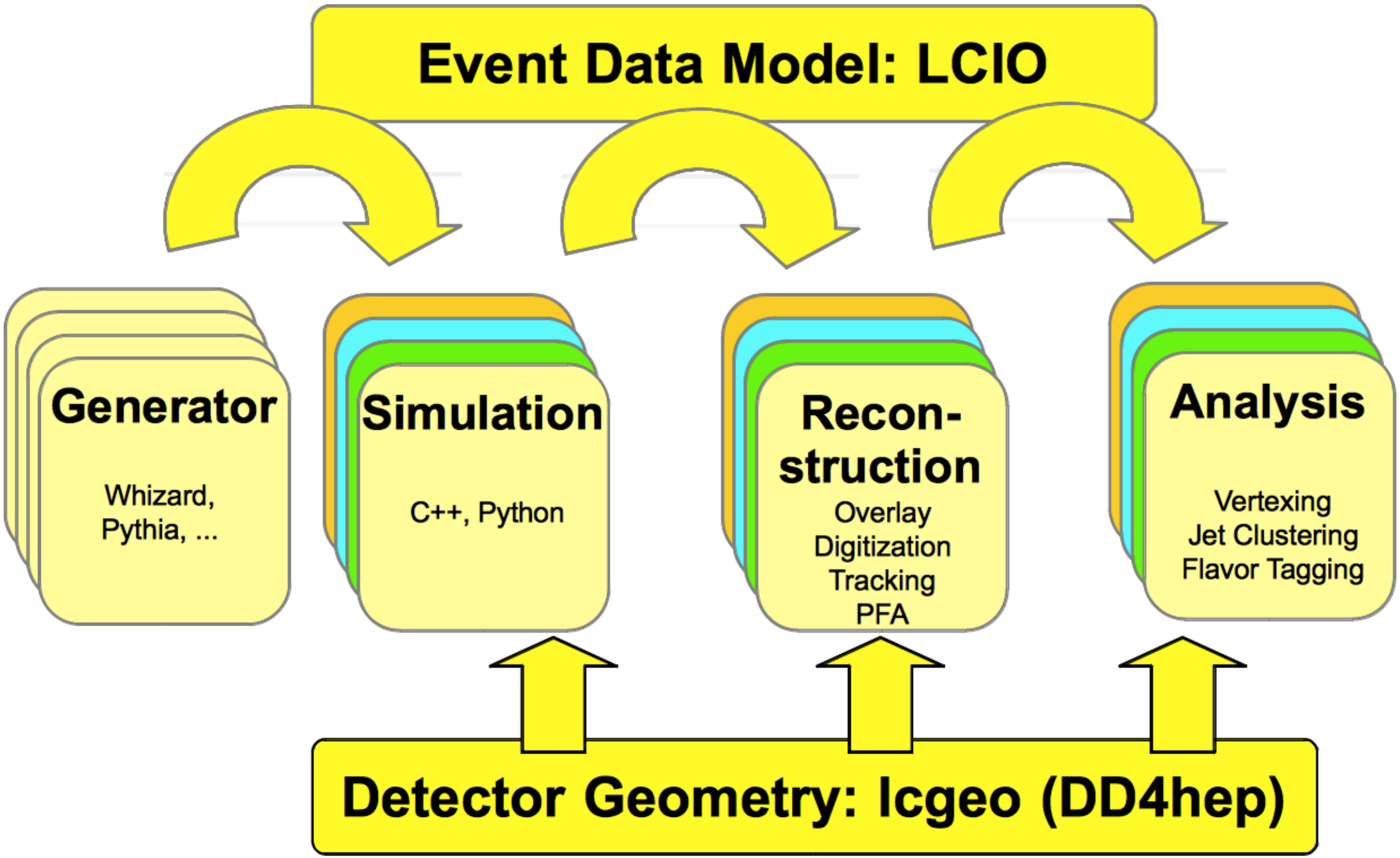
ILD Simulation Model

Shaojun Lu
shaojun.lu@desy.de

05.11.2015

Outline

- The core simulation tools
  - DD4hep/lcgeo, DDG4
- ILD simulation model
  - ILD Detector Models and Envelopes
- ILD Reconstruction
  - DD4hep/DDRec and DDMarlinPandora
- Summary


Event Data Model: LCIO
Generator
Whizard,
Pythia, ...
Simulation
C++, Python
Recon-
struction
Overlay
Digitization
Tracking
PFA
Analysis
Vertexing
Jet Clustering
Flavor Tagging
Detector Geometry: lcgeo (DD4hep)

DD4hep - Simulation Models

- Complete Detector Description
  - Includes geometry, materials, visualization, readout, alignment, calibration, etc.
- Support Full Experiment life cycle
  - Detector concept development, detector optimization, construction, operation
  - Easy transition from one phase to the next
- Consistent Description, Single source of information
  - Use in simulation, reconstruction, analysis, etc.
- Ease of Use, Few places to enter information, Minimal dependencies.

DDG4 - Geant4 gateway

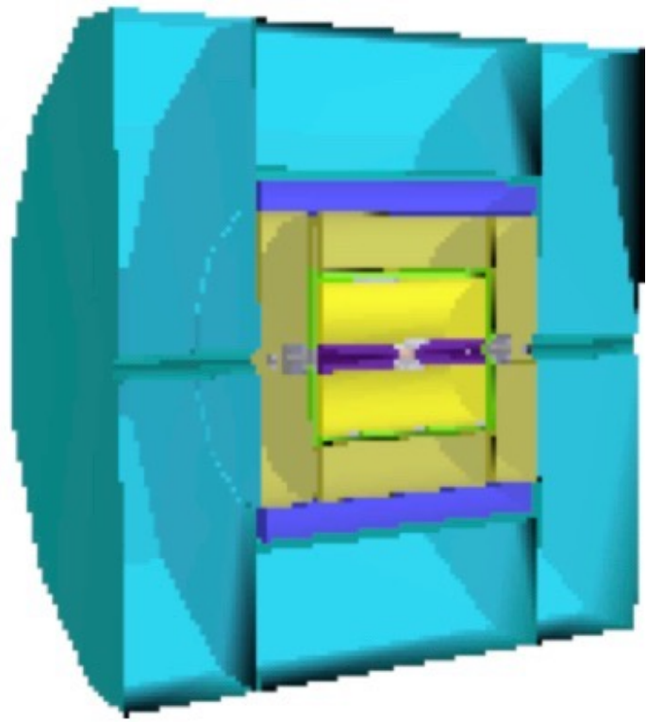
- facilitates in memory conversion of TGeo geometry to Geant4 geometry
- modular design using plugin mechanism for
  - sensitive detectors, Geant4 user actions: stepping, tracking, handling of MCTruth link for hits, …
- configuration mechanism:
  - XML, python or CINT
  - physics lists, limits, fields,…
  - define sequences for input, sensitive detectors, user actions, output,…

lcgeo - LC detector geometry description

- DD4hep based simulation models for next round of ILD detector optimization
- a common LC detector description package for ILD and CLICdp
- possible to use common (CLICdp/ILD) geometry drivers for sub-detectors, e.g. beamcal, Ecal, Hcal, …
- simulation is provided via DDG4
- use a python script for configuring and running simulation: ddsim.py

Envelopes in ILD simulation model

- introduced ‘mandatory’ envelops into the ILD simulation model, in order to
  - speed up the simulation (navigation)
  - have well defined ‘real estate’ for detectors
  - synchronize more easily with CAD models (place holder volumes)
  - facilitates development of new detector drivers and models
  - eventually allow for some well defined scaling behaviour



Envelope in ILD_o1_v05

detector	inner radius	outer radius	half length min z, max z	additional parameters	

VXD	16.0	60.0	177.6	`VXD_cone_min_z`
				`VXD_cone_max_z`
				`VXD_inner_radius_1`
					80.0
					150.0
					24.1

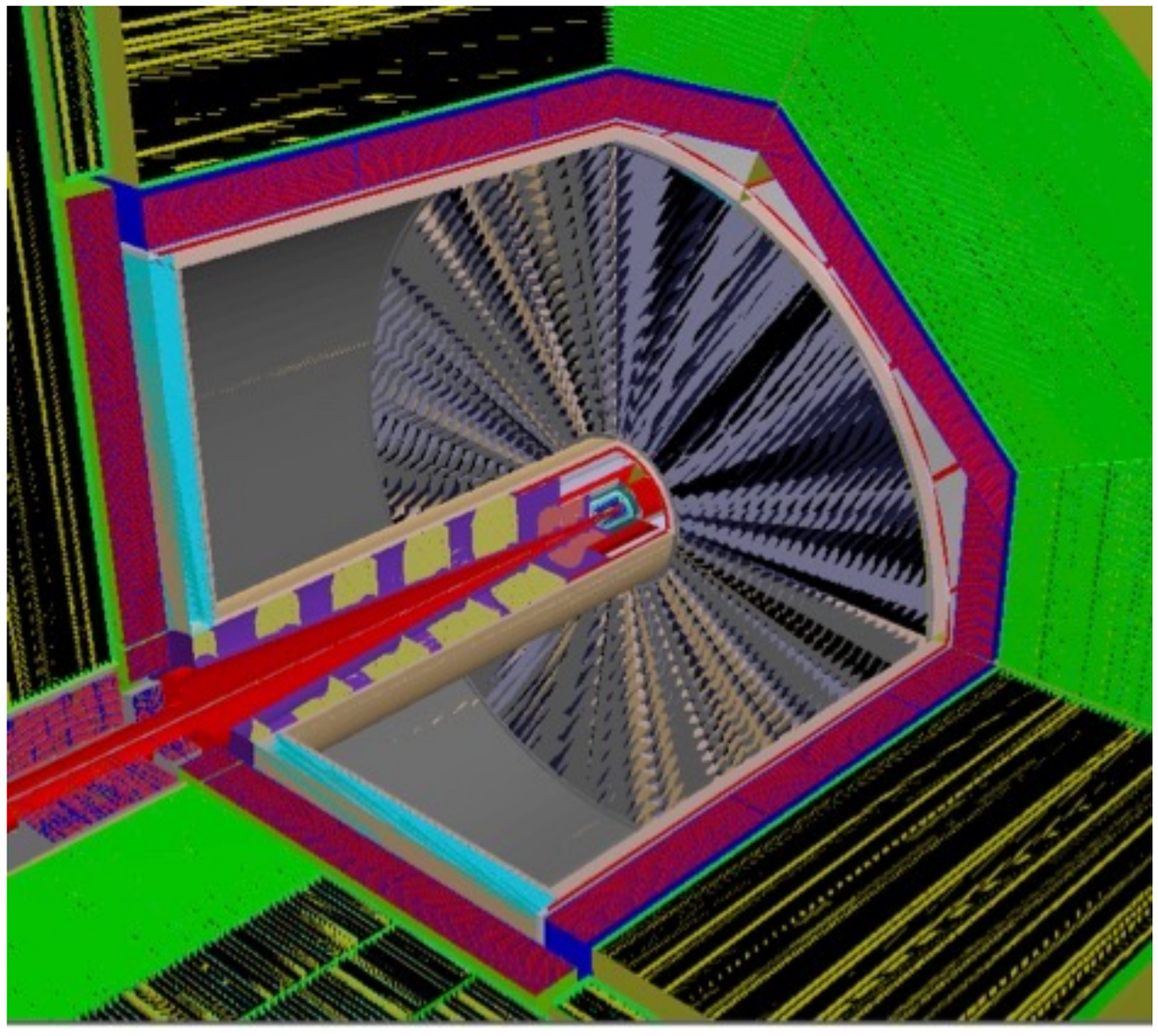
FTD	25.1	328.9	2350.0	`FTD_outer_radius_1`
				`FTD_outer_radius_2`
				`FTD_min_z_0`
				`FTD_min_z_1`
				`FTD_min_z_2`
				`FTD_cone_min_z`
				`FTD_cone_radius`
					152.8
					299.7
					177.7
					368.2
					644.2
					230.0
					184.1

SIT	152.9	324.6	644.1	`SIT_outer_radius_1` `SIT_half_length_1`	299.8 368.1
TPC	329.0	1808.0	2350.0		
SET	1808.1	1827.9	2350.0		
Ecal	1843.0	2028.0	2350.0	`Ecal_Hcal_symmetry` `Ecal_symmetry`	8.0 8.0
EcalEndcap	400.0	2088.8	2450.0, 2635.0		
EcalEndcapRing	250.0	390.0	2450.0, 2635.0		
Hcal	2058.0	3395.5	2350.0	`Hcal_inner_symmetry`	8.0
HcalEndcap	350.0	3395.5	2670.7, 3957.7	`EcalEndcap_symmetry`	8.0
HcalEndcapRing	2138.8	3137.0	2450.0, 2635.0	`HcalEndcapRing_symmetry`	8.0
Coil	3425.0	4175.0	3872.0		
Yoke	4424.0	7725.0	4047.0	`Yoke_symmetry`	12.0
YokeEndcap	300.0	7725.0	4072.0, 7373.0	`YokeEndcap_symmetry`	12.0
YokeEndcapPlug	300.0	3395.5	3981.5, 4072.0	`YokeEndcapPlug_symmetry`	12.0
BeamCal	20.0	150.0	3475.0, 3695.0	`BeamCal_thickness` `BeamCal_tubeIncoming_radius`	220.0 15.0
LHCal	100.0	325.0	2680.0, 3200.0	`LHCal_thickness`	520.0
LumiCal	80.0	195.2	2500.0, 2630.7	`LumiCal_thickness`	130.7

- goal: have set of drivers that use a well defined set of parameters and have a well defined scaling behaviour

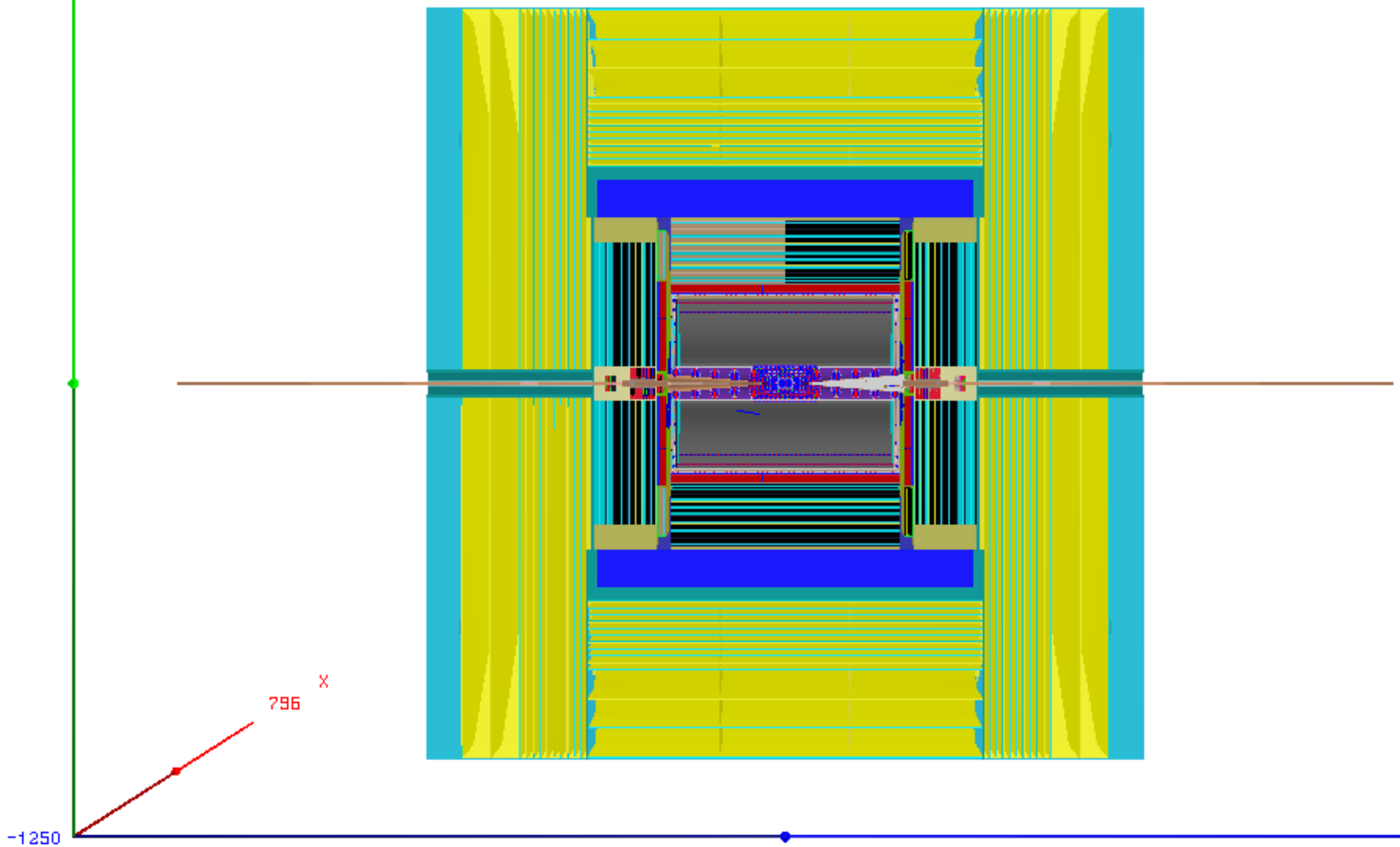
ILD_o1_v05 model in lcgeo

- ported current Mokka model ILD_o1_v05 to DD4hep
  - VXD, FTD, SIT, TPC, SET, beam pipe (F. Gaede)
  - Ecal, Hcal, Yoke (S.Lu)
  - Beamcal, Lcal, LHcal (A.Sailier, M.Petric)



- start to modify and improve them

Overview of ILD_o1_v05



- Next few sides will provide a little bit detail introduction with zoom in.

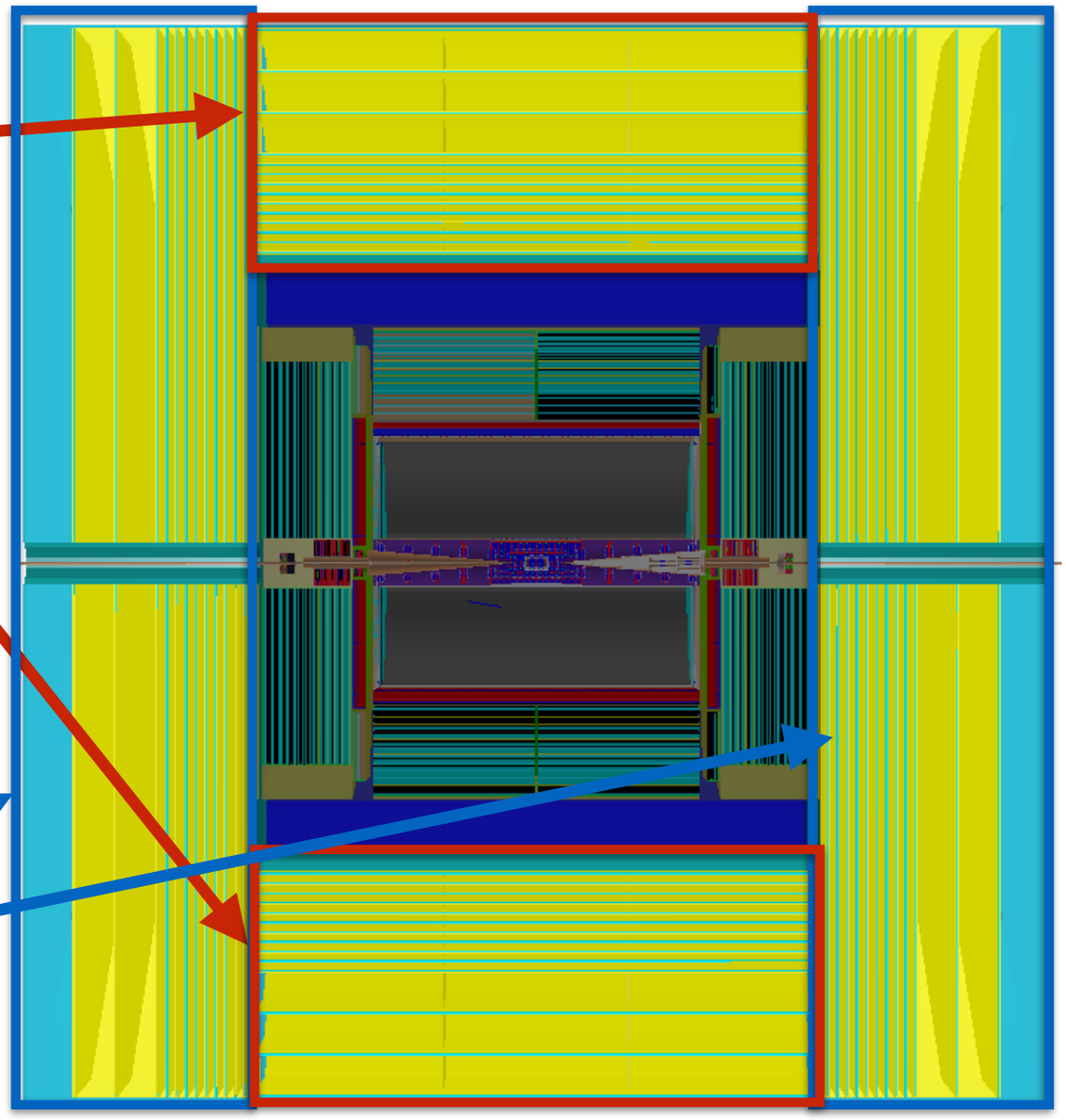
Yoke in DD4hep/lcgeo

Yoke Barrel

- There is three modules in Barrel

- 12 folds Endcaps

Yoke Endcaps



Hcal/Ecal in DD4hep/lcgeo

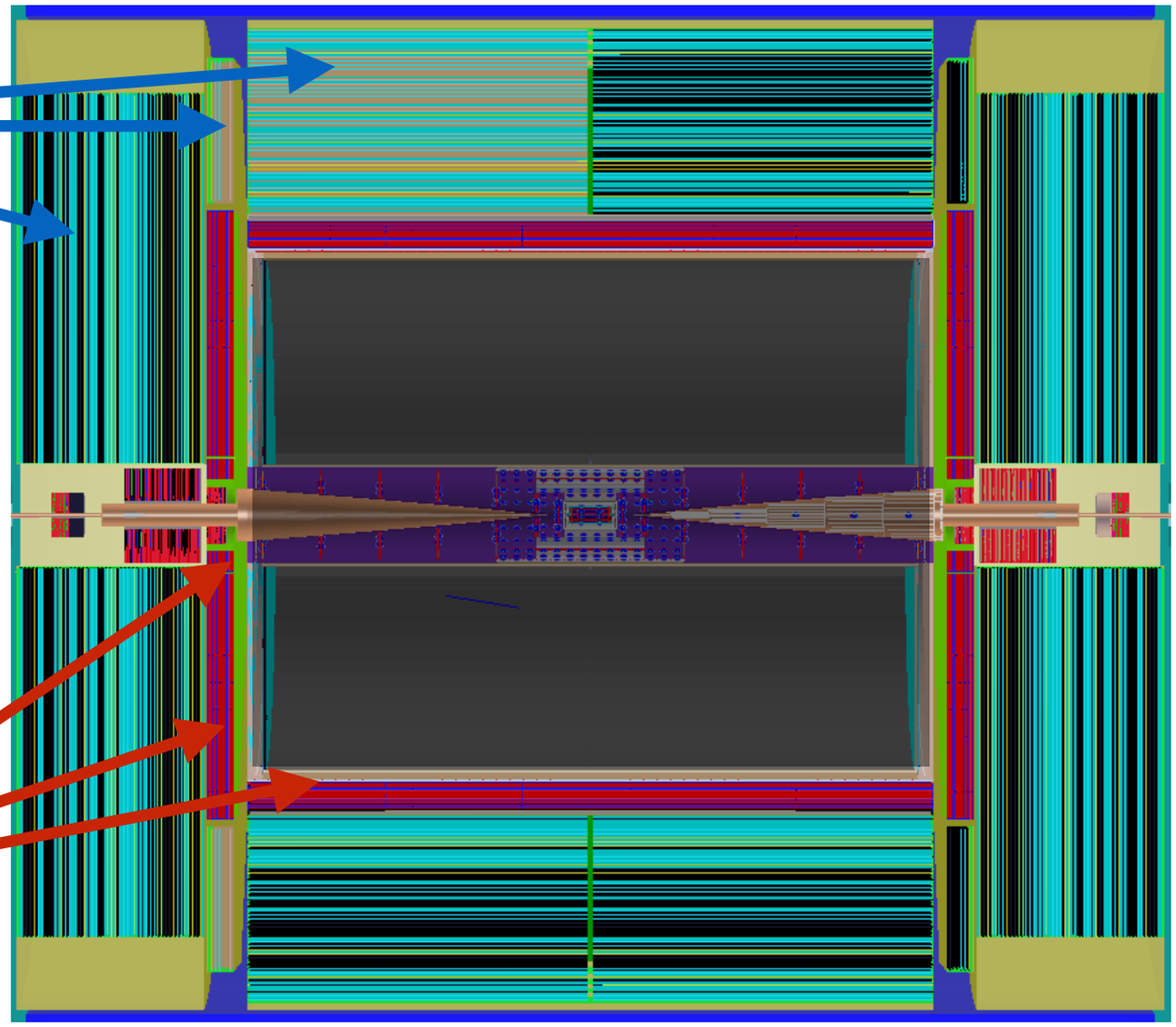
AHcal

- AHcal has two modules in Barrel

- Ecal has five modules in Barrel

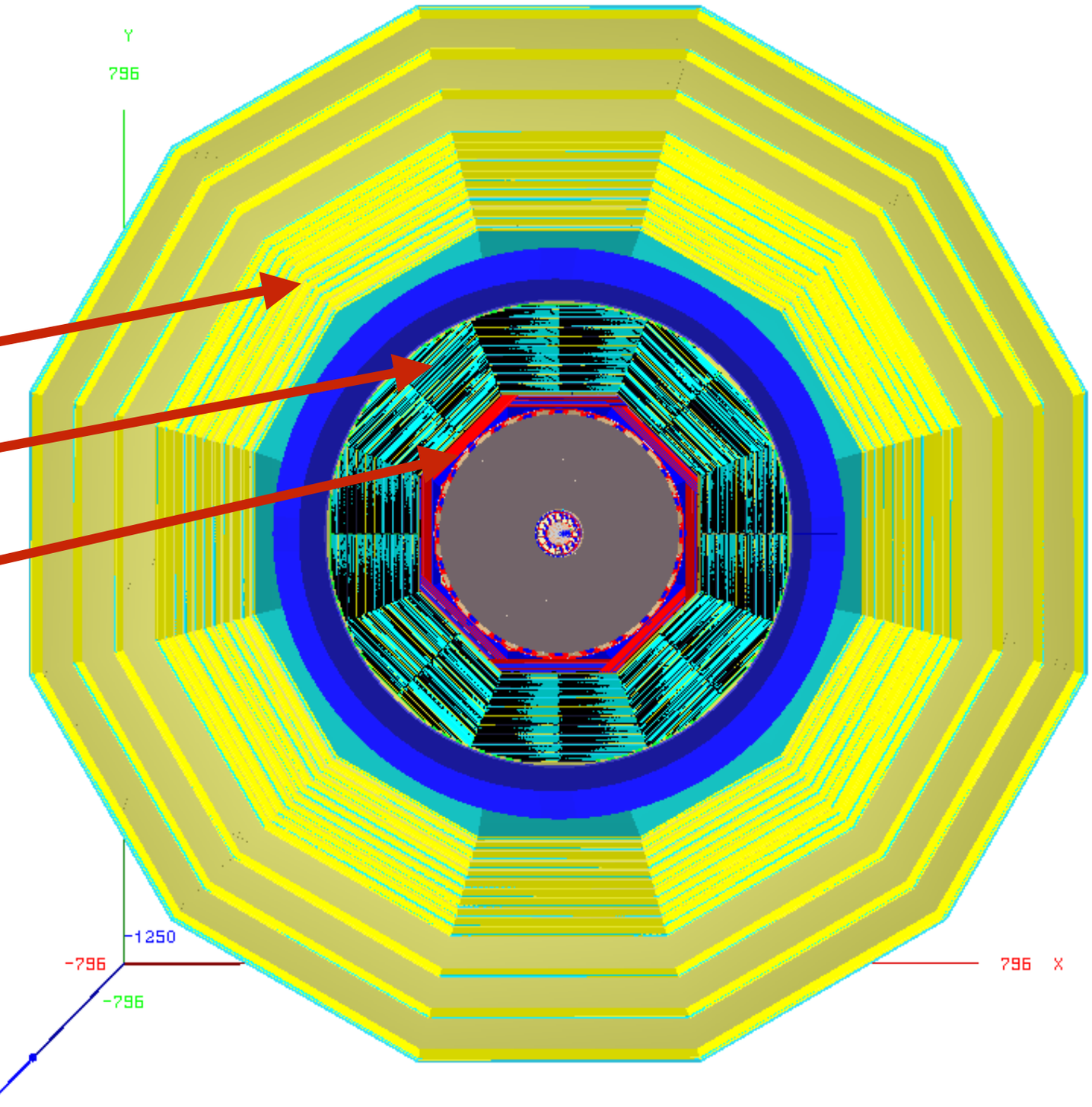
- Both have 8 folds

Ecal



Front view of ILD_o1_v05

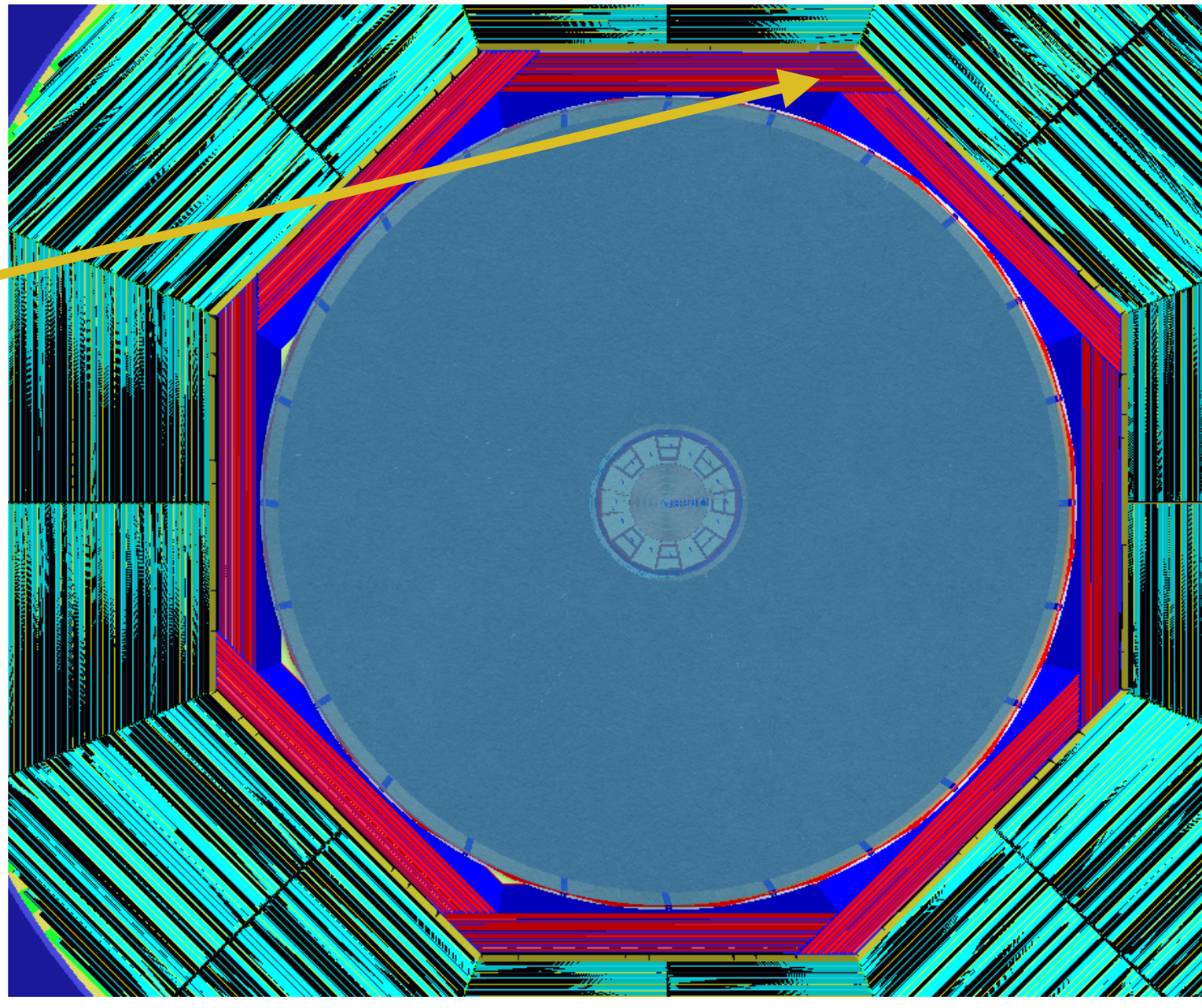
- Yoke has 12 folds
- Hcal has 8 folds
- Ecal has 8 folds



Ecal Barrel in DD4hep/lcgeo

- Clear Ecal layer structure could be seen here.

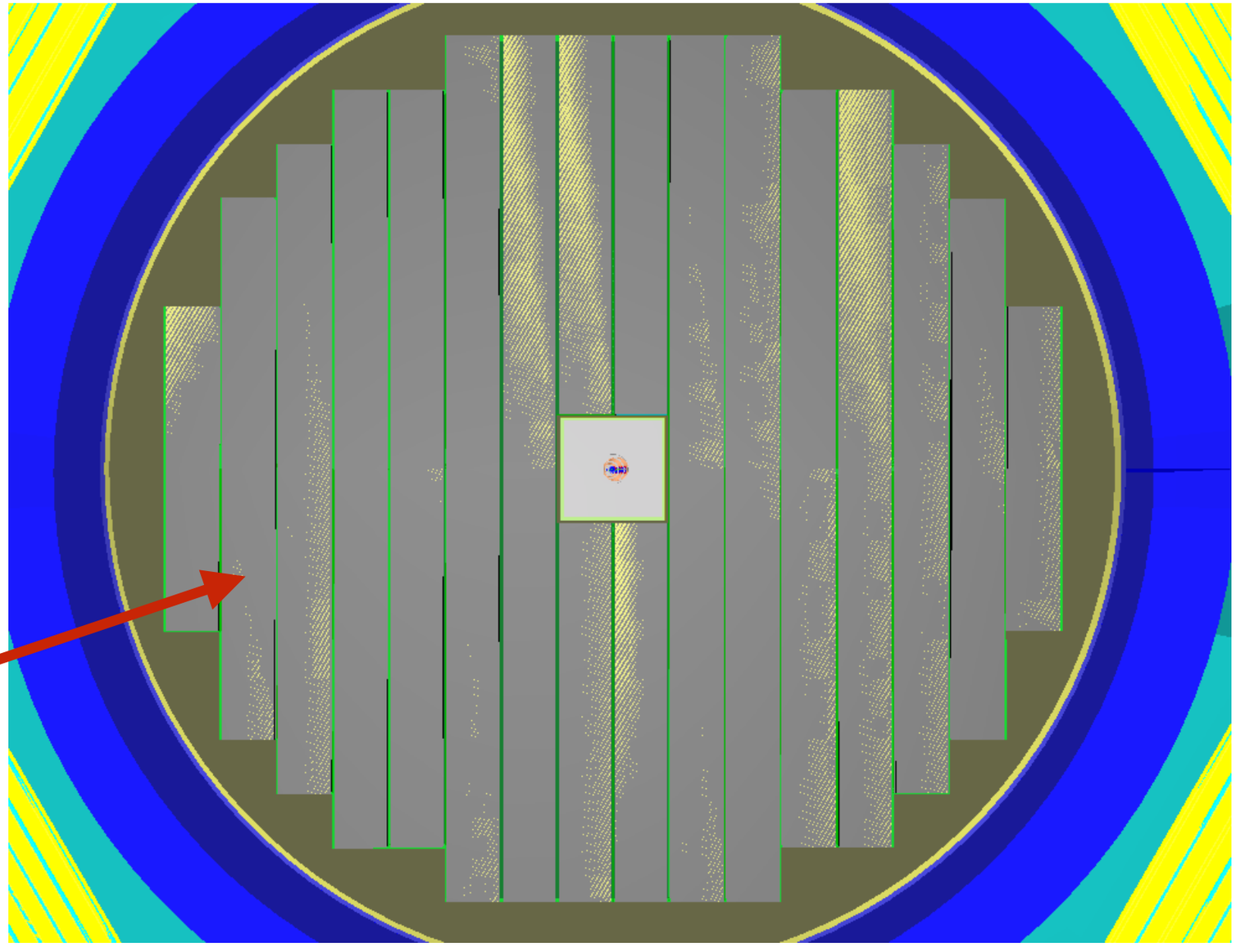
- Ecal has a double layers structure where could be seen clearly in the event display later.



AHcal Endcaps in DD4hep

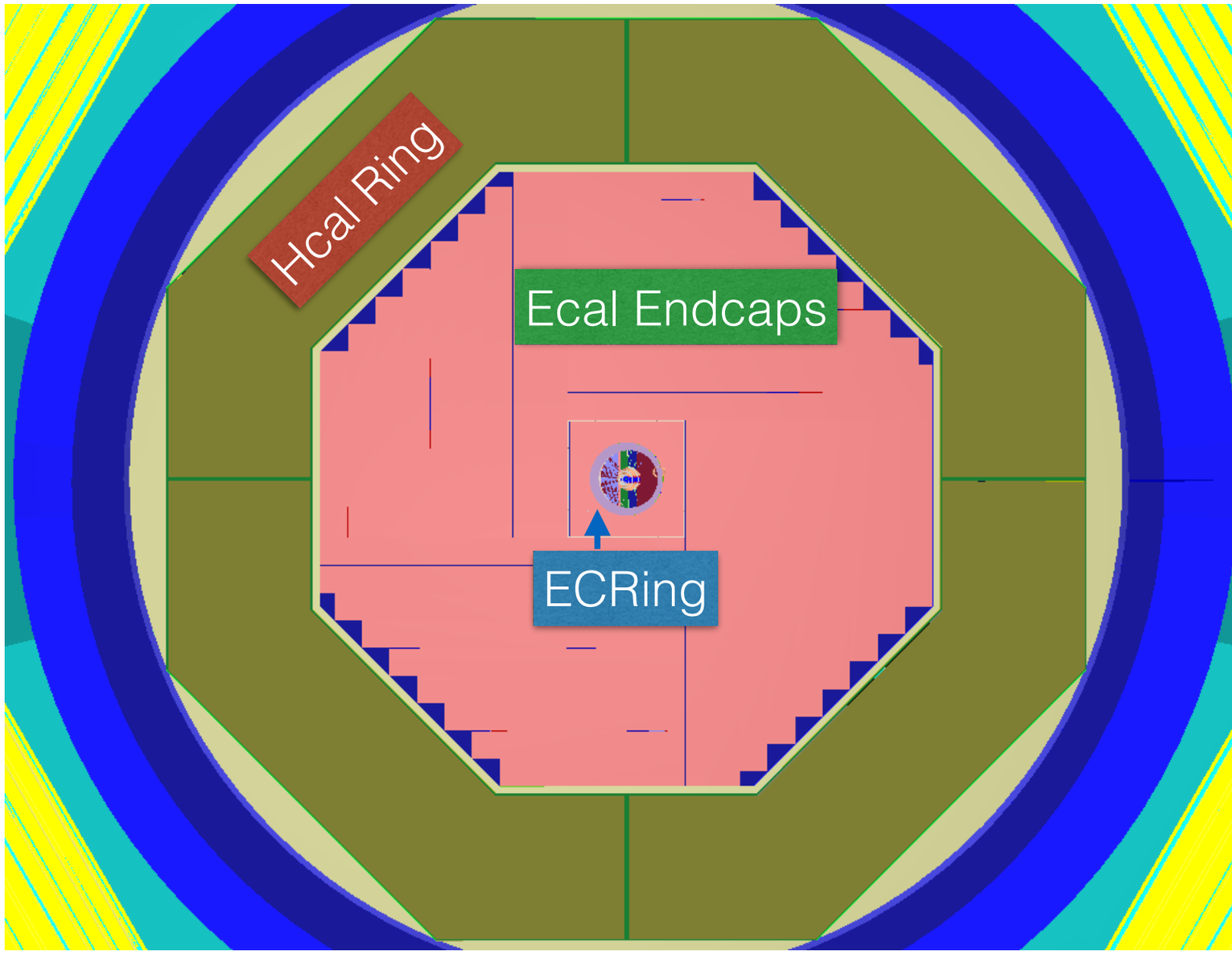
- Hcal Endcaps, the lastest HBU engineer design model has been implemented in the lcgeo.

- It include 16 towers, which has the HBU type size.



Endcaps model in lcgeo

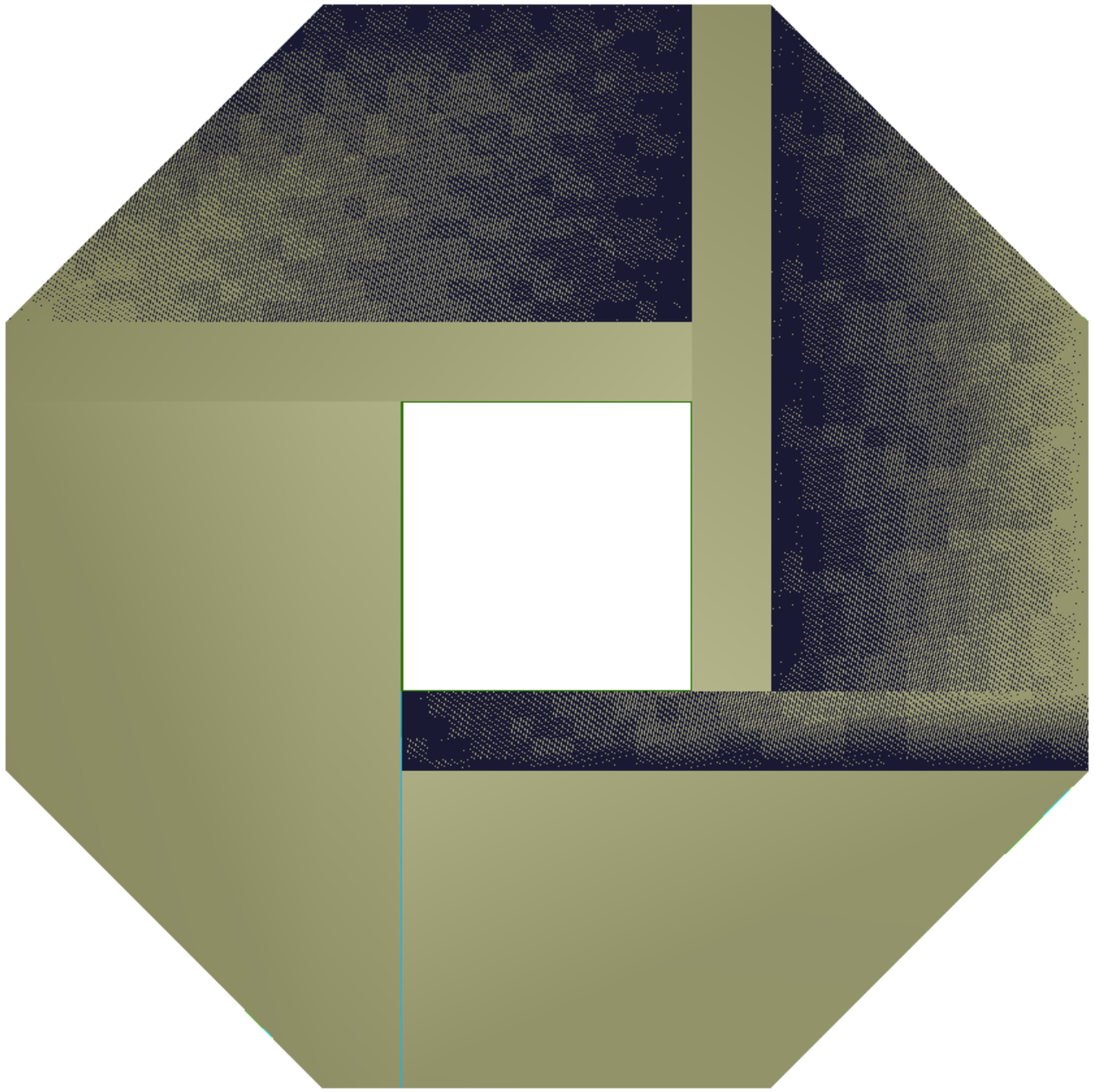
- Simple Hcal Ring, which ported from Mokka
- The EcalEndcap and EcalECRing, which ported from Mokka



New EcalEndcaps

- There is a new design of Ecal Endcaps which has been implemented into DD4hep

- The ILD will follow the updated new Ecal Endcaps, it may be used in ILD in the future



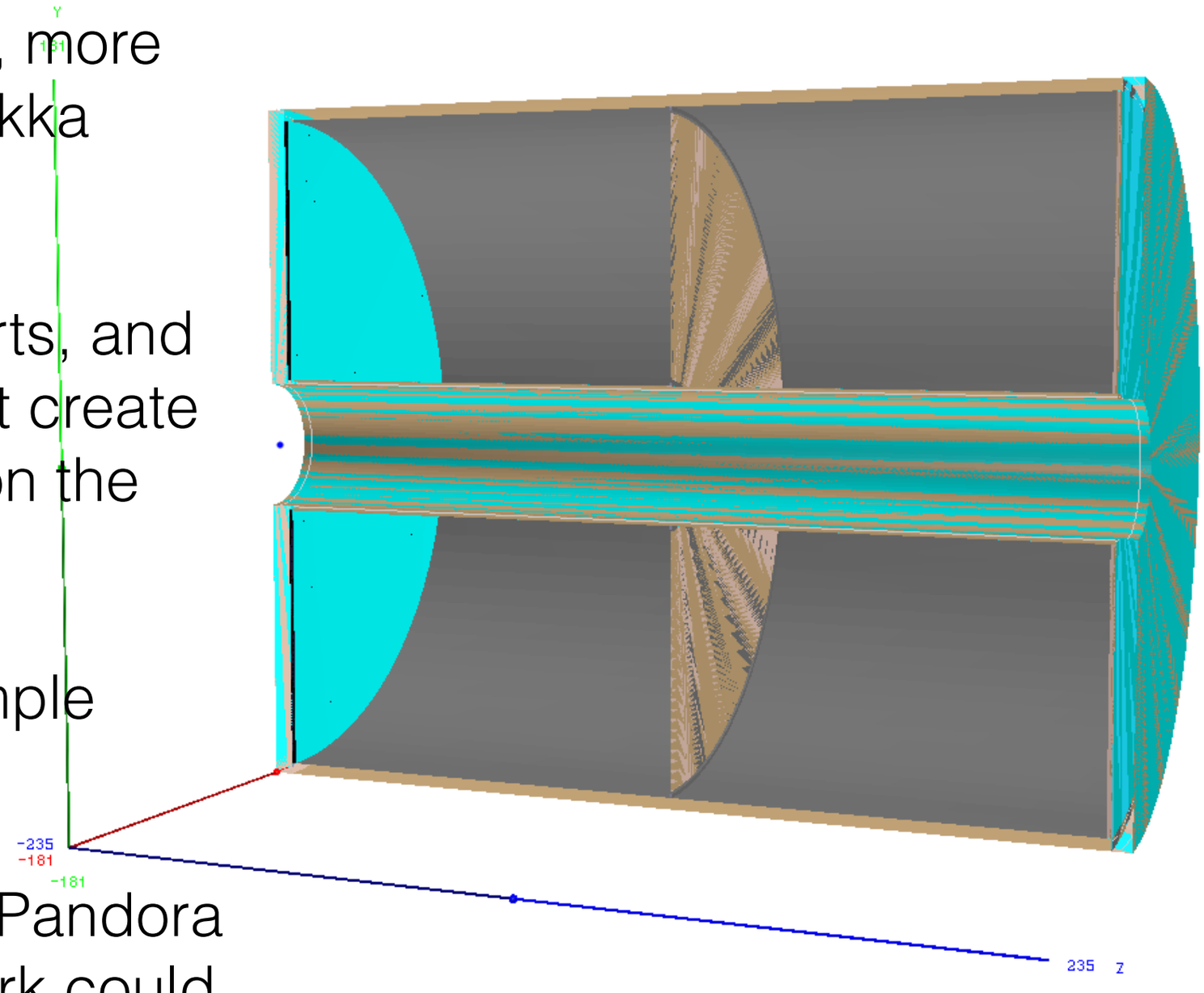
TPC in lcgeo

- TPC has been updated, more realistic compare to Mokka DBD version.

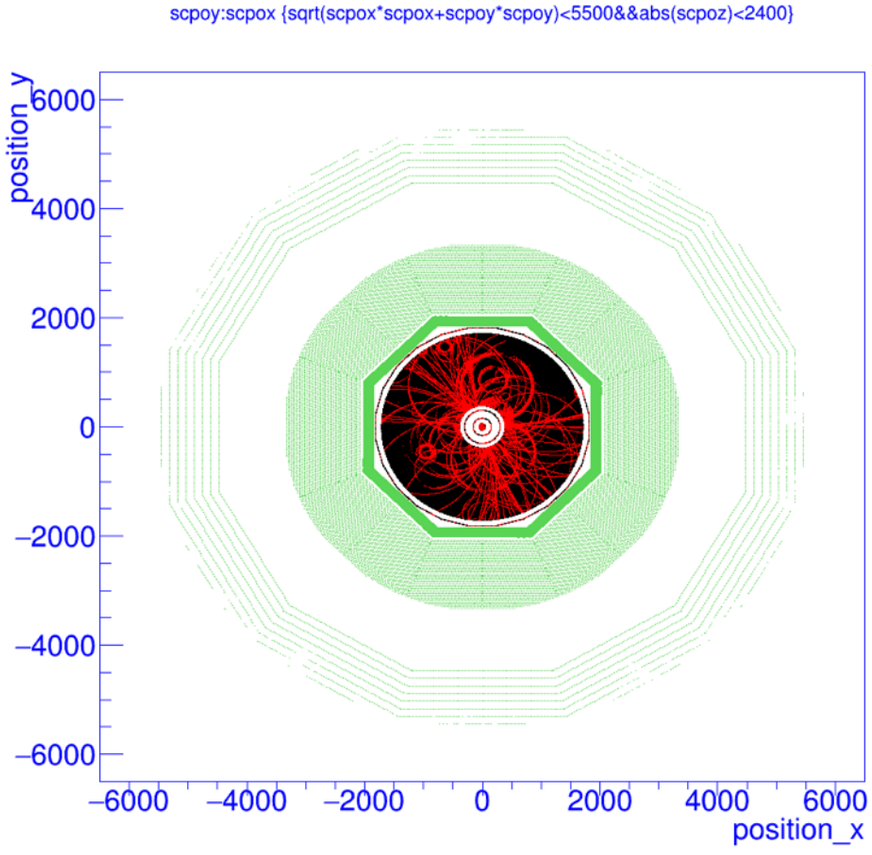
- The volume has two parts, and the TPC digitiser will not create TPC hits in the middle on the cathode.

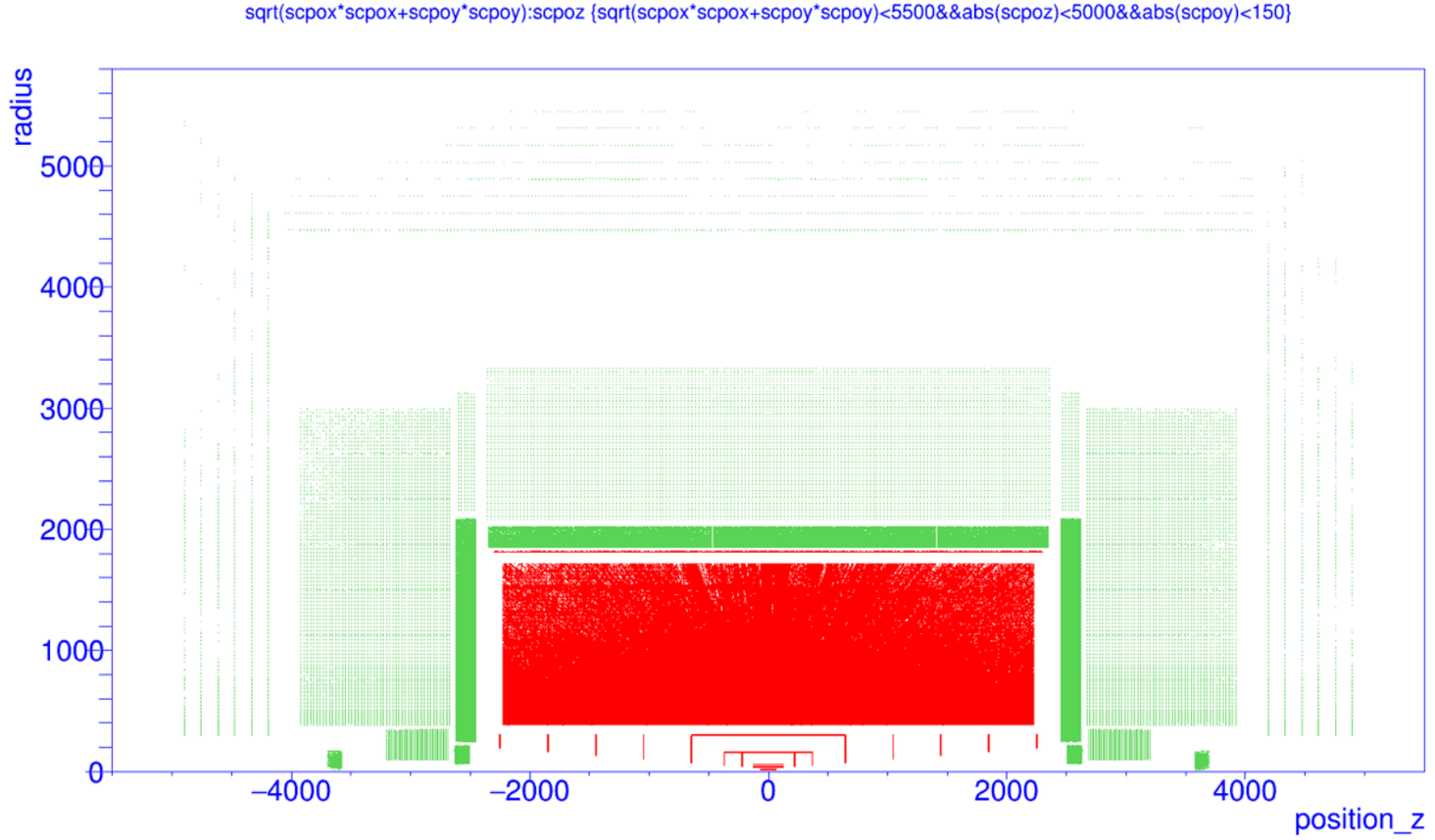
- Mokka has only one simple volume.

- New DD4hep/DDMarlinPandora reconstruction framework could handle this.



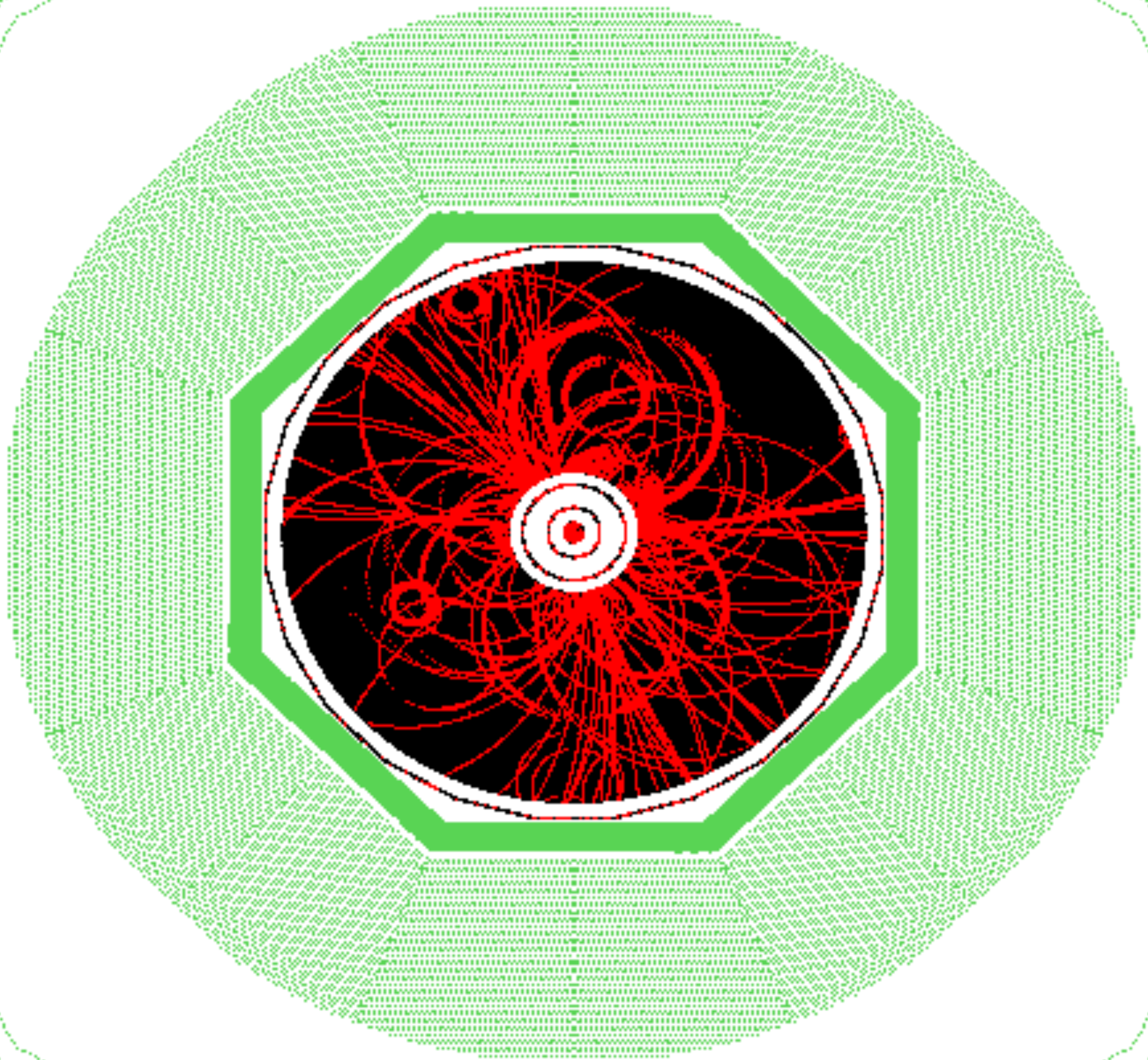
Hits map in ILD_o1_v05

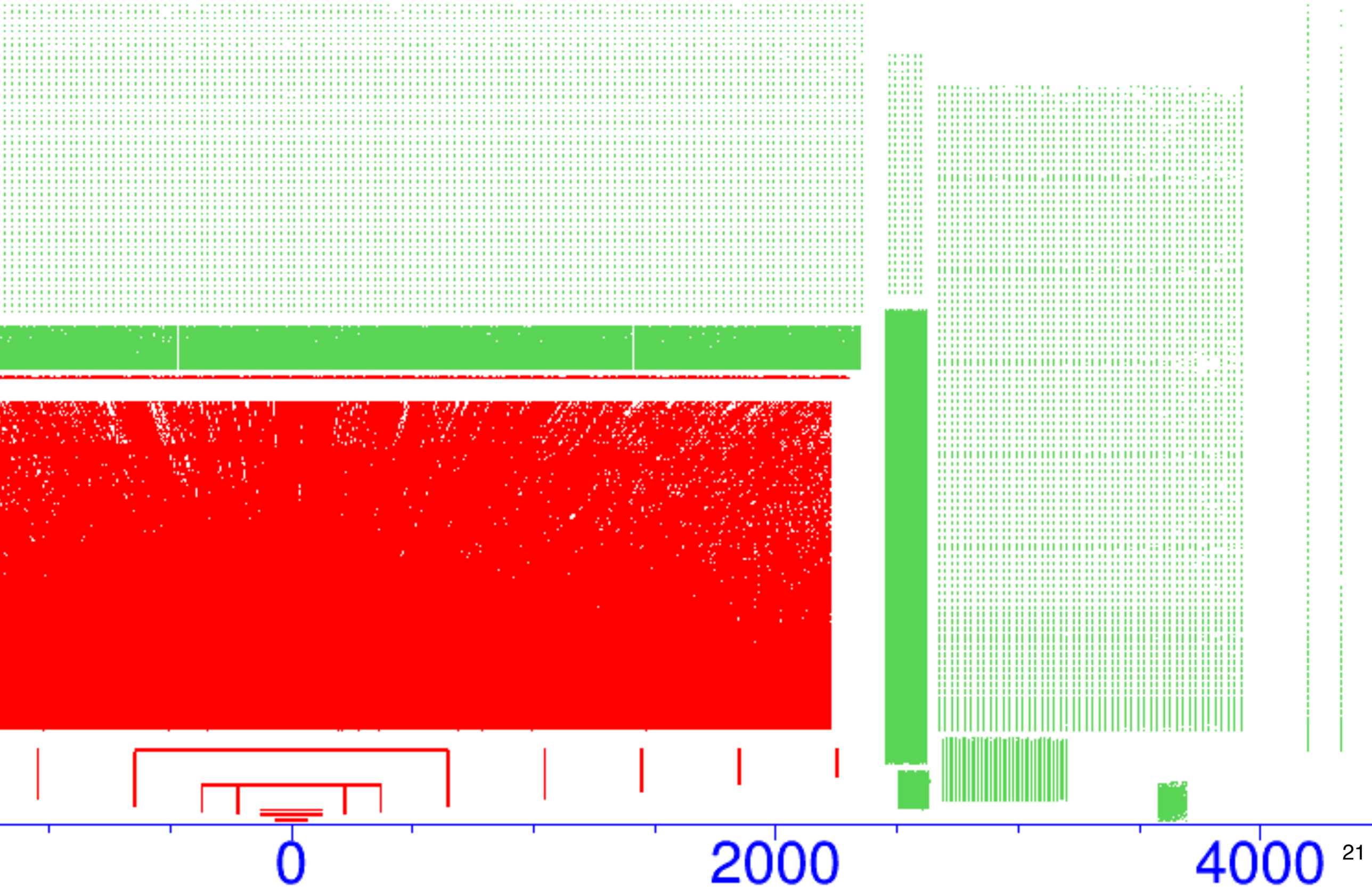




- can now fully simulate new DD4hep based ILD_o1_v05 model (so far using ‘canonical’ sensitive detectors)

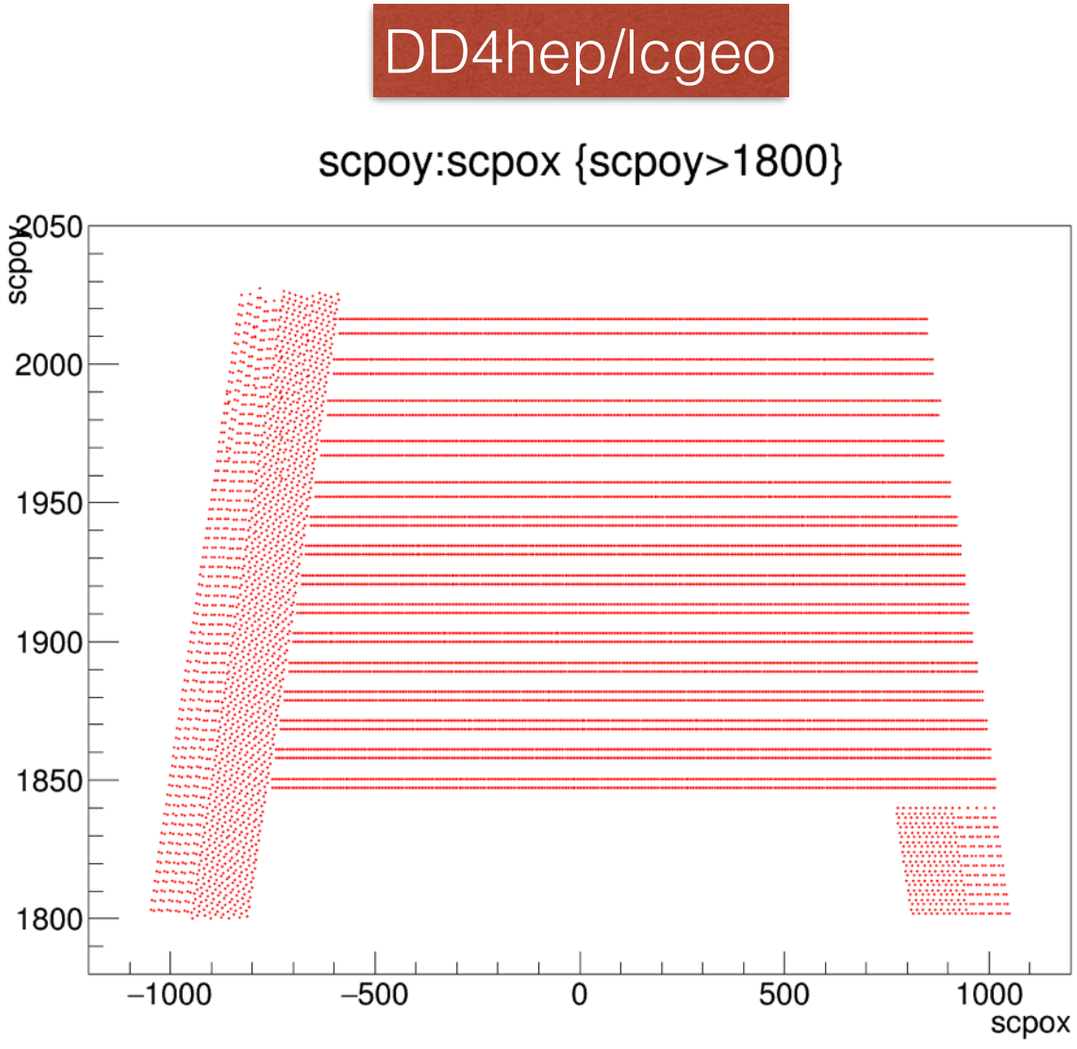
- The green part are calorimeters, and the red part are trackers. The inner Si trackers are displayed clearly here in the hits map



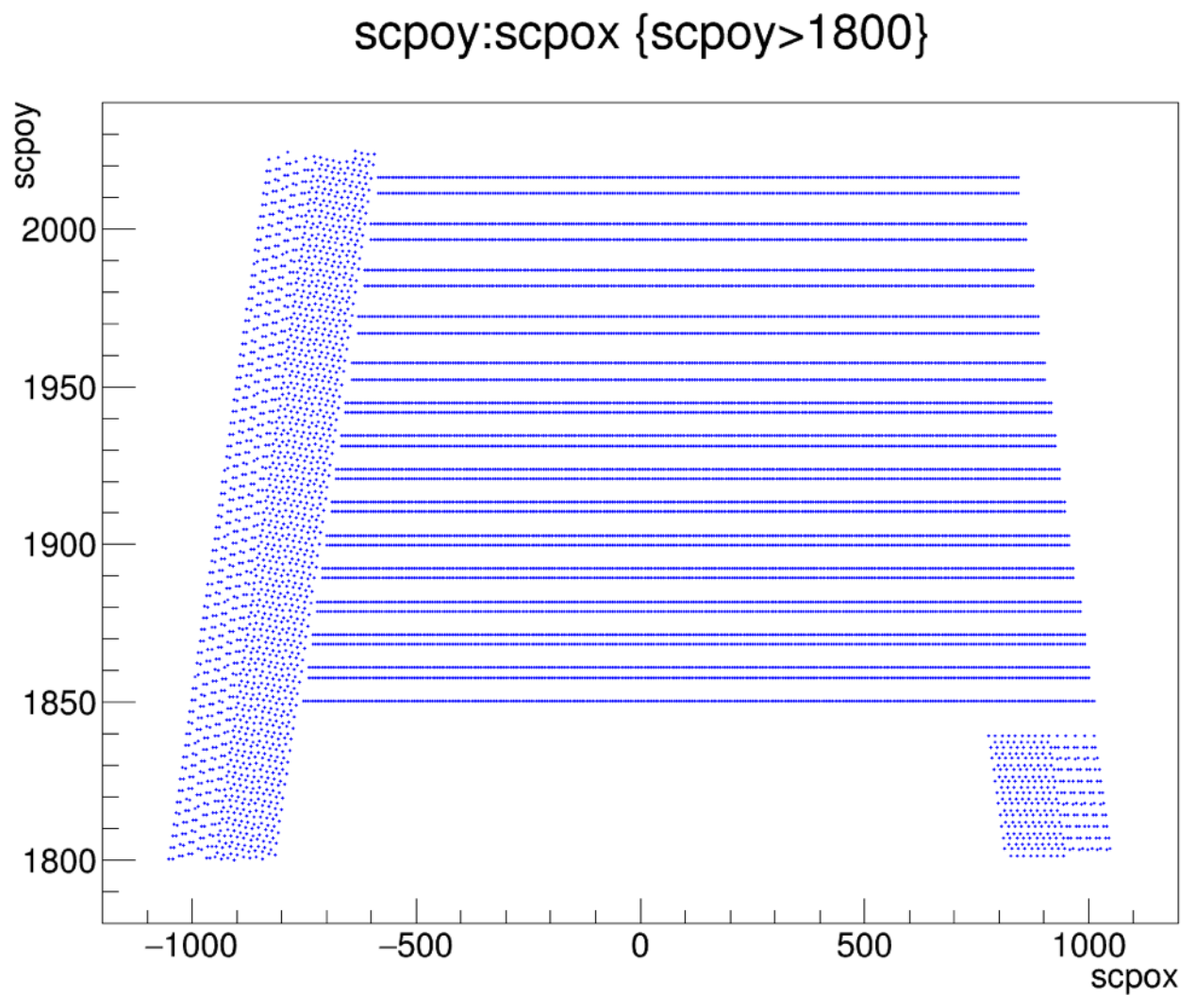

0
2000
4000

Ecal Hits map in ILD_o1_v05

DD4hep/lcgeo



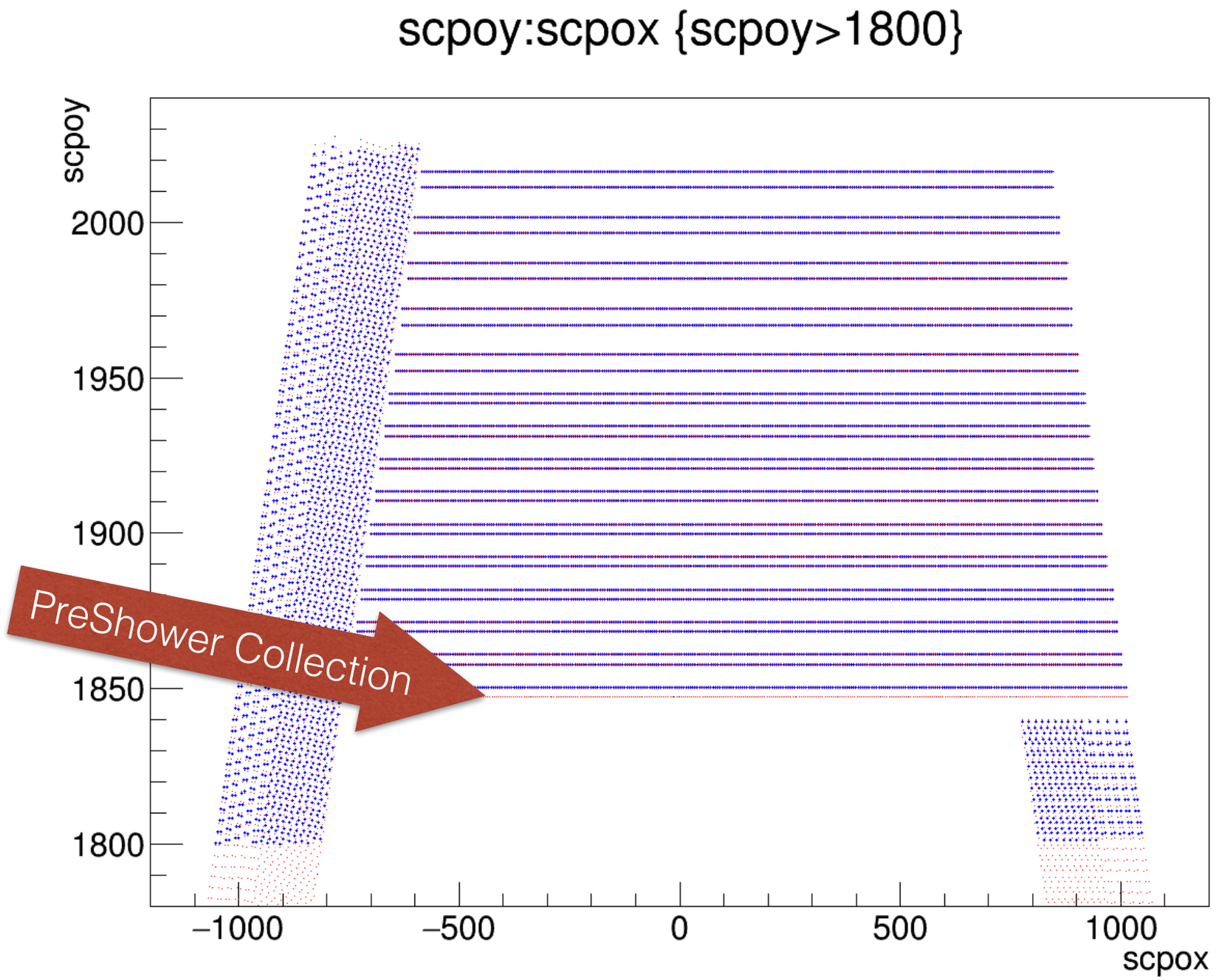
Mokka

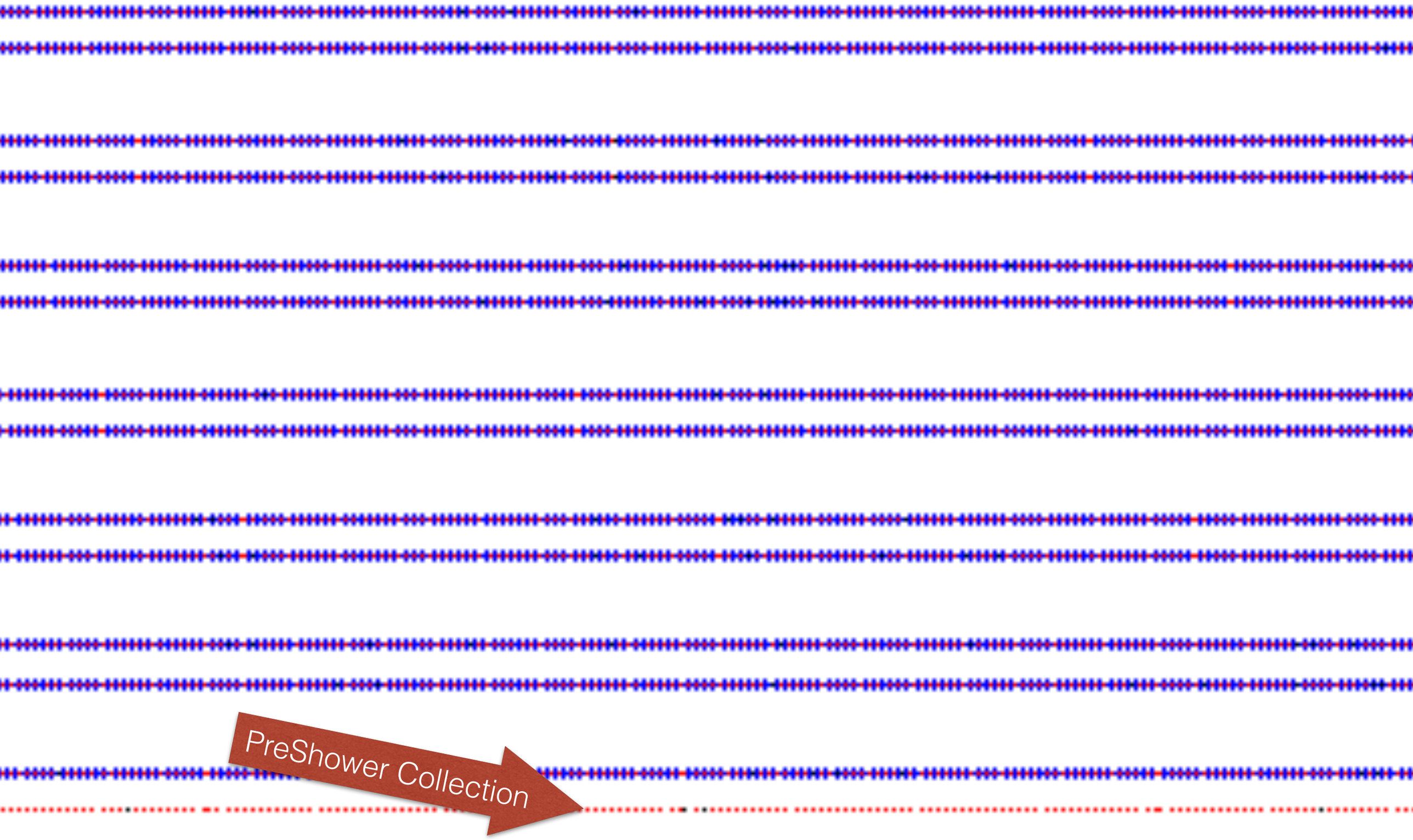


- one issue found during the validation, the most inner Ecal has separated collection, EcalPreShower collection.

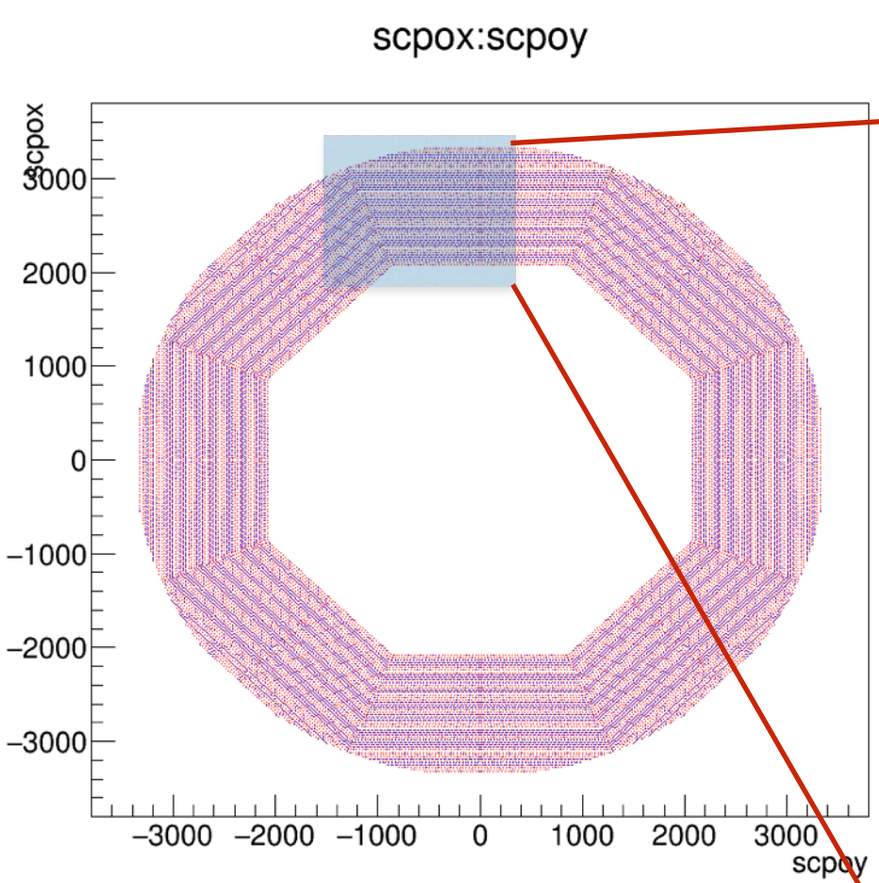
Validation of the Ecal Geometry

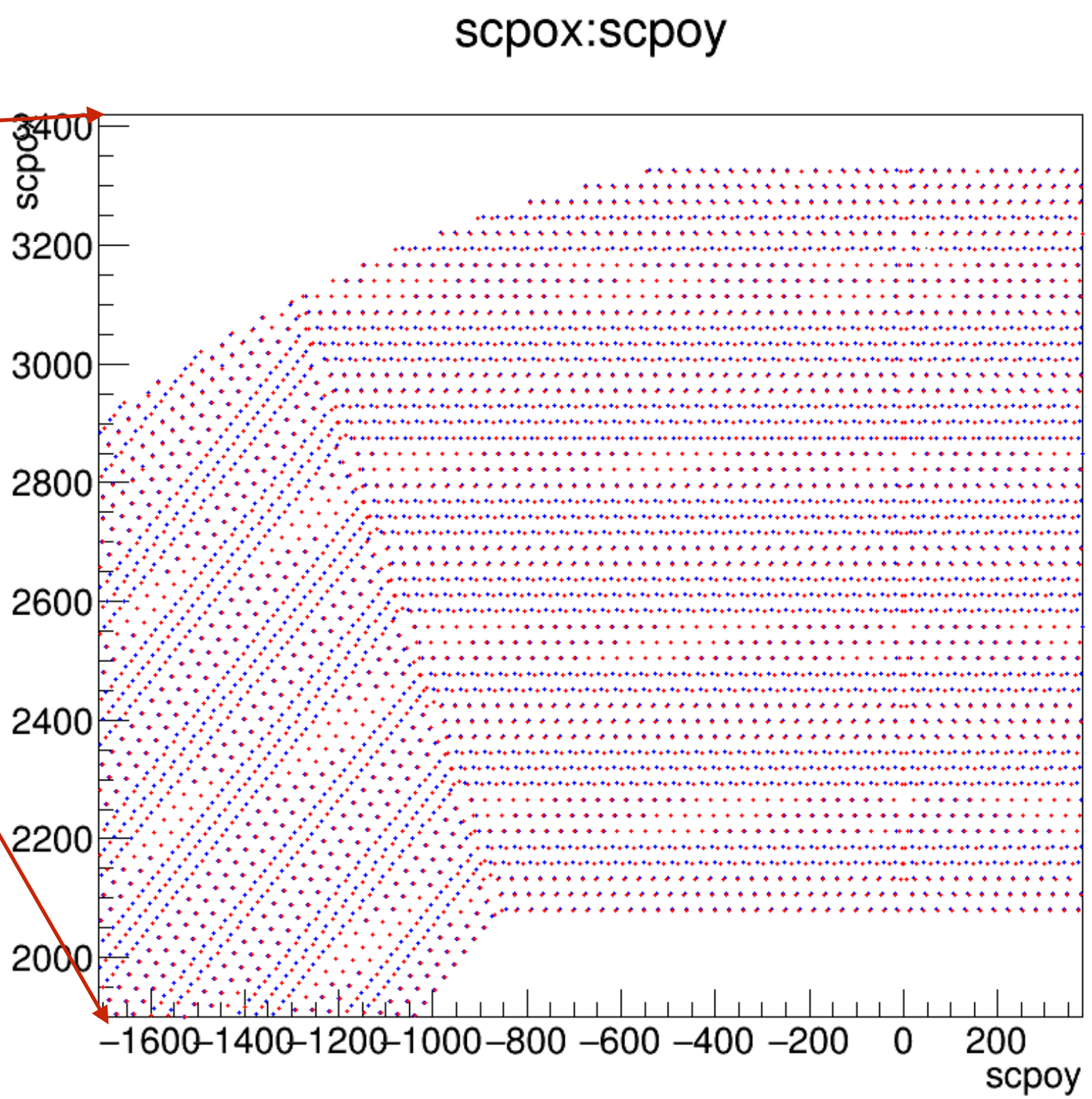
- The red inner most layer has been put into EcalPreShower collection.
- The geometry sensitive layer centre represent the Mokka.
- It clearly validated the geometry



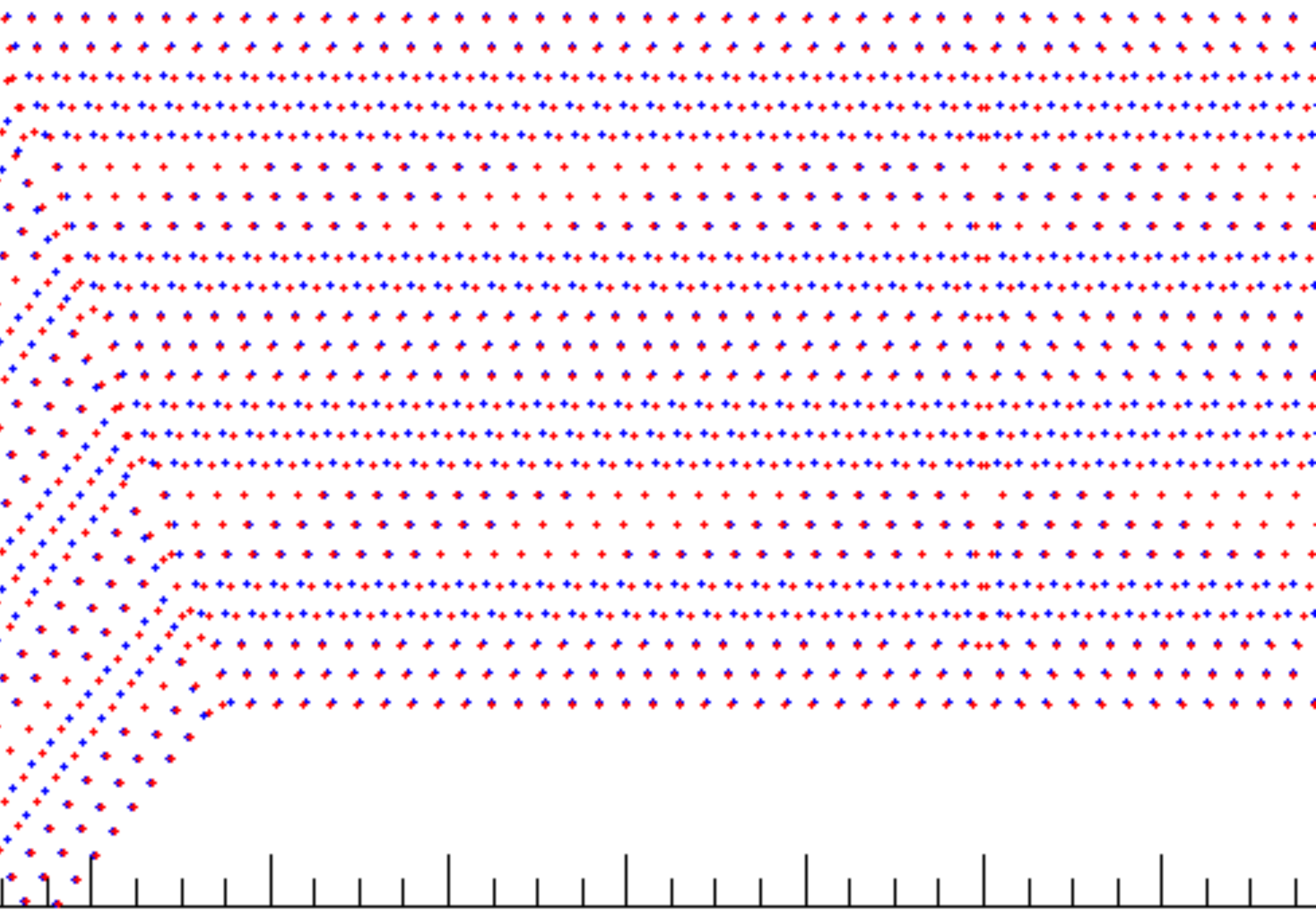

PreShower Collection

Validation of the AHcal Geometry





- It clearly validated the AHcal geometry
- DD4hep in red


−1000
−800
−600
−400
−200
0
200

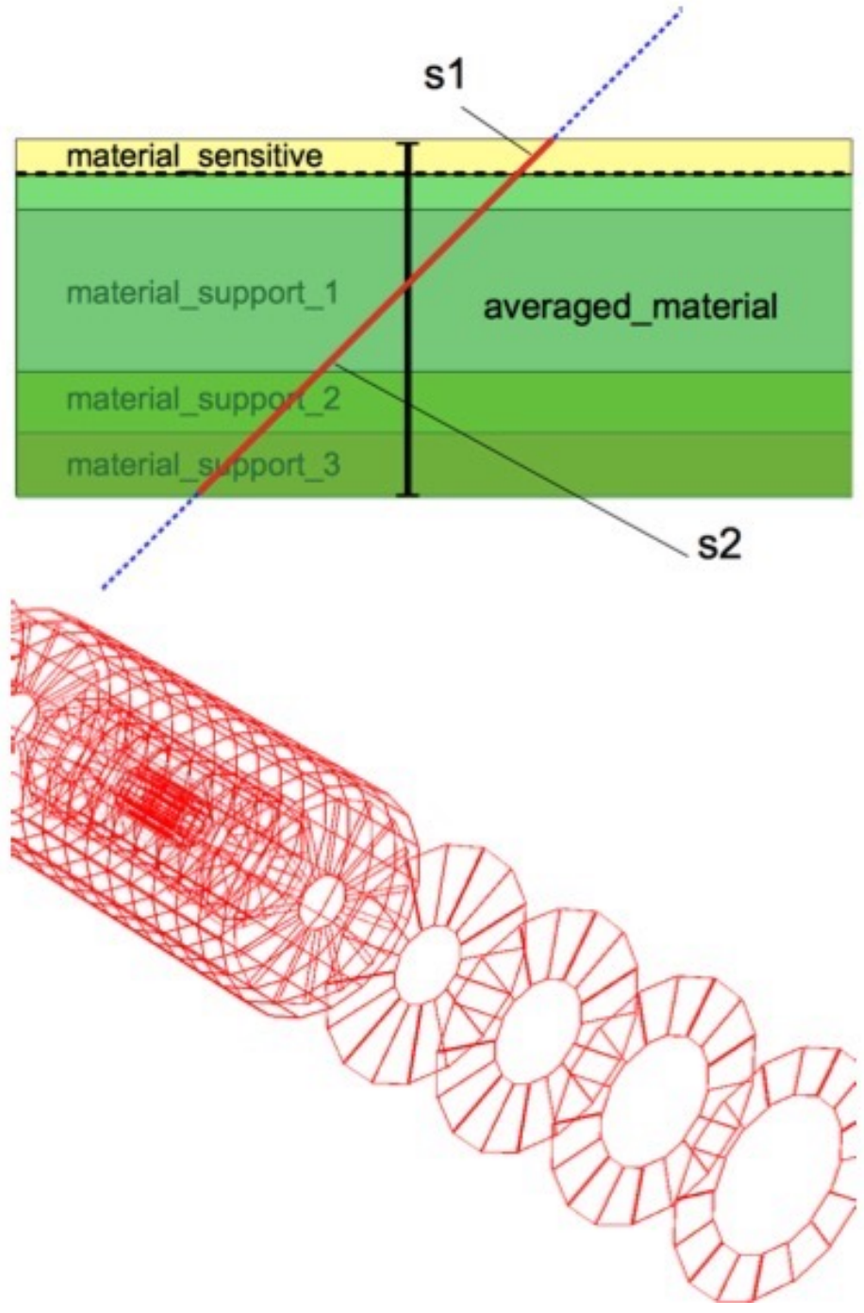
DDRec - Reconstruction extensions

Data Structure	Detector Type	Example
`ConicalSupportData`	Cones and Tubes	BeamPipe
`FixedPadSizeTPCData`	Cylindrical TPC	TPC
`LayeredCalorimeterData`	Sandwich Calorimeters	ECal, HCal, fwd Calos
`ZPlanarData`	Planar Silicon Trackers	VXD, SIT, SET
`ZDiskPetalsData`	Forward Silicon Trackers	FTD

- holding a summary of more abstract information useful for reconstruction
  - Populate during driver construction
    - symmetry,extent,layer thicknesses,... (info. available in driver)
    - #X0, #λI,... (Use material map)
- more simple data structures available,
- but users can even attach their own more complicated objects
- could have dynamic calculation at call time

DDRec - Surfaces for tracking

- Special type of extension, used primarily in tracking
  - Attached to DetElements and Volumes (defining their boundaries) and provide:
    - u,v,normal and origin vectors
    - inner and outer thicknesses and material - automatically averaged from detailed model
    - global to local and local to global coordinate transforms (u,v)<->(x,y,z)



Details in Frank talk “Track Reconstruction”

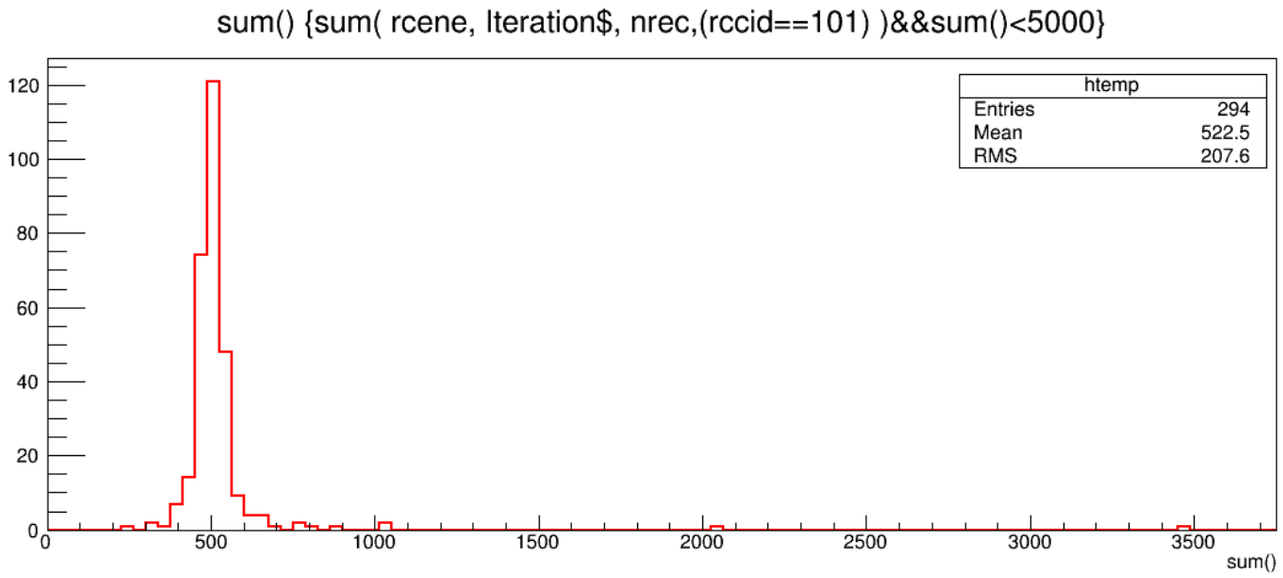
DDMarlinPandora - a pandora client

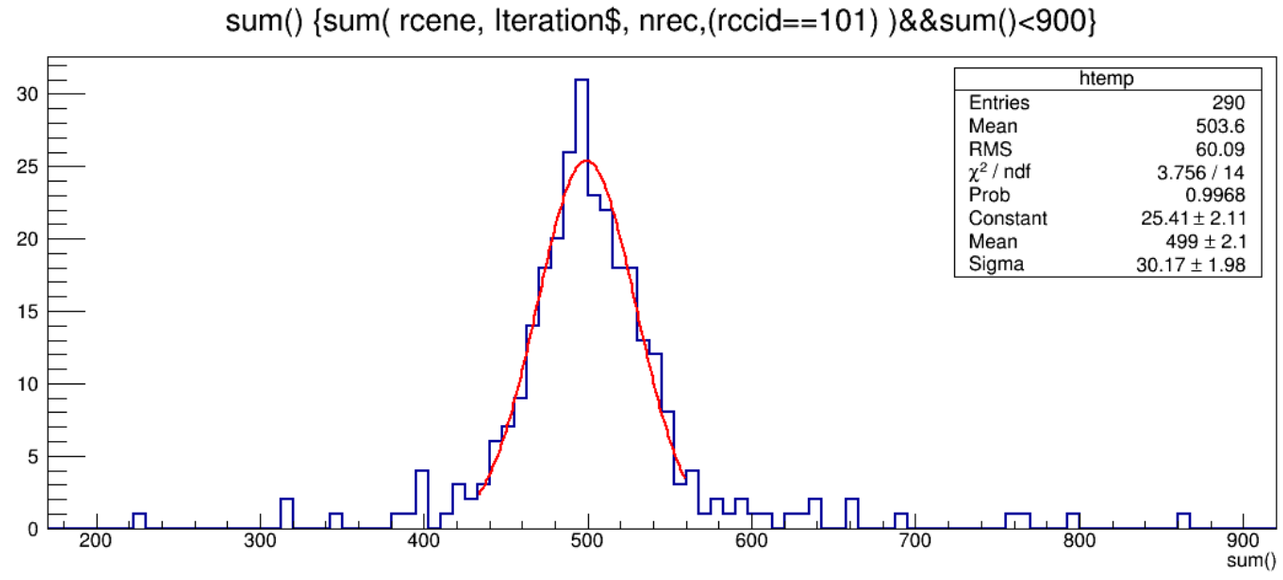
- Pandora consists of the PandoraSDK (Software Development Kit)
  - comprises a single library and header files with flexible APIs.
- Pandora algorithms declared to be suitably generic
  - designed for use by multiple client applications.
- Algorithms are always registered via the client application, using the PandoraAPIs,
  - the client application can link against as many content libraries as required.
- DDMarlinPandora:
  - a Pandora client application for ILD/CLICdp
  - link against DD4hep, create geometry, track, calohit, …

Details in Nikiforos talk “DD4hep-based reconstruction”

Full Reconstruction

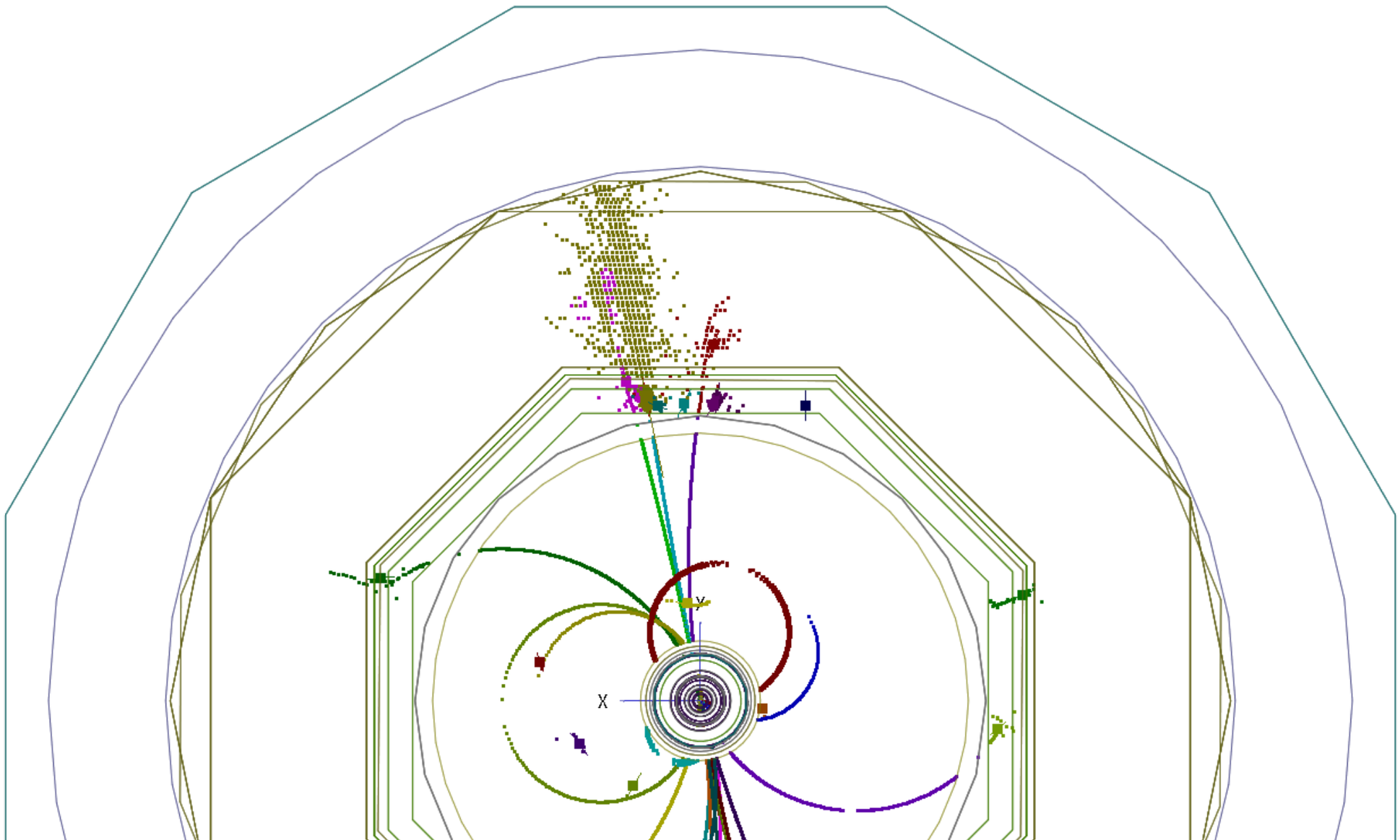
- A new reconstruction with DD4hep/DDRec and DDMarlinPandora framework has been initialised.
- The majority of reconstructed energy is around 500GeV (ucam 500GeV stdhep samples)
- The validation is ongoing





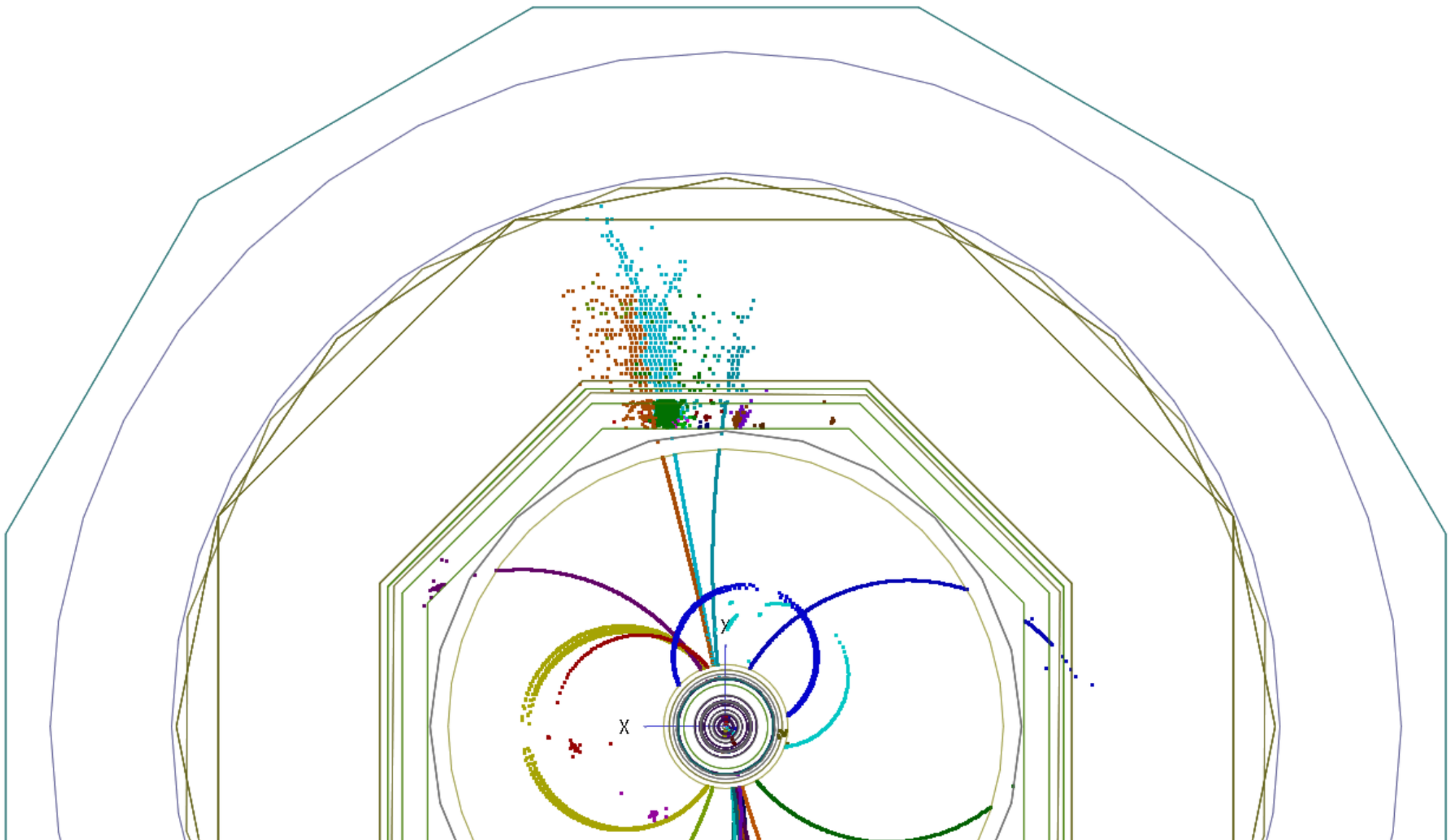
Known Issue

- Few event has large recounted energy, more than 500GeV for some neutral particles, (gamma, neutron,…)



DBD reference

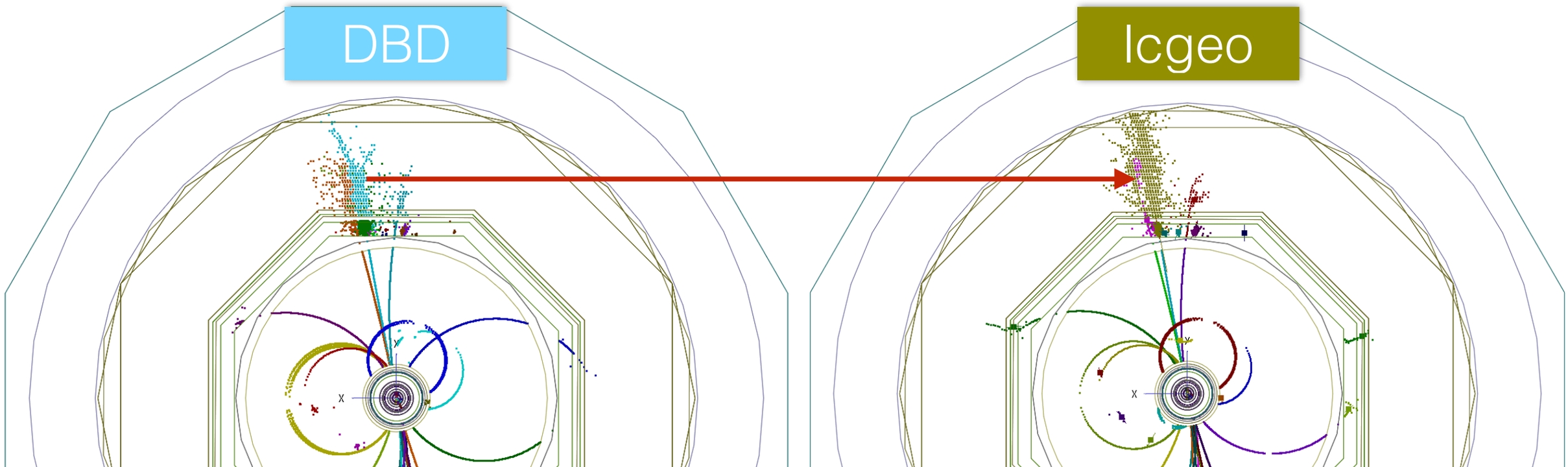
- The same event, this issue has not been seen in Mokka with ILD standard reconstruction.



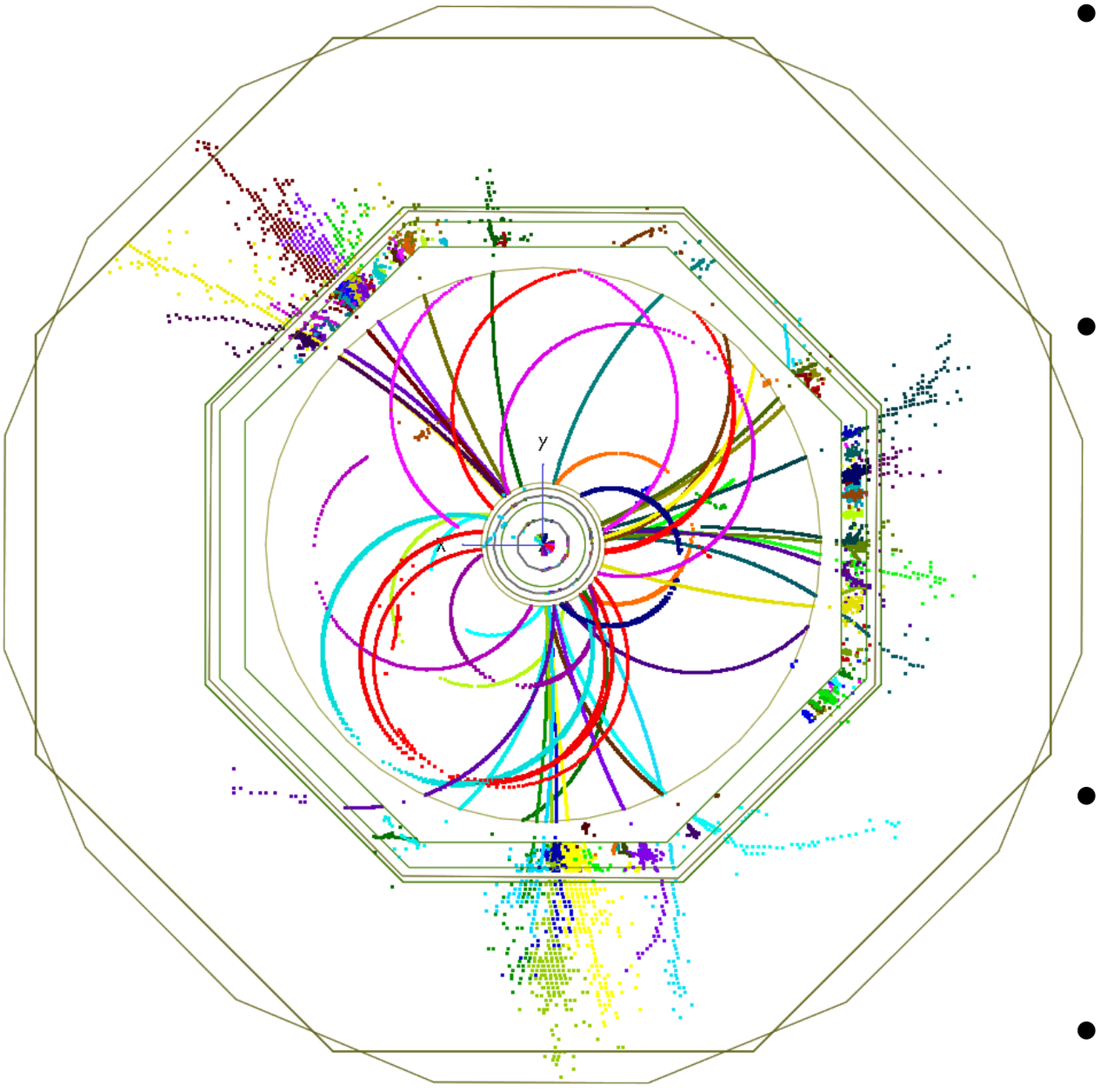
Know Issue

- And clearly, wrong split cluster in calorimeter, and connection to tracker could be seen for this event.

- This is a neutron, the investigation is ongoing.



Reconstruction - all together



RECO: DDRec + DDMarlinPandora

- can run existing standard reconstruction using DDRec interface to create GEAR file.
- DDMarlinPandora: (ongoing)
  - use the DDRec and lcgeo as input, and provide the output for PandoraPFANew
- can gradually move from GEAR to new DDRec classes
- after the stdreco, use High Level Reconstruction Tools

Summary

- ☑ The lcgeo is a new DD4hep based detector geometry description and simulation package has a first complete simulation model ILD_o1_v05.

- ☑ From the hits map, which is the sensitive layer centre, and they are on top of each other for both Mokka and lcgeo. which have validated the geometry .

- ☑ ddsim simulation use DDG4 the DD4hep build-in Geant4 gateway. pass ucam 500GeV,10,000 events statistic test run.

- 📌 DDMarlinPandora uses the DDRec and lcgeo as input, and provide the output for PandoraPFANew. The reconstruction validation and improvement is ongoing.