Five years of Key4hep
Towards production readiness and beyond

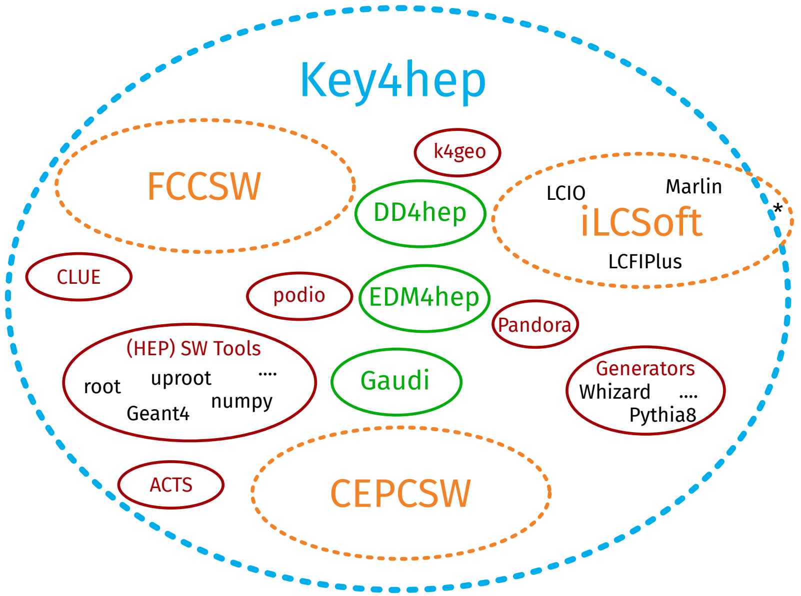
ICHEP 2024

Swathi Sasikumar on behalf of the Key4hep team

20 July 2024

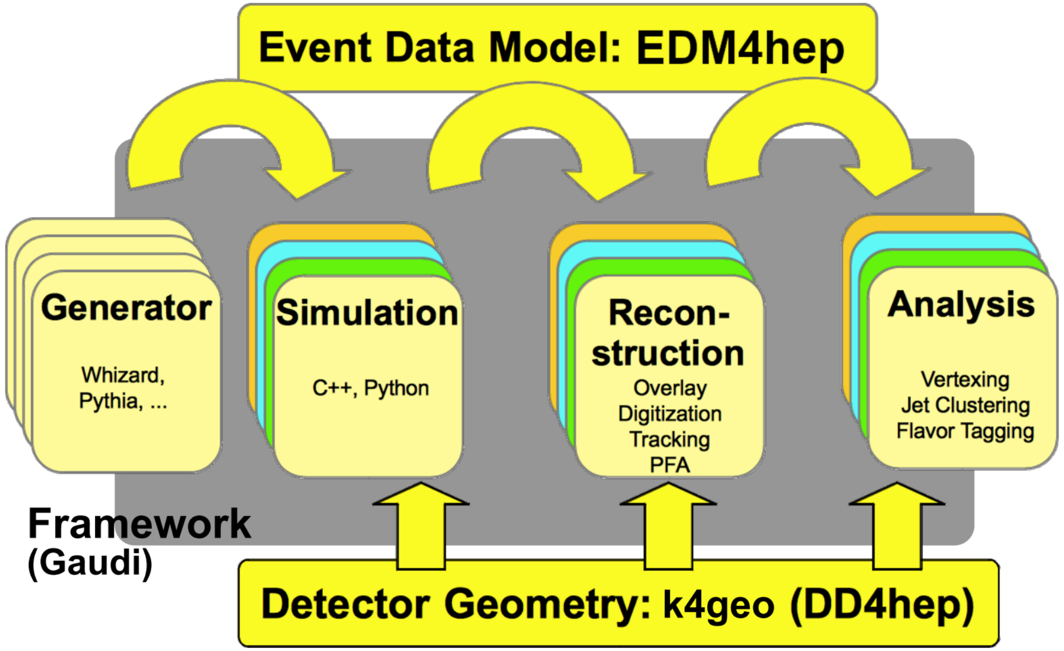
Introduction to Key4hep

- For two decades, generation, simulation, reconstruction and analysis tools developed in iLCSoft for the International Linear Collider
- Propositions for future lepton colliders with uncertainties of which to be next
- Key4hep a common turnkey software for future colliders
- Share components to reduce maintenance and development cost and allow everyone to benefit from its improvements
- Complete data processing framework from generation to data analysis
- Community with people from different future experiments: FCC, ILC, CLIC, CEPC, EIC, Muon Collider



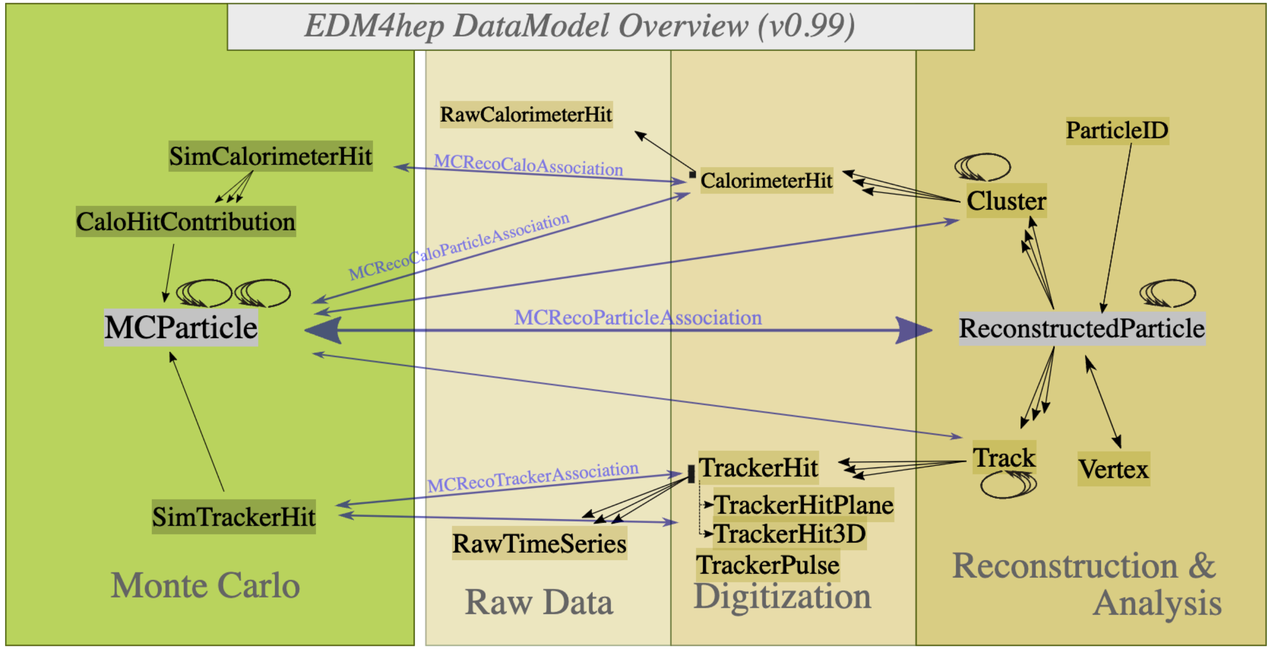
Key4hep an ensemble of state-of-the-art tools

- EDM4hep: The event data model used in Key4hep
- Podio: An event data model toolkit to build EDM4hep
- Gaudi: event processing framework
  - Gaudi allows to use algorithms in multithreaded environment
  - Used by LHCb, ATLAS, Key4hep …
- DD4hep: Provides a complete detector description
  - Geometry, materials, visualisation, alignments, readout, callibration
  - Simulation through Geant4



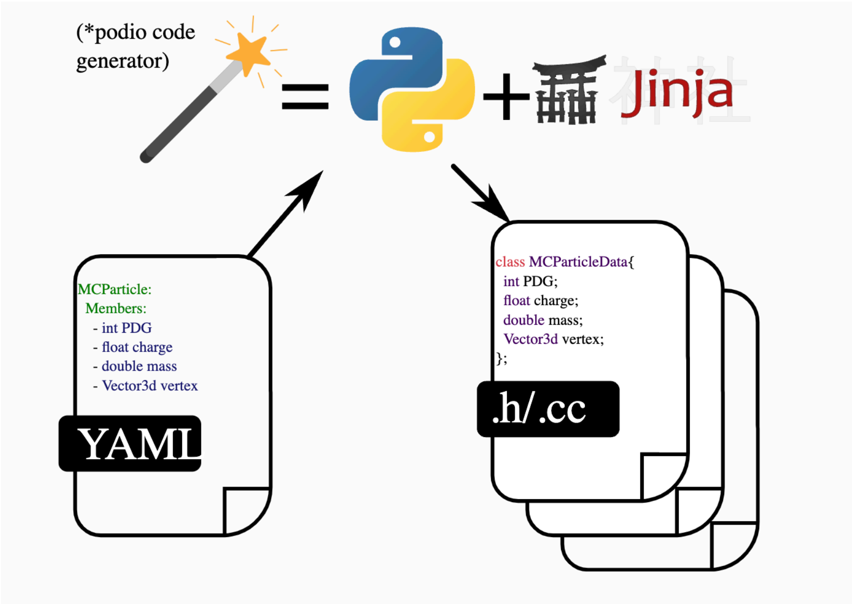
The Key4hep Event Data Model: EDM4hep

- Data model used in Key4hep, common language that all components must speak
- Goals: be generic and address the needs of the experiments
- Evolves through consensus among all stakeholders
- Diagrammatic overview of EDM4hep with all available datatypes:
  - Data type: MCParticle, ReconstructedParticle, TrackerHit
  - Relations/Associations: within a data type or between different data types



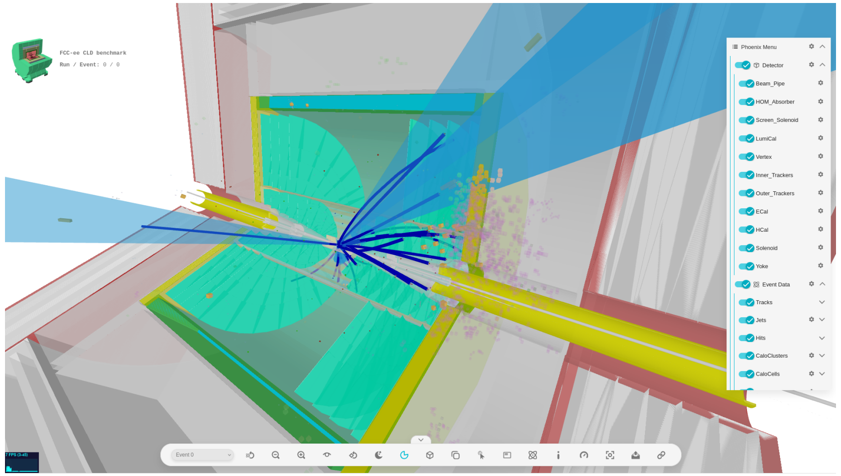
The Podio EDM toolkit

- Podio (plain old data IO) is a toolkit for the creation of EDMs like EDM4hep
- Main purpose is to have an efficiently implemented, thread safe EDM
- The podio code generator reads in the EDM definition in YAML format and generates all the necessary code via the Jinja2 template engine
- Podio provides backends for reading/writing data from EDM4hep:
  - New format (RNTuple) to be used instead of TTree developed by ROOT
  - Less space usage than TTree and better IO throughput depending on the task

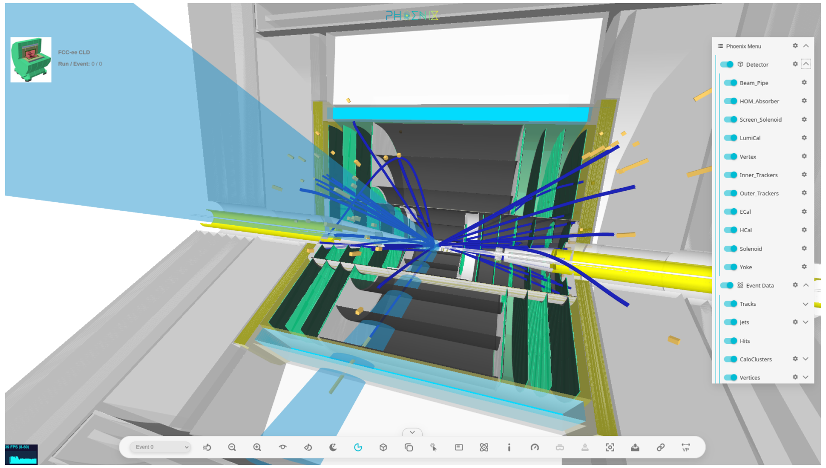


Phoenix: a web-based visualisation tool

- Visualize events and detectors
- Phoenix: detector independent event display using JSROOT
- JSROOT offers possibility to work with ROOT files on the web
- Separate event data and detector description



- Event:
  - Described in EDM4hep event data model
  - Convert ROOT files into JSON files
  - EDM4hep data structure is kept



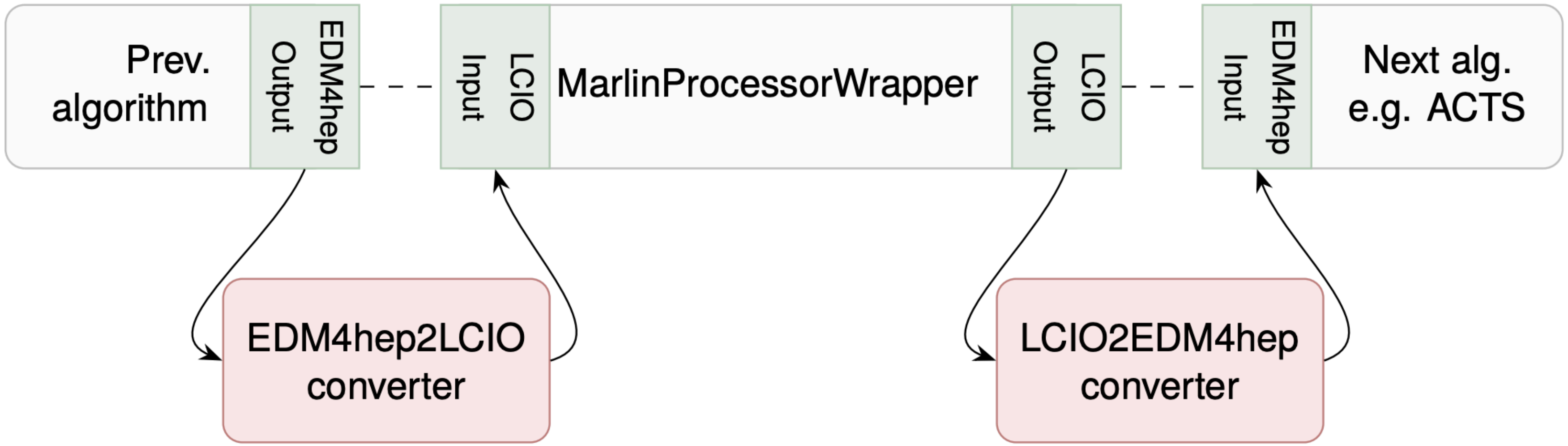
- Detector:
  - Detector is described in DD4hep compact files
  - Convert XML into ROOT file for JSROOT

The Key4hep Framework

- Gaudi based core framework:
  - k4Gen for integration with generators
  - k4FWCore provides interface between EDM4hep and Gaudi
  - k4MarlinWrapper to call any Marlin (linear collider) processor
  - k4SimDelphes for integration with Delphes for fast simulation
  - k4GaudiPandora to add particle flow tools to key4hep
  - k4ACTSTracking to include a common tracking software to key4hep
  - ....

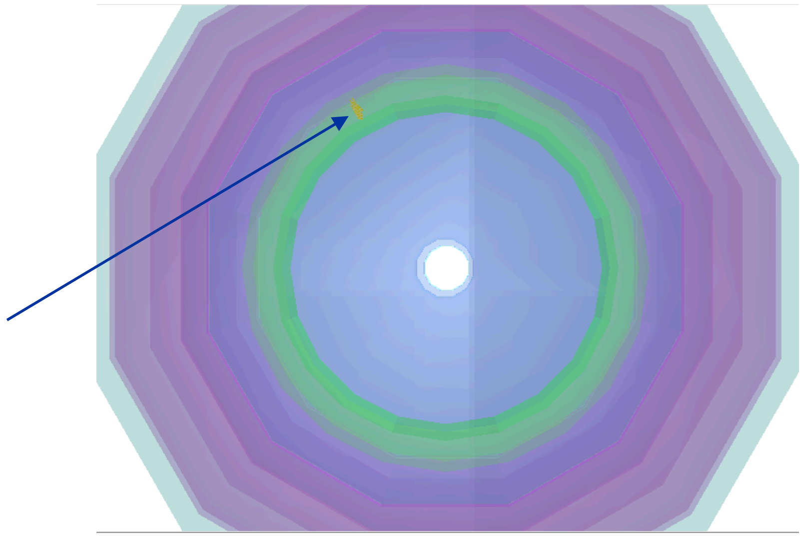
Linear Collider processor converters

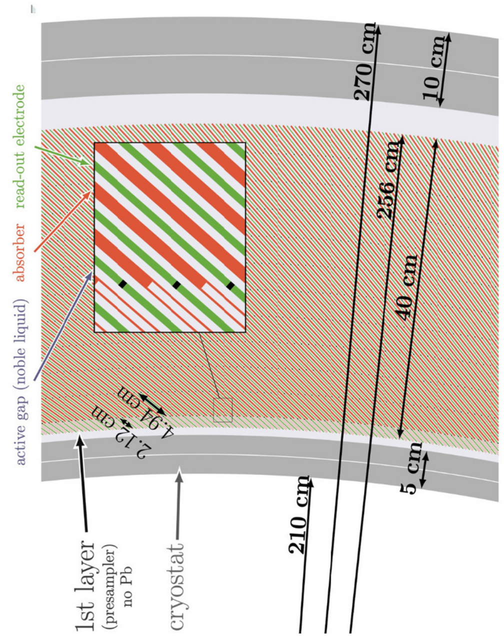
- Algorithms in Key4hep use Gaudi framework
- Marlin processors (algorithms in iLCsoft) can be used in Gaudi using the `MarlinProcessorWrapper`
- EDM4hep input can be used seamlessly in processors taking LCIO input and giving LCIO output
- Standalone converter lcio2edm4hep to convert files



Pandora Particle flow in Key4hep

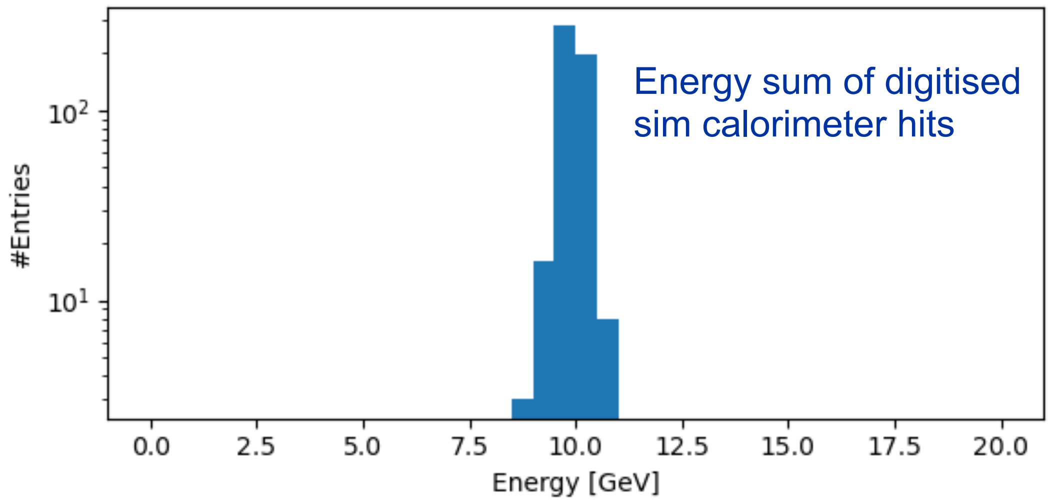
- **Particle flow reconstruction** for optimal jet energy resolutions at future colliders
- Pandora particle flow algorithm (PandoraPFA) developed to study particle flow calorimetry for CALICE
- Integration of PandoraPFA in Key4hep to **use it across multiple detector models**
- Study of PandoraPFA conducted on Nobel Liquid Argon Calorimeter of FCC
- Pandora uses radiation lengths and other material properties to determine shower shapes
- Dynamic methods used to obtain such information: very different geometry
- **First PandoraPFOs** could be observed for the LAr Calorimeter

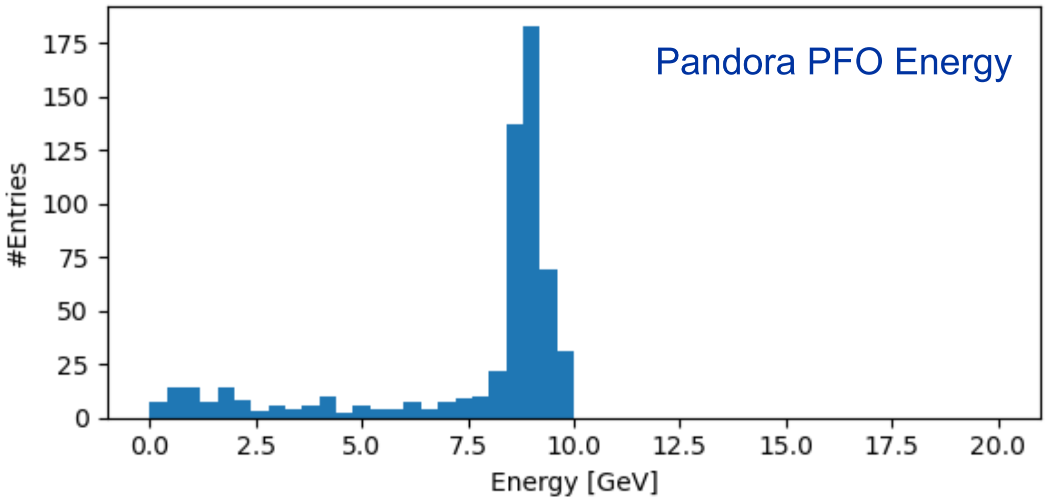




Energy of Pandora PFO

- 500 events of photons using a particle gun was simulated at an energy of 10 GeV and reconstructed
- The sum of the energies of the digitised sim calorimeter hits peaks nicely at 10 GeV as expected
- The energy of the pandora PFO seen in the second figure mostly peaked at 9 GeV and has a tail
- The correction factor for photon energies needs to be adapted to the LAr calorimeter from CLD
- With the corrections even better results expected: work in progress (Poster 198)





The Key4hep Stack

- Software provided in *stacks* deployed on cvmfs
- More than 500 packages built with Spack
- Releases in `/cvmfs/sw.hsf.org` with tagged versions of the packages
- Nightly builds in `/cvmfs/sw-nightlies.hsf.org` with the latest version of the Key4hep packages and other packages
- Easy setup with cvmfs:

```
source /cvmfs/sw.hsf.org/key4hep/releases/setup.sh            # Latest release
source /cvmfs/sw-nightlies.hsf.org/key4hep/releases/setup.sh  # Latest nightly
```

- Questions, problems, complaints and anything else related to the packages happens mostly https://github.com/key4hep/key4hep-spack

Key4hep Validation: Simulation and Reconstruction

- Validation of the algorithms, either newly developed or ported from other places is very important
- Regular check of simulation and reconstruction chain performed with the latest key4hep nightlies
- Plots of the relevant quantities are made and compared to the reference samples
- https://key4hep-validation.web.cern.ch/
- Work in progress

Summary

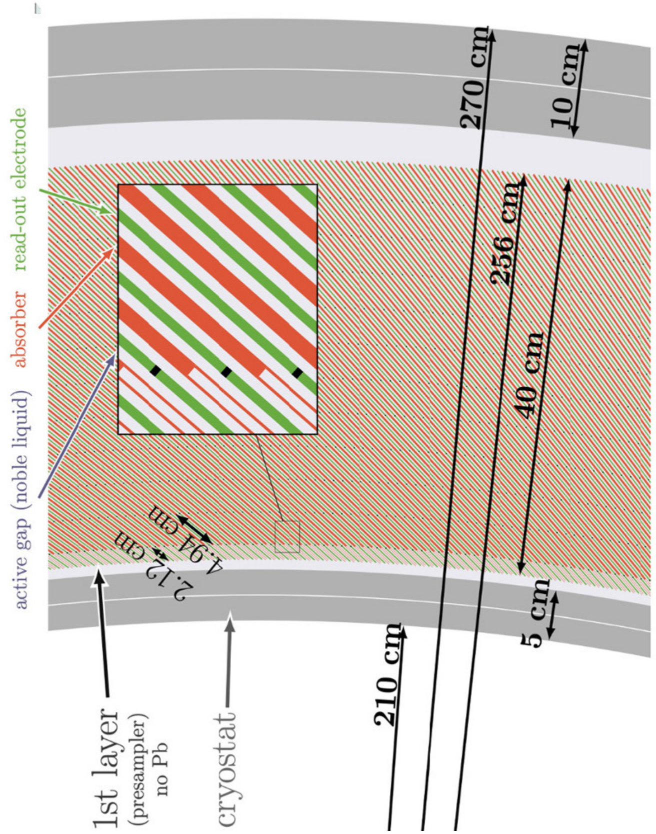
- Lots of progress in Key4hep in different areas
- It is actively developed and used for ECFA Higgs/Electroweak/Top factories studies
- A full chain from generation to analysis
- Integration of novel and existing methods to Key4hep framework
- More to come, expect more integrations and native algorithms in Key4hep framework, bug fixes and quality of life improvements

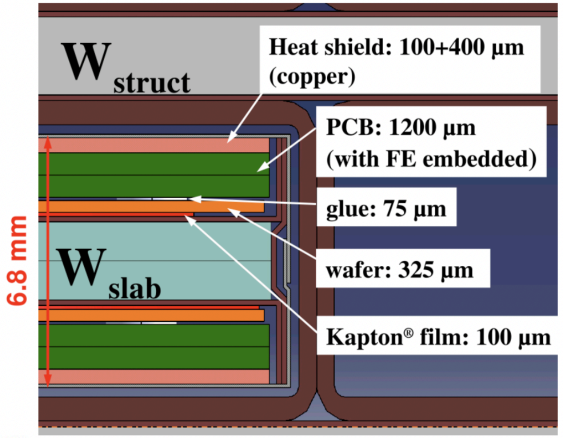
This project has received funding from the European Union's Horizon 2020 Research and Innovation programme under GA no 101004761.

Backup Slides

Geometry information for PandoraPFA

- DDMarlinPandora designed with high granularity CALICE sandwich calorimeters
- LAr calorimeter has a very different structure : an ensemble of different materials in a cell varying in density and homogeneity
- Density of material also varies from the inner radius to the outer radius of the barrel
- Moreover, the inclination of the segments play a role
- Challenging to calculate radiation length or interaction length for LAr





Pandora PFA and Key4hep

- Important ingredient for performance of future Higgs factory experiments: particle flow reconstruction for optimal jet energy resolutions
- Pandora particle flow algorithm (PandoraPFA) developed to study particle flow calorimetry
  - DDMarlin Pandora is the Marlin integration to iLCSoft framework to study particle flow at high granularity CALICE calorimeters
- Goals:
  - To enable use of PandoraPFA across multiple detector models (e.g. Liquid-Argon Calorimeter), important to integrate it into Key4hep
  - Replace the DDMarlinPandora and K4MarlinWrapper combination with DDGaudiPandora
- Study of PndoraPFA conducted on Nobel Liquid Argon Calorimeter of FCC

Pandora PFA and Layered Calorimeter Data

- PandoraPFA uses material properties e.g. radiation lengths and interaction lengths to determine the depth of the particle shower in the detector
- Particle flow clustering with Pandora uses the extensions attached to the detector geometries to provide the properties of the calorimeter
- The **`DD4hep::rec::LayeredCalorimeterData`** provides details like radiation length, interaction length and dimensions to the reconstruction algorithms

```
dd4hep::rec::LayeredCalorimeterData::Layer caloLayer;

caloLayer.distance = rad_first;
caloLayer.inner_nRadiationLengths   = value_of_x0/2.0;
caloLayer.inner_nInteractionLengths = value_of_lambda/2.0;
caloLayer.inner_thickness           = difference_bet_r1r2/2.0;
```

Material Manager

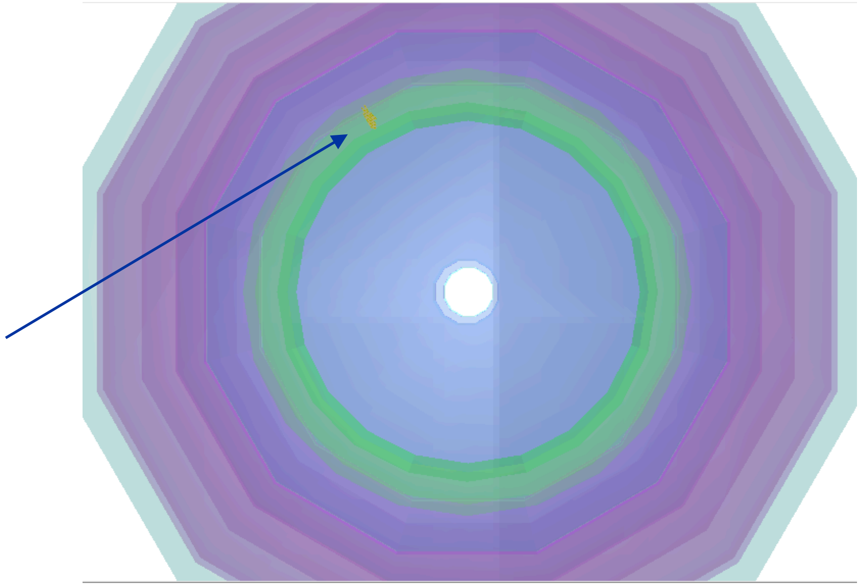
- Such information for the LAr calorimeter is obtained in a more dynamic way
- **`MaterialManager`** is a tool from DD4hep that helps extracting the necessary information between arbitrary space points
- **`MaterialManager`** returns the list of materials and their thickness along the vector
- By averaging the material between the arbitrary points material properties of the averaged material was extracted
- Crosscheck: The sum of the radiation lengths across the layers sums up to 22 X_0 as expected for the calorimeter

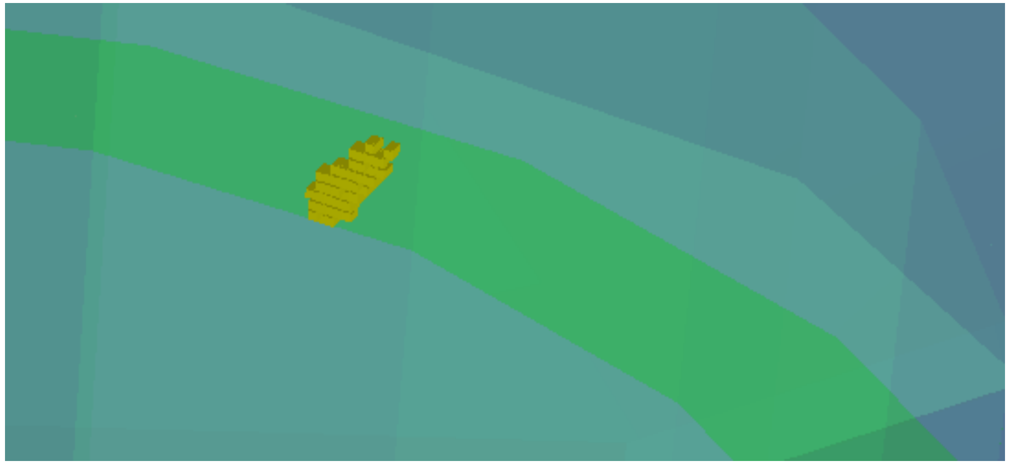
```
const dd4hep::rec::MaterialVec& materials = matMgr.materialsBetween(ivr1, ivr2);
auto mat = matMgr.createAveragedMaterial( materials) ;
nRadiationLengths = mat.radiationLength();
nInteractionLengths = mat.interactionLength();
double difference_bet_r1r2 = (ivr1-ivr2).r();
double value_of_x0 = layerHeight[il] / nRadiationLengths;
double value_of_lambda = layerHeight[il] / nInteractionLengths;
```

Pandora on other detector models

- 500 events of photons using a particle gun was simulated at an energy of 10 GeV for the CLD_LAr detector model

- By running reconstruction with all the digitized hit collections provided, PandoraPFOs could be observed for the LAr Calorimeter





I/O RNTuple

- New format to be used instead of TTree
- Significantly less space usage than TTree and better IO throughput depending on the task
- File-based and object storage
- 20% savings for ATLAS production

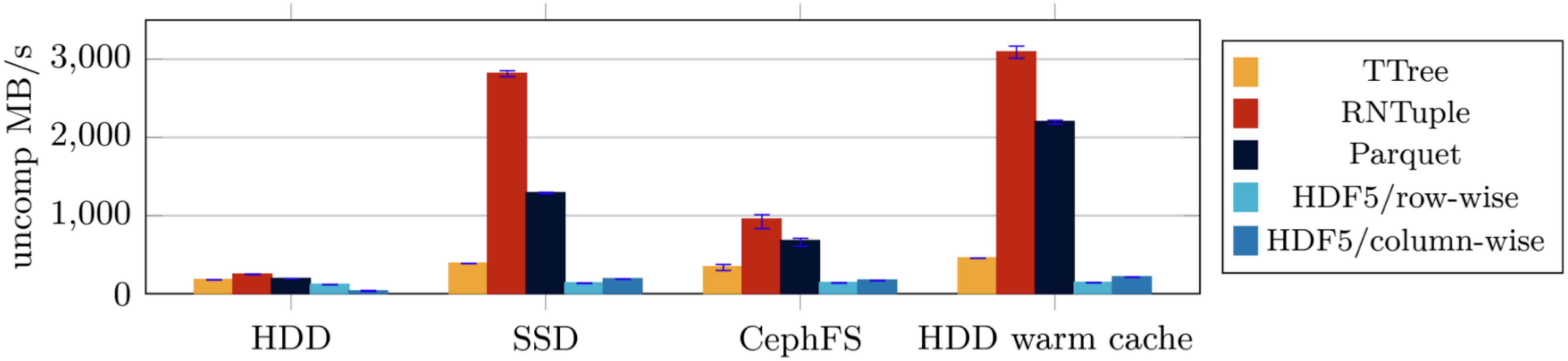


Table 1

9/121	**Test**	**#4:**	**read_frame_root**	**..**	**Passed**	**2.32**	**sec**
4/121	Test	#5:	write_frame_root	....	Passed	0.86	sec
21/121	Test	#12:	read_rntuple ...	....	Passed	0.67	sec
1/121	Test	#11:	write_rntuple ..	....	Passed	0.65	sec

home.cern