Standalone Geant4 simulation for Compton polarimeter

Zhengqiao Zhang

Motivation

- **We used EicRoot for our fast simulation of Compton polarimeter;**
- **For the IR study, EicRoot works fine but not flexible;**
- **The visualization on Rcf is quite slow if I work at home;**
- **People are discussing all the options (Fun4all, g4e, etc.) for the future detector simulation;**
- **There is no harm for us to try a pure Geant4 simulation;**

Geant4 is the Object-Oriented toolkit which provides functionalities required for simulations in HEP and other fields.

Benefits of Object-Orientation help you to realize a detector simulator which is

- Easy to develop and maintain
- Well modularized
- Readable and Understandable to the collaborators

Installation of Geant4

OS/Software Prerequisites:

Geant4 Toolkit Source Code;
macOS: Apple Clang (Xcode) 10 or higher;
CMake 3.8 or higher;
Qt User Interface and Visualization;

Building and Installing:

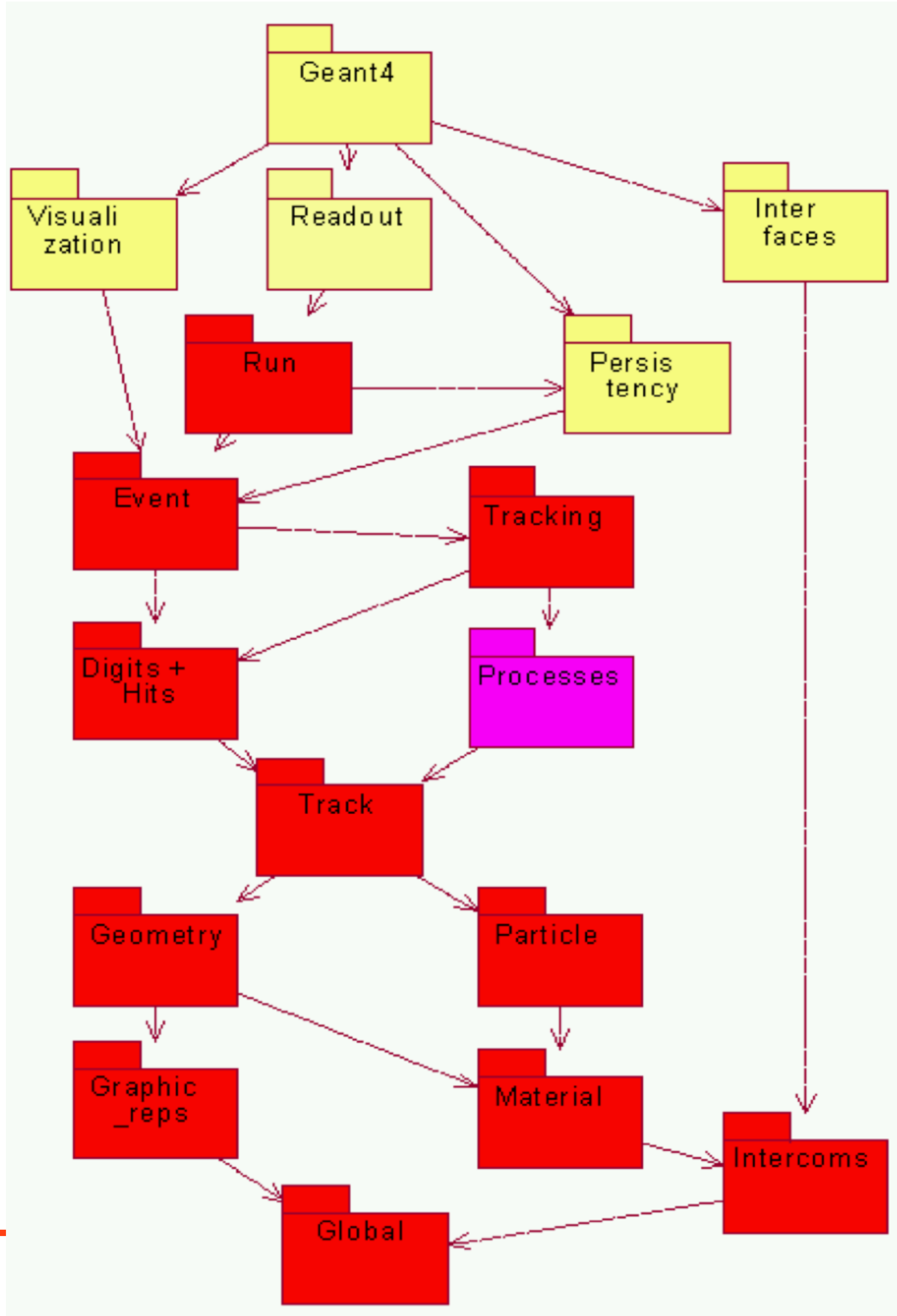
```
$ cd /path/to
$ mkdir geant4.10.06-build
$ ls
geant4.10.06  geant4.10.06-build
```

```
$ cd /path/to/geant4.10.06-build
$ cmake -DCMAKE_INSTALL_PREFIX=/path/to/geant4.10.06-install /path/to/geant4.10.06

$ cmake -DGEANT4_INSTALL_DATA=ON -DGEANT4_USE_OPENGL_X11=ON -DGEANT4_USE_XM=ON -DGEANT4_USE_QT=ON
-DGEANT4_USE_SYSTEM_CLHEP=ON  .

$ Make -j4
$ make install
$ . geant4.sh.  (source geant4.csh)
```

Geant4 work structure



CMakeLists.txt
README.md
build
data
events.dat
gui.mac
include
init_vis.mac
input
macro
output
polarimeter_magnet.dat
run.cxx
run.mac
src
vis.mac

How to run:

```
cd build
cmake ../
make
./run
```

Magnet field setup

```
G4RotationMatrix* yRot = new G4RotationMatrix;
yRot->rotateY(angle*1e-3*rad);

G4String nam_inner = fNam+"_inner";
G4Cons *shape_inner = new G4Cons(nam_inner, 0, r2, 0, r1, length/2, 0, 360*deg);

G4Material *mat_inner = G4NistManager::Instance()->FindOrBuildMaterial("G4_Galactic");
G4LogicalVolume *vol_inner = new G4LogicalVolume(shape_inner, mat_inner, nam_inner);
vol_inner->SetVisAttributes( G4VisAttributes::GetInvisible() );

G4UniformMagField *field = new G4UniformMagField(G4ThreeVector(0, bfield, 0));
G4FieldManager *fman =  new G4FieldManager();

fman->SetDetectorField(field);
fman->CreateChordFinder(field);
//fman->SetChordFinder(fChordFinder);
//fman->GetChordFinder()->SetDeltaChord(1e-7*meter);

vol_inner->SetFieldManager(fman, true);

//put the inner core to the top volume
new G4PVPlacement(yRot, G4ThreeVector(xpos,ypos, zpos), vol_inner, nam_inner, top, false, 0);

//cylindrical outer shape
G4Tubs *shape_outer = new G4Tubs(fNam+"_outer", 0., dout, length/2-1e-4*meter, 0., 360.*deg);

//magnet vessel around the inner magnetic core
G4SubtractionSolid *shape_vessel = new G4SubtractionSolid(fNam, shape_outer, shape_inner);

G4Material *mat_outer = G4NistManager::Instance()->FindOrBuildMaterial("G4_Fe");
G4LogicalVolume *vol_vessel = new G4LogicalVolume(shape_vessel, mat_outer, fNam);

//vessel visibility
G4VisAttributes *vis_vessel = new G4VisAttributes();
vis_vessel->SetColor(0, 0, 1); // blue
vis_vessel->SetLineWidth(2);
vis_vessel->SetForceSolid(true);
//vis_vessel->SetForceAuxEdgeVisible(true);
vol_vessel->SetVisAttributes(vis_vessel);

//put the magnet vessel to the top volume
new G4PVPlacement(yRot, G4ThreeVector(xpos, ypos, zpos), vol_vessel, fNam, top, false, 0);
```

```
ActionInitialization.cxx
BeamMagnetDipole.cxx
BeamMagnetQuadrupole.cxx
DetectorConstruction.cxx
EventAction.cxx
EventReader.cxx
GeneratorAction.cxx
MCEvent.cxx
ParticleReader.cxx
RootOut.cxx
RunAction.cxx
```

There are many propagation setup for the field in Geant4, we can have more study on this to find the most favorable one for us.

Read the magnets parameters

name	center_x	center_y	center_z	rin(z-in)	rin(z-out)	dout	length	angle	B	gradient
##	[m]	[m]	[m]	[m]	[m]	[m]	[m]	[mrad]	[T]	[T/m]
DB DB23_1	0.352014	0	-0.7563	0.12	0.12	0.5	4.22367	-29.669	-0.272916	0
QF QD12	0.285309	0	-3.7023	0.1	0.1	0.5	0.6	-22.714	0	-12.8444
DB DB23	0.238873	0	-6.6488	0.09	0.09	0.5	4.22367	-12.699	-0.272916	0
QF QF11	0.228737	0	-9.5956	0.09	0.09	0.5	0.6	-6.322	0	14.2077
QF QF12	0.228737	0	-20.319	0.085	0.085	0.5	0.6	-0.00	0	-6.87619
QF QF13	0.228737	0	-31.0425	0.1	0.1	0.5	0.6	-0.00	0	7.06014

We can use the same madx file in the EicRoot.

```
ActionInitialization.cxx
BeamMagnetDipole.cxx
BeamMagnetQuadrupole.cxx
DetectorConstruction.cxx
EventAction.cxx
EventReader.cxx
GeneratorAction.cxx
MCEvent.cxx
ParticleReader.cxx
RootOut.cxx
RunAction.cxx
```

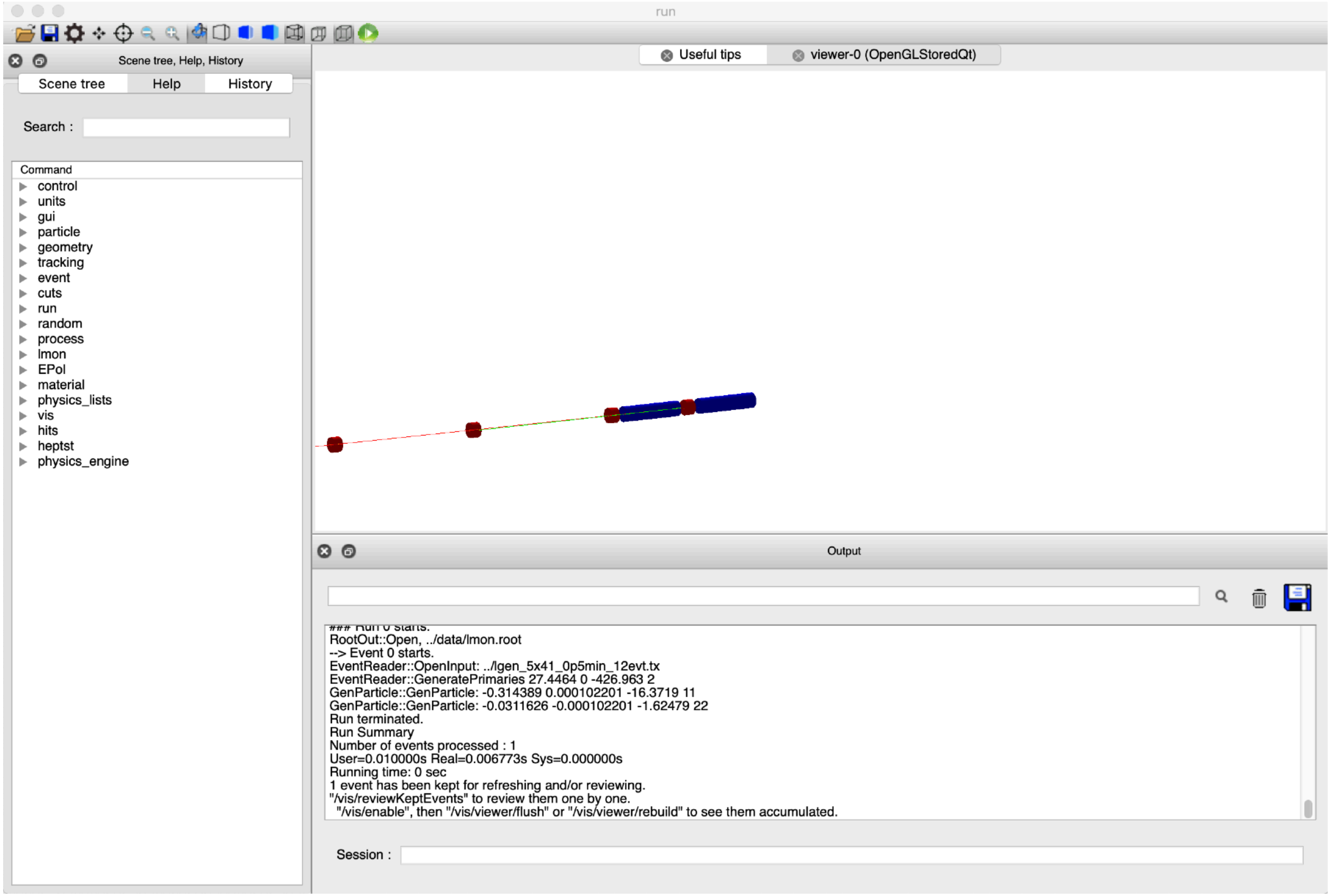
Read the events

```
EVENT: 1 2 27.4464 0  -426.963
TRACK:  11  -0.314389   0.000102201   -16.3719
TRACK:  22  -0.0311626   -0.000102201  -1.62479

EVENT: 2 2 27.4464 0  -426.963
TRACK:  11 -0.240207   5.16457e-05   -12.5151
TRACK:  22 -0.105345   -5.16457e-05   -5.48161
```

```
ActionInitialization.cxx
BeamMagnetDipole.cxx
BeamMagnetQuadrupole.cxx
DetectorConstruction.cxx
EventAction.cxx
EventReader.cxx
GenParticle.cxx
GeneratorAction.cxx
MCEvent.cxx
RootOut.cxx
RunAction.cxx
```

IR12 layout


run
Scene tree, Help, History
Scene tree
Help
History
Search :
Command
control
units
gui
particle
geometry
tracking
event
cuts
run
random
process
lmon
EPol
material
physics_lists
vis
hits
heptst
physics_engine
Useful tips
viewer-0 (OpenGLStoredQt)
Output
RootOut::Open, ../data/lmon.root
--> Event 0 starts.
EventReader::OpenInput: ../lgen_5x41_0p5min_12evt.tx
EventReader::GeneratePrimaries 27.4464 0 -426.963 2
GenParticle::GenParticle: -0.314389 0.000102201 -16.3719 11
GenParticle::GenParticle: -0.0311626 -0.000102201 -1.62479 22
Run terminated.
Run Summary
Number of events processed : 1
User=0.010000s Real=0.006773s Sys=0.000000s
Running time: 0 sec
1 event has been kept for refreshing and/or reviewing.
"/vis/reviewKeptEvents" to review them one by one.
"/vis/enable", then "/vis/viewer/flush" or "/vis/viewer/rebuild" to see them accumulated.
Session :

Thanks.