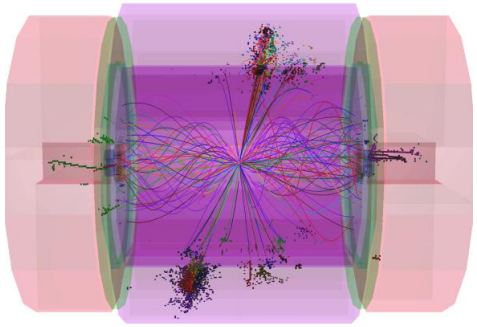


PandoraPFA

Particle Identification & High Energy Performance

John Marshall,
University of Cambridge

ILD Workshop, LAL Orsay, May 24 2011

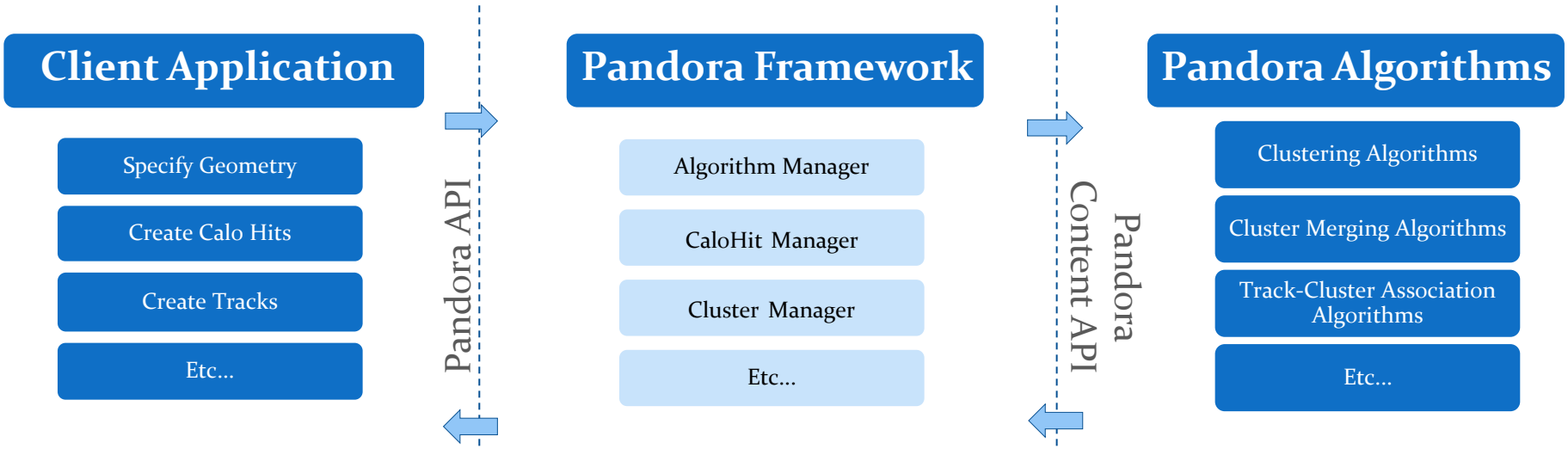
Overview

- Since the last presentation of PandoraPFA in an ILD meeting, many important changes have been made. Some changes were driven by requirements for the CLIC CDR.
  - Jet energy reconstruction has improved, particularly at high energies (250GeV+ jets). The default reconstruction now includes a managed transition to “energy flow” calorimetry.
  - The reconstruction algorithms no longer depend on any “fine details” of the detector model. Most algorithms simply use hit positions and projections of tracks to calorimeter.
  - The reconstruction now offers particle identification functionality.
  - The reconstruction is faster and more efficient.

- Today, will present the following:
  - A quick review of the Pandora software structure (technical, but necessary).
  - An overview of the particle identification currently offered by Pandora, describing particle id “helper functions” and full particle reconstruction algorithms.
  - The latest performance figures for reconstruction of jets in ILD.

Pandora Structure

Pandora is divided into three sections. Communication between these sections is achieved via C++ APIs, Application Programming Interfaces, which each provide a "high-level" service:



Isolates specific details of software framework and detector, allowing framework to be dependency-free.

Specifies parameters for calo hits, tracks, etc. so that Pandora objects can be self-describing.

Owns named collections of Pandora objects: calo hits, tracks, clusters and PFOs.

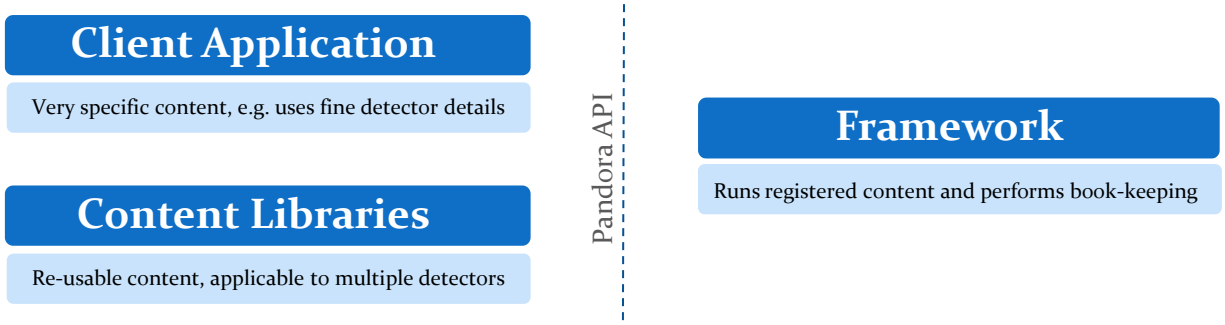
Can perform memory management, as content can only be provided or accessed via APIs.

Use APIs to access Pandora objects and carry out particle flow reconstruction tasks.

Physics-driven code, with nested structure promoting re-use of code to perform specific tasks.

Pandora Structure

- Pandora is a C++ development framework for algorithms, which perform the particle flow reconstruction, aided by helper functions and other Pandora "content".
- A powerful feature of the framework is the ability to register content from different libraries and combine their functionality in the final reconstruction.



- The idea is that each Pandora client application registers the content it needs to perform its specific reconstruction within the framework.
- Content can often be re-used for different detector models, so can be bundled together. ILD-applicable content lives in the FineGranularityContent library, which offers 60+ algorithms, particle id functions, a pseudolayer calculator and a shower-profile calculator.
- FineGranularity content assumes an inner tracker, fine ECAL, coarser HCAL and a coarse yoke.

Algorithms & Particle Id Functions

- Pandora algorithms are responsible for tagging particle flow objects with a PDG code.
- Particle id helper functions can be registered to aid the algorithms. Simply create a function with a Pandora-defined function prototype, register it and assign it to a specific role:
  - C++: PandoraApi::RegisterParticleIdFunction(*m_pPandora, "MyFastMuonId", &MyClass::MyFastMuonId);
    xml : <MuonFastFunction> MyFastMuonId </MuonFastFunction>

- Algorithms can call the helper functions and decide how to respond to the results.
- Alternatively, algorithms can perform their own particle identification. This can include targeted clustering and id for specific particle types, followed by separation of these particles from other hits/tracks, to reduce confusion for remaining reconstruction.

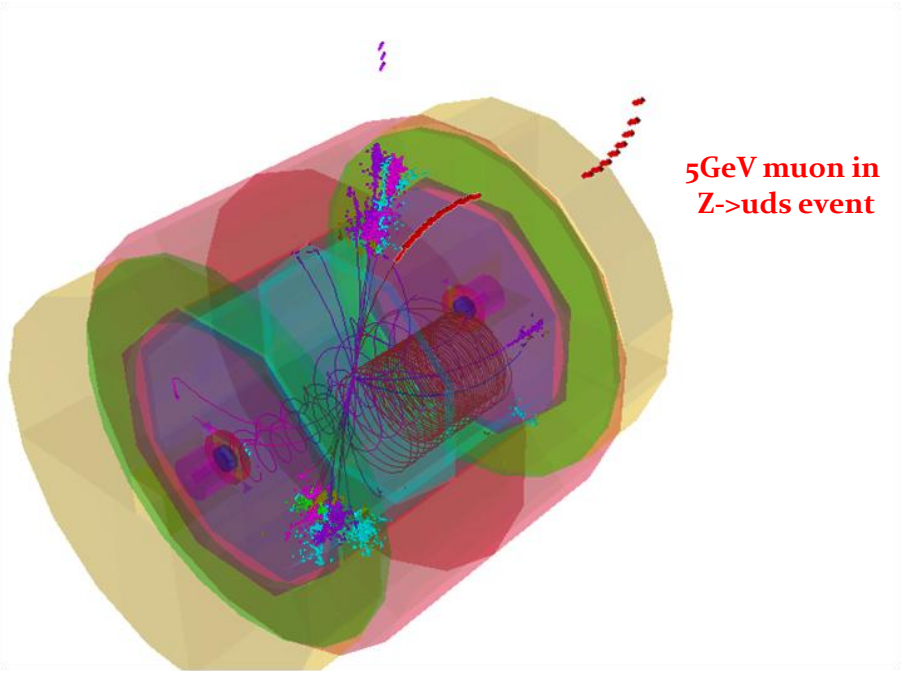
- What happens in default Pandora reconstruction? All of this!
  - Some algorithms (e.g. FragmentRemoval) want to avoid working with objects that look like muons/electrons, so call the fast muon/electron id.
  - Some algorithms (e.g. FinalParticleId) simply call the helper functions and apply the results to the particle flow objects.
  - Some algorithms (e.g. PhotonReconstruction) perform a full reconstruction and identification of specific types of particle.

Fast Muon Id Function

- The fast muon identification is cut-based and looks for an inner detector track, followed by consistent, minimal energy deposition throughout the calorimeters and muon yoke. It targets muons with energy greater than 2.5GeV

Cuts are placed on:

- The number of occupied layers in each of the ECAL, HCAL and YOKE regions.
- The energy deposited in the ECAL and HCAL regions. Energies are direction-corrected and cuts are linear functions of associated track energy.
- The RMS values for straight-line fits in the ECAL, HCAL and YOKE regions.
- The fraction of mip-like hits in the ECAL and HCAL regions.
- The number of muon yoke hits



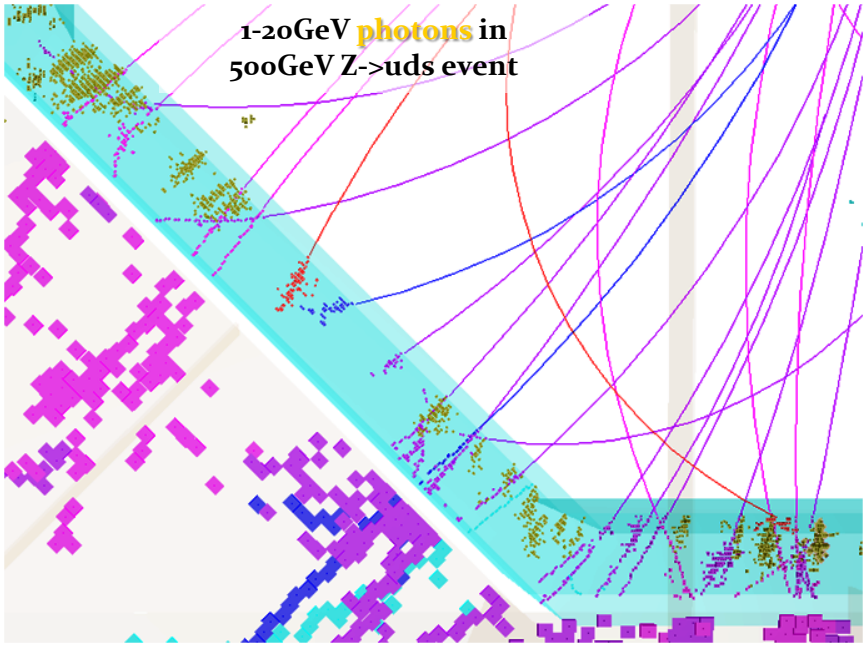
5GeV muon in Z->uds event

Fast Photon Id Function

- The fast photon identification looks for clusters that have no associated tracks and which pass fast electromagnetic shower identification cuts. Much of this code is shared with the fast electron id function.

Cuts are placed on:

- The cluster inner layer, which must lie within the ECAL.
- The fraction of mip-like hits in the cluster.
- The radial direction cosine and RMS, as obtained from a straight-line fit.
- The cluster longitudinal shower profile. Cuts are applied to the number of radiation lengths before the ShowerStart, Layer90 and ShowerMax layers.
- The cluster transverse shower profile. Cuts are applied to the Radial90 distance.

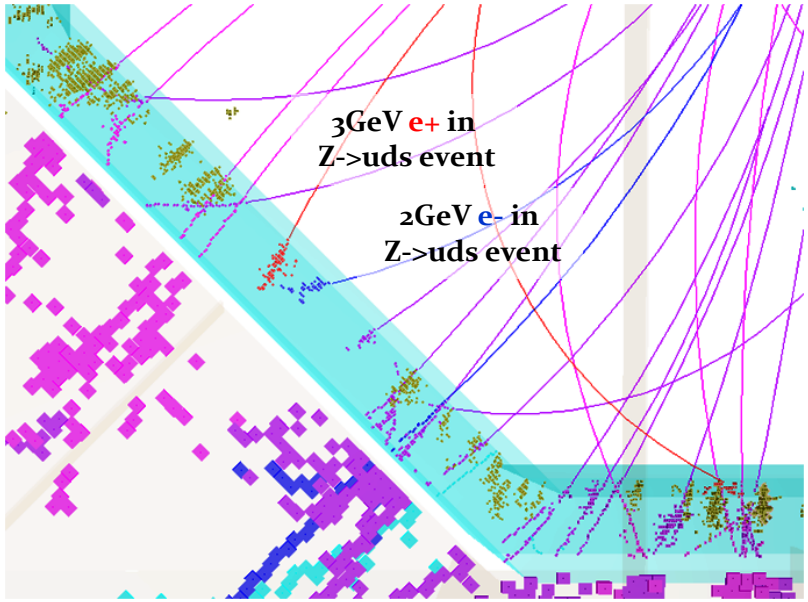


Fast Electron Id Function

- The fast electron identification requires that a cluster has an associated inner detector track and that it passes the electromagnetic shower identification, previously described.

Cuts are placed on:

- The number of radiation lengths before the observed start of the longitudinal shower profile.
- The discrepancy between the observed shower profile and the expectation for an electromagnetic shower.
- The absolute value of $(E_{\text{cluster}}/P_{\text{track}} - 1)$ for the cluster/track pairing.

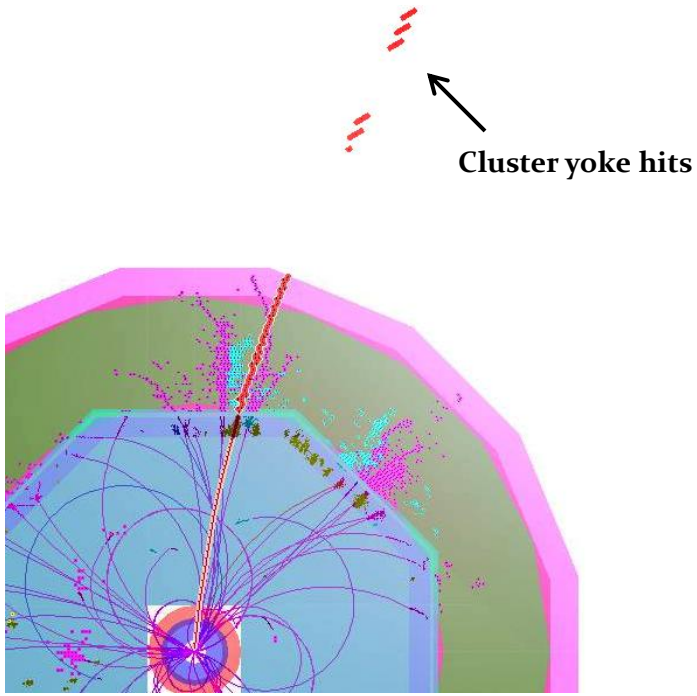


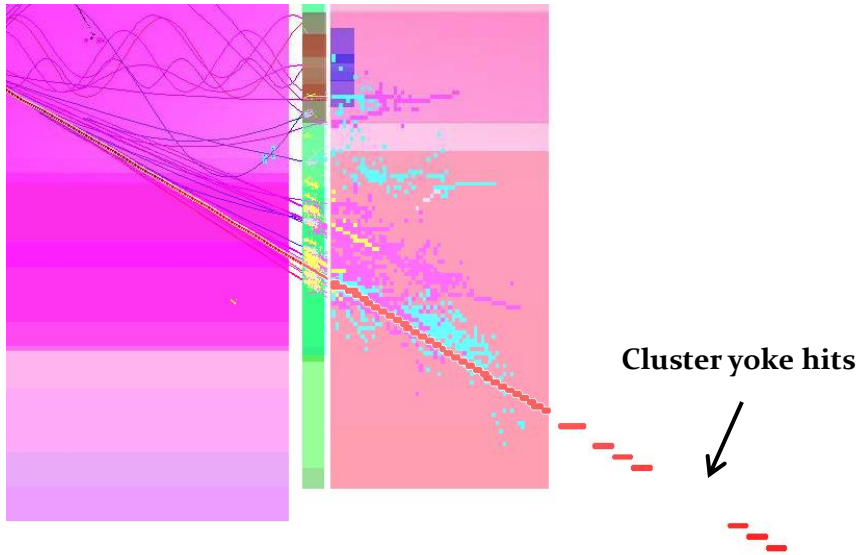
- Any cluster not identified a muon, electron or photon is assumed to be a pion (associated track), or a neutron (no associated track). A specific algorithm searches for Vo's.

Muon Reconstruction Algorithm

The muon reconstruction algorithm was designed in collaboration with Erik van der Kraaij. It aims to improve the efficiency for identifying high energy muons as follows:

1. Yoke track candidates are identified using an instance of the Pandora cone-based clustering algorithm, configured appropriately for the coarse instrumentation in this region. Clusters crossing all yoke layers, whilst containing a minimal number of hits are selected.



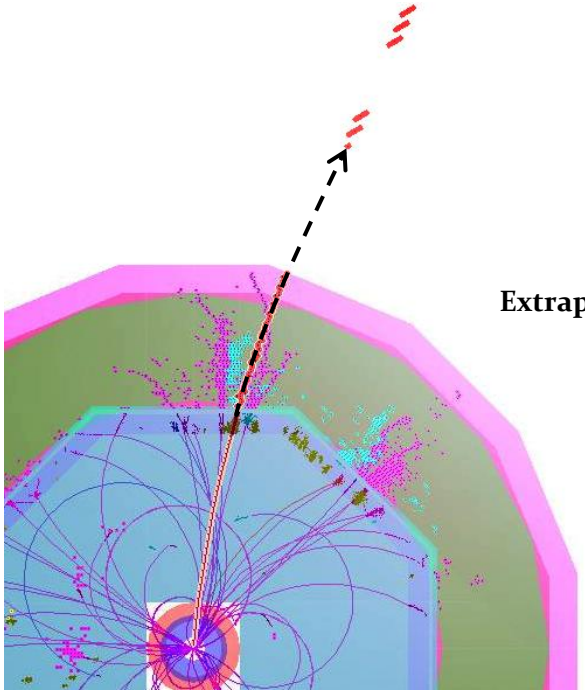


Muon Reconstruction Algorithm

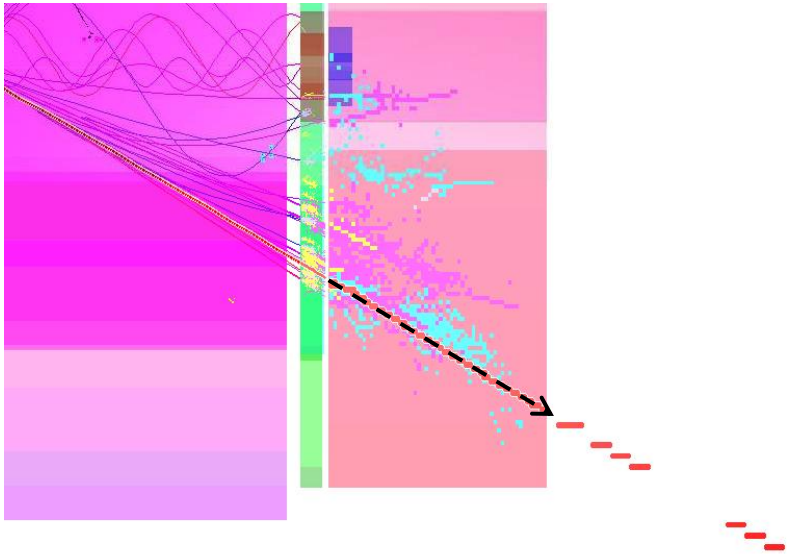
2. For each inner detector track above 7GeV, a helix fit to the track is extrapolated to the position of each yoke cluster. This extrapolation accounts for changes in the B-field upon crossing the coil.

   The helix extrapolation is used to calculate the distance of closest approach to the yoke cluster and also the angle between the helix direction and a linear fit to the cluster.

   Track candidates with opening angles greater than 0.2rad, or distances greater than 200mm are excluded. The closest track is selected and used to calculate the muon properties.

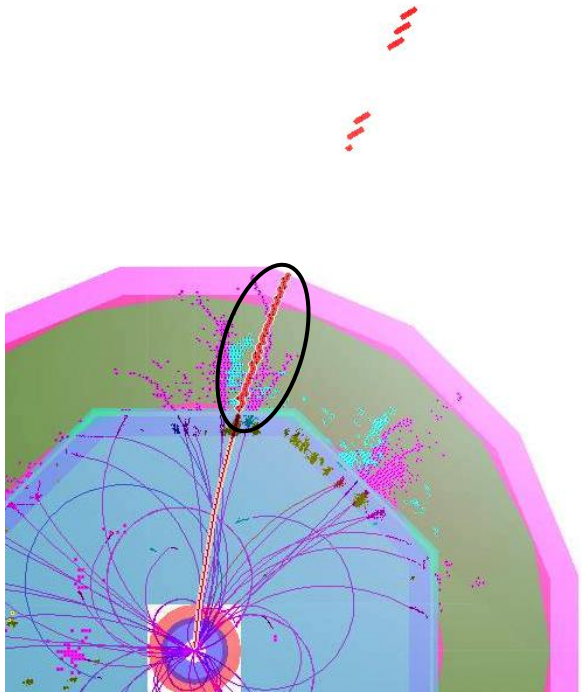


Extrapolate inner detector tracks to yoke

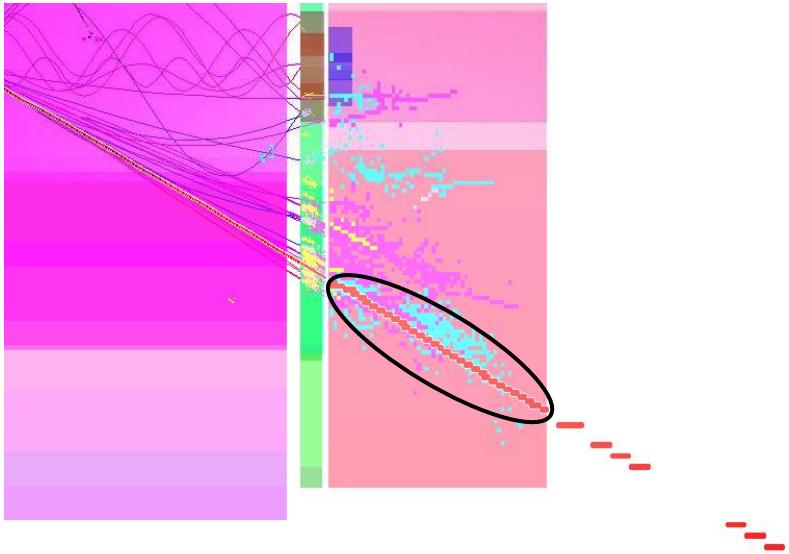


Muon Reconstruction Algorithm

3. The helix fit is projected through the calorimeters to identify remaining hits from the muon. Hits are added to the reconstructed muon based upon distance from the helix. If the muon is not deemed to be isolated (based on number of nearby hits), only the single closest hit in each layer is added. If the muon is isolated, all nearby hits (within a certain distance) are included. Finally, all muon components (hits/tracks) are removed from the subsequent reconstruction.



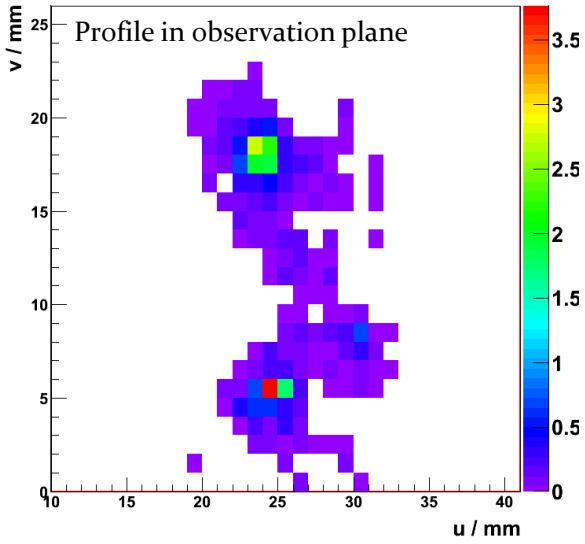
Select calorimeter hits

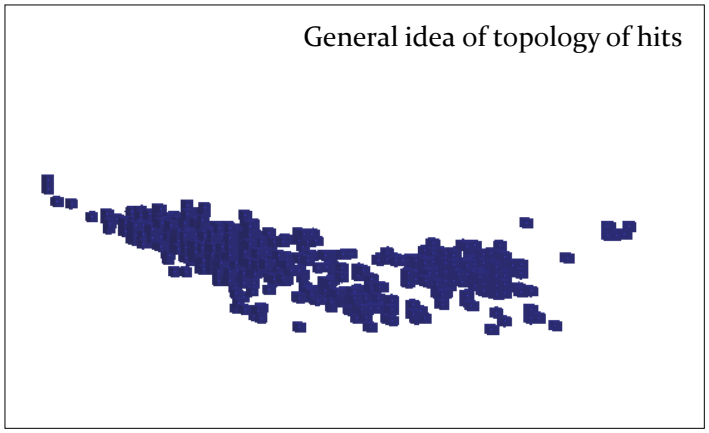


Photon Reconstruction Algorithm

The photon reconstruction aims to reconstruct, tag and remove all photons before the standard Pandora reconstruction, reducing confusion and improving the jet energy reconstruction.

1. The cone-based clustering algorithm is applied to the ECAL hits, with all of its track-seeding options disabled. The transverse shower profiles of the clusters are then examined in detail. Any peaks in the profile are identified and characterised.

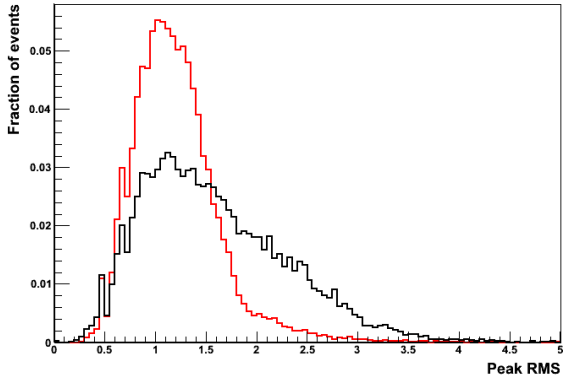


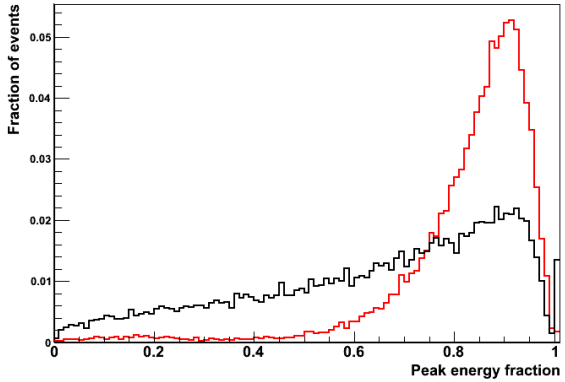


Not originally identified as a photon, but 17GeV from the 30GeV cluster is actually from a true photon. This is evident from the profile – split cluster up.

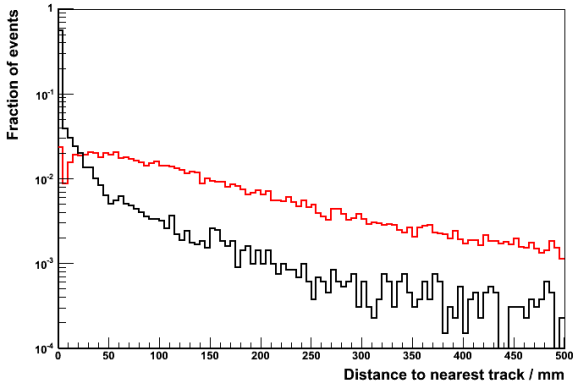
Photon Reconstruction Algorithm

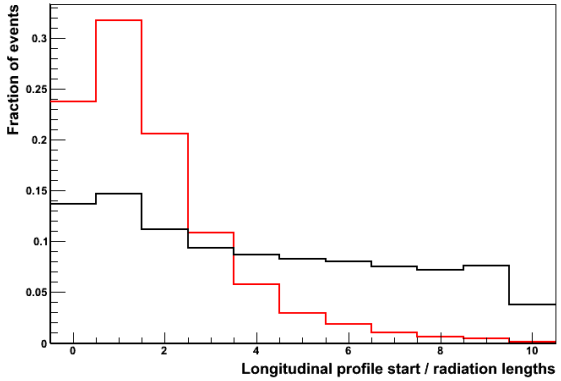
2. For each peak, a new photon cluster candidate is created and examined. Cuts are placed on the longitudinal shower profile of the new cluster and a multivariate/PID analysis is used to decide whether to accept the cluster as a photon.

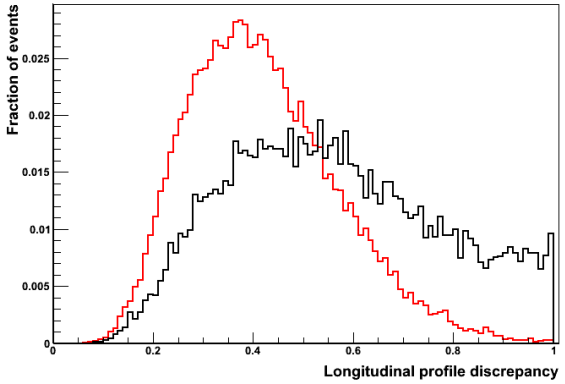




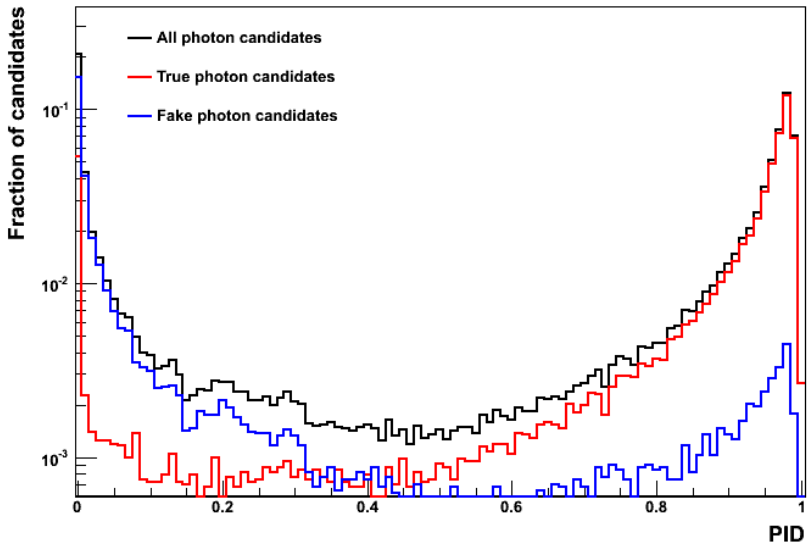
PDFs used for photon identification, constructed using 500GeV ILD00 $Z \rightarrow uds$ events

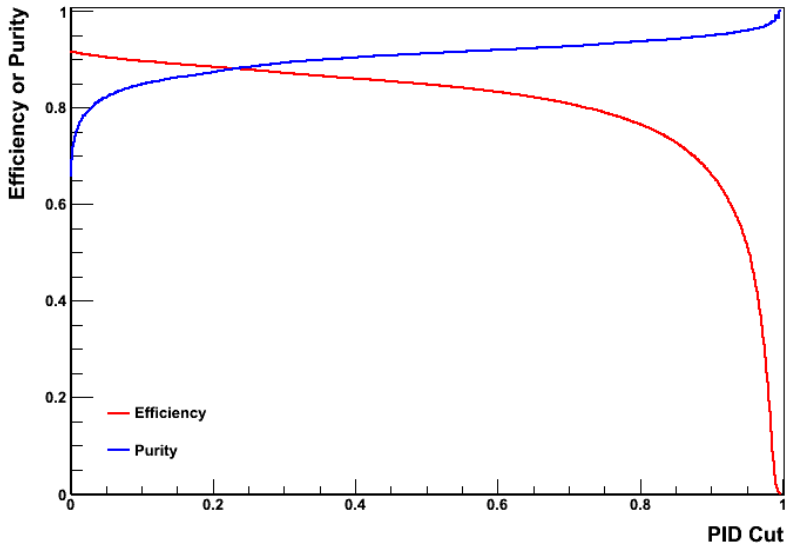






Photon Reconstruction Algorithm





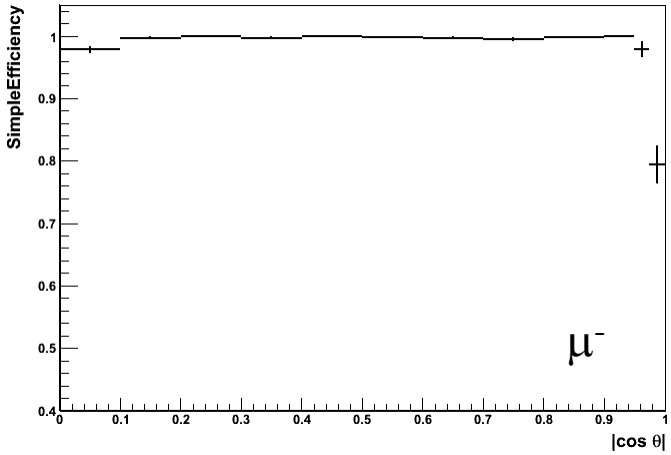
3. If a peak cluster is accepted, it is tagged as a photon and saved; the original cluster is deleted. If the peak represents the majority of the energy in original cluster, original may be used instead.

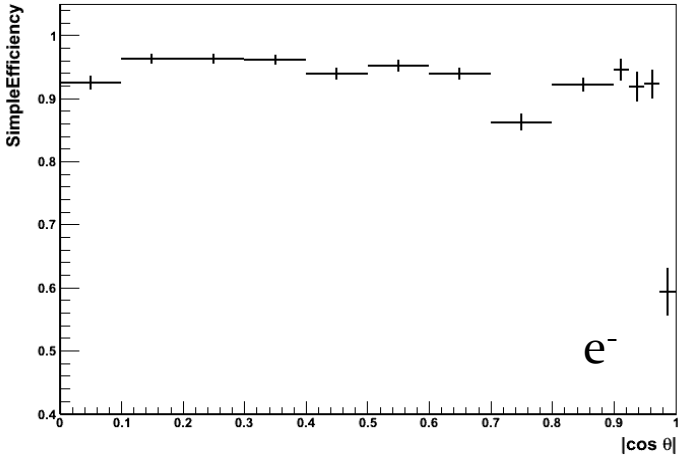
   With the exception of the addition of isolated hits, the photon clusters can remain unchanged and can be used to form photon particle flow objects in the PfoCreation algorithm.

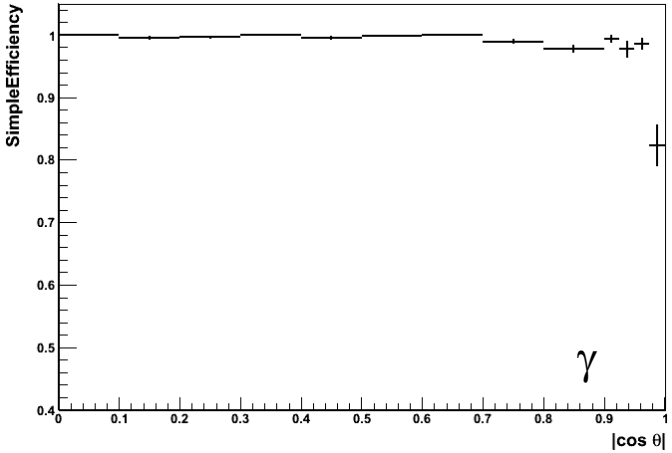
Will later examine the impact of the photon reconstruction algorithm on jet energy reconstruction.

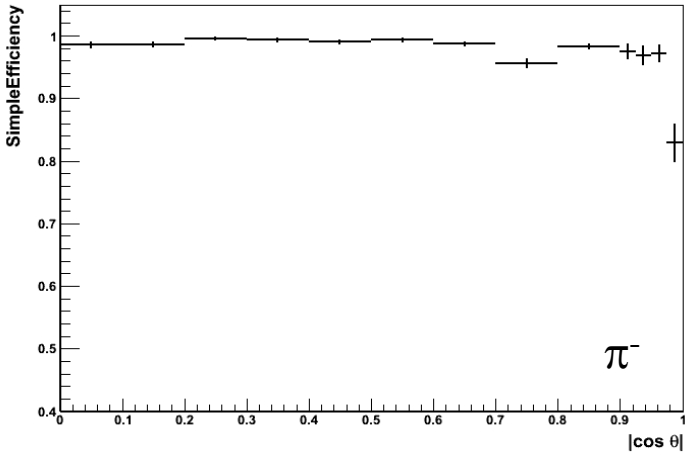
Efficiencies

Efficiency of single particle identification in ILDoo, as a function of |cos(θ)|



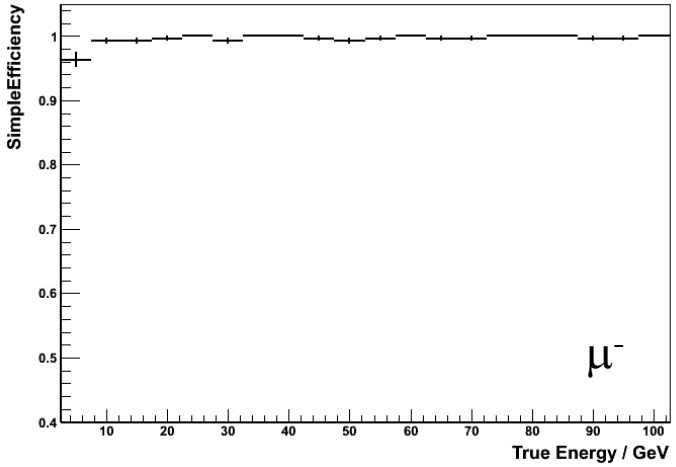


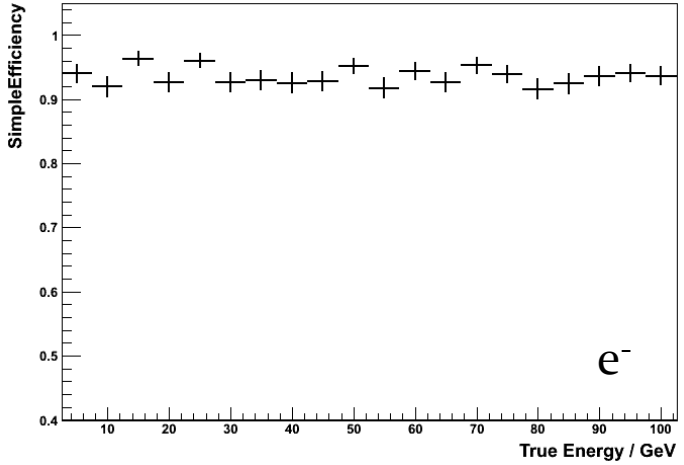


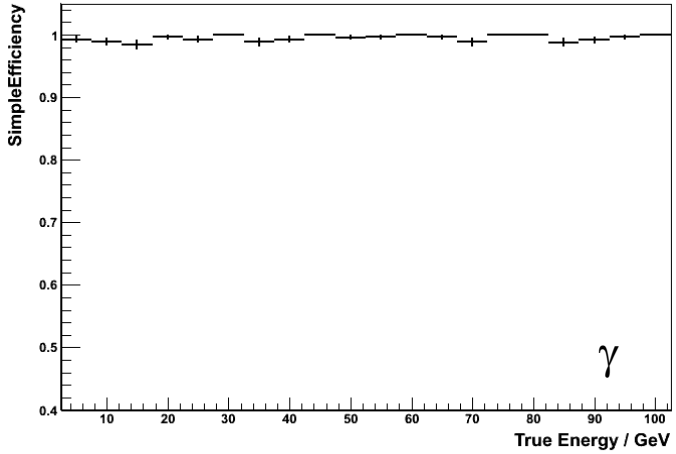


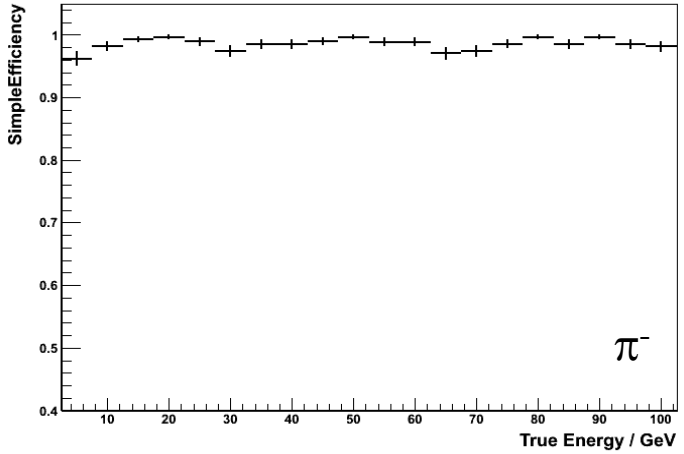
Efficiencies

Efficiency of single particle identification in ILDoo, as a function of true energy, |cos(θ).<0.9|









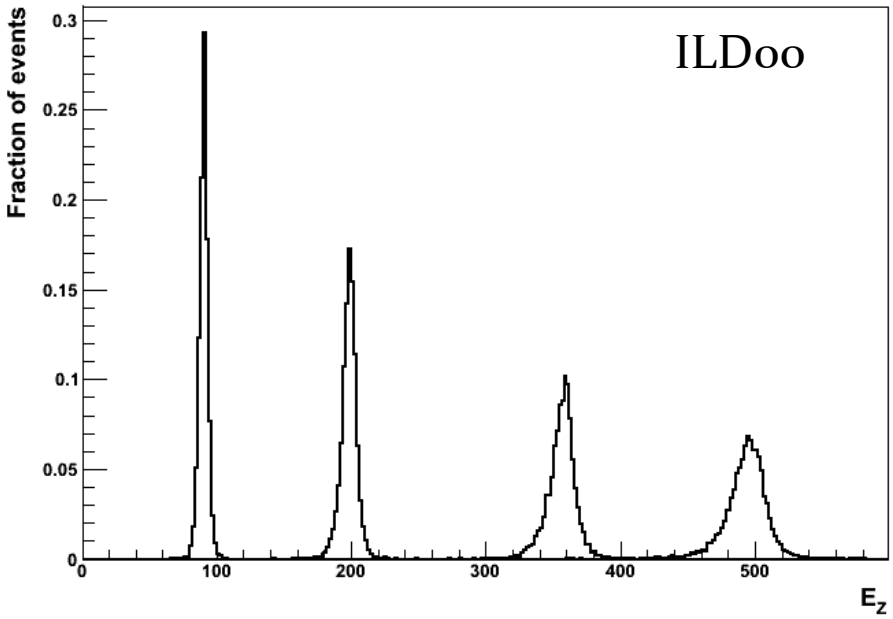
Jet Energy Performance

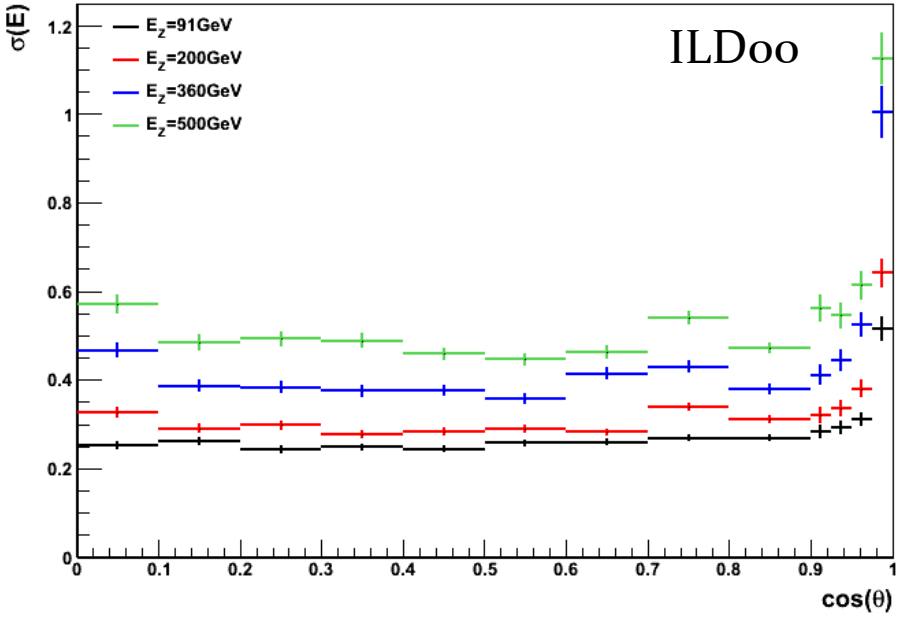
- Performance studied for ILD00, using MC samples of 10,000 $Z \rightarrow uds$ events.
- Performance quoted in terms of rms_{90}, which is defined to be the rms in the smallest range of reconstructed energy containing 90% of the events.
- A cut on the polar angle is applied to avoid the barrel/endcap overlap region: $|\cos\theta| < 0.7$

- Muon reconstruction algorithm confirmed to have negligible impact on jet energy reconstruction, as should be expected.
- Use of the photon reconstruction algorithm at the start of the reconstruction provides an improved jet energy reconstruction at high energies.
- Further improvements can be made to photon reconstruction algorithm to successfully deal with high energy photons that leak into the HCAL.

ILD00, E_z (= 2 * E_j)	91GeV	200GeV	360GeV	500GeV
Standard Pandora, $rms_{90}(E_j) / E_j$	3.64 ± 0.05	2.93 ± 0.04	3.00 ± 0.04	3.19 ± 0.04
Photon Clustering, $rms_{90}(E_j) / E_j$	3.69 ± 0.05	2.88 ± 0.04	2.91 ± 0.04	3.03 ± 0.05

Jet Energy Performance

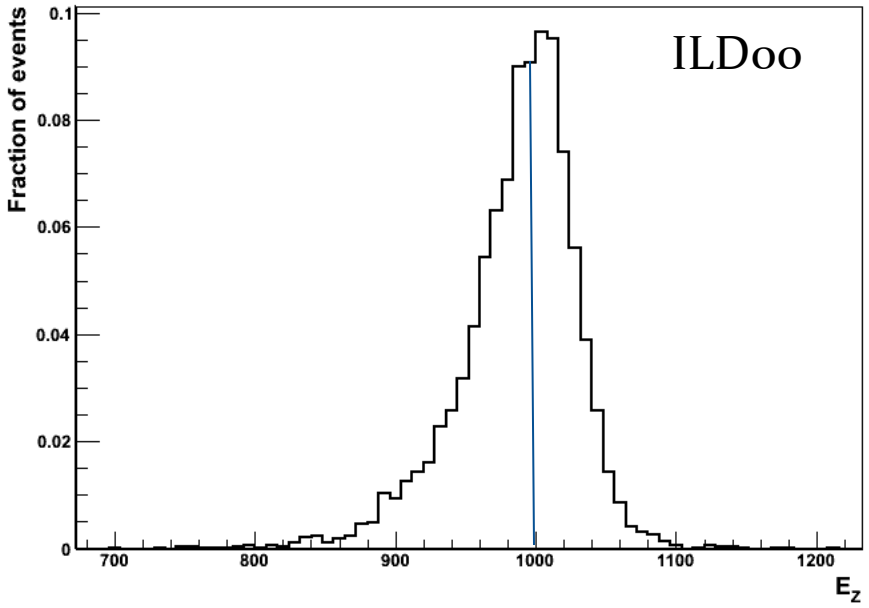


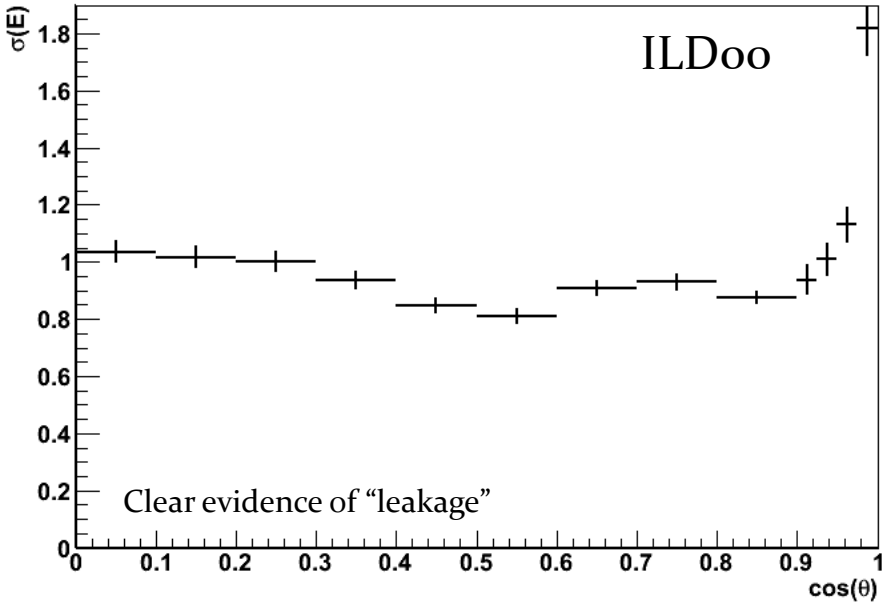


ILDoo, E_z (= 2 * E_j)	91GeV	200GeV	360GeV	500GeV
Photon Clustering, $rms_{90}(E_j)$ / E_j	3.69 ± 0.05	2.88 ± 0.04	2.91 ± 0.04	3.03 ± 0.05

Best ever Pandora performance for ILDoo

High Energy Performance





E_z (= 2 * E_j)	91GeV	200GeV	360GeV	500GeV	1TeV
ILDoo, $rms_{90}(E_j)$ / E_j	3.69 ± 0.05	2.88 ± 0.04	2.91 ± 0.04	3.03 ± 0.05	**4.16 ± 0.06**
CLIC_ILD_CDR, $rms_{90}(E_j)$ / E_j	3.73 ± 0.05	2.99 ± 0.04	-	2.86 ± 0.05	**3.07 ± 0.05**

Photon reconstruction used for both detector models. Beware different reconstruction chains for ILDoo, CLIC_ILD_CDR, making for difficult direct comparisons (e.g. different digitization).

Summary

- The Pandora framework provides a number of ways to perform particle identification. Algorithms make the final decisions, but they can be supported by plugin helper functions.
- Algorithms and helper functions can be non detector-specific, and so re-usable, or they can rely on direct knowledge of detector details. They can even include external software dependencies.

- Provided with Pandora are a number of particle identification functions and algorithms that are designed for use with any Fine Granularity particle flow detector.
- These include fast muon, electron and photon identification functions, used throughout the default Pandora reconstruction.
- Also included are muon and photon reconstruction algorithms, which aim to fully reconstruct and tag specific particles and remove them from the subsequent reconstruction.

- The photon reconstruction algorithm proves beneficial at high energies, improving jet energy reconstruction performance.
- Current jet energy performance figures are the best ever obtained for the ILDoo detector model. However, for 500GeV jets, the reconstruction suffers from leakage problems.

Backup

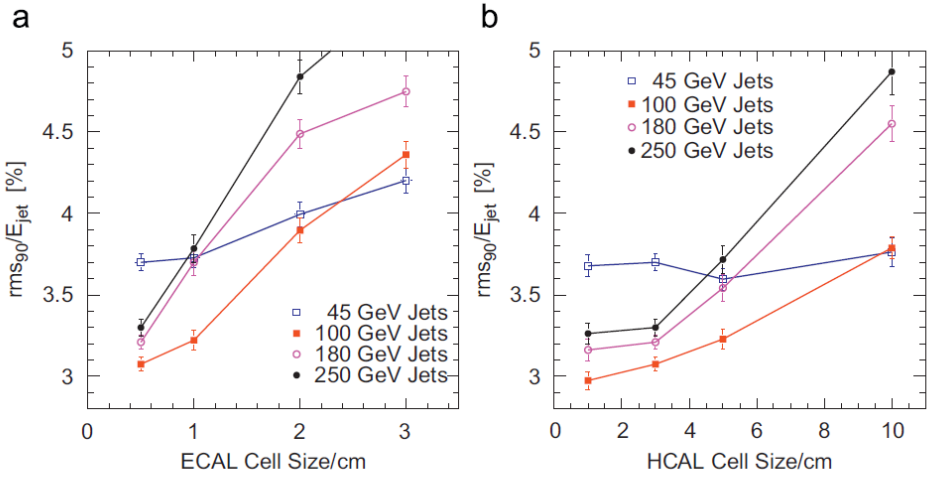


Fig. 13. (a) The dependence of the jet energy resolution (rms_{90}) on the ECAL transverse segmentation (Silicon pixel size) in the LDCPrime model and (b) the dependence of the jet energy resolution (rms_{90}) on the HCAL transverse segmentation (scintillator tile size) in the LDCPrime model. The resolutions are obtained from $Z \to u\bar{u}, d\bar{d}, s\bar{s}$ decays at rest. The errors shown are statistical only.

Extracts from
NIMA 611 (2009) 25-40

Results now superseded, but change in trends unlikely

Comparison of the jet energy resolution obtained using different hadronic shower physics lists.

Physics list	Jet energy resolution $r = rms_{90}(E_j)/E_j$			
	45 GeV	100 GeV	180 GeV	250 GeV
LCPhys (%)	(3.74 ± 0.05)	(2.92 ± 0.04)	(3.00 ± 0.04)	(3.11 ± 0.05)
QGSP_BERT (%)	(3.52 ± 0.06)	(2.95 ± 0.06)	(2.98 ± 0.06)	(3.25 ± 0.07)
QGS_BIC (%)	(3.51 ± 0.06)	(2.89 ± 0.05)	(3.12 ± 0.07)	(3.20 ± 0.07)
FTFP_BERT (%)	(3.68 ± 0.08)	(3.10 ± 0.06)	(3.24 ± 0.06)	(3.26 ± 0.08)
LHEP (%)	(3.87 ± 0.07)	(3.15 ± 0.06)	(3.16 ± 0.06)	(3.08 ± 0.06)
χ^2 (4 d.o.f)	23.3	17.8	16.0	6.3
rms/mean ($\sigma_r/\bar{r}$) (%)	4.2	3.9	3.5	2.5

The χ^2 consistency of the different models for each jet energy are given as are the rms variations between the five models.