Migration to new EDM

New EDM content

The new class is called Analysis::TauJet.

- TauJet - the main class for both algorithms. It inherits from ParticleBase (information about origin and charge) and P4EEtaPhiM (4momentum information)
  - author
  - pointer to TauPID object
  - vector of links to TauDetails
  - vector of links to Tracks
  - Two links to CaloCluster
  - CaloCluster *cluster() - for tauRec seed cluster, for tau1p3p the nearest calorimetric cluster
  - CaloCluster *cellCluster() - for both algorithms cluster with cells selected for discrimination variables/energy building
- TauPID - holds results of the identification algorithms. It is implemented as vector od key-velue pairs. One can insert following keys and values associated with them
  - DiscNN
  - Likelihood
  - DiscPDERS
  - TauJetLikelihoodLowPt
  - TauELikelihoodLowPt
  - TauJetNeuralNetwork
  - TauENeuralNetwork
  - EfficNN
  - EfficPDERS
  - TauElTauLikelihood
  - TauPID contains also isTau flag determining whether the object passed the cuts or not
  - bitset for electron/muon veto flags
- TauDetails classes are meant to algorithm speciffic information. There are two detail classes implemented for each algorithm.
- TauRecDetails - to be stored in ESD and AOD
  - emRadius
  - isolationFraction - ratio of the uncalibrated transverse energy within dR < 0.1 and dR < 0.2
  - centralityFraction - centrality fraction (ET(dr<0.1)/ET(dr<0.4) for all calos
  - stripWidth2 - uncalibrated transverse energy weighted width in the strip layer within dR < 0.4
  - numStripCells
  - HepLorentzVector sumEM - EM part of the energy
  - etEMCalib - calibrated EM transverse energy, EM is Presampler + EM1 + EM2
  - etHadCalib - calibrated HAD transverse energy, HAD is fudge*(cryo + EM3 + TILE1 + TILE2 + TILE3)
  - Trk :: RecVertex *secVertex
  - numTrack
  - trackCaloEta(int i)
  - trackCaloPhi(int i)

    - leadingTrackPT - PT of leading track - for Trigger
- TauRecExtraDetails - to be stored in ESD only
    - numEMCells - number of EM cells within dR < 0.4, with E > m_cellEthr
    - stripET - uncalibrated sum of ET in the strip layer within dR < 0.4
    - nTracksdrdR - number of tracks pT>1GeV between dR = cmsdr and dR = cmsdR from the jobOptions
    - emCentralityFraction - EM Centrality Fraction (ET(dr<0.1)/ET(dr<0.4) for EM calos only
    - etHadAtEMScale
    - etEMAtEMScale
    - energy - energy sum of all cells within dR < 0.4
    - emEnergy - energy sum of all em cells within dR < 0.4 (Presampler + EM1 + EM2)
    - sumPTTracks
    - Rec :: TrackParticle *lowPtTrack(unsigned i)
    - numLowPtTrack
- Tau1P3PDetails - to be stored in ESD and AOD
    - emRadius
    - HepLorentzVector sumEM()
    - isolationFraction
    - stripWidth2
    - numStripCells
    - etChrgHAD
    - etIsolEM
    - etIsolHAD
    - nAssocTracksCore
    - nAssocTracksIsol
    - massTrk3P
    - rWidth2Trk3P
    - signD0Trk3P
    - etHadAtEMScale
    - etEMAtEMScale
    - etEMCL
    - etChrgEM
    - etNeuEM
    - etResNeuEM
    - etChrgEM01Trk(unsigned tr)
    - etResChrgEMTrk(unsigned tr)
    - Trk :: RecVertex *secVertex
    - numPi0
    - CaloCluster pi0(unsigned num)
- Tau1P3PExtraDetails - to be stored in ESD only
    - CaloCell closestEtaTrkVertCell(int itrk, int sampling)
    - CaloCell closestPhiTrkVertCell(int itrk, int sampling)
    - CaloCell closestEtaTrkCell(int itrk, int sampling)
    - CaloCell closestPhiTrkCell(int itrk,int sampling)
    - etaTrackCaloSamp(int itrk, int sampling)
    - phiTrackCaloSamp(int itrk, int sampling)
    - sumPTTracks()

How to migrate your code

The best example is tauRec/src/CBNT_tau.cxx. Here we are providing basic informations. Cointainers names are:

- **TauRecContainer**
- **TauRecDetailsContainer**
- **TauRecExtraDetailsContainer**
- **Tau1P3PContainer**
- **Tau1P3PDetailsContainer**
- **Tau1P3PExtraDetailsContainer**

Following includes should be used:

- **#include "tauEvent/TauJetContainer.h"**
- **#include "tauEvent/TauJet.h"**
- **#include "tauEvent/Tau1P3PDetails.h"**
- **#include "tauEvent/Tau1P3PExtraDetails.h"**
- **#include "tauEvent/TauRecDetails.h"**
- **#include "tauEvent/TauRecExtraDetails.h"**
- **#include "tauEvent/TauPID.h"**

Example how to get variables from different containers:

```
const Analysis::TauJetContainer  *tau_container;
StatusCode sc = m_storeGate->retrieve(tau_container, m_tauContainerName);
Analysis::TauJetContainer::const_iterator ftau = tau_container->begin();
Analysis::TauJetContainer::const_iterator etau = tau_container->end();
for (; ftau != etau; ftau++)
{
    double energy  = (*ftau)->et();   //access methods from TauJet
    int charge         = (*ftau)->charge();

    //for discriminant information
    const Analysis::TauPID*              p_tauid    =  (*ftau)->tauID();
    double pders = p_tauid->discriminant( TauJetParameters::DiscPDERS );

    //algorithm specific
    if ((*ftau)->author() == 2) //tau1p3p3
    {
       const Analysis::Tau1P3PDetails*  p_taudetails = (*ftau)->details<const Analysis::Tau1P3PDe
       int strips=p_taudetails->numStripCells();
     }
    if ((*ftau)->author() == 1) //tauRec
    {
      //details
      const Analysis::TauRecDetails*  p_taudetails = (*ftau)->details<const Analysis::TauRecDetai
      double isol = p_taudetails->isolationFraction();

      //extradetails
      const Analysis::TauRecExtraDetails*  p_tauextradetails =
          (*ftau)->details<const Analysis::TauRecExtraDetails>(m_tauExtraDetailsContainerName);

      int numcells = p_tauextradetails->numEMCells();

    }
}
```

- with the new EDM using cells (connected to object via cluster) is possible. Example how to do can be found in tauCellBulder.cxx and tau1p3pAddCaloInfo.cxx:

```
const CaloCluster *p_cluster = tau->cellCluster();
```

```
CaloCluster :: cell_iterator firstcell = p_cluster->cell_begin();
CaloCluster :: cell_iterator lastcell = p_cluster->cell_end();
const CaloCell *p_cell;
for( ; firstcell != lastcell; firstcell++ )
{
   ..............
}
```

This topic: Main > TauEDM
Topic revision: r1 - 2007-05-16 - AndrewHamilton