L3 Muon Reconstruction in CMS: a Status Report

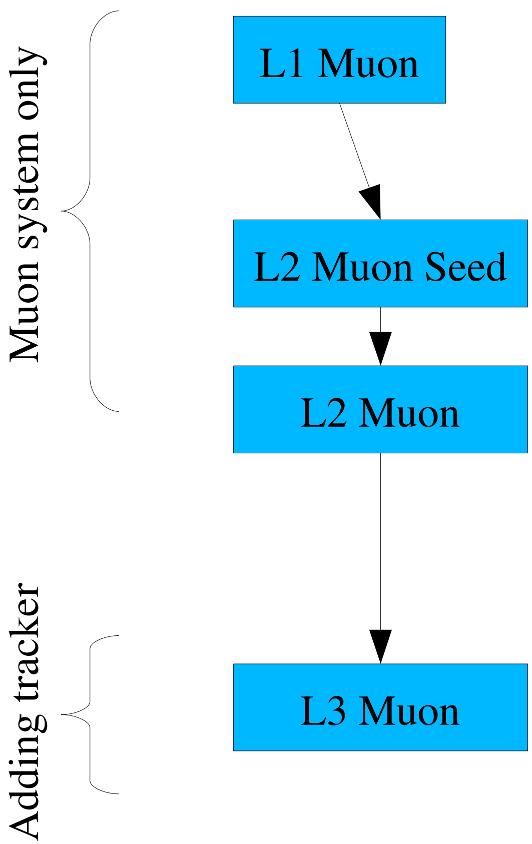
J.-R. Vlimant, on behalf of the μHLT team

Outline

- Overview of the muon HLT: focusing on L3 muon reconstruction
  - ➔ Regional seeding in the tracker
  - ➔ Pattern recognition in the tracker
  - ➔ Global matching and global refit
- Lots of improvement in the recent months
  - ➔ Overview and prospectives
- Global matching issue in tau→3μ
  - ➔ Will talk about that

Code Architecture
A snapshot of the "past"

- Code as it is, up until 180 (included)



"L3 Muon" is just one module that does it all

- Seeding in the tracker
- Pattern recognition in the tracker
- Arbitration of trajectories
- Global refitting

Issues

- No intermediate products
- Maintenance is not easy
- Does not automatically inherit from improvement in the tracker code
- One set of tracks: hltL3Muons
  no by products without arbitration/global refit

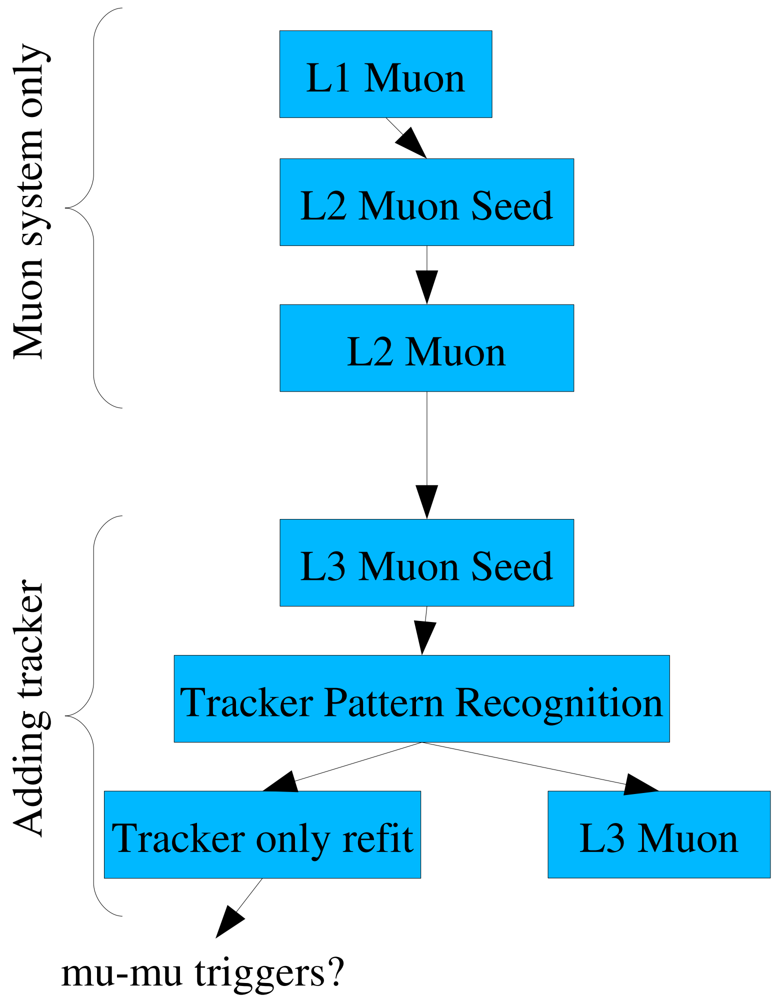
Not talking about muon HLT isolation in this presentation

Code Architecture

A snapshot of the present

- Code as it is, in 200

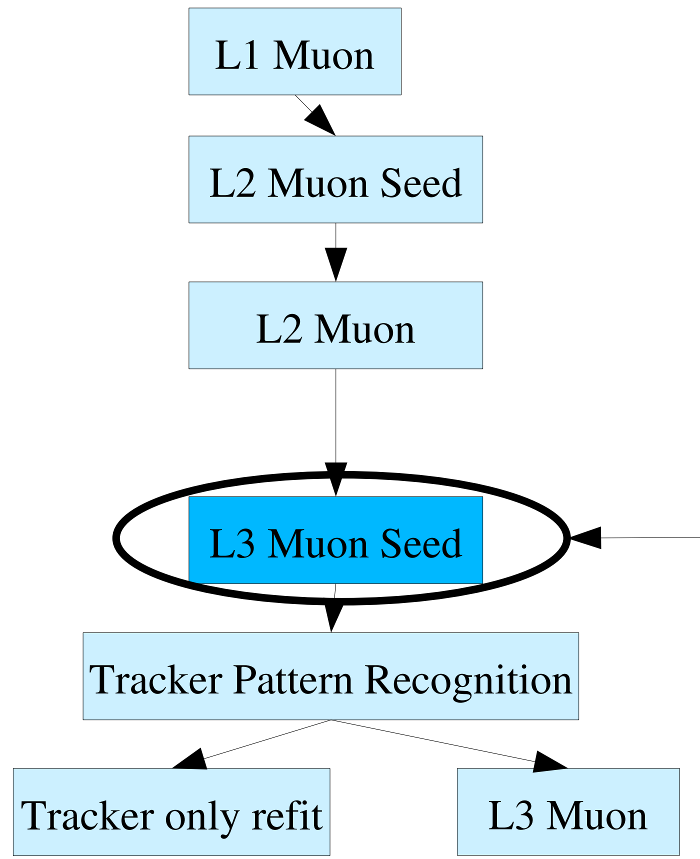
http://indico.cern.ch/conferenceDisplay.py?confId=29596 (A. Everett)



Issues **solved**

- No intermediate products: **solved**
  - TrajectorySeed
  - TrackCandidate (tracker only, no arbitration)
- Maintenance is not easy: **solved**
  - Simple module for each step
- Does not automatically inherit from improvement in the tracker code: **solved**
  - Use plain RecoTracker modules
- One set of tracks: hltL3Muons
  no by products without arbitration/global refit: **solved**
  - see above

Same **functionalities**, but in simple separate modules.
Next slides: describe functionalities



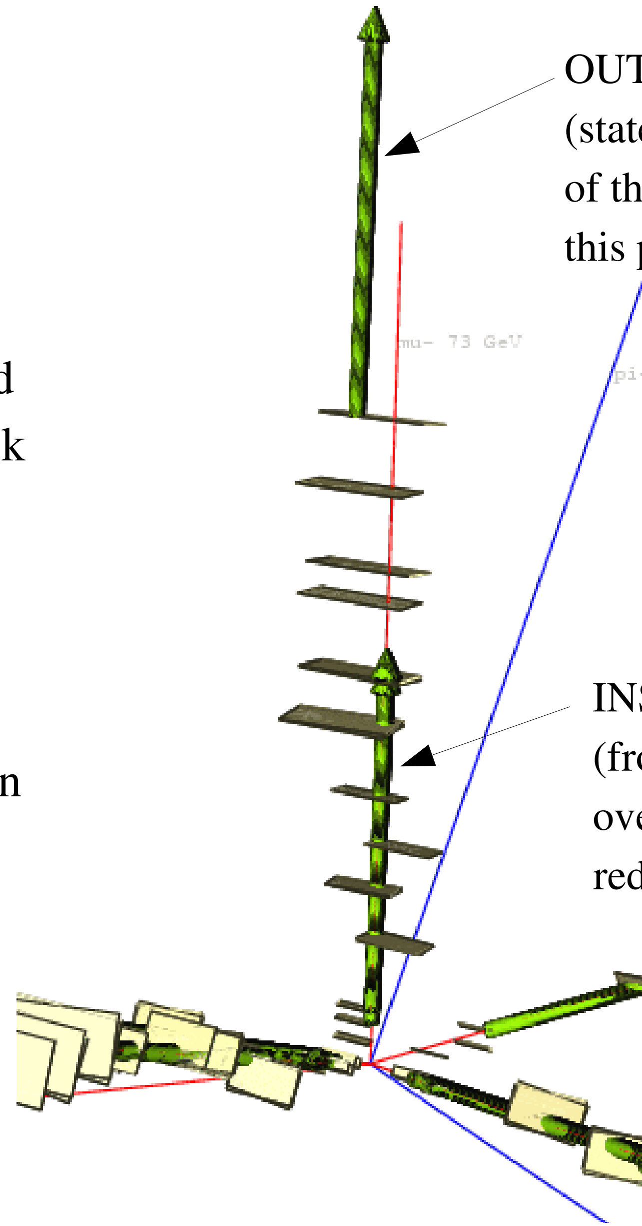
Next slides:

Seeding in the tracker

Tracker Trajectory Seeding

Description valid for all versions of CMSSW

- Two major **type** of seeds
  - Hit-based:
    - Pair/triplet of rechits in a region of the tracker
  - State-based:
    - In L3 reconstruction, there is already a good estimate of the track state: the L2Muon track
    - Use this state, propagated to a tracker layer as a seed
- Two major **method** of seeding/pattern recognition
  - Inside-out:
    - Uses inner layers of SiPixels or SiStrip
  - Outside-in:
    - Uses outer layers of SiStrip tracker



Tracker Trajectory **Hit-based** Seeding

Description valid for all versions of CMSSW

Inside-out

- Define a region around L2 track
  - Definition has improved
- Combinatorial pair/triplet seed generator
  - Commonly used in CMSSW

➔ Requires seeding cleaning
  - Used to give too many seeds
  - Tracker seed cleaning inherited with modularization
  - Recent improvements to be validated

Outside-in

- Propagate the L2 track to outer surface
  - Significantly rescale the error matrix
- Look for compatible hits
  - Equivalent to the first step of pattern recognition
- Update state with found hits

➔ Recent improvements to be validated

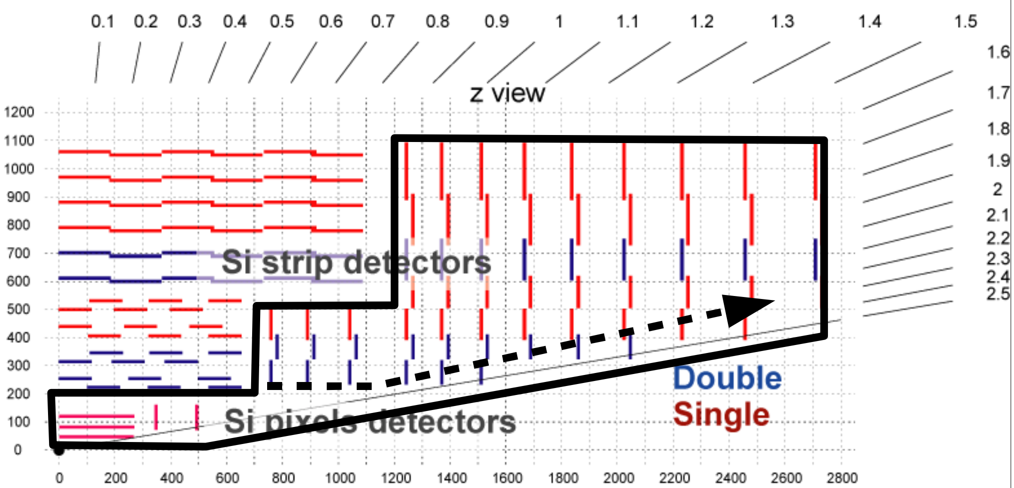
Tracker Trajectory **State-based** Seeding

Description valid for all versions of CMSSW

http://indico.cern.ch/conferenceDisplay.py?confId=17939 (J-R Vlimant)

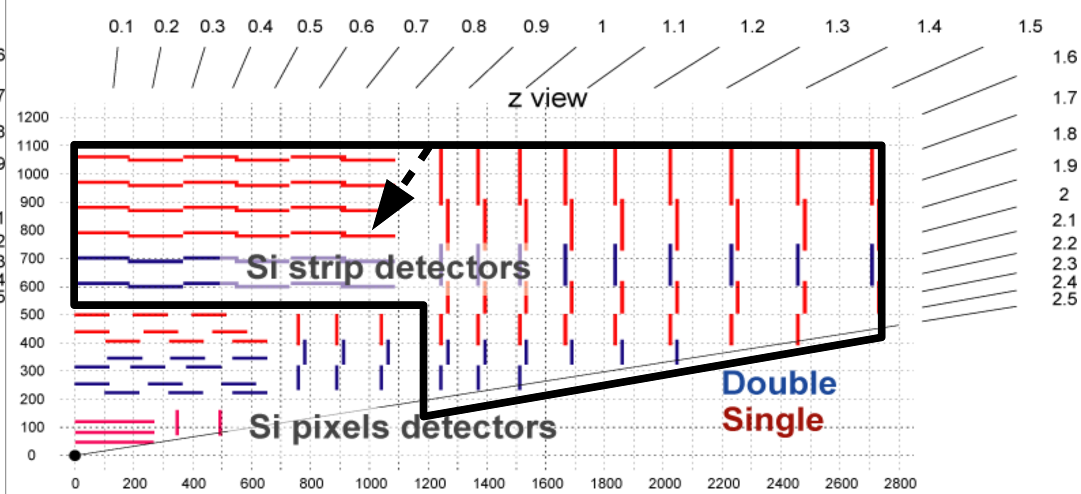
Inside-out

- Propagate the state to the first layer/disk of pixel it can find
- Go to forward SiStrip disk if no state is found



Outside-in

- Propagate the L2 to the tracker envelop
  - r=1.15 m or |z|=2.8 m
- State on the outer barrel or forward disk depending on state location
  - Specific case for the |eta|~1 "gap": state on barrel layer 5,4,3,...



Tracker Trajectory Seeding

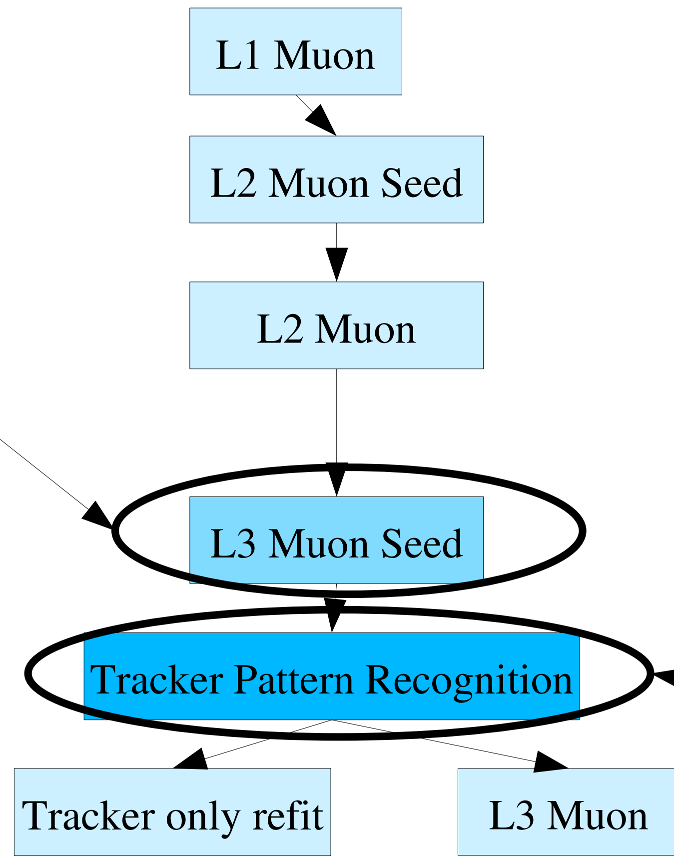
- All four seeding methods are available from 16X-on
  - Show different behaviors, see later on.
- Pixel pair/triplet finding uses the *BeamSpot* from the *Event*

- Further developments in the pipeline
  - Seeding cleaning by L2 direction: **implemented**
    reject pixel pair/triplet according to L2 direction

  - Redundant seeding:
    Combine two seeding methods
  - Conditional seeding:
    Use different methods in separate seeding phase space
  - Cascade seeding:
    If one seeder fails for a L2, use another one

  - Seeding from L1: **implemented**, need some adjustments
    http://indico.cern.ch/conferenceDisplay.py?confId=23152 (M.Konecki)

Previous slides:
seeding functionality



<u>Next slides:</u>
pattern recognition

L2 Muon Error Rescaling

Description valid for all versions of CMSSW

- Reported errors on L2 Muon tracks are under-estimated
  - ➔ let's say it's “alright”, we don't use them directly

- Reported errors on L2 Muon tracks **after update** to beam spot are completely biased because of constraint fit
  - ➔ Need to adjust these errors if one want to use them
    - ➢ **Crucial for state-based seeding**

      http://indico.cern.ch/conferenceDisplay.py?confId=7692 (J-R Vlimant)

- Rescaling actually done at state-base seeding time.
  - ➔ Rescale factors based on muon gun MC

    http://indico.cern.ch/conferenceDisplay.py?confId=20377 (F.Rebassoo)
  - ➔ 1.0 to ~4, eta and pT dependent

Limited Number of Tracker Hits

Description valid for all versions of CMSSW

- Regular Ckf pattern recognition

- HLT exercise (13X cycle)
  - ➔ Number of hits limited to 7 (>5) for timing purpose
    - ✗ Has been shown to degrade pT resolution from 2% to 6%
      http://indico.cern.ch/conferenceDisplay.py?confId=19406 (J-R. Vlimant)
- 16X-on
  - ➔ No limitation on number of hits
    - ✓ Optiminal pT resolution from tracker
    - ✗ Timing increases
      http://indico.cern.ch/conferenceDisplay.py?confId=27058 (M. Pierini)
    - ✓ Other timing improvements buy us the extra timing

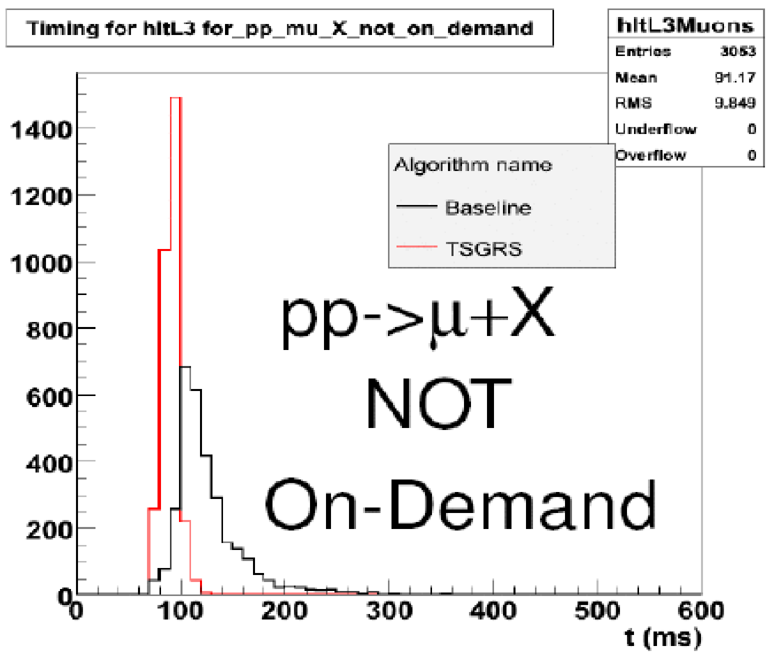
Tracker Pattern Recognition Module

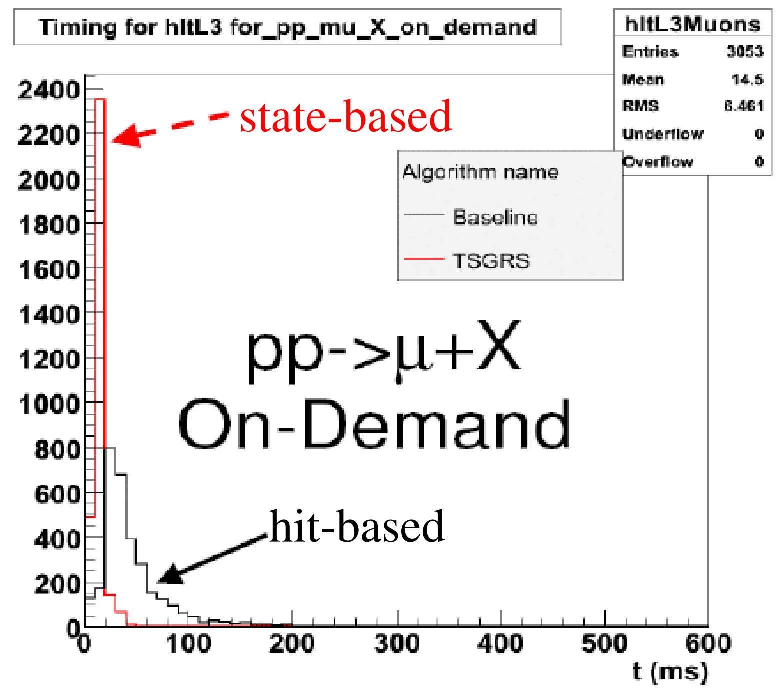
Comments only **valid for 200**

- Use *CkfTrajectoryMaker* module
  - Siamese to the well known ***CkfTrackCandidateMaker***
    - Benefit from well supported tracker code
    - *Trajectory* is a transient version of *TrackCandidate*
  - *Trajectory* is the input to the next module (L3 Muon)
  - Optional *TrackCandidate* collection can be put in the EventContent
    - If OK with memory footprint: seems to be the case
    - If OK with event size: seems to be the case
  - TrackCandidate are important for
    - Refitting tracker track without arbitration (c.f. $\mu\mu$ filter, see later on)
    - Diagnostics of muon HLT (DQM, off-line analysis)

Performance: Timing

$$pp \rightarrow \mu + X$$



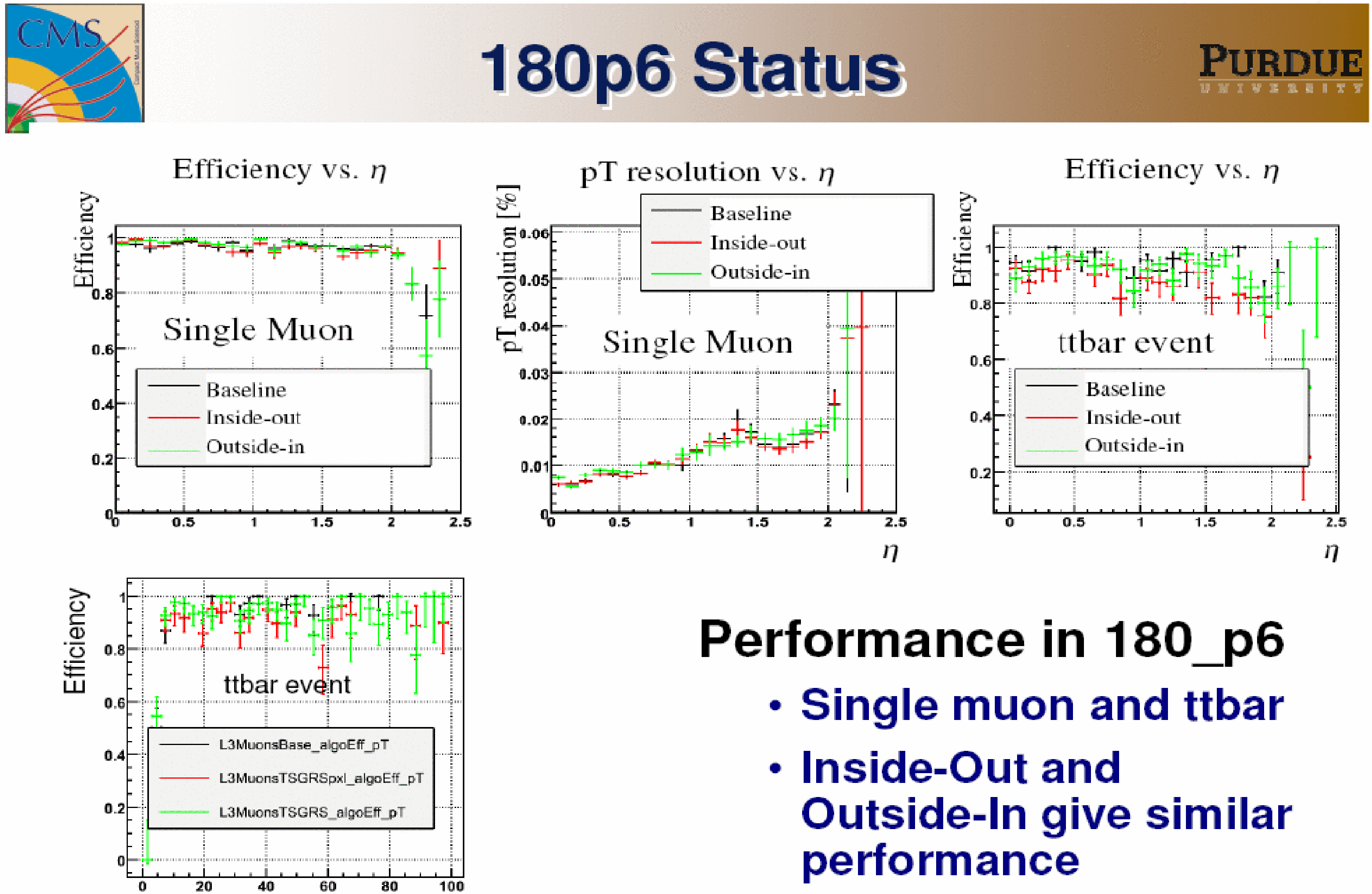


http://indico.cern.ch/conferenceDisplay.py?confId=23151 (J. Richman)

- State-based seeding/pattern recognition is faster (1.6.5 see above) indication that difference has been reduced with recent developments
- On-demand unpacking brings significant timing improvement (shared with other trigger path)

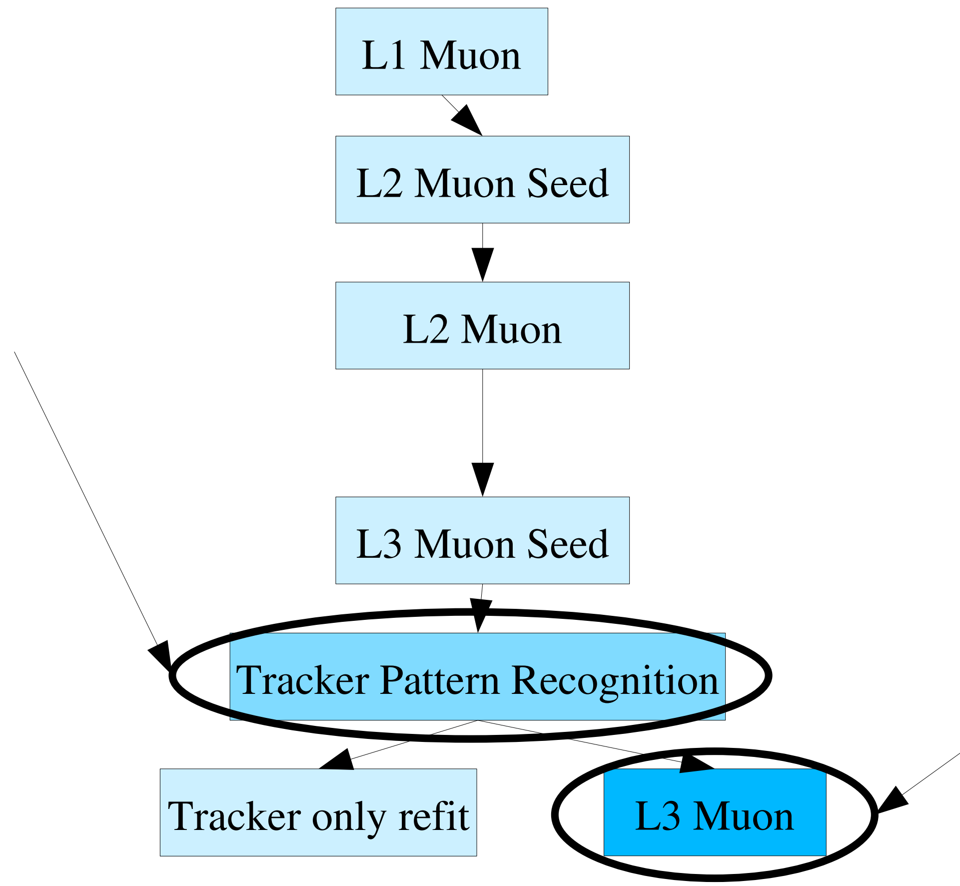
http://indico.cern.ch/conferenceDisplay.py?confId=25552 (J-R Vlimant)

Performance: efficiency



http://indico.cern.ch/conferenceDisplay.py?confId=25446 (A. Everett)

Previous slides:
pattern recognition



Next slides: L3 muon

- Matching
- Global fit

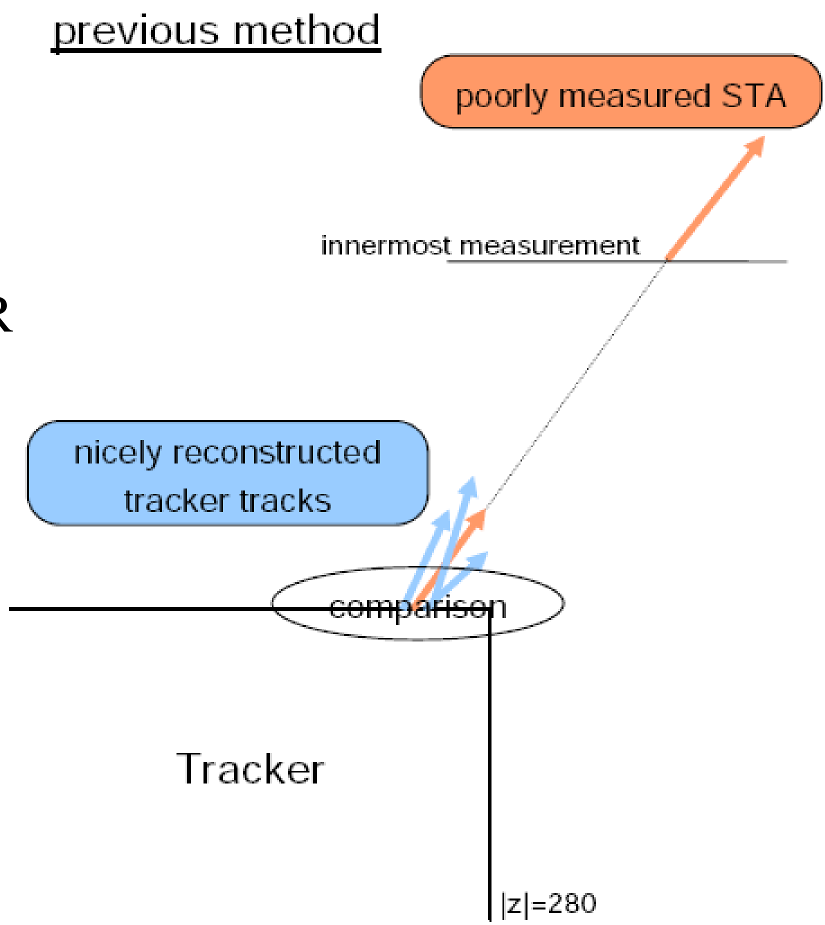
Global Track Matching "Old"

- For **one** L2, we get **N** tracker trajectories
  - ➔ Goal is to select **one** out of **N**
  - ➔ Match trajectories to L2 tracks

- **"OLD"** method: Matching at tracker outer surface
  - ➔ Propagate L2 and trajectory states to
  - ➔ Estimate parameter chi2, position ΔR, momentum ΔR

  - ✗ Reported to get confused in dense environment
    http://indico.cern.ch/conferenceDisplay.py?confId=23151 (J. Richman)
  - ✗ Cause for low pT inefficiency in tau->3muon
    http://indico.cern.ch/conferenceDisplay.py?confId=28907 (M. Giffels)

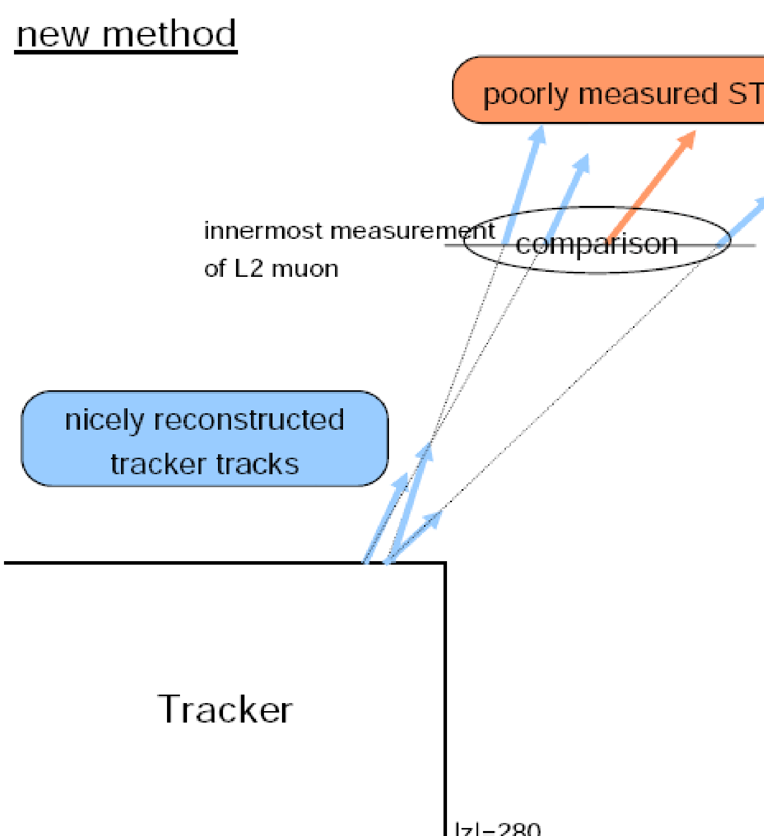
  - ✓ **Recent fix** back-ported in 16X (will be in 16.11)



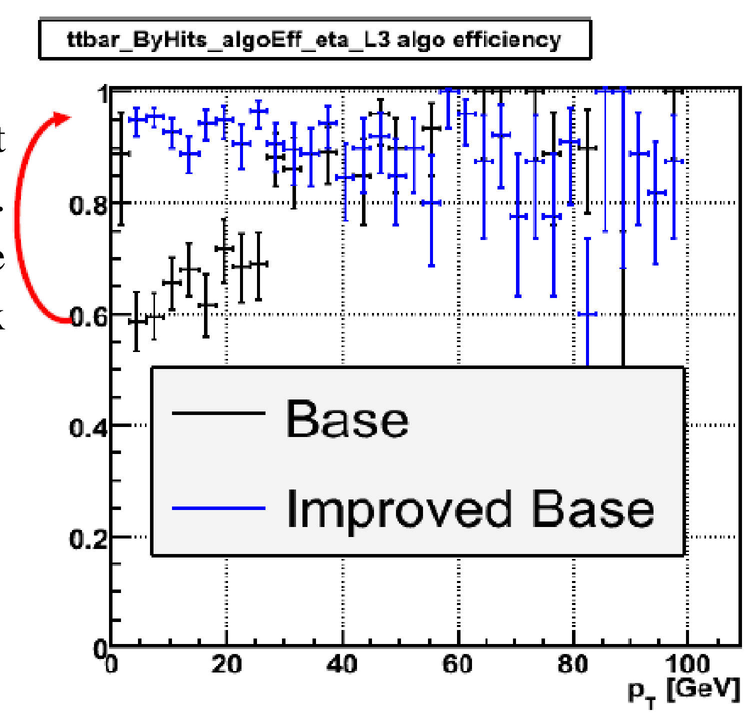
Global Track Matching "New"

- **"NEW"** method: Matching at first L2 muon hit surface (muon system surface)
  - Utilizes the fine pointing resolution from the tracker
  - Propagate trajectory states to surface of the innermost hit of L2 muon
    - Estimate parameter chi2, position distance, position ΔR, momentum ΔR

- This is the fix back ported in 16X (will be in 16.11) and present in 18X and 20X.

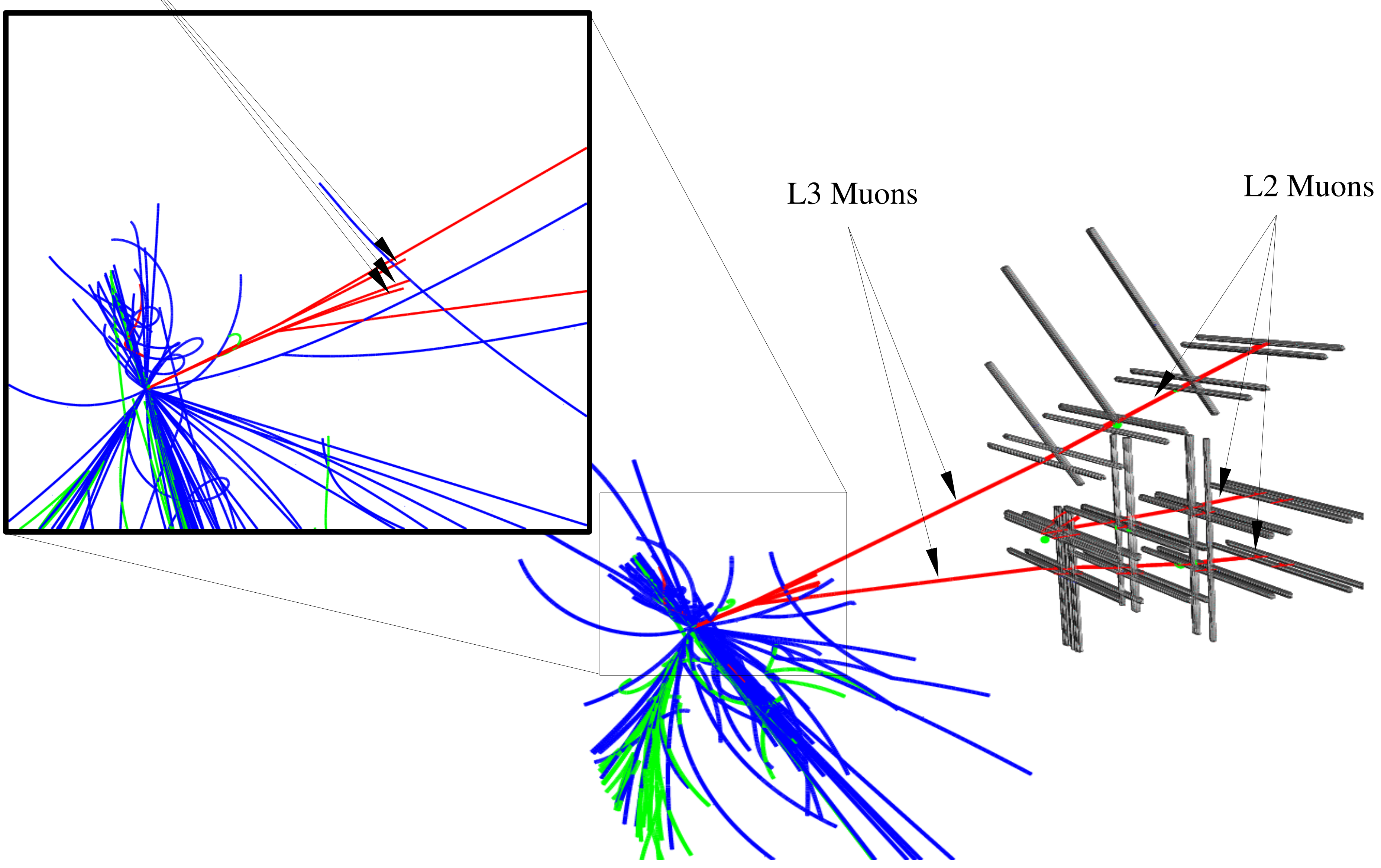


Efficiency improvement for muon in ttbar. Using a similar fix to the official fix



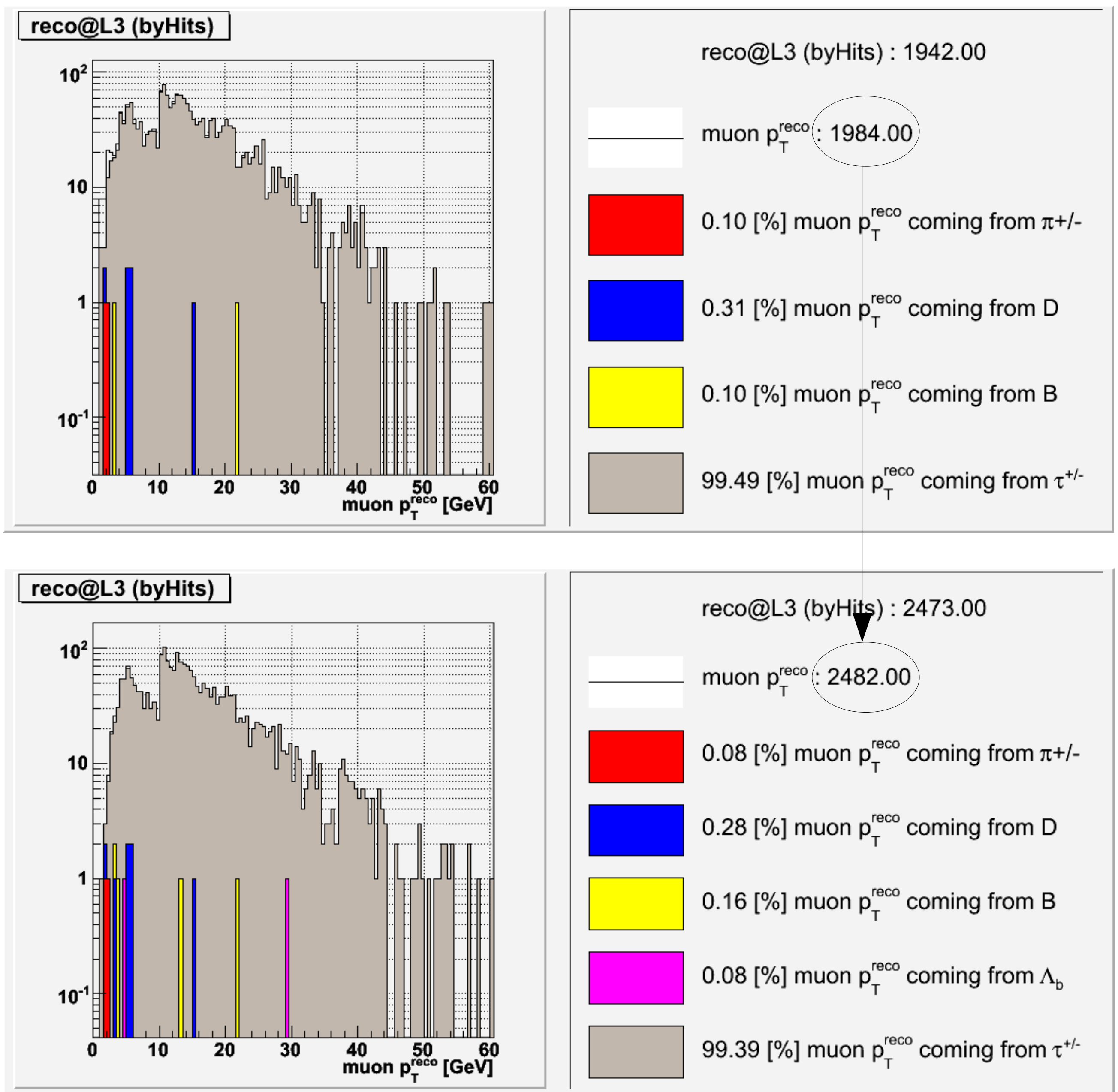
Tau→3μ Issue

- Generated muons are really close together



Tau→3μ Issue “Solved”

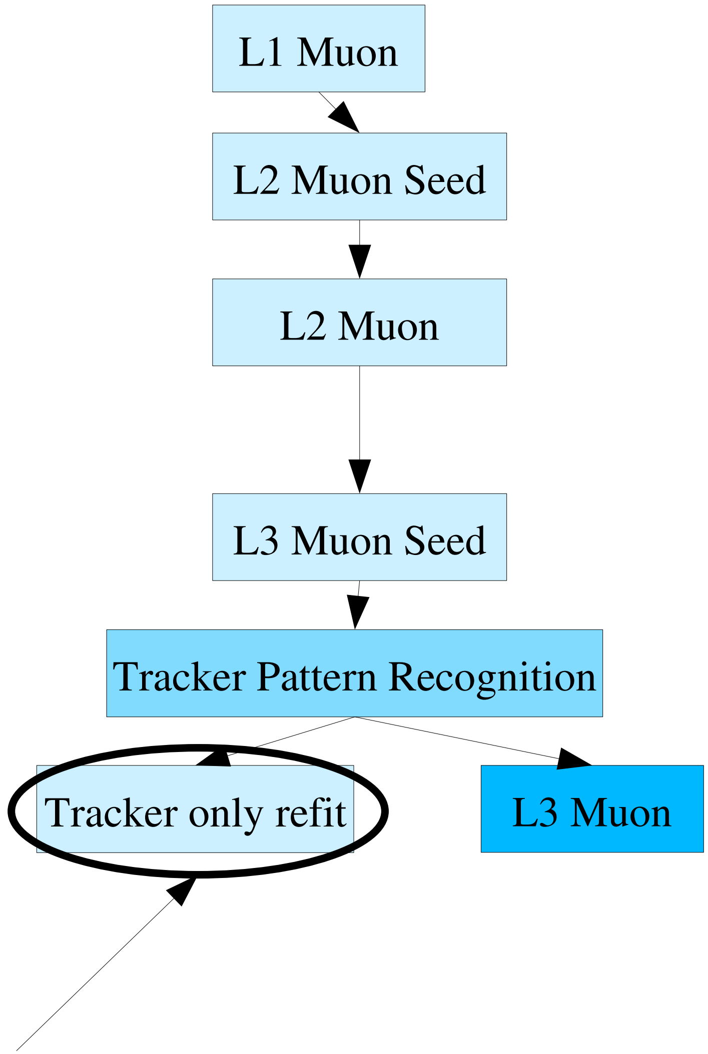
No L2/L3 filtering in these plots



OLD track matcher

“NEW” track matcher

- 25% more tracks
- Mostly muon track from tau decay



<u>Next slides:</u>

tracker only tracks

Tracker L3 Tracks

Feature only **available for 200**

- Tracker *TrackCandidate* **before arbitration** are available

- Natural input to the well known ***ctfWithMaterialTracks***
- Natural input to displaced muon filters

- Potential significant timing improvement for displaced muon trigger

Muon Related Trigger Path

- The ones using *l3muonreco* sequence
  - HLT1MuonIso
  - HLT1MuonNoIso
  - CandHLT1MuonPrescaleVtx2cm
  - ...

- The ones redoing l3 muon reconstruction
  - HLTBJPsiMuMu
  - $\mu\mu$k proposal

    http://indico.cern.ch/conferenceDisplay.py?confId=28907 (M. Giffels)

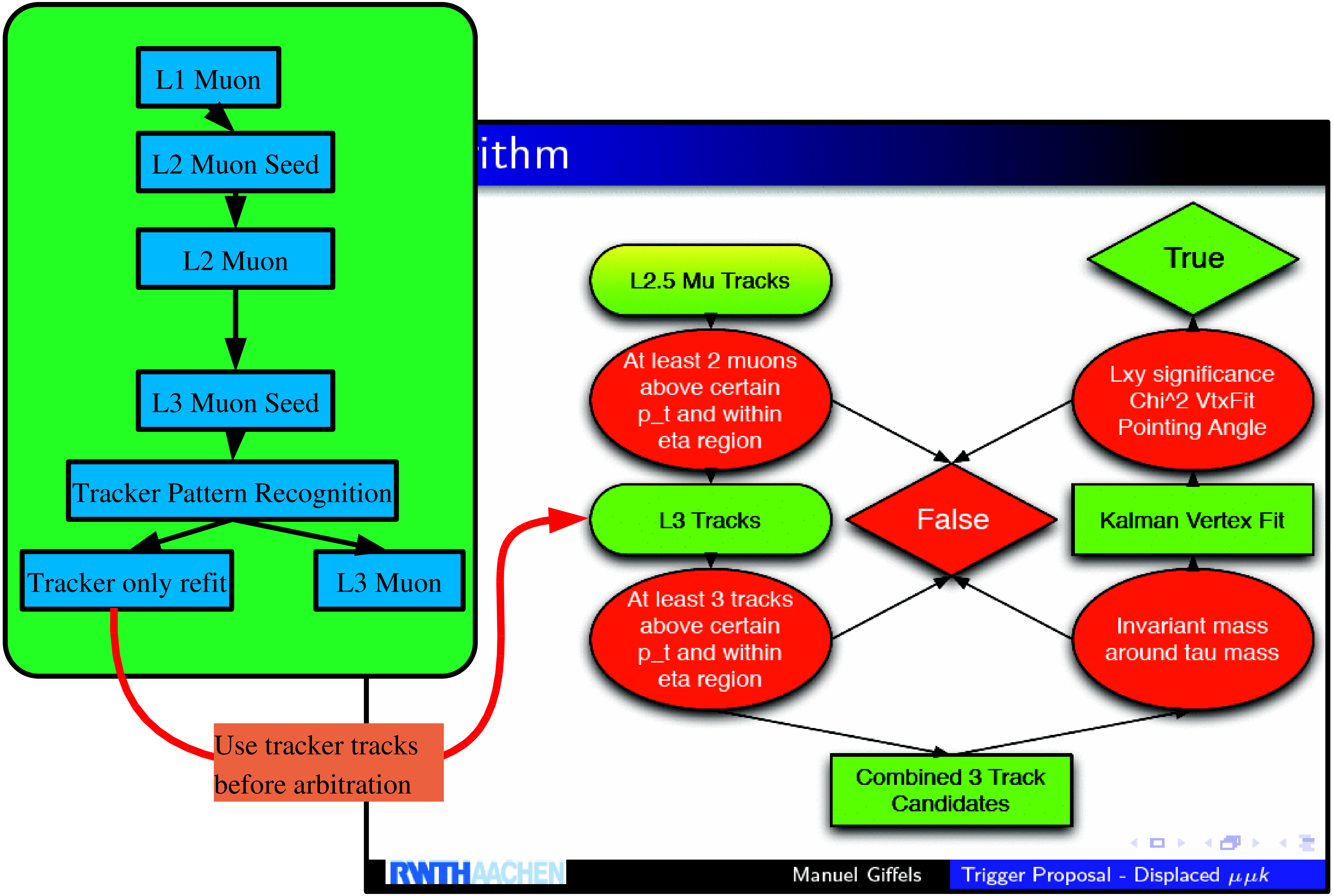
→ Actual seeding and pattern recognition are not that different

→ Wasting precious time in duplicating pattern recognition

→ Should work toward merging HLT paths: use only “one” l3 muon reco

  - ✓ to clearly identify the issue of muon HLT, when there is one

$\mu\mu k$ Proposal: Revised



Conclusions

- tau→3μ
  - ➔ identified source of inefficiency (there might be others)

- A lot of improvement in the past year for L3 Muon reconstruction
  - ➔ Different seeding methods and much more for a robust muon HLT
  - ➔ Timing improvement for a faster muon HLT
  - ➔ Thorough investigation in specific physics samples lead to improvements
  - ➔ Modularization for a more robust and flexible muon HLT

- Realistic possibility of merging displaced $\mu\mu$ trigger
  - ➔ Centrally monitored and maintained l3 muon reconstruction

- Did not talk about filtering or isolation
  - ➔ Needs a dedicated presentation on filtering
  - ➔ Room for improvement in isolation: please volunteer

The muon HLT is in great shape ! Thanks to the constant effort of the whole μHLT team