Report on CEPC software workshop in Hong Kong and future plans

Xingtao HUANG
Shandong University
2020/02/14

HK Mini-Workshop: Experiment / Detector - Software and Physics Requirements for e+e- Colliders

- Two days meeting organized by Paolo and Joao
  - Jan. 16: Physics and Detector
  - Jan. 17: Software

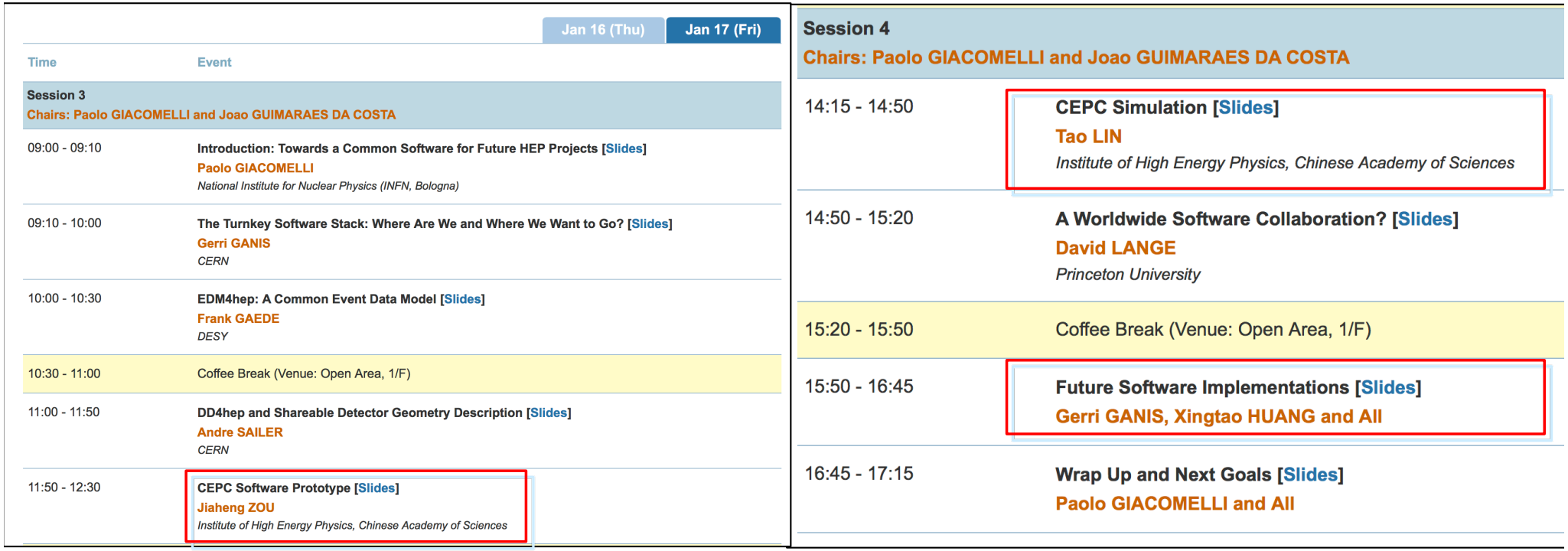
Paolo Giacomelli

Purpose of today's workshop

- Keep the same collaborative spirit shown at the meeting in Bologna!
- Continue in the footsteps of the first workshop
- Show the development done since Bologna's workshop
- Illustrate and detail the next steps of the development of the common software
  - Describe the action items
  - Possibly show the timeline of the future implementations
- Plan the next workshop

Agenda, People and Talk

http://iasprogram.ust.hk/hep/2020/workshop_experiment.php



Jan 16 (Thu) | Jan 17 (Fri)

Time	Event
Session 3 **Chairs: Paolo GIACOMELLI and Joao GUIMARAES DA COSTA**	
09:00 - 09:10	**Introduction: Towards a Common Software for Future HEP Projects [Slides]** **Paolo GIACOMELLI** *National Institute for Nuclear Physics (INFN, Bologna)*
09:10 - 10:00	**The Turnkey Software Stack: Where Are We and Where We Want to Go? [Slides]** **Gerri GANIS** *CERN*
10:00 - 10:30	**EDM4hep: A Common Event Data Model [Slides]** **Frank GAEDE** *DESY*
10:30 - 11:00	Coffee Break (Venue: Open Area, 1/F)
11:00 - 11:50	**DD4hep and Shareable Detector Geometry Description [Slides]** **Andre SAILER** *CERN*
11:50 - 12:30	**CEPC Software Prototype [Slides]** **Jiaheng ZOU** *Institute of High Energy Physics, Chinese Academy of Sciences*

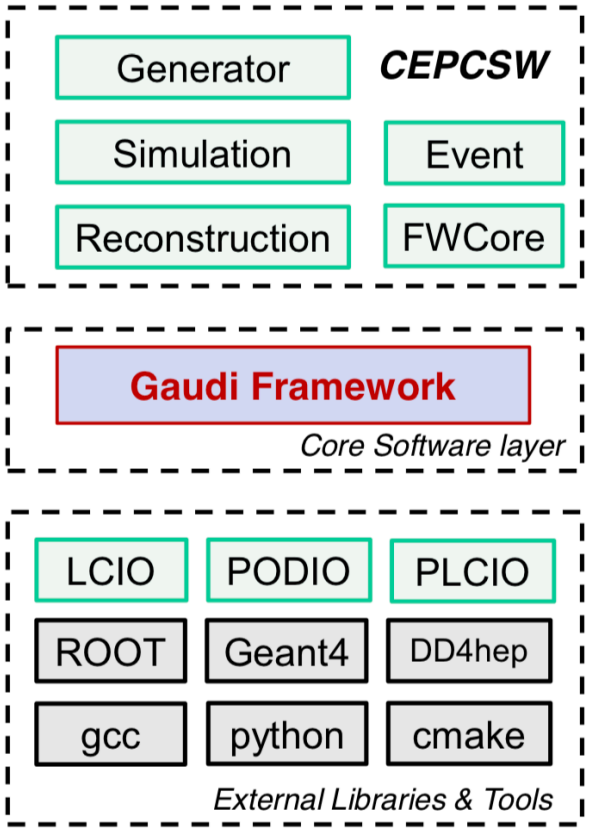
Session 4 **Chairs: Paolo GIACOMELLI and Joao GUIMARAES DA COSTA**	
14:15 - 14:50	**CEPC Simulation [Slides]** **Tao LIN** *Institute of High Energy Physics, Chinese Academy of Sciences*
14:50 - 15:20	**A Worldwide Software Collaboration? [Slides]** **David LANGE** *Princeton University*
15:20 - 15:50	Coffee Break (Venue: Open Area, 1/F)
15:50 - 16:45	**Future Software Implementations [Slides]** **Gerri GANIS, Xingtao HUANG and All**
16:45 - 17:15	**Wrap Up and Next Goals [Slides]** **Paolo GIACOMELLI and All**

- Representatives from CEPC,CLIC,FCC and ILC
- Total 9 talks
  - 1 for introduction
  - 3 for common software: Key4hep, EDM4hep and DD4hep
  - 2 for CEPC software: Software Prototype and Simulation
  - 1 for collaboration
  - 1 for discussion
  - 1 for summary

Status of CEPC Software Prototype

Jiaheng Zou

- Since the Bologna workshop (June 2019), CEPCSW prototype has been developed using Gaudi, DD4hep, Geant4 and PLCIO, etc.

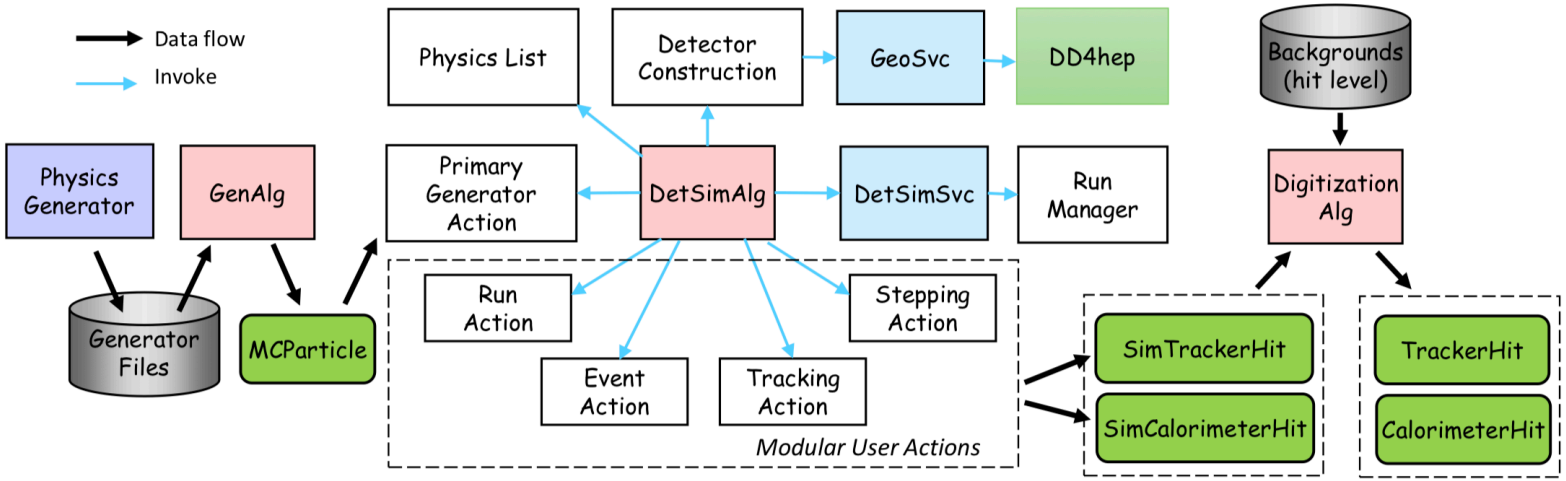


Components	Tasks	Status
General	Software infrastructure (Gaudi) Core modules (FWCore)	√
EDM & I/O	PLCIO data model and I/O LCIO compatible reader	√
Geometry and Simulation	DD4hep integration Simulation framework	√
Reconstruction	SiliconTracking	√
	More reconstruction algorithms	In progress
Build and release	Git, CMake, CVMFS	Ready

Status of CEPC Simulation

Tao Lin

- Based on Gaudi and DD4hep.
  - Reuse part of interfaces defined in FCCSW.
- In the current prototype, “Tracker” is setup.



- A simulation framework prototype is developed.
  - Configurable Geometry with DD4hep’s XML files: support multiple options of detectors and beam test geometry.
  - Physics generator: Integrate with external physics generators easily.
  - Modular user actions to collect data in simulation: Save more information other than the event data model.

EDM4hep and PLCIO

Status of EDM4hep

- Common event data model for future collider
  - Provide foundation of the Common Software Stack (Key4hep)
  - Use PODIO high-level EDM generator
  - Rely on experience with LCIO and fcc-edm
- Meetings every 2-4 weeks since bologna's workshop
  - https://indico.cern.ch/category/11461/

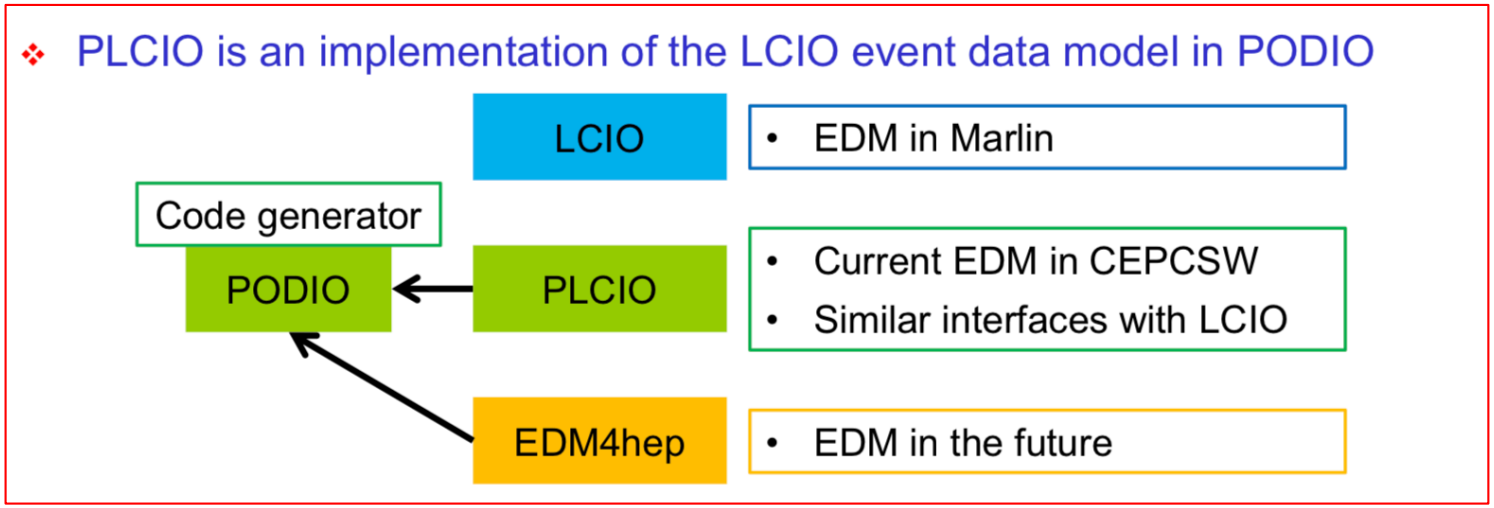
  (CEPC People remotely join the meeting)
- So far implemented the simulation model, i.e.
  - MCParticle, SimTrackerHit and SimCalorimeterHit
  - Vector3f, Vector3d, Vector2i
- Created github page:
  - https://github.com/HSF/EDM4hep

Plan of EDM4hep

- The project is in rather early phase
- Start implementation of reconstruction part, i.e.
  - Track Parameters and TrackState
- Provide navigation between TracketHit and SimTrackerHit
  - using appropriate helper functions
- Support ACTS with different parameters
- Add different I/O implementations in PODIO
  - Currently support ROOT
  - Working on HDF5, SIO …
- Benchmark the I/O performance in MT mode
- Investigate automatic generation of lambda functions for RDataFrame

PLCIO: LCIO EDM in PODIO

- Currently used by CEPC until moving to EDM4hep
- CEPC is the first user of PLCIO
- Need helper classes for event navigation between TracketHit and SimTrackerHit
- Optimize Performance of PLCIO
- At the same time, CEPC should work on EDM4hep



DD4hep

Detector Geometry Description :DD4hep

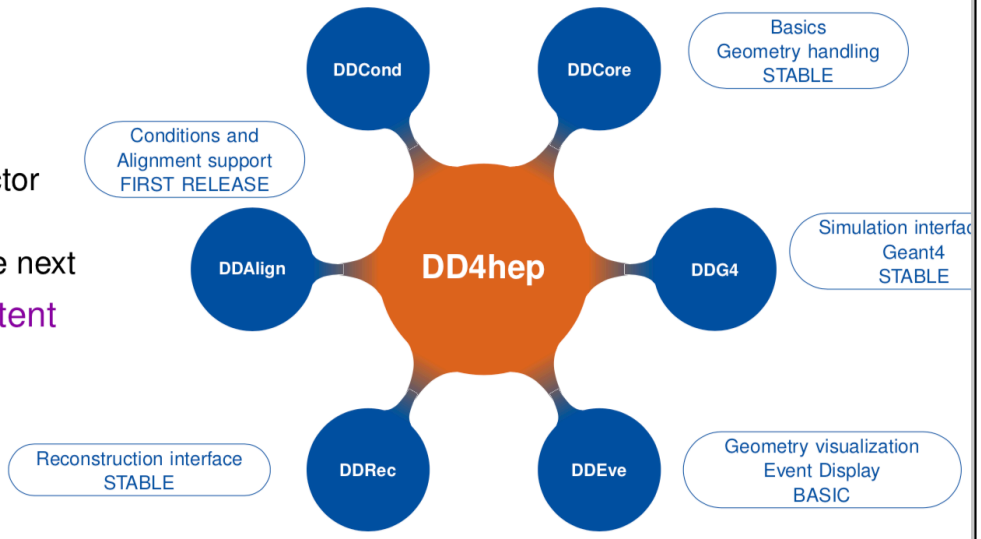
- Relative stable project
- Used by ILC and CLIC, FCC and CEPC (currently “Tracker”)
- Default palette of usable sub-detector solution for Key4hep

DD4hep – Overview

AIDA 2020

CERN

- Complete Detector Description
  - Providing geometry, materials, visualization, readout, alignment, calibration. . .
- Supports full experiment life cycle
  - Detector concept development, detector optimization, construction, operation
  - Facile transition from one stage to the next
- Single source of information → consistent description
  - Use in simulation, reconstruction, analysis, etc.
- Ease of Use
- Few places for entering information
- Minimal dependencies



Use XML file and C++ driver to build Detectors

Detector XML

- XML structure to set parameters for detectors
- C++ driver to interpret XML parameters and create `DetElements` and `Volumes`
  - Define sensitive parts (attached with `SensitiveDetector`) and radiator, which has to be known fo
- Attach sensit
  in XML

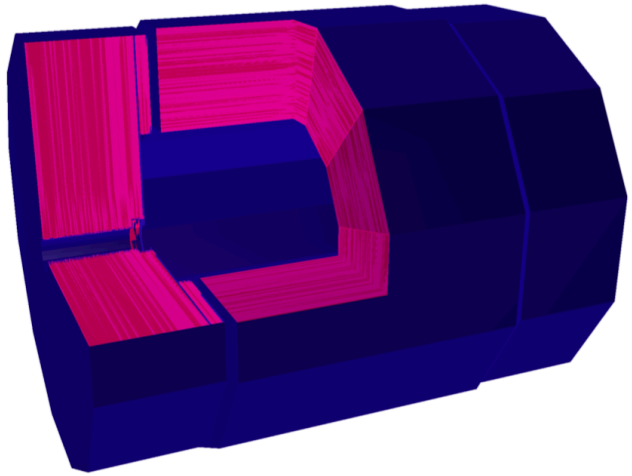
```
<readout name
  <segmentati

  <id>...x:32
</readout>
```

```
<detector
    name="ECalBarrel"
    type="GenericCalBarrel_o1_v01"
    id="42" readout="ECB">
  <dimensions
      numsides="ECalBarrel_symmetry"
      rmin="ECalBarrel_inner_radius"
```

A. Sailer

Detector Driver

- C++ model of separation of 'data' and 'behaviour'
- Drivers return single 'reference' to the DetElement object



```
static dd4hep::Ref_t create_element(
    dd4hep::Detector& description,
    xml_h element,
    dd4hep::SensitiveDetector sens) {
  xml_det_t e = element;
  DetElement aDetector(e.nameStr(), e.id());
  //...
  sens.setType("calorimeter");
  //...
  return aDetector;
}
DECLARE_DETELEMENT(AName, create_element)
```

Sharing Drivers

- DD4HEP's plugin manager can load drivers at runtime
- Expected XML structure needs to be known to other users
- Existence of driver needs to be known to other users

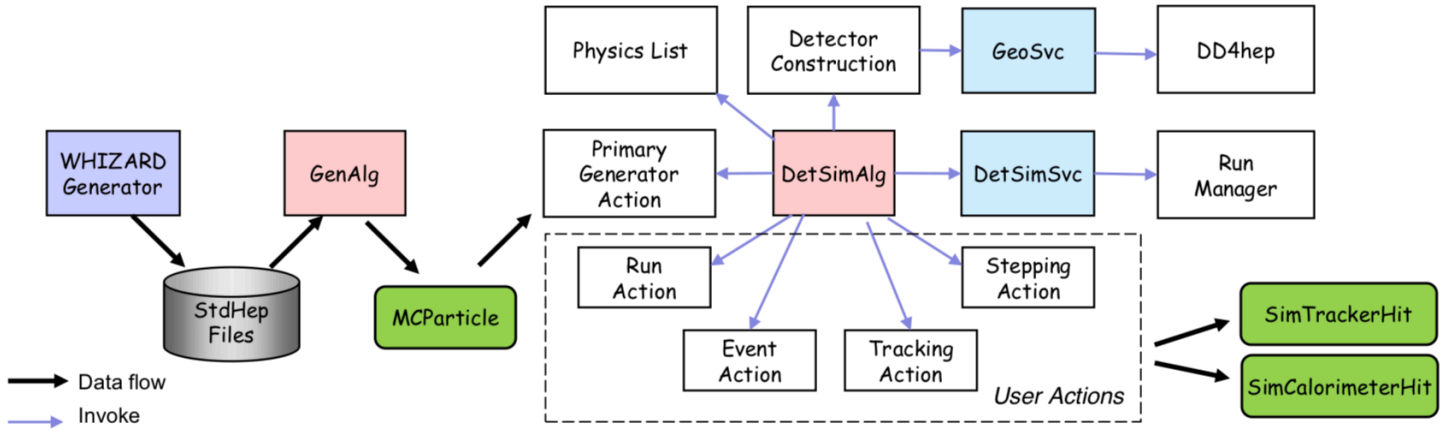
Existing Palettes:

- DD4hep: https://github.com/AIDASoft/DD4hep/tree/master/DDDetectors
- lcgeo: https://github.com/iLCSoft/lcgeo/tree/master/detector
- FCCSW https://github.com/HEP-FCC/FCCSW/tree/master/Detector
- CEPC?

Place common drivers in DD4hep or different package?

Plan of DD4hep

- DD4hep description of IDEA, but need to implement
  - Dual readout calorimeter
  - Muon system
- Optimize common interface for reconstruction in DD4hep
  - Already generic classes that describe all the subdetectors
  - Work on new subdetectors, i.e. the dual readout calorimeter
- Validate DD4hep-based (unified-geometry)-Service simulation and reconstruction for CEPC



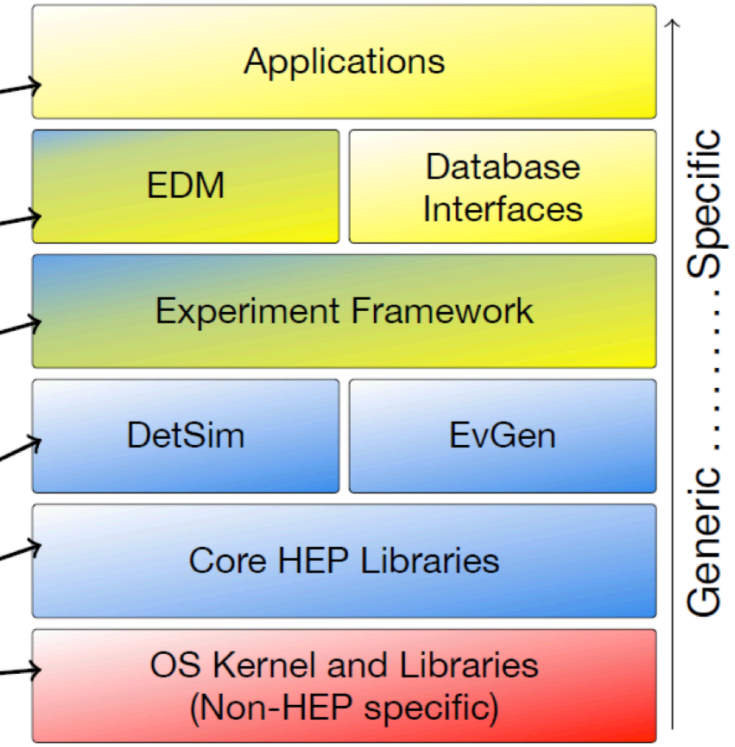
Key4hep

Common Software Stack (Key4hep)

A typical HEP Software Stack

Applications usually rely on large number of libraries, where some depend on others

- Interfaces to tracking and reconstruction libraries (PandoraPFA, ACTS)
- (More or less) experiment specific event data model libraries
- Experiment core orchestration layer, which controls everything else: Marlin, Gaudi, CMSSW, AliRoot
- Packages used by many experiments: DD4hep, Pythia, ...
- Usual core libraries (ROOT, Geant4, CLHEP, ...)
- Non-HEP libraries: boost, python, cmake ...



Gaudi Framework

- A good candidate for Key4hep
- Provided required interfaces and services
- Need Data Service handling with EDM4hep/pLCIO
  - FCC developed FWCore

❖ FCCSW FWCore

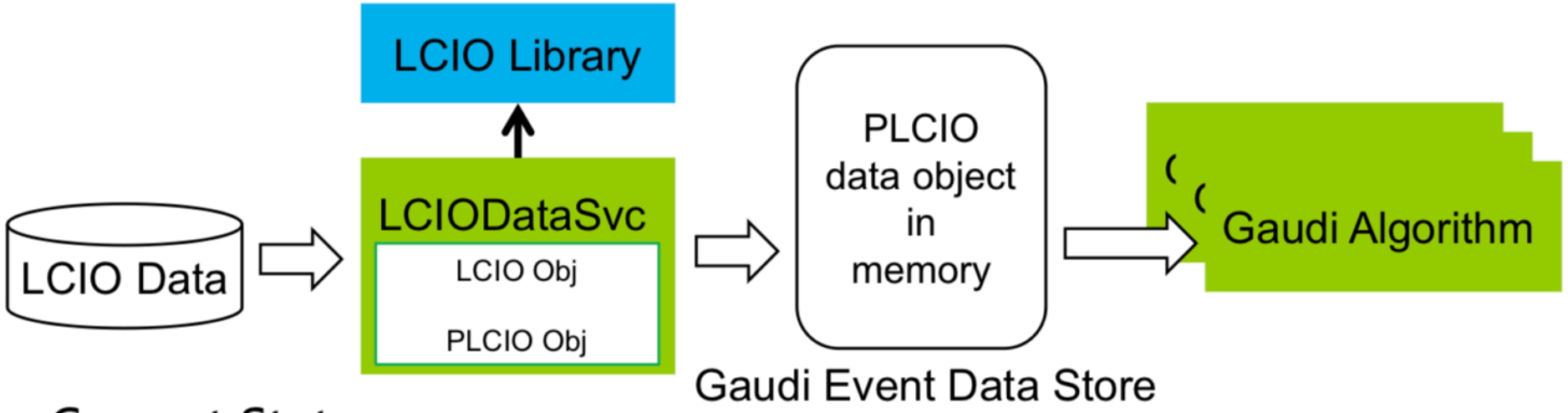
- DataWrapper: PLCIO data collection -> DataObject in Gaudi
- DataHandle: user interface to register/retrieve data to/from Gaudi TES (Transient Event Store)
- PODIO data service: read/write PODIO data objects

❖ CEPCSW FWCore

- Mainly taken from FCCSW FWCore (many thanks)
- Extension to read LCIO data generated by Marlin

Read the Existing LCIO Data

- LCIODataSvc
  - Read LCIO files via the LCIO library
  - Convert LCIO data objects to PLCIO data objects
  - Register PLCIO data objects to Gaudi Event Data Store



- Current Status
  - Most LCIO data types can be retrieved as PLCIO objects in CEPCSW
  - Some of the data relations are not fully recovered (there are some limitation for data analysis now)

Detector Simulation

- Streamline Gaudi/FCCSW/G4 interface
- Develop a parallelized simulation solution for CEPC
  - Based on Geant4-MT and MPI
- Digitization IDEA Dual Readout calorimeter and Muon system
- Develop a coherent simulation framework allowing mixing of full and fast simulation
  - Test setup available in FCCSW ?
- Fast simulation with Delphes
  - Validation of Delphes cards for proposed solutions
  - Uniformize {full sim + reco} and Delphes outputs

Reconstruction

- Provide vertexing, solid Particle ID and c,b tagging
  - Migration of all existing algorithms from Marlin to Gaudi-based frameworks
    - For tracking, Particle-Flow, Jet Flavor tagging
    - Validation of the migrated algorithms
- Integration of ACTS
- Integrate Particle Flow (Pandora, Arbor, ...)
- Integration of tensorflow and ML techniques for reconstruction
- ...

Available in FCCSW:

- Tracking
  - Track seeding (TrickTrack) for silicon tracker
  - Hough Transform for drift chambers (not yet in the master)
  - Under implementation / investigation: ACTS integration, Conformal tracking
- Calorimeters
  - Sliding window (rectangular/ellipse)
  - Topo-clustering
  - Under investigation: deep learning

Integration of ACTS into Key4hep

- CEPC groups are interested in getting ACTS to work
  - eventually all groups would be
- CEPC could contribute ACTS gaudi wrapper
- Potentially extension to EDM4hep (Trackerhits, track state,…)
  - see point above under Reconstruction
- Need of pattern recognition algorithms
  - should investigate porting the Conformal Tracking from CLIC
  - in a first step run this in the Marlin-wrapper

Key4hep Activities after HK workshop

- Setup mail list: key4hep-sw@cern.ch
- First meeting on Feb. 12, 2012
  - > 20 people joined the meeting
  - CEPC people joined remotely
- Technical Issues
  - Use Spack to build the whole stack
  - Installations based on LCG releases
  - Use /cvmfs/sw.hsf.org/key4hep repository
  - Move EDM4hep to key4hep
- Discussion on the first face to face workshop
  - Longer workshop (2 or 3 days)
  - For developer
  - Late June, 2020
- Goal: have a first working release

Future Plan

CEPCSW Plan in 2020

- First Version (May-Release)
  - Software environment
  - EDM: relationship
  - Uniform geometry: Simulation and Reconstruction
  - Tracker: silicon / TPC
- Second Version (October-Release)
  - PFA: Arbor and/or Pandora
  - Jet / Flavor tag
  - Integration of ACTS
  - Geometry: DD4Hep
  - Fast simulation framework: tracker and/or calorimeter
  - Multi-threading testing / EDM performance optimization
  - Detector design
  - Visualization ?

CEPCSW Plan for 2021 and 2022

- 2021
  - PLCIO ➔ EDM4Hep
  - Support data analysis of beam test
  - Fast simulation: data production and physics analysis
  - Non-uniformity of magnetic field, noise and background mixing: optimization of reconstruction algorithm and physics analysis
- 2022
  - Integration with Key4Hep
  - Optimization, performance and validation of reconstruction algorithms, physics analysis
  - Online event filter
  - Parallel computing
  - Application of Machine learning in Reconstruction

Remark

- Most of issues or action items have been listed in the CEPCSW three years' Plan
  - Support CEPC R&D
- Adjust our plan according to the communities schedules
- CEPC should contribute the communities
  - Join development of common toolkits: DD4hep, EDM4hep, Key4hep, FWCore…
  - Develop more innovative applications, such as new sub-detector, new reconstruction algorithms…
- Need more people working on CEPC software and Common Software Toolkits!

Thanks