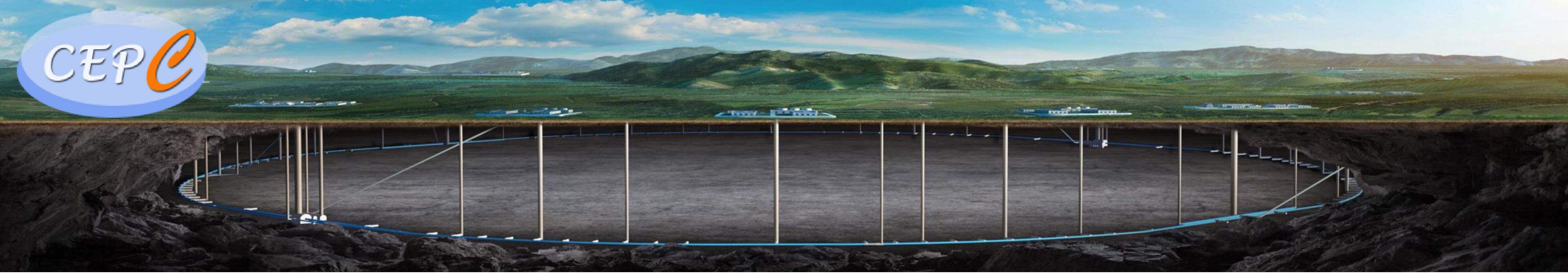


# CEPC Software and Computing

Weidong Li (IHEP)

representing CEPC software and computing teams

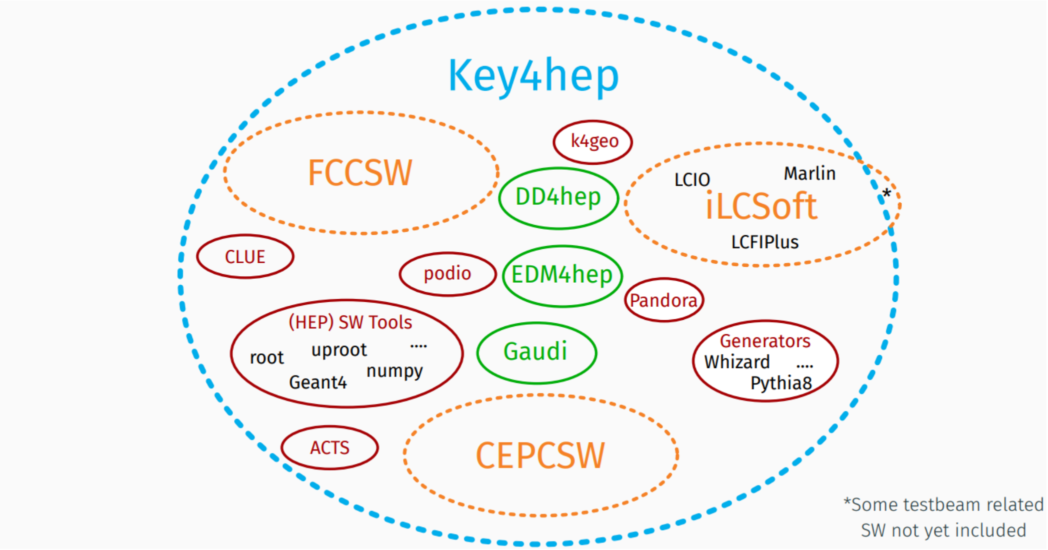
中国科学院高能物理研究所
Institute of High Energy Physics
Chinese Academy of Sciences

Aug. 7th, 2024, CEPC Detector Ref-TDR Review

# Content

# 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



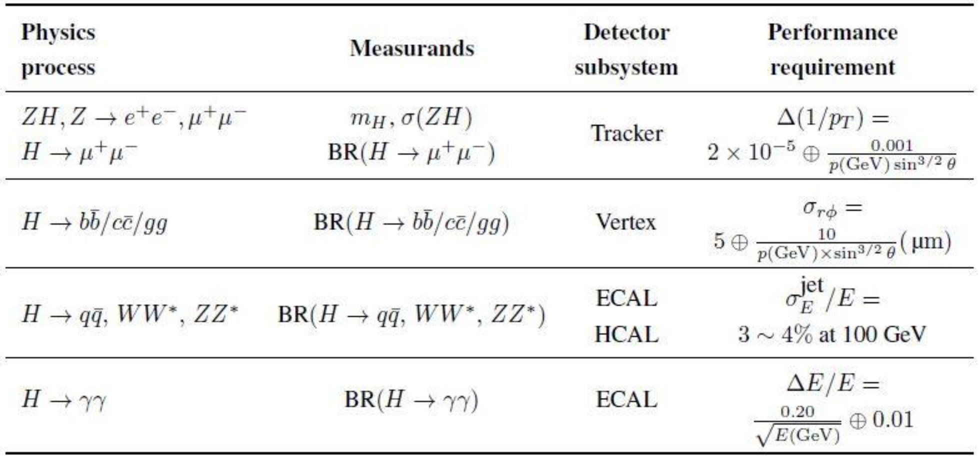
*Some testbeam related SW not yet included

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 experiment

# Requirements

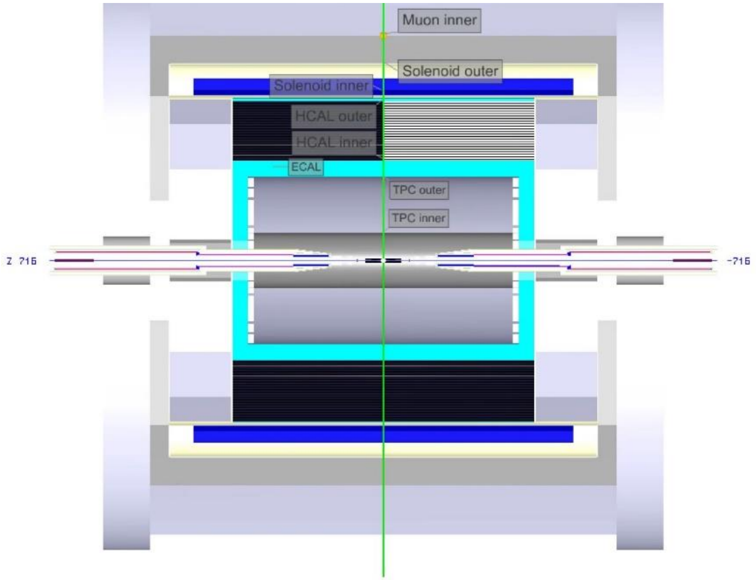
- Physics requirements
  - High tracking efficiency (~100%), momentum resolution (<0.1%), PID ($2\sigma$ p/K separation at P < ~ 20 GeV/c), BMR better than 4% etc.
- CEPC Reference Detector
  - VTX: Vertex Detector
  - ITK/OTK: Inner Silicon Tracker/Outer Silicon Tracker
  - TPC: Time Projection Chamber
  - 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 |
|---|---|---|---|
| $ZH, Z \to e^+e^-, \mu^+\mu^-$<br>$H \to \mu^+\mu^-$ | $m_H, \sigma(ZH)$<br>$\mathrm{BR}(H \to \mu^+\mu^-)$ | 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<br>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



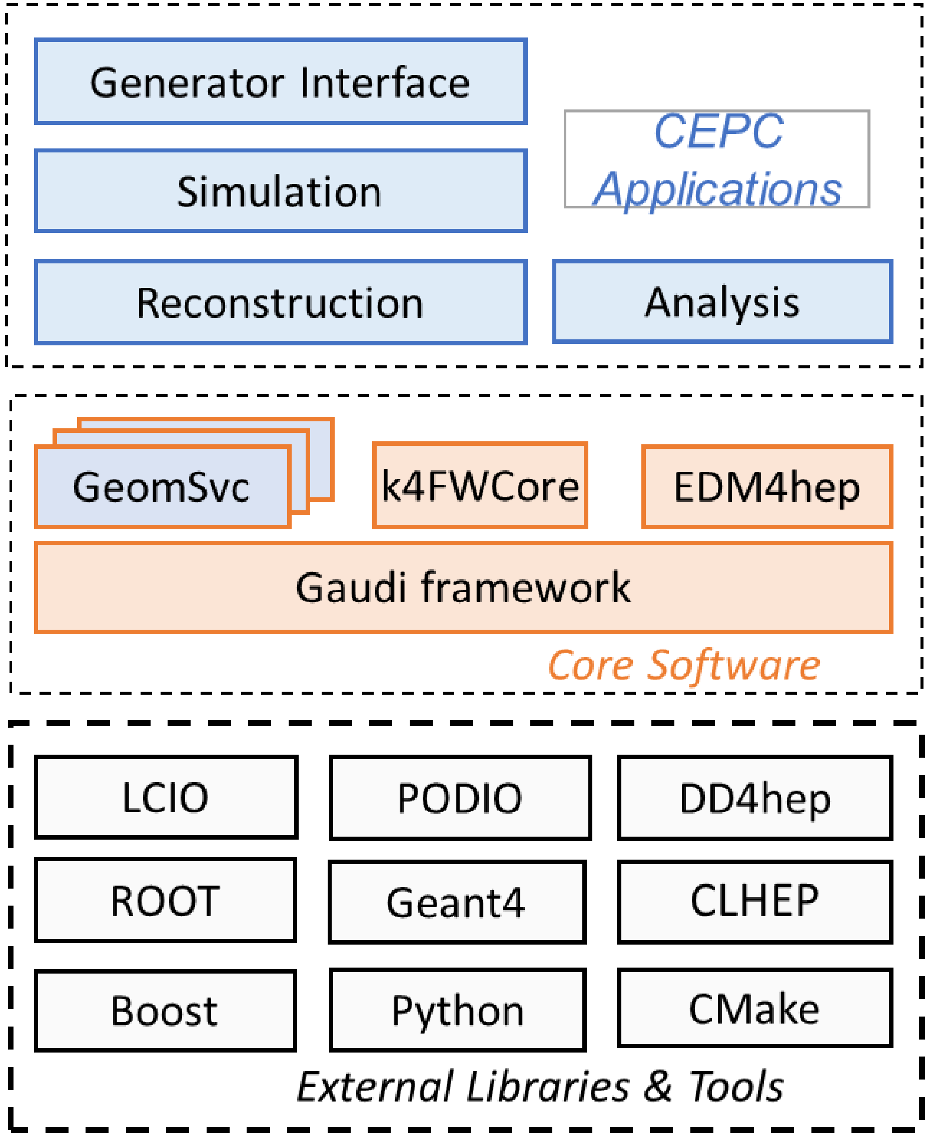
# Technical Options

- Main technical challenges
  - How to develop large scale software which can be run on diverse hardware 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 our choices
  - Some important HEP software packages are chosen 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

# Software: Architecture of CEPCSW

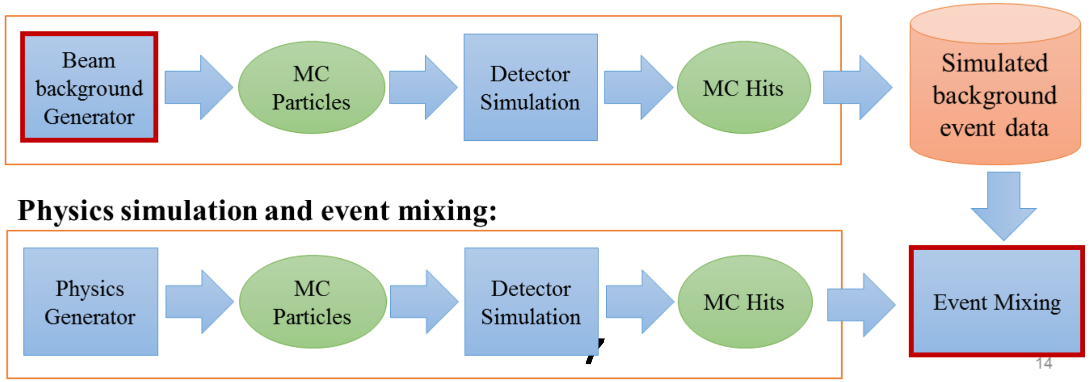
- CEPCSW is organized as a multi-layer structure
  - Applications
    - simulation, reconstruction and analysis
  - Core software
  - External libraries

- 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.



# Software: Simulation Framework

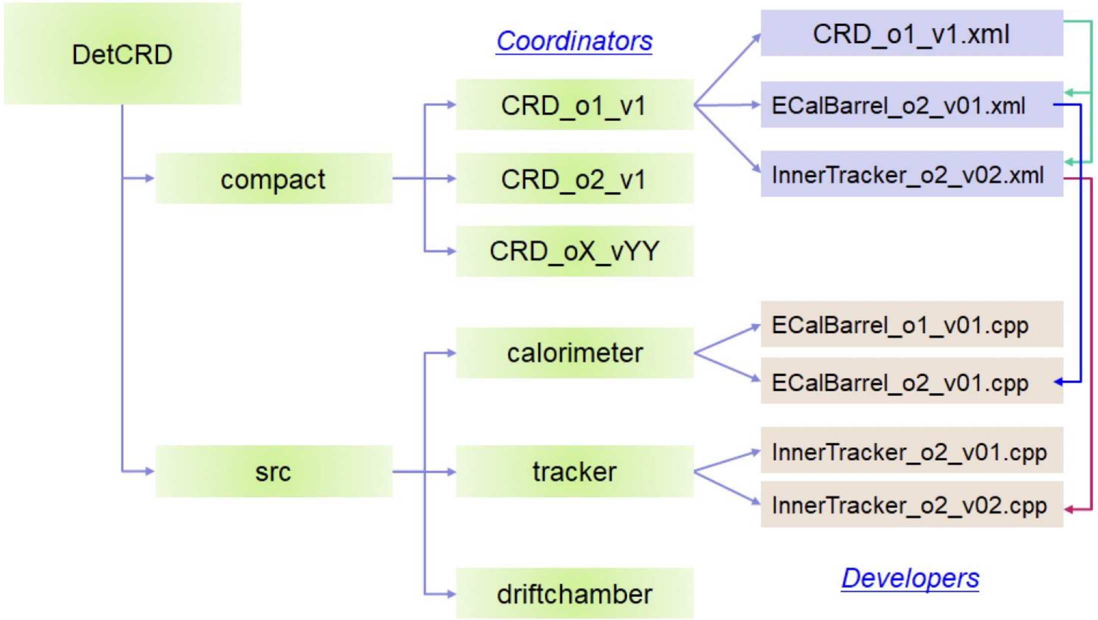
- The simulation framework was developed
  - based on which the simulation chain for sub-detectors was built
- The region-based fast simulation interface is available
  - to integrate different fast simulation modules into the simulation chain
- An event mixing tool was provided
  - to mix different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


Beam background Generator
MC Particles
Detector Simulation
MC Hits
Simulated background event data

Physics simulation and event mixing:

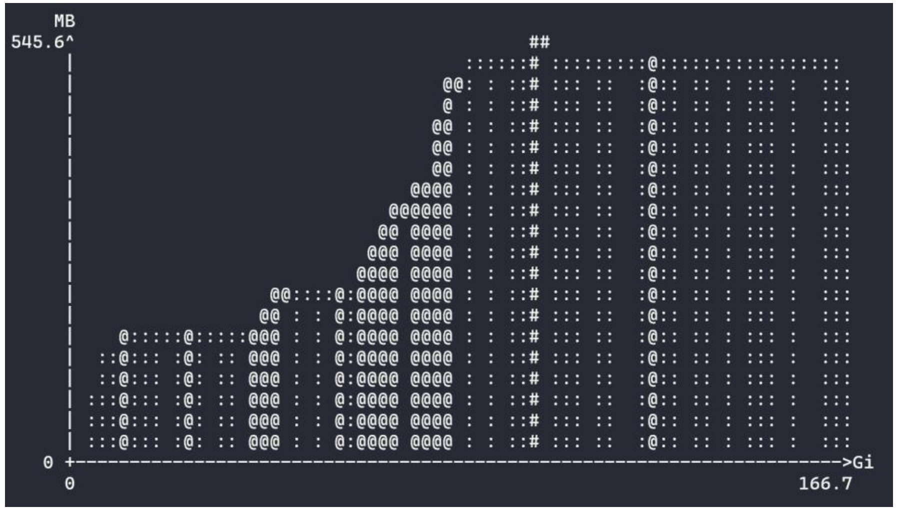
Physics Generator
MC Particles
Detector Simulation
MC Hits
Event Mixing

- 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


DetCRD
compact
src
*Coordinators*
CRD_o1_v1
CRD_o2_v1
CRD_oX_vYY
CRD_o1_v1.xml
ECalBarrel_o2_v01.xml
InnerTracker_o2_v02.xml
calorimeter
tracker
driftchamber
ECalBarrel_o1_v01.cpp
ECalBarrel_o2_v01.cpp
InnerTracker_o2_v01.cpp
InnerTracker_o2_v02.cpp
*Developers*

# Software: Simulation R&D

- Multi-threading mandatory
  - Reduce memory footprint



```
    MB
545.6^                                           ##
     |                                     ::::::# :::::::::@:::::::::::::::::
     |                                   @@: : ::# ::: ::  :@:: :: : ::: :  :::
     |                                   @ : : ::# ::: ::  :@:: :: : ::: :  :::
     |                                  @@ : : ::# ::: ::  :@:: :: : ::: :  :::
     |                                  @@ : : ::# ::: ::  :@:: :: : ::: :  :::
     |                                  @@ : : ::# ::: ::  :@:: :: : ::: :  :::
     |                                @@@@ : : ::# ::: ::  :@:: :: : ::: :  :::
     |                              @@@@@@ : : ::# ::: ::  :@:: :: : ::: :  :::
     |                             @@ @@@@ : : ::# ::: ::  :@:: :: : ::: :  :::
     |                            @@@ @@@@ : : ::# ::: ::  :@:: :: : ::: :  :::
     |                           @@@@ @@@@ : : ::# ::: ::  :@:: :: : ::: :  :::
     |                   @@::::@:@@@@ @@@@ : : ::# ::: ::  :@:: :: : ::: :  :::
     |                  @@ : : @:@@@@ @@@@ : : ::# ::: ::  :@:: :: : ::: :  :::
     |     @:::::@:::::@@@ : : @:@@@@ @@@@ : : ::# ::: ::  :@:: :: : ::: :  :::
     |   ::@::: :@: :: @@@ : : @:@@@@ @@@@ : : ::# ::: ::  :@:: :: : ::: :  :::
     |   ::@::: :@: :: @@@ : : @:@@@@ @@@@ : : ::# ::: ::  :@:: :: : ::: :  :::
     |  :::@::: :@: :: @@@ : : @:@@@@ @@@@ : : ::# ::: ::  :@:: :: : ::: :  :::
     |  :::@::: :@: :: @@@ : : @:@@@@ @@@@ : : ::# ::: ::  :@:: :: : ::: :  :::
     |  :::@::: :@: :: @@@ : : @:@@@@ @@@@ : : ::# ::: ::  :@:: :: : ::: :  :::
   0 +------------------------------------------------------------------------>Gi
     0                                                                    166.7
```

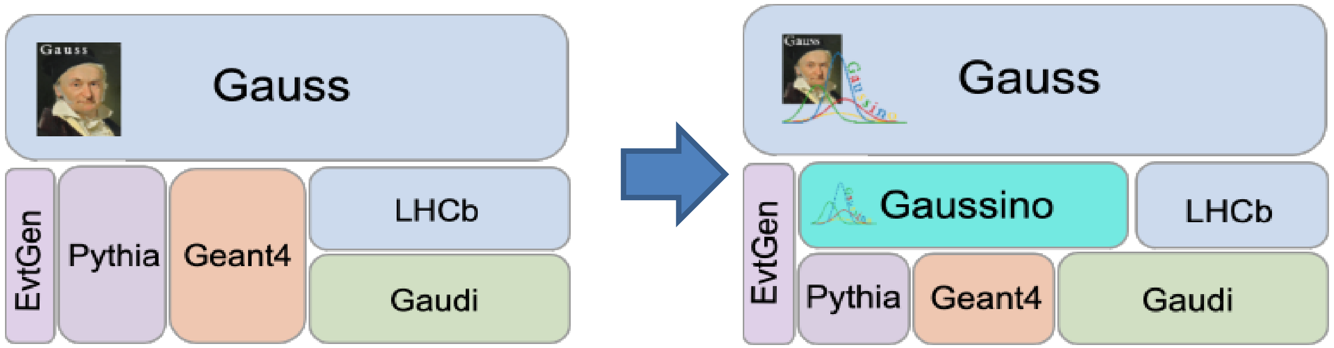
**Heap memory usage (serial simulation)**

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]

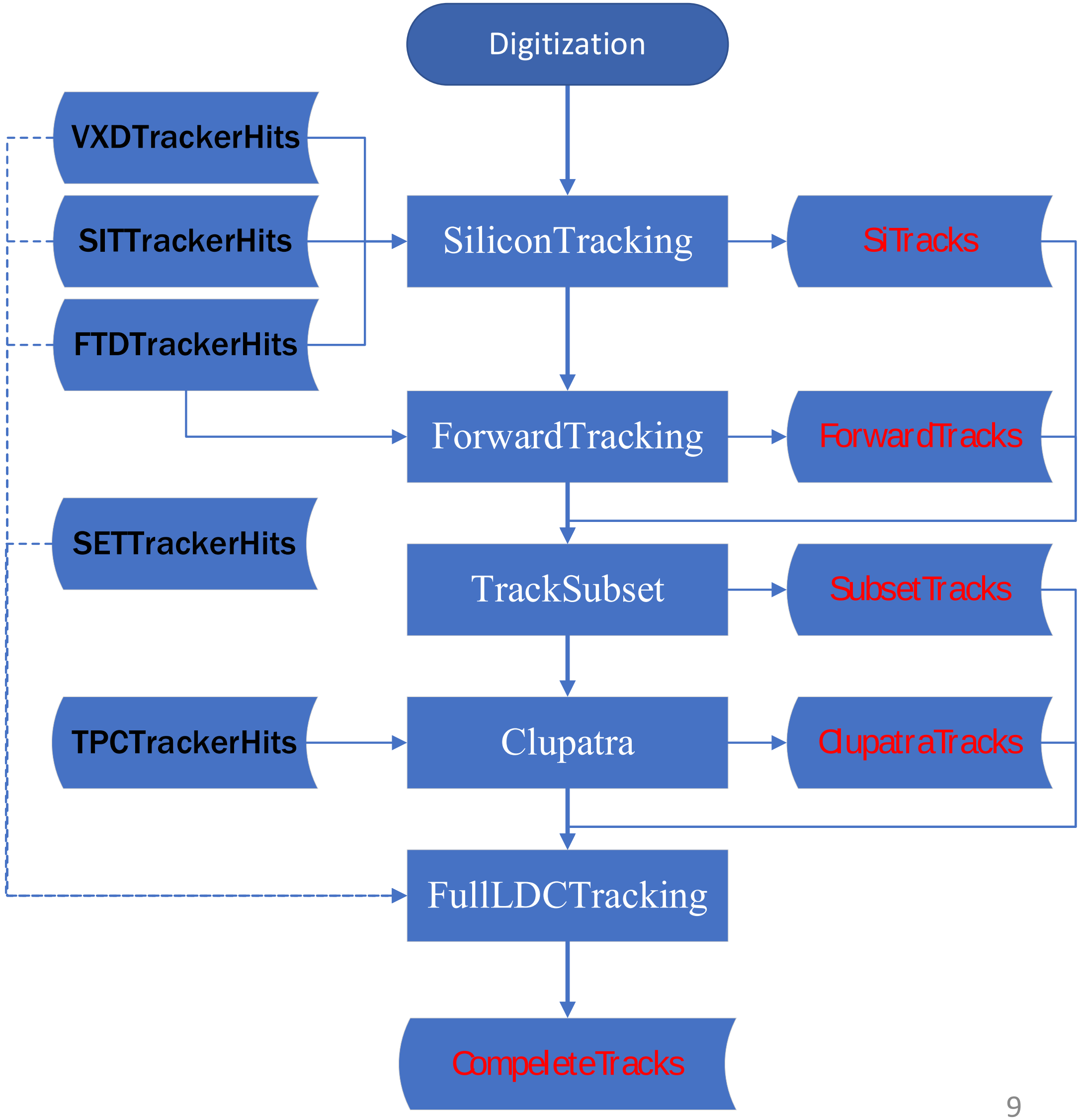


- Implementation of  CEPC-on-Gaussino prototype
  - Cooperated with the software experts from CERN
  - Removed dependencies on the LHCb software stack
  - As a demo, VXD simulation was implemented

ICHEP 2024 Parallel Talk by Tao Lin

# Software: Tracking

- 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



Digitization

VXDTrackerHits

SITTrackerHits

FTDTrackerHits

SETTrackerHits

TPCTrackerHits

SiliconTracking

ForwardTracking

TrackSubset

Clupatra

FullLDCTracking

SiTracks

ForwardTracks

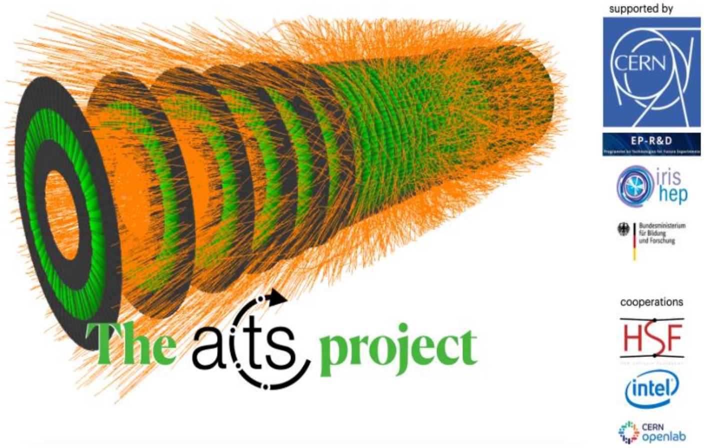
SubsetTracks

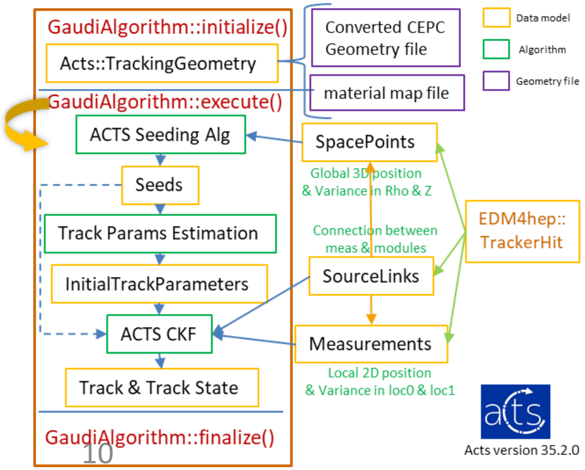
ClupatraTracks

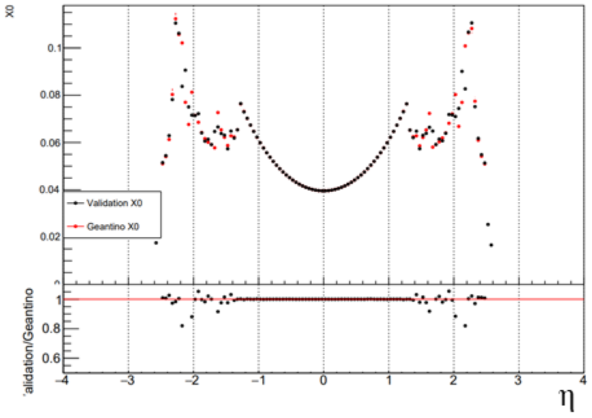
CompeleteTracks

# Software: Tracking R&D

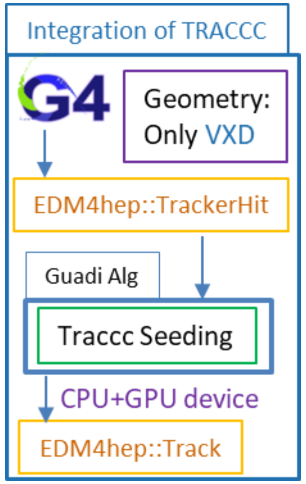
- ACTS is an experiment-independent toolkit for  track reconstruction
  - High-performance parallel software for complex hardware (CPU/GPU/…)
- Applying ACTS to track reconstruction in CEPC silicon trackers
  - Convert the CEPCSW’s geometry to the ACTS’s format
  - Map Geant4 material to the ACTS’s material
- 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

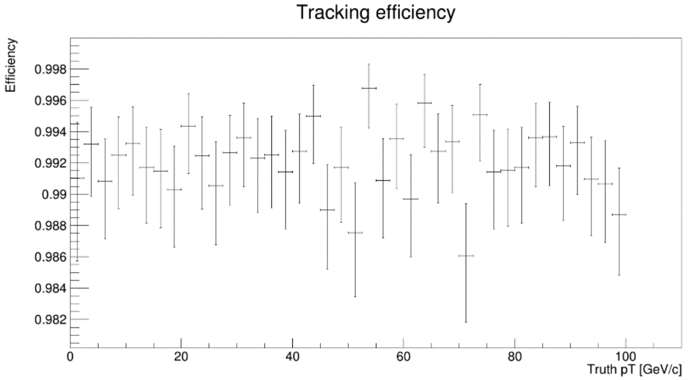






Validation of Material Mapping

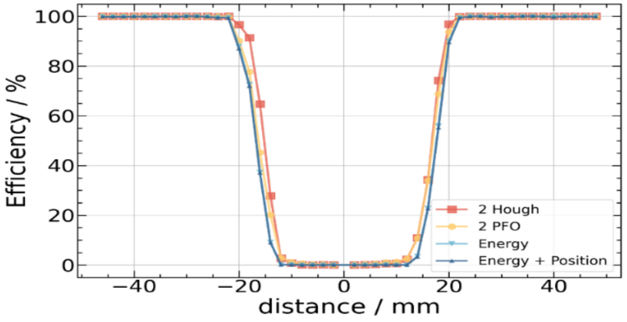




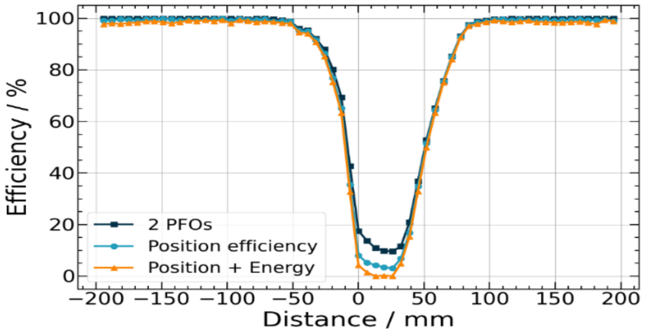
ACTS Tracking efficiency with pT

# Software: Crystal-bar ECAL

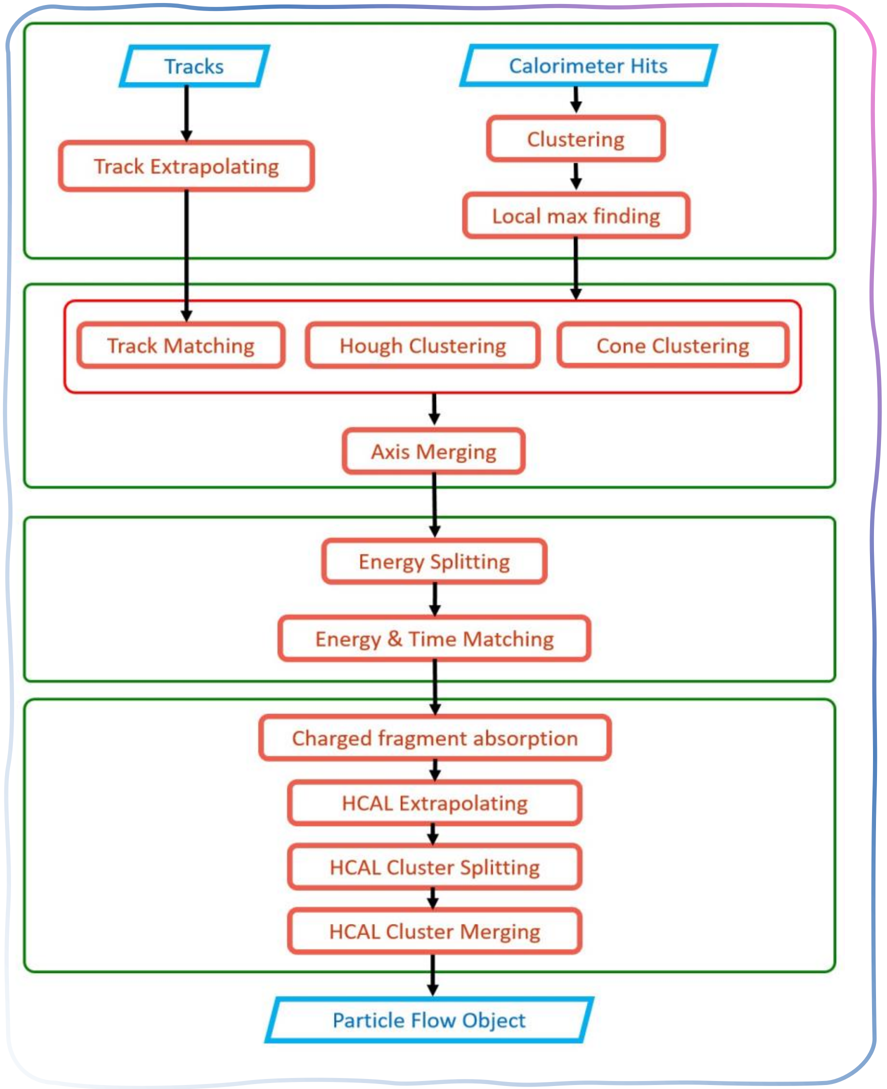
- Development of a new particle flow algorithm for crystal-bar ECAL
  - Clustering
  - Shower recognition
  - Energy splitting and ambiguity removal
  - Re-clustering
- Reconstruction performance
  - Separation capability of two particles
    - $\gamma - \gamma$ separation: 2.2 cm @ 100% efficiency
    - $\gamma - \pi$ separation: 10 cm @ 100% efficiency



**$\gamma - \gamma$ separation for 5 GeV photons**



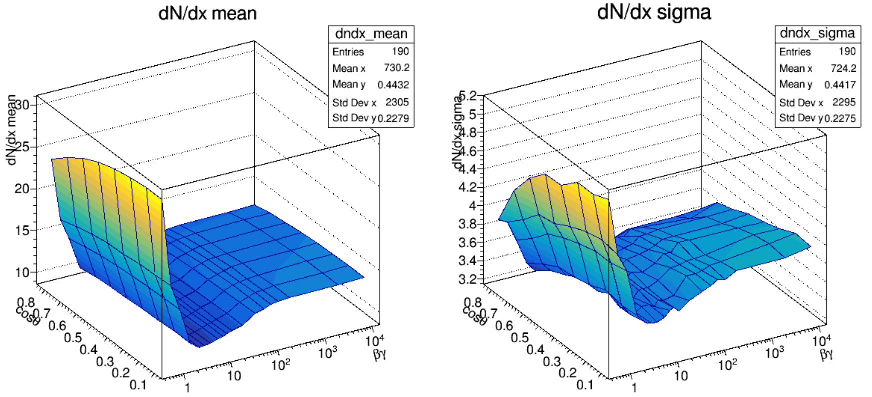
**$\gamma - \pi$ separation for 5 GeV $\gamma$ and $\pi^-$**


Tracks
Calorimeter Hits
Clustering
Track Extrapolating
Local max finding
Track Matching
Hough Clustering
Cone Clustering
Axis Merging
Energy Splitting
Energy & Time Matching
Charged fragment absorption
HCAL Extrapolating
HCAL Cluster Splitting
HCAL Cluster Merging
Particle Flow Object

# Software: Particle Identification

- Simulation of track-level dN/dx in TPC
  - A parameterized model was established based on results from Garfield++ simulation including gas ionization , drift and diffusion of electrons in the gas
  - A Gaudi algorithm is used to generate the track’s dN/dx information according to the parameters of an input track

TPC



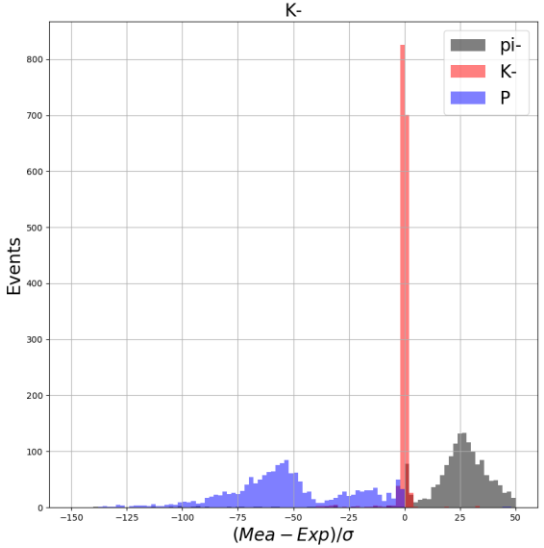
dN/dx mean

dN/dx sigma

- Simulation of Time-of-Flight in OTK
  - A Gaudi Algorithm is used to calculate TOF of a track

    Geometry: R=1850mm, L/2=2350mm (will be updated based on newest geometry)

    TOF model: $t_{mea} = tof_{truth} + Gaussian(0, t_{bunch} = 20\text{ps}) + Gaussian(0, \sigma = 50\text{ps})$

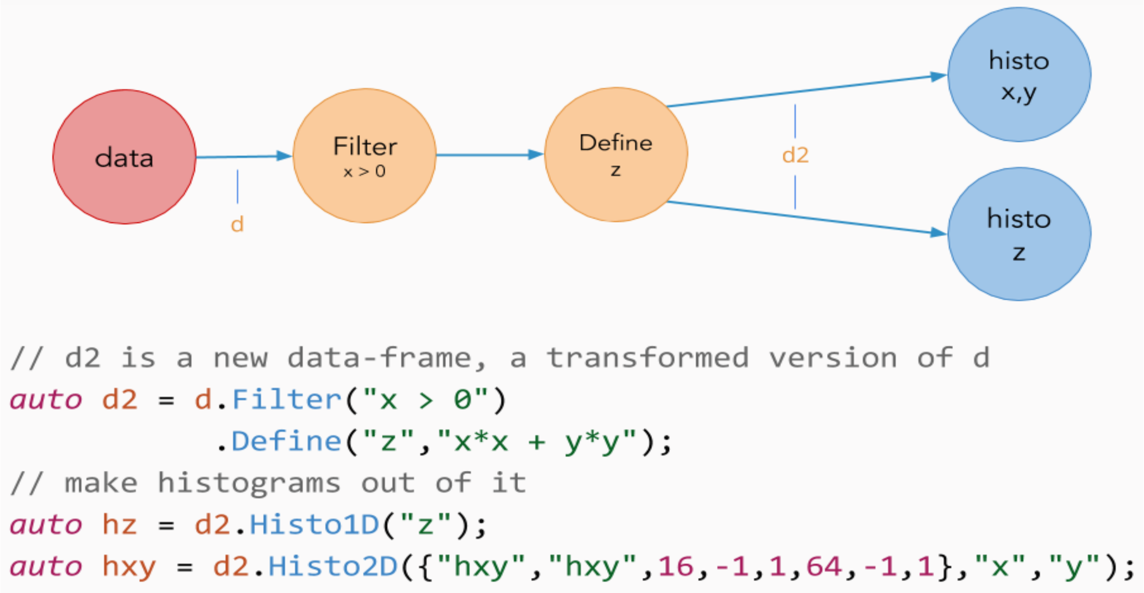


# Software: Detector Software Summary

| Sub Detector | Options | Detector description/Simulation | Digitization | Reconstruction |
|---|---|---|---|---|
| MDI+LumiCal | | Implemented | None | None |
| VTX | Baseline | None | | Clusters are formed and then converted into space points.<br>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 |
| TPC | Baseline | 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 |
| ECAL-Barrel | Baseline | Materials and geometry from the preliminary design | Model based on testbeam data | New PFA algorithm |
| ECAL-Endcap | | | None | Being validated with MC data |
| HCAL-Barrel | | | Model based on testbeam data | Being developed |
| HCAL-Endcap | | | None | |
| MUON-Barrel | | Added materials and geometry | None | The reconstructed tracks are extrapolated to the Muon Detector and matched with the muon track according to the truth information. |
| MUON-Endcap | | | None | |

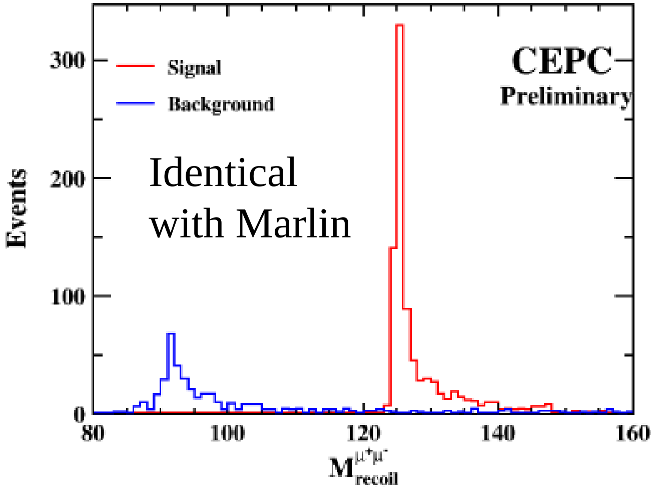
# Software: Analysis Framework

- RDataFrame is a powerful tool for parallel data analysis
  - Program language: Python and C++
  - Declarative programming and parallel processing
  - Used by many experiments such as FCC-ee

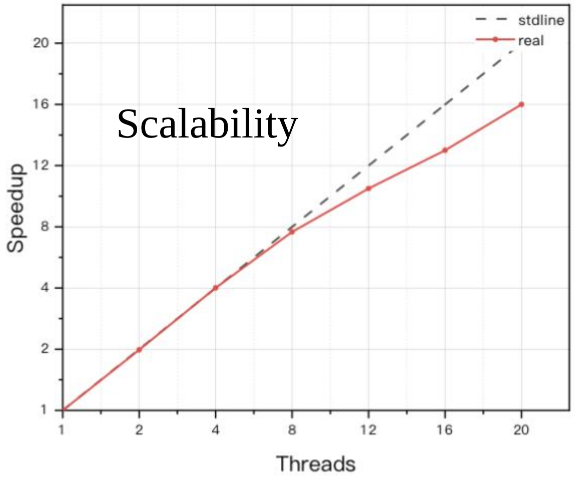


```
// d2 is a new data-frame, a transformed version of d
auto d2 = d.Filter("x > 0")
           .Define("z","x*x + y*y");
// make histograms out of it
auto hz = d2.Histo1D("z");
auto hxy = d2.Histo2D({"hxy","hxy",16,-1,1,64,-1,1},"x","y");
```

- Recent 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 test with two physics channels
    - e+e- -> Z(mumu)H
    - e+e- ->H(2jet) mumu



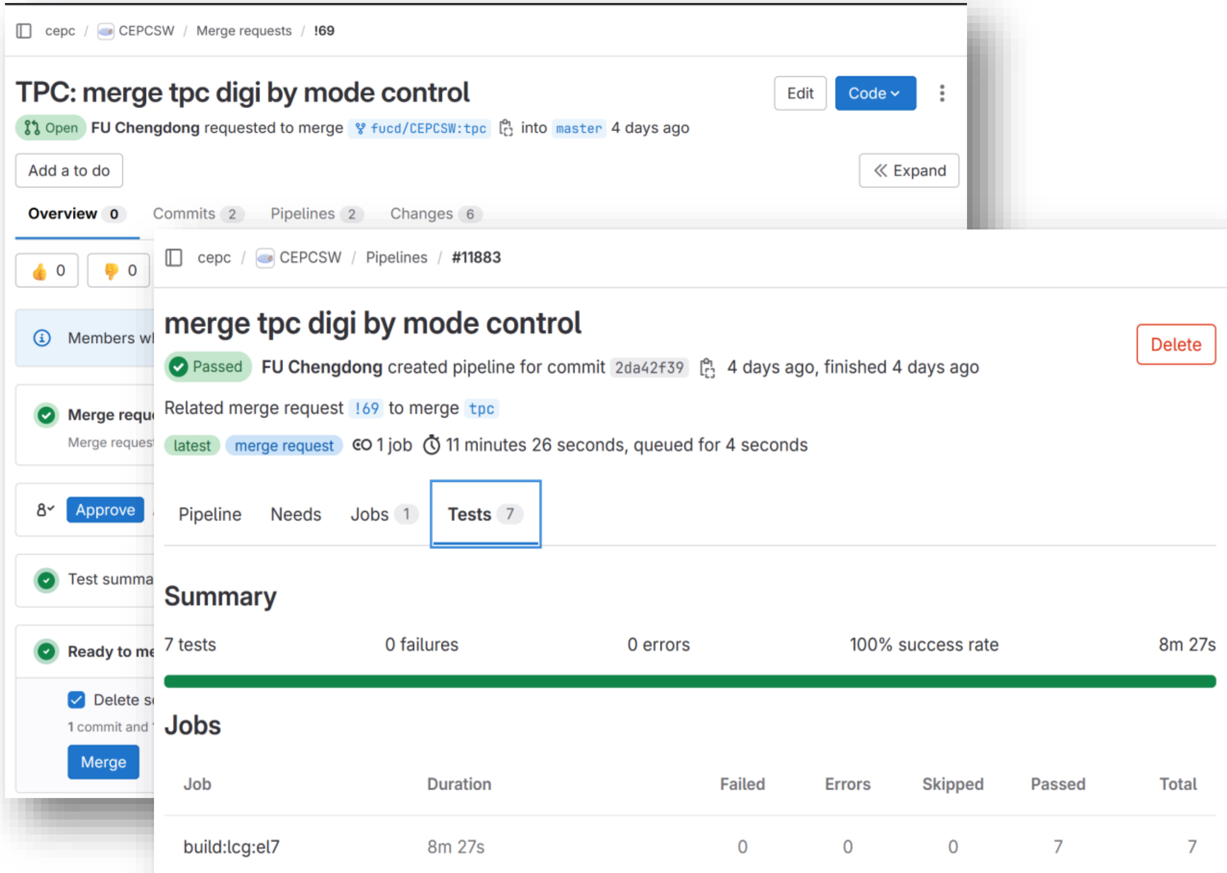
Identical with Marlin



Scalability

# Software: Development Environment

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



cepc / CEPCSW / Merge requests / !69

**TPC: merge tpc digi by mode control**

**FU Chengdong** requested to merge `fucd/CEPCSW:tpc` into `master` 4 days ago

cepc / CEPCSW / Pipelines / #11883

**merge tpc digi by mode control**

Passed **FU Chengdong** created pipeline for commit `2da42f39` 4 days ago, finished 4 days ago

Related merge request !69 to merge `tpc`

latest merge request 1 job 11 minutes 26 seconds, queued for 4 seconds

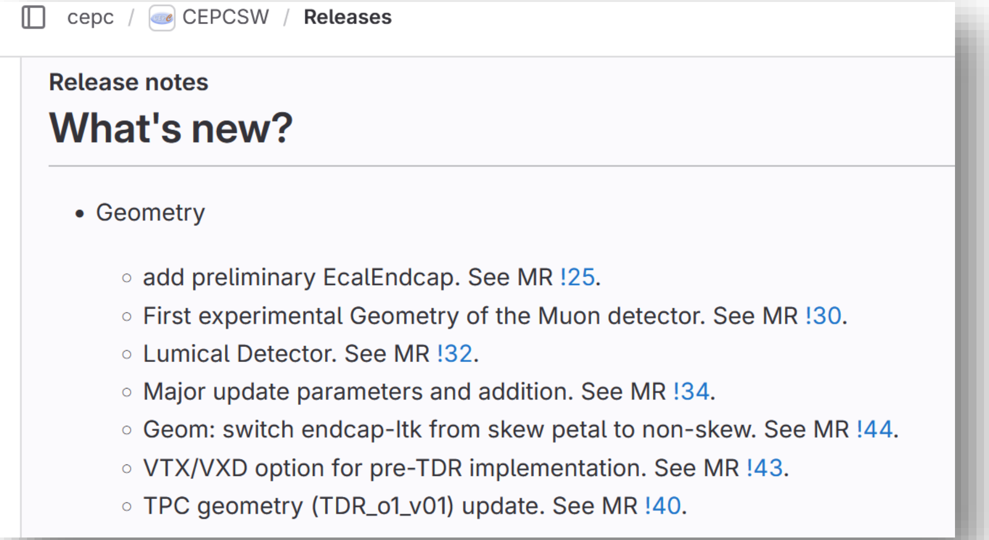
**Summary**

7 tests 0 failures 0 errors 100% success rate 8m 27s

**Jobs**

| Job | Duration | Failed | Errors | Skipped | Passed | Total |
|---|---|---|---|---|---|---|
| build:lcg:el7 | 8m 27s | 0 | 0 | 0 | 7 | 7 |

- “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 / CEPCSW / **Releases**

**Release notes**

## What's new?

- Geometry
  - add preliminary EcalEndcap. See MR !25.
  - First experimental Geometry of the Muon detector. See MR !30.
  - Lumical Detector. See MR !32.
  - Major update parameters and addition. See MR !34.
  - Geom: switch endcap-Itk from skew petal to non-skew. See MR !44.
  - VTX/VXD option for pre-TDR implementation. See MR !43.
  - TPC geometry (TDR_o1_v01) update. See MR !40.

# CEPC Computing (1)

## Technical challenges

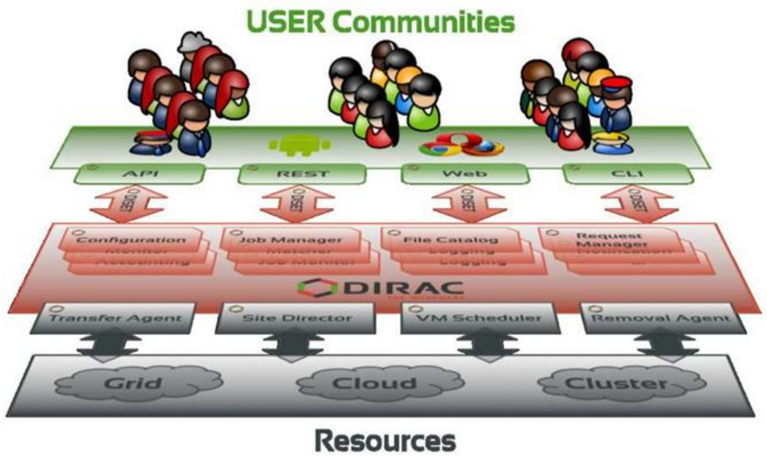
- Management of diverse hardware needed by new HEP applications
- Increasing complexity in data management caused by large data volume
- More complicated workflow for data transfer and job scheduling
- Growing complications in using opportunistic resources efficiently

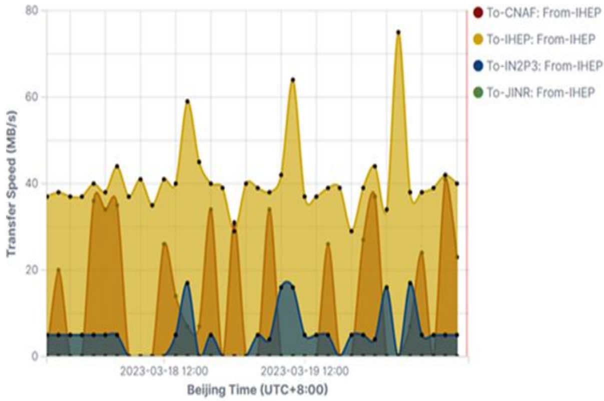
## Technology survey and our choices

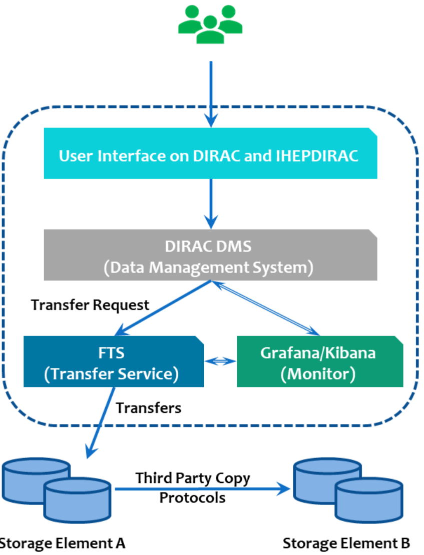
- DIRAC chosen as the distributed computing middleware
  - Having strong capabilities proven by other HEP experiments including LHCb and Belle II
  - Being able to integrate heterogeneous computing resource, e.g. Grid, HTC Cloud and supercomputer
  - DIRAC consortium provides a platform to share experiences 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 TPC test system was established and monitoring was implemented with ElasticResearch+Kibana/Grafana






User Interface on DIRAC and IHEPDIRAC

DIRAC DMS
(Data Management System)

Transfer Request

FTS
(Transfer Service)

Grafana/Kibana
(Monitor)

Transfers

Third Party Copy
Protocols

Storage Element A

Storage Element B

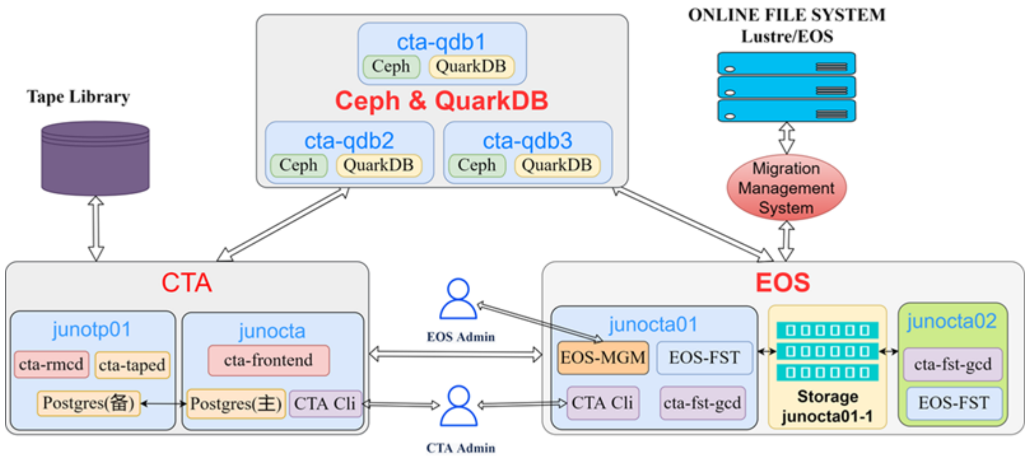
# CEPC Computing (3)

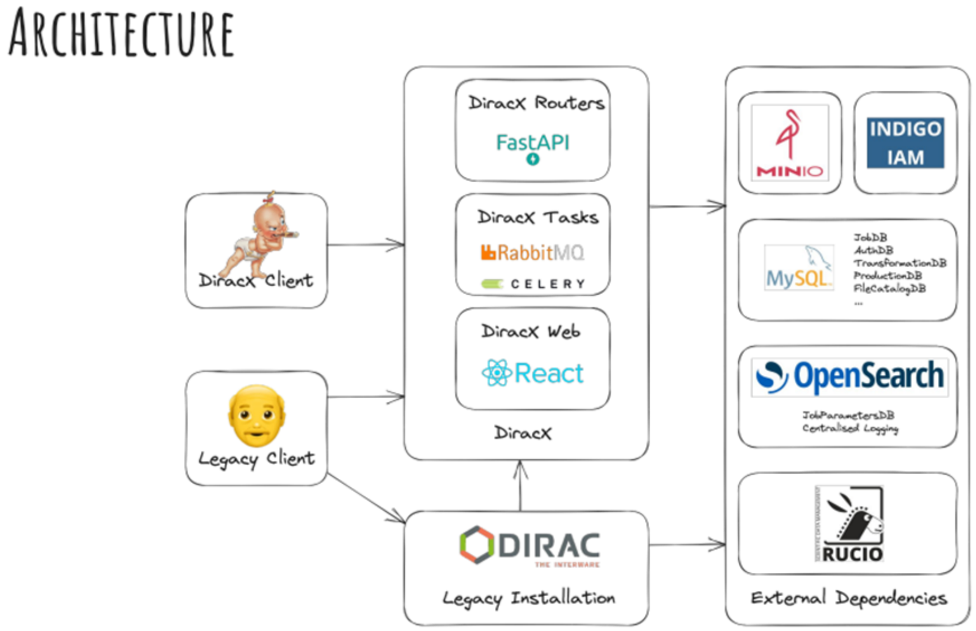
## Four data storage systems were deployed at IHEP

- 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





# Research team and working plan

- Software
  - Institutes: IHEP, SDU, USTC, SYSU, FDU, SCNU, ZZU
  - Manpower: 15 faculties, 2 Postdocs, 4 PhD students, 3 MSc students
  - Work plan
    - Application of multithreading with Gaudi and adding the support of data-parallel computing
    - Re-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 faculties
  - 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)
  - One of earliest users of Key4hep
  - Extension of Edm4hep, Application of Gaussino Framework, Integration of Pandora with Key4hep, Automated performance validation, etc.
- Collaboration with IDEA detector group in R&D of drift chamber
  - Implementation of waveform reconstruction with machine learning method
- Collaboration with the ACTS group
  - Adding precise simulation of drift chamber into the Open Data Detector
  - Implementation of track reconstruction in silicon trackers with ACTS/TRACCC
- IHEP is an official member of the DIRAC consortium
  - Integration of cloud resources, support of multi-core jobs, etc.
  - Evaluation of DiracX

# RefDet TDR: Software Chapter

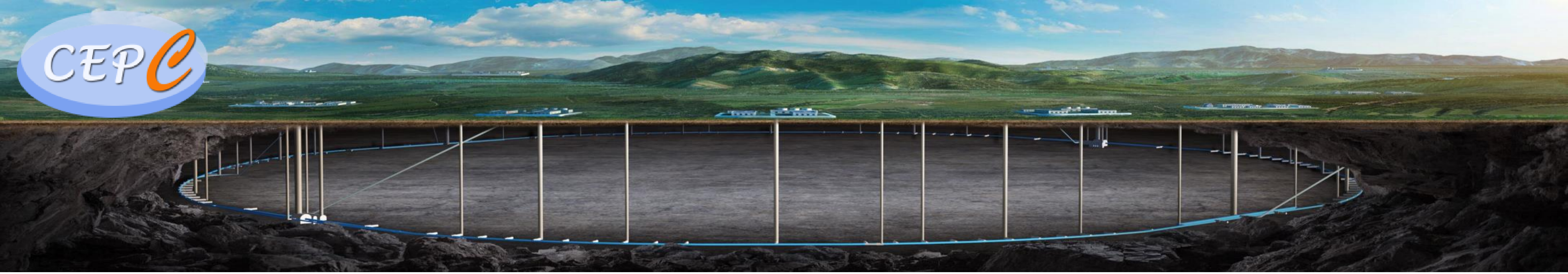
- Introduction
- Core software
  - Software Architecture
  - Integration with Key4hep
  - Data management
  - Control flow
- Simulation
  - Generator
  - Simulation framework
  - Fast simulation
- Reconstruction algorithms
  - Tracking algorithms
  - PID Algorithms
  - Particle Flow Algorithms
- Analysis 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

# Summary

- The CEPCSW is being developed to
  - support detector R&D and physics potential studies
  - build the offline system of the experiment
- Dirac-based distributed computing platform was deployed
  - technical assessment on DiracX and Data Lake ongoing
- 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*