CEPCSW tutorial and AHCAL
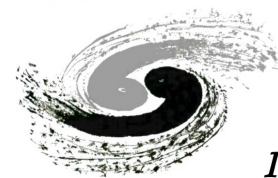
Fangyi Guo

Institute of High Energy Physics, Chinese Academy of Sciences

Introduction

- **CEPCSW: a common software framework for future collider experiments.**
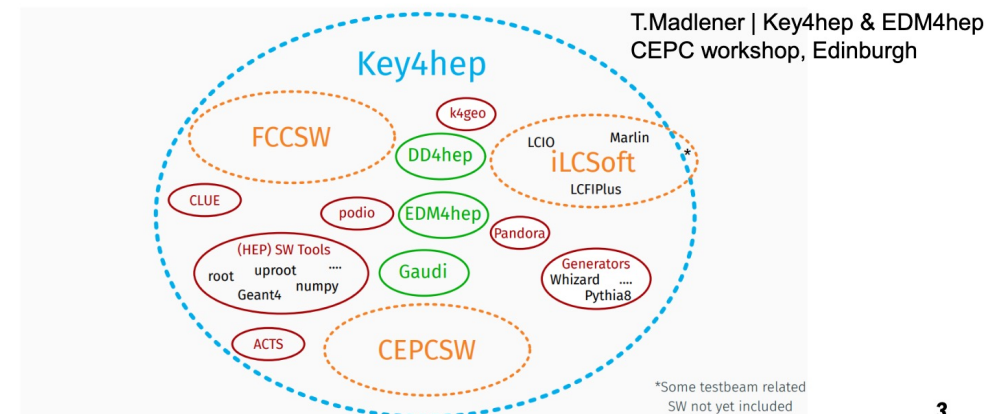 - Underlying framework: Gaudi
  - Event data model: EDM4hep
  - Data management: k4FWCore
  - Detector description: DD4hep
  - Other external libraries (G4, ROOT, etc)
  - CEPC applications (digitization, reconstruction, etc.)

- **Git: cepc / CEPCSW · GitLab (ihep.ac.cn)**
  - master branch in development.
  - Latest tag: tdr 24.4.1

- **Previous tutorial:**
  - In Sep. 2020: New CEPCSW Tutorial and detector study · Indico
  - In March 2024: CEPCSW Tutorial (March 8, 2024) · Indico of IHEP (Indico)

T.Madlener | Key4hep & EDM4hep
CEPC workshop, Edinburgh

*Some testbeam related
SW not yet included

3

Introduction

- **Hands on: follow Lin Tao's instruction**
  - Clone CEPCSW from git, check the branch tdr.24.4.1
  - Compile the package: ./build.sh
    - Next time when you login: source setup.sh
  - Test the Hello World job: ./run.sh Examples/options/helloalg.py

```
-bash-4.2$ ./run.sh Examples/options/helloalg.py
# setting LC_ALL to "C"
# --> Including file '/cefs/higgs/guofy/CEPCSW_tdr24.4.1/Examples/options/helloalg.py'
# <-- End of file '/cefs/higgs/guofy/CEPCSW_tdr24.4.1/Examples/options/helloalg.py'
ApplicationMgr       SUCCESS
=============================================================================
=============
                                    Welcome to ApplicationMgr (GaudiCoreSvc v36r16)
                                    running on lxslc714.ihep.ac.cn on Wed Jun  5 11:21:11 2024
=============================================================================
=============
ApplicationMgr          INFO Application Manager Configured successfully
helloAlg                INFO MyInt: 42
EventLoopMgr         WARNING Unable to locate service "EventSelector"
EventLoopMgr         WARNING No events will be processed from external input.
ApplicationMgr          INFO Application Manager Initialized successfully
ApplicationMgr          INFO Application Manager Started successfully
ApplicationMgr          INFO Application Manager Stopped successfully
EventLoopMgr            INFO Histograms converted successfully according to request.
ApplicationMgr          INFO Application Manager Finalized successfully
ApplicationMgr          INFO Application Manager Terminated successfully
```

Job options

- **Python job options**
  - e.g. in Examples/options/helloalg.py

```python
#!/usr/bin/env python

from Gaudi.Configuration import *

from Configurables import HelloAlg

helloalg = HelloAlg("helloAlg")
helloalg.MyInt = 42

# ApplicationMgr
from Configurables import ApplicationMgr
ApplicationMgr( TopAlg = [helloalg],
                EvtSel = 'NONE',
                EvtMax = 10,
)
```

Import self-defined algorithm HelloAlg:
  Algorithm input par: MyInt = 42.

Run manager:
  run helloalg,
  no selection
  run 10 events.

- Example of detector simulation: see in Tao's slide.
  - Test to run the simulation: in dir CEPCSW:
    ./run.sh Detector/DetCRD/scripts/CRD_o1_v01-SimRec.py
  - Output: CRD-o1-v01-SimRec00.root

Output file

```
root [1] .ls
TFile**        CRD-o1-v01-SimRec00.root
 TFile*        CRD-o1-v01-SimRec00.root
  KEY: TTree    events;1        events data tree
  KEY: TTree    configuration_metadata;1        configuration_metadata data tree
  KEY: TTree    metadata;1      metadata data tree
  KEY: TTree    podio_metadata;1        metadata tree for podio I/O functionality
```

Info are stored in EDM4hep format, e.g. Hcal hits in edm4hep::SimCalorimeterHitData.

```
HcalBarrelCollection = (vector<edm4hep::SimCalorimeterHitData>*)0xa8053b0
HcalBarrelCollection.cellID = 31244023188816758, 31244006009078646, 31243984454373238,
07670, 31525403676902262, 31525386497164150, 31525365022458742, 31806822819431286, 31806
, 32088220487254902, 32088203307516790, 32088181832811382, 32088164653073270, 3237061888
70597405524598, 32370584520753782, 32652046612693622
HcalBarrelCollection.energy = 0.000001, 0.000003, 0.000002, 0.000000, 0.000001, 0.00000
, 0.000006, 0.000001, 0.000001, 0.000001, 0.000001, 0.000001, 0.000003, 0.000002, 0.0000
HcalBarrelCollection.position.x = -2328.110107, -2354.199951, -2375.290039, -2401.37988
-2474.649414, -2495.739258, -2521.829102, -2569.009033, -2590.098877, -2616.188721, -264
58252, -2722.896240, -2758.795898, -2786.035156, -2813.274170, -2840.513428
HcalBarrelCollection.position.y = 1179.654907, 1187.685913, 1204.377075, 1212.408203, 1
61499, 1261.852783, 1269.883789, 1294.606079, 1311.297241, 1319.328369, 1327.359375, 134
138, 1385.666138, 1400.666138, 1415.666138, 1430.666138
HcalBarrelCollection.position.z = 230.800003, 230.800003, 230.800003, 230.800003, 240.8
240.800003, 250.800003, 250.800003, 250.800003, 260.799988, 260.799988, 260.799988, 260.
270.799988, 270.799988, 280.799988
```

To extract info:
 - a ROOT script, read branch as array.
 - an algorithm to re-write from edm4hep to ntuple.

Algorithm

- **Algorithm in CEPCSW: follow Gaudi process.**
  - Use HelloAlg as example ([git](git))

```h
h HelloAlg.h    395 B

1    #ifndef HelloAlg_h
2    #define HelloAlg_h
3
4    #include <GaudiKernel/Algorithm.h>
5    #include <Gaudi/Property.h>
6
7    class HelloAlg: public Algorithm {
8    public:
9        HelloAlg(const std::string& name, ISvcLocator* pSvcLocator);
10
11       StatusCode initialize() override;
12       StatusCode execute() override;
13       StatusCode finalize() override;
14
15   private:
16
17       Gaudi::Property<int> m_int{this, "MyInt", 42};
18   };
19
20
21   #endif
22
```

A class HelloAlg inherited from GaudiKernel::Algorithm

Consist of 3 main parts.

Readin parameter from script.

```cpp
C++  HelloAlg.cpp    428 B

1    #include "HelloAlg.h"
2
3    DECLARE_COMPONENT(HelloAlg)
4
5    HelloAlg::HelloAlg(const std::string& name, ISvcLocator* pSvcLocator)
6    : Algorithm(name, pSvcLocator) {
7
8    }
9
10   StatusCode
11   HelloAlg::initialize() {
12       StatusCode sc;
13
14       info() << "MyInt: " << m_int.value() << endmsg;
15
16       return sc;
17   }
18
19   StatusCode
20   HelloAlg::execute() {
21       StatusCode sc;
22       return sc;
23   }
24
25   StatusCode
26   HelloAlg::finalize() {
27       StatusCode sc;
28       return sc;
29   }
30
31
32
```

  - Add in CMakeList.txt to be included in compiling.

```
3    # Modules
4    gaudi_add_module(Examples
5                    SOURCES src/HelloWorld/HelloAlg.cpp
6                            src/FirstSvc/FirstSvc.cpp
7                            src/SecondAlg/SecondAlg.cpp
```

Algorithm

- **Another example: read tracker hits and tracks, save in ntuple**
  - Code: /cefs/higgs/guofy/CEPCSW_tdr24.4.1/Examples/src/ReadDigi/
  - Script: /cefs/higgs/guofy/CEPCSW_tdr24.4.1/readTrk.py
  - Output: defined in script and code.

```python
from Configurables import ReadDigiAlg
readtrk = ReadDigiAlg("ReadDigiAlg")
readtrk.OutFileName = "ReadTrk_SW_onlyTrk.root"
```

```cpp
m_wfile = new TFile(s_outfile.c_str(), "recreate");
m_mctree = new TTree("MCParticle", "MCParticle");
m_trktree = new TTree("RecoTrk", "RecoTrk");

m_mctree->Branch("N_MCP", &N_MCP);
m_mctree->Branch("MCP_px", &MCP_px);
m_mctree->Branch("MCP_py", &MCP_py);
m_mctree->Branch("MCP_pz", &MCP_pz);
m_mctree->Branch("MCP_E", &MCP_E);
m_mctree->Branch("MCP_endPoint_x", &MCP_endPoint_x);
m_mctree->Branch("MCP_endPoint_y", &MCP_endPoint_y);
m_mctree->Branch("MCP_endPoint_z", &MCP_endPoint_z);
m_mctree->Branch("MCP_pdgid", &MCP_pdgid);
m_mctree->Branch("MCP_gStatus", &MCP_gStatus);

m_trktree->Branch("N_SiTrk", &N_SiTrk);
m_trktree->Branch("N_TPCTrk", &N_TPCTrk);
m_trktree->Branch("N_fullTrk", &N_fullTrk);
m_trktree->Branch("N_VTXhit", &N_VTXhit);
m_trktree->Branch("N_SIThit", &N_SIThit);
m_trktree->Branch("N_TPChit", &N_TPChit);
m_trktree->Branch("N_SEThit", &N_SEThit);
m_trktree->Branch("N_FTDhit", &N_FTDhit);
m_trktree->Branch("N_SITrawhit", &N_SITrawhit);
m_trktree->Branch("N_SETrawhit", &N_SETrawhit);
m_trktree->Branch("trk_px", &m_trk_px);
```

```cpp
try{
const edm4hep::MCParticleCollection* const_MCPCol = m_MCParticleCol.get();
if(const_MCPCol){
    N_MCP = const_MCPCol->size();
    for(int i=0; i<N_MCP; i++){
        edm4hep::MCParticle m_MCp = const_MCPCol->at(i);
        MCP_px.push_back(m_MCp.getMomentum().x);
        MCP_py.push_back(m_MCp.getMomentum().y);
        MCP_pz.push_back(m_MCp.getMomentum().z);
        MCP_E.push_back(m_MCp.getEnergy());
        MCP_endPoint_x.push_back(m_MCp.getEndpoint().x);
        MCP_endPoint_y.push_back(m_MCp.getEndpoint().y);
        MCP_endPoint_z.push_back(m_MCp.getEndpoint().z);
        MCP_pdgid.push_back(m_MCp.getPDG());
        MCP_gStatus.push_back(m_MCp.getGeneratorStatus());
    }
}
}catch(GaudiException &e){
    debug()<<"MC Particle is not available "<<endmsg;
}
```

AHCAL Barrel in CEPCSW

- **AHCAL dd4hep geometry has 2 part:**

  - c++ source code:
    [Detector/DetCEPCv4/src/calorimeter/SHcalSc04_Barrel_v04.cpp · master · cepc / CEPCSW · GitLab (ihep.ac.cn)](#)

  - xml compact file:
    [Detector/DetCRD/compact/CRD_common_v01/SHcalSc04_Barrel_v04_01.xml · master · cepc / CEPCSW · GitLab (ihep.ac.cn)](#)

- **Full CEPC detector geometry configuration (AHCAL is not used)**

  - Can be modifid from [CRD_o1_v05](#)

```
28    <include ref="./CRD_Dimensions_v01_05.xml"/>
29
30    <include ref="../CRD_common_v01/Beampipe_v01_01.xml"/>
31    <include ref="../CRD_common_v01/VXD_v01_01.xml"/>
32    <include ref="../CRD_common_v01/FTD_SkewRing_v01_01.xml"/>
33    <include ref="../CRD_common_v01/SIT_SimplePixel_v01_01.xml"/>
34    <include ref="../CRD_common_v01/TPC_CEPCv4.xml"/>
35    <include ref="../CRD_common_v01/SET_SimplePixel_v01_01.xml"/>
36    <include ref="../CRD_common_v01/Ecal_Crystal_Barrel_v01_02.xml"/>
37
38    <!-- <include ref="../CRD_common_v01/Coil_Simple_v01_01.xml"/> -->
39    <include ref="../CRD_common_v01/SHcalGlass_Barrel_v04_01.xml"/>
40    <!-- <include ref="../CRD_common_v01/SHcalSc04_Endcaps_v01_01.xml"/> -->
41    <!-- <include ref="../CRD_common_v01/Yoke_Barrel_v01_01.xml"/> -->
42    <!-- <include ref="../CRD_common_v01/Yoke_Endcaps_v01_01.xml"/> -->
43    <!--include ref="../CRD_common_v01/Lcal_v01_01.xml"/-->
```
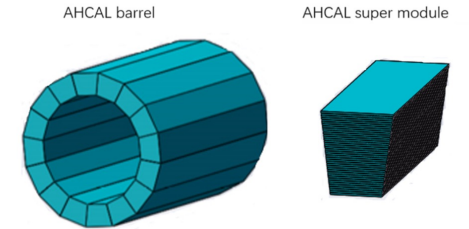
Replace with "../CRD_common_v01/SHcalSc04_Barrel_v04_01.xml"

AHCAL Barrel in CEPCSW

AHCAL barrel   AHCAL super module

- **AHCAL compact file [git]**
  - Bi-readout mode, for RPC and scintillator at the same time (can be disabled).

```
 4    <define>
 5      <constant name="Hcal_cell_size" value="10*mm"/>
 6      <constant name="Hcal_inner_radius" value="Hcal_barrel_inner_radius"/>
 7      <constant name="Hcal_half_length" value="Hcal_barrel_half_length"/>
 8      <constant name="Hcal_inner_symmetry" value="Hcal_barrel_symmetry"/>
 9      <constant name="Hcal_nlayers" value="38"/>
10      <constant name="Hcal_radiator_thickness" value="20.0*mm"/>
11      <constant name="Hcal_chamber_thickness" value="6.5*mm"/>
12      <constant name="Hcal_back_plate_thickness" value="15*mm"/>
13      <constant name="Hcal_lateral_structure_thickness" value="10*mm"/>
14      <constant name="Hcal_stave_gaps" value="0*mm"/>
15      <constant name="Hcal_middle_stave_gaps" value="0*mm"/>
16      <constant name="Hcal_modules_gap" value="2*mm"/>
17      <constant name="Hcal_layer_air_gap" value="0*mm"/>
18      <constant name="HcalSD_glass_anode_thickness" value="0.7*mm"/>
19      <constant name="HcalSD_sensitive_gas_gap" value="1.2*mm"/>
20      <constant name="HcalSD_glass_cathode_thickness" value="1.1*mm"/>
21      <constant name="Hcal_scintillator_thickness" value="3.0*mm"/>
22      <constant name="Ecal_outer_radius" value="Ecal_barrel_outer_radius"/>
23      <constant name="Hcal_readout_segmentation_slice" value="3"/>
24    </define>
```
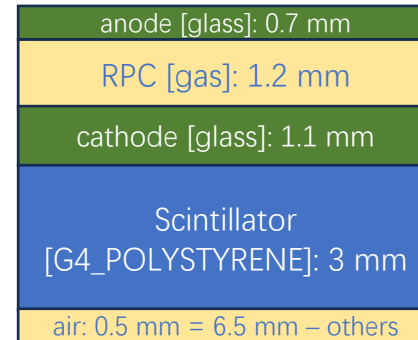
Cell size: 10 mm

Global geometry read from outer (e.g. CRD_Dimensions_v01_05.xml)

38 layers.
 for each layer: 20 mm steel absorber, 6.5 mm cell.

A gap at theta=90 deg.

| anode [glass]: 0.7 mm |
| RPC [gas]: 1.2 mm |
| cathode [glass]: 1.1 mm |
| Scintillator [G4_POLYSTYRENE]: 3 mm |
| air: 0.5 mm = 6.5 mm – others |

*No SiPM (no SiPM-on-tile design), no cooling, etc.

```
46        <layer repeat="Hcal_nlayers" vis="SeeThrough">
47          <slice material="FloatGlass" thickness="HcalSD_glass_anode_thickness" vis="Invisible"/>
48          <slice material="RPCGAS2"    thickness="HcalSD_sensitive_gas_gap" sensitive="yes" limits="cal_limits" vis="YellowVis"/>
49          <slice material="FloatGlass" thickness="HcalSD_glass_cathode_thickness" vis="Invisible"/>
50          <slice material="G4_POLYSTYRENE" thickness = "Hcal_scintillator_thickness" sensitive = "yes"   limits="cal_limits"  vis="CyanVis"   />
51          <slice material="Air"       thickness="Hcal_chamber_thickness - ( HcalSD_glass_anode_thickness + HcalSD_sensitive_gas_gap + HcalSD_glass_ca
52        </layer>
```