中國科學院高能物理研究所
Institute of High Energy Physics
Chinese Academy of Sciences

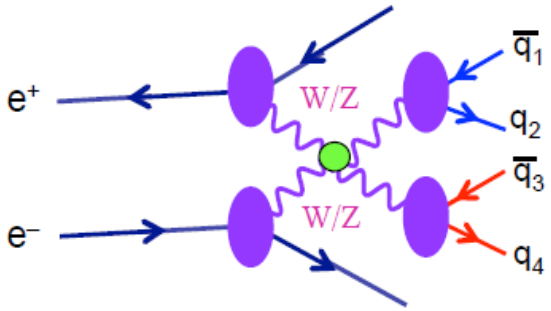
Pandora immigration

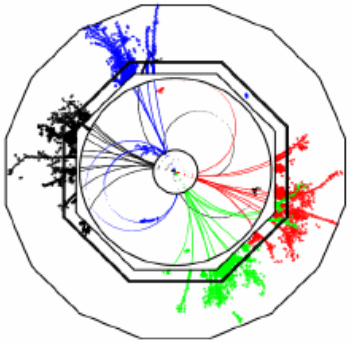
Wenxing Fang(IHEP)

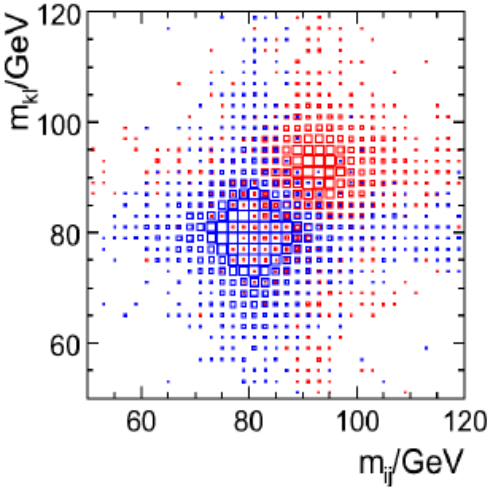
Group meeting (30th March 2020)

LC Calorimetry Goals

- Jet energy resolution requirements depend on physics...
- Likely to be primarily interested in di-jet mass resolution.
- Strong desire to separate W/Z hadronic decays.

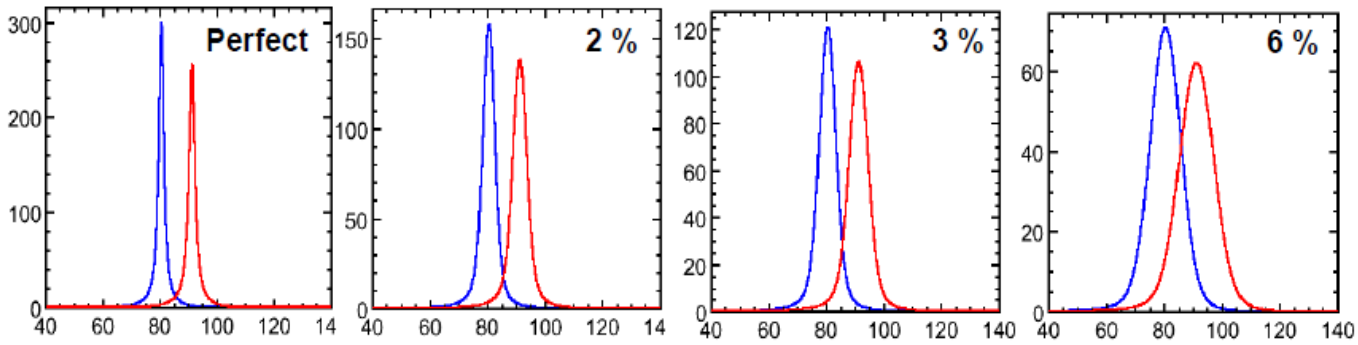






- 3-4% jet energy resolution gives decent 2.6-2.3σ W/Z separation.
- Sets a reasonable choice for LC jet energy minimal goal ~3.5%.
- For W/Z separation, not much further gain; limited by natural widths.

W/Z sep:
$(m_Z - m_W) / \sigma_m$

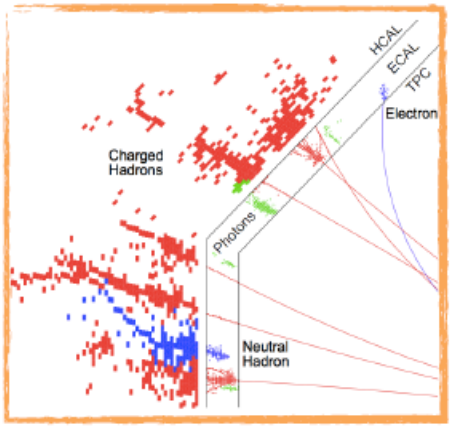


Jet E res.	W/Z sep
Perfect	3.1 σ
2%	2.9 σ
3%	2.6 σ
4%	2.3 σ
5%	2.0 σ
10%	1.1 σ

Fine Granularity Particle Flow

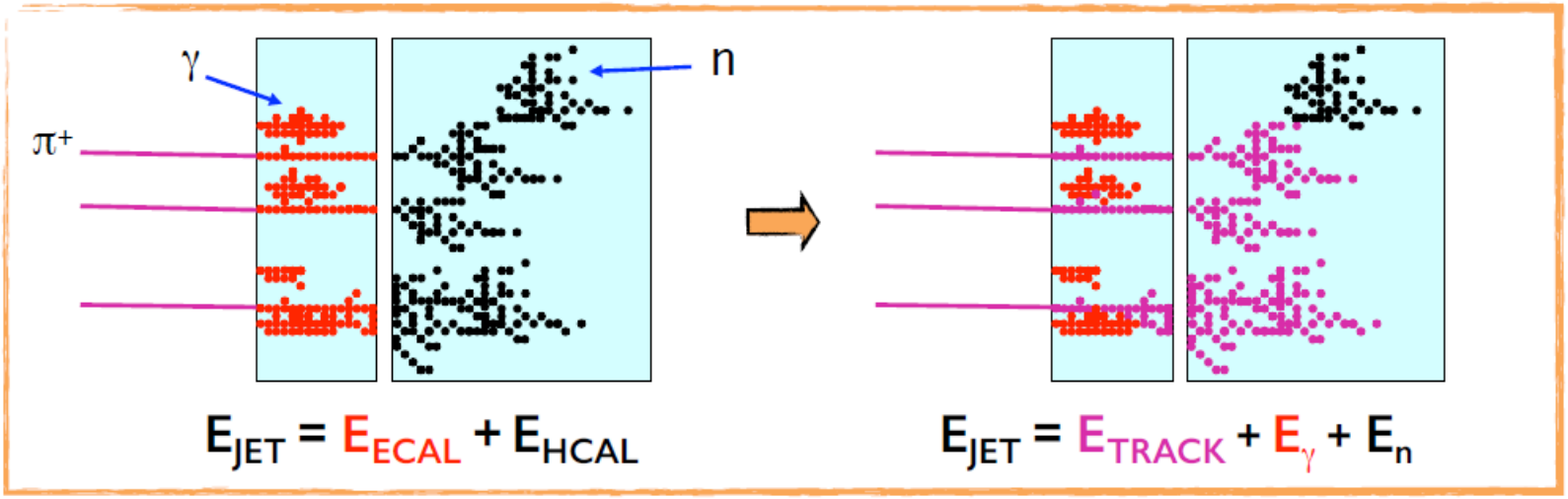
In a typical jet:

- 60 % of jet energy in charged hadrons
- 30 % in photons (mainly from $\pi^0 \rightarrow \gamma\gamma$))
- 10 % in neutral hadrons (mainly n and K_L)



Traditional calorimetric approach:

- Measure all components of jet energy in ECAL/HCAL
- Approximately 70% of energy measured in HCAL: $\sigma_E/E \approx 60\% / \sqrt{E(GeV)}$



Fine granularity Particle Flow Calorimetry: reconstruct individual particles.

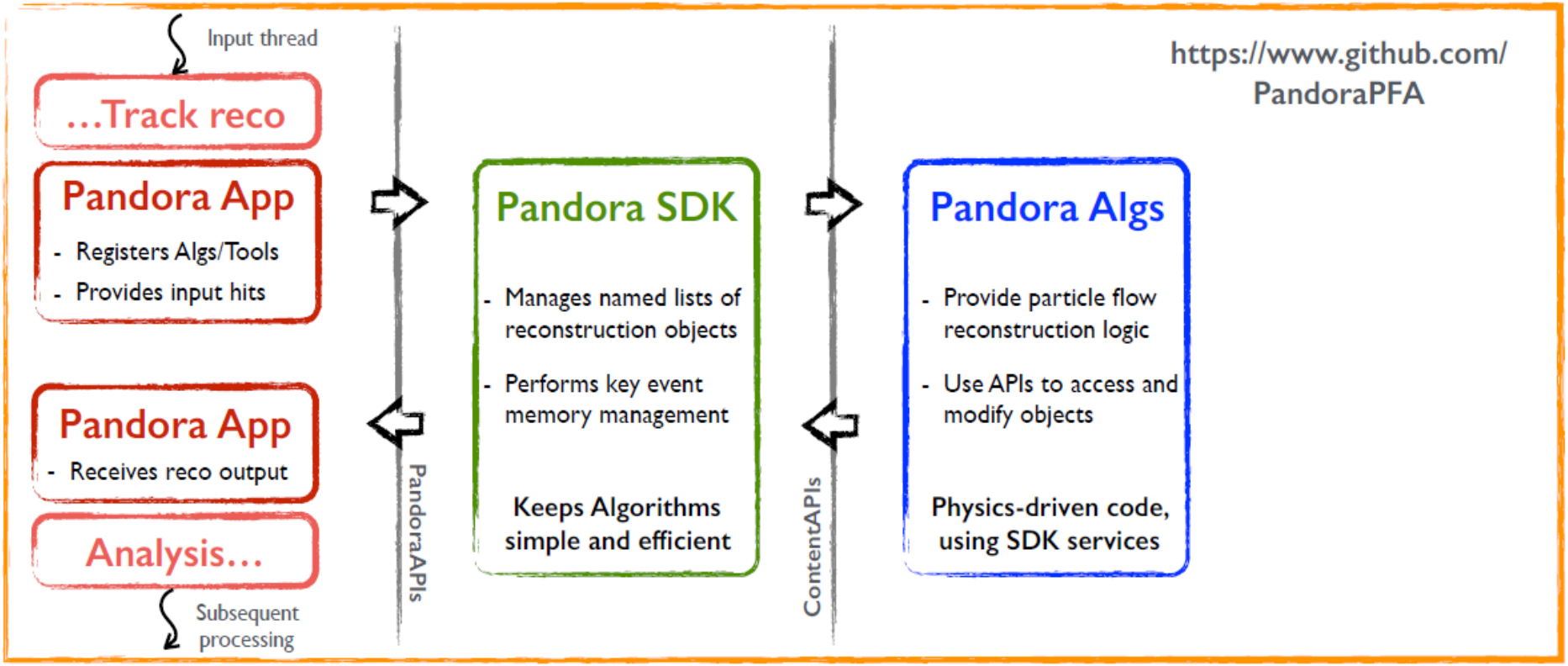
- Charged particle momentum measured in tracker (essentially perfectly)
- Photon energies measured in ECAL: $\sigma_E/E < 20\% / \sqrt{E(GeV)}$
- Only neutral hadron energies (10% of jet energy) measured in HCAL: much improved resolution.

Introducing the Pandora SDK

The **Pandora Software Development Kit** is engineered to provide an environment in which:

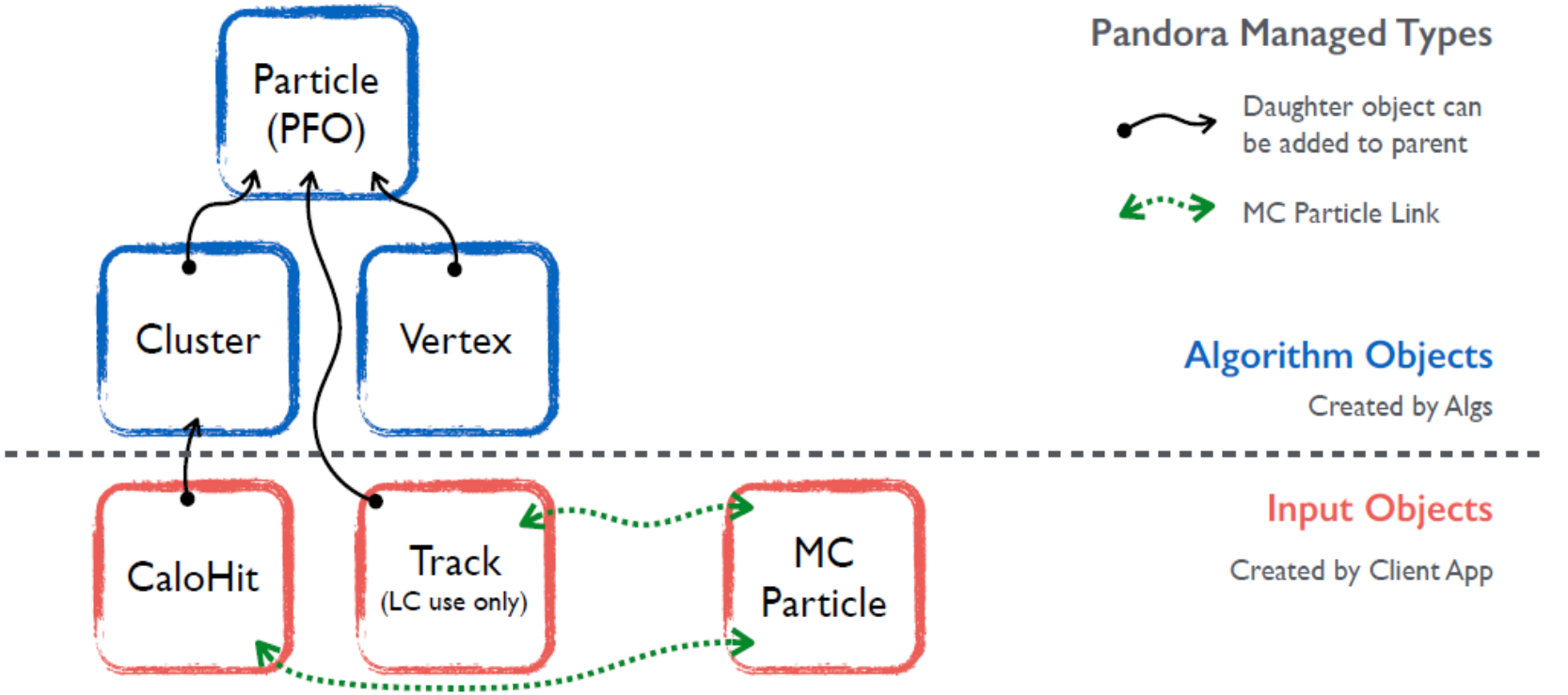
1. It is easy for users to provide the building-blocks that define a pattern recognition problem.
2. Logic required to solve pattern recognition problems is cleanly implemented in algorithms.
3. Operations to access or modify building-blocks, or to create new structures, are requested by algorithms and performed by the Pandora framework.

It actively promotes use of large numbers of algorithms, each addressing specific event topologies.



Event Data Model

- **EDM consists of classes to represent the input building-blocks for pattern-recognition problems and the structures that can be created using these building-blocks.**
- Provides well-defined development environment for managing pattern-recognition problems and allows for independence of algorithms, which can only communicate via the EDM.
- EDM aims to be self-describing, with each object providing all the information required to allow investigation and processing by the pattern-recognition algorithms.



Input Objects

- **Input Objects are the building-blocks for pattern recognition, typically created by the client app before algorithm operations begin.**
- Their properties are defined at creation and cannot be changed. They are instead used to build new constructs, termed "Algorithm Objects".
- The usage of all Input Objects is monitored to ensure that no double-counting/usage occurs.

CaloHit	Primary building-block, defining a position and extent in space (or time), with an associated intensity or energy measurement and detector location details.
Track (LC use only)	Represents a continuous trajectory of well-defined space-points, with helix parameterisation. Track parent-daughter and sibling relationships supported.
MC Particle	For development purposes, provide details of true pattern-recognition solution. Support parent-daughter links and can be associated to CaloHits and Tracks.

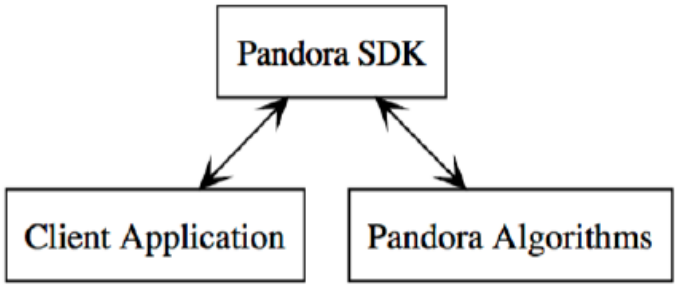
Algorithm Objects

- **Algorithm Objects represent the higher-level structures created in order to solve pattern-recognition problems.**
- Pandora carefully manages the allocation and manipulation of these objects and all non-const operations can only be requested by algorithms via the Pandora Content APIs.
- Pandora is then able to perform the memory-management for these objects.

Cluster	Collection of CaloHits and main working-horse for algorithms (which create, merge, split Clusters). Provides some derived properties of CaloHit collection.
Vertex	The identification and classification of a specific point in space, typically used to flag positions of particle creation or decay.
Particle	Container of Clusters, Tracks and Vertices, together with metadata describing e.g. particle type. Ultimate Pandora output and can represent a hierarchy.

Client Application

- **As previously discussed, client app is responsible for providing Input Objects that define the pattern-recognition problem and for persisting the output Particles.**
- Also responsible for creating Pandora instances, bringing-together (collections of) algorithm implementations and for configuring the reconstruction via the Pandora Settings XML file.
- Algorithms depend on Pandora SDK, but can also have as many external dependencies as required. The client app depends on Pandora and on all libraries providing algorithms.
- The actual algorithm instances used in the reconstruction are not created unless specified in the Pandora Settings; created when the XML file is parsed by Pandora.



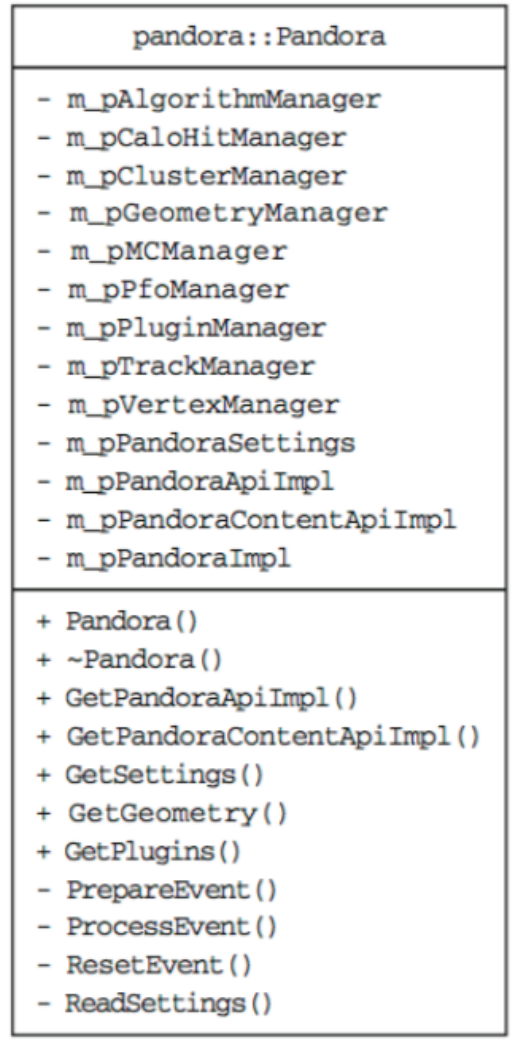
Built with client app, or in libraries of Pandora content

Algorithm Pseudocode description of a client application for LAr TPC event reconstruction in a single drift volume

```
1:  procedure MAIN
2:      Create a Pandora instance
3:      Register Algorithms and Plugins
4:      Ask Pandora to parse XML settings file
5:      for all Events do
6:          Create CaloHit instances
7:          Create MCParticle instances
8:          Specify MCParticle-CaloHit relationships
9:          Ask Pandora to process the event
10:         Get output PFOs and write to file
11:         Reset Pandora before next event
```

Managers

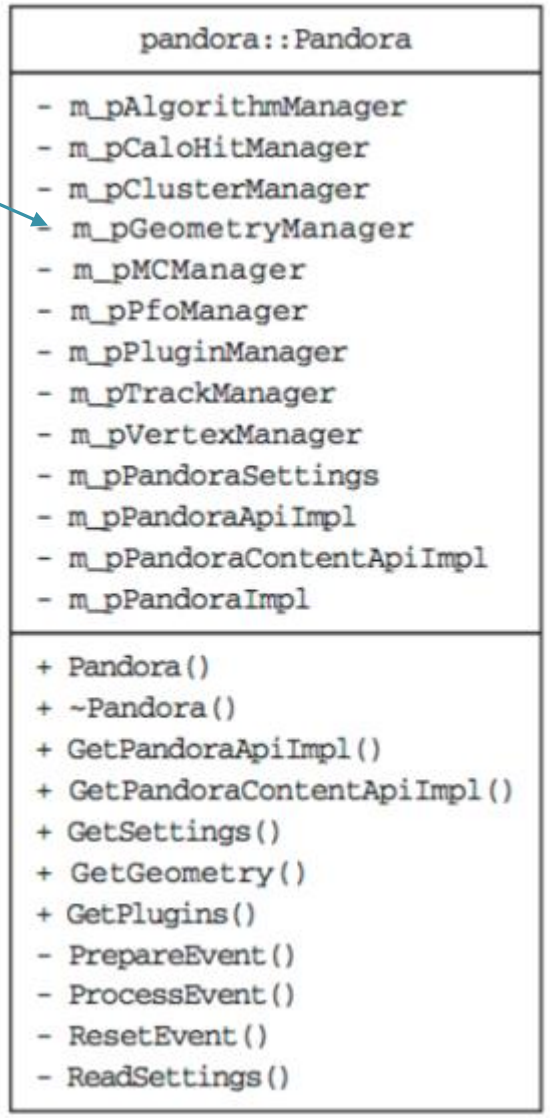
- **At very heart of Pandora design are the Managers, which own all instances of objects in Pandora EDM.**
- The Managers are designed to provide a complete set of low-level object manipulation functions.
- Algs request high-level services (e.g. merge two Clusters), which are then satisfied when the hidden implementation calls the low-level Manager functions in the correct order.
- Approach helps ensure that implementation is extensible, easy to maintain and rather human-readable.
- Key part of design is that algorithms can *only* access or modify managed objects via the APIs, so Managers are able to perform memory-management.



A Pandora instance is simply a container of Manager instances and API implementation instances

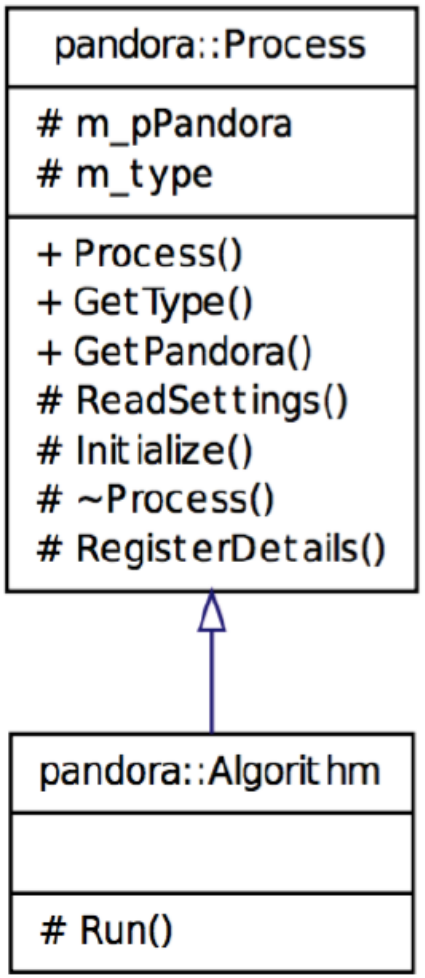
Geometry information in Pandora

- The geometry information is saved in pandora's geometry manager in client application once.
- It includes the sub-detector type (e.g. ECAL_BARREL, ECAL_ENDCAP), R, Z, layer information and so on.
- The algorithms will use PandoraContentApi to get the geometry manager of pandora and get the needed geometry information.



Algorithms

- **Algs contain step-by-step instructions, using Pandora APIs to request object creation/modification services.**
- Algs inherit from the Pandora Process abstract base class. Inherited functionality controls handshaking between Pandora instance and algorithm instance.
- Process provides ability to receive a ReadSettings callback with an XML handle (tinyxml) from which configurable parameters can be extracted. Also an Initialize callback.
- The Algorithm purely abstract base class provides the interface for the Run callback, which is called each event and is the entry point for all event processing.



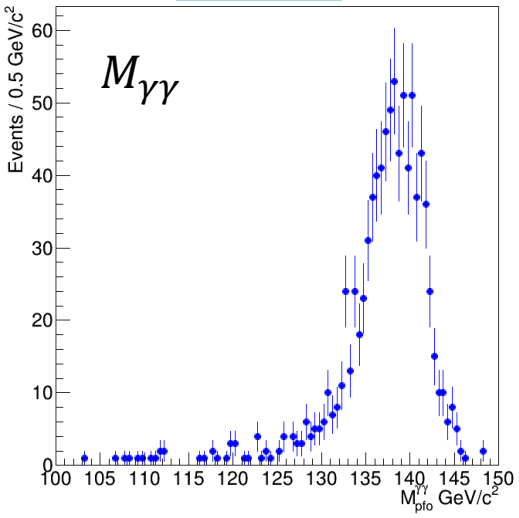
- **Algorithm Factories registered (under a specific name), by the client app are extremely simple:**
- Must allocate instance of derived algorithm type and return pointer to Algorithm base class.

Immigration MarlinPandora

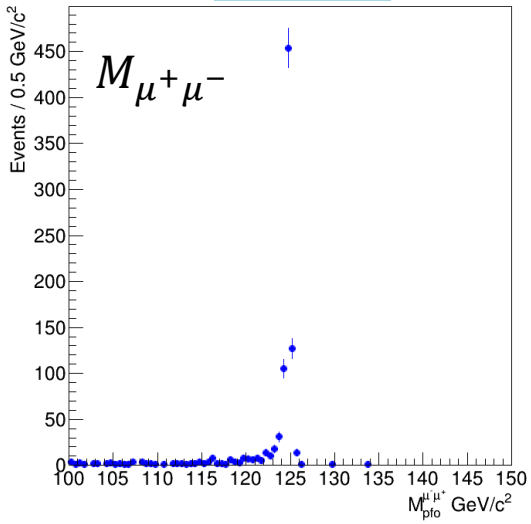
- Getting the Pandora and MarlinPandora from github https://github.com/PandoraPFA
- Immigrating MarlinPandora to GaudiPandora
- The LCIO data is converted to plcio data as input
- Current status:
  - Finished:
    - MC particle creation
    - CaloHit creation
    - Track creation
    - Geometry creation (using Gear to get the needed input information)
    - Pof creation (from pandora ParticleFlowObject to plcio object)
  - Not Yet:
    - Need relation between Track and MC:
      SetTrackToMCParticleRelationship
    - Need relation between Calorimeter and SimluationCalorimeter and MC:
      CaloHitToMCParticleRelationship
    - Need information of reconstructed vertex and the associated particle:
      SetTrackParentDaughterRelationship
      SetTrackSiblingRelationship
      pandora Track PID and mass are set to be the default one Pion

Test the performance of GaudiPandora

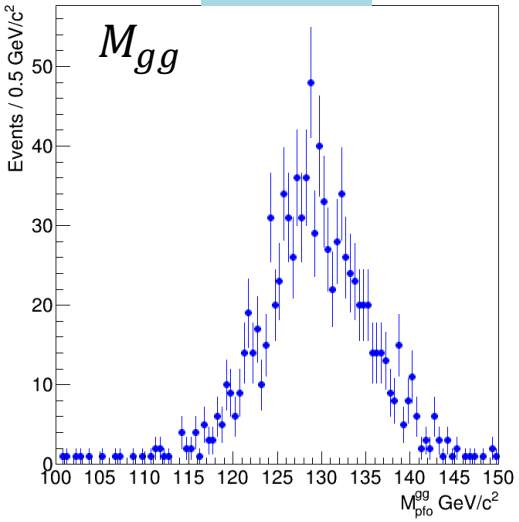
nnHaa



nnH$\mu\mu$



nnHgg



nnHbb