EPS HEP 2015, Vienna

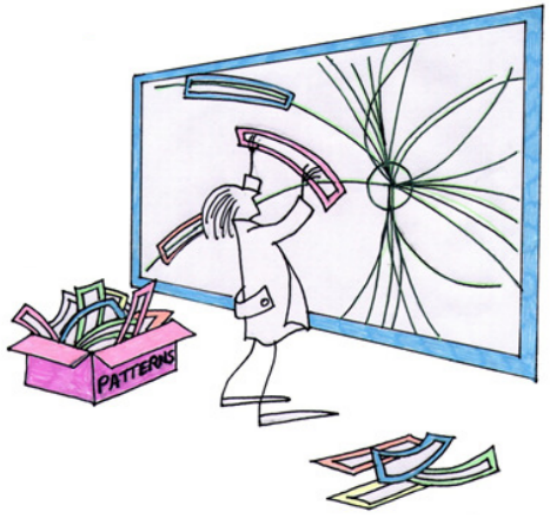
Hardware-based tracking at trigger level for ATLAS:

The Fast TracKer (FTK) Project

Johanna Gramling
for the ATLAS Collaboration

24.07.2015

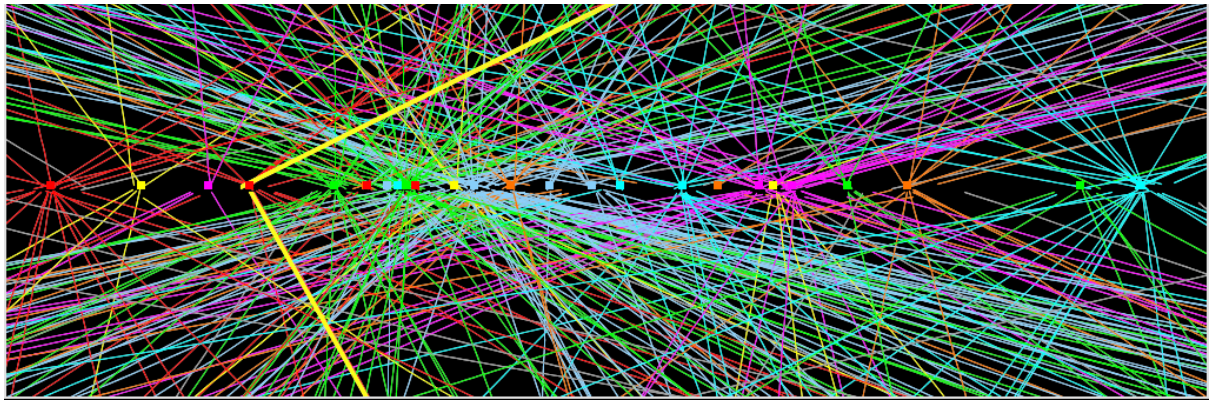
ATLAS-TDR-021



johanna.gramling@cern.ch

Why?

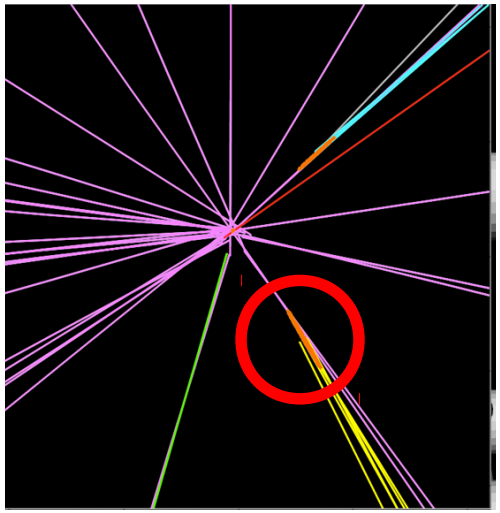
- Run II of LHC started - 13 TeV collisions!
  - With more than twice the luminosity compared to Run1, average of 40-50 collisions per bunch crossing expected
- Tracking at trigger level is essential to control rates while maintaining good efficiency for relevant physics processes – especially with high pile-up!
  - Measuring number and position of vertices improve robustness especially of jet and missing E_T triggers with changing pile-up conditions



8 TeV Z → μμ event with 25 reconstructed collision vertices (seen by the Inner Detector)

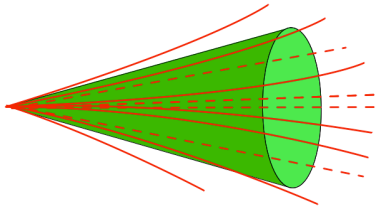
Why?

- Higgs found!
  - Fermionic-coupling measurements important to pin down (SM-like?) Higgs properties
  - Channels involving b and τ final states crucial, but:
    - Large QCD backgrounds
    - Low trigger efficiencies

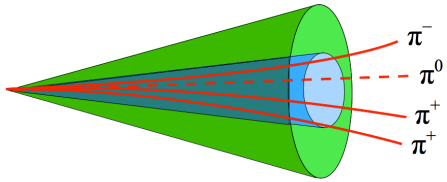


B-jets:
Displaced vertex

- SUSY could be hiding with light sbottom, stops, staus
  - Would lead to final states with moderate-momentum b's and τ's
    → difficult to trigger and select



"typical" jet



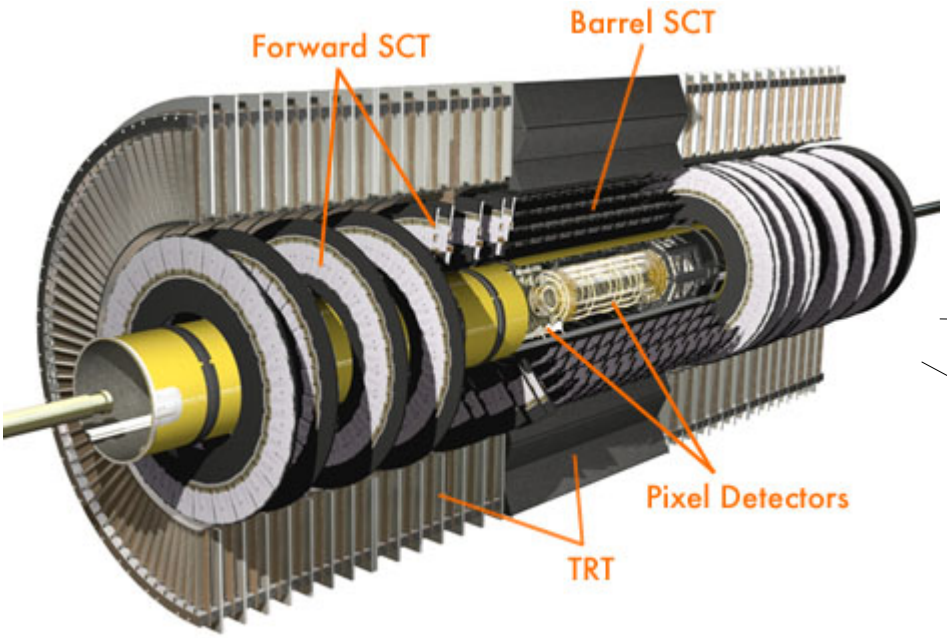
τ jet:
fewer tracks in smaller cone

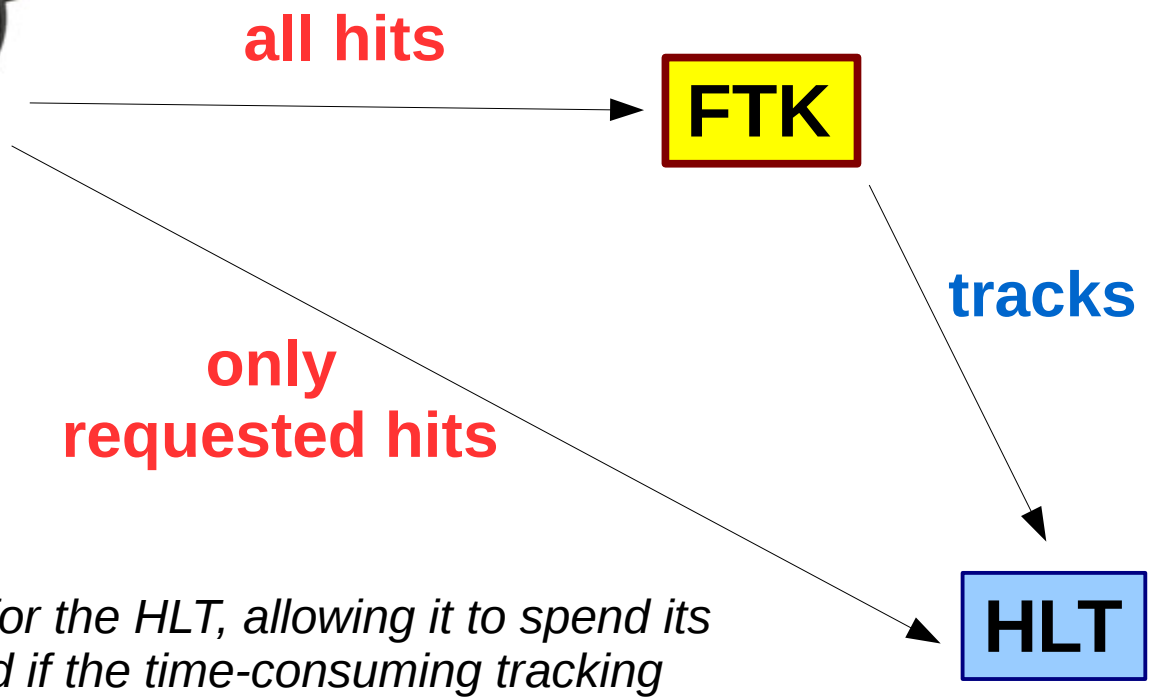
→ **Tracking and hence resolving topology of b- and τ jets at trigger level crucial for such analyses!**

Main Concepts of FTK

- Tracking at trigger level
  - For every event passing the Level-1 trigger, FTK receives data from the 98 million channels of the silicon detectors and provides tracking information to high level trigger (HLT)

More details on ATLAS trigger system:
see talk from Patrick CZODROWSKI

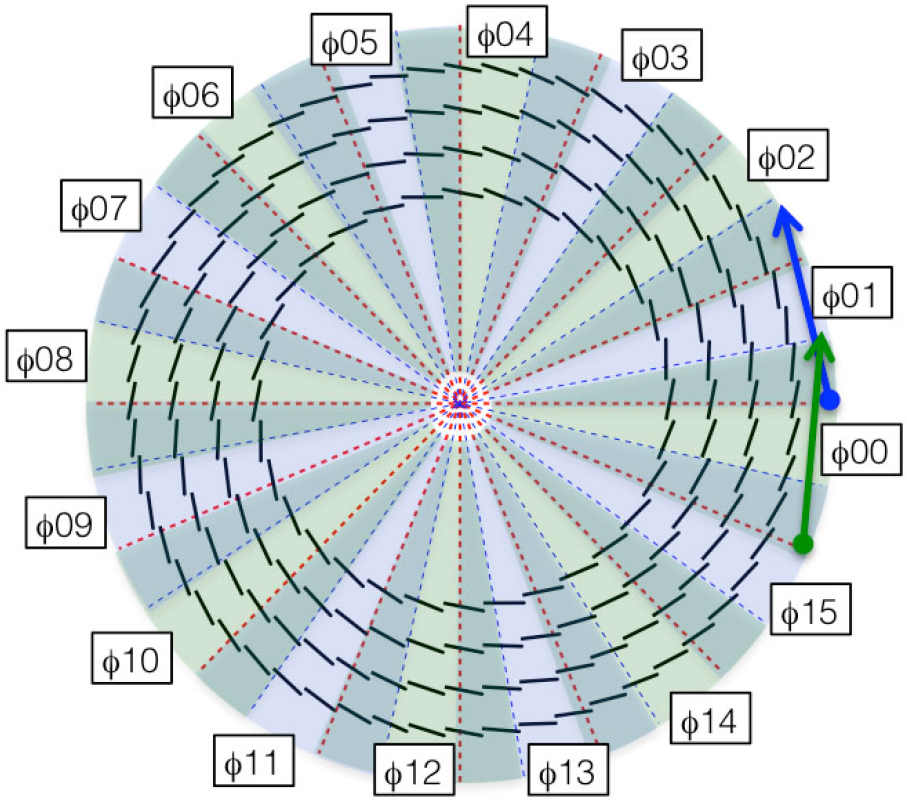


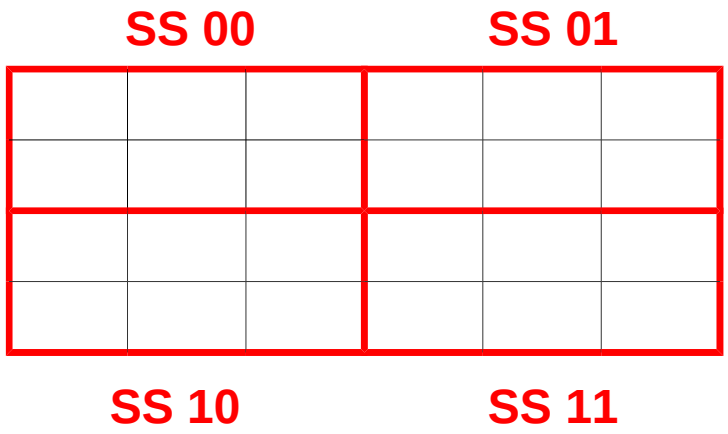


The FTK does the tracking for the HLT, allowing it to spend its time on algorithms instead if the time-consuming tracking

Main Concepts of FTK

- Tracking has to be very fast - maximum rate of 100 kHz
  - **Process in parallel:** decompose detector data into independent regions (64 ***Towers***)
  - **Data reduction:** each cluster of adjacent pixels/strips defines one "hit"
    - When appropriate, re-bin hit information to coarser resolution → ***SuperStrips***

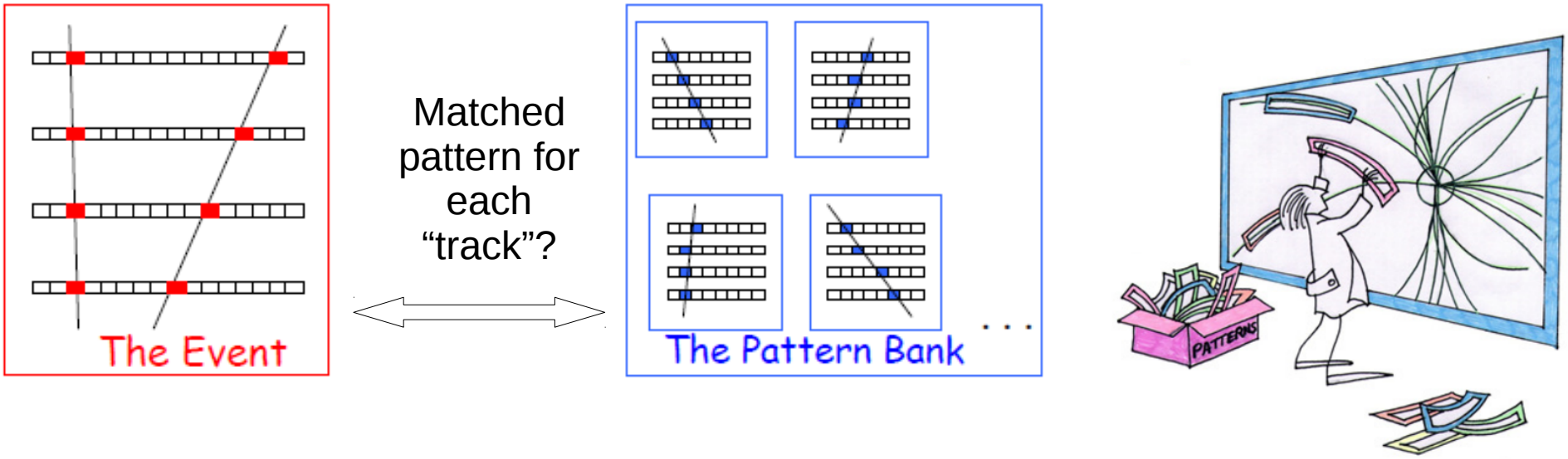




Several pixels are grouped together into one SuperStrip (SS)

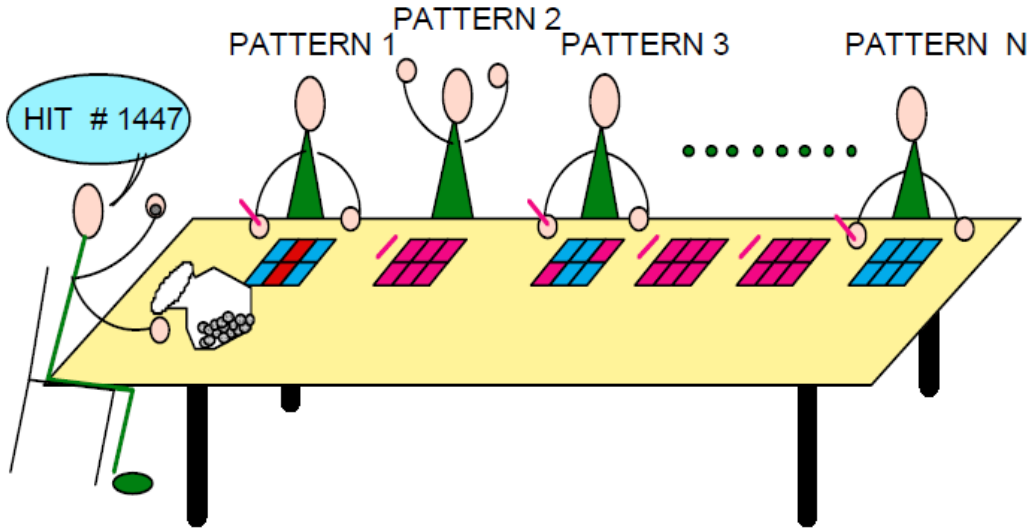
Main Concepts of FTK

- Tracking has to be very fast - maximum rate of 100 kHz
  - **Process in parallel:** Reorganize the received hits
  - **Data reduction:** each cluster of adjacent pixels/strips defines one "hit"
  - Perform **tracking in 2 steps**
    1. Find track candidates: ***Roads***
       - Compare fired SuperStrips to predefined track trajectories: ***Pattern matching***
       - Minimum number of matched layers programmable



Pattern Matching in detail

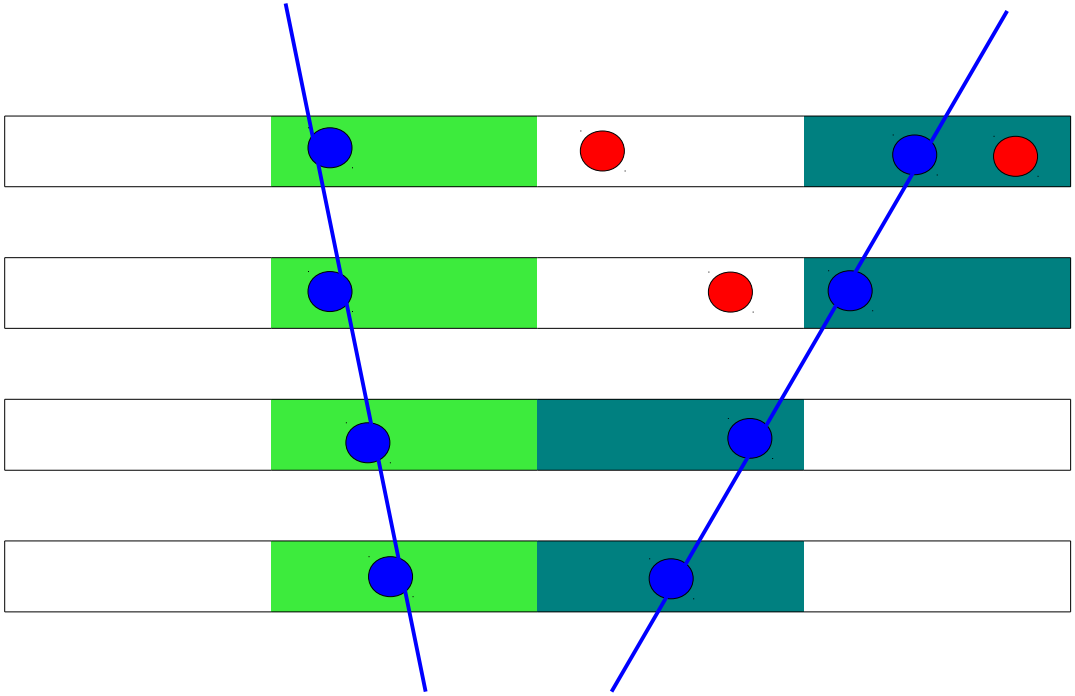
- Pattern matching is performed in customly-designed chip using Associative Memory (AM)
  - Like playing Bingo
  - Incoming data is **simultaneously** compared to **all** stored patterns
    → very fast matching!



More details on pattern matching:
see talk by *Francesco CRESCIOLI*

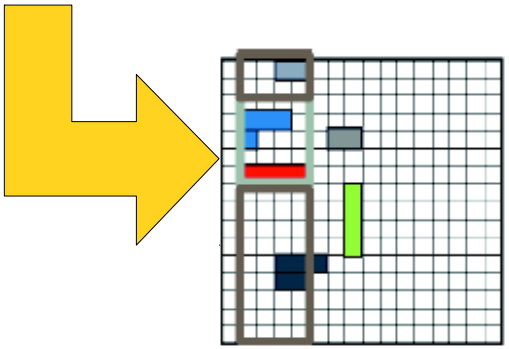
Main Concepts of FTK

- Tracking has to be very fast - maximum rate of 100 kHz
  - **Process in parallel:** Reorganize the received hits
  - **Data reduction:** each cluster of adjacent pixels/strips defines one "hit"
  - Perform **tracking in 2 steps**
    1. Find track candidates: ***Roads***
    2. Perform full-resolution ***track fitting*** INSIDE roads
       - Combinatorics reduced → faster!



Data Flow

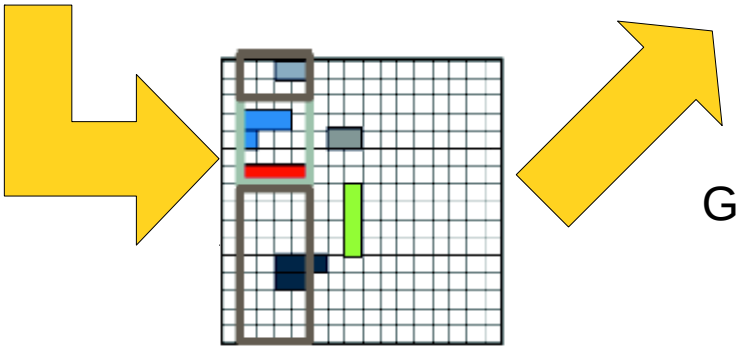
Tracking detectors



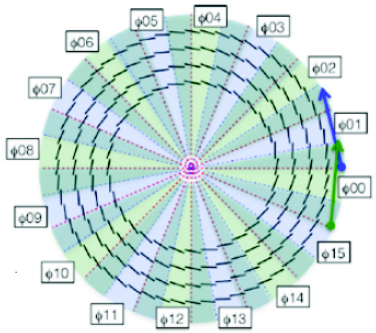
Custom pixel clustering algorithm on FPGAs

Data Flow

Tracking detectors



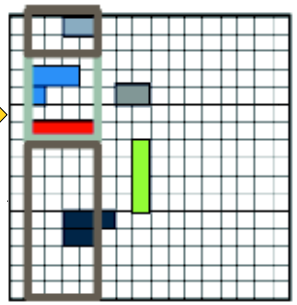
Custom pixel clustering algorithm on FPGAs



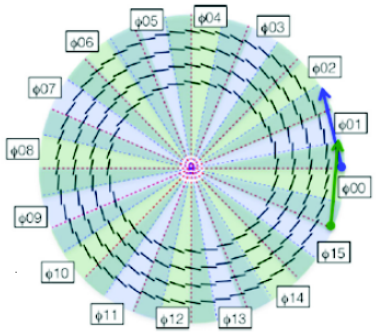
Geometrically grouped data distributed to processing units

Data Flow

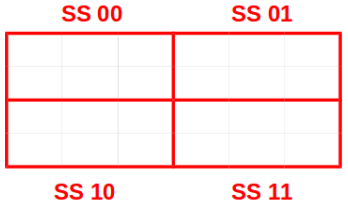
Tracking detectors



Custom pixel clustering algorithm on FPGAs



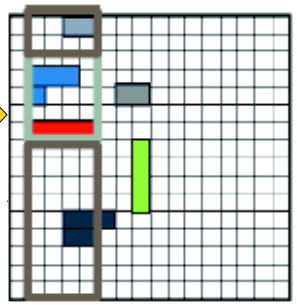
Geometrically grouped data distributed to processing units



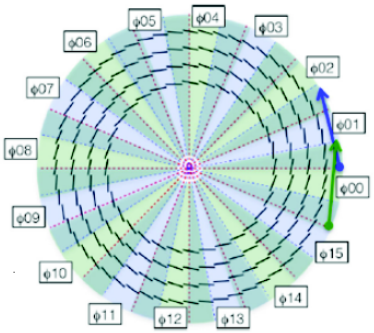
Transformation to coarse-resolution hits

Data Flow

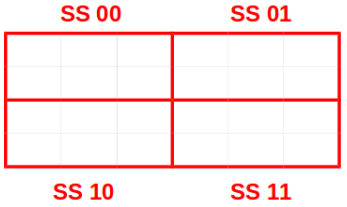
Tracking detectors



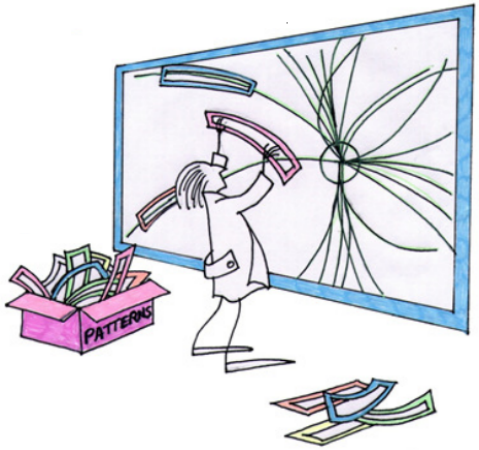
Custom pixel clustering algorithm on FPGAs



Geometrically grouped data distributed to processing units



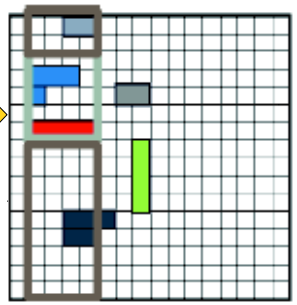
Transformation to coarse-resolution hits



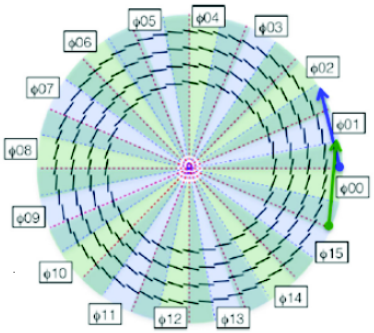
Comparison to reference track patterns at coarse resolution

Data Flow

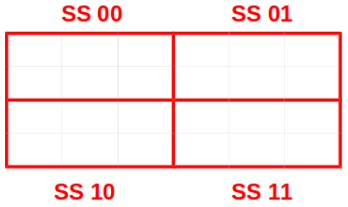
Tracking detectors



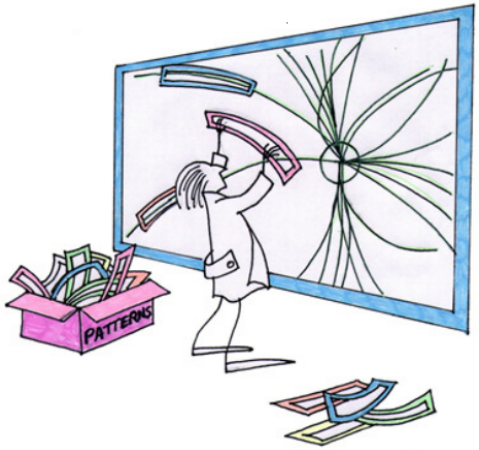
Custom pixel clustering algorithm on FPGAs



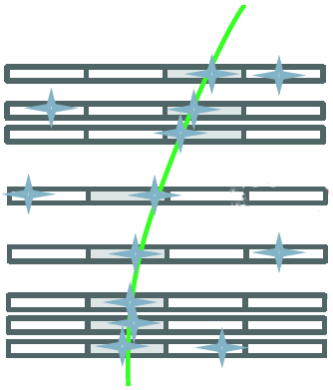
Geometrically grouped data distributed to processing units



Transformation to coarse-resolution hits



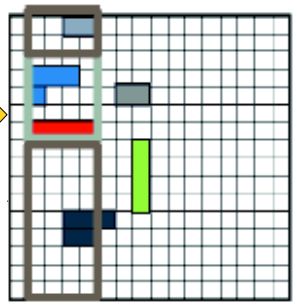
Comparison to reference track patterns at coarse resolution



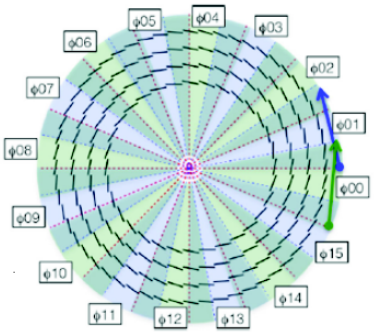
Track fit with full-resolution info for track candidates

Data Flow

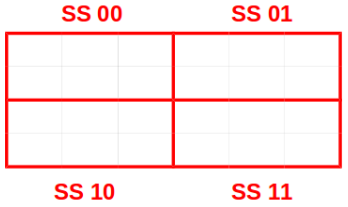
Tracking detectors



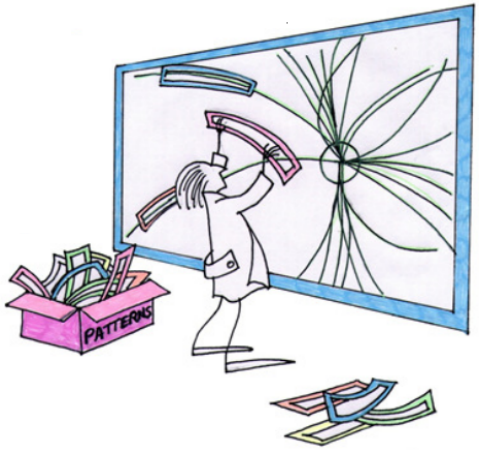
Custom pixel clustering algorithm on FPGAs



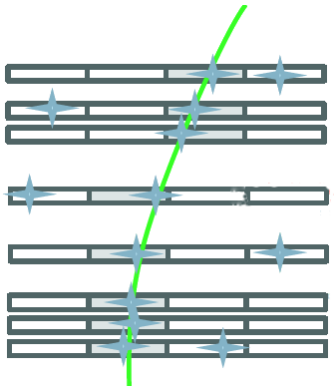
Geometrically grouped data distributed to processing units



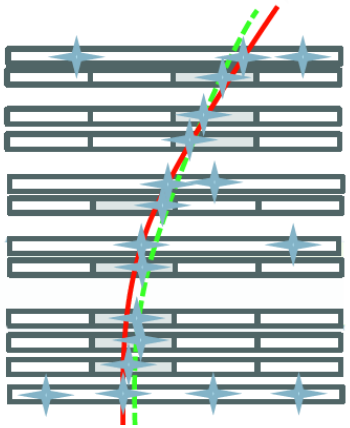
Transformation to coarse-resolution hits



Comparison to reference track patterns at coarse resolution



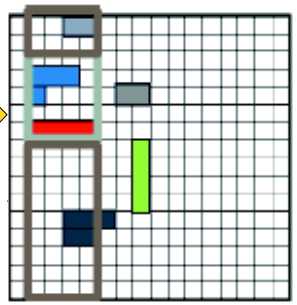
Track fit with full-resolution info for track candidates



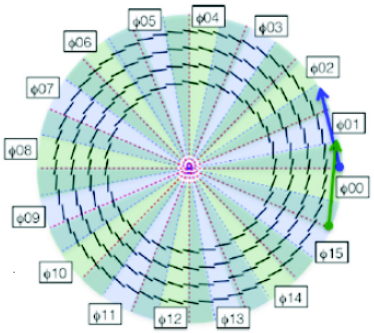
Precise fitting with good tracks being extrapolated to missing layers

Data Flow

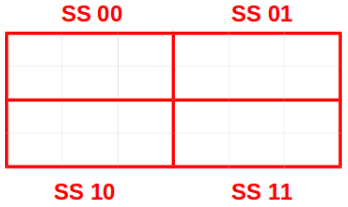
Tracking detectors



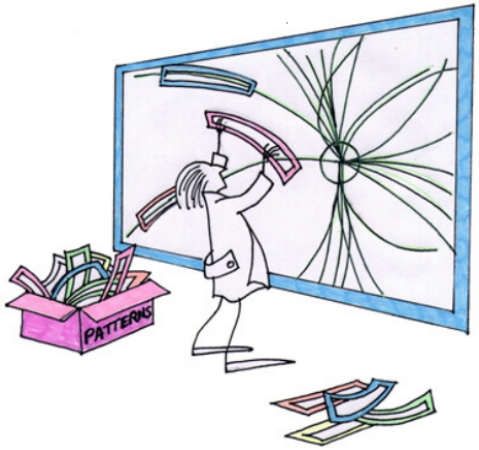
Custom pixel clustering algorithm on FPGAs



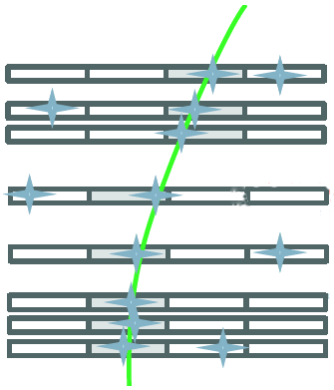
Geometrically grouped data distributed to processing units



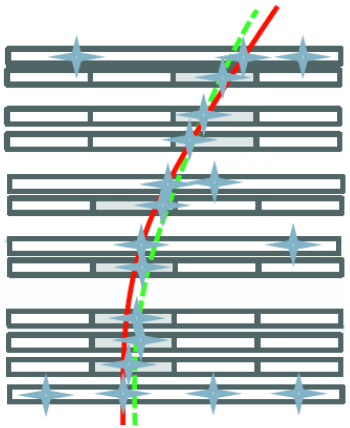
Transformation to coarse-resolution hits



Comparison to reference track patterns at coarse resolution



Track fit with full-resolution info for track candidates



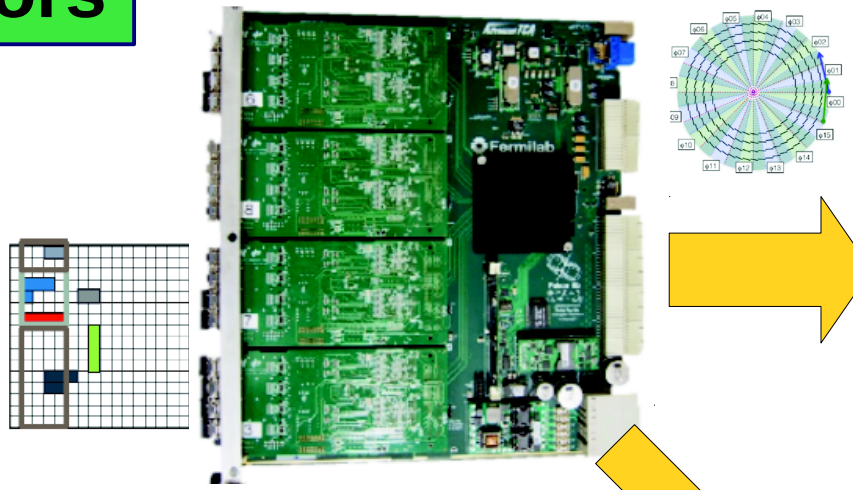
Precise fitting with good tracks being extrapolated to missing layers

HLT

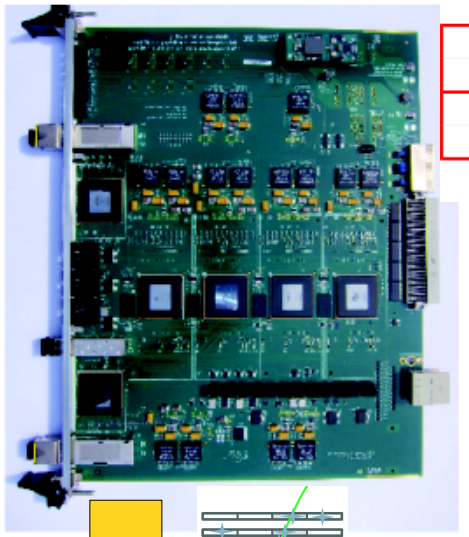
Data Flow – in hardware components

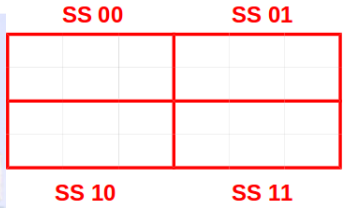
Tracking detectors

32x Input Mezzanine (IM) and Data Formatter (DF)

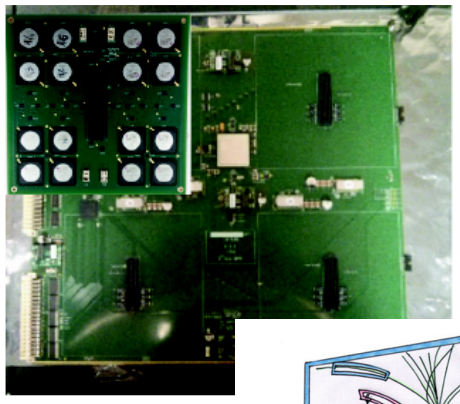


128x Auxiliary Boards (AUX)

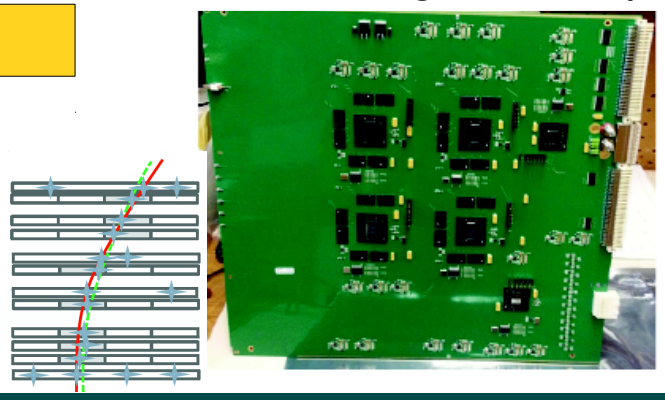




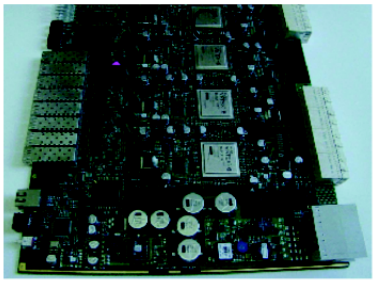
128x Associative Memory Boards (AMB)



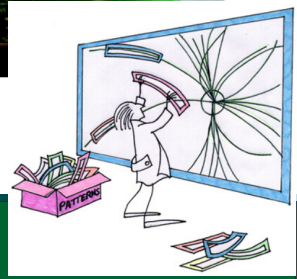
32x Second Stage Board (SSB)



2x Level-2 Interface Crate (FLIC)

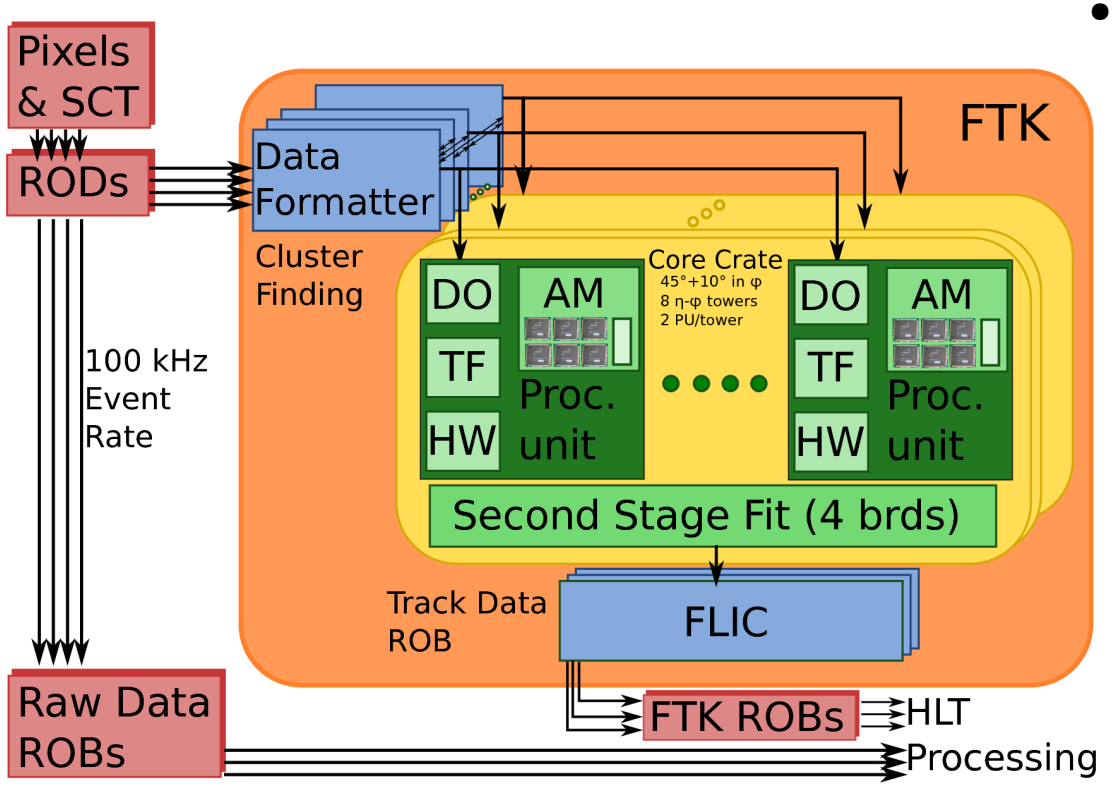


HLT



Data Flow

FTK = (very complex) custom parallel supercomputer

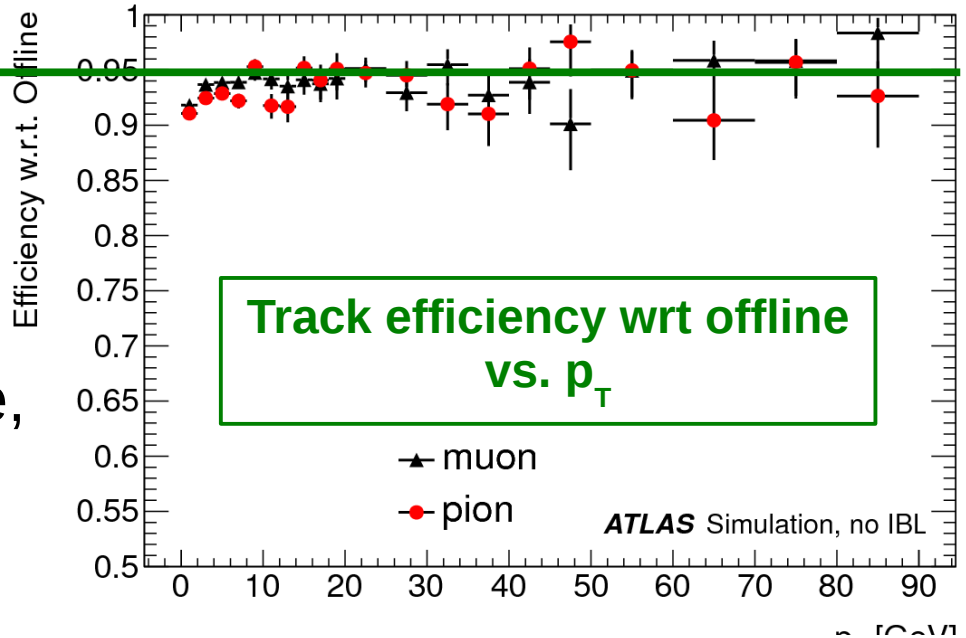


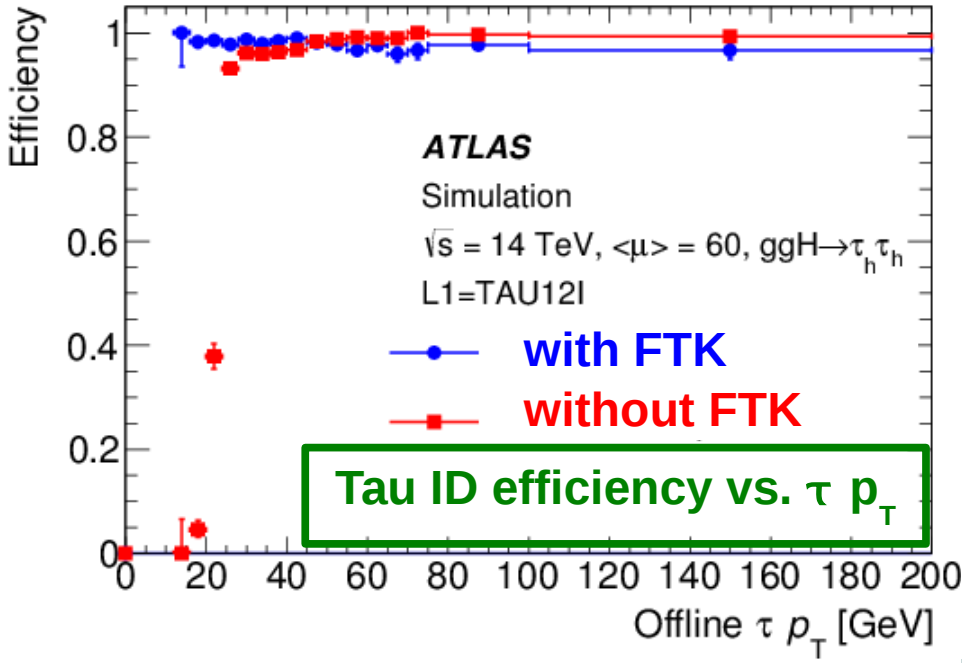
- FTK is a big system
  - 8 full 9U VME crates
  - 5 ATCA shelves
- Many different boards
  - 2 types of ATCA boards
    - Data Formatter (DF), FTK-to-Level-2 Interface Crate (FLIC)
    - mezzanine for clustering (input mezzanine)
  - 9U Auxiliary board (AUX)
    - data organizer, track fitting and fake reduction functions
  - 2 types of 9U VME boards
    - Associative Memory Board (AMB), Second Stage Board (SSB)
    - 9U/4 mezzanine for Associative Memory chip (AMchip06)

~2000 FPGAs and ~8000 custom AM chips

Expected Performance

- FTK online track quality comparable to offline **95%**
  - Slightly worse resolution
  - Small effect from pile-up
- b-tagging efficiency similar to offline, with high light-jet rejection when using FTK
- Linear correspondence between number of FTK and offline vertices
  - Independent of pile-up
- Trigger efficiency for 1-prong τ's in $H \rightarrow \tau\tau$ significantly improved when optimizing algorithms including FTK tracks





How to profit from FTK?

- **Jet Trigger**
  - Profit from precise pile-up information
    - energy corrections based on number of vertices found by FTK
    - Better jet quality cuts to reduce pile-up effects: especially important for multijets or b-jets (room for improvement for b-tagging!)
  - Use FTK tracks to recover efficiency for low-p_T or close-by jets
- **Muon/Electron Trigger**
  - Pile-up dependence of calorimeter-isolation and electron shower shapes
  - Track-based isolation for muons
  - Recover di-muon events via high-p_T FTK tracks
- **Missing E_T Trigger**
  - Improve MET resolution by track/pile-up info from FTK
- **Tau Trigger**
  - Refine algorithms by including FTK tracks
  - Improved calorimeter shower shapes due o precise pile-up inforation
  - Recover L1 inefficiencies via high-p_T FTK tracks

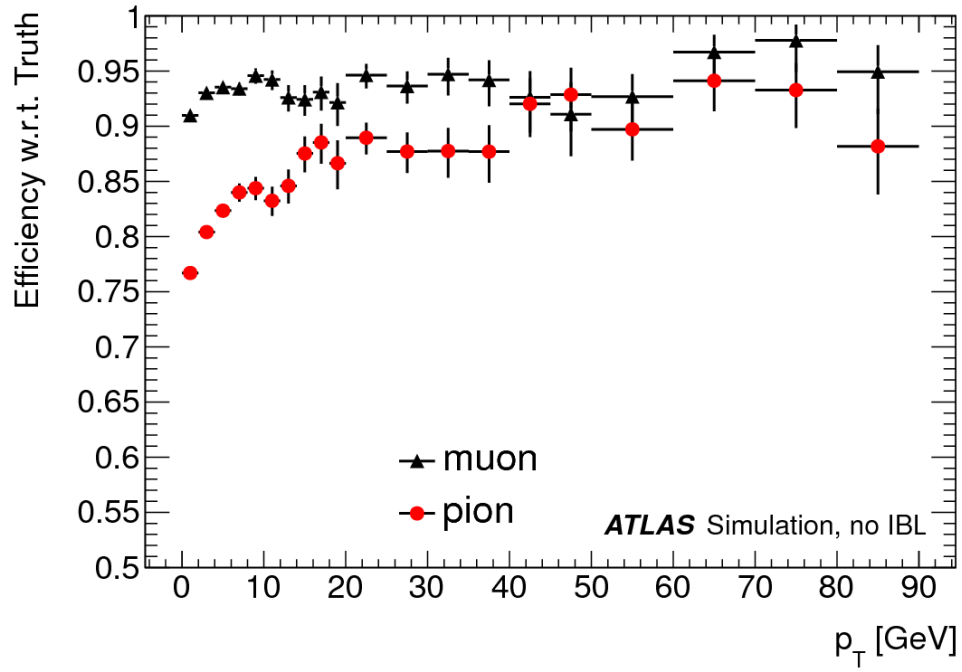
SUMMARY

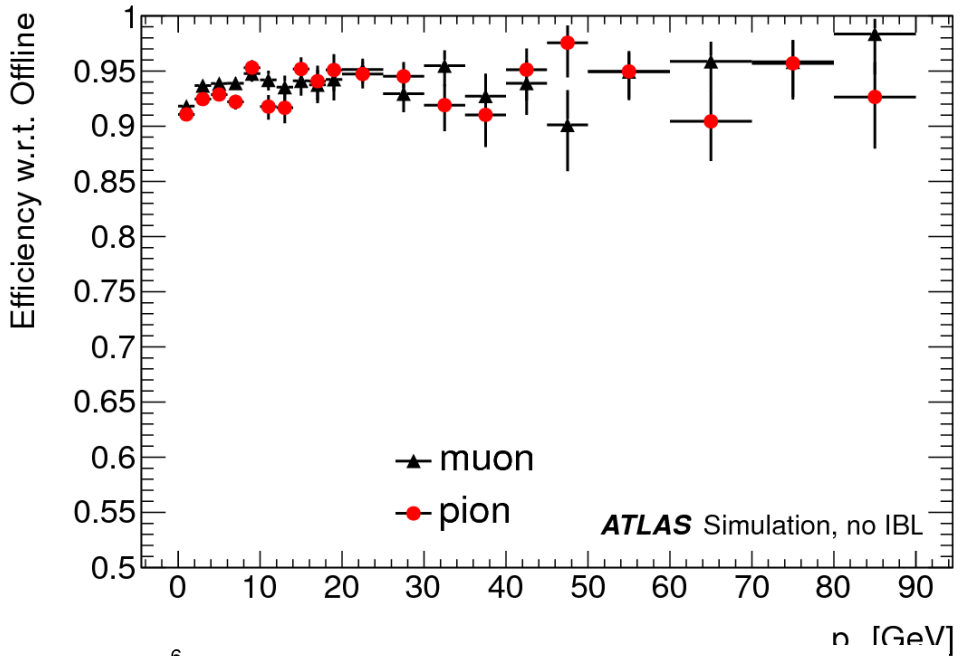
- **FTK aims at providing tracking information at trigger level**
  - Especially beneficial for signatures relying on medium-p_T b's or τ's
  - Constraints on pile-up improve many trigger selections
- **System optimized to achieve rates of 100 kHz**
  - Parallel processing of event data
  - 2-stage tracking: pattern matching to identify track candidates
- **Pattern matching done with custom AM chips**

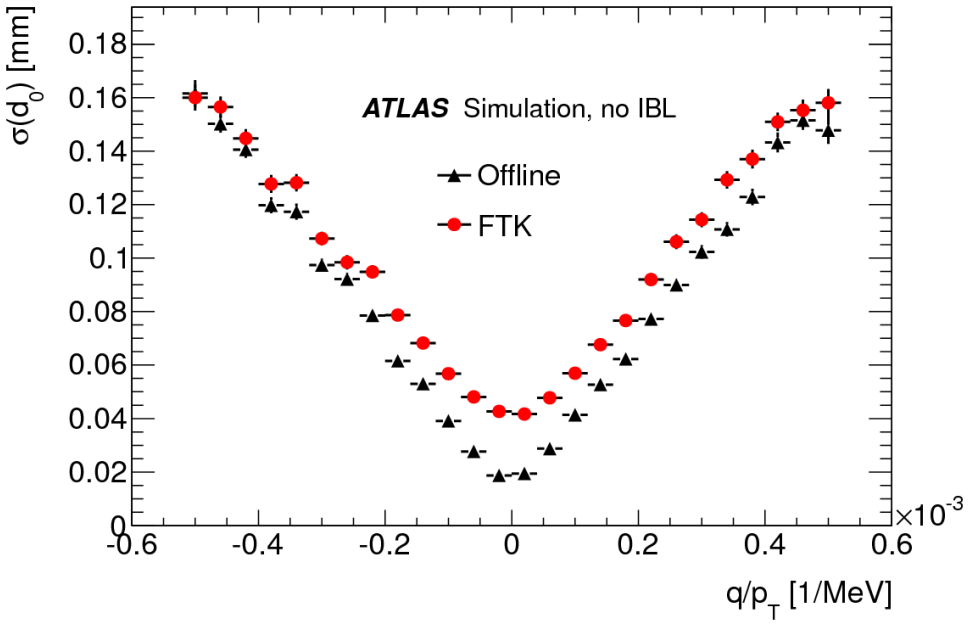
  → *Learn more in next talk by Francesco Crescioli*
- **FTK track quality and efficiency close to offline performance**
- **Integration of first boards ongoing**
  - Barrel-only FTK processing expected in spring 2016, then HLT integration and extension to full coverage will follow
- **HL-LHC: possibly extend and refine FTK concept to cope with high-luminosity conditions**
  - Tracking at Level-1 using an AM-based system with upgraded AM chip

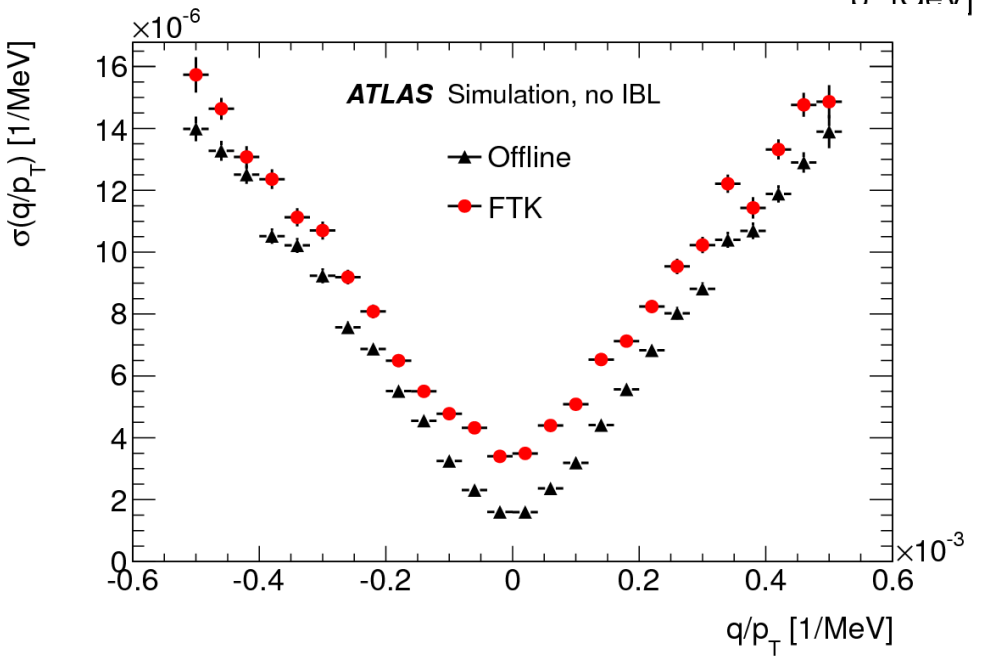
BACKUP

Expected Performance



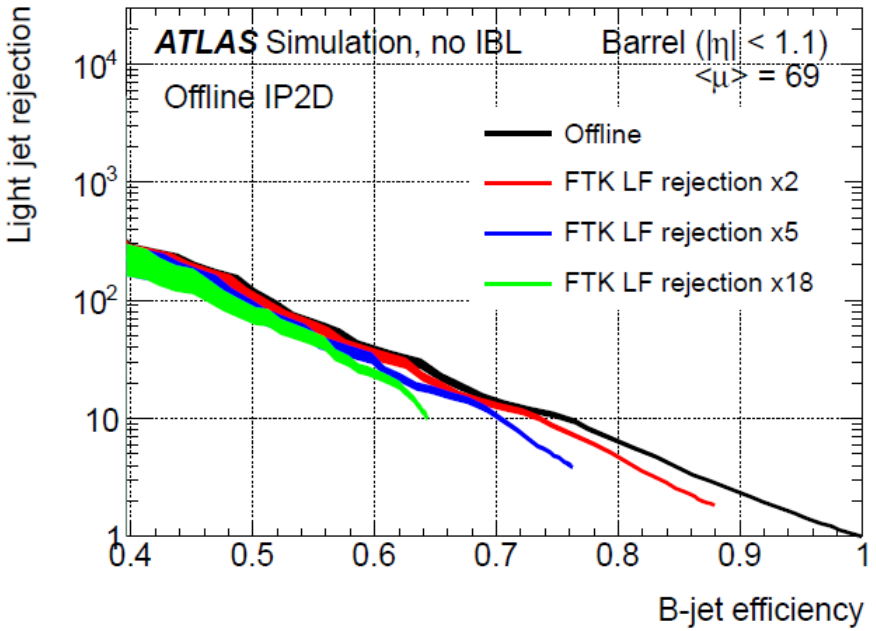


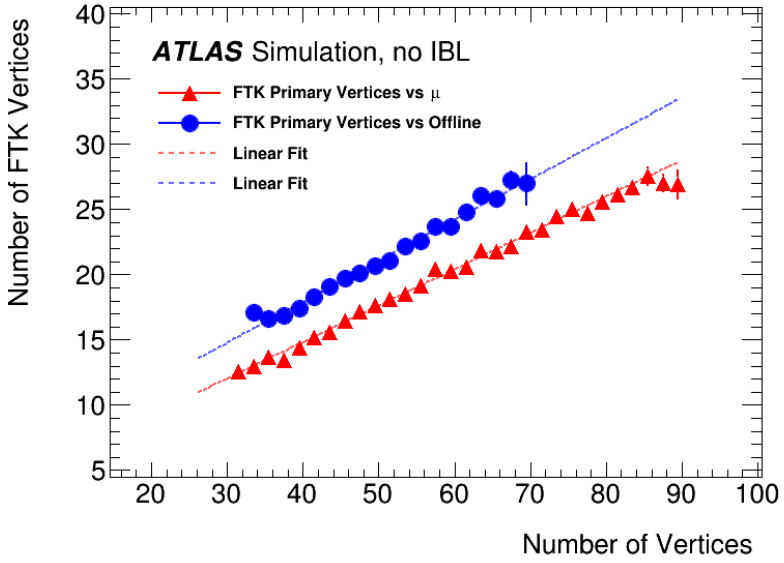


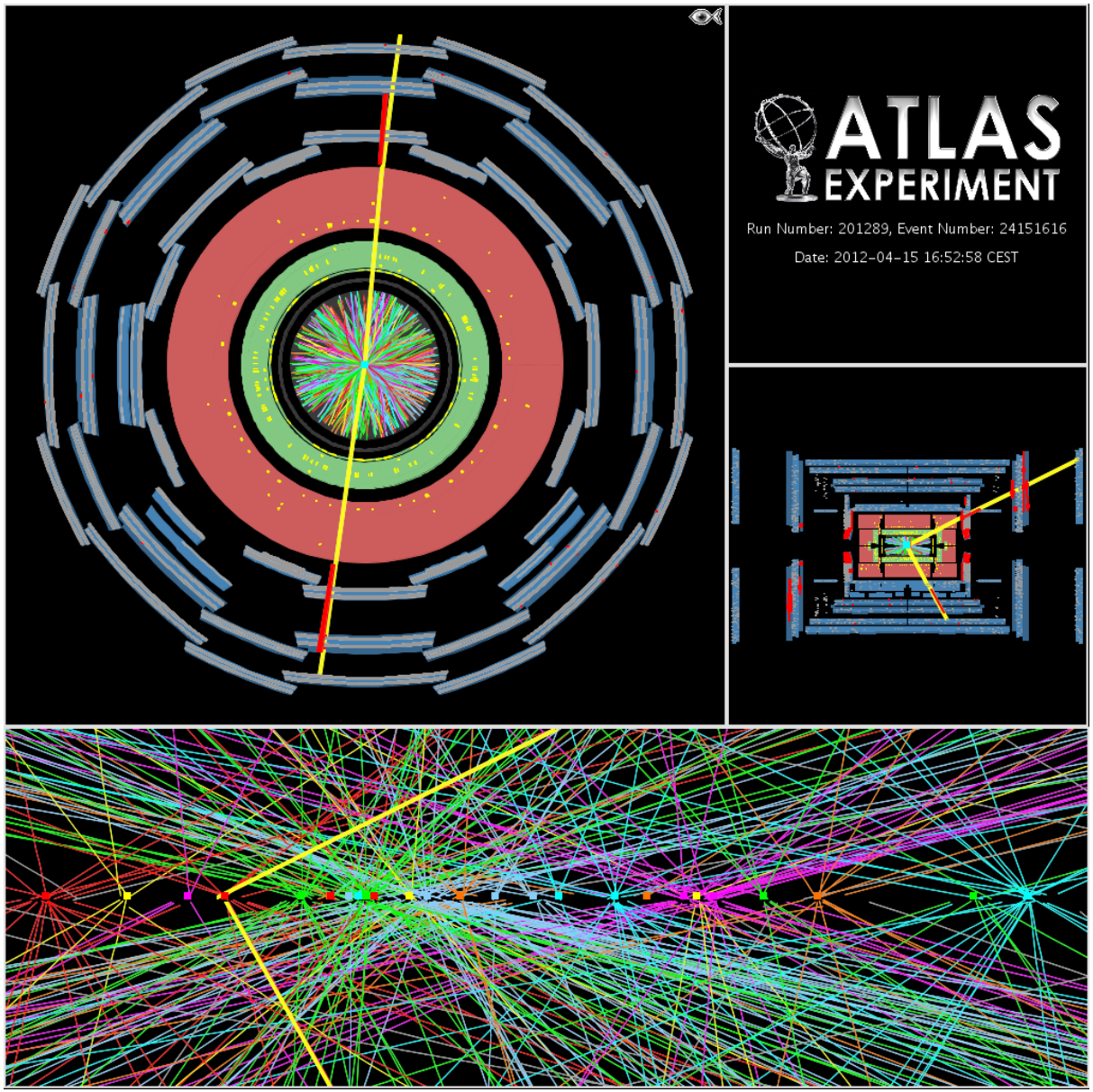


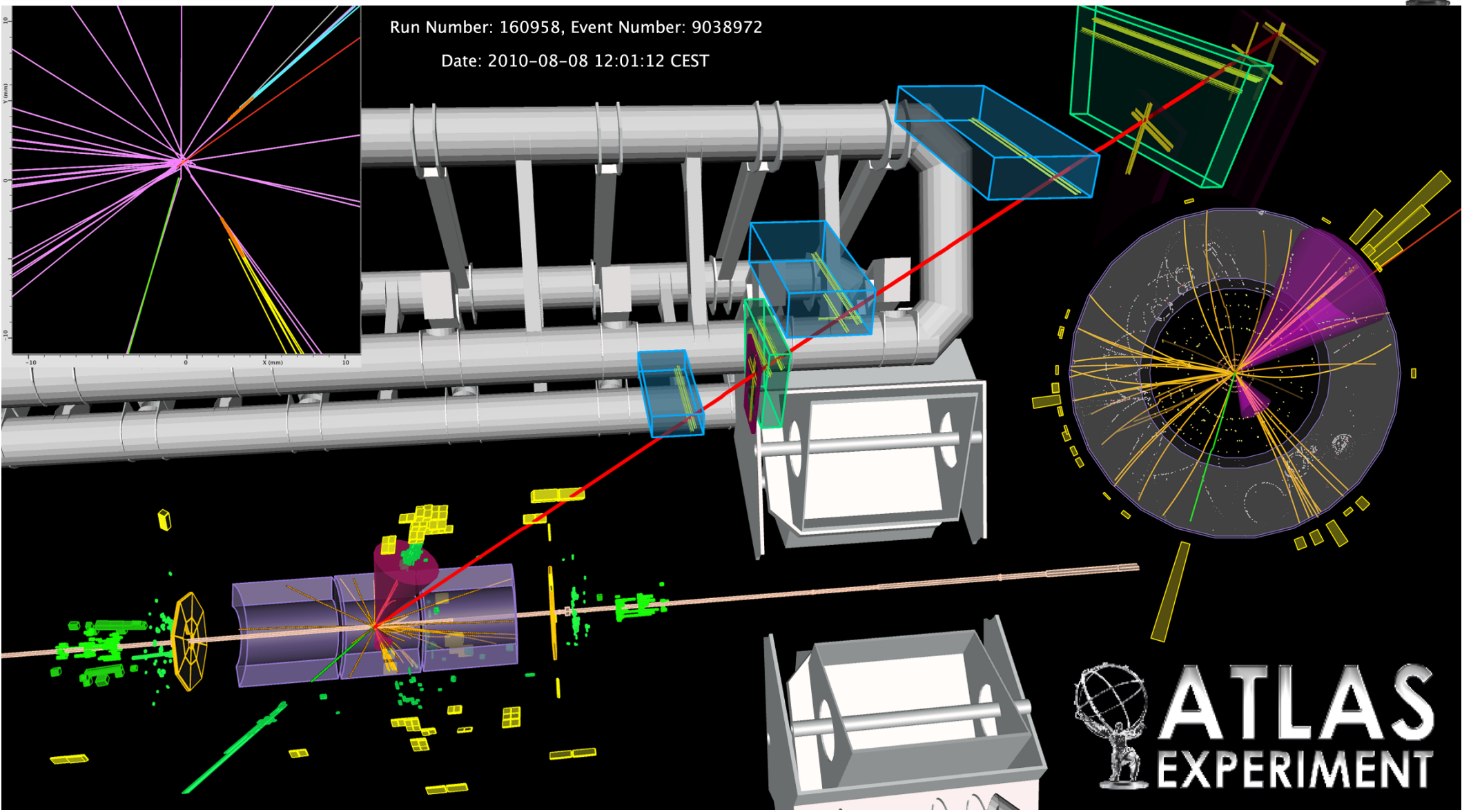
Vertexing, b-tagging

- Similar b-tagging efficiency with high online FTK light-jet rejection
- Linear correspondence between number of FTK and offline vertices
  - Independent of pile-up






ATLAS
EXPERIMENT
Run Number: 201289, Event Number: 24151616
Date: 2012-04-15 16:52:58 CEST


Run Number: 160958, Event Number: 9038972
Date: 2010-08-08 12:01:12 CEST
ATLAS
EXPERIMENT