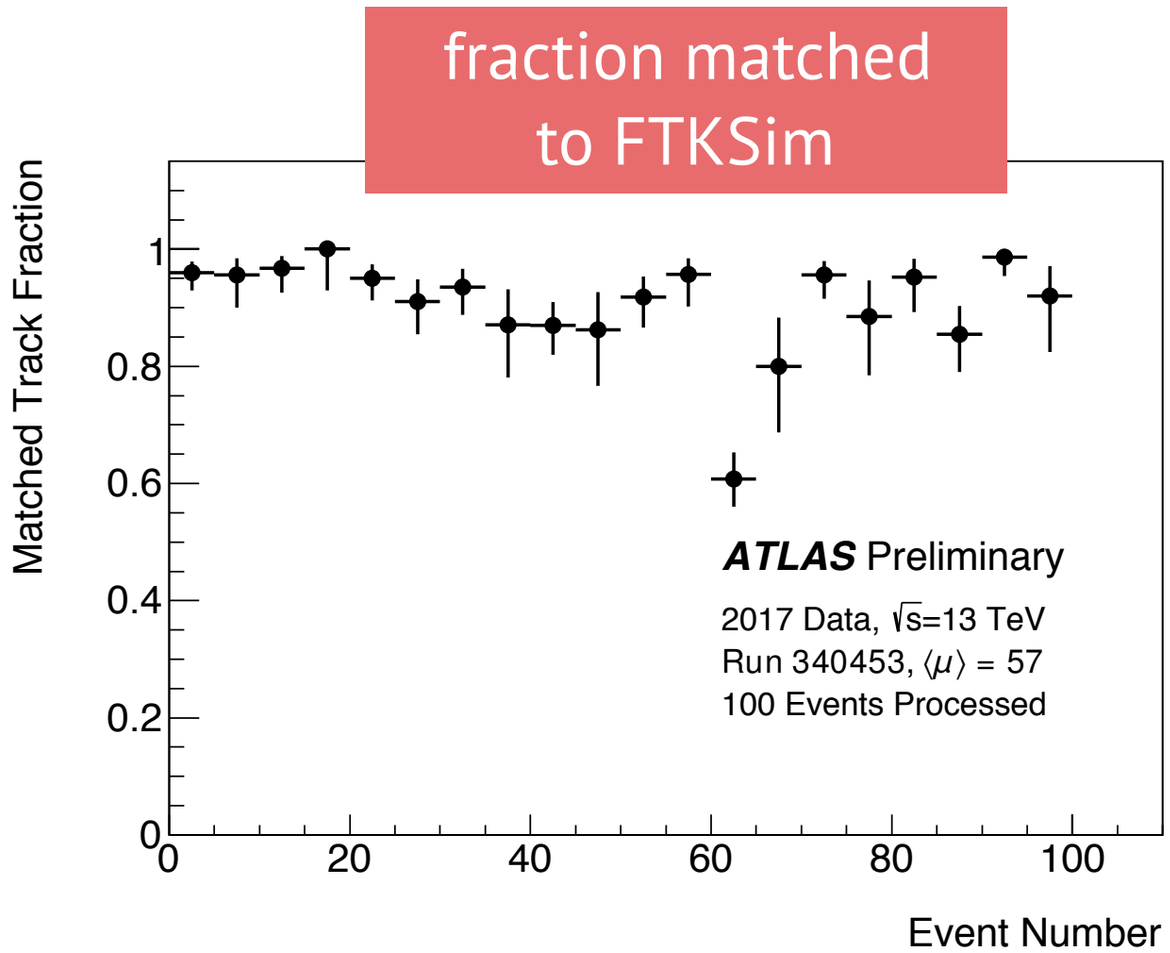
raw hits
clusters
8-layer tracks
12-layer tracks

FTK COMMISSIONING

FTK Status at the end of 2017

- Commissioning with vertical slices (dataflow through 1/64 towers)
- At the end of 2017, a vertical slice outputting 8-layer tracks was incorporated into ATLAS
  - validated 8-layer data output with FTK functional simulation





FTK Today

- The full system is nearly ready to be installed in the ATLAS detector
  - 165/194 boards in FTK racks
- FTK vertical slices join in ATLAS runs and output 8- and 12-layer tracks
- The system works with ideal inputs
  - Working on improving stability, handling real data-taking conditions in ATLAS

Going forward

- By the end of this year we aim to run stably with 50% coverage of the ATLAS detector
  - The remaining 50% will be commissioned during the long shutdown
- When LHC starts again (2021) FTK will provide the HLT full-scan tracking for all events
  - Essential for ATLAS to continue providing low-threshold triggers at high pileup
- Daily progress as we commission with beam!