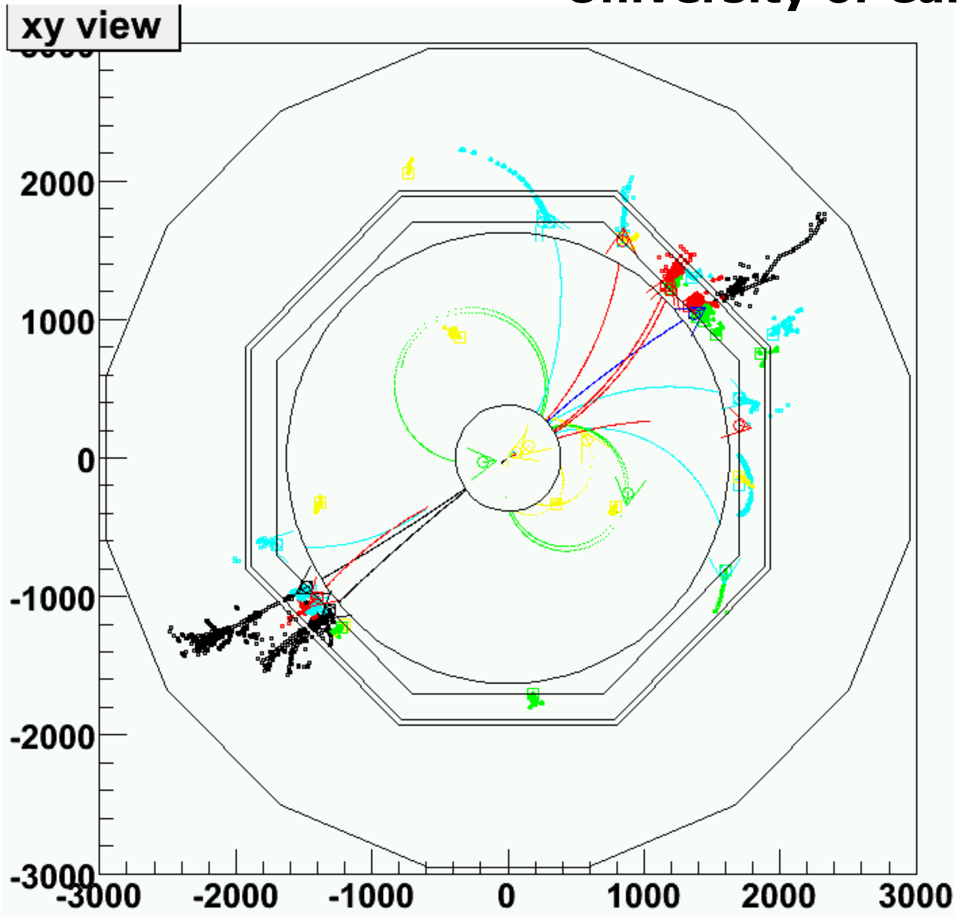
A Topologic Approach to Particle Flow "PandoraPFA"

Mark Thomson
University of Cambridge

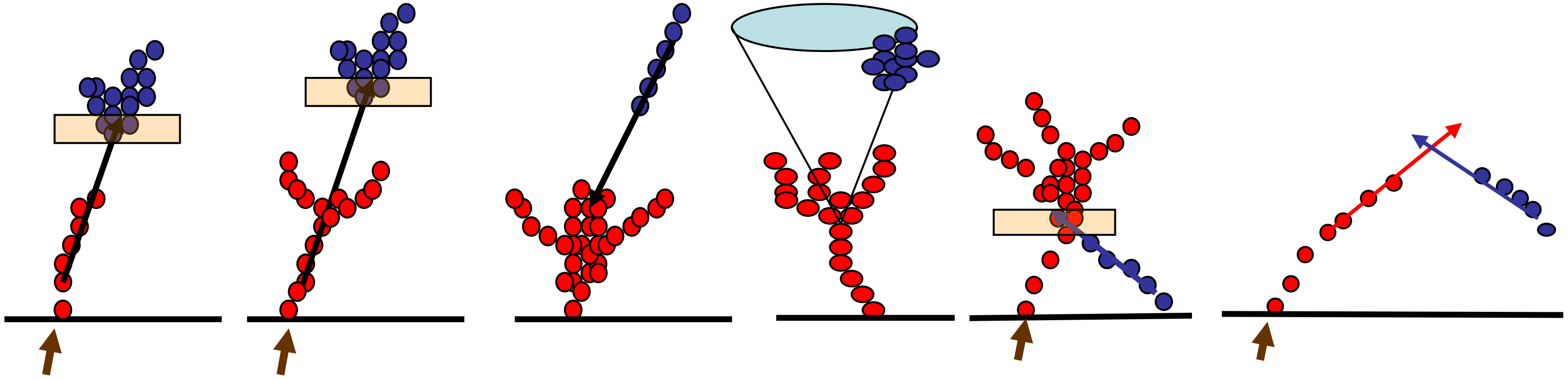


This Talk:

★ Philosophy
★ The Algorithm
★ Some First Results
★ Conclusions/Outlook

❶ Philosophy

- ★ Work from the premise that PFA is not a pure ECAL/HCAL clustering problem
- ★ PFA and calorimeter clustering performed together
- ★ Start by applying loose clustering
- ★ Then join clusters using topology



- ★Algorithm defined by loose cluster + topological rules

Goals/Framework

- ★ Try to develop "generic" PFA which will take advantage of a high/very high granularity ECAL
- ★ Clustering and PFA performed in a single algorithm
- ★ Aim for fairly generic algorithm:
  - very few hard coded numbers
  - use GEAR to get basic geometry
- ★ Clustering uses tracking information
- ★ Initial clustering is fairly loose → ProtoClusters
- ★ Topological linking of ProtoCluster

Runs in MARLIN framework using:

- ✦ GEAR (geometry interface)
- ✦ Marlin SimpleDigitisation
- ✦ Track finding/fitting : TrackCheater
- ✦ PFA Utility classes, e.g. Helix class for track extrap. (Alexei R.)

❷ The Algorithm

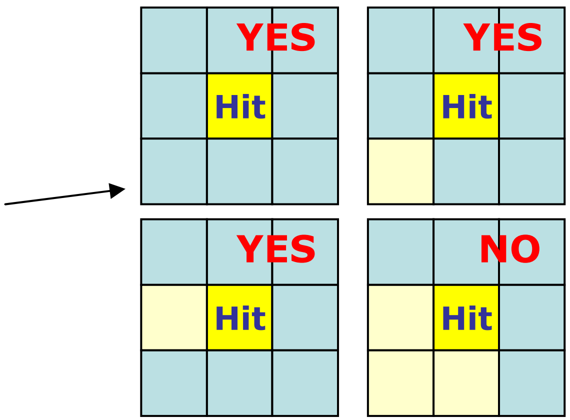
Overview:

- ★ **Preparation**
  - ★ **Isolation cuts, hit ordering, track quality**
- ★ **Initial clustering to form ProtoClusters**
  - ★ **ProtoClusters are heavyweight object:**
    - ★ **collection of hits**
    - ★ **know how to grow (configured when created)**
    - ★ **information about shape, direction, isPhoton,…**
    - ★ **+much more (not all used)…**
- ★ **Cluster association/merging**
  - ★ **Tight Topological linking of clusters**
  - ★ **Looser merging of clusters**
  - ★ **Track-driven merging**
- ★ **PFA**
  - ★ **Final track-cluster matching**

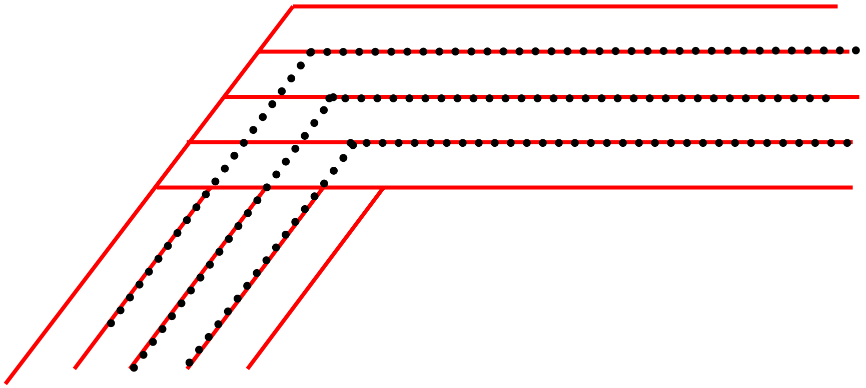
- **This talk gives flavour of what's done in each stage skipping details**

Preparation I: Extended Hits

- Create internal **ExtendedCaloHits** from **CaloHits**
- **ExtendedCaloHits** contain extra info:
  - pointer to original hit
  - pseudoLayer (see below)
  - measure of isolation for other hits
  - is it MIP like (to ID "tracklike objects")
  - actual layer (decoded from **CellID**)
  - Pixel Size (from GEAR)
- hits are now self describing

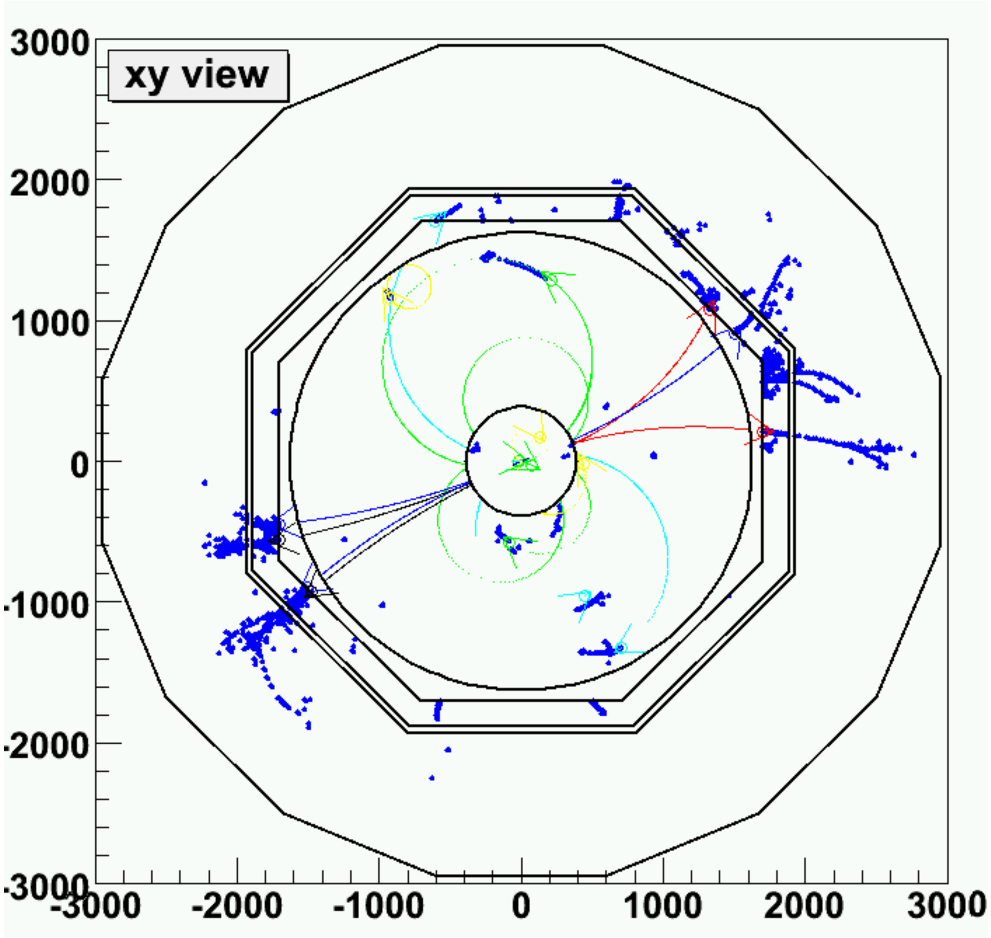


- Arrange hits into **PSEUDOLAYERS (e.g. Chris Ainsley's MAGIC)**
  - **i.e.** order hits in increasing depth within calorimeter
  - PseudoLayers follow detector geometry



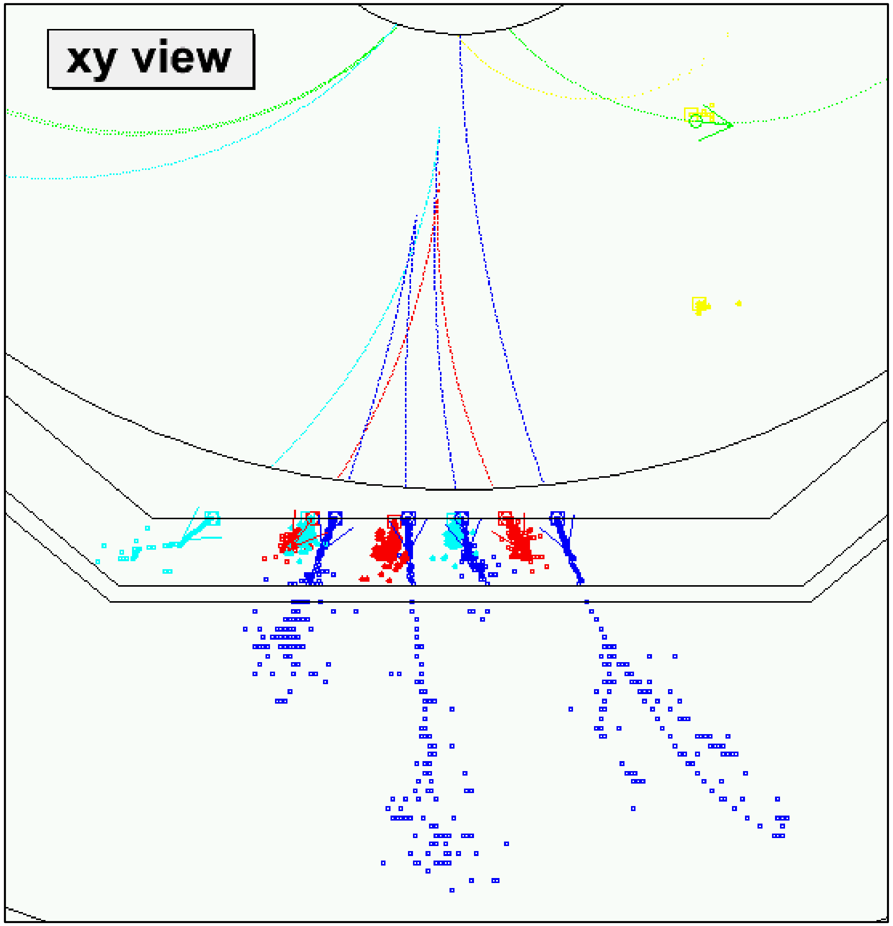
Preparation II: Isolation

- ★ Divide hits into isolated and non-isolated
- ★ Only cluster **non-isolated** hits
- ★ "Cleaner"/Faster clustering
- ★ Significant effect for scintillator HCAL



- ✦ Removal of isolated hits degrades HCAL resolution
- ✦ <u>e.g. D10scint:</u>
  50 %/√E/GeV → 60 %/√E/GeV

Preparation III: Tracking



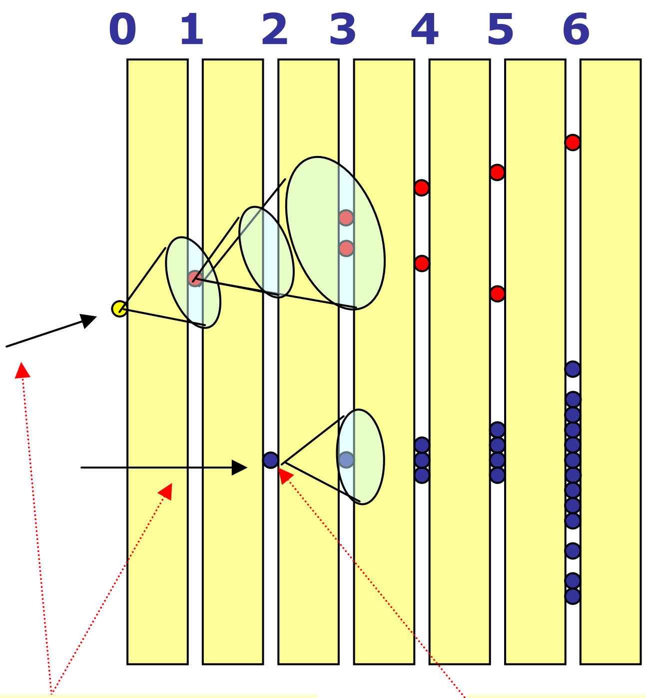
- ★Use MARLIN **TrackCheater**
- ★Tracks formed from MC Hits in **TPC/FTD/VTX**
- ★ HelixFit (Alexei R) ⇒ track params
- ★ Cuts (primary tracks):
  - ♦ $|d_0| < 5$ mm
  - ♦ $|z_0| < 5$ mm
  - ♦ >4 non-Si hits

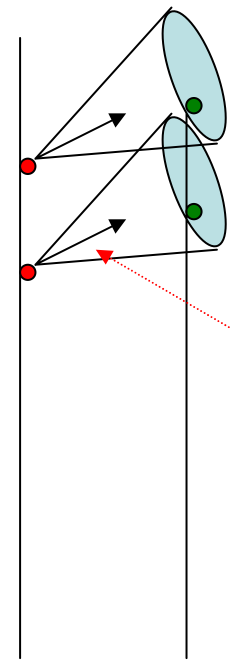
+ V_0 and Kink finding:

- ✦Track resolution better than cluster
- ✦Improves PFA performance by ~2 %

PandoraPFA Clustering II

- Start at inner layers and work outward
- Associate Hits with existing Clusters
- If multiple clusters "want" hit then Arbitrate
- Step back **N** layers until associated
- Then try to associate with hits in current layer (M pixel cut)
- If no association made form new Cluster
- + tracks used to seed clusters





Simple cone algorithm based on current direction + additional N pixels

Cones based on either:
initial PC direction or
current PC direction

Initial cluster direction

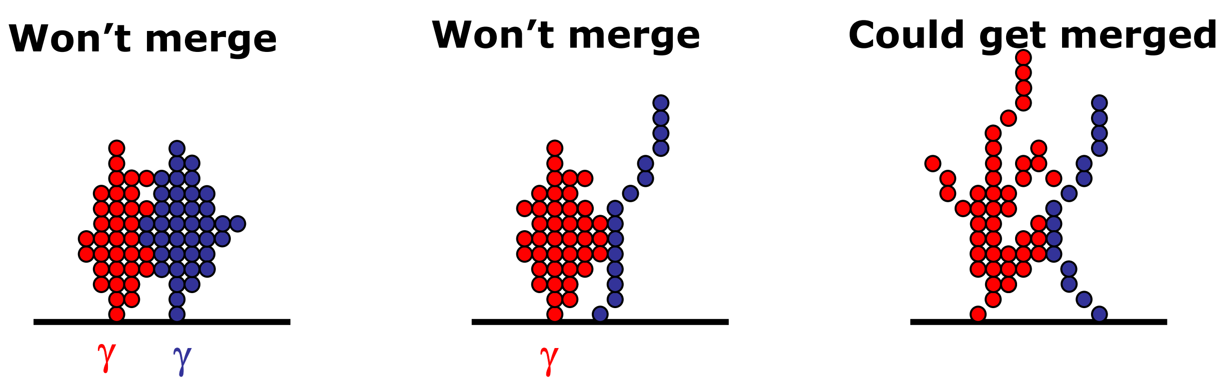
Unmatched hits seeds new cluster

Cluster Association

- ✦ **By design clustering errs on side of caution**
  i.e. clusters tend to be split
- ✦ **Philosophy: easier to put things together than split them up**
- ✦ **Clusters are then associated together in two stages:**
  - **1) Tight cluster association - clear topologies**
  - **2) Loose cluster association – catches what's been missed but rather crude**

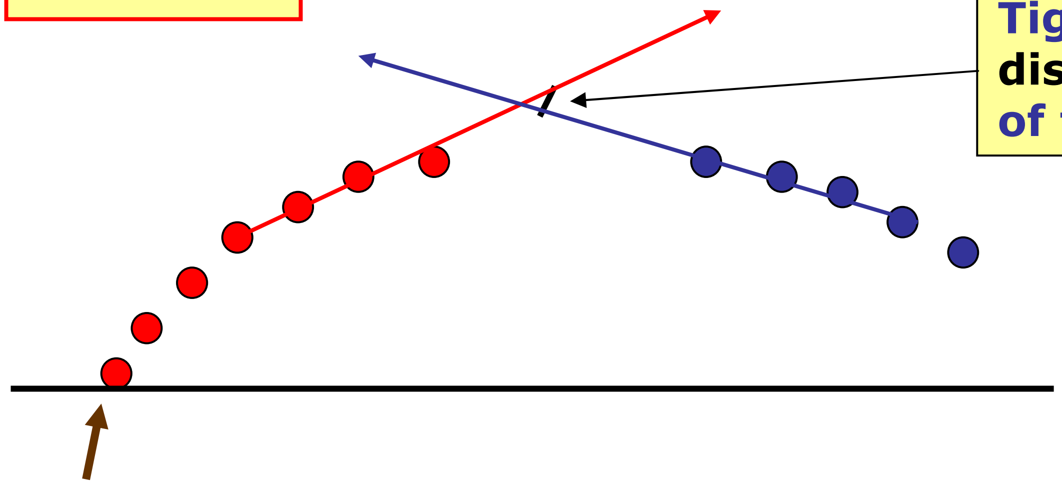
Photon ID

- ★ **Photon ID plays important role**
- ★ **Simple "cut-based" photon ID applied to all clusters**
- ★ **Clusters tagged as photons are immune from association procedure – just left alone**



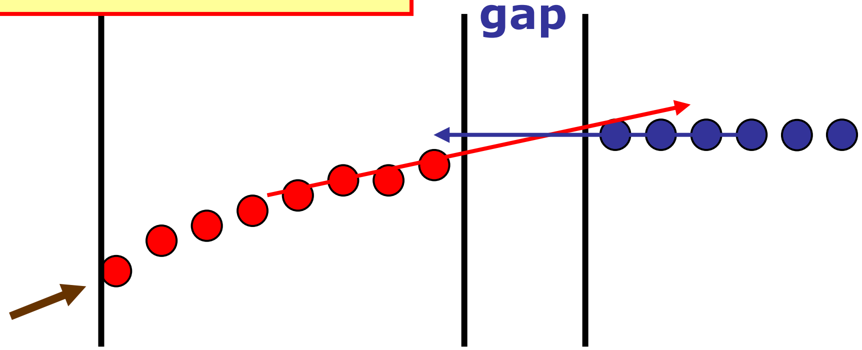
Cluster Association I : track merging

LOOPERS



Tight cut on extrapolation of distance of closest approach of fits to ends of tracks

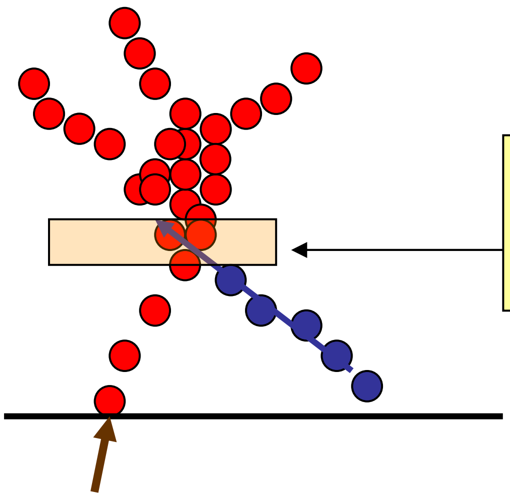
SPLIT TRACKS



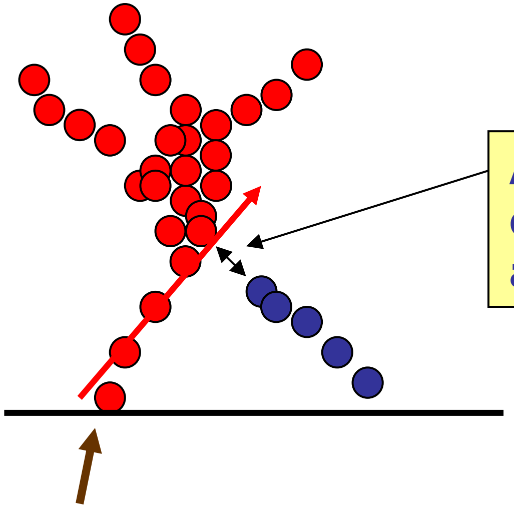
Tight cut on extrapolation of distance of closest approach of fits to end of inner tracks and start of outer track

Cluster Association II : Backscatters

★ **Forward propagation clustering algorithm has a major drawback: back scattered particles form separate clusters**



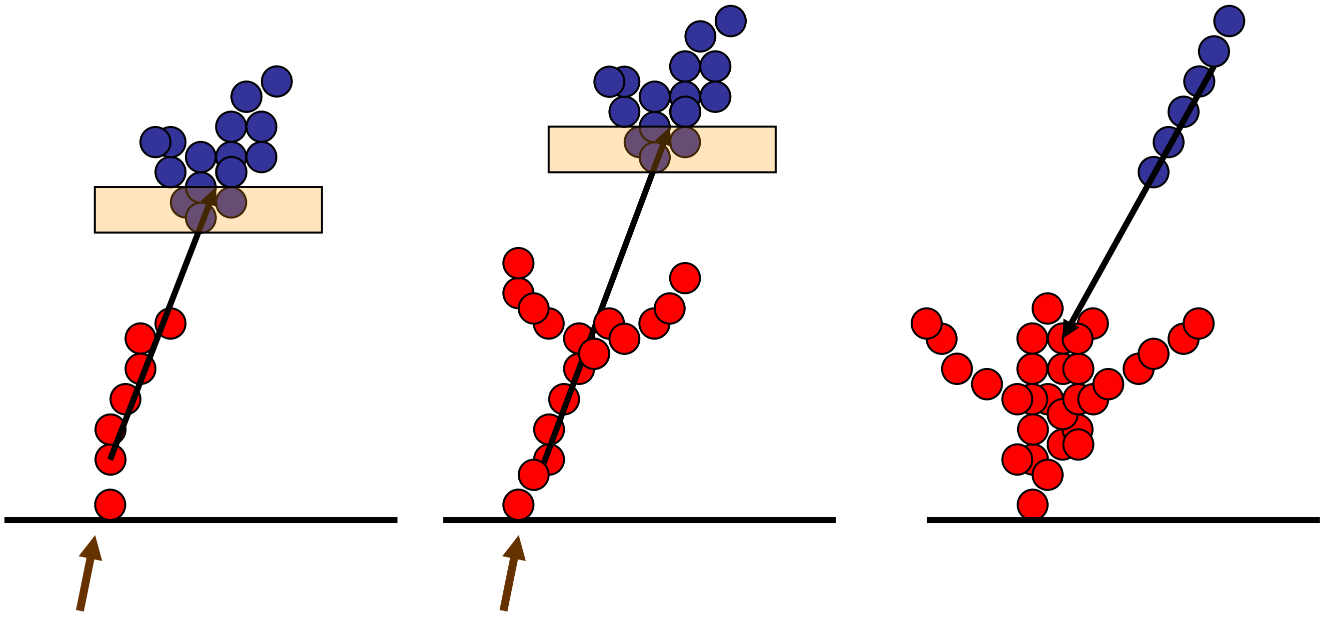
Project track-like clusters forward and check distance to shower centroids in subsequent N layers



Also look for track-like segments at start of cluster and try to match to end of another cluster

Cluster association III : MIP segments

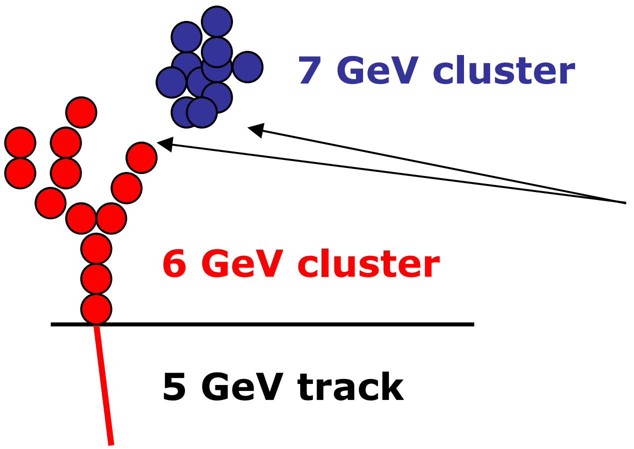
★Look at clusters which are consistent with having tracks segments and project backwards/forward



★Apply tight matching criteria on basis of projected track [NB: + track quality i.e. chi2]

Cluster Association Part II

- Have made very clear cluster associations
- Now try "cruder" association strategies
- BUT first associate tracks to clusters (temporary association)
- Use track/cluster energies to "veto" associations, e.g.

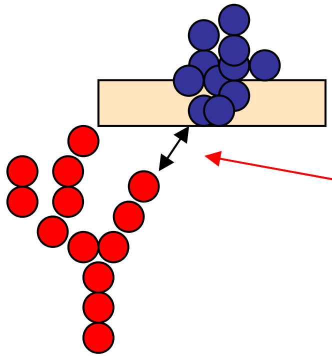


This cluster association would be forbidden if $|E_1 + E_2 - p| > 3\,\sigma_E$

Provides some protection against silly mistakes

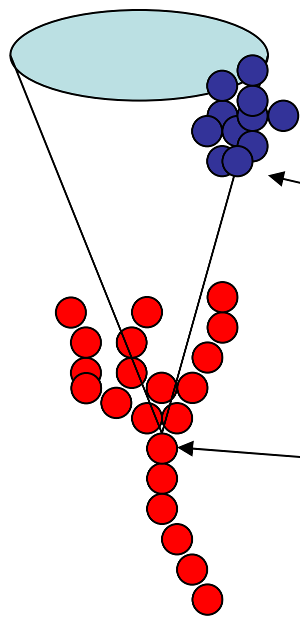
Sledgehammer Cluster Association

Proximity



Distance between hits -limited to first layers

Shower Cone



Associated if fraction of hits in cone > some value

Shower start identified

+Track-Driven Shower Cone

Apply looser cuts if have low E cluster associated to high E track

Performance

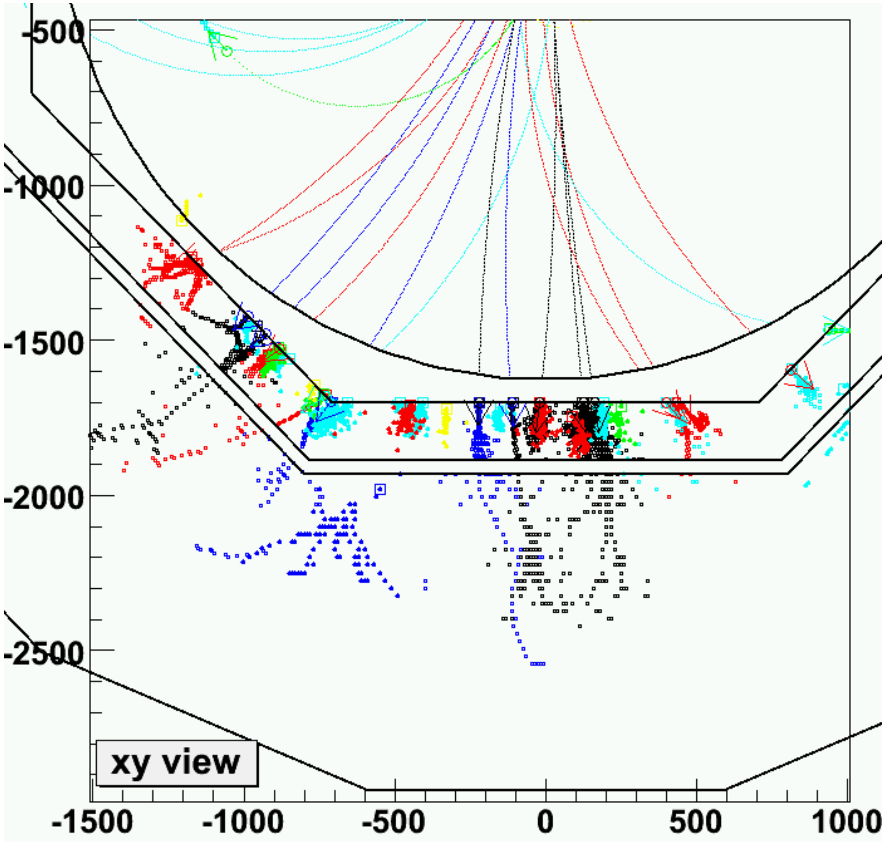
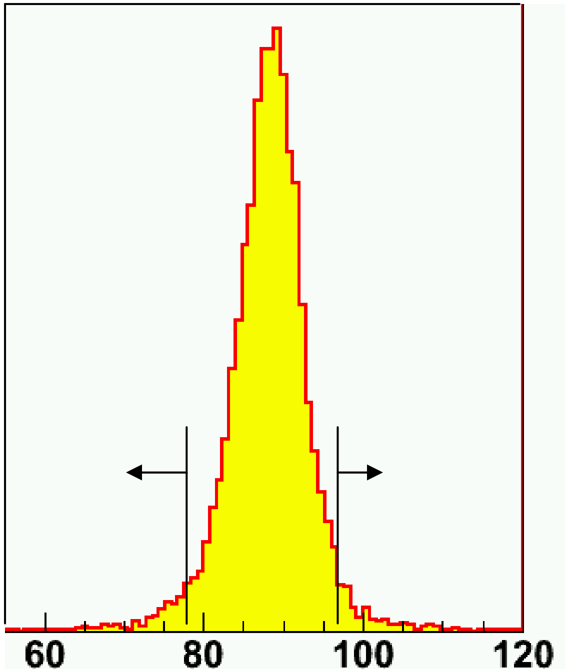


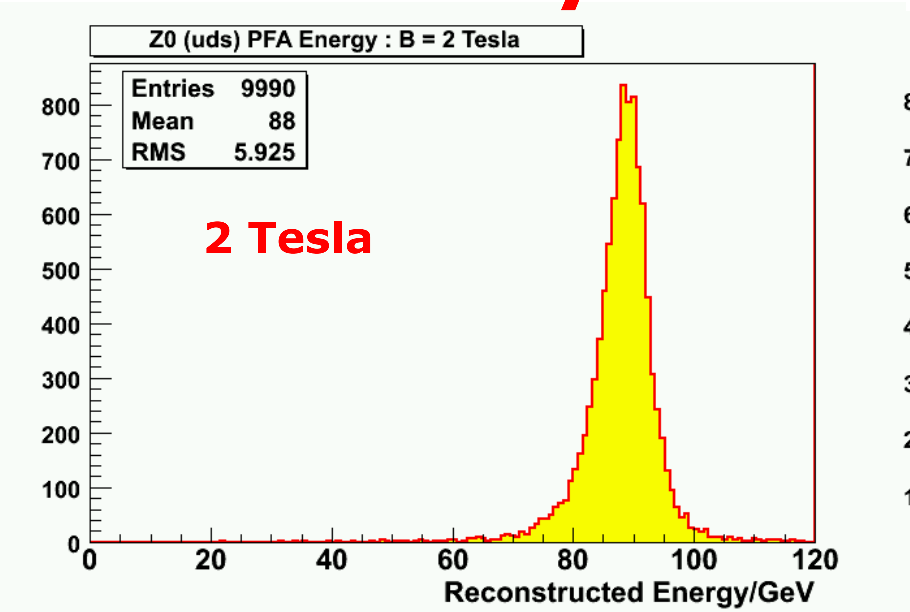
Figure of Merit:

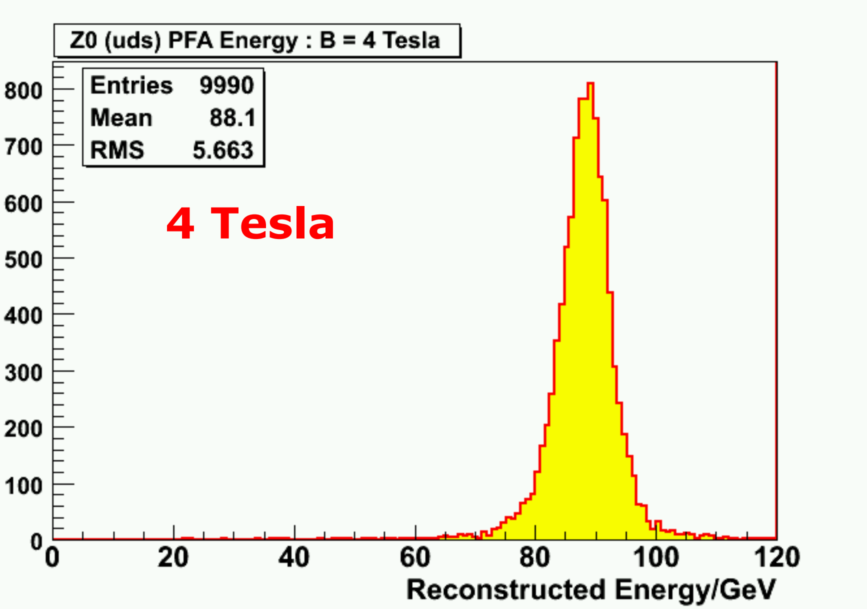


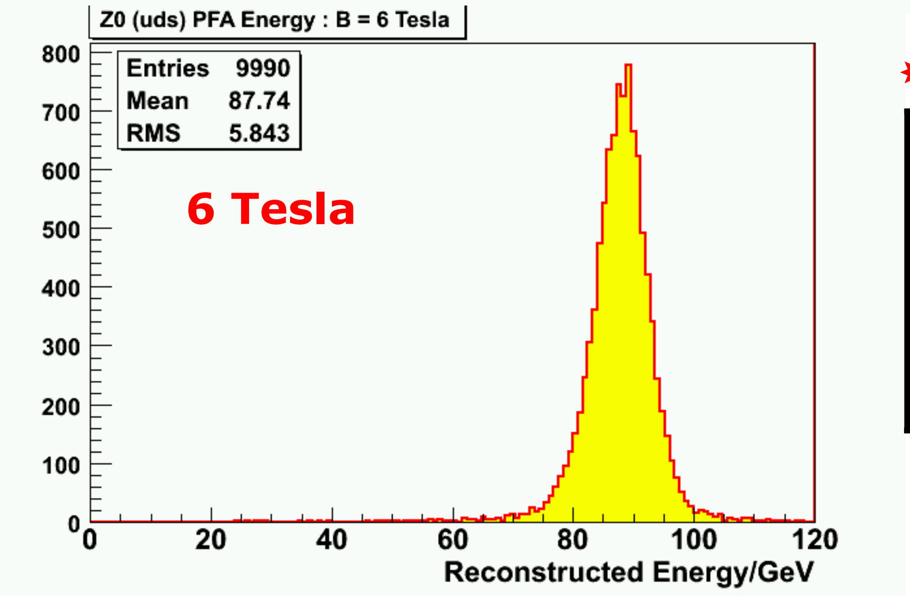
- ★**Find smallest region containing 90 % of events**
- ★**Determine rms in this region**

More robust than fitting double Gaussian

Preliminary Results : Z →uds events







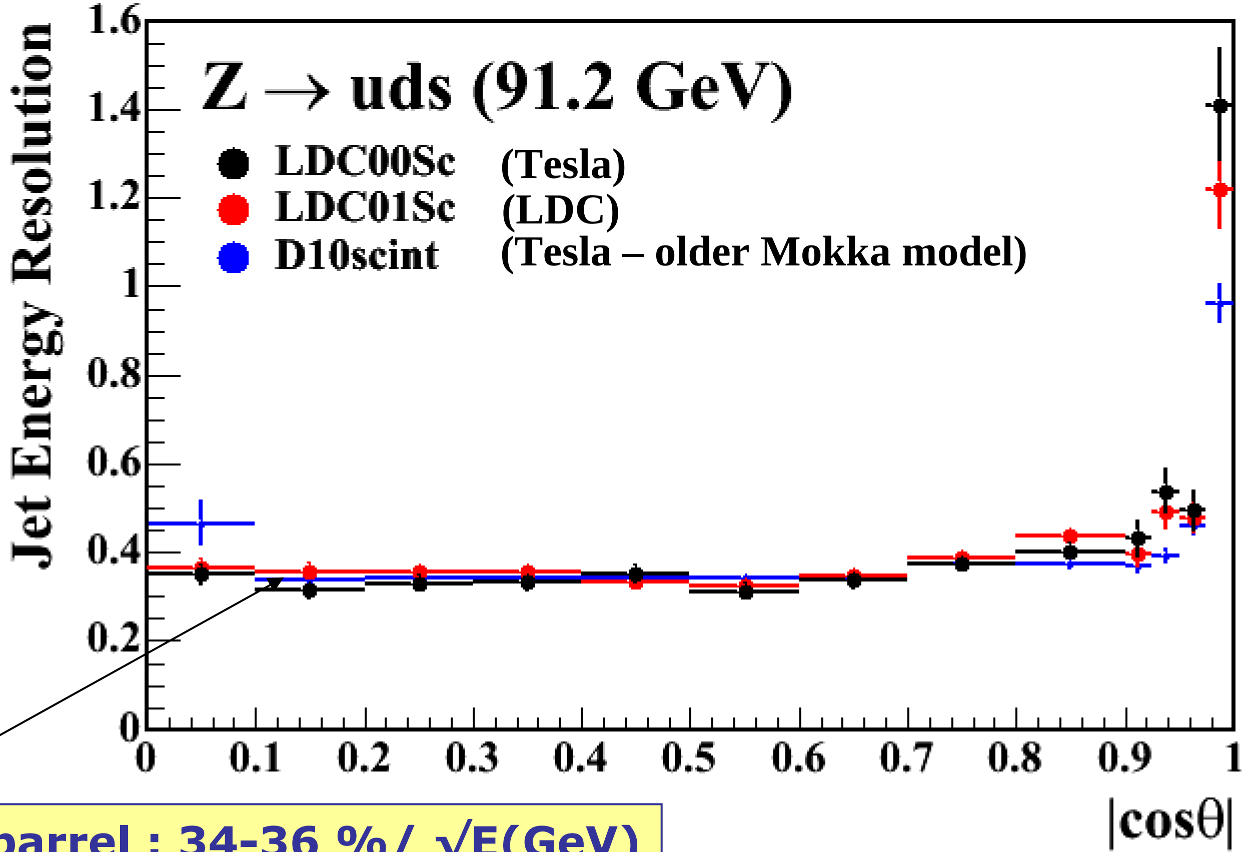
- RMS of Central 90 % of Events

B-Field	$\sigma_E/E = \alpha\sqrt{(E/GeV)}$
2 Tesla	37.8±0.4%
4 Tesla	35.9±0.4 %
6 Tesla	37.4±0.4 %

- only weakly depends on B

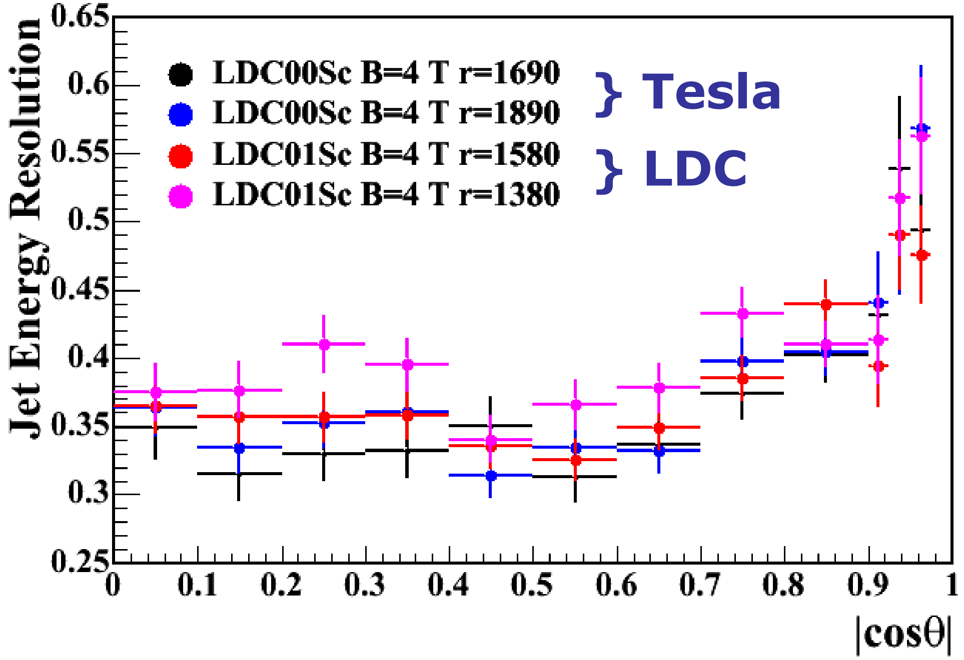
Results : Z uds events
Angular dependence

✦ **Plot resolution vs generated polar angle of qq system**



✦ **In barrel : 34-36 %/ √E(GeV)**

Radial Dependence



Model	all angles	cosθ < 0.8
Tesla r_{tpc} = 1690, l_{tpc}=2730	35.3±0.6 %	33.9±0.7 %
Tesla r_{tpc} = 1890 l_{tpc}=2930	36.6±0.6 %	35.8±0.6 %
LDC r_{tpc} = 1580 l_{tpc}= 2200	37.1±0.6 %	35.7±0.6 %
LDC r_{tpc} = 1380 l_{tpc}= 2000	39.7±0.6 %	38.9±0.7 %

★Some evidence that going to small radii gives worse performance
★BUT… don't take too seriously, Z events + algorithm not perfect

Outlook

- ★ Looks promising - good performance for 91.2 GeV Z events
- ★ Can be improved:
  - ✦ still a few features (i.e. does something silly)
  - ✦ some problems with tracking
  - ✦ photon ID is quite basic
  - ✦ + some new ideas (for high density events)
- ★ Code runs within **Marlin** framework and is "nearly" ready for release - aim to optimise on higher energy jets
- ★ + code needs tidying up
  - ✦ started with decent OO structure
  - ✦ then grew organically...
- ★ Aim to have complete algorithm ~ end April
- ★ Hopefully, soon ready to start full simulation detector optimisation studies