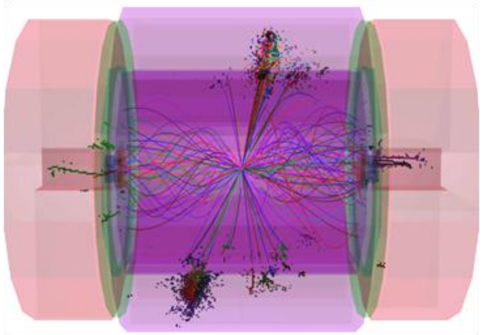


Current Status of PandoraPFA

John Marshall,

University of Cambridge

LCWS11, Granada, September 27 2011

Overview

- At the time of LCWS10, Pandora was undergoing an extensive redesign and reimplementation.
- The aim was to convert Pandora from a proof-of-principle into a robust, elegant and efficient framework for designing and running decoupled algorithms for particle flow calorimetry.
- This talk will:
  - Briefly summarise the status at LCWS10.
  - Review some development highlights leading to the CLIC CDR release.
  - Discuss the options for customising and working with Pandora.
  - Describe recent updates to Pandora since the CLIC CDR release.

Status @ LCWS10

Client Application

- Specify Geometry
- Create Calo Hits
- Create Tracks
- Etc...

Pandora API

Pandora Framework

- Algorithm Manager
- CaloHit Manager
- Cluster Manager
- Etc...

Pandora Content API

Pandora Algorithms

- Clustering Algorithms
- Cluster Merging Algorithms
- Track-Cluster Association Algorithms
- Etc...

Client Application:

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.

Pandora Framework:

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

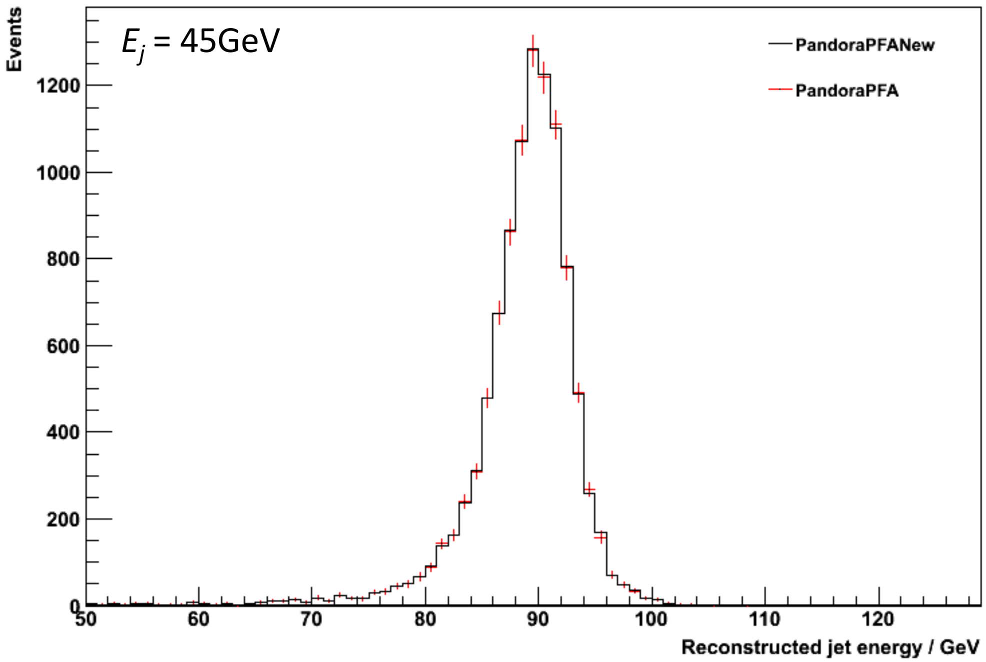
Can perform memory management, as content can only be provided or accessed via APIs.

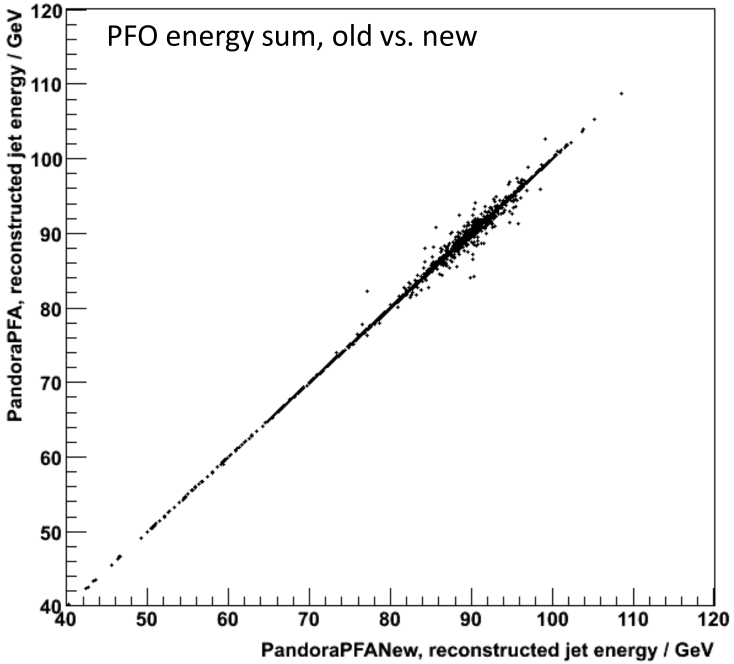
Pandora Algorithms:

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.

Status @ LCWS10





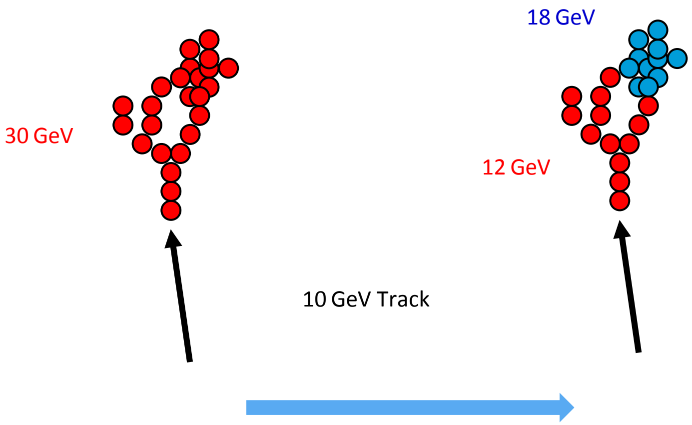
- Tests using ILD00 $Z \rightarrow uds$ events, with E_z = 91.2GeV, exercised new framework and algorithms, without needing specific treatment to resolve 'confusion' that occurs at higher-energies.
- **Excellent agreement observed, by construction.**

Reclustering

To resolve confusion at high energies, Pandora performs statistical reclustering. Hits are redistributed in order to improve the consistency between cluster energies and associated track momenta.

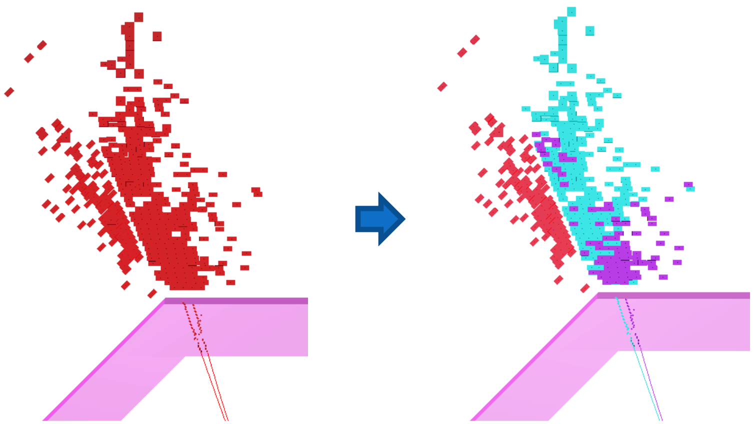
Algorithm operations:

1. Identify inconsistent pairing of track and cluster(s) and ask to recluster these.
   - Relevant clusters moved to new temporary cluster list. Current hit/track lists changed.
2. Ask to run a clustering algorithm.
   - Creates another uniquely named temporary cluster list, filled by daughter clustering algorithm.
3. Calculate figure of merit for consistency of track and new cluster(s).
4. Repeat stages 2. and 3. as required.
   - Can re-use original clustering algorithm, with different parameters, or try entirely new algorithms.
5. Choose most appropriate cluster(s).
   - All lists will be reorganised and tidied accordingly.

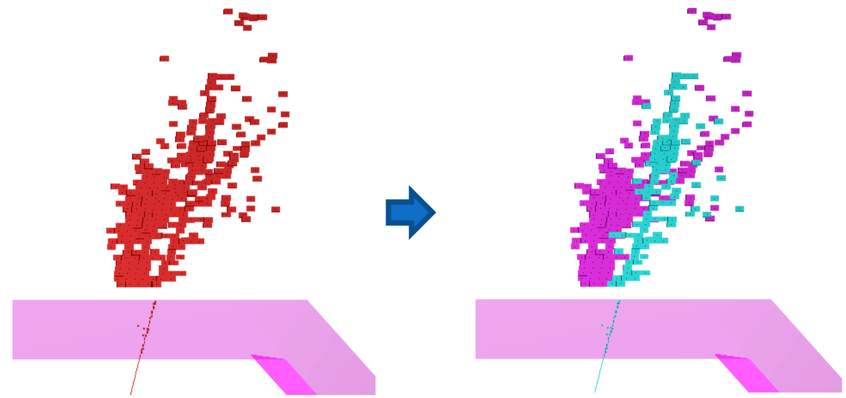


Change clustering parameters and/or clustering algorithm until cluster splits and get sensible track-cluster match

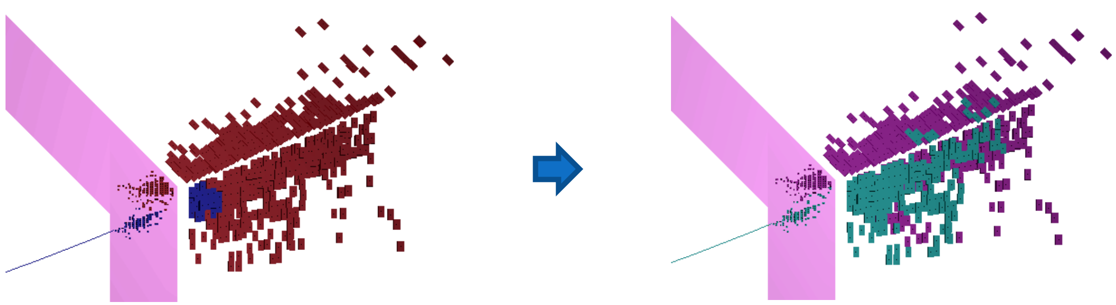
Reclustering Strategies



1. Multiple tracks associated to single cluster – split cluster.

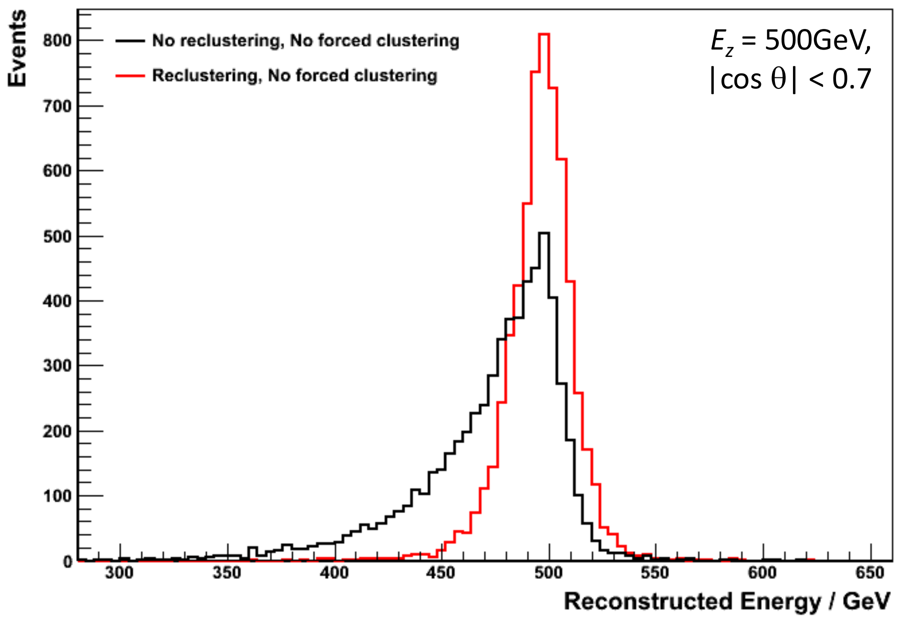


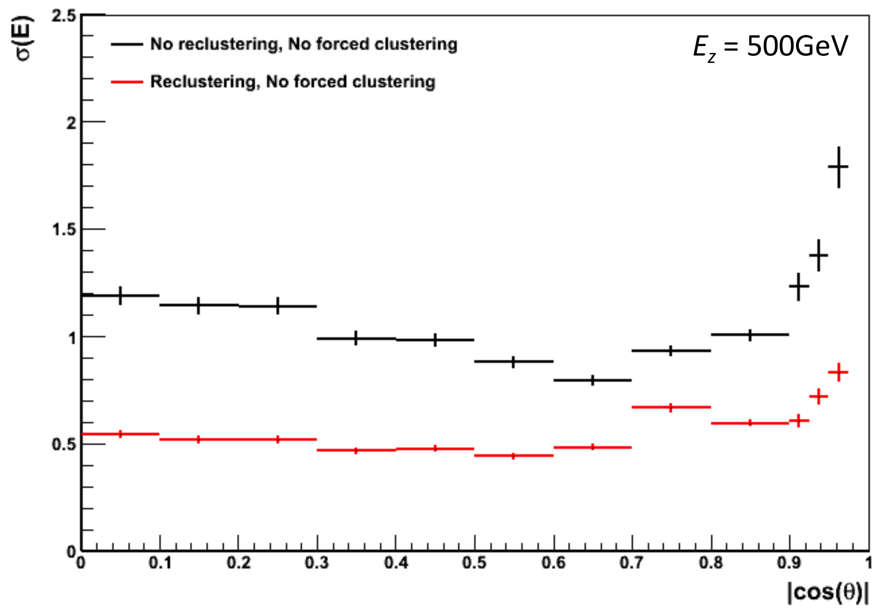
2. Cluster energy much greater than track momentum – split cluster.



3. Track momentum much greater than cluster energy – bring in nearby clusters and reconfigure.

Reclustering Results

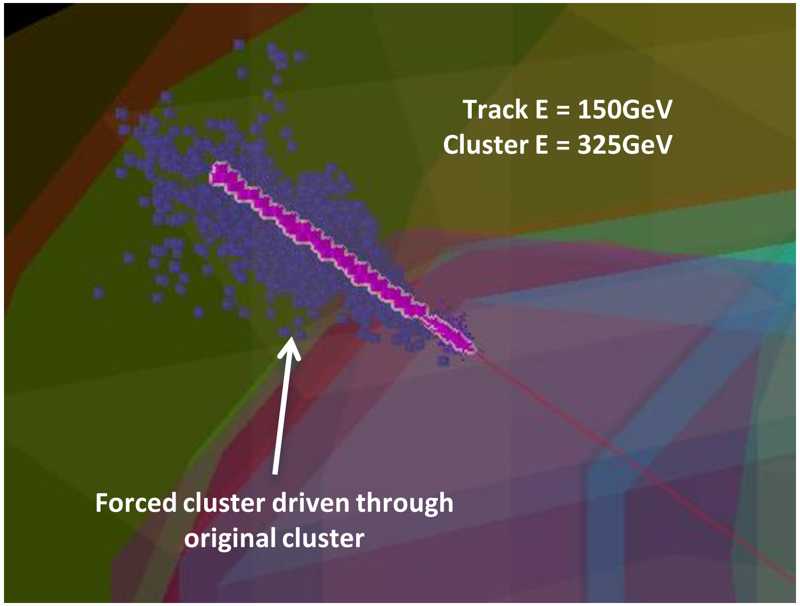




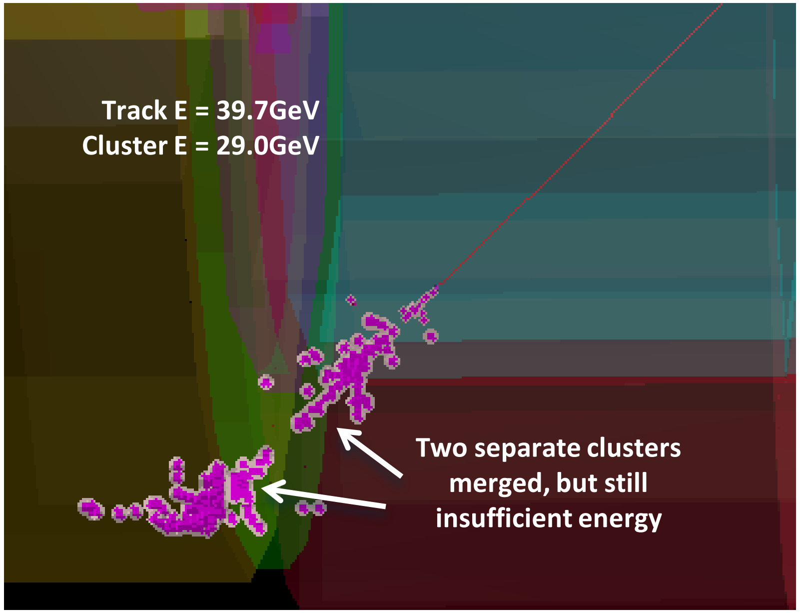
CLIC_ILD_CDR, E_z (= 2 * E_j)	91GeV	200GeV	500GeV	1TeV
No Reclustering, No Forced Clustering, $rms_{90}(E_j) / E_j$	3.75 ± 0.05	3.72 ± 0.05	6.33 ± 0.08	10.33 ± 0.18
Reclustering, No Forced Clustering, $rms_{90}(E_j) / E_j$	**3.71 ± 0.05**	**3.02 ± 0.04**	**3.07 ± 0.04**	**3.88 ± 0.07**

Forced Clustering

- Pandora allows a transition from particle flow to “energy flow” calorimetry if it is evident there is a problem that cannot be fixed by reclustering.
- Simply add Forced Clustering algorithm to the end of list of algorithms to be used in reclustering. For a poorly matched track and cluster, the algorithm will select best hits to force compatible energies.

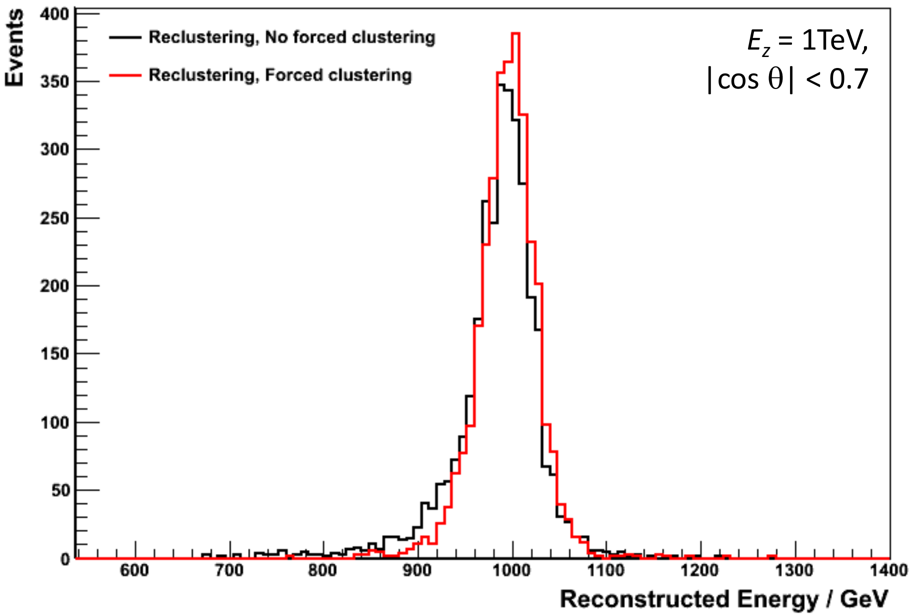


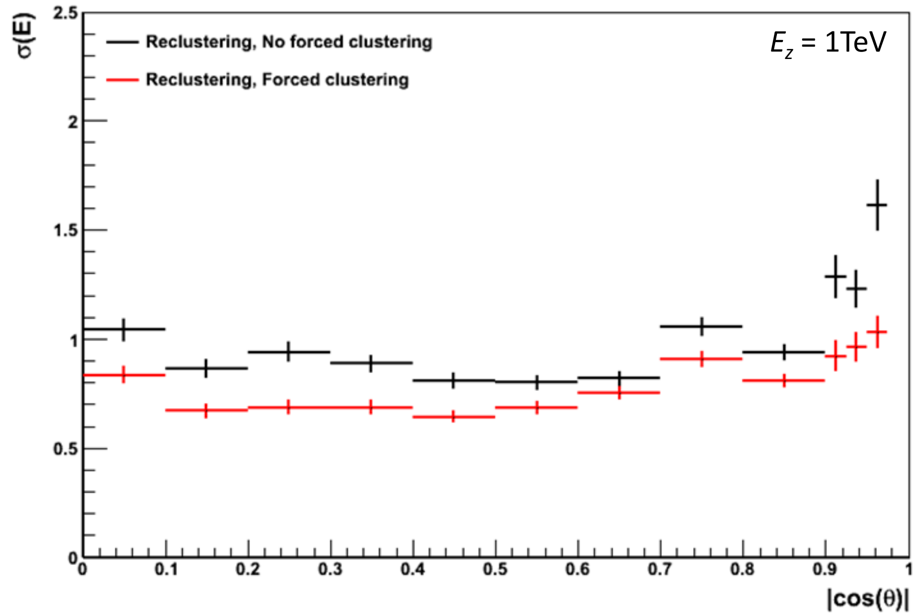
1 . ClusterE > TrackE



2. TrackE > ClusterE

Forced Clustering

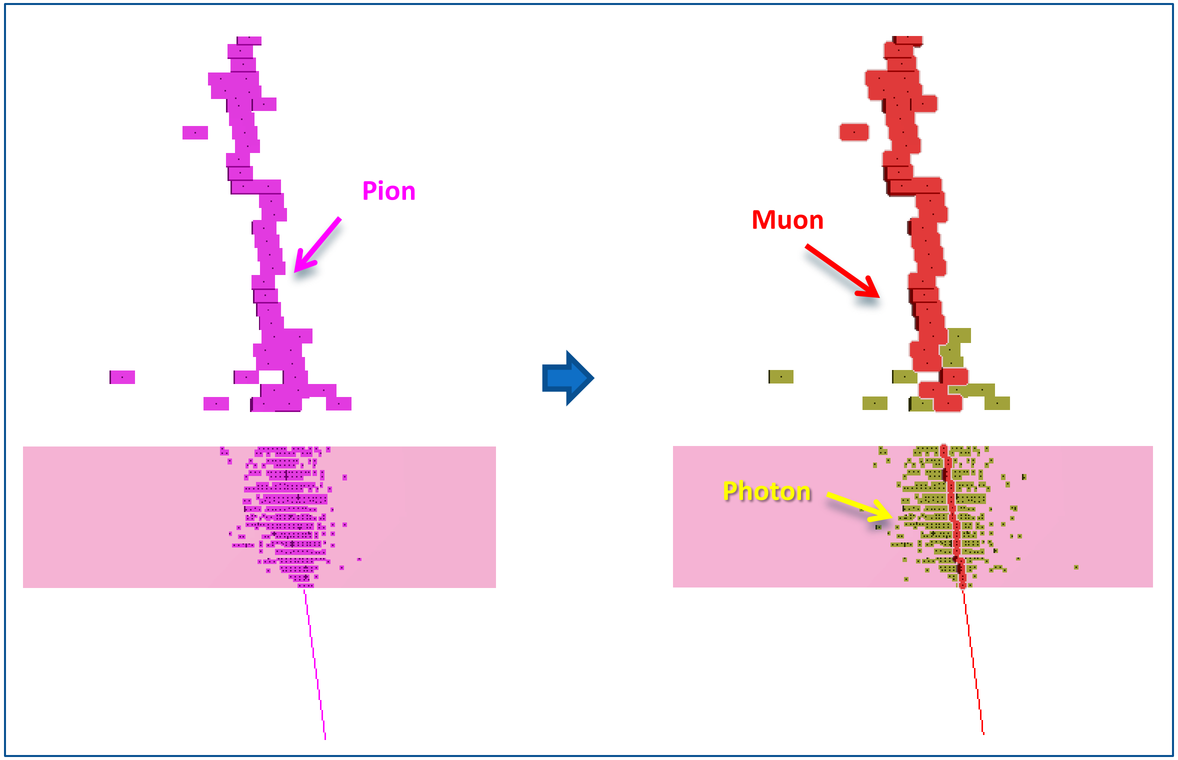




CLIC_ILD_CDR, E_z (= 2 * E_j)	91GeV	200GeV	500GeV	1TeV
Reclustering, No Forced Clustering, $rms_{90}(E_j) / E_j$	3.71 ± 0.05	3.02 ± 0.04	**3.07 ± 0.04**	**3.88 ± 0.07**
Reclustering, Forced Clustering, $rms_{90}(E_j) / E_j$	3.71 ± 0.05	3.02 ± 0.04	**2.96 ± 0.04**	**3.17 ± 0.06**

Cluster Fragmentation

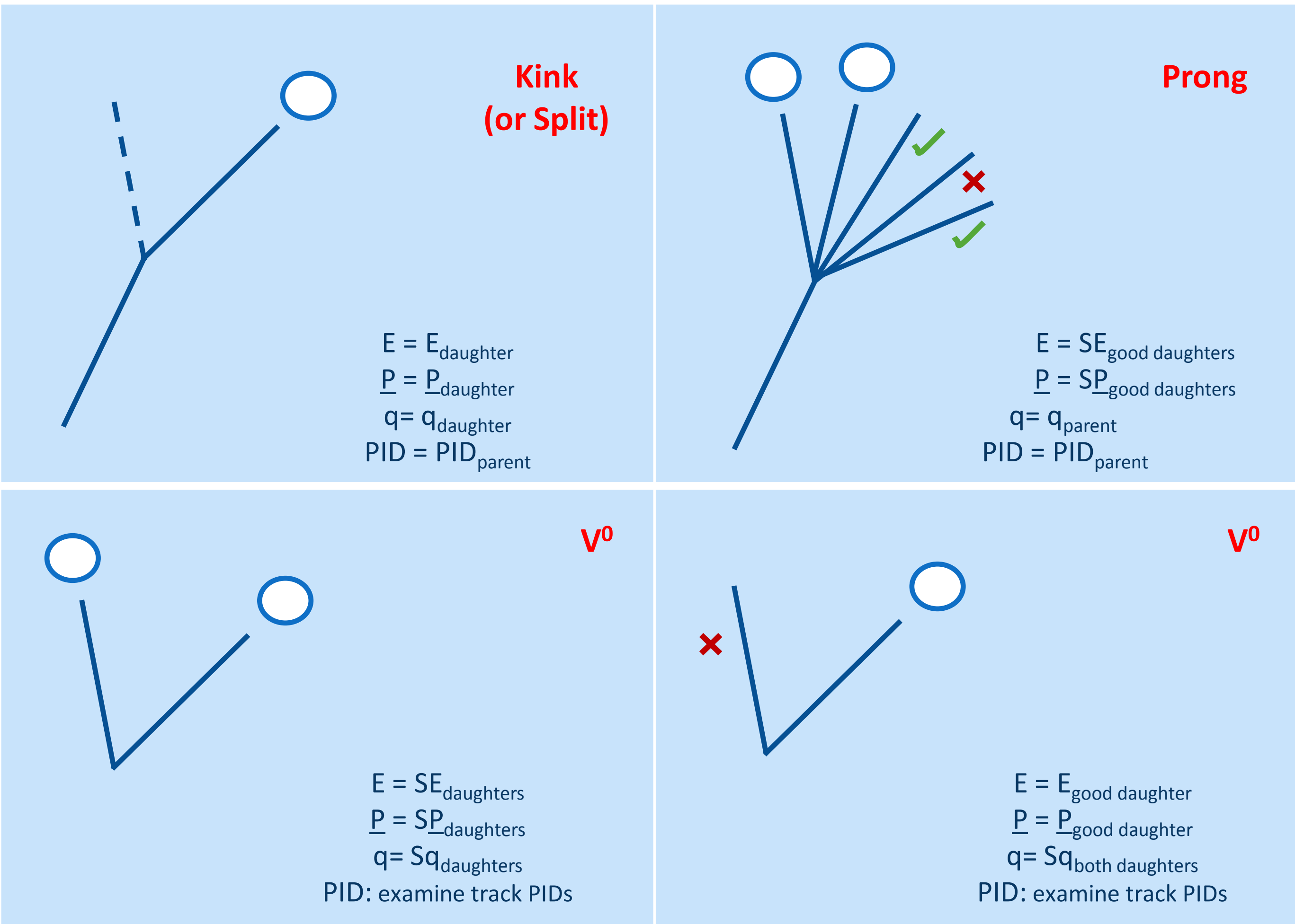
- Fragmentation is a special case of the reclustering mechanism. It allows multiple configurations of the hits in a given cluster to be examined at the same time, within a single algorithm.
- The mechanism is ideal for accurate identification of sub-clusters. Once the best hit configuration is identified, the EndFragmentation API will perform all memory-management.



Typical use: division of single cluster into mip-like cluster and photon-like cluster .

Important for particle id.

Track Information



Cluster associated with good track

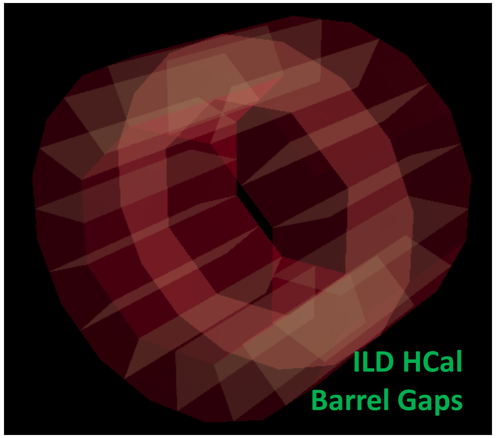
Pass/fail track pfo selection cuts

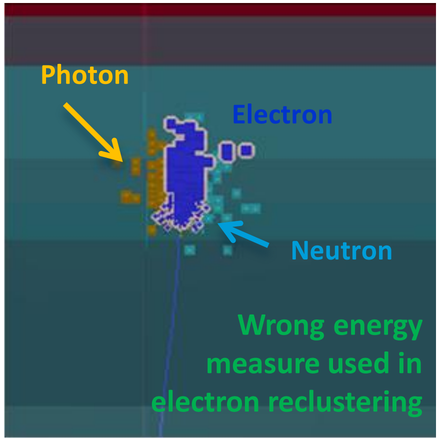
Tracks can have quality flags and also parent, daughter and sibling relationships.

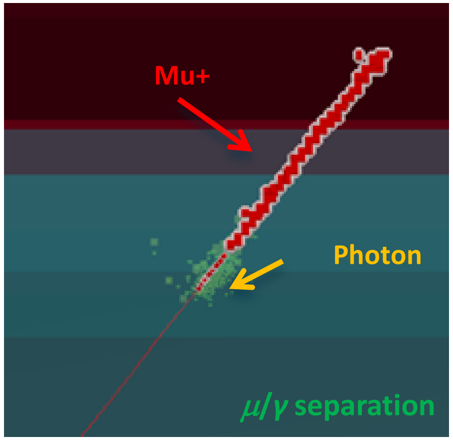
Information is used in track-cluster association and when building charged PFOs.

Particle Identification

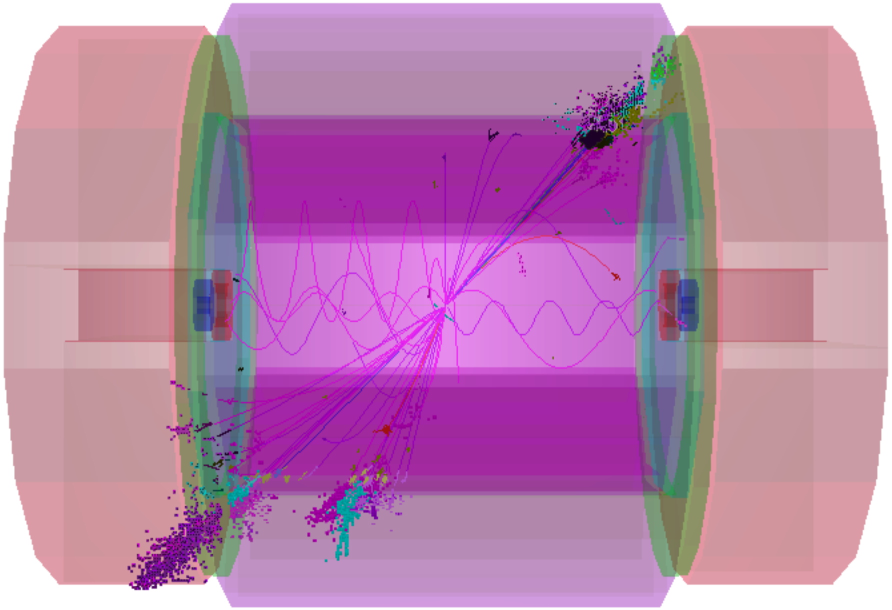
- For fine granularity detectors, Pandora can now identify photons, charged leptons or particles associated with displaced vertices (V^0) or decays in the tracking volume. See talk on Thursday.
- This functionality required more than simple characterisation of reconstructed PFOs. Needed to improve purity and completeness of Pandora single particle reconstruction, via:
  - **Separation of merged muons and photons.**
  - **Treatment of gaps in detector active material.**
  - **Treatment for reclustering high energy electrons.**
  - **Algorithms to mop-up neutral particle fragments.**
  - **Avoidance of incorrect removal of particle fragments.**

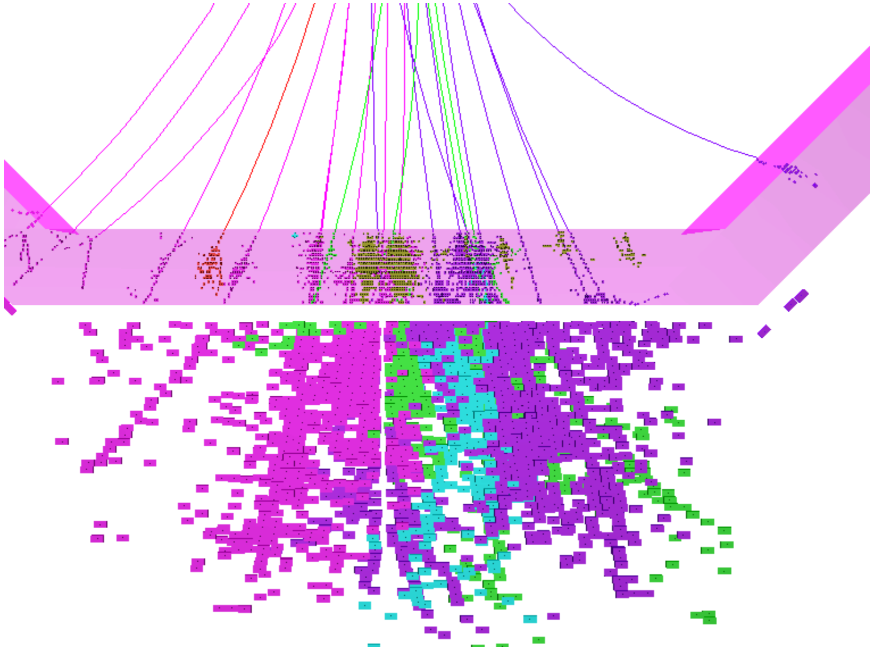






Pandora Monitoring





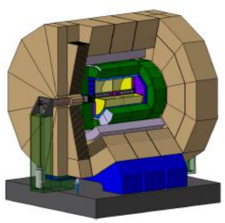
- Pandora now includes a monitoring library, which has a ROOT dependency to offer histogram, tree and event display capabilities (including TEve elements originally added by P. Speckmayer).
- Algorithms can use Monitoring APIs to draw hits, tracks, clusters or PFOs, during the reconstruction. Alternatively, user can add instances of VisualMonitoring algorithm to PandoraSettings.xml.

Working with Pandora

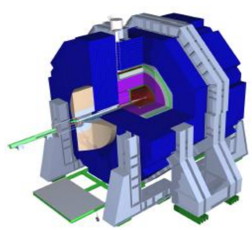
- A powerful feature is the ability to register content (i.e. algorithms, helper functions, etc.) from different libraries and combine their functionality in the reconstruction (configured via xml).
- Each client application registers the content that it needs to perform its specific reconstruction within the framework. Content can often be re-used, so can be bundled together in a library.

Very specific content, e.g. uses fine detector details

Client Applications



ILD/Marlin



SiD/org.lcsim

Re-usable content, applicable to multiple detectors

Content Libraries

- FineGranularity Content
- CoarseGranularity Content
- Monitoring Content

Pandora APIs

Framework

- Algorithm Manager
- CaloHit Manager
- Cluster Manager
- MC Manager
- PFO Manager
- Plugin Manager
- Track Manager

Runs all registered content and performs book-keeping

Working with Pandora

Content	Description
Algorithms	Responsible for performing reconstruction; make use of all information provided by objects, helper functions, calculators, etc. to make decisions and create PFOs.
PseudoLayer calculator	Responsible for dividing hits into layers that broadly follow structure of detector; helps to isolate algorithms from need to know specific geometry.
B-field calculator	Responsible for providing signed B-field value for given Cartesian coordinates; often a wrapper for a full field map in client software framework.
Shower-profile calculator	Responsible for examining longitudinal and transverse profile of cluster energy deposits and performing comparison with expectation for EM shower.
Particle id functions	Responsible for providing (“fast” or “full”) particle id information to algorithms, which may want to avoid certain particle-types, or simply apply results to PFOs.
Energy corrections	Responsible for applying corrections, improvements or custom calibrations to reported hadronic or electromagnetic cluster energy values.
Geometry	Optional detector description, which can be used by an algorithm if necessary.
Objects	Self-describing properties for tracks, hits and optional MC particles.

Working with Pandora

Configure PandoraSettings.xml to create your own custom reconstruction using...

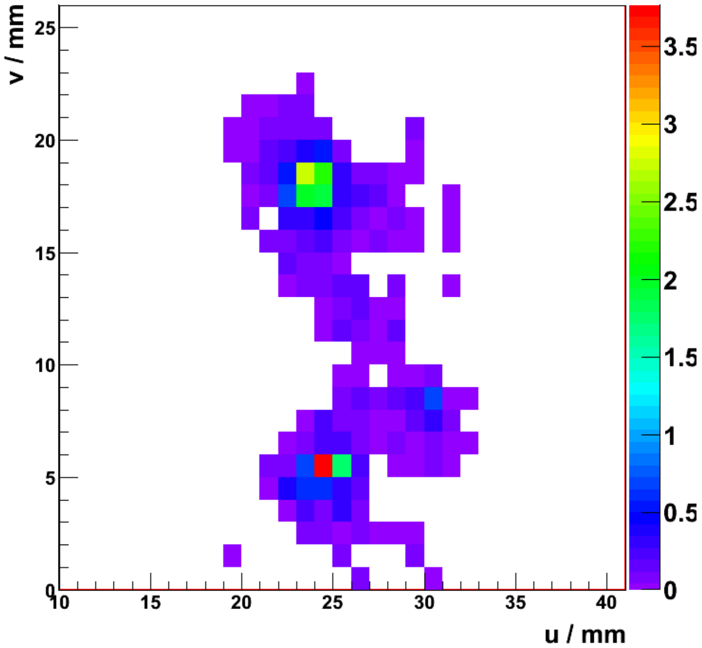
Algorithms for fine granularity detectors
4 clustering algorithms, forward and reverse cone-based, *k*-means and 2D Hough transform
5 algorithms for removing cluster fragments, including charged hadron fragments and photon fragments
2 standalone algorithms for reconstruction of muons and photons, **2** for performing particle id after reconstruction
4 algorithms for PFO creation and selection
7 reclustering algorithms, **15** topological association algorithms, **5** track-cluster association algorithms
6 algorithms for monitoring reconstruction between algorithms or providing event display functionality
3 list management algorithms, **2** algorithms for reading/writing Pandora binary files and **5** 'perfect' PFA algorithms

Helper functions & calculator classes for fine granularity detectors
Pseudo layer calculator
Longitudinal and transverse shower profile characterisation functions
EM shower, photon, electron and muon identification functions
Energy correction functions for addressing energy fluctuations in hadronic showers and energy loss in coil

Photon Reconstruction

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.

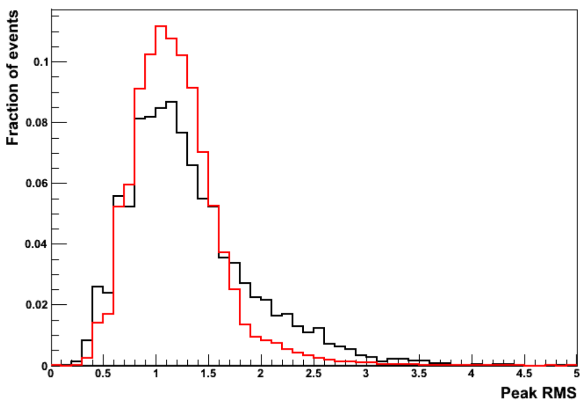


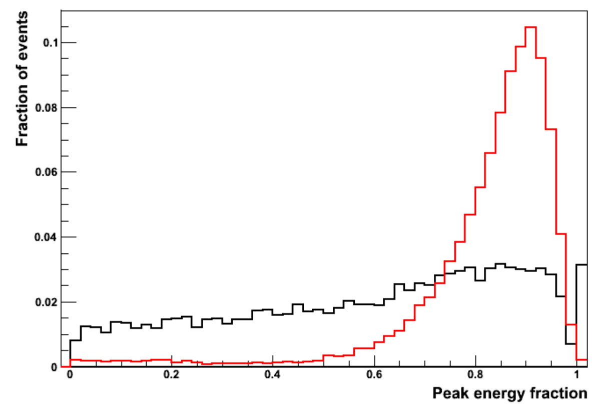
Not originally identified as a photon, but 17GeV from the 30GeV cluster actually from a true photon.

This is evident from the profile – split cluster using fragmentation mechanism.

Photon Reconstruction

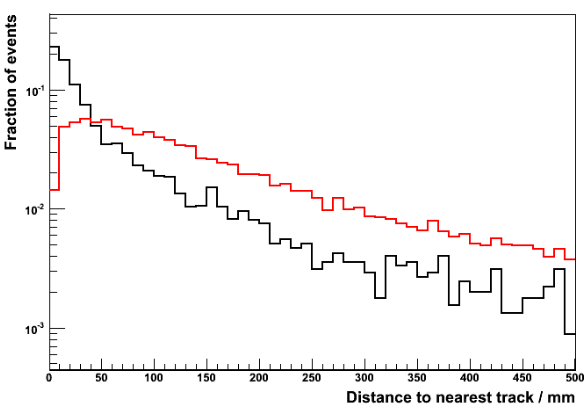
2. For each peak, a new photon cluster candidate is created and examined. Cuts are placed on the longitudinal shower profile of the new cluster and a multivariate/PID analysis is used to decide whether to accept the cluster as a photon.

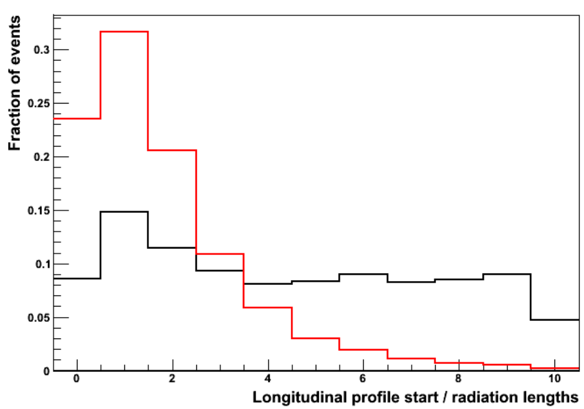


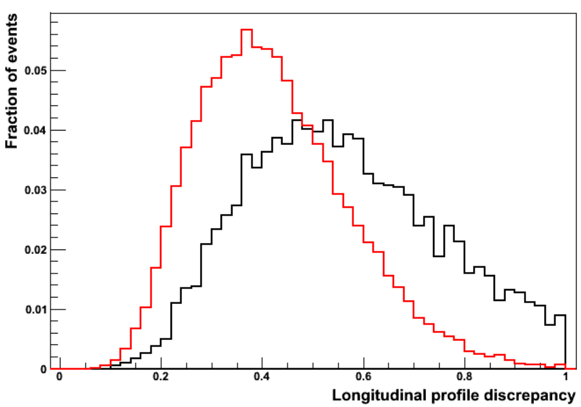


PDFs constructed using 500GeV $Z \rightarrow uds$ events.

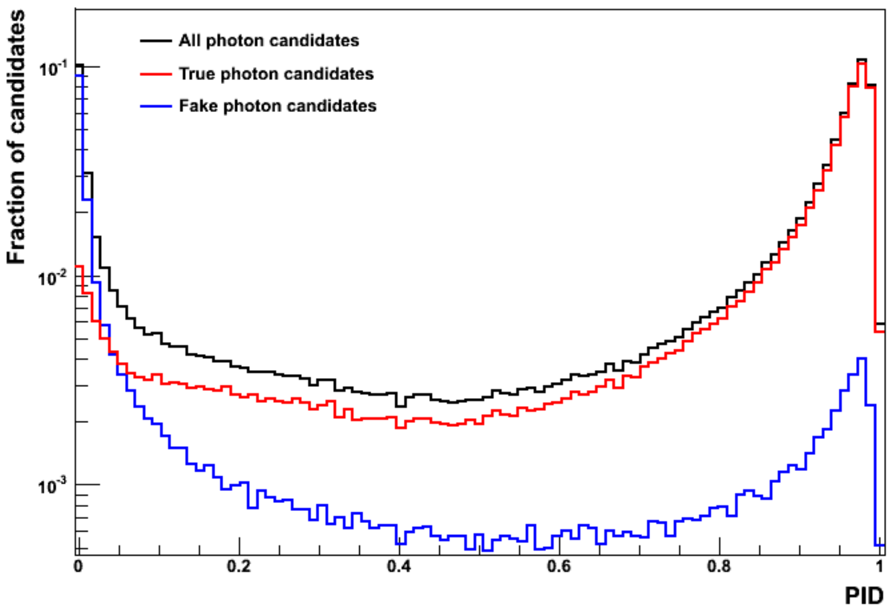
Separate PDFs for 9 reconstructed energy bins.

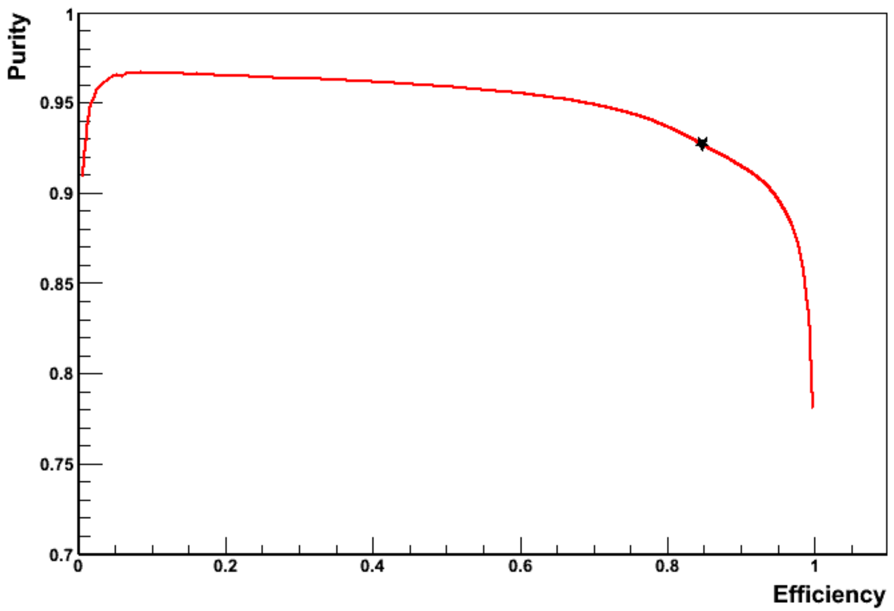






Photon Reconstruction

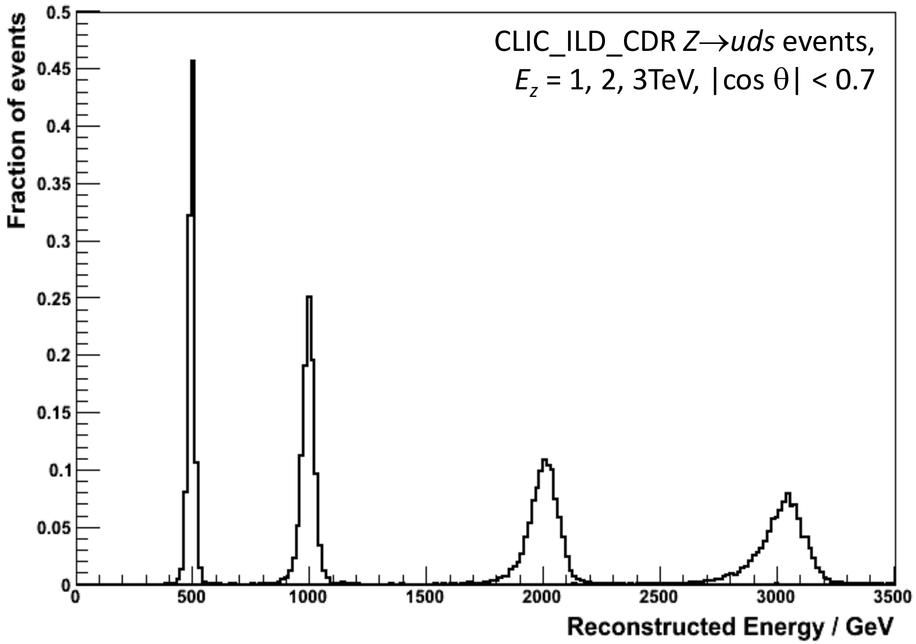


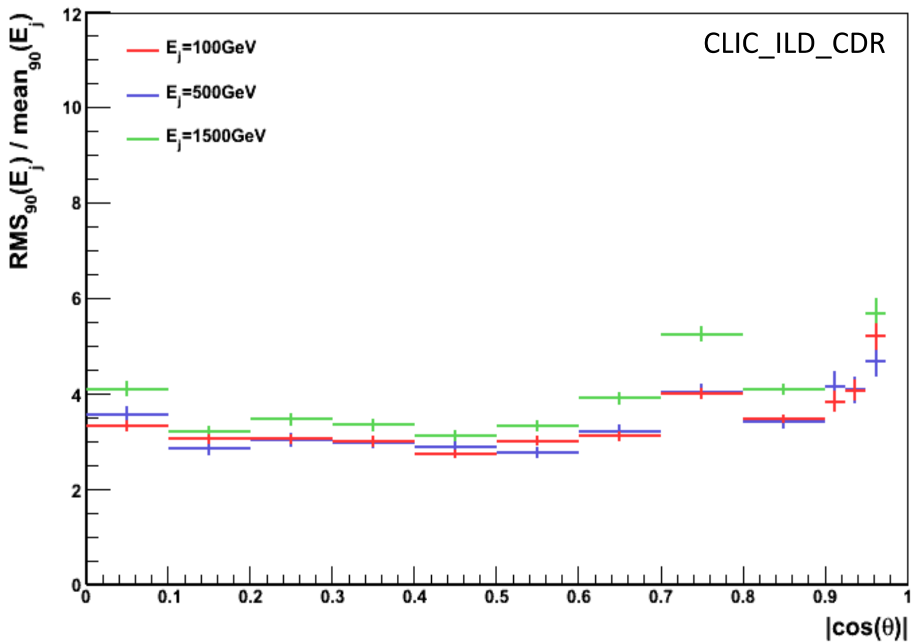


3. If a peak cluster is accepted, it is tagged as a photon and saved; the original cluster is deleted. If the peak represents the majority of the energy in original cluster, original may be used instead.

   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.

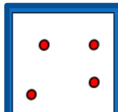
Jet Energy Performance





CLIC_ILD_CDR, E_z (= 2 * E_j)	91GeV	200GeV	500GeV	1TeV	2TeV	3TeV
No Photon Clustering, $rms_{90}(E_j)$ / E_j	3.71 ± 0.05	3.02 ± 0.04	2.96 ± 0.04	3.17 ± 0.06	3.48 ± 0.05	3.60 ± 0.05
Photon Clustering, $rms_{90}(E_j)$ / E_j	**3.72 ± 0.05**	**2.98 ± 0.04**	**2.86 ± 0.04**	**3.03 ± 0.05**	**3.46 ± 0.05**	**3.60 ± 0.05**

Manager Improvements



CaloHitManager
+hit-usage features

TrackManager
+track-relationship features

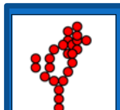
InputObjectManager

Objects made from parameters specified by client application.

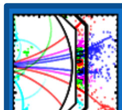
Copies of (addresses of) objects can be saved in multiple lists.

Manager

Basic list management and controls handshake between algorithms and managers.



ClusterManager
+extra reclustering features



PFOManager

New feature: algorithms can save/modify multiple PFO lists.

AlgorithmObjectManager

Algorithms make new objects when new temporary list is ready.

Objects only exist in single list, no copies.

Hit Fragmentation

Basic Hit Lists

Two Pandora hit lists, which happen to contain potential ‘parent’ hit **p**

See what happens if we split hit during most complex process; nested reclustering...

- When working with a coarse granularity detector, where sampling cells may contain energy deposits from many particles, a useful function is the ability to fragment a hit into multiple daughter hits.
- Have recently implemented this feature and the reverse process; ability to re-merge daughter hits.
- Potentially very complex; need to be able to carry out procedures in nested reclustering algorithms, plus allow daughter hits to be further fragmented and/or allow subset of fragments to be re-merged.

Hit Fragmentation

Basic Hit Lists

List1 hit p

List2 hit p

Recluster → List 2

Recluster 1 Hit Lists

R1List1 p

Add:
Replace:

R1List2 ~~p~~ d1 d2

A: d1 d2
R: p

R1List3 ~~p~~ d3 d4

A: d3 d4
R: p

Hit Fragmentation

Basic Hit Lists

List1 hit p

List2 hit p

Recluster List 2 →

Recluster 1 Hit Lists

R1List1 p — Add: / Replace:

R1List2 ~~p~~ d1 d2 — A: d1 d2 / R: p

R1List3 ~~p~~ d3 d4 — A: d3 d4 / R: p

Recluster R1 List 3 →

Recluster 2 Hit Lists

R2List1 d3 d4 — A: / R:

R2List2 ~~d3~~ d3a d3b d4 — A: d3a d3b / R: d3

R2List3 ~~d3~~ d3c d3d d4 — A: d3c d3d / R: d3

Hit Fragmentation

Basic Hit Lists

List1 hit p

List2 hit p

Recluster → List 2

Recluster 1 Hit Lists

R1List1 p

Add:
Replace:

R1List2 ~~p~~ d1 d2

A: d1 d2
R: p

R1List3 ~~p~~ ~~d3~~ d3c d3d d4

A: d3 d4 | d3c d3d
R: p | d3

Recluster 2 Hit Lists

R2List1 d3 d4

A:
R:

R2List2 ~~d3~~ d3a d3b d4

A: d3a d3b
R: d3

R2List3 ~~d3~~ d3c d3d d4

A: d3c d3d
R: d3

Select R2List3 and apply changes to R1List3

Hit Fragmentation

Basic Hit Lists

List1 d3c d3d d4

List2 d3c d3d d4

Recluster List 2

Select R1List3 and apply changes to all basic lists

Recluster 1 Hit Lists

R1List1 p
Add:
Replace:

R1List2 ~~p~~ d1 d2
A: d1 d2
R: p

R1List3 ~~p~~ ~~d3~~ d3c d3d d4
A: d3 d4 | d3c d3d
R: p | d3

Select R2List3 and apply changes to R1List3

Recluster 2 Hit Lists

R2List1 d3 d4
A:
R:

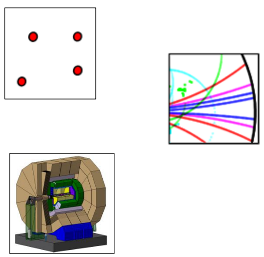
R2List2 ~~d3~~ d3a d3b d4
A: d3a d3b
R: d3

R2List3 ~~d3~~ d3c d3d d4
A: d3c d3d
R: d3

Complex, but robust and efficiently implemented.

Lightweight Persistency

- To aid development in complex software frameworks, have developed lightweight Pandora persistency model, allowing input object/geometry parameters to be written/read to/from binary.
- Once file available, removes need for client application; only need tiny piece of code to launch Pandora. Private development tool, but worthy of note as has proven to be exceptionally useful.



Client application, e.g. MarlinPandora

SERIALIZE

Via Pandora EventWriting algorithm

0007 0000 6170 646e 726f 0061
15e8 0000 0000 0000 0000 0000
0000 0000 0000 0b00 8746 c444
0a19 8f44 0d87 023f 48b7 053f
3504 f83f 3504 003f 0000 0000
1997 663f 0666 8040 dae8 f13d
8eaa f13f e443 053b eb1d 003b
0000 0000 0000 0000 9000 d33a
ab00 a2aa ab40 a2aa 0040 0000
d900 6c0b e244 d117 ab44 6bc2
a33e 481e 703f e1a4 f03e 3504
003f 0000 0000 a800 a240 1997
9940 e7ad f340 e26c 183c 8d84
e53c e937 003c 0001 0000 0000

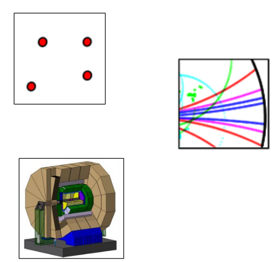
Binary .pndr file records all self-describing parameters.

0007 0000 6170 646e 726f 0061
15e8 0000 0000 0000 0000 0000
0000 0000 0000 0b00 8746 c444
0a19 8f44 0d87 023f 48b7 053f
3504 f83f 3504 003f 0000 0000
1997 663f 0666 8040 dae8 f13d
8eaa f13f e443 053b eb1d 003b
0000 0000 0000 0000 9000 d33a
ab00 a2aa ab40 a2aa 0040 0000
d900 6c0b e244 d117 ab44 6bc2
a33e 481e 703f e1a4 f03e 3504
003f 0000 0000 a800 a240 1997
9940 e7ad f340 e26c 183c 8d84
e53c e937 003c 0001 0000 0000

Same ~40 line application for any .pndr file.

DESERIALIZE

Via Pandora EventReading algorithm



Objects/geometry ready to be used by algorithms.

Summary

- Since LCWS10, many important changes have been made to Pandora...

- Reclustering and fragmentation mechanisms have been added to the framework and these mechanisms have been exploited to great effect by a number of algorithms.
- Particle identification functionality has been added and improvements have been made to the purity and completeness of the reconstructed particles.

- The first releases of the new Pandora have been made publicly available, alongside releases of the MarlinPandora and SlicPandora client applications. The releases have been used for the CLIC CDR.
- The Pandora framework provides a rich environment for the development of pattern recognition and particle flow algorithms. There are many ways in which it can be customized.

- Since the CLIC CDR release, a new photon reconstruction has been implemented and improvements have been made to aid development for coarse granularity environments.
- Development continues.