Software for future collider studies: ILD and beyond

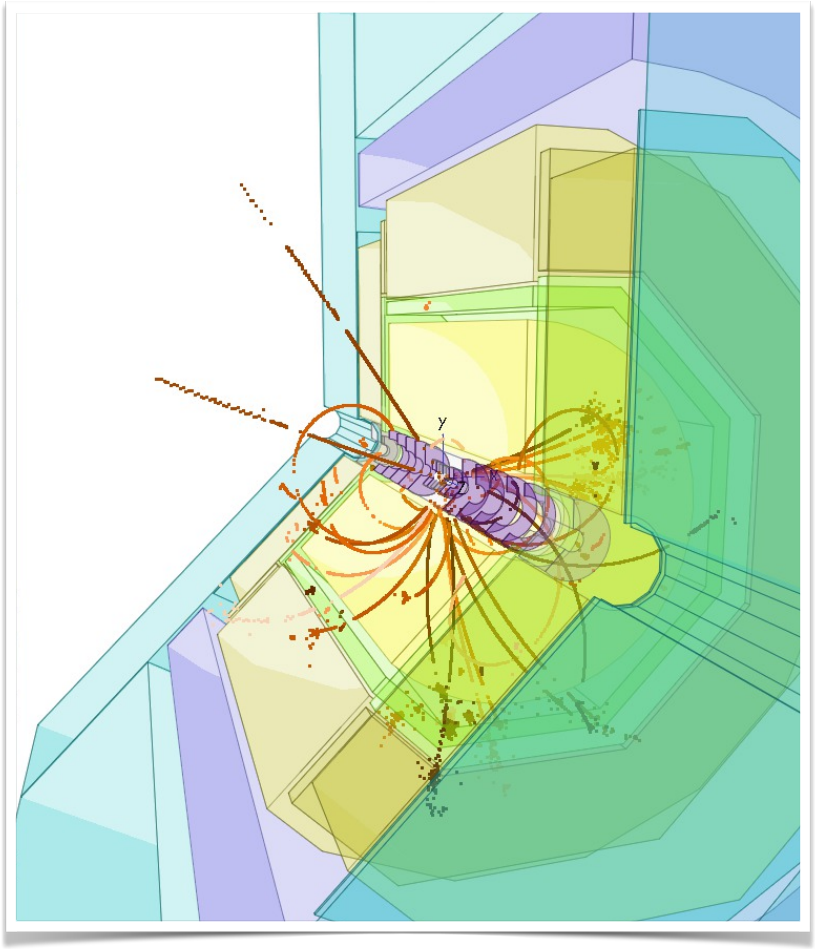
ILD Meeting connected to ECFA 2024

Frank Gaede, DESY
LPHNE, Paris
Oct 8, 2024

HELMHOLTZ RESEARCH FOR GRAND CHALLENGES

Outline

- Introduction and Reminder: Key4hep
- DD4hep detector models and reconstruction
- Standard ILD reconstruction algorithms and “Transition” to to Key4hep
- **AI/ML in Key4hep**:
  - fast simulation and
  - recent HLR algorithms w/ ML
- Summary and Outlook

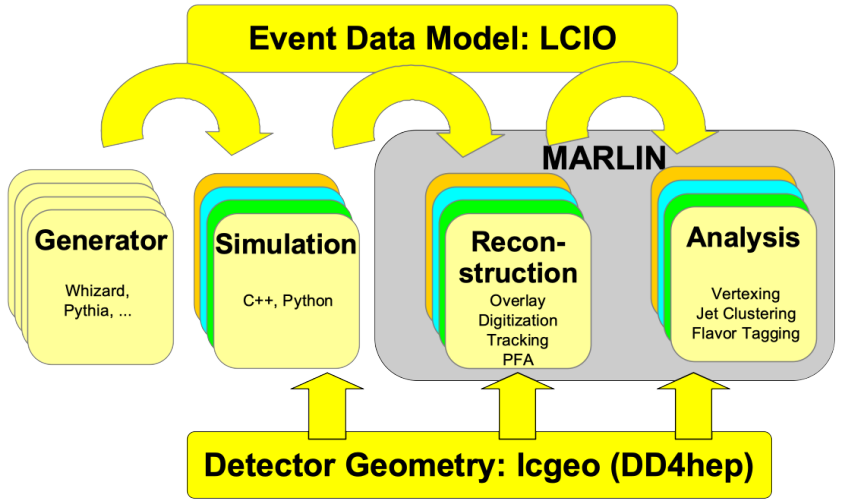


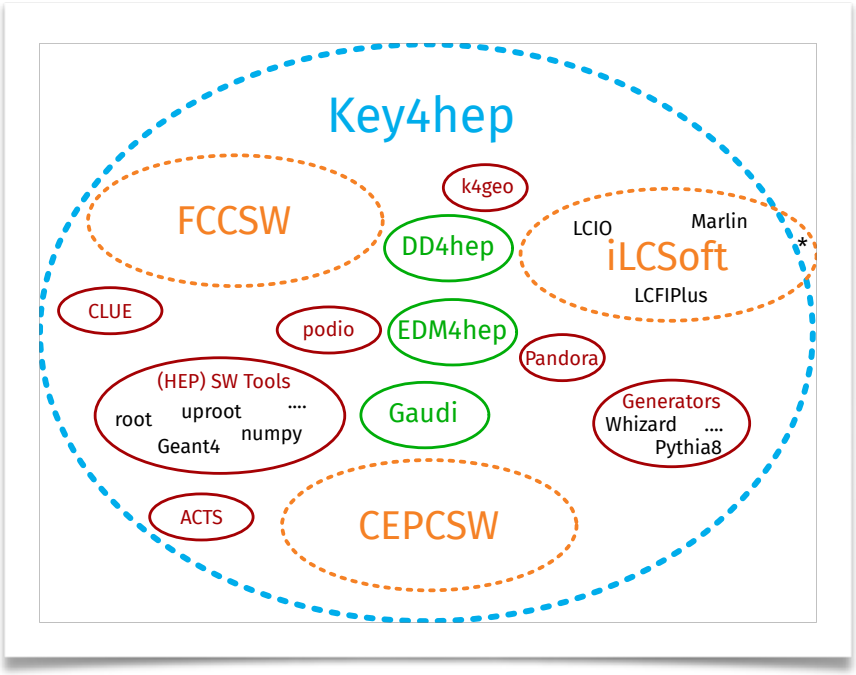
The common software vision

iLCSoft as integral part of Key4hep

- complete set of tools for
  - **generation, simulation, reconstruction, analysis**
  - build, package, test, deploy

- core ingredients of current **Key4hep**
  - **PODIO** for **EDM4hep** (based on LCIO and FCC-edm)
  - **Gaudi** framework, devel/used for (HL-)LHC
  - **DD4hep** for geometry
    - originally developed for LC now adopted by community
  - **spack** package manager



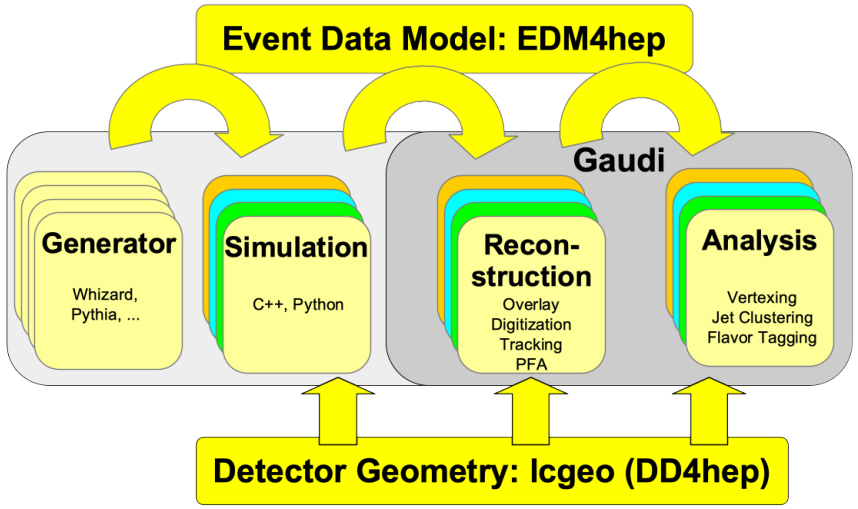


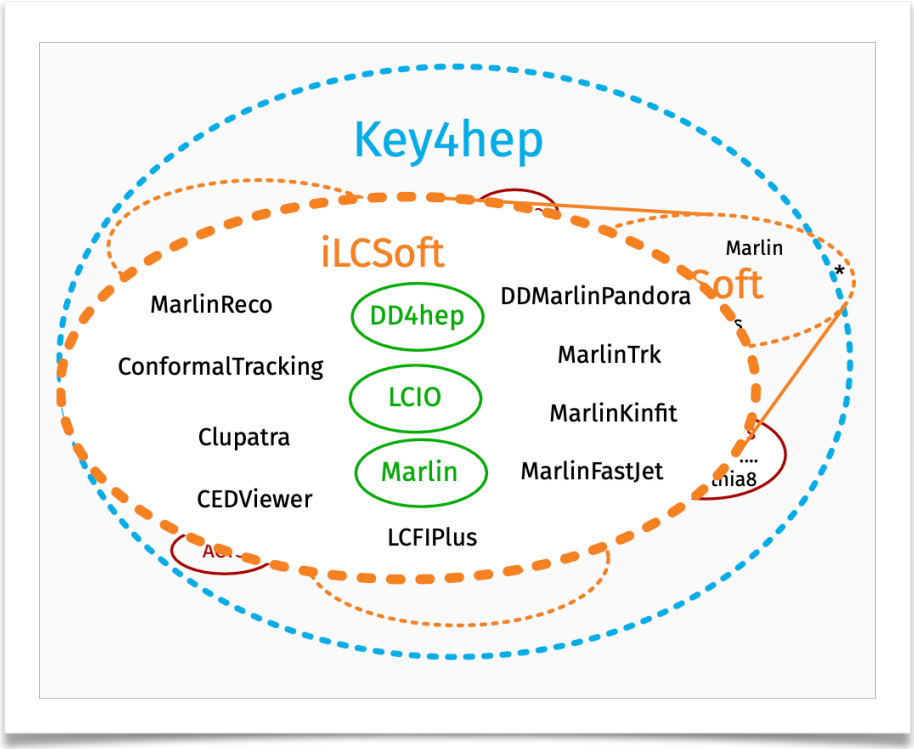
The common software vision

iLCSoft as integral part of Key4hep

- complete set of tools for
  - **generation, simulation, reconstruction, analysis**
  - build, package, test, deploy

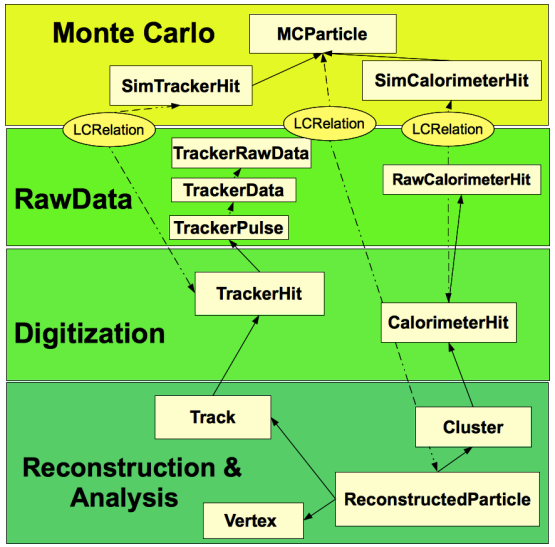
- core ingredients of current **Key4hep**
  - **PODIO** for **EDM4hep** (based on LCIO and FCC-edm)
  - **Gaudi** framework, devel/used for (HL-)LHC
  - **DD4hep** for geometry
    - originally developed for LC now adopted by community
  - **spack** package manager

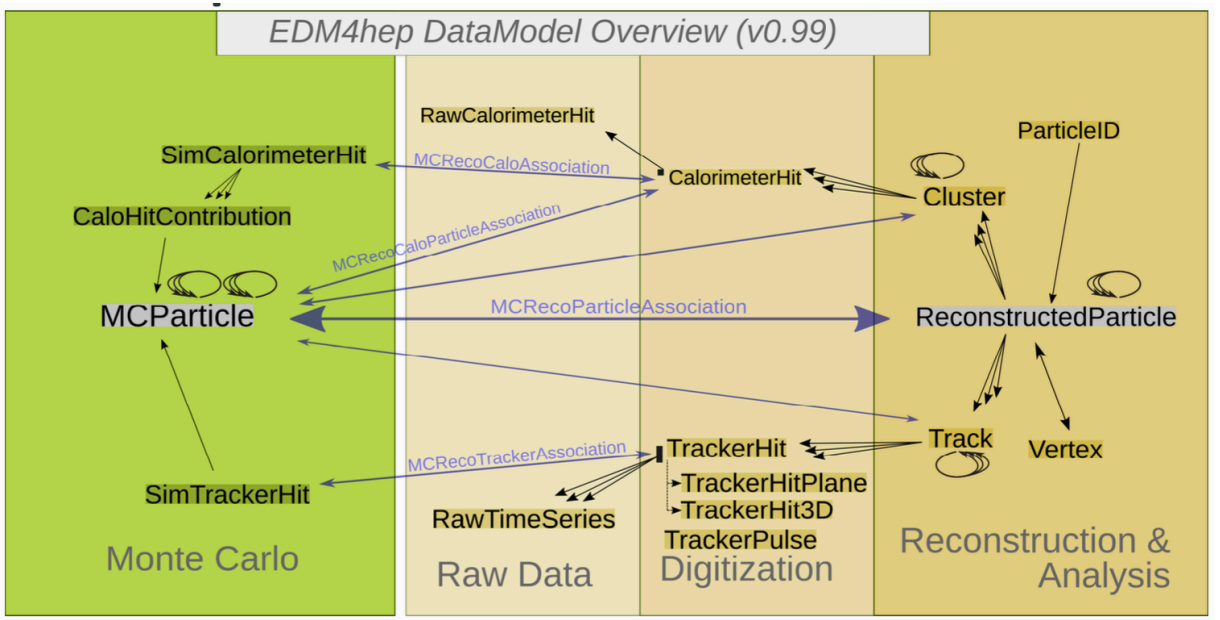




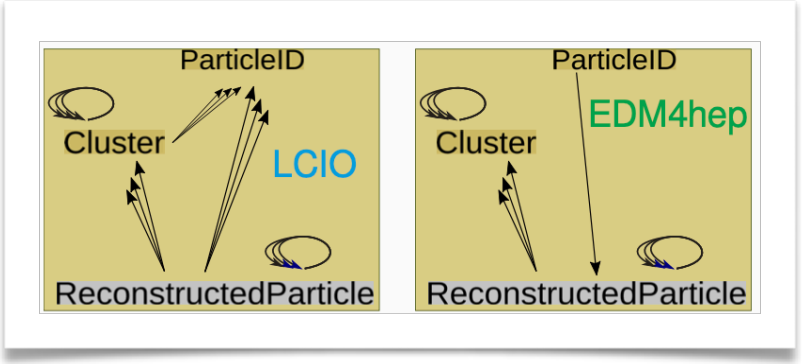
EDM4hep - Event Data Model in Key4hep

the designated successor of LCIO



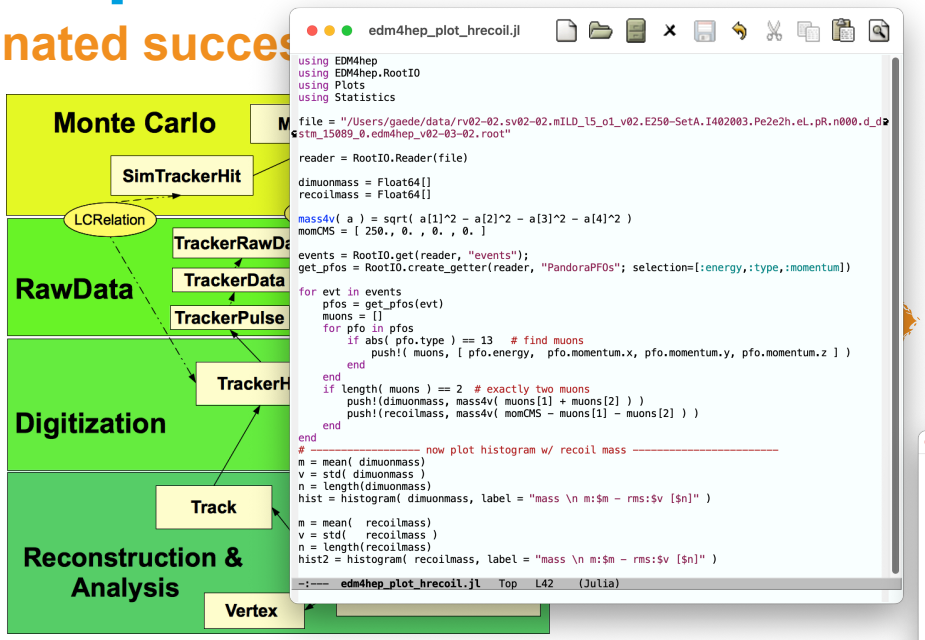


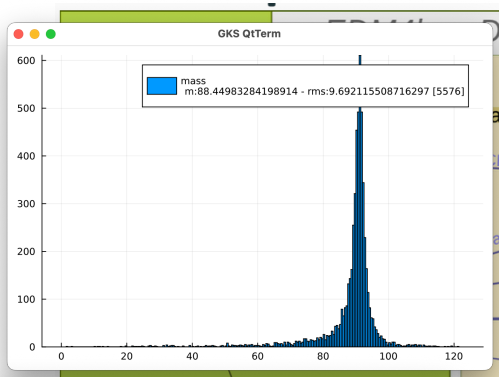
- EDM4hep to large extend reproducing (battle-proven) LCIO EDM - yet with
  - more strict **mutability** concept (**thread safety**)
  - more **consistent** linking to *lower level* data objects
  - more performant reading (ROOT files)
- ***k4EDM4hep2LcioConv*** provides consistent conversion between the two
  - **production version** v01-00 to be released very soon

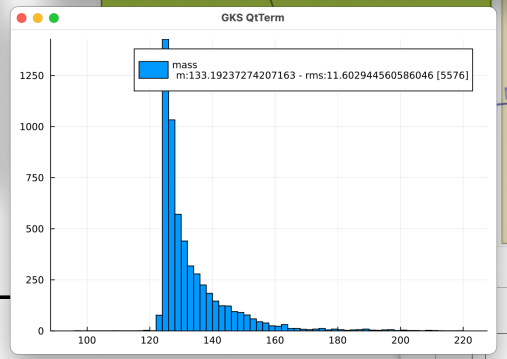


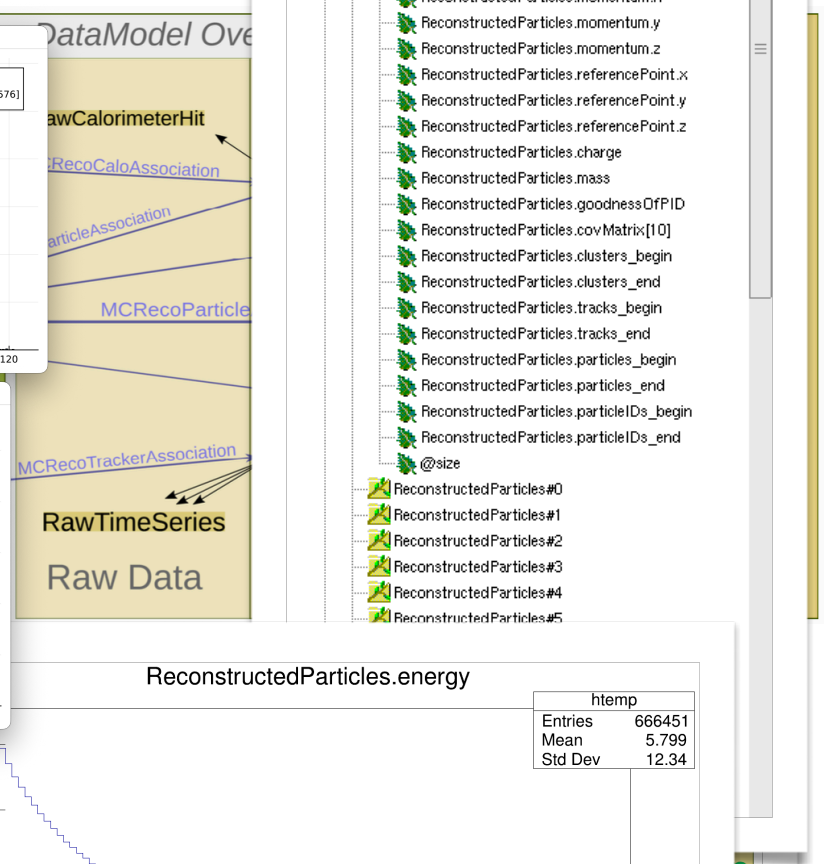
EDM4hep - Event Data Model in Key4hep

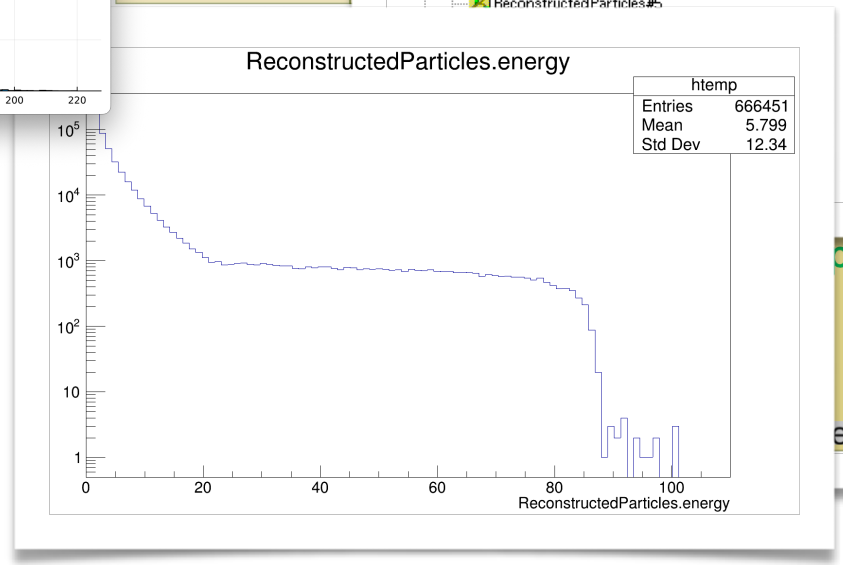
the designated succes











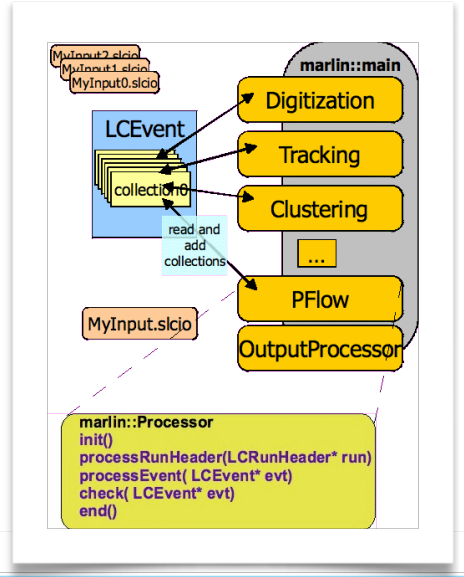
- EDM4hep to large extend reproducing (battle-proven) L
  - more strict **mutability** concept (**thread safety**)
- the two

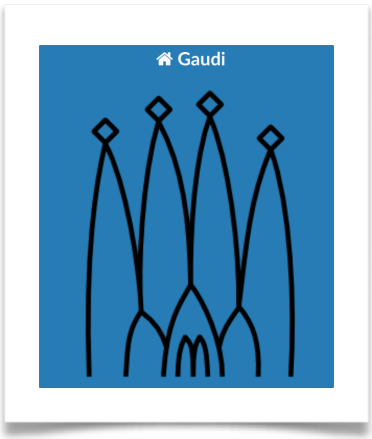
using **ROOT** files w/ columnar storage (TTree or RNTuple) - (optionally SIO files)
fast analysis w/ **C++** or **Python** using EDM4hep
RDataFrame + EDM4hep (FCCAnalysis)
Python or **Julia** reading ROOT files directly

Gaudi - application framework in Key4hep

designated successor of Marlin

- C++ application framework for HEP
- developed at CERN
- used in production for
  - LHCb and ATLAS (*battle-proven*)
  - FCC-SW and smaller experiments
  - and now in Key4HEP
- highly configurable
  - EDM, workflows (algorithms)
- allows parallelisation through multi-threading
- integration of heterogeneous resources
  - CPUs, GPUs, FPGAs,…





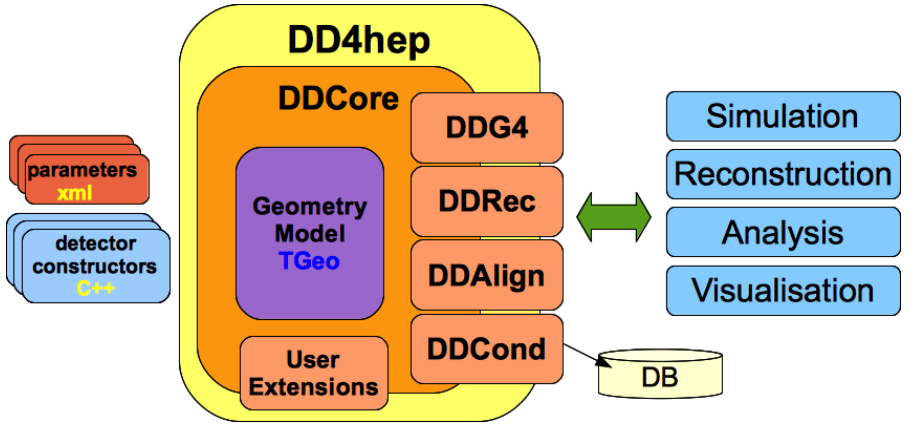
	Marlin	Gaudi
language	c++	c++
working unit	Processor	Algorithm
config language	XML	Python
transient data format	LCIO	anything
set up function	`init`	`initialize`
work function	`processEvent`	`execute`
wrap up function	`end`	`finalize`

GAUDI similar to MARLIN framework
yet more powerful and larger user basis

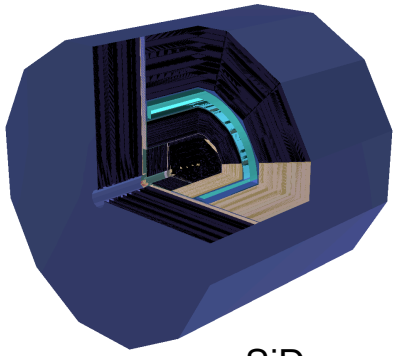
DD4hep geometry toolkit

defining the detector geometry and different views on it

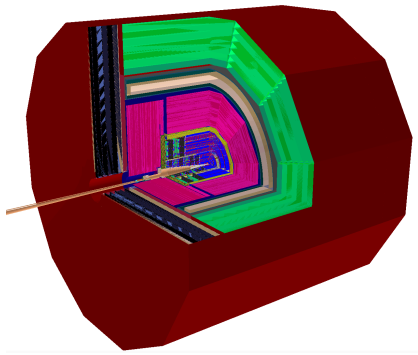
- supporting the full life cycle of the experiment
- **single source** of information for full **simulation, reconstruction, conditions, alignment, visualisation and analysis**
  - used by CEPC, CLIC, CMS, EIC, FCC, ILC, LHCb, …



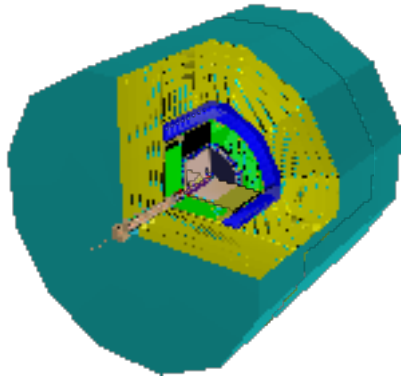
DD4hep: de facto industry standard



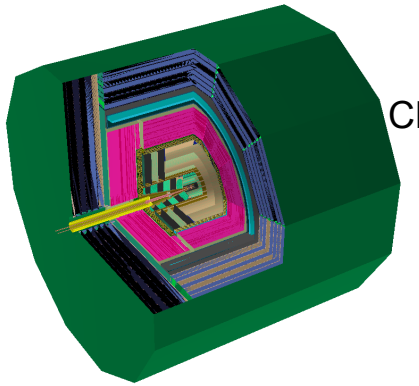
SiD



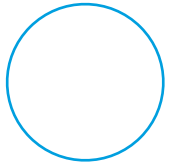
CLICdet

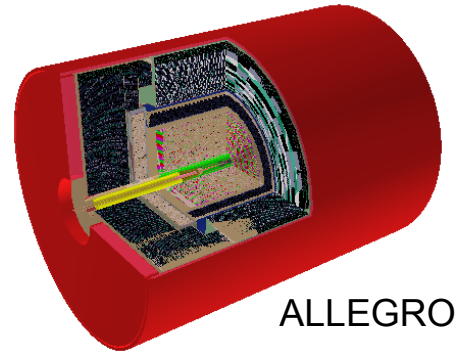


ILD

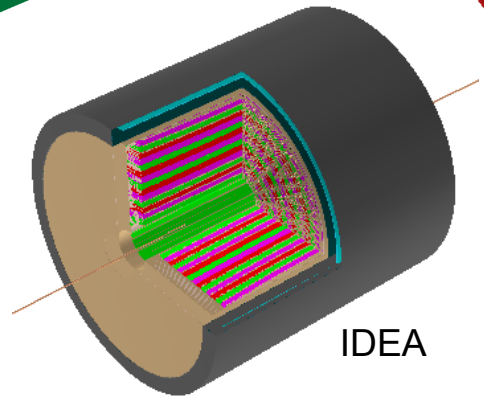


CLD





ALLEGRO



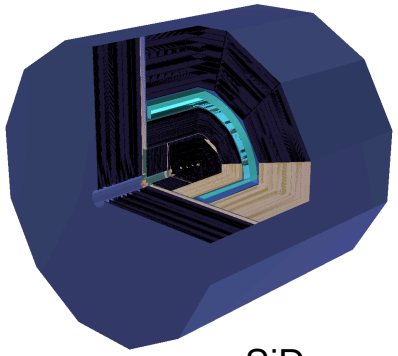
IDEA

all future Higgs factory detector simulation models in **one package**
https://github.com/key4hep/k4geo
=> allows to **re-use reconstruction code** (for similar sub detectors) across detector concepts

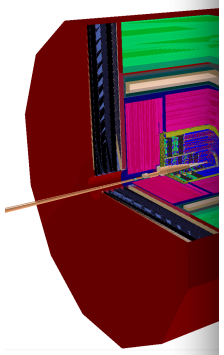
DD4hep geometry toolkit

defining the detector geometry and different views on it

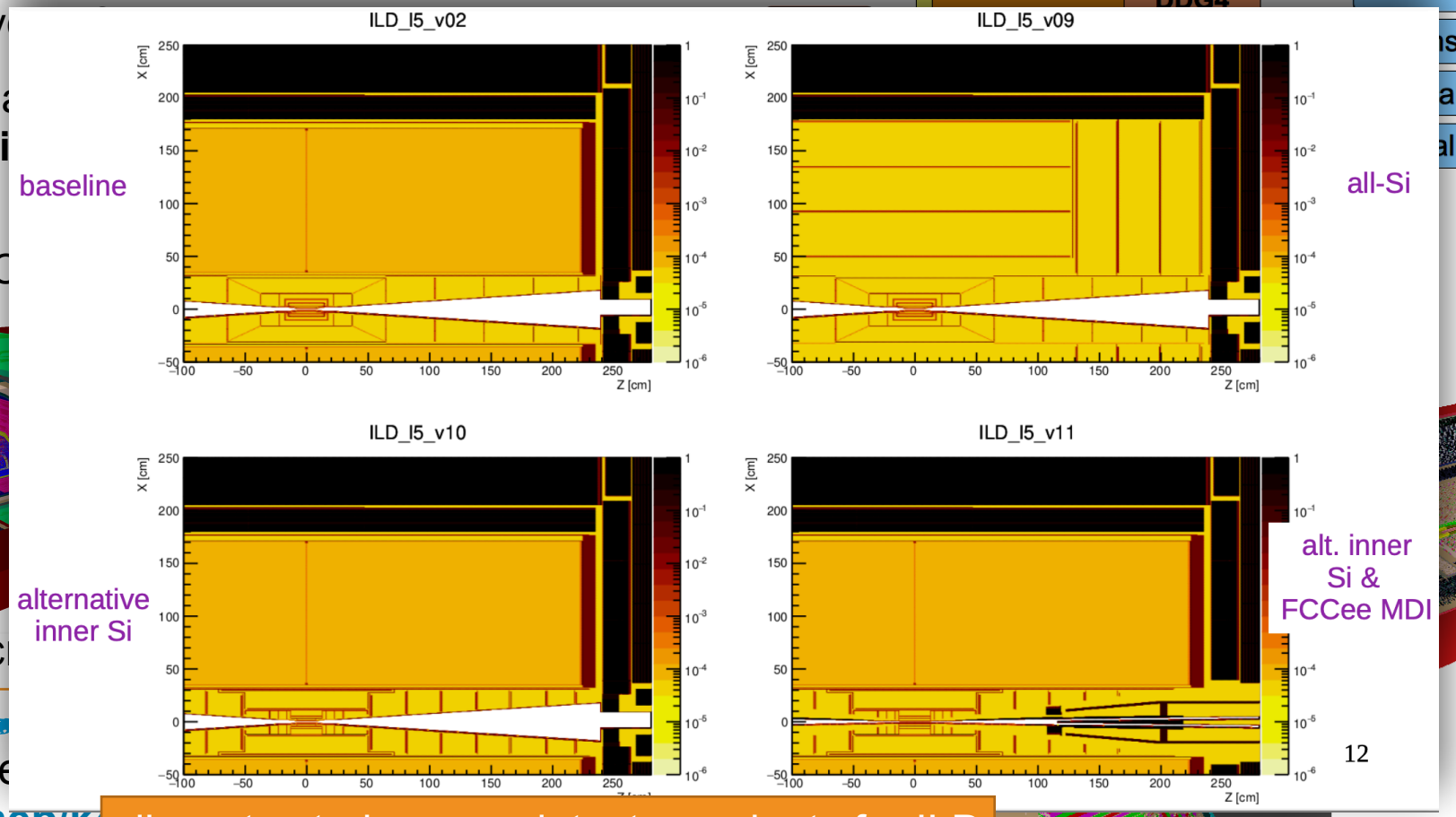
- supporting the full life cy
- **single source** of informa
  reconstruction, conditi
  analysis
  - used by CEPC, CLIC

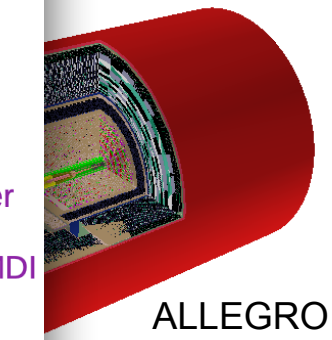


SiD



C

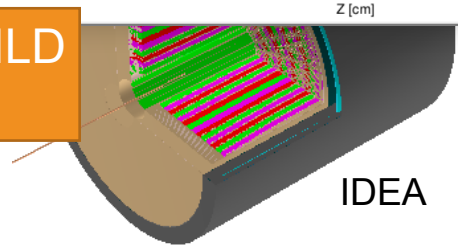




ALLEGRO

all future Higgs factory dete
https://github.com/key4hep/k
=> allows to **re-use reconstru**
across detector concepts

allows to study many detector variants for ILD
see: talk by D.Jeans



IDEA

large reconstruction code base in iLCSoft

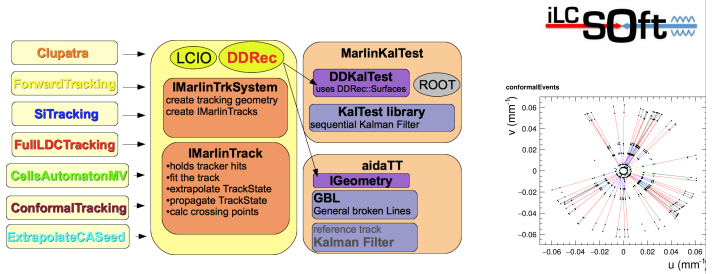
Developed over >15 years for linear collider detectors - e.g. ILD

- realistic detector models for incl. tracking/reconstruction geometry
- track reconstruction
  - generic API for fitting algorithms
  - large number of pattern recognition algorithms

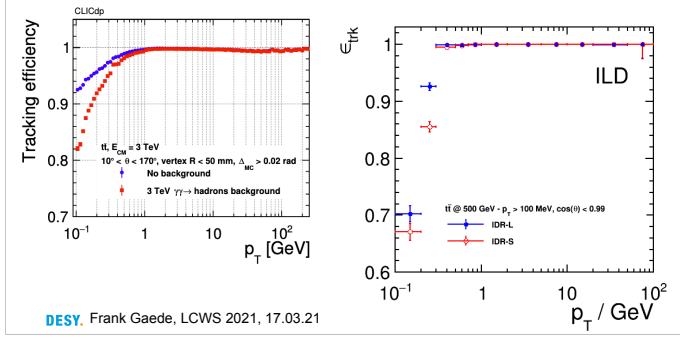
Tracking in iLCSoft

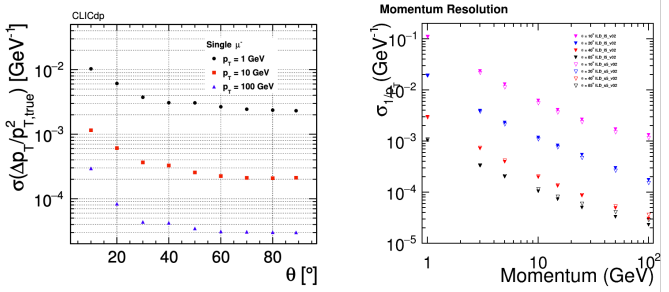
pattern recognition and Kalman-Filter

- generic tracking API MarlinTrk based on DDRec material surfaces
- many pattern recognition algorithms exist, e.g.
- **ConformalTracking**:
  - generic algorithm that works for all Si-Trackers
  - used by CLICdet and SiD (also works for ILD inner)



achieve excellent tracking efficiencies and resolution w/ realistic tracking codes





large reconstruction code base in iLCSoft

Developed over >15 years for linear collider detectors - e.g. ILD

- realistic detector models for incl. tracking/reconstruction geometry
- track reconstruction
  - generic API for fitting algorithms
  - large number of pattern recognition algorithms
- particle flow algorithms
  - PandoraPFA and Arbor, AprilPFA

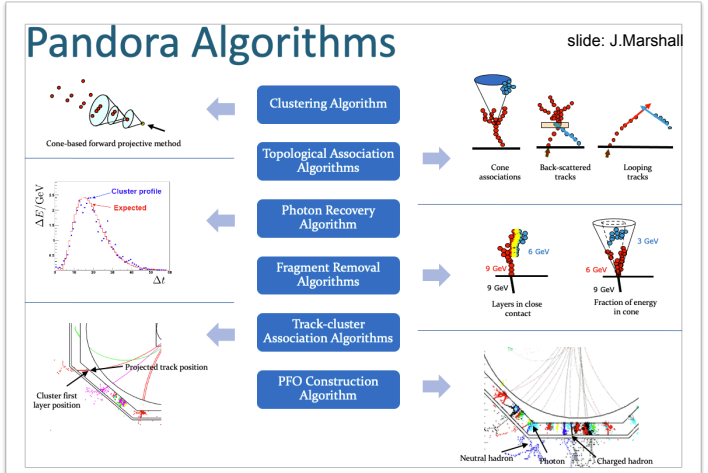
Tracking in iLCSoft

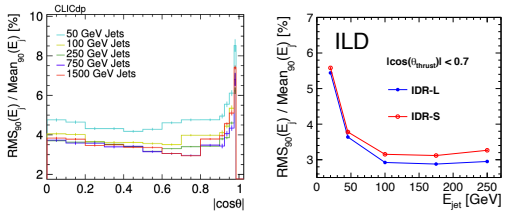
pattern recognition and Kalman-Filter

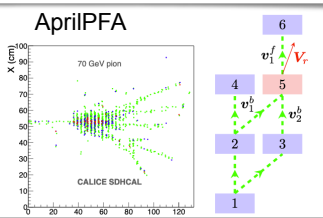
Particle Flow Algorithms

highly granular calorimeter reconstruction

- all current detector concepts for LC are based on highly granular calorimeters
  - optimised for the Particle Flow Algorithm
- **PandoraPFA** is the de **facto standard** used by ILD, SiD and CLICdp
- alternative PFA algorithms exist and provide possibility to cross check
  - Arbor (CEPC), April (SDHCal prototype)







DESY. Frank Gaede, LCWS 2021, 17.03.21

7

large reconstruction code base in iLCSoft

Developed over >15 years for linear collider detectors - e.g. ILD

- realistic detector models for incl. tracking/reconstruction geometry
- track reconstruction
  - generic API for fitting algorithms
  - large number of pattern recognition algorithms
- particle flow algorithms
  - PandoraPFA and Arbor, AprilPFA
- high level reconstruction
  - jet finding, flavor tagging, PID, TOF,…

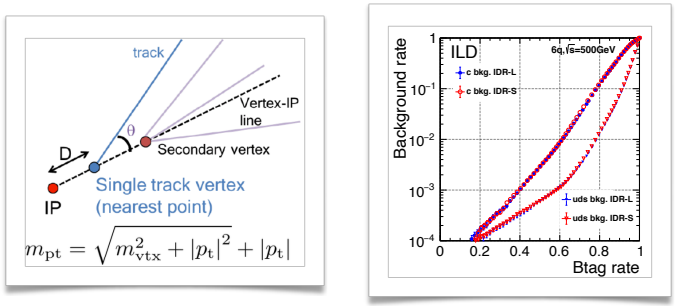
Tracking in iLCSoft

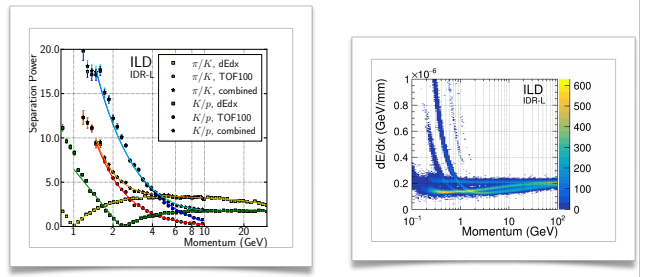
pattern recognition and Kalman-Filter

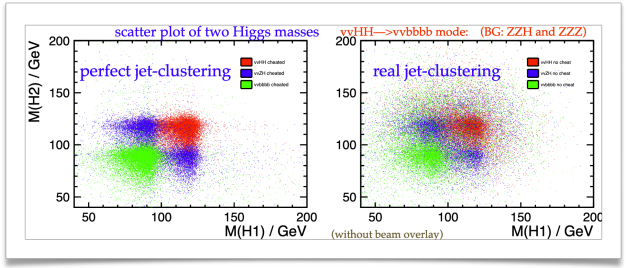
Particle Flow Algorithms

High Level Reconstruction

analysing the Particle Flow Objects







$\delta\lambda_{HHH}$ improves by 40% w/ perfect jet clustering

- **High-Level reconstruction** algorithms are crucial to achieve the ultimate physics reach of detectors
- vertex finding and flavor tagging: **LCFIPlus**
- PID tools: dE/dx, TOF, shower shapes,…
- Jet clustering: Durham, Valencia, …

- very active field of development
  - already good set of tools available
- further improvement in HLR tools often directly impacts the final physics performance

DESY. Frank Gaede, LCWS 2021, 17.03.21

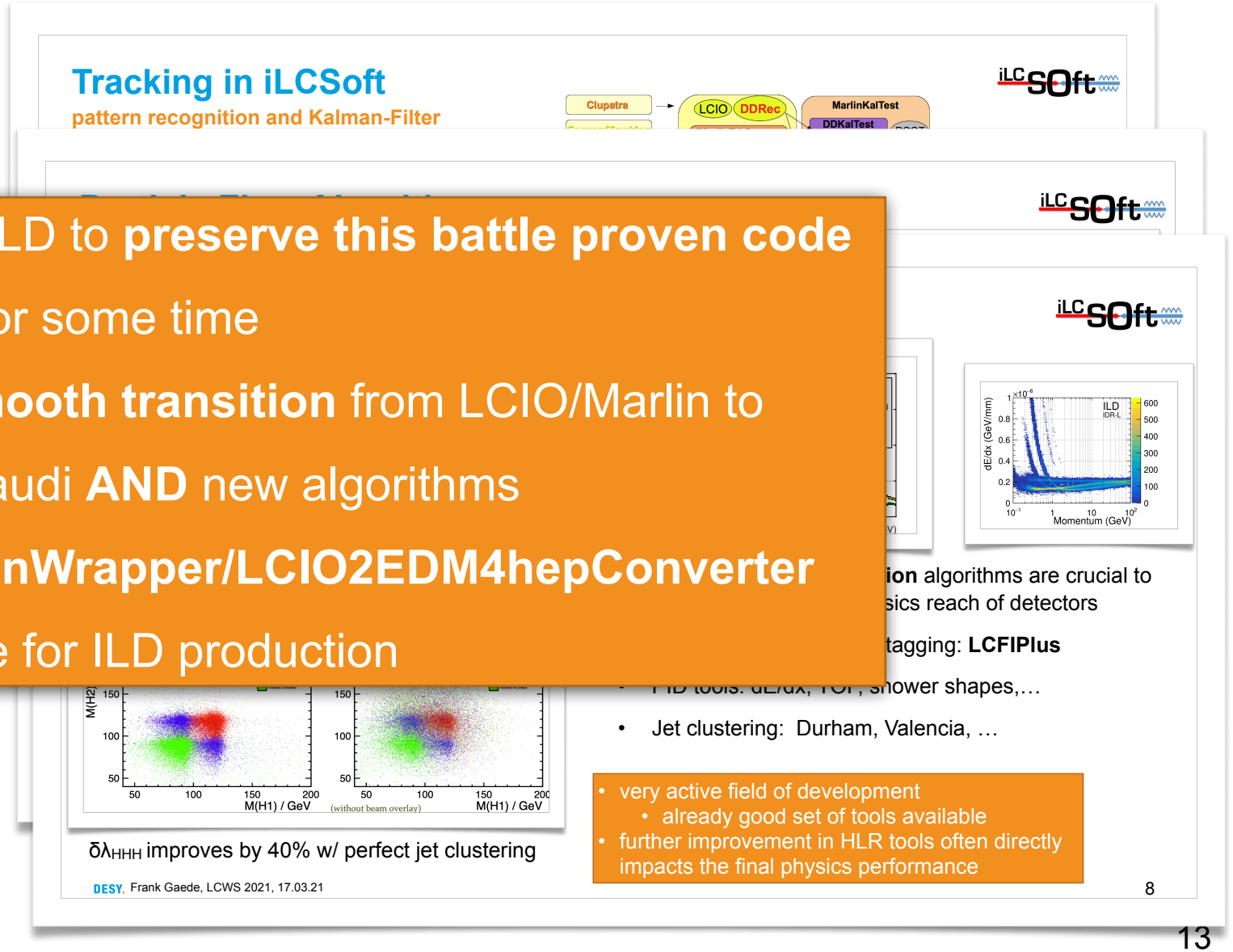
8

large reconstruction code base in iLCSoft

Developed over >15 years for linear collider detectors - e.g. ILD

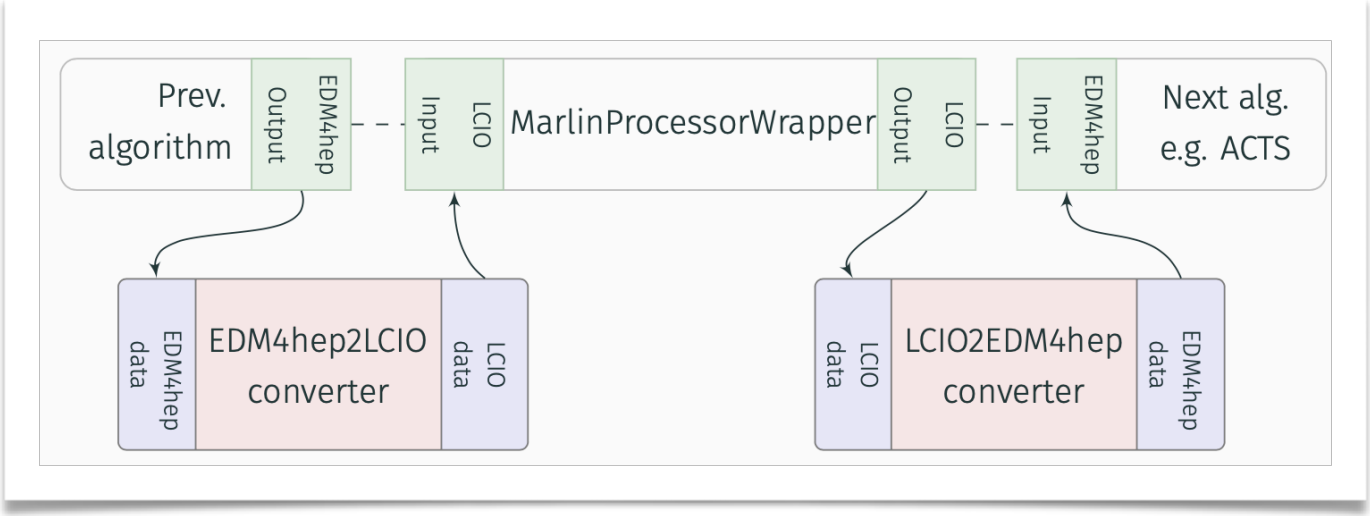
- realistic detector models for incl. tracking/reconstruction geometry
- track reconstruction
  - generic API for f
  - large number of recognition algo
- particle flow algorith
  - PandoraPFA ar
- high level reconstru
  - jet finding, flavor tagging, PID, TOF,…

- it is **vital** for ILD to **preserve this battle proven code** in Key4hep for some time
- -> need a **smooth transition** from LCIO/Marlin to EDM4hep/Gaudi **AND** new algorithms
- will use **MarlinWrapper/LCIO2EDM4hepConverter** for some time for ILD production



K4MarlinWrappper

the vision: mix and match Marlin and Gaudi algorithms



C.Hensel, ILD Meeting 2024

- in a transition phase algorithms **developed in the new EDM4hep/Gaudi** world can gradually replace older algorithms
  - e.g. eventually one might want to replace track fitting with **ACTS** also for LC detectors
- some technicalities are address *under-the-hood* via ***k4EDM4hep2LcioConv***

all existing (high level) reconstruction algorithms as Marlin processors fully available in Key4hep !

ILD Standard Reconstruction

now smoothly runs in in Key4hep

- translation of ***MarlinReco.xml*** to GAUDI python steering file ***ILDReconstruction.py***
- plan to use this as new standard (soon ?)

ILD standard reconstruction in Key4hep

T.Madlener

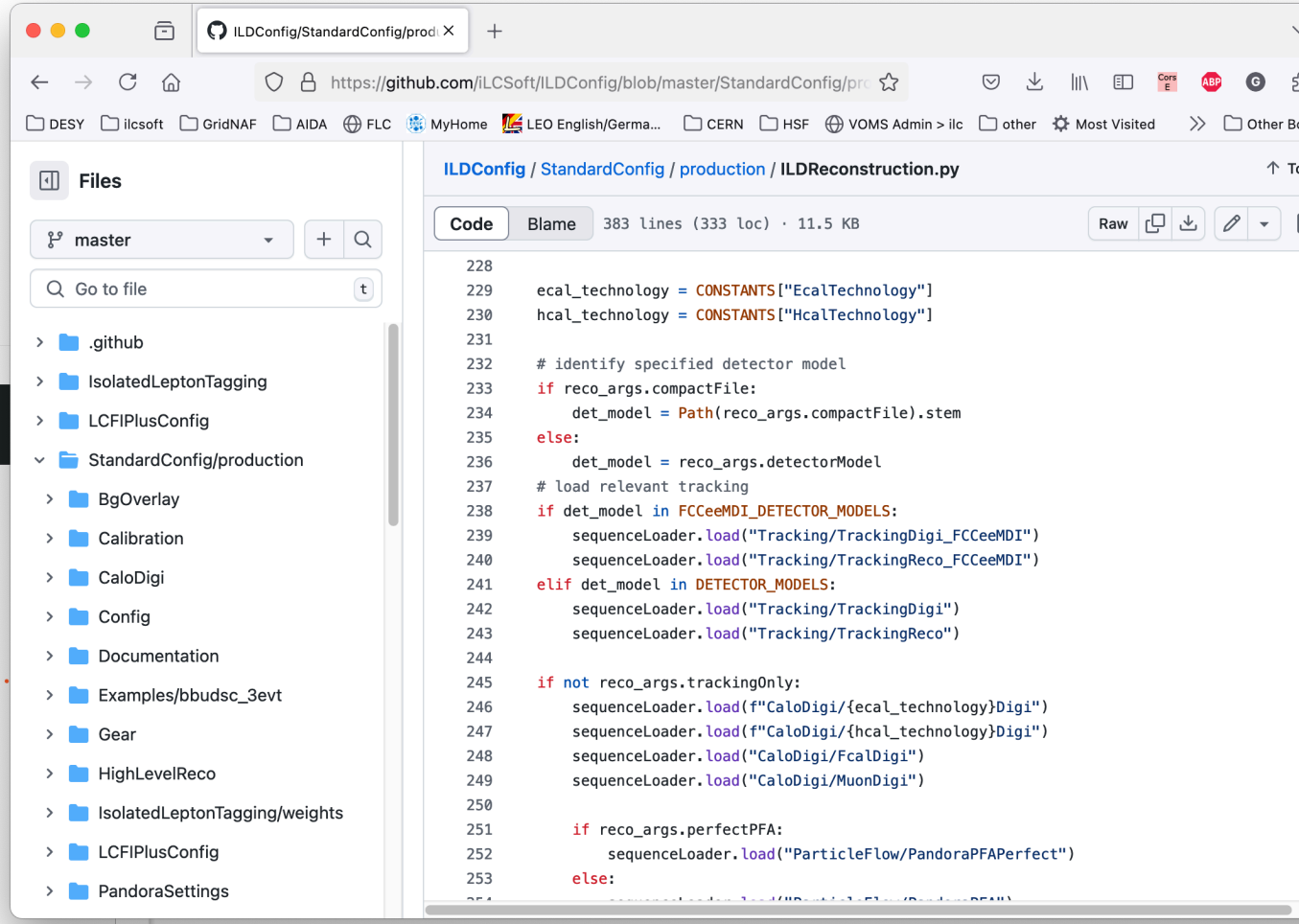
- All configuration available from iLCSoft/ILDConfig
- Everything that works in iLCSoft also works in Key4hep!

```
Marlin MarlinStdReco.xml --global.LCIOInputFiles=<input-file> [.
```

- Now also with Gaudi

```
k4run ILDReconstruction.py --inputFiles=<input-file> [...]
```

  - Works with EDM4hep and LCIO inputs
  - EDM4hep output by default, LCIO output via `--lcioOutput=[true|only]`
- Facilitates collaboration with other projects, e.g. CLD
- Full migration of all workflows will take some time but process started
  - Some new developments already done exclusively in Gaudi configuration

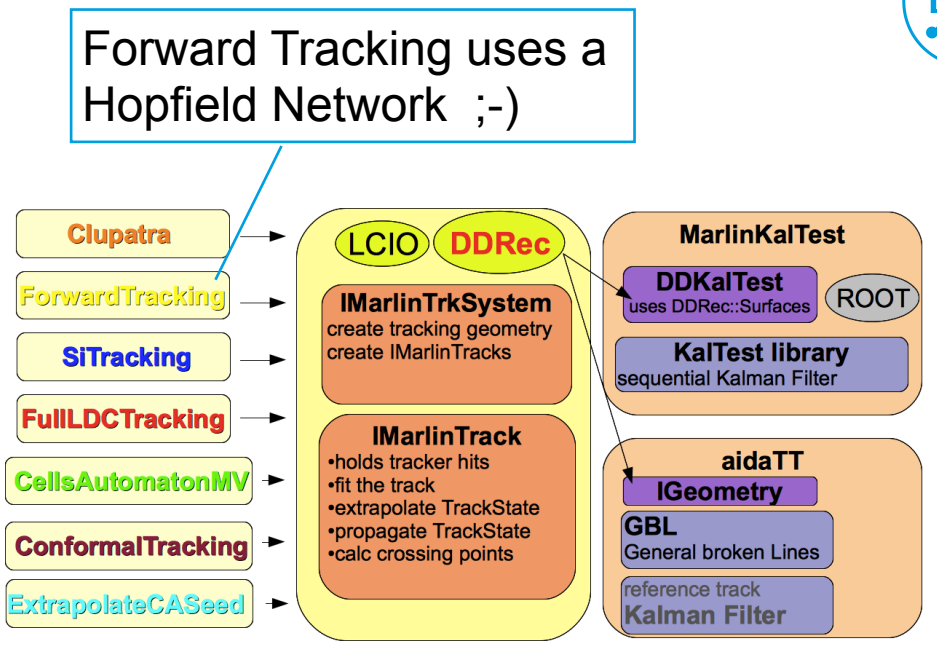


already started to develop new algorithms (or steering variants) in GAUDI only

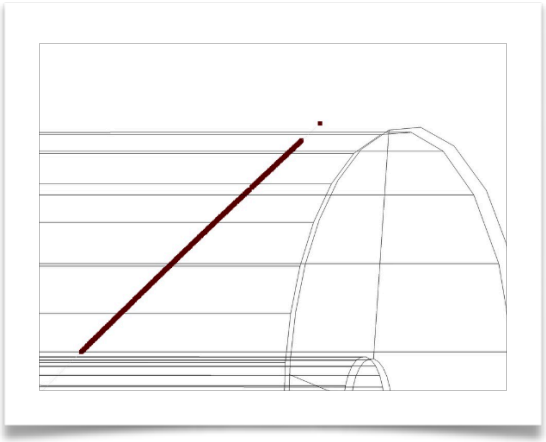
Reconstruction for ILD@FCCee

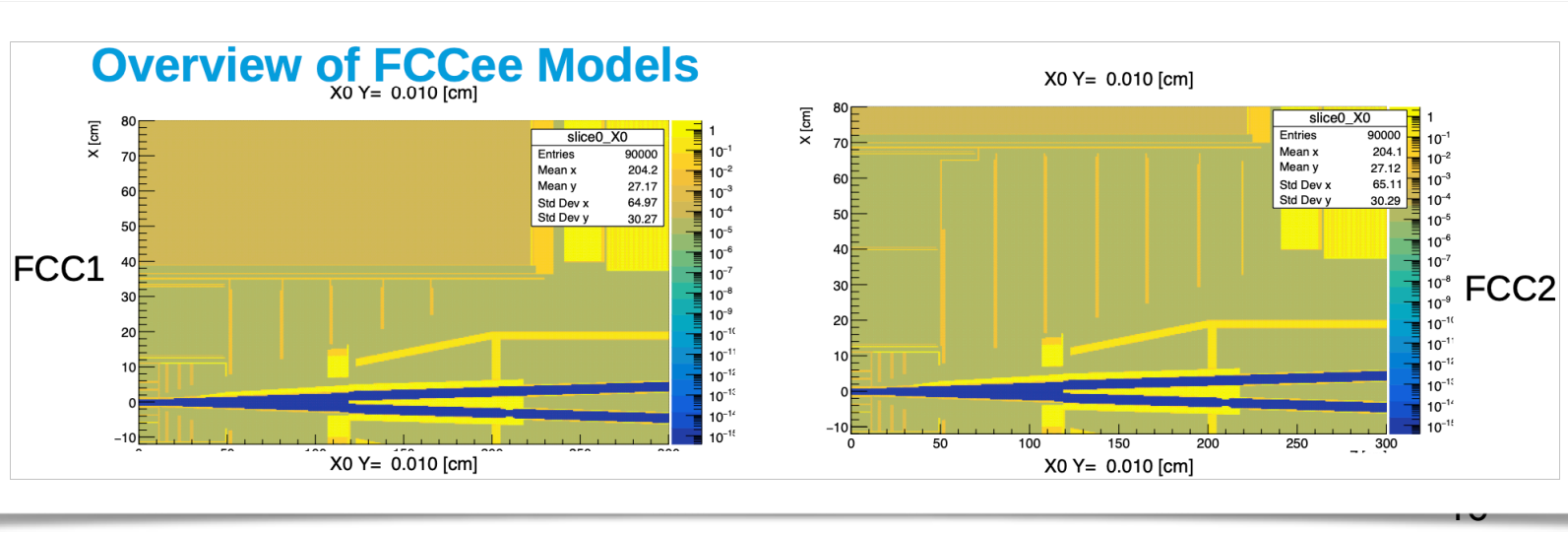
combining simulation models and tracking algorithms

- ILD models for FCCee w/ TPC, FCC-MDI (mask) and CLD inner Si-Tracking
- can run *ConformalTracking* for Si-tracking and *Clupatra* for TPC tracking
- done using **MarlinWrapper** and **ILDReconstruction.py**
  - combining tracks is work in progress…
  - => maybe write this in GAUDI ?



V.Schwan, D.Jeans



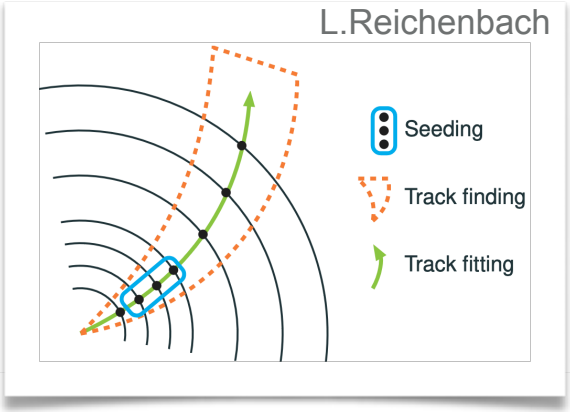


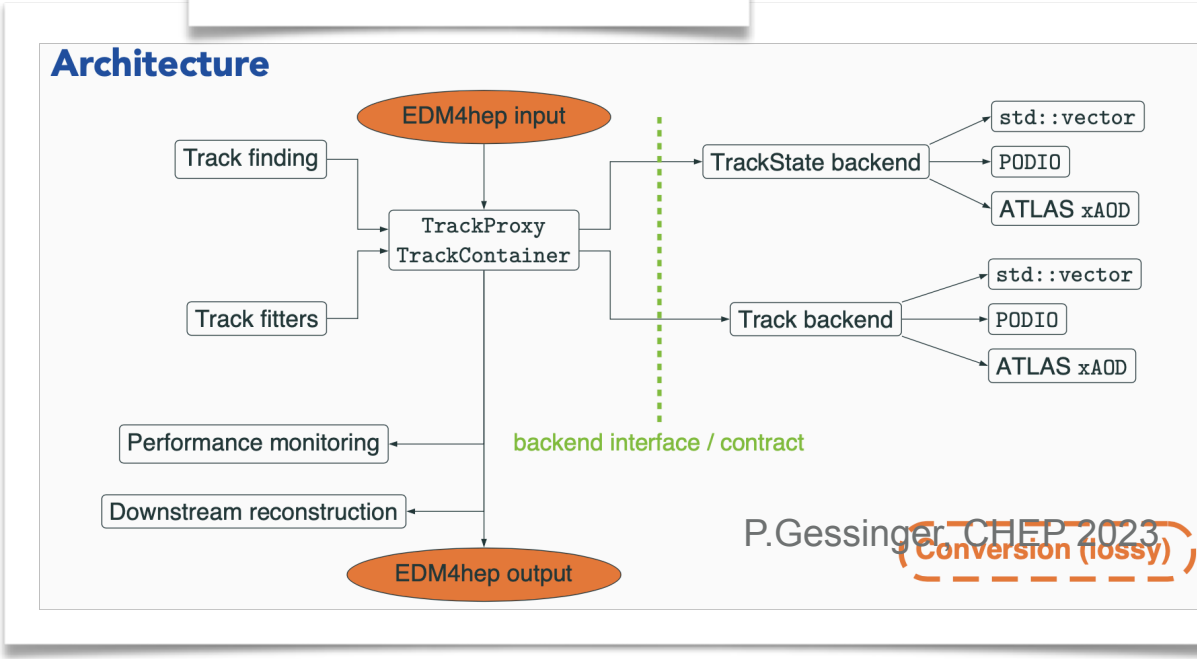
ACTS - A Common Tracking Toolkit

used in Key4hep

- **ACTS** tracking toolkit is the the current choice for track fitting (and finding) in Key4hep
- new package **k4ACTSTracking** provides ACTS fitter
  - implemented as GAUDI algorithm
  - uses DD4hep geometry
- first implementation of TrueTrackfinder for CERN OpenDetector
- successfully used for MuonCollider w/ handcrafted tracking geometry

- ongoing work on **automatic** and **transparent** construction of **tracking geometry** for all detectors in k4geo (Key4hep)



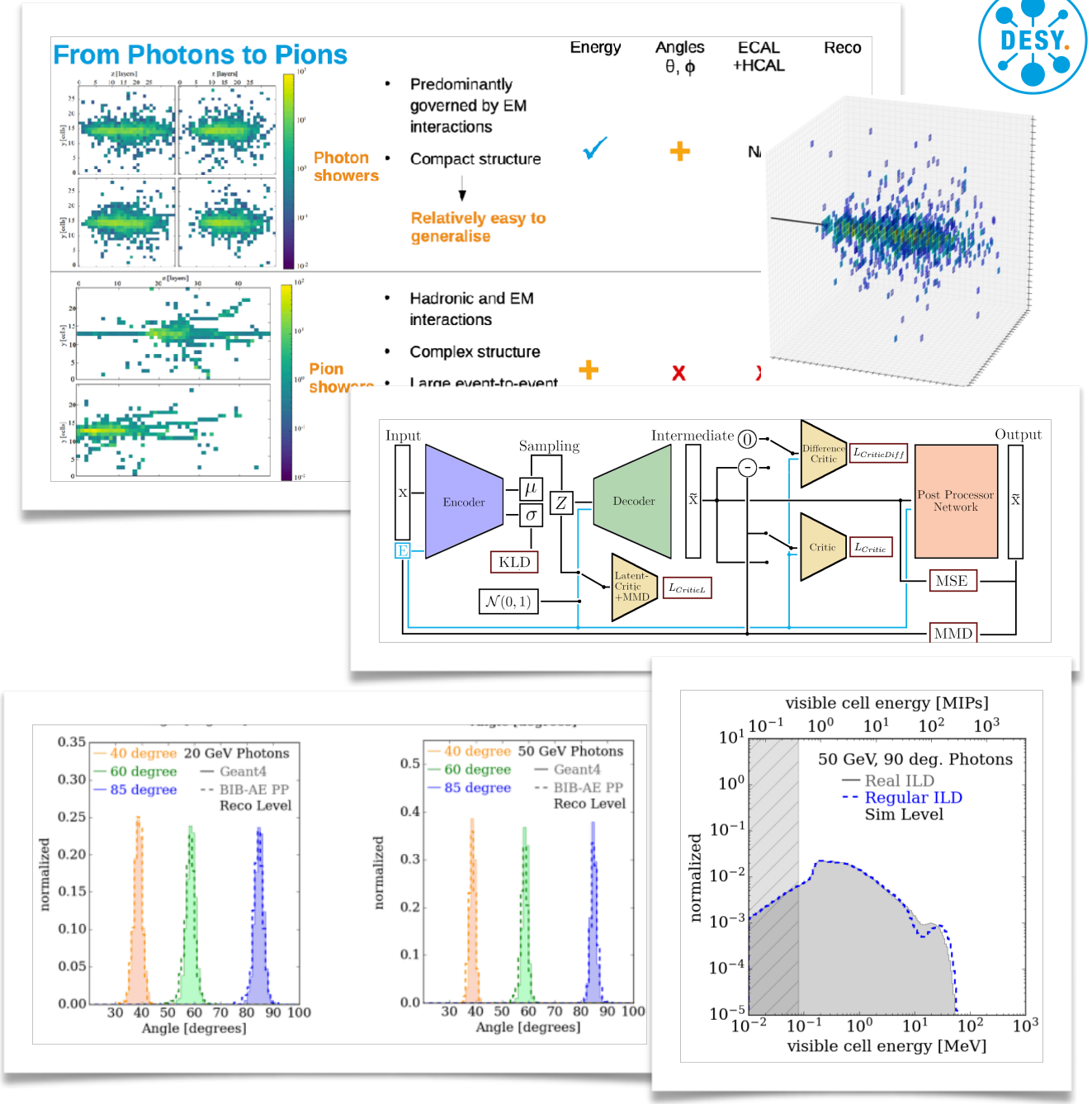


- might eventually benefit from this implementation
- expect significant effort to adapt to ILD tracking (w. TPC - cylinders) and all pattern recognition
- probably more a **midterm** project …

AI/ML in Key4hep

generative ML for fast shower simulation

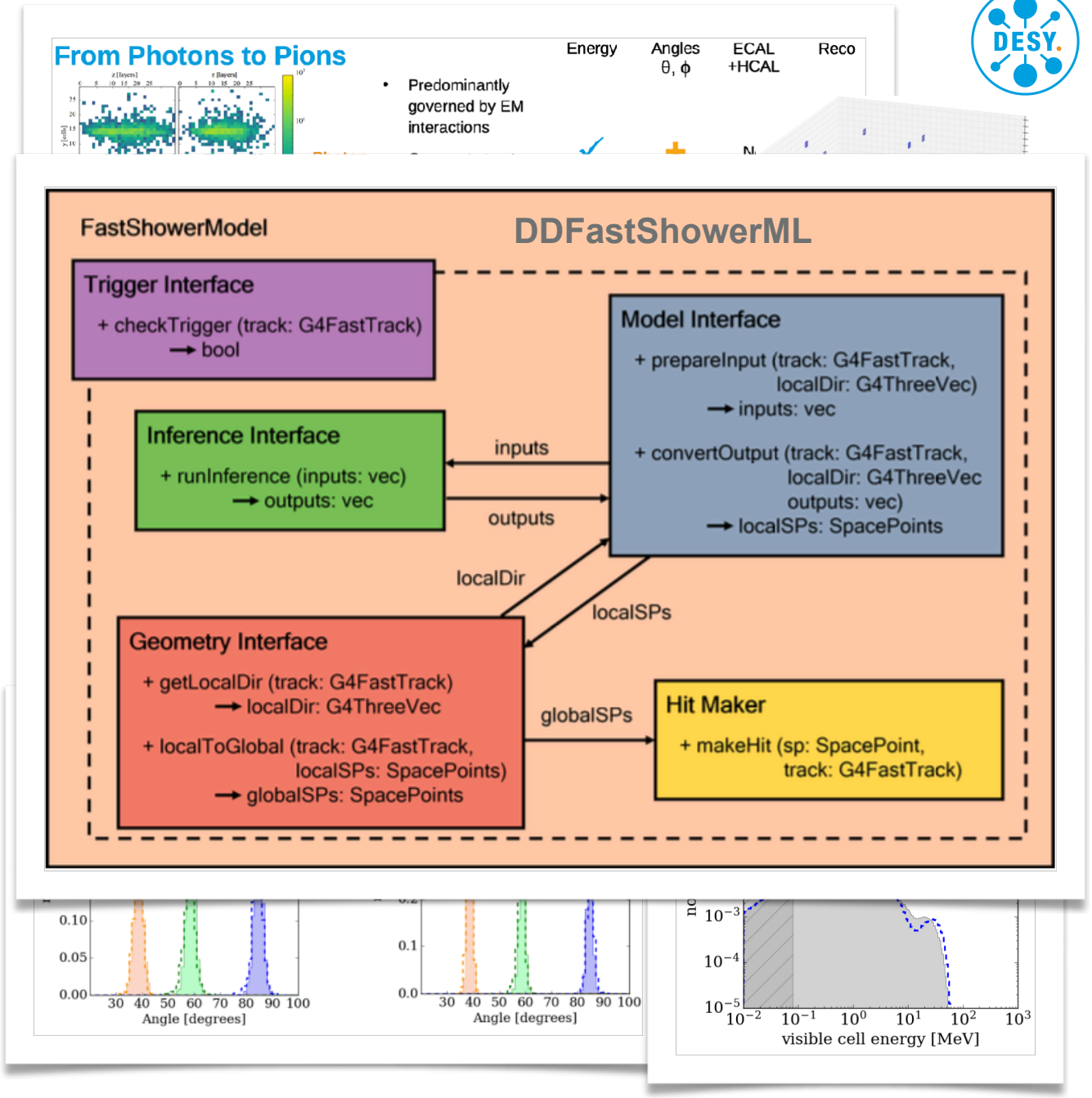
- shower generation with ML offers great potential O(10e3) for faster and more sustainable simulation in HEP
- DESY/UHH QU group studying many generative models: (W)GANs, VAE, BIB-AEs, Normalising Flows, Point Cloud Diffusion Models,…
- achieve high fidelity on individual physics distributions



AI/ML in Key4hep

generative ML for fast shower simulation

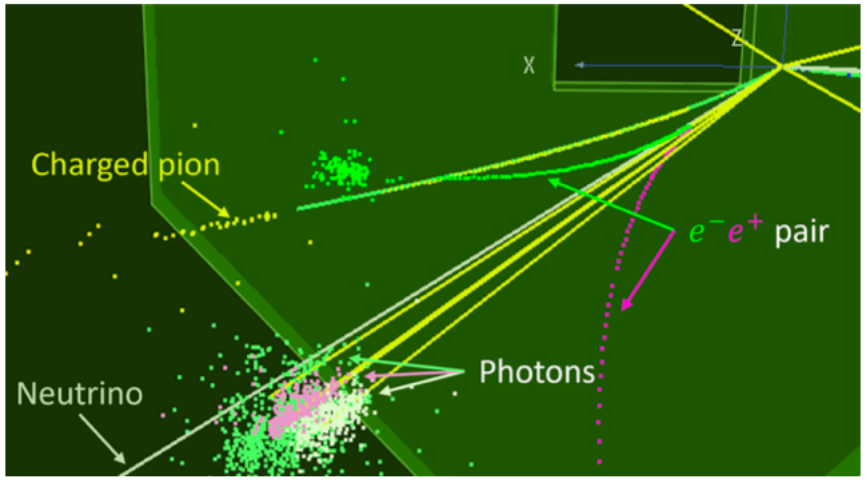
- shower generation with ML offers great potential O(10e3) for faster and more sustainable simulation in HEP
- DESY/UHH QU group studying many generative models: (W)GANs, VAE, BIB-AEs, Normalising Flows, Point Cloud Diffusion Models,…
- achieve high fidelity on individual physics distributions
- developed *DDFastShowerML* library in Key4hep to transparently call ML inference during full simulation with DDG4/Geant4



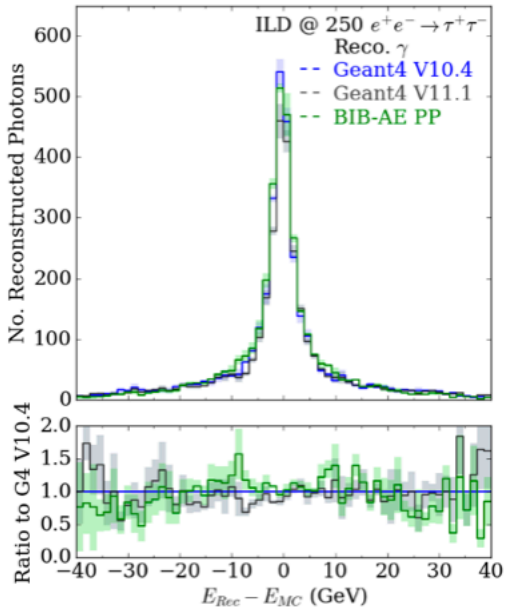
AI/ML in Key4hep

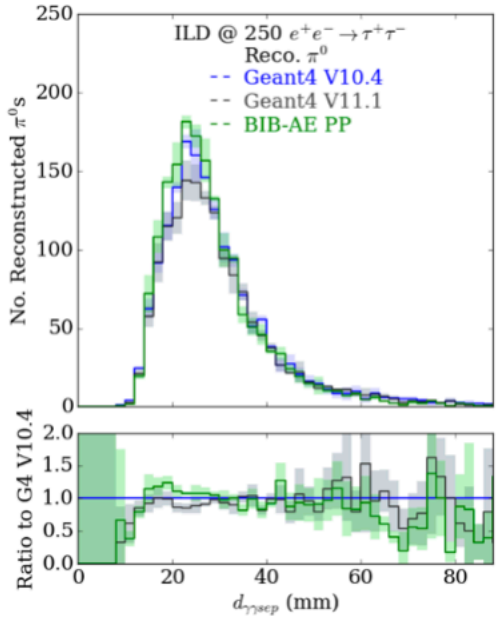
generative ML for fast shower simulation

- shower generation with ML offers great potential O(10e3) for faster and more sustainable simulation in HEP
- DESY/UHH QU group studying many generative models: (W)GANs, VAE, BIB-AEs, Normalising Flows, Point Cloud Diffusion Models,…
- achieve high fidelity on individual physics distributions
- developed *DDFastShowerML* library in Key4hep to transparently call ML inference during full simulation with DDG4/Geant4
- allows for **realistic benchmarking** of ML fastsim on **real physics observables** after full **reconstruction**



P.McKeown: PhD thesis 2024

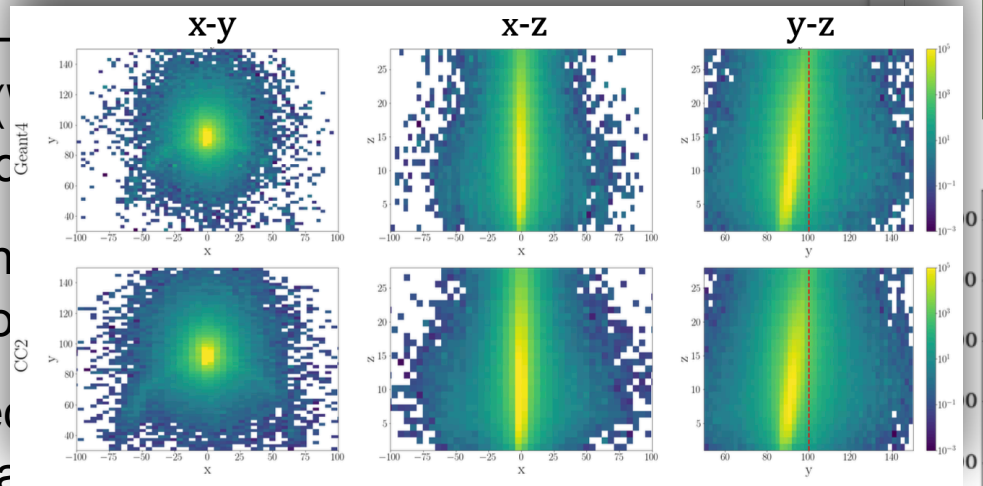




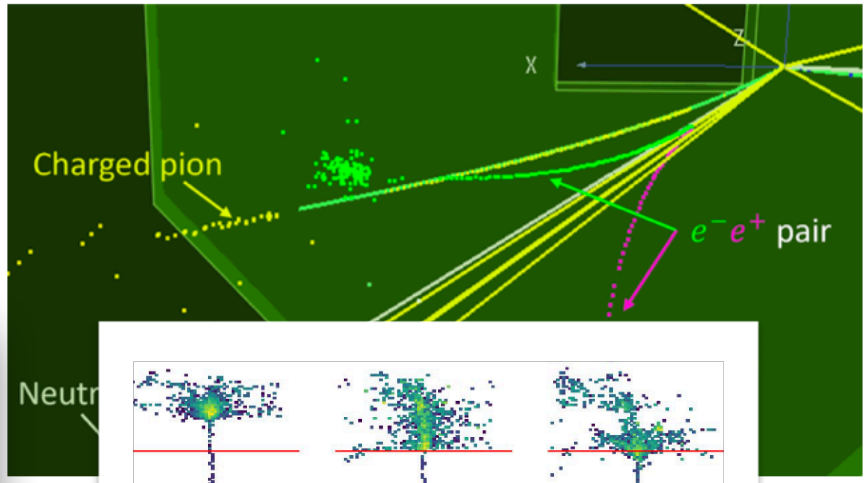
AI/ML in Key4hep

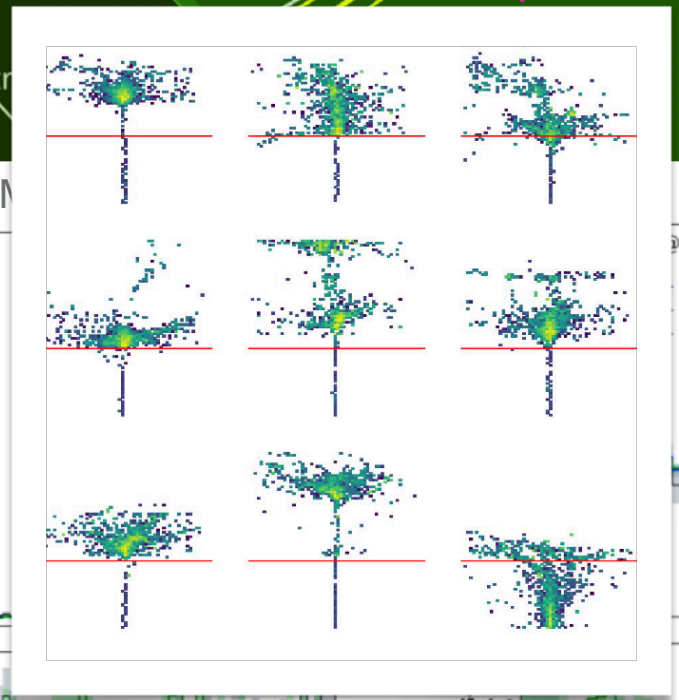
generative ML for fast shower simulation

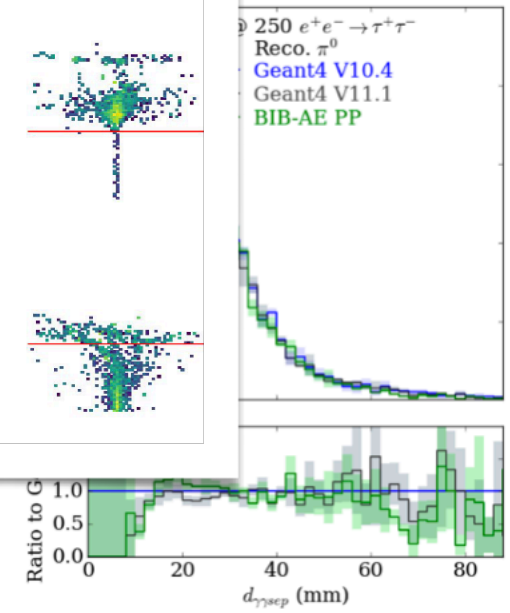
- shower generation with ML offers great potential O(10e3) for faster and more sustainable simulation in HEP
- DESY/UH… models: (… Flows, Po…
- achieve h… distributio…
- develope… to transpa…
- …



ongoing work:
- apply CaloClouds diffusion model to **CMS HGCal** and use for **hadronic showers** in ILD







AI/ML in Key4hep

T.Suehara et al

Flavour tagging with deep learning methods

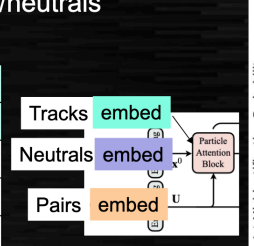
- implemented ParticleTransformer flavour tagging for ILD
  - achieve **dramatically better** results than LCFIPlus
- observe strange improvement w/ more training data at FCC -> to be studied !
- framework (Marlin/Gaudi?) inference work in progress
- started to look into **strange tagging**

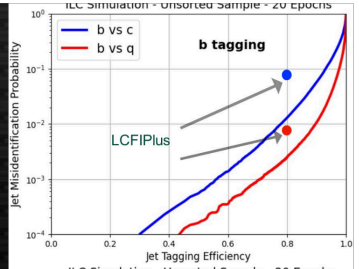
Improvements wrt. LCFIPlus

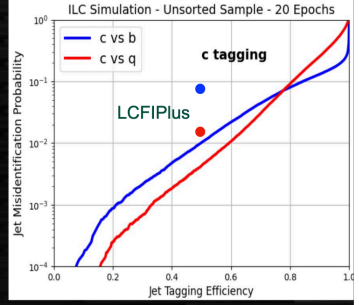
- Factor (3-9) improvement at ParT from LCFIPlus without any tuning
- Another factor (max 3) improvement by tuning
  - Optimizing input variables
  - Separate embedding for tracks/neutrals

	b-tag 80% eff.		c-tag 50% eff.	
background	c jets	uds jets	b jets	uds jets
+LCFIPlus (BDT)	6.3%	0.79%	7.4%	1.2%
*ParT (initial)	1.3%	0.25%	1.0%	0.43%
**ParT (improved)	0.48%	0.14%	0.86%	0.34%

+LCFIPlus (BDT) 250 GeV nnqq
*ParT (initial) 91 GeV qq, default settings
**ParT (improved) 250 GeV nnqq, b/c/d separation

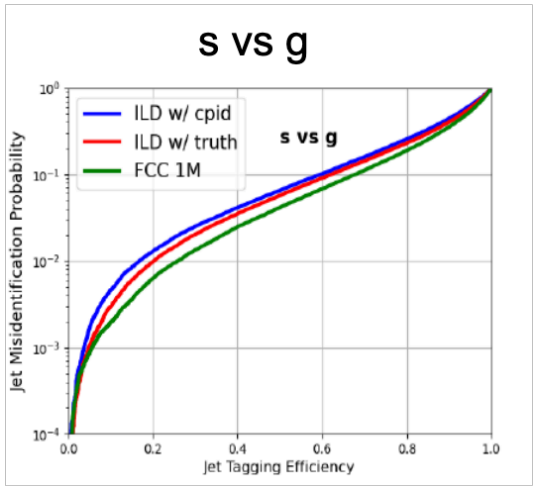


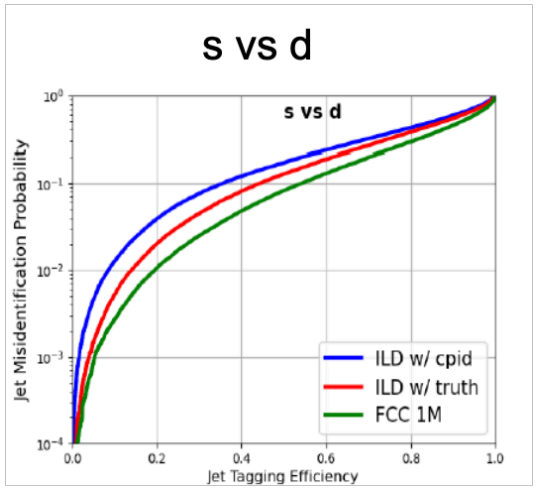




Sample / sample size	b-tag 80% eff.		c-tag 50% eff.	
background	c jets	uds jets	b jets	uds jets
ILD full-sim 1M (optimized)	0.48%	0.14%	0.86%	0.34%
FCCee Delphes 1M (reduced)	0.47%	0.12%	0.64%	0.10%
FCCee Delphes 1M (full)	0.21%	0.054%	0.36%	0.059%
FCCee Delphes 4M	0.045%	0.025%	0.20%	0.033%
FCCee Delphes 6M	0.014%	0.010%	0.13%	0.022%
FCCee Delphes 8M	0.007%	0.006%	0.076%	0.021%

We see mild consistency between ILD and FCC!





AI/ML in Key4hep

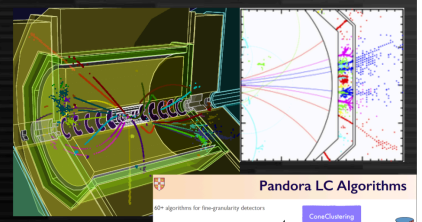
PFA with ML

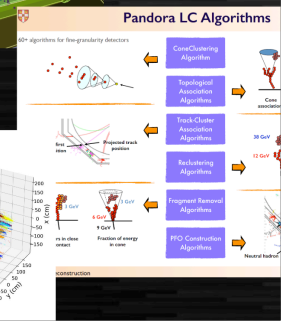
T.Suehara

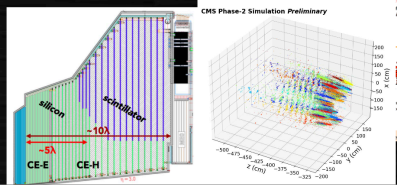
- started to develop deep neural networks for particle flow - partly based on CMS HGCal
  - using GravNet and Object Condensation
- after early, promising results **made significant progress**

Particle flow with DNN: introduction

- Separation of cluster at calorimeter
  - Charged or neutral cluster
- Essential for jet energy resolution
- Current algorithm: PandoraPFA
  - Combination of various process
  - Not easy to optimize or adding more info
- CMS HGCal clustering
  - Similar to ILD calo
  - Good for starting point

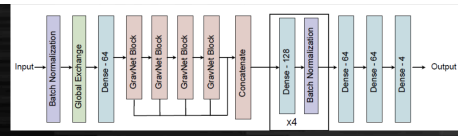






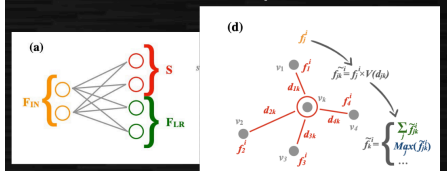
PFA: clustering algorithm

- Input: position/energy/timing of each hit
- Output: virtual coordinate and β for each hit



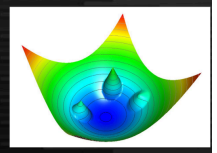
GravNet arXiv:1902.07987

- The virtual coordinate (S) is derived from input variables with simple MLP
- Convolution using "distance" at S (bigger convolution with nearer hits)
- Concatenate the output with MLP



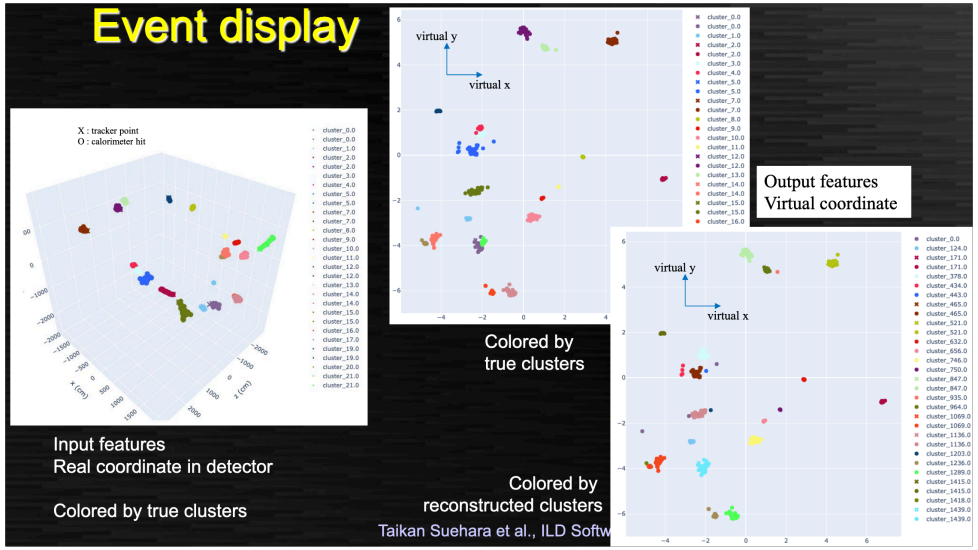
Object Condensation (loss function) arXiv:2002.03605

$$L = L_p + s_C(L_\beta + L_V)$$



- Condensation point: The hit with largest β at each (MC) cluster
- L_V: Attractive potential to the condensation point of the same cluster and repulsive potential to the condensation point of different clusters
- L_β: Pulling up β of the condensation point
- L_p: Regression to output features

Event display



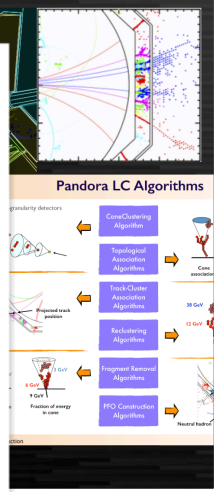
AI/ML in Key4hep

PFA with ML

T.Suehara

- started to develo
  flow - partly bas
  - using GravN
- after early, prom
  progress

Particle flow with DNN: introduction



Results on efficiency and purity

Algorithm train/test	Electron eff.	Pion eff.	Photon eff.	Electron pur.	Pion pur.	Photon pur.
GravNet 10 taus/10 taus	99.1%	96.5%	99.0%	91.8%	98.9%	97.1%
PandoraPFA 10 taus	99.3%	94.0%	99.1%	91.8%	94.6%	97.2%
GravNet jets/jets	94.5%	93.1%	95.2%	94.6%	93.2%	92.4%
PandoraPFA jets	80.2%	90.4%	79.0%	75.0%	90.6%	77.7%
PandoraPFA jets (ILCSoft truth)	96.7%	95.5%	96.4%	97.1%	90.4%	97.7%

At least in our measure, performance of GravNet-based algorithm exceeds PandoraPFA
→ Promising as full PFA (but energy regression to be done)
Definition of MC truth clusters needs to be tuned (see ILCSoft truth)

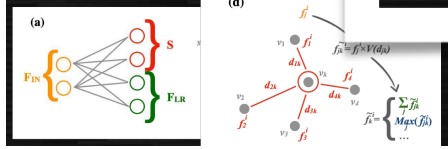
Taikan Suehara et al., ILD Software and Analysis meeting, 2 Oct. 2024, page 21

PFA: clustering a

- Input: position/energy/timing
- Output: virtual coordinate a

GravNet arXiv:1902.07987

- The virtual coordinate (S) is
  from input variables with sim
- Convolution using "distance
  (bigger convolution with nea
- Concatenate the output with



point of different clusters

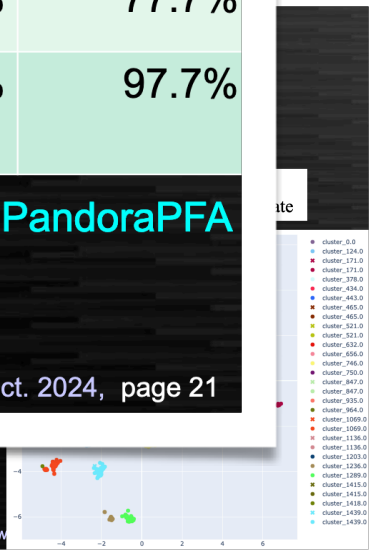
- L_β: Pulling up β of the condensation point
- L_p: Regression to output features

Real coordinate in detector

Colored by true clusters

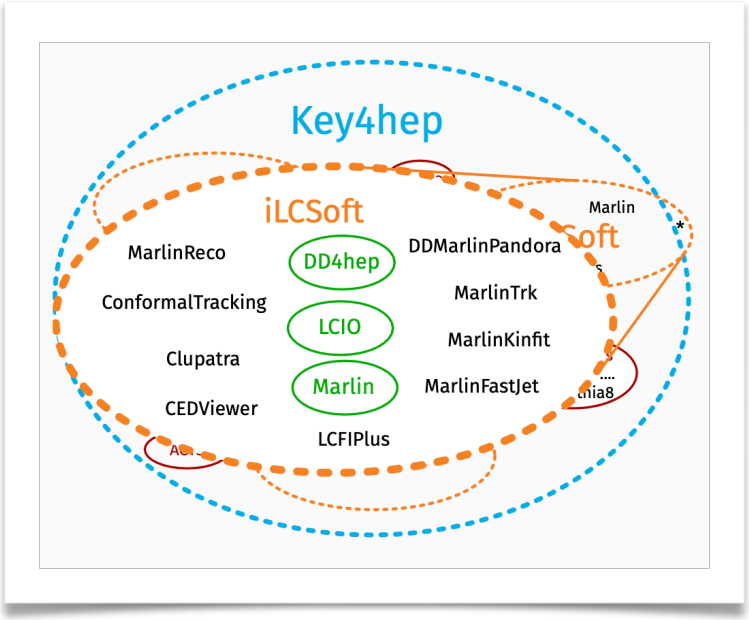
Colored by reconstructed clusters

Taikan Suehara et al., ILD Softw



Summary and Outlook

- **Key4hep** started as a new future collider community wide effort in 2020 to put together a modern turnkey software stack
  - contributors: CEPC, CLIC, FCC, EIC, ILC, LUXE, Muon Collider …
- **iLCSoft and ILD software integral part of Key4hep from the start**
- **battle proven ILD standard reconstruction** can be run in Key4hep w/ **MarlinWrapper** as before - or w/ EDM4hep output
- many **AI/ML activities ongoing** (fast sim and (high level) reconstruction)
  - eventually need to validate and integrate in ILD standard reconstruction



- Key4hep offers great opportunity to **modernise ILD software** stack AND **collaborate** w/ **other Higgs factories** - when studying ILD for FCC
- yet, limiting factor for all software developments: **manpower**
- need to somewhat balance the work on core tools and novel (AI) algorithms w/ ILD detector optimisation for FCC (and ILC)

pointers to documentation

entry points to Key4hep

- Key4hep GitHub Project
  - https://github.com/key4hep
- Key4hep main documentation page
  - https://key4hep.github.io/key4hep-doc/
- Doxygen available., e.g. for EDM4hep
  - https://edm4hep.web.cern.ch/
- iLCSoft Github Project
  - https://github.com/ilcsoft