Tau Reconstruction & Event Data Model

Thomas Burgess
Stan Lai
Anna Kaczmarska
Michel Janus

Outline

Tau reconstruction and identification

Update on tau reconstruction code

Update on tau event data model

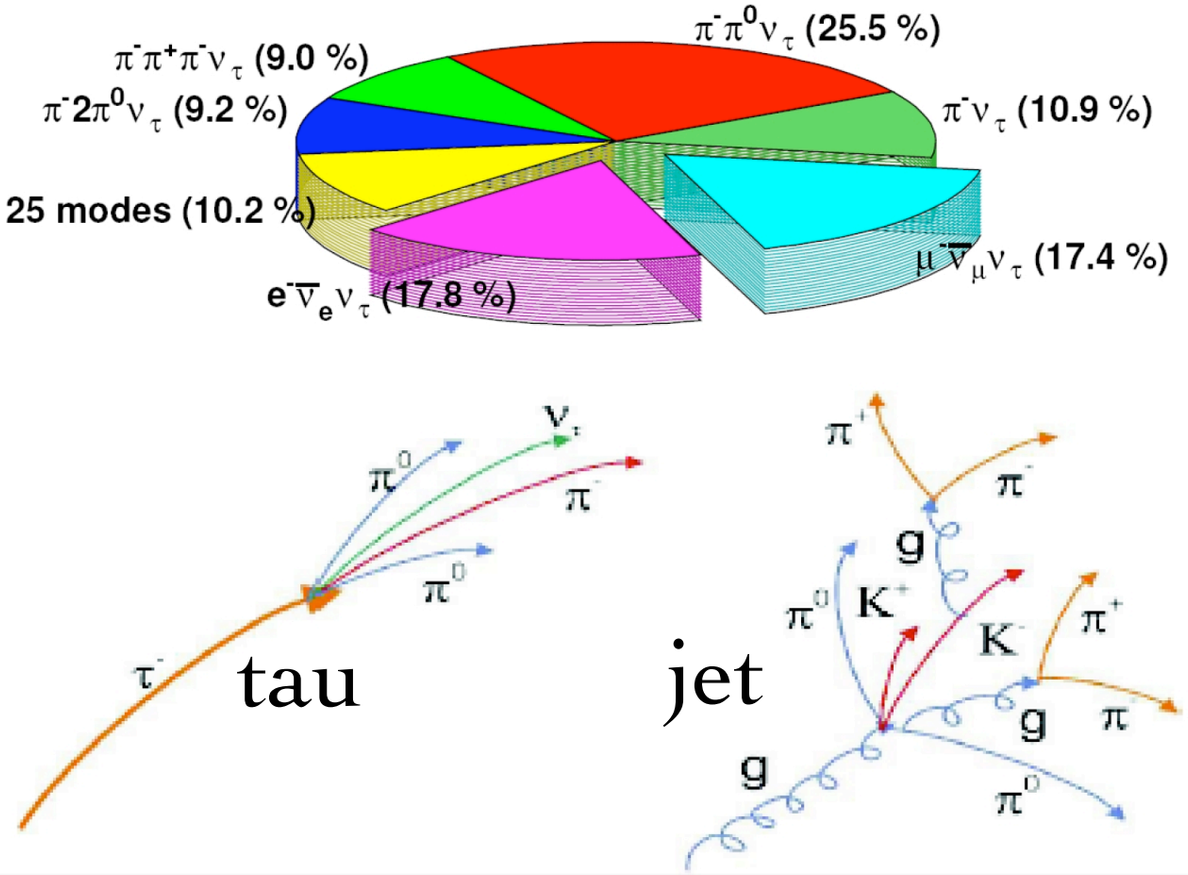
Tau reconstruction and identification

- Taus are heavy and decay rapidly m_τ=1.7 GeV, $c\tau$=87 μm

- Tau events can be reconstructed from the well collimated mesons produced by hadronically decaying taus (60% of all τ events)

- Most hadronic τ decays have either one or three (77% and 23% of cases respectively) charged tracks (prongs)

- The leading meson direction reproduces the original τ direction well



Reconstructing tau events

Tracking and calorimeter seeded reconstruction

* Use high quality track as seed ($p_T > 6$ GeV)
* Use candidates with 1-8 quality tracks ($p_T > 1$ GeV) within $\Delta R < 0.2$
* Reconstruct η, ϕ using p_T-weighting of tracks
* Check charge consistency
* Find matching cone4 jets ($E_T > 10$ GeV, $\Delta R < 0.2$) as calorimeter seed
* E_T using cells from calorimeter seed
* Energy flow algorithm
* Reconstruct π^0 subclusters

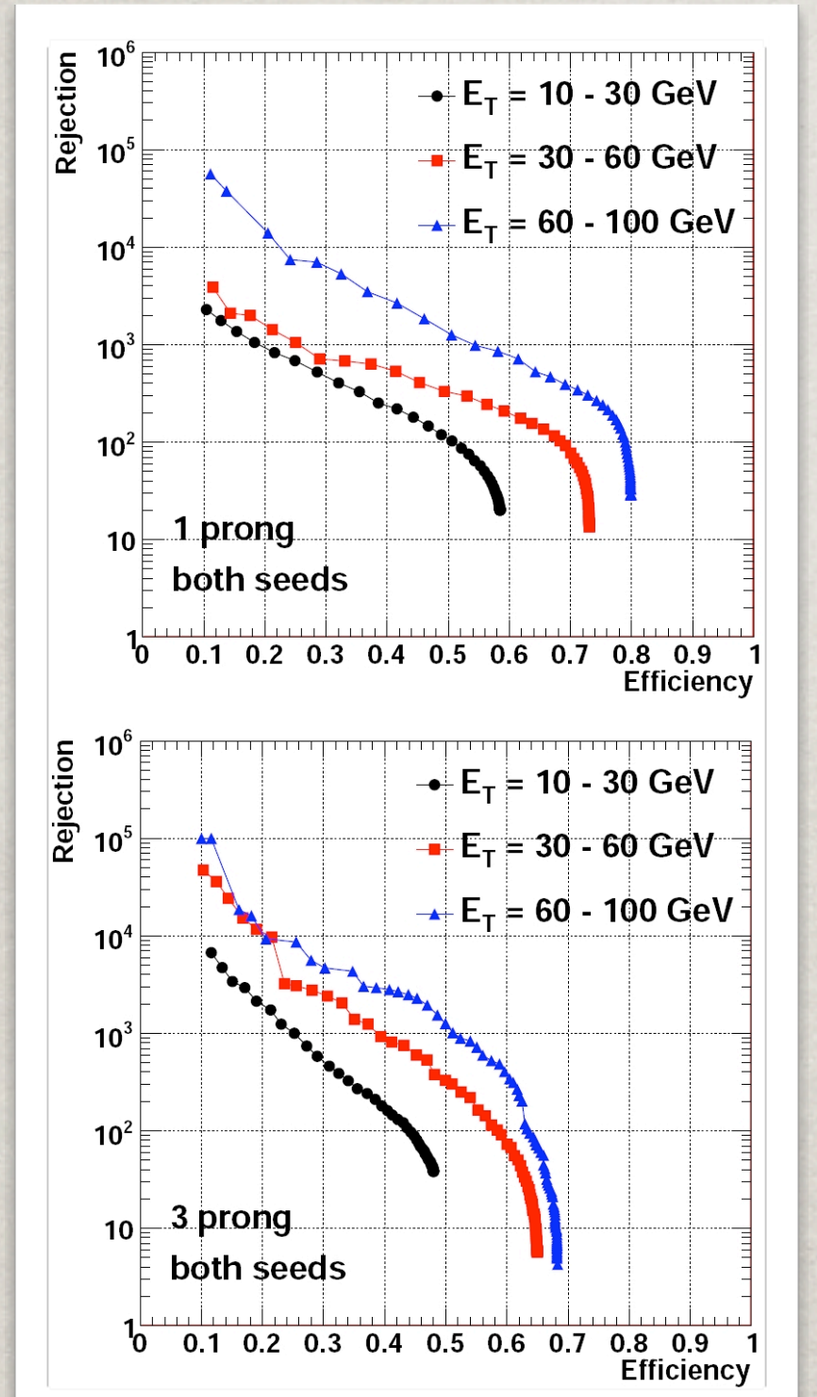
Calorimeter only seeded reconstruction

* Use remaining clusters as seed
* Define η, ϕ of τ candidate from cluster
* Looser track quality selection ($p_T >$ 1GeV)

Tracking only seeded reconstruction

* Tiny fraction of candidates (a few %)

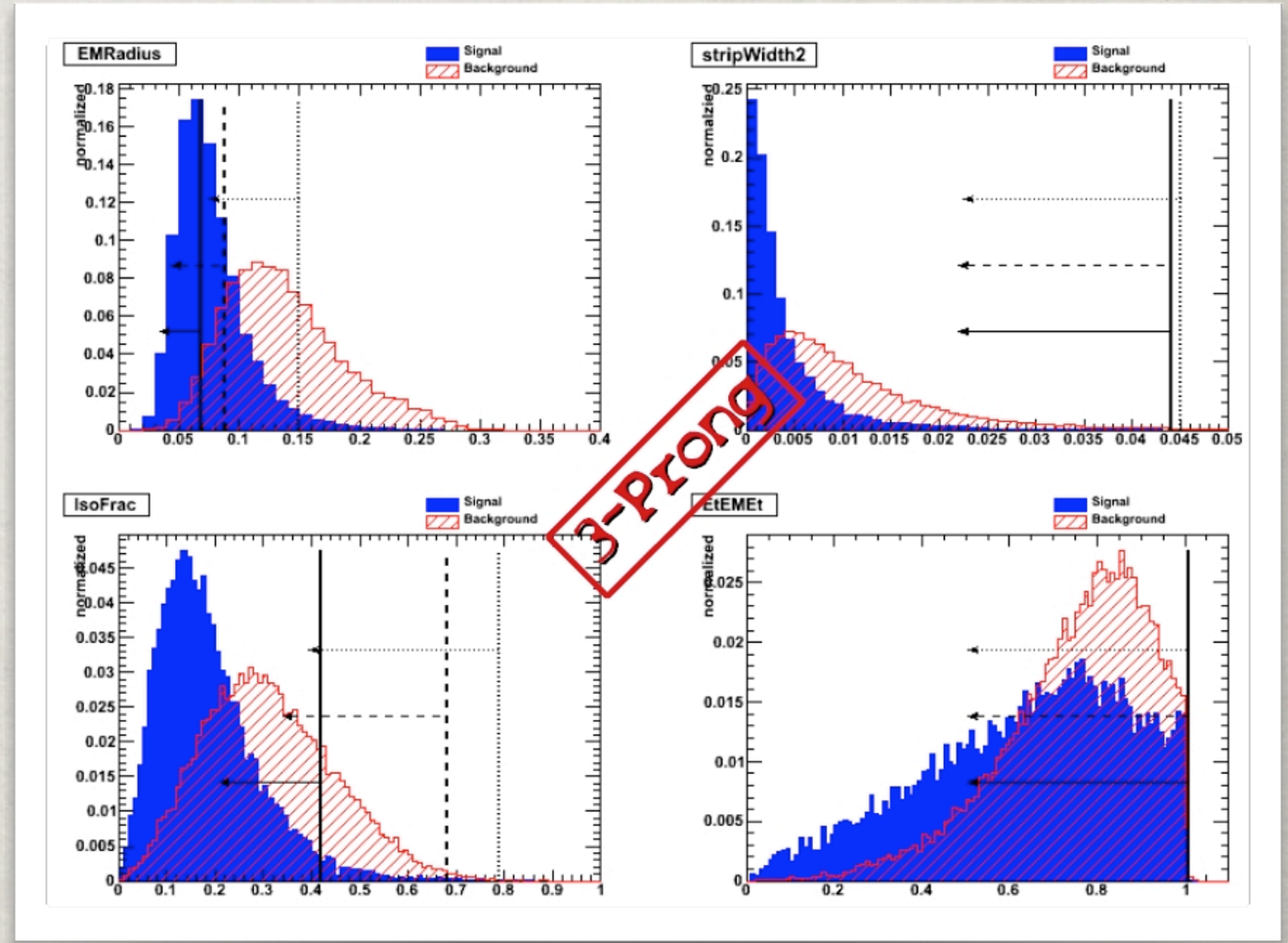
Identifing tau events



- Tracking information
  - Low track multiplicity (1 or 3), collimated tracks, secondary vertex reconstrucion (in 3-prong case), isolation from other tracks (cone))
- Calorimetry information
  - Collimated energy deposit in calorimeter, strong EM component in 1-prong case, possibility to identify π^0 clusters, use EM and HAD component
- Various multidimensional methods combing data for ID available

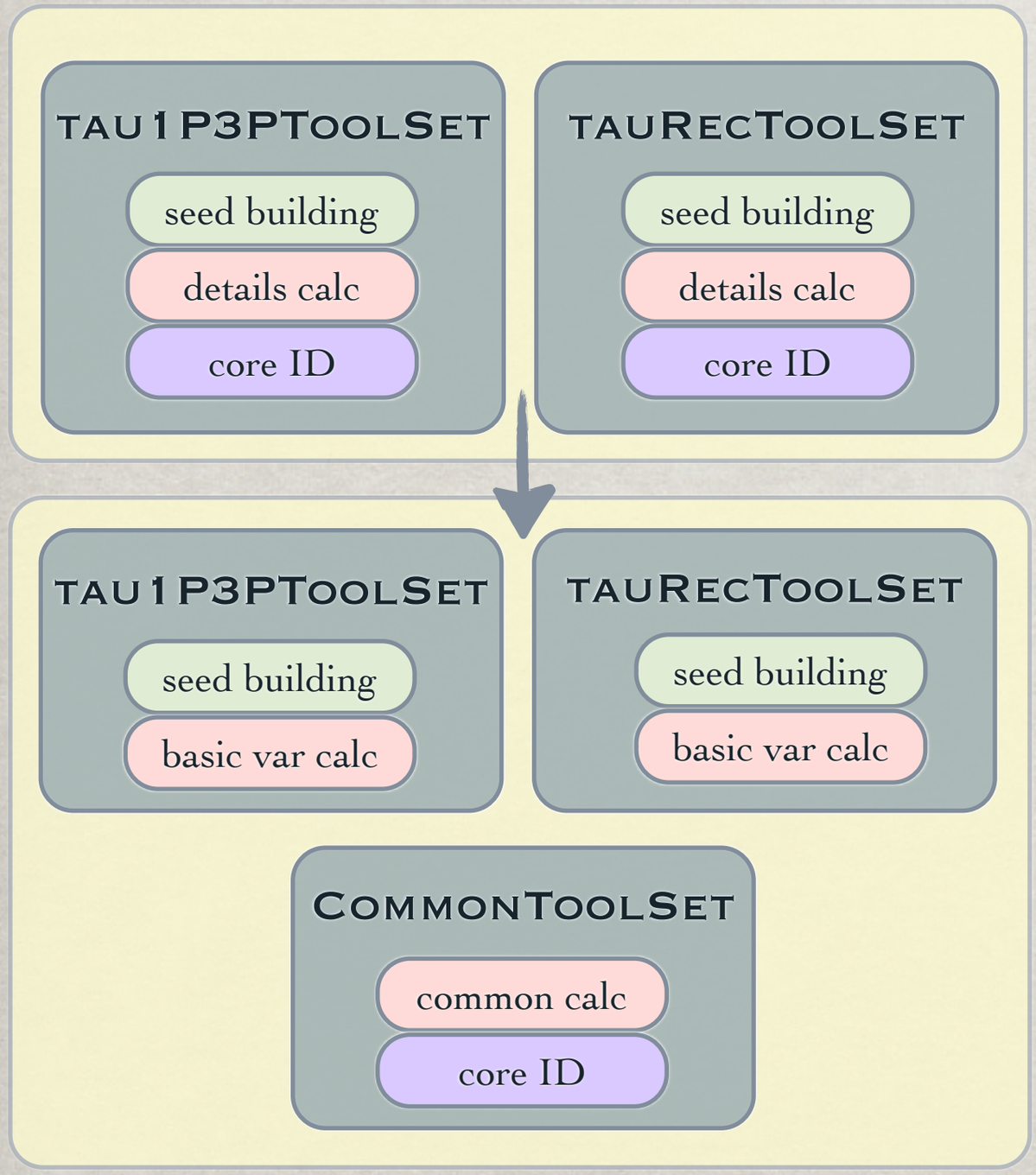
tauRec update

- Several new features available in rel 15
- Conversion tracks are identified and stored (tracks from photon conversions in τ decays with π^0s)
- New and improved π^0 reconstruction, using ECAL after subtraction of $\pi^{\pm}$ deposits - available in tauRec but not yet on by default
- Safe variables tau identification



Examples of some calorimter based variables and cut values used for safe tau identification

Restructuring of tauRec tool sets



- Further integration of common features in track and calo seeded reconstructions
- Seed dependent tasks kept in dedicated tools
- New common tool set introduced for seed independent tasks

Tau Id

- The tau Identification now has three levels of cuts based on safe variables (variables less likely affected by data-MC discrepancies)

TauPID::tauCut[Calo]Safe[LooselMediumlTight]()

- Use the Calo cuts for calorimeter only variables

- To use on data produced before rel 15 you must first convert to the new EDM (see slide 12 in a moment) and re-run TauID

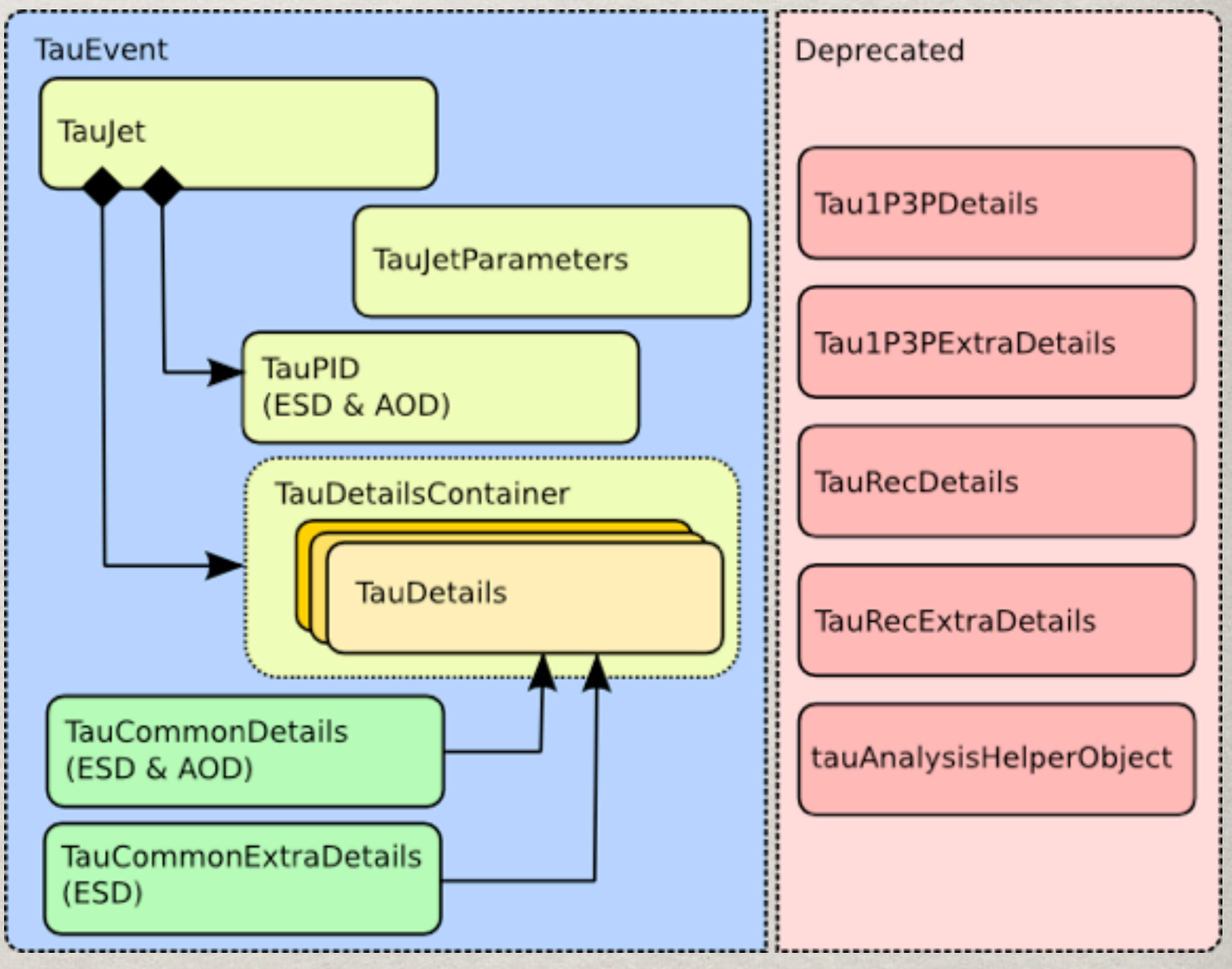
```
from tauRec.tauRecConf import TauCommonSetIsTau
theJob += TauCommonSetIsTau(OutputLevel = INFO)
```

tauRec to do list

- Validate current π^0 reconstruction and extend to 3 prong case
- Searching for additional and more effective safe ID variables
- Test using only TRT tracks as input
- Try looser track selection for cosmic data

Tau Event Data Model

- Major effort for rel 15: migration from seed dependent to common details
- Seed dependent info in common details prefixed with seedTrack or seedCalo
- Old Tau[1P3P|Rec][Extra]Details are now deprecated



Backwards compatibility issues

- Reading in old (<15.0.0) files works fine with the merged EDM
- Writing deprecated details is prohibited!
- Conversion to new details done with TauJet::MergeToCommonDetails
- This caused very annoying bugs (#47902) (fixed in RecTPCnv-00-06-04)

How to merge to common details

- TauJet::MergeToCommonDetails() requires non-const access to TauJet, but StoreGate returns const

- To circumvent const access use the global function Analysis::mergeToCommonDetails(taujet) (available since tauEvent-00-05-04)

  - To merge from jobOptions, use the athena algorithm TauTools/TauCommonDetailsMerger (available since TauTools-00-04-09)

```
from TauTools.TauToolsConf import TauCommonDetailsMerger
job += TauCommonDetailsMerger("TauCommonDetailsMerger")
```

Tau EDM outlook

- Commit operators == and != and str (to string function) for TauJet, and TauCommon(Extra)Details

- Fix RecTPCnv to avoid warnings when using old EDM files (recipe tested and seems to work but not fully validated yet)

- Fix non-critical bugs persistency problem in closest[Phi|Eta]Trk[Vert]Cell and the (intentional) memory leak in MergeToCommonDetails

- Hopefully entering a less intensive phase of development soon

Thank you!

Any questions?