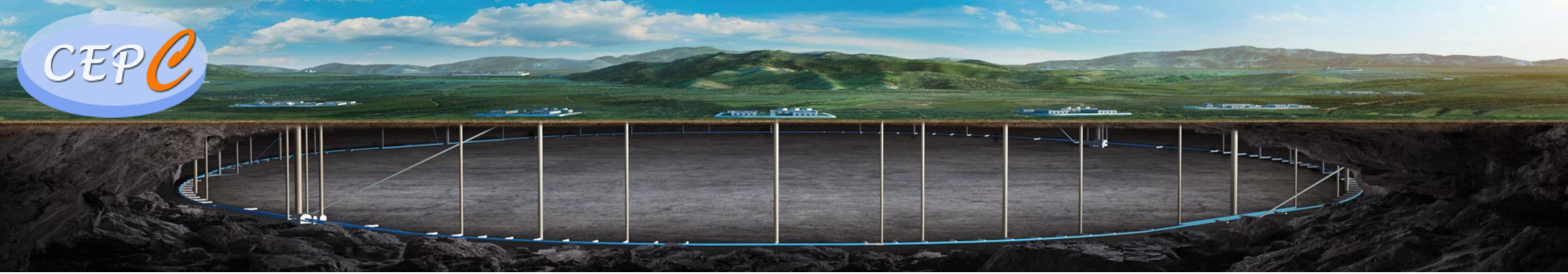


CEPC Software and Computing

Weidong Li (IHEP)

representing CEPC software and computing teams

中国科学院高能物理研究所
Institute of High Energy Physics
Chinese Academy of Sciences

Content

- Introduction
- Requirements
- Technical Options
  - Technical challenges
  - Technology survey and our choices
- Software
- Computing
- Research Team and Working Plan
- International Collaboration
- Summary

Introduction

- The development of CEPC software started with the iLCSoft
  - Developed CEPC components for simulation and reconstruction
  - Generated M.C. data for detector design and physics potential studies
  - Particularly, CEPC CDR studies done with the iLCSoft

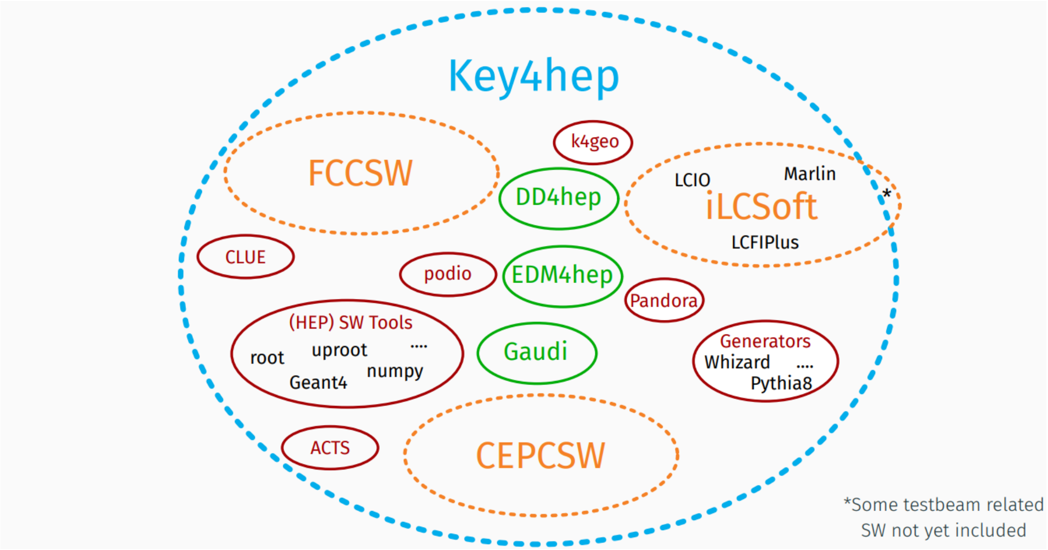


Figure from F. Gaede

- The consensus among CEPC, CLIC, FCC, ILC and other future experiments was reached at the Bologna workshop in 2019
  - Develop a Common Turnkey Software Stack, Key4hep, for future collider experiments
  - Maximize the sharing of software components among different experiments

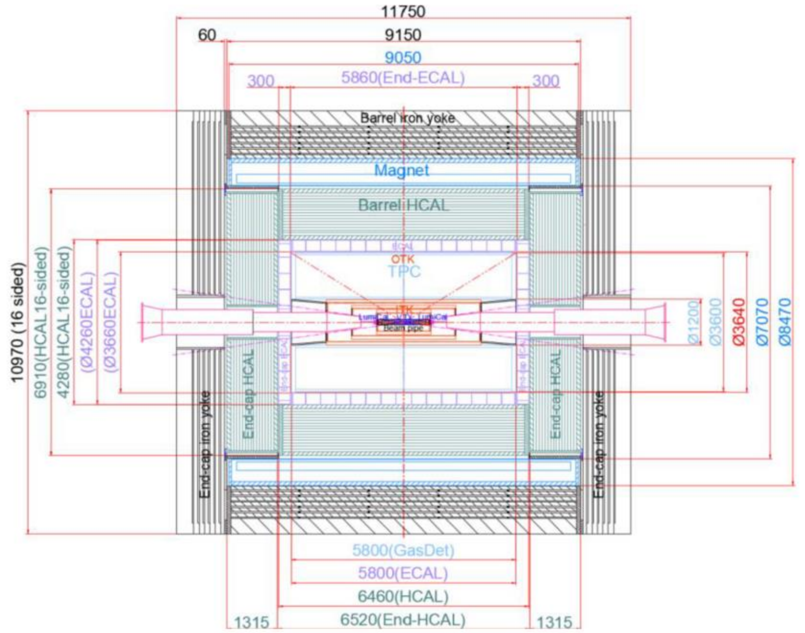
Requirements

- Physics requirements
  - High tracking efficiency (~100%), momentum resolution (<0.1%), PID (2σ p/K separation at P < ~ 20 GeV/c), BMR better than 4% etc.
- CEPC Reference Detector
  - VTX: Vertex Detector
  - ITK/OTK: Inner Silicon Tracker/Outer Silicon Tracker
  - TPC: Time Projection Chamber
  - ECAL: Crystal-bar ECAL
  - HCAL: Glass Scintillator HCAL
  - Muon: Muon Detector
- Both detector design and physics potential studies need software

Physics process	Measurands	Detector subsystem	Performance requirement
$ZH, Z \to e^+e^-, \mu^+\mu^-$ $H \to \mu^+\mu^-$	$m_H, \sigma(ZH)$ $\mathrm{BR}(H \to \mu^+\mu^-)$	Tracker	$\Delta(1/p_T) =$ $2 \times 10^{-5} \oplus \frac{0.001}{p(\mathrm{GeV}) \sin^{3/2}\theta}$
$H \to b\bar{b}/c\bar{c}/gg$	$\mathrm{BR}(H \to b\bar{b}/c\bar{c}/gg)$	Vertex	$\sigma_{r\phi} =$ $5 \oplus \frac{10}{p(\mathrm{GeV}) \times \sin^{3/2}\theta} (\mu\mathrm{m})$
$H \to q\bar{q}, WW^*, ZZ^*$	$\mathrm{BR}(H \to q\bar{q}, WW^*, ZZ^*)$	ECAL HCAL	$\sigma_E^{\mathrm{jet}}/E =$ $3 \sim 4\%$ at 100 GeV
$H \to \gamma\gamma$	$\mathrm{BR}(H \to \gamma\gamma)$	ECAL	$\Delta E/E =$ $\frac{0.20}{\sqrt{E(\mathrm{GeV})}} \oplus 0.01$

Table 3.3: Physics processes and key observables used as benchmarks for setting the requirements and the optimization of the CEPC detector.

CEPC CDR



Technical Options

Main technical challenges

- How to develop large scale software which can be run on diverse hardware like CPU/GPU/FPGA/other accelerators
- How to take advantage of novel technologies such as AI and Quantum Computing technologies to cope with the tasks that involve complicated data processing

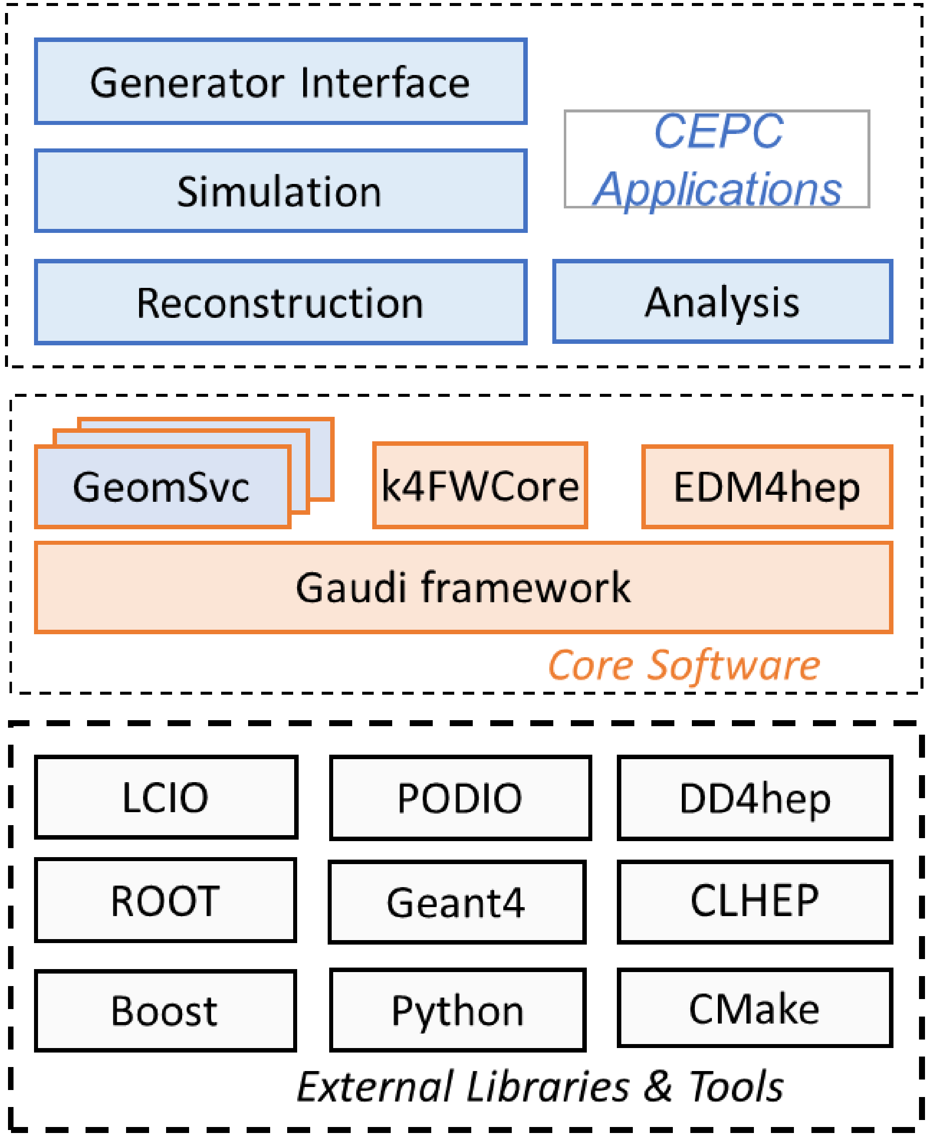
Technology survey and choice

- Choose important HEP software packages as the cornerstone of CEPCSW
  - Software framework: GAUDI
  - Simulation: GEANT4
  - Analysis: ROOT
  - Detector Description: DD4hep
  - Event Data Model: EDM4hep
- Join the efforts to resolve the common challenges confronted by future HEP experiments
- At the same time, develop CEPC specific software to meet the experiment's needs

Architecture of CEPCSW

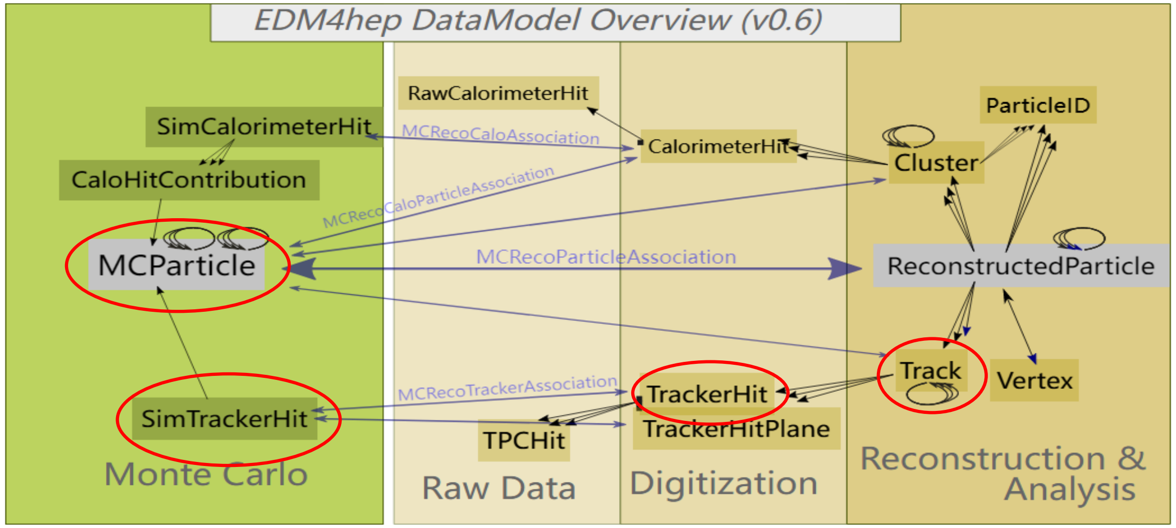
- CEPCSW is organized as a multi-layer structure
  - Applications
    - simulation, reconstruction and analysis
  - Core software
  - External libraries

- The key components of core software include:
  - Gaudi: defines interfaces to all software components
  - k4FWCore: management of event data objects
  - EDM4hep: generic event data model
  - DD4hep: detector geometry description
  - CEPC-specific components: GeomSvc, simulation framework, analysis framework, beam background mixing, fast simulation, machine learning interface, etc.

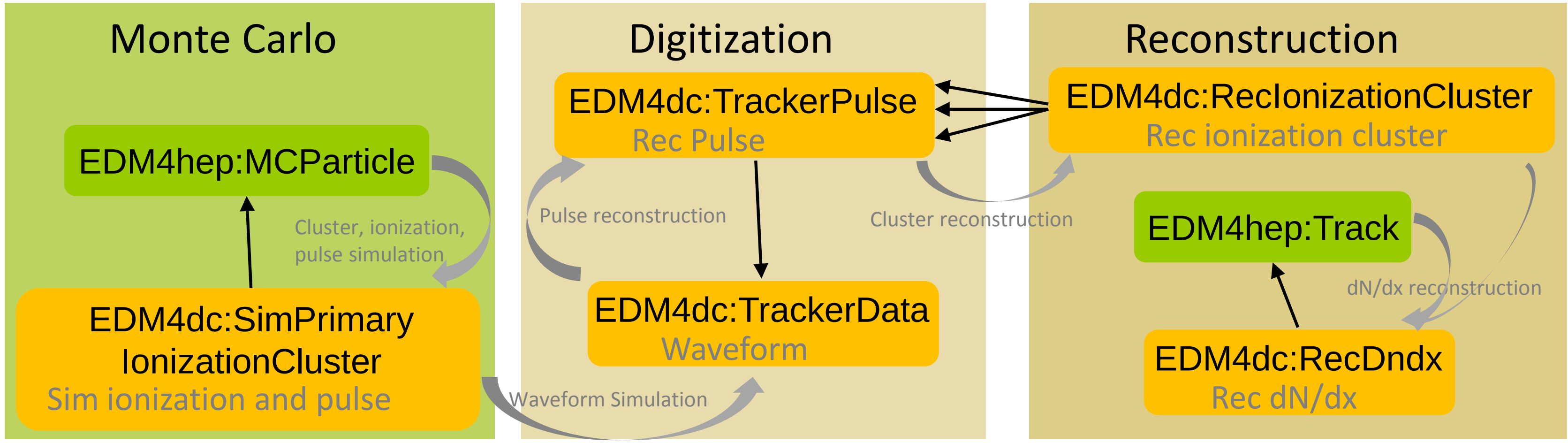


Event Data Model

- EDM4hep is the common event data model (EDM) developed for the future experiments like CEPC, CLIC, FCC, ILC, etc.
  - EDM4hep describes event objects created at different data processing stages and also reflects the relationship between them

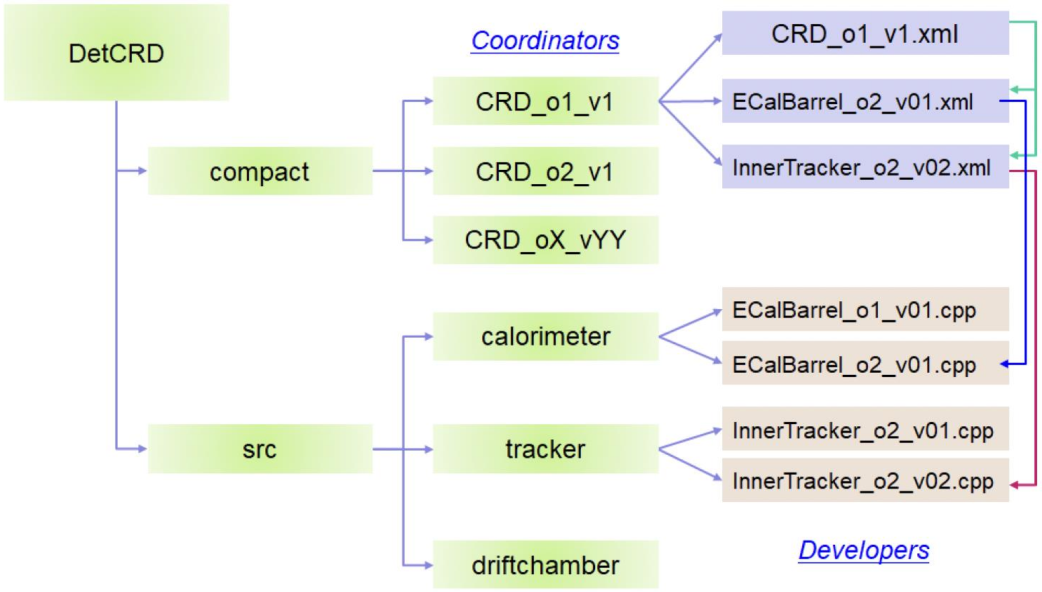


- EDM4hep has been extended to meet the CEPCSW requirements
  - By using the upstream mechanism of PODIO, more EDM4hep classes have been created for TPC, drift chamber and OTK TOF

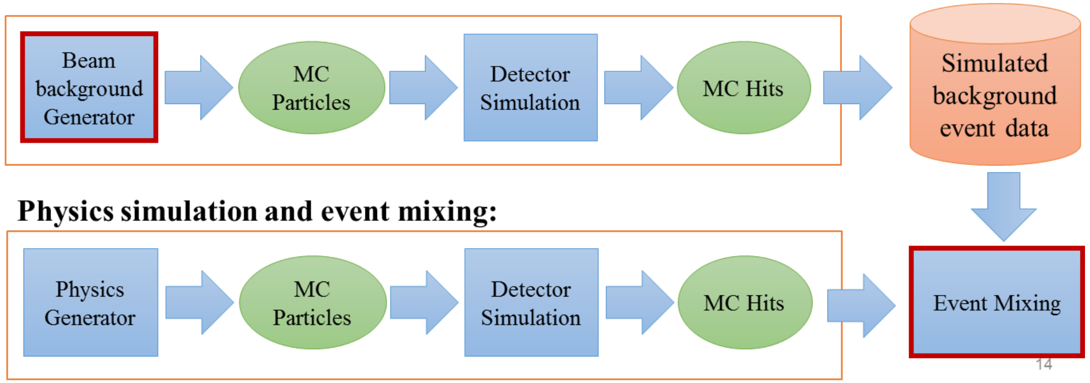


Simulation Framework (1)

- DD4hep was adopted to provide a full detector description
  - Geometry generated from a single source
- Different detector design options are managed in the Git repository
  - Simulation jobs can be easily configured in runtime

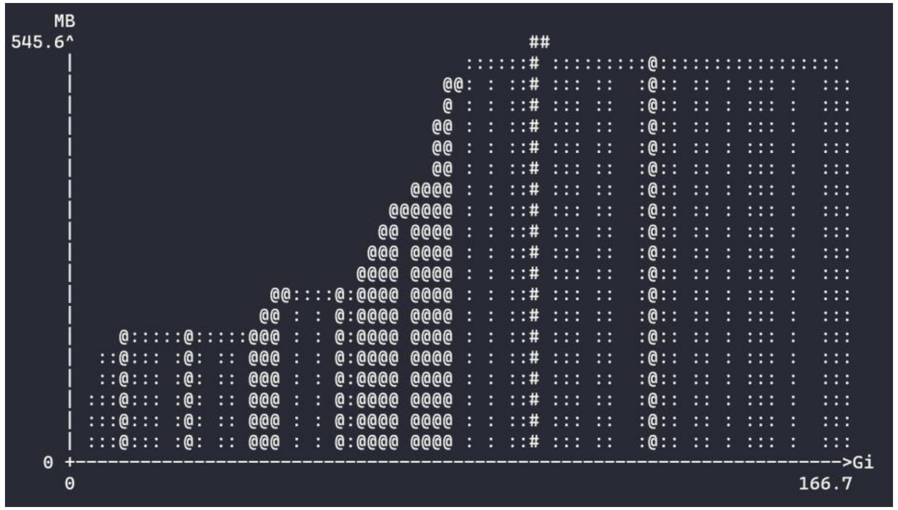


- The simulation framework has been developed
  - based on which the simulation chain for sub-detectors was built
- The region-based fast simulation interface is also available
  - to integrate different fast simulation modules into the simulation chain
- An event mixing tool is provided
  - to mix different types of background events with physics signals at hit level



Simulation Framework (2)

- Multi-threading mandatory
  - Reduce memory footprint

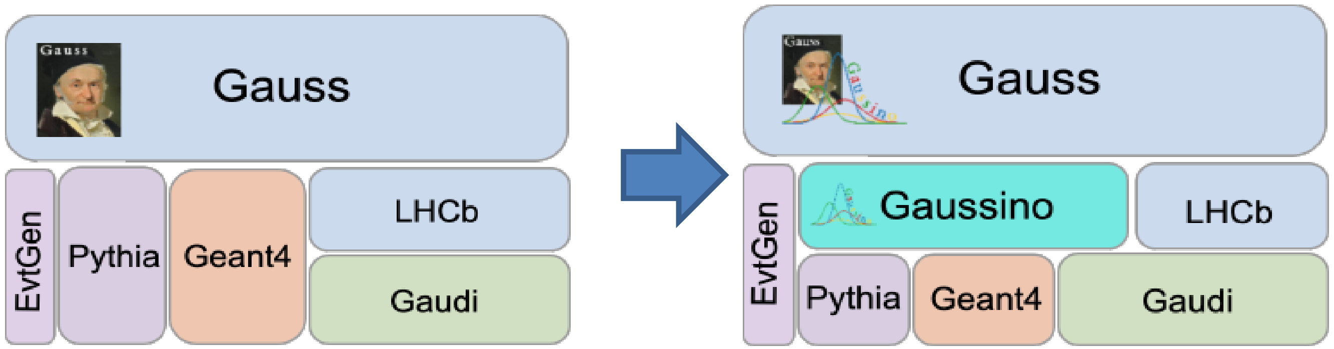


Heap memory usage (serial simulation)

Simulation setup:
- Detector: TDR_o1_v01
- Physics list: QGSP_BERT
- Generation: single muons
- N events: 100

About 1GB memory needed for reading detector geometry at the initialization stage.

- Use LHCb's Gaussino as a common framework
  - Planned by Key4hep. [arXiv: 2312.08152]

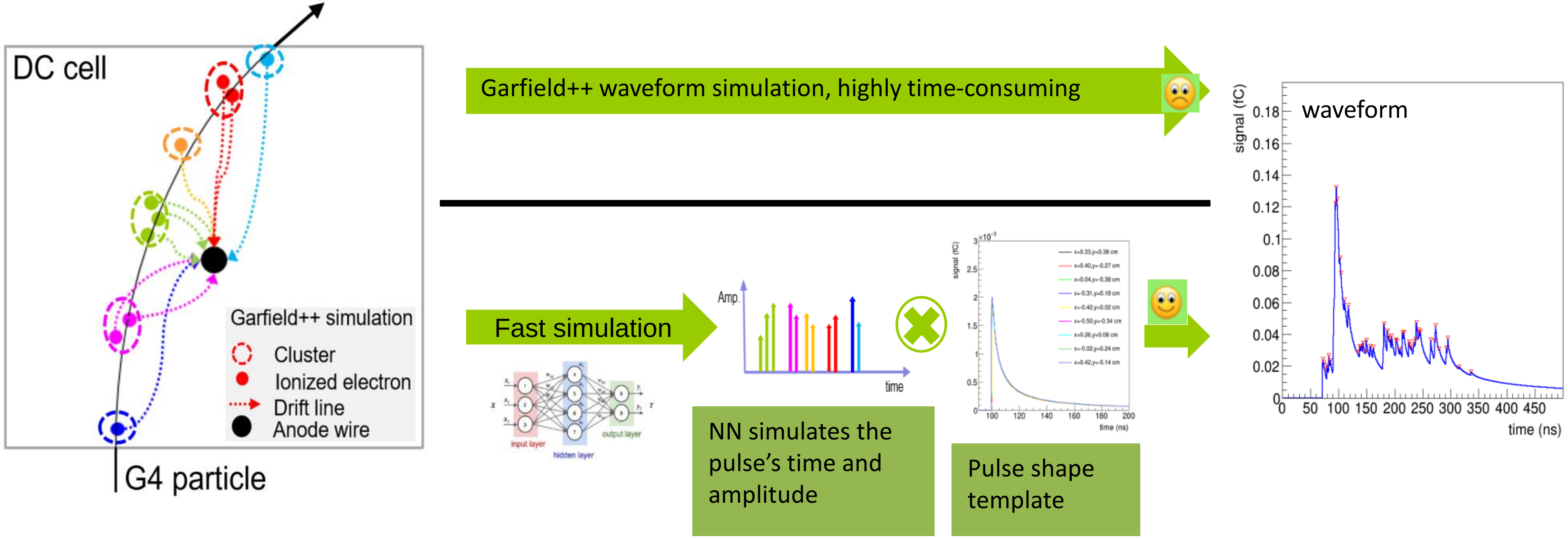


- Implementation of CEPC-on-Gaussino prototype
  - Cooperated with the software experts from CERN
  - Removed dependencies on the LHCb software
  - As a demo, VTX simulation was implemented

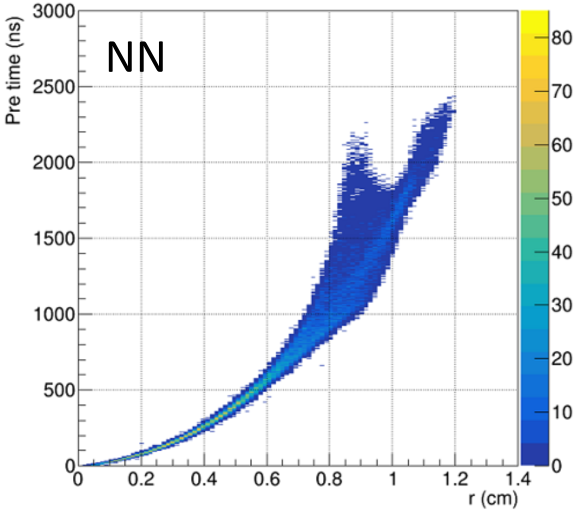
ICHEP 2024 Parallel Talk by Tao Lin

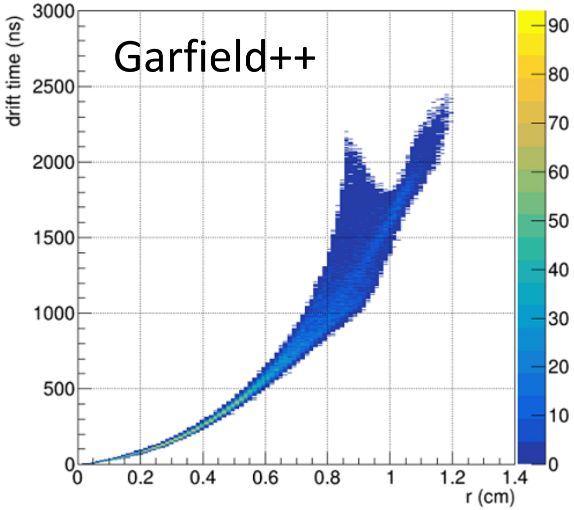
Gaseous Detector Simulation (1)

- TrackHeedSimTool (Gaudi tool) was implemented by combining Geant4 and neural network to simulate the complete response of the gaseous detector
  - Input: G4Step information (particle type, initial position, momenta, and step length)
  - Using TrackHeed(from Garfield++) to create the ionization electron-ion pairs (for both primary and secondary ionizations), the deposited energy will be used to update the energy of the G4Particle
  - Using NN to simulate the time and amplitude of each pulse for each ionized electron (for fast waveform simulation)

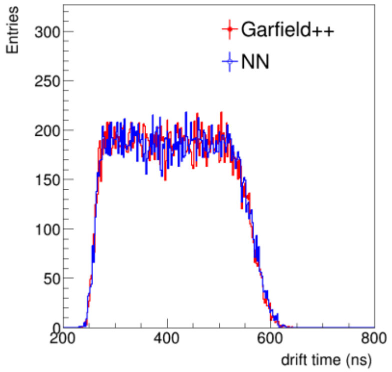


Gaseous Detector Simulation (2)

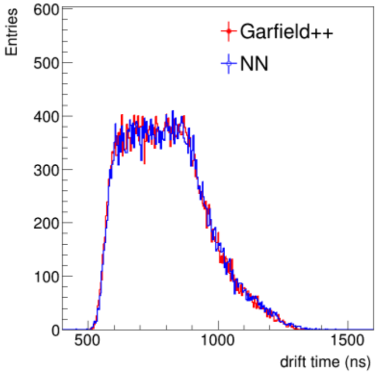




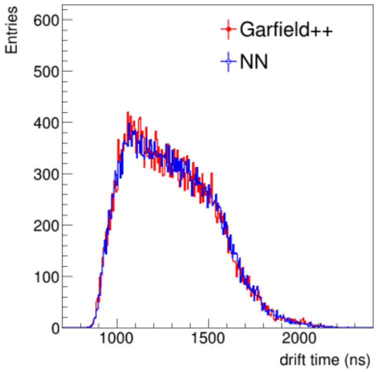
r: 0.4-0.6 cm



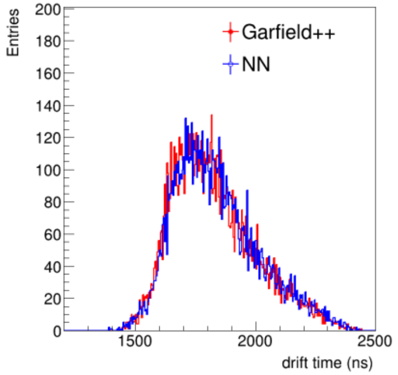
r: 0.6-0.8 cm



r: 0.8-1.0 cm



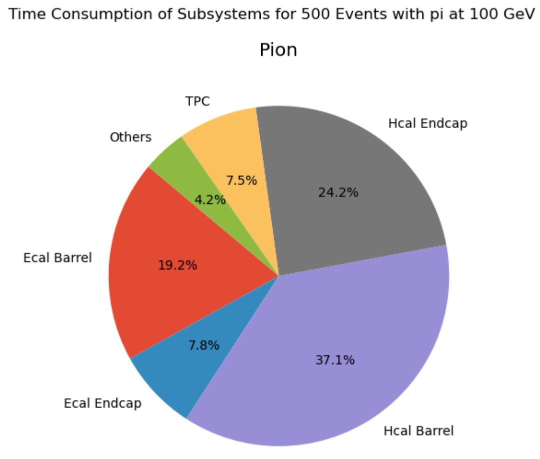
r: 1.0-1.2 cm

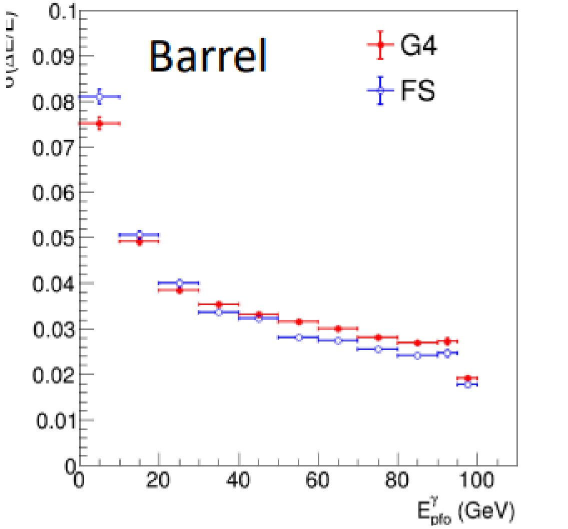


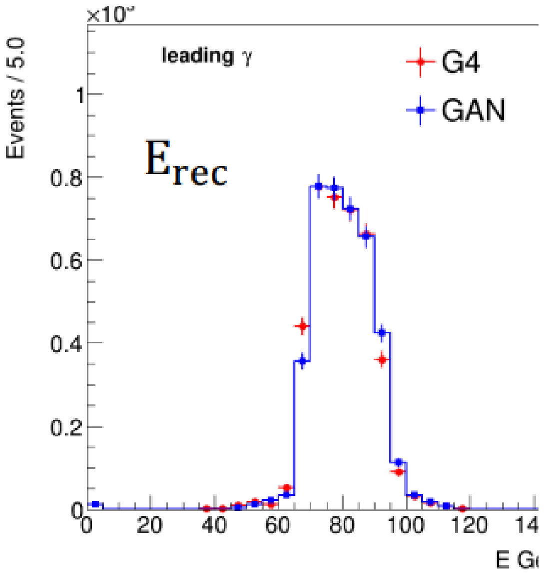
- Good agreement between the NN and Garfield++ simulation

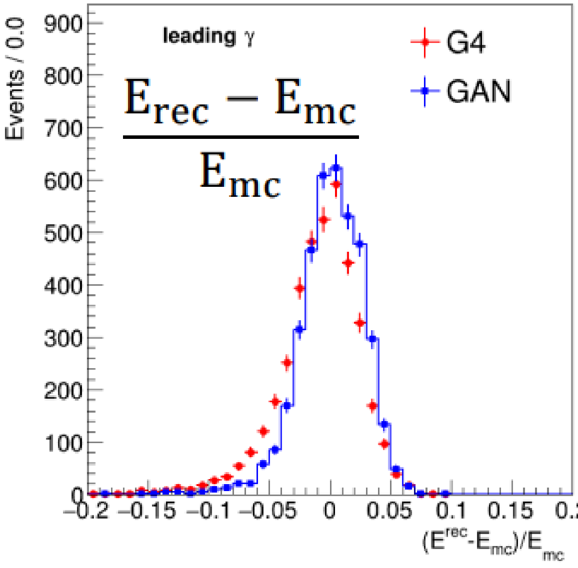
Fast Simulation

- More than 80% of the CPU power is devoted to simulating the calorimeters
- To reduce overall simulation time, fast calorimeter simulation is is essential
  - Study was based on silicon tungsten ECAL
  - Developed method is being applied to the ref-detector
- Frozen shower method (GS)
  - Good agreement between FS and Geant4 simulation
  - The simulation speed was doubled
- Machine learning method based on GAN
  - For simplicity, only the simulation of central barrel was studied
  - Good agreement between GAN and Geant4 was reached
  - More than 2 magnitudes speed-up can be achieved



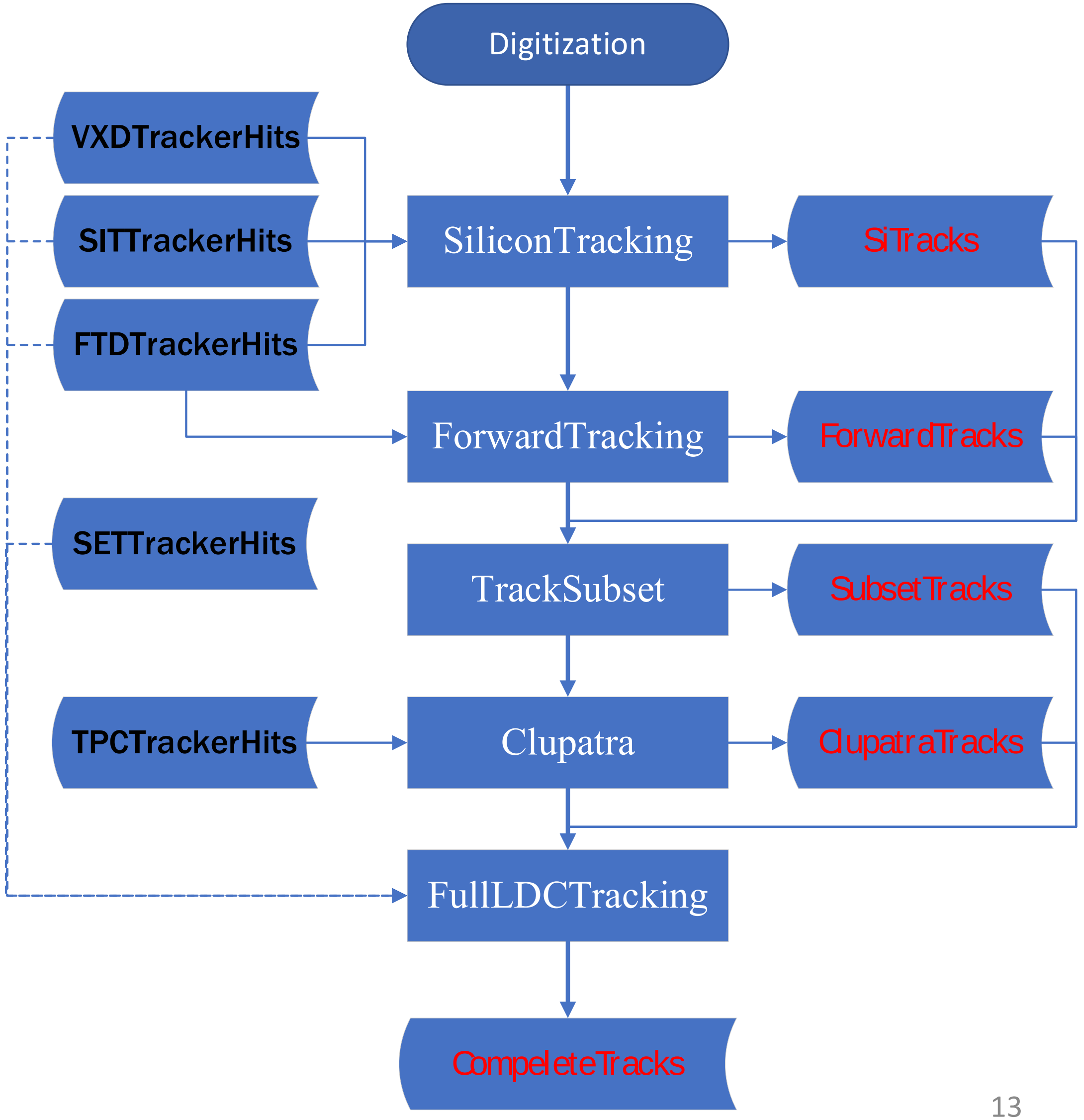






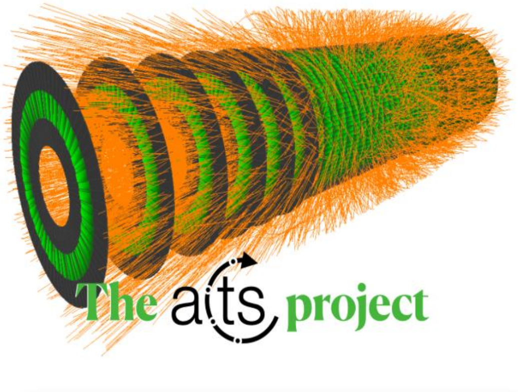
Tracking Software (1)

- Track reconstruction in VTX, ITK, TPC, and OTK
  - Track finding starts from the outmost layer in the ITK and then moves from outside to inside
  - For a found track candidate in ITK and VTX, the OTK hits will be selected and added to it
  - The search for tracks in TPC is independent from track finding in silicon trackers
  - A combined fit is applied to all the track hits in silicon trackers and TPC
- Algorithms and tools originally from ILD
  - SiliconTracking: VTX and ITK
  - ForwardTracking: ITK
  - TrackSubset: removing duplicated tracks
  - Clupatra: TPC
  - FullILDTracking: combined fit

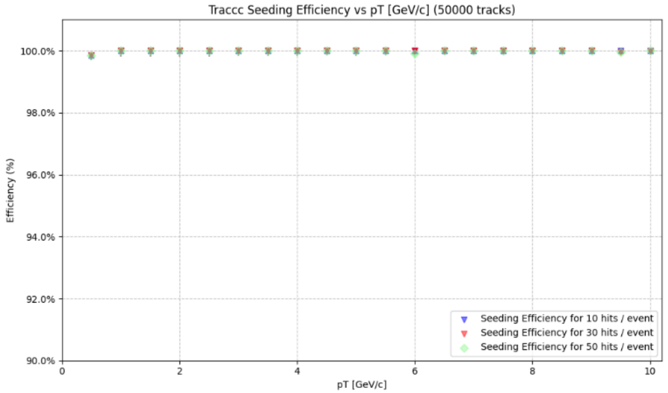


Tracking Software (2)

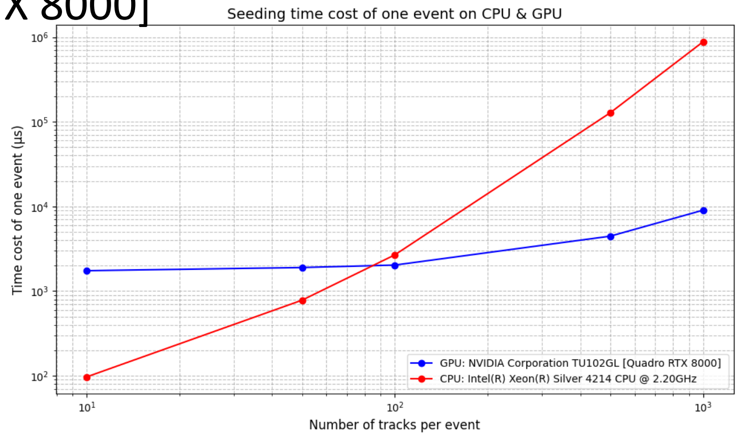
- ACTS is an experiment-independent toolkit for track reconstruction
  - High-performance parallel software for complex hardware (CPU/GPU/…)
  - TRACCC is a GPU tracking demonstrator and candidate for ATLAS Online reconstruction



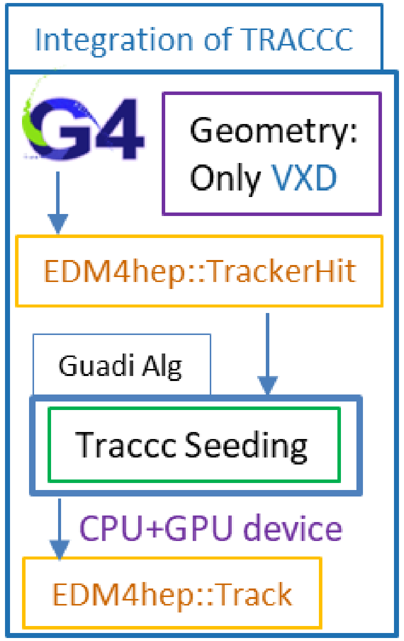
- Developing CEPC seeding algorithm based on TRACCC
  - Integration of TRACCC with the CEPCSW
  - Seeding algorithm of VXT was implemented and can be on both CPU and GPU
- Computing evaluation of TRACCC seeding
  - CPU: Intel(R) Xeon(R) Silver 4214 CPU @ 2.20GHz
  - GPU: NVIDIA Corporation TU102GL [Quadro RTX 8000]



Seeding efficiency versus transverse momentum (p=10GeV)

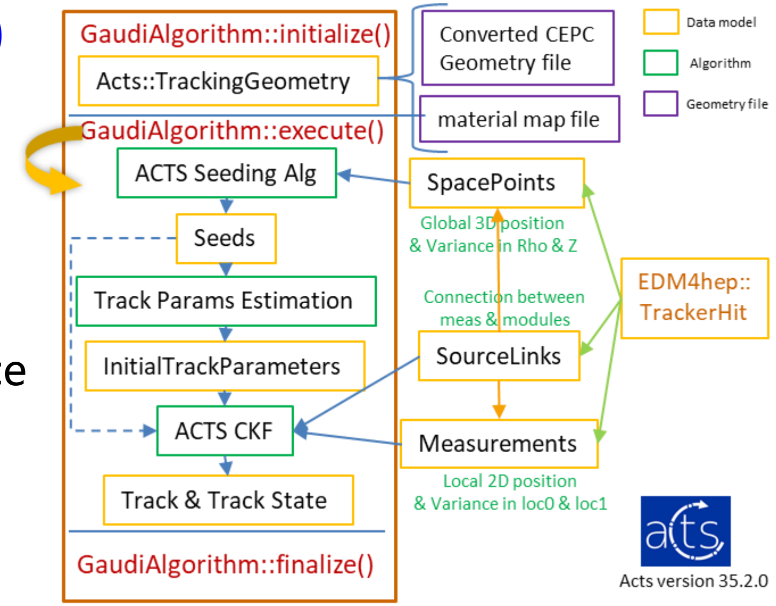


Seeding time versus # of track / event on GPU & CPU

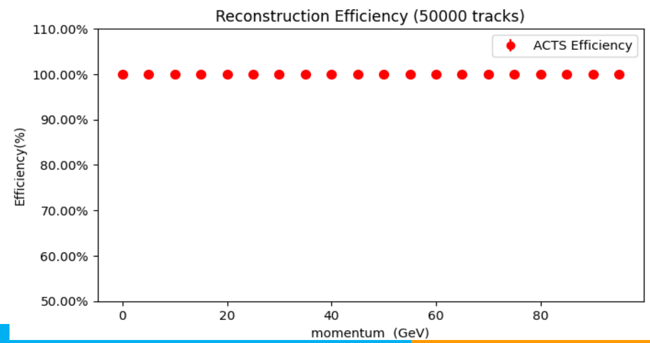


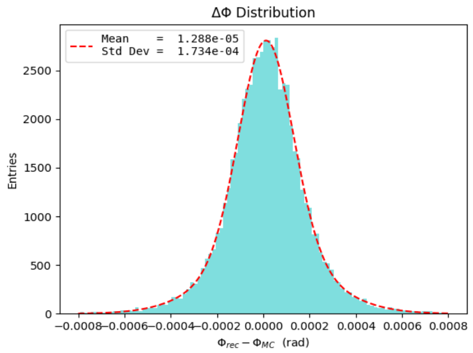
Tracking Software (3)

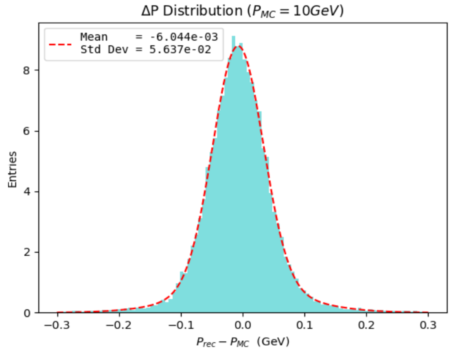
- Implementation of track reconstruction with ACTS for Vertex Detector (VTX) and Inner Silicon Tracker (ITK) of the CEPC experiment
  - A Gaudi algorithm has been developed for track reconstruction, incorporating ACTS seeding and track finding based on the CKF method.
  - Required geometry and material are prepared using various ACTS tools
    - TGeoNode, the volume based description, is translated into Acts::Surface objects by using Acts::TGeoLayerBuilder
    - The materials in the Geant4 simulation are mapped onto different surfaces, and validation shows strong agreement between the materials used in Geant4 and those in ACTS.

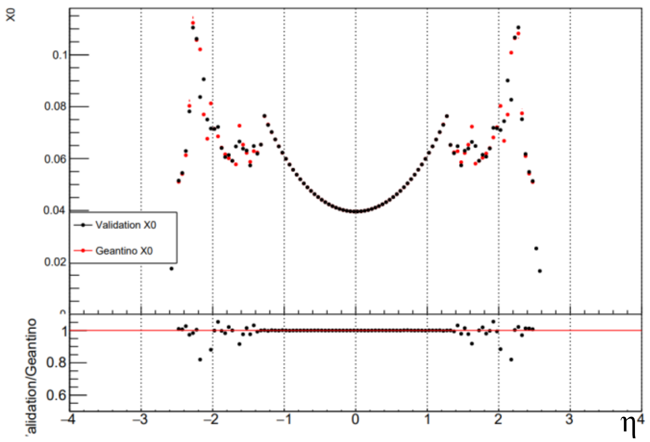


- Track reconstruction performance
  - The tracking performance has been evaluated using 10 GeV single muons, and both resolutions and efficiencies are satisfactor





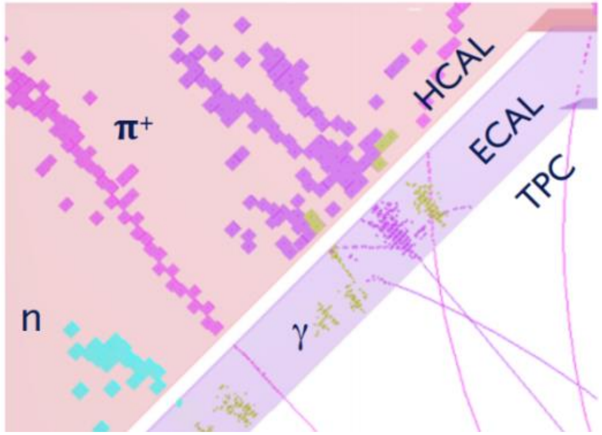




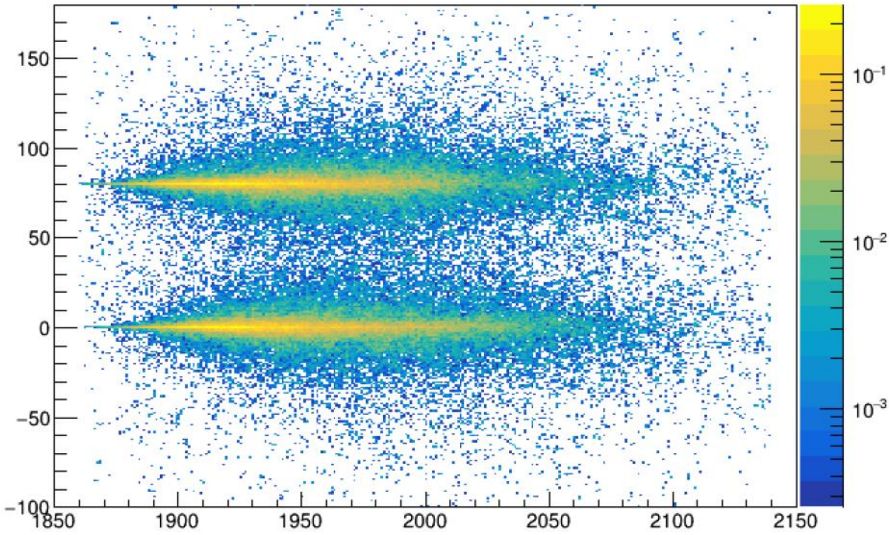
Validation of Material Mapping (VTX + ITK)

PFA Software in Calorimeter (1)

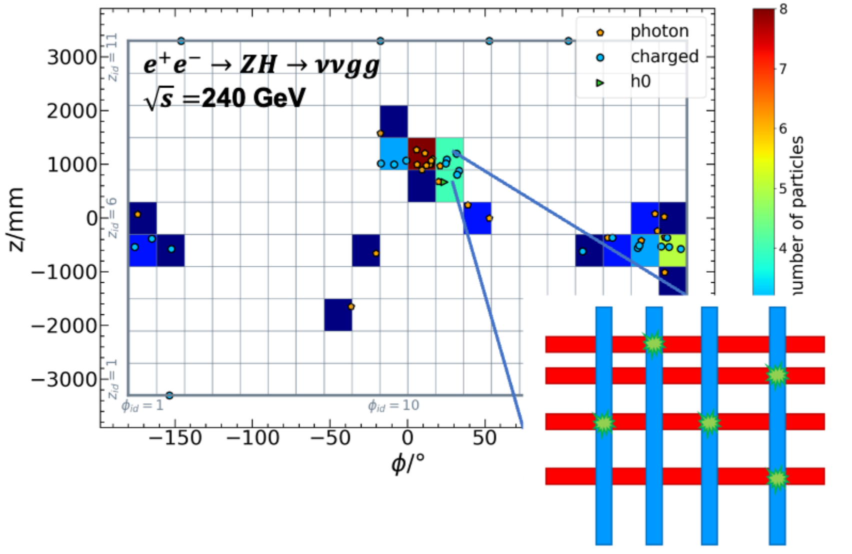
- Particle flow reconstruction for crystal bar ECAL and glass tile HCAL
  - PFA: combined object reconstructed in tracker, ECAL and HCAL
    - Advanced pattern recognition algorithm is required
  - Key issues need to be solved for crystal bar ECAL
    - Overlapping for showers with a large width
    - Ambiguity arising from orthogonally arranged bars



EM shower in BGO ECAL



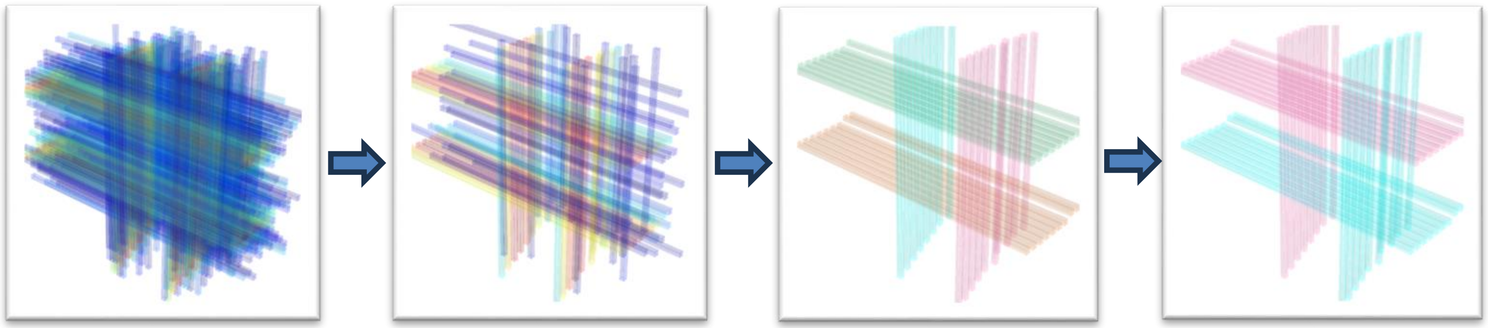
Ambiguity issue in jet event

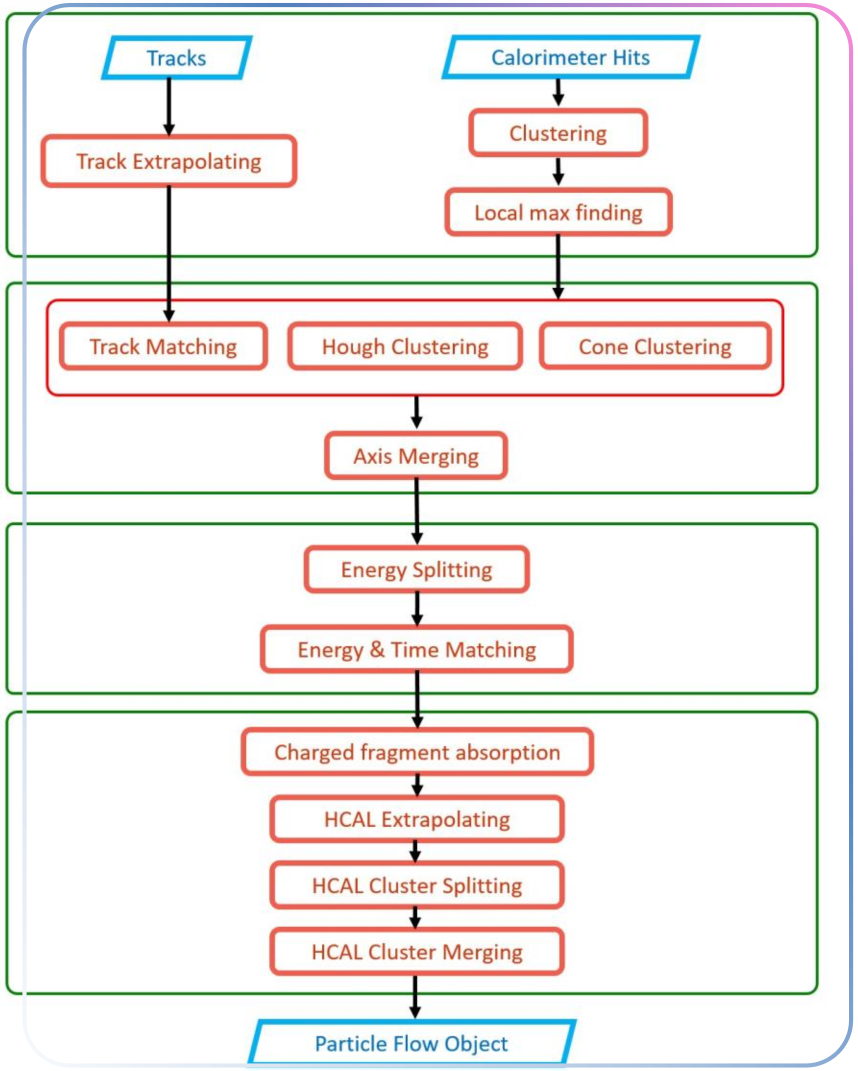


PFA Software in Calorimeter (2)

- The developed CyberPFA algorithm consists of the following steps:
  - Clustering through searching for neighboring hit crystals
  - Identifying the shower core by locating local maxima
  - Shower recognition
    - 3 sub-algorithms are employed for different shower patterns
  - Energy splitting and predicting energy based on the shower profile
  - Removing ambiguity by utilizing timing information, neighbouring showers, and tracks found in inner trackers

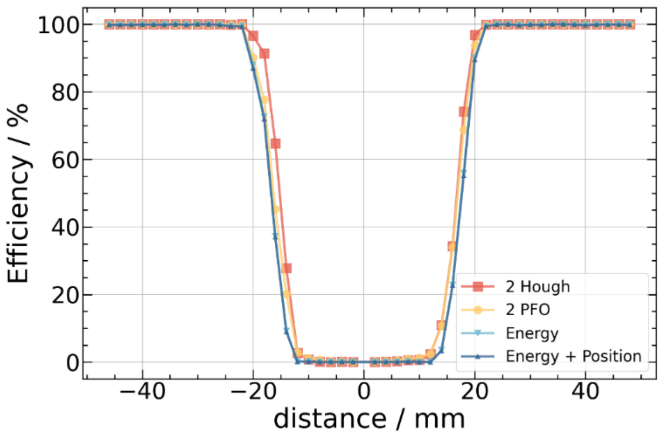
Event display: 2 photons, $E_{\gamma} = 5$ GeV, distance = 15×15 cm.



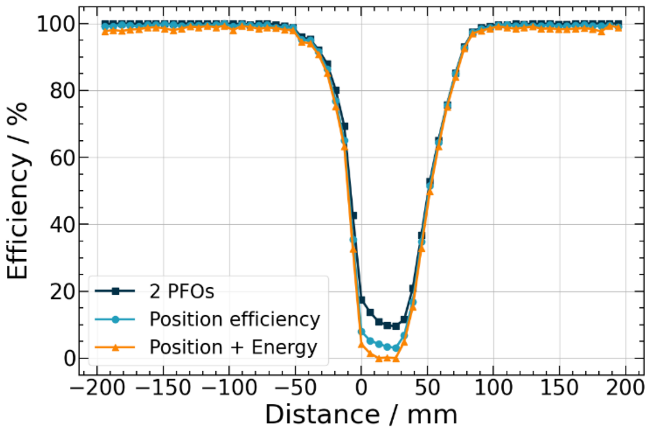


PFA Software in Calorimeter (3)

- Separation of nearby particles

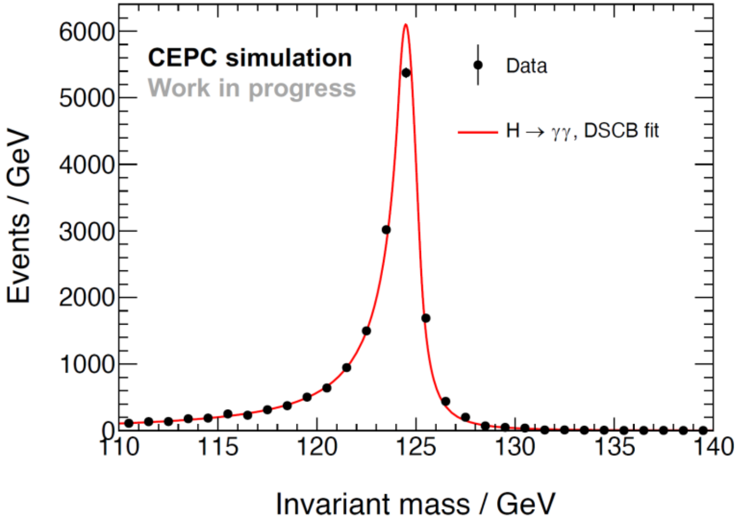


$\gamma - \gamma$ separation:
2.2 cm @ 100% eff.

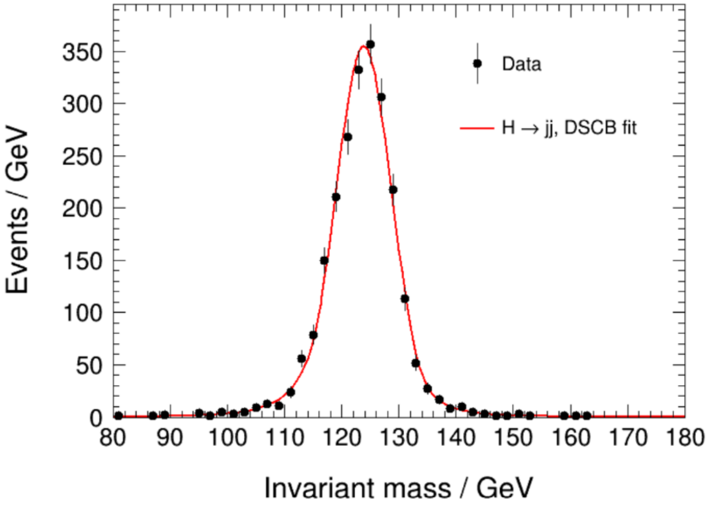


$\gamma - \pi$ separation:
10 cm @ 100% eff.

- Reconstruction of physical processes



$ee \rightarrow ZH \rightarrow \nu\nu\gamma\gamma$:
$\sigma(m_{\gamma\gamma})$ = 0.57 GeV

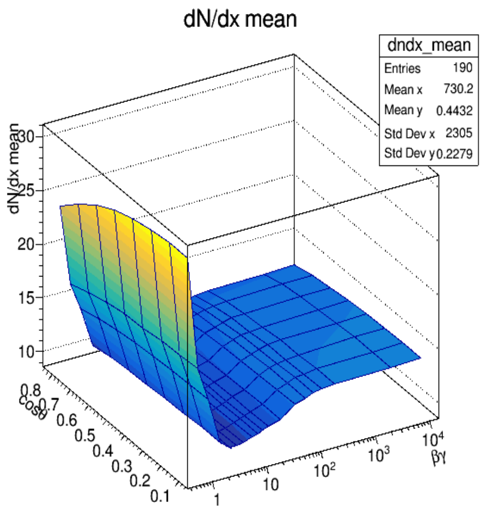


$ee \rightarrow ZH \rightarrow \nu\nu gg$:
Full simulation and reconstruction:
$m_{jj} = 123.81$ GeV,
$\sigma(m_{jj}) = 4.89$ GeV
BMR 3.95%.

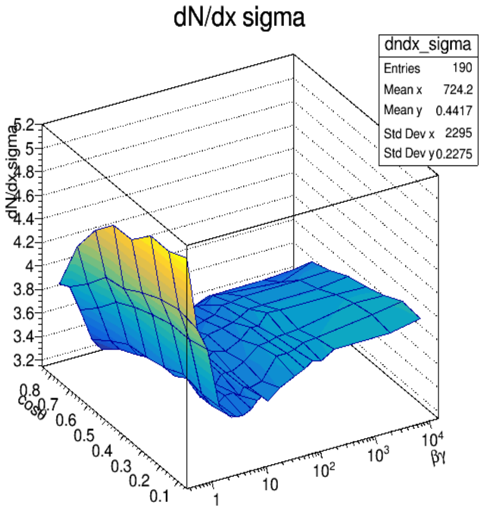
PID Software (1)

- dN/dx in gaseous detectors (TPC and DCH)
  - Track-level dN/dx parameterization model based on sampling according to βγ and cosθ

The parameterization model derived from simulation results using Garfield++



dN/dx mean



dN/dx sigma

Standalone full simulation toolkit

Simulation:

- Complete geometry of the TPC
- Ionizations generated by Garfield++ TrackHeed

Digitization (from experiment):

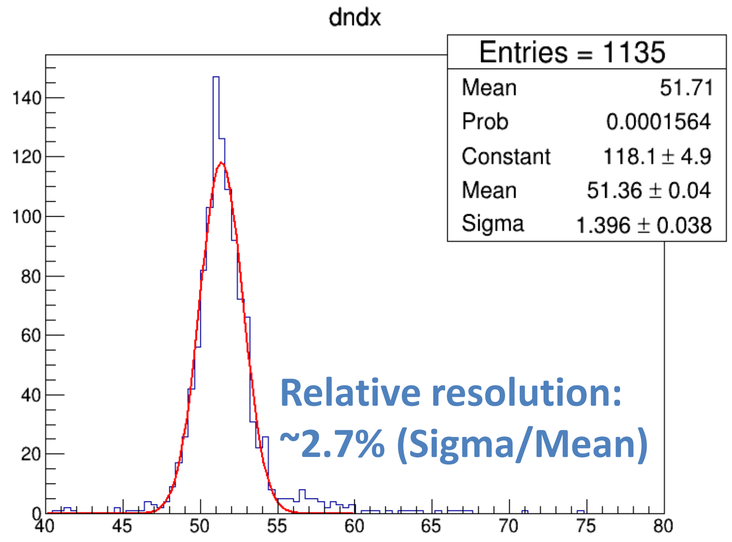
- Electronics noise: 100 electrons
- Amplification:
  - Number of electrons: x2000
  - Signal size in space: 100 um
- Pixelated readout: 500x500 um^2

Reconstruction:

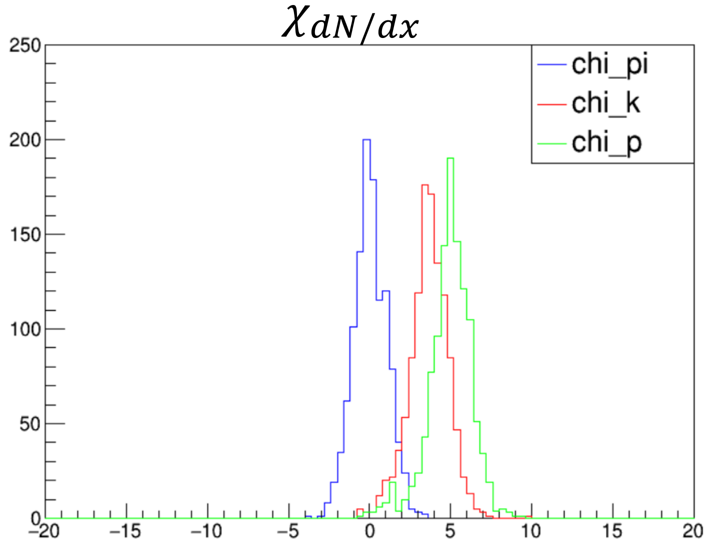
- Reconstruction by counting the number of fired pixels that pass the threshold
- Stable and good linearity

PID Software (2)

dN/dx performance validation with simulated pions
- P = 5 GeV/c
- Theta = 60 deg

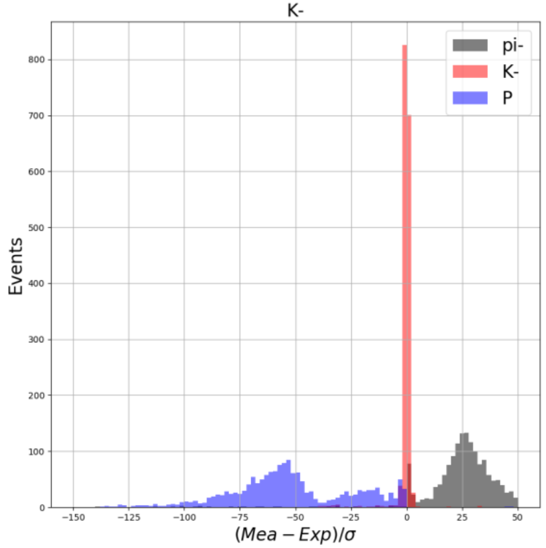


Relative resolution: ~2.7% (Sigma/Mean)



χ with pion (correct) hypothesis is centered around 0

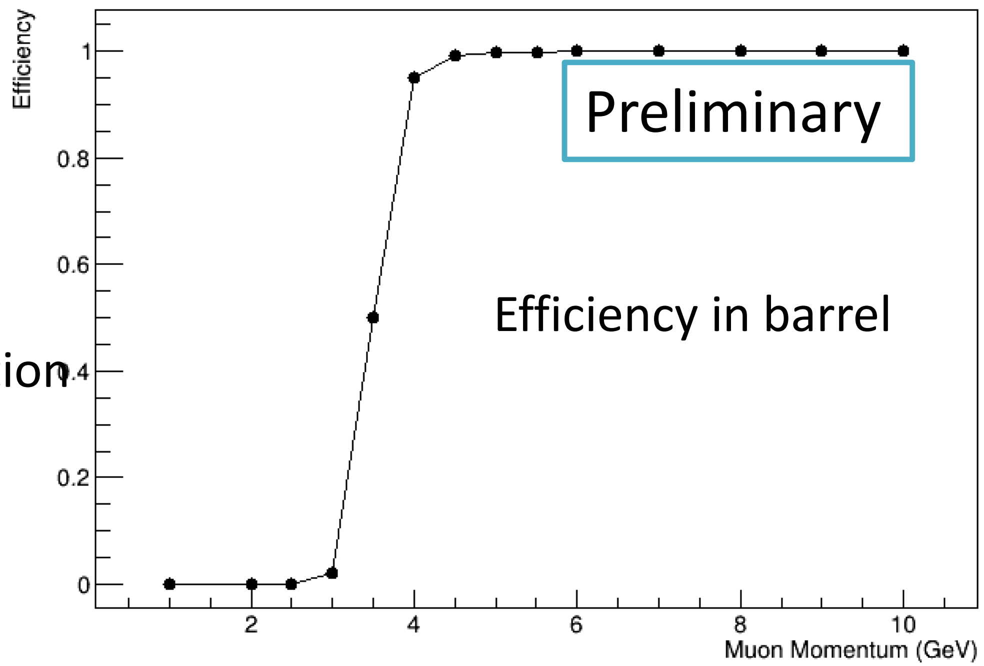
- Time-of-Flight provided by the Outer Tracker
  - Track-level TOF with fixed timing resolution
    - Geometry: R=1850 mm, L/2=2350 mm (To be updated based on the design)
    - TOF: $t_{mea} = tof_{truth} + Gaus(0, t_{bunch} = 20\text{ps}) + Gaus(0, \sigma = 50\text{ps})$



Reconstruction in Muon Detector

- A truth-based Muon ID Algorithm has been implemented
  - Truth-based Muon ID Tag is associated with each "Track" object
- Performance validation of the simulated muons
  - A track is identified as a muon if it has 3 or more Muon Detector hits
- Next steps
  - Improve the truth-based Muon ID algorithm:
    - Allow certain fake rate: e.g. pion mis-id as muon.
    - Add tag to Particle Flow Object
  - Implement Muon ID algorithm based on real reconstruction

Muon id efficiency vs. momentum

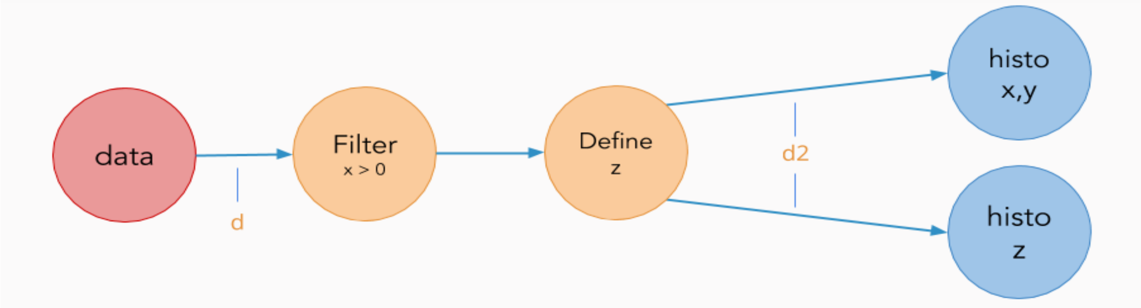


Detector Software Summary

Sub Detector	Options	Detector description/Simulation	Digitization	Reconstruction
MDI+LumiCal		Implemented	None	None
VTX	Ref-Det	Implemented	Smearing	Clusters are formed and then converted into space points. Track finding starts from the most outer layers in the ITK and searches for space points of a track from outside to inside.
VTX	Backup	Cooling, electronics, and part of support structure	Smearing	
ITK		Equivalent material for sensitive detector and support structure		
FTK				
OTK				After adding the OTK hits, track fitting will be executed to produce track parameters.
OTK_PID		Generation of TOF through a parametric model	None	None
TPC	Ref-Det	Implemented	Model based Garfield simulation	Searching for tracks in TPC first and then performing a combined fit to all the hits from both TPC and silicon trackers
TPC_PID		Generation of dEdx (or dN/dx) through a parametric model	None	None
ECAL-Barrel	Ref-Det	Materials and geometry from the preliminary design	Model based on testbeam data	New PFA algorithm
ECAL-Endcap				Being validated with MC data
HCAL-Barrel			Parametric model	Being developed
HCAL-Endcap				
MUON-Barrel		Added materials and geometry	Model based on lab measurements	Being developed
MUON-Endcap				

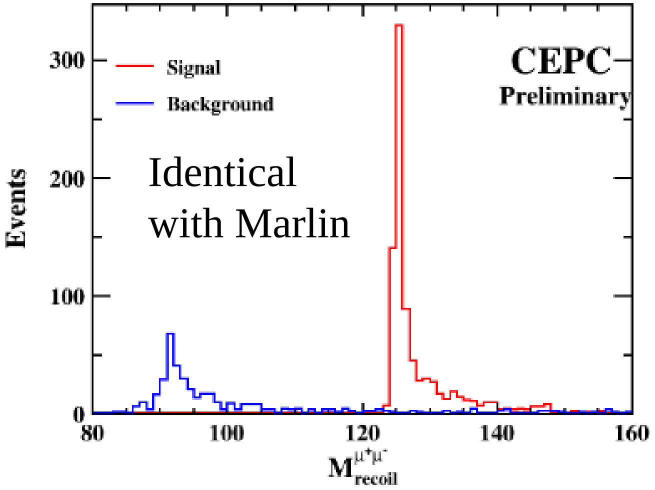
Analysis Software Framework

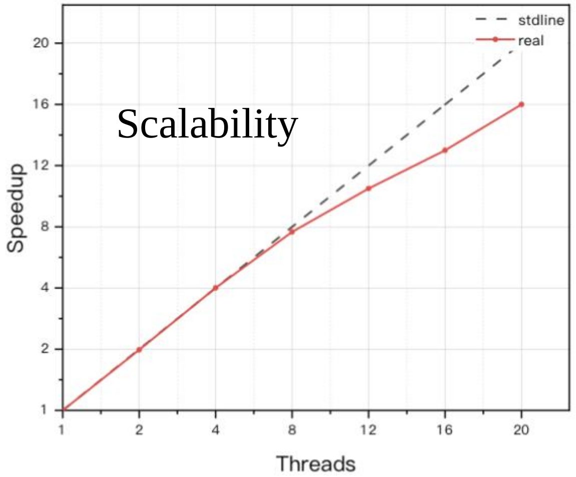
- RDataFrame is a powerful analysis framework for paralleled data analysis
  - Program language: Python and C++
  - Supports declarative programming and parallel processing
  - Used by many experiments like FCC-ee experiment




```cpp
// d2 is a new data-frame, a transformed version of d
auto d2 = d.Filter("x > 0")
           .Define("z","x*x + y*y");
// make histograms out of it
auto hz = d2.Histo1D("z");
auto hxy = d2.Histo2D({"hxy","hxy",16,-1,1,64,-1,1},"x","y");
```

- Software developments
  - Developed common data input interfaces to support both LCIO data and EDM4hep data
  - Several algorithms were ported from Marlin
    - JetClustering , KinematicFit
  - More are being implemented
    - VertexFit, JetTagging, PID etc.
  - Performance tests with different physics channels
    - e+e- -> Z(mumu)H
    - e+e- ->H(2jet) mumu





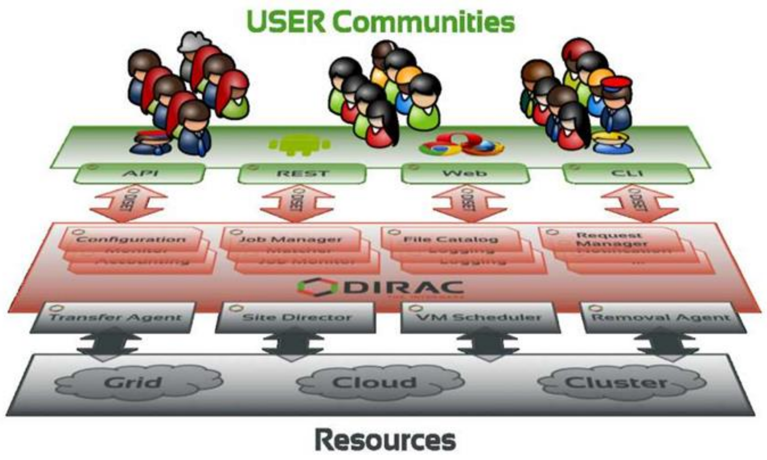
CEPC Computing (1)

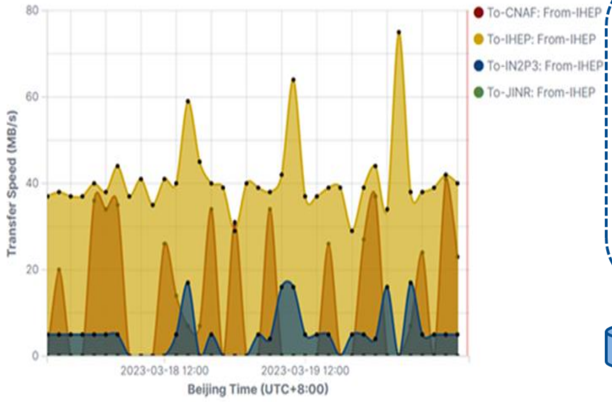
- Technical challenges
  - Management of diverse hardware needed by new HEP applications
  - Increasing complexity in data management caused by large data volume
  - More complicated workflow for data transfer and job scheduling
  - Growing complications in using opportunistic resources efficiently

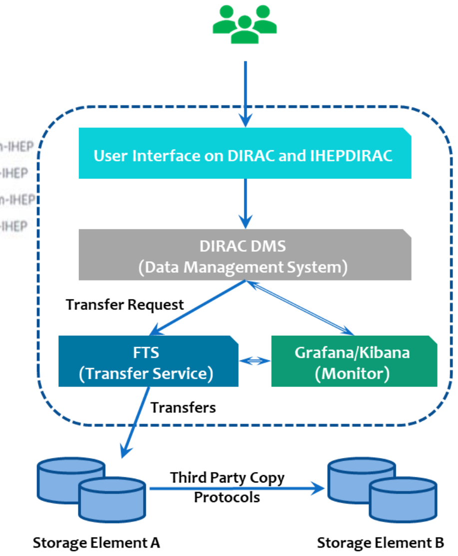
- Technology survey and our choices
  - DIRAC chosen as the distributed computing middleware
    - Having strong capabilities proven by other HEP experiments including LHCb and Belle II
    - Being able to integrate heterogeneous computing resource, e.g. Grid, HTC Cloud and supercomputer
    - DIRAC consortium provides a platform to share experiences with other experiments
  - Rucio chosen as the data management system
    - Providing services and associated libraries for managing large volumes of data
    - Used by many experiments and allows adding experiment-specific customisations
    - Playing an important role in the Data Lake prototype being developed for HL-LHC

CEPC Computing (2)

- Requirements
  - Data volume and computing resources will be estimated with full detector simulation
- Computing Model
  - Tier 0: IHEP serves as the primary hub for computing and data processing
  - Tier 1: Large sites act as major support centers with strong capabilities of computing and storage system
  - Tier 2: Smaller sites provide supplementary computing power and storage capabilities
- A DIRAC-based platform established
  - IHEP holds 2,000 dedicated cores (including 640 cores shared with ILC) and other sites contribute the rest 2600 cores
  - The prototype for workload and data management was developed
  - Regular network test was established and the monitoring system was implemented with ElasticResearch+Kibana/Grafana







Research Team and Working Plan

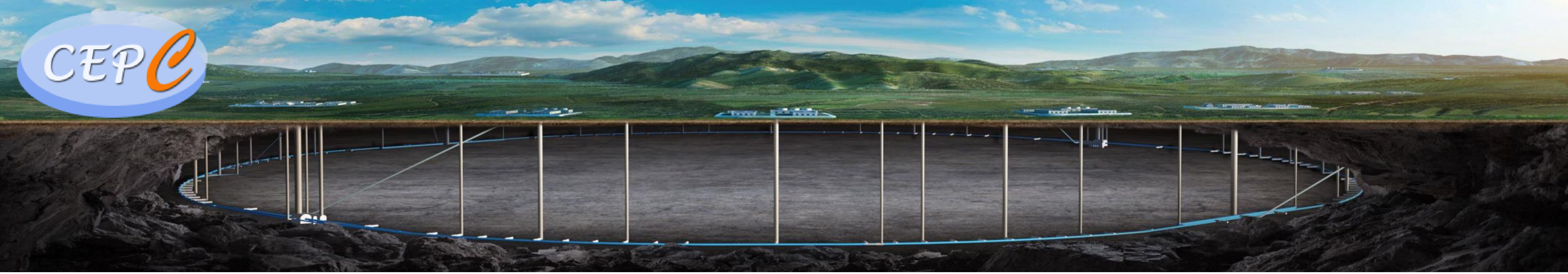
- Software
  - Institutes: IHEP, SDU, USTC, SYSU, FDU, SCNU, ZZU
  - Manpower: 20 staff members, 2 Postdocs, 4 PhD students, 3 MSc students
  - Work plan
    - Application of multithreading with Gaudi and adding the support of data-parallel computing
    - Implementing the simulation chain with Gaussino in collaboration with the Key4hep team
    - Application of ACTS to silicon trackers and investigating its possible extension to gaseous detectors
    - Further development of calorimeter and PID software to validate the performance of the reference detector
    - Implementation of detector visualization and event display tool with Phoenix
- Computing
  - Institutes: IHEP
  - Manpower: 8 staff members
  - Work plan
    - Moving towards a token-based authorization and authentication infrastructure (AAI)
    - Migration to DiracX to enhance system performance
    - Providing support for Machine Learning and complex workflows

International Collaboration

- IHEP and SDU are non-EU members of Key4hep project (AIDAinnova WP12)
  - 7 staff members, 10 proceeding papers (CHEP/ICHEP/ACAT), 10 reports presented at the WP12 meetings
  - One of the earliest users of Key4hep and also contributed to software development: Extension of Edm4hep, Application of Gaussino Framework, Integration of Pandora with Key4hep, Automated performance validation, etc.
  - In 2022, a performance validation cluster with 640 CPU cores was established at IHEP, shared by the CEPC and ILC experiments
- Gaseous Detector Software
  - 3 staff members collaborated with the IDEA detector group on the R&D of the drift chamber
  - Journal paper: Peak finding algorithm for cluster counting with domain adaptation (Computer Physics Communications 300 (2024) 109208)
- Collaboration with the ACTS group
  - 3 staff members
  - Implementing precise simulation of drift chamber for the Open Data Detector [ACAT2024, CHEP2024]
  - Implementation of track reconstruction in silicon trackers with ACTS/TRACCC [CHEP2024]
  - Organized an international workshop in Zhengzhou: "Workshop on Tracking in Particle Physics Experiments" , held from May 16-19, 2024.
- IHEP is an official member of the DIRAC consortium
  - 2 staff members
  - Mainly work on integration of cloud resources, support of multi-core jobs, evaluation of DiracX
  - Presentations at CHEP2023/ACAT2024/CHEP2024 and DIRAC workshops in 2021, 2022, 2023 and 2024

Summary

- The CEPCSW is being developed to
  - support detector R&D and physics potential studies
  - build the offline system of the experiment
- A Dirac-based distributed computing platform has been deployed and is currently evolving
- The chapters of offline software and computing of the RefDet TDR
  - being prepared



Thank you for your attention!

中国科学院高能物理研究所
Institute of High Energy Physics
Chinese Academy of Sciences

Backup

RefDet TDR: Software Chapter

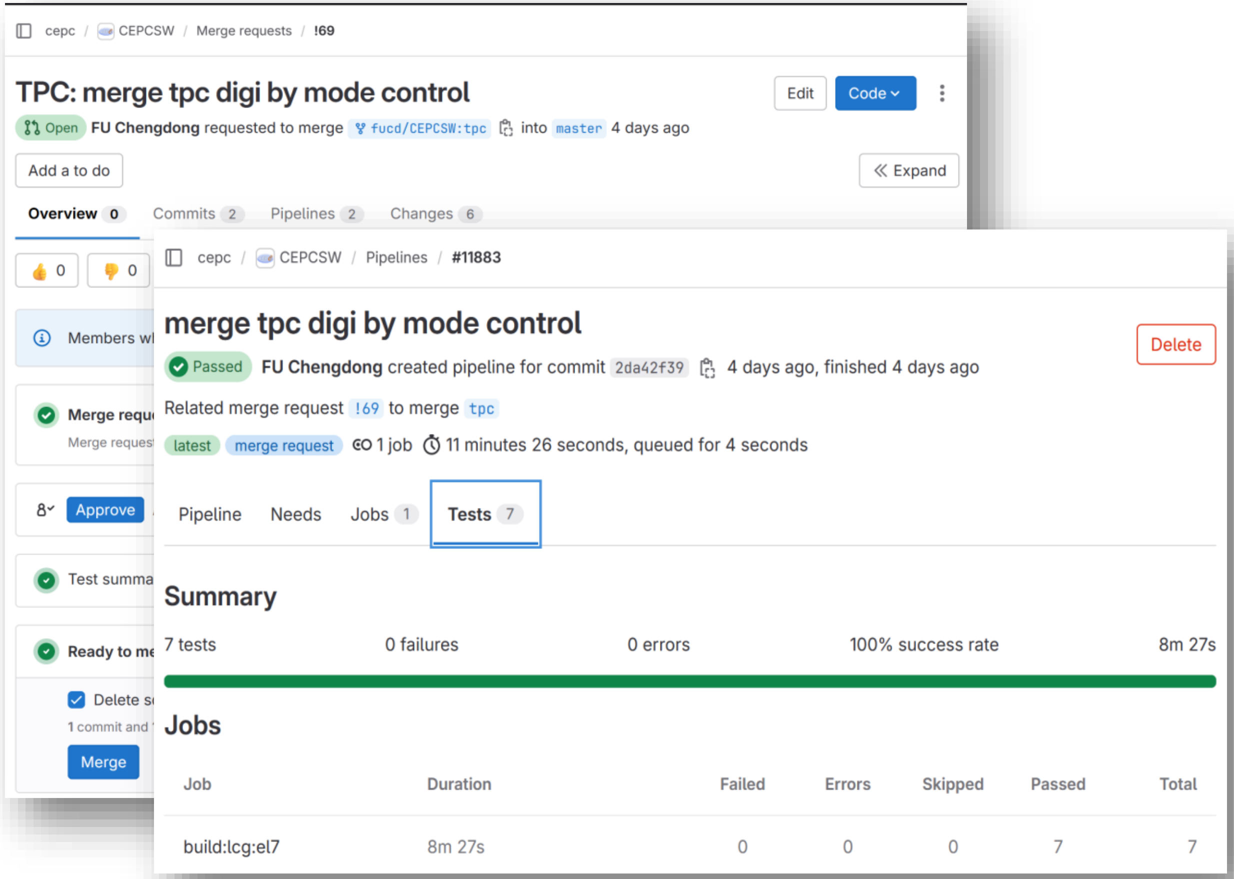
- Introduction
- Core software
  - Software Architecture
  - Integration with Key4hep
  - Data management
  - Control flow
- Simulation
  - Generator
  - Simulation framework
  - Fast simulation
- Reconstruction algorithms
  - Tracking algorithms
  - PID Algorithms
  - Particle Flow Algorithms
- Analysis software
  - RDataframe-based framework
  - Analysis tools
- Detector visualization and event display
- R&D activities
  - Simulation with Gaussino
  - Implementation of ACTS/TRACCC
  - Applications of machine learning
  - Etc.
- Software development environment
  - Development process
  - Performance validation

RefDet TDR: Computing Chapter

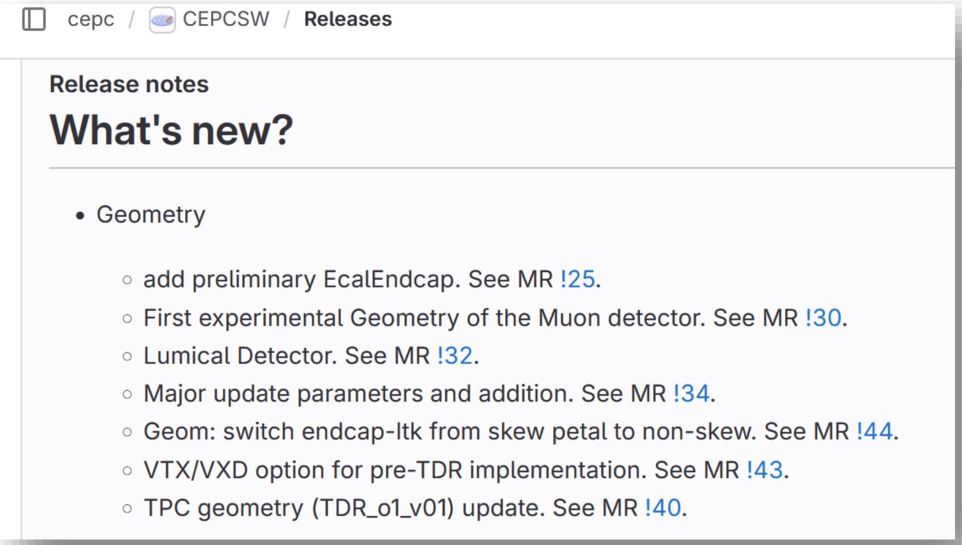
- Requirements
  - Resource estimation
- Design
  - Data processing flow
  - Computing model
- Computing system
  - Distributed computing
  - High throughput computing
  - High performance computing
- Storage system
  - Data management
  - Disk/Tape system
- Networking
- Cyber Security
  - Authentication and authorization
  - Security monitoring
- System monitoring
  - Monitoring platform

Software Development Environment

- "Fork - Merge Request" workflow
  - IHEP GitLab: https://code.ihep.ac.cn/cepc/CEPCSW
  - Unit test with "ctest" in CI/CD.



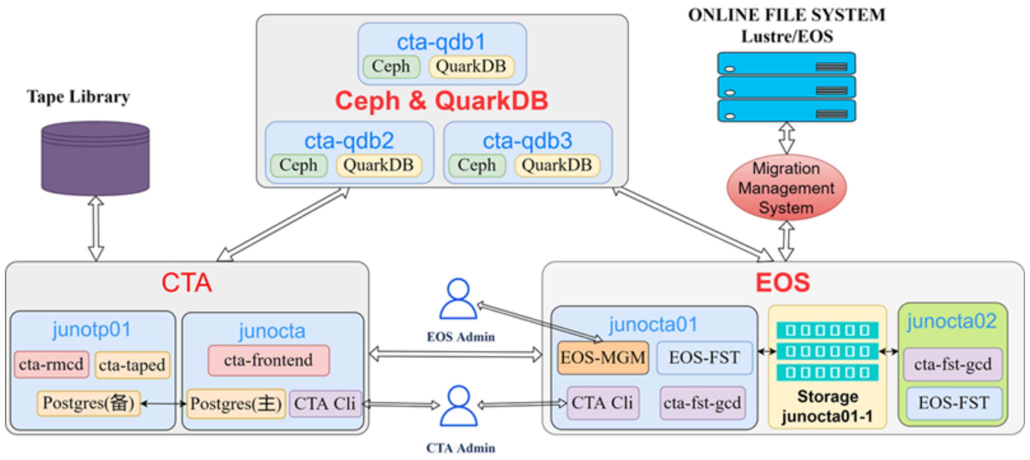
- "Release early, release often"
  - To support fast iteration of detector design
- New "time-based" version numbering scheme for TDR: tdrYY.MM.NN
  - E.g., tdr24.5.0: the first release in May 2024.
  - Release notes: https://code.ihep.ac.cn/cepc/CEPCSW/-/releases

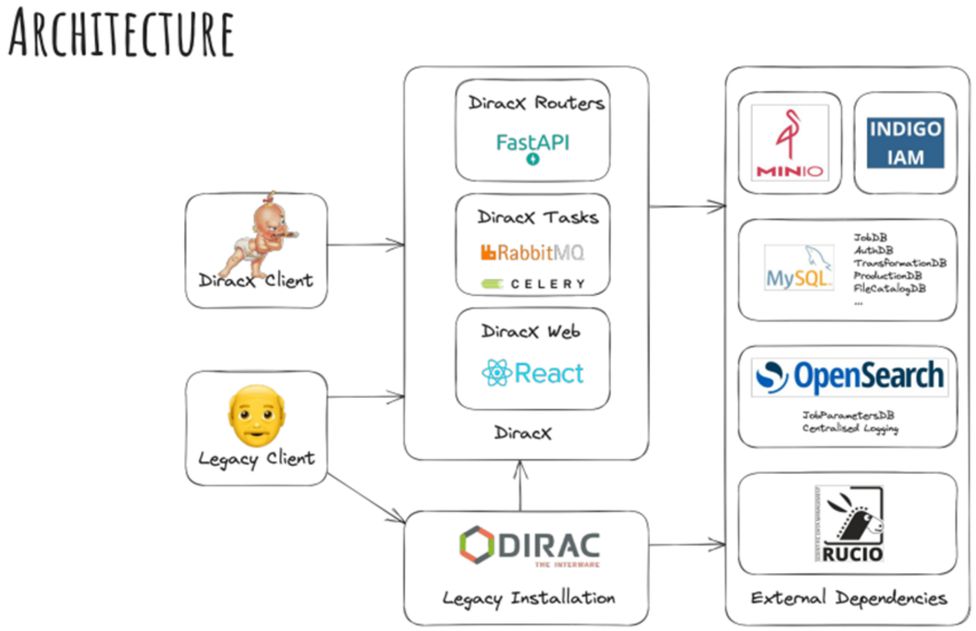


CEPC Computing (3)

Four data storage systems were deployed at IHEP

- Lustre: an open source parallel file system for industry
  - Adopted by more than 70% of the world's supercomputing centers
  - IHEP and GSI are the largest units deploying LUSTRE in the field of high energy physics
- EOS: open source storage system
  - Based on XRootD protocol, with good scalability and performance
  - IHEP is one of the largest EOS deployment site, approaching 100PB
- CTA: An open source tape library storage system
  - Based on the previous generation system CASTOR
  - It supports EOS/ dCache disk storage system
- CVMFS: software storage system
  - 6 warehouses, 14 terabytes
  - Deploy software repositories for high energy physics experiments





Work in progress

- Evaluating “Data Lake” model with Rucio and XCache
- Optimizing the performance of data access