

Monte Carlo 3

2006/12/07

What is Geant ?

- Monte Carlo simulation tool for nuclear and particle physics
- Set up detectors
 - Sensitive detector which returns useful information such as energy deposit and hit position.
- Take into account physical processes which you register
 - EM process
 - Decay process
 - Hadronic process
- Transport particle according to the equation of motion at the field.
 - Electric field
 - Magnetic field

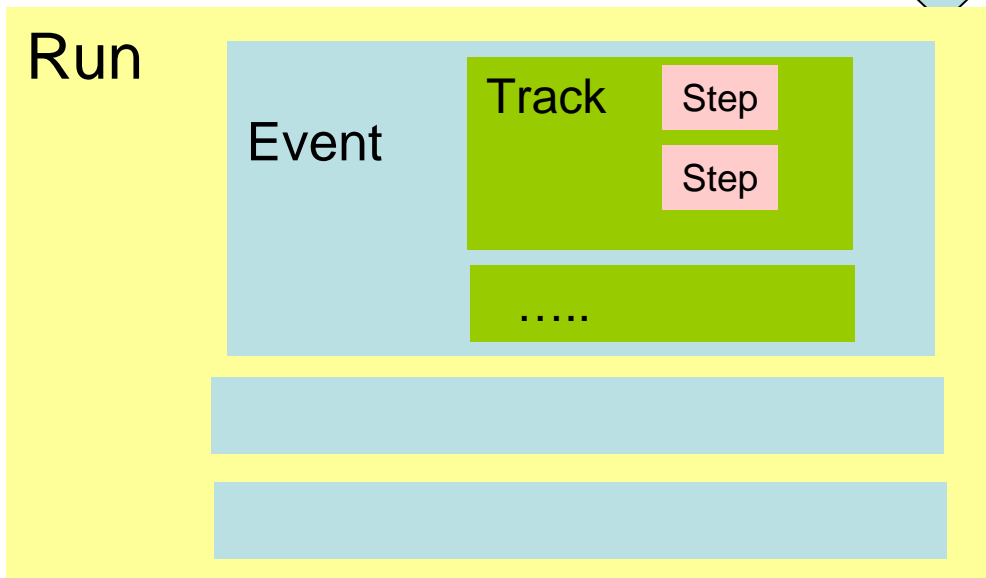
Preparation of GEANT

- We use GEANT4 (ver4.6.2.p02)
 - Installed at farm:/cern/geant4
- Set up of your environmental variable in .cshrc
 - LD_LIBRARY_PATH
 - setenv LD_LIBRARY_PATH /cern/geant4/lib/Linux-g++
- Copy example programs at farm:/home/miwa9/geant4/novice
 - cp -r /home/miwa9/geant4/novice .
- Go to the directory. There are 6 examples.

Execute Geant4!

- As a example, let's execute No3.
 - cd N03
- Compile the program
 - make (コンパイルするためのコマンド。ルールは Makefileに書いてある。)
- Execute the program
 - ./exampleN03
 - Event display will appear
 - Idle> /run/beamOn 1
 - Start simulation of 1 event

Overview of program



Start simulation
> beamOn #eventNv

Start simulation

- Detector set up
- Physics list taken into account

- One event consists of many tracks
- One Tracks consists of many steps

Registrar of each manager

- Example of ExN00
- exampleN00.cc (main function)
 - Define run manager
 - Register other managers which define “Detector”, “Physics” , “PrimaryGeneratorAction” and “EventAction”
 - The simulation is executed based on the registered processes.

```
// Construct the default run manager
G4RunManager * runManager = new G4RunManager;

// set mandatory initialization classes
ExN00DetectorConstruction* detector = new ExN00DetectorConstruction;
runManager->SetUserInitialization(detector);
runManager->SetUserInitialization(new ExN00PhysicsList);

// set user action classes
runManager->SetUserAction(new ExN00PrimaryGeneratorAction());
ExN00EventAction* eventaction = new ExN00EventAction;
runManager->SetUserAction(eventaction);
```

Detector setting (Material)

- We have to define a material of detector.
- G4Element (describes the properties of the atoms)
 - Atomic number
 - Atomic mass
- G4Material (describes the macroscopic properties of matter)
 - Density
 - State,
 - temperature

```
G4Element* H = new G4Element("Hydrogen",symbol="H" , z= 1., a= 1.01*g/mole);  
G4Element* C = new G4Element("Carbon" ,symbol="C" , z= 6., a= 12.01*g/mole);;
```

```
new G4Material("Aluminium", z=13., a=26.98*g/mole, density=2.700*g/cm3);  
new G4Material("liquidArgon", z=18., a= 39.95*g/mole, density= 1.390*g/cm3);  
new G4Material("Lead" , z=82., a= 207.19*g/mole, density= 11.35*g/cm3);
```

```
// define a material from elements. case 1: chemical molecule
```

```
G4Material* Sci =  
  new G4Material("Scintillator", density= 1.032*g/cm3, ncomponents=2);  
Sci->AddElement(C, natoms=9);  
Sci->AddElement(H, natoms=10);
```

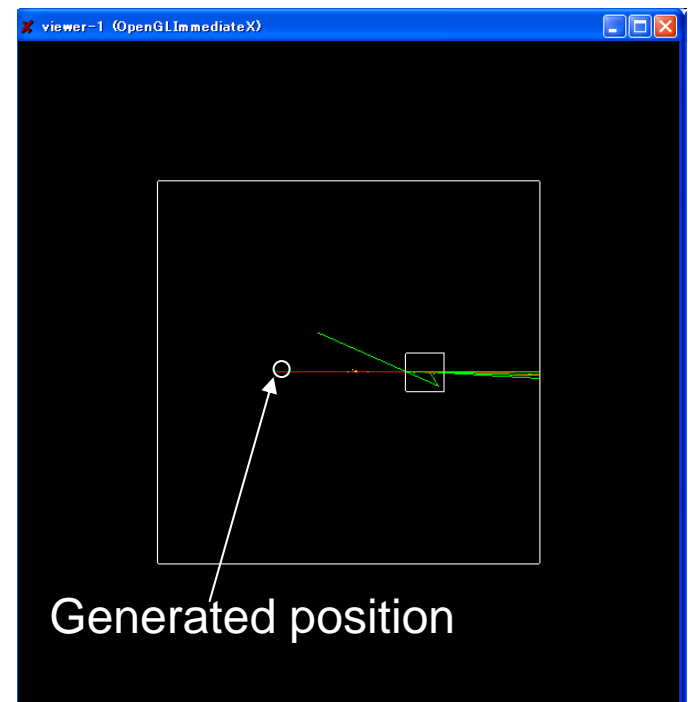

Primary Generation Action

- We have to specify how a primary event should be generated
- Kind of particle, energy, generated position, direction etc..
- G4ParticleGun generates primary particle(s) with a given momentum and position

```
G4int n_particle = 1;
particleGun = new G4ParticleGun(n_particle);

// default particle kinematic

G4ParticleTable* particleTable = G4ParticleTable::GetParticleTable();
G4String particleName;
G4ParticleDefinition* particle
    = particleTable->FindParticle(particleName="e-");
particleGun->SetParticleDefinition(particle);
particleGun->SetParticleMomentumDirection(G4ThreeVector(1.,0.,0.));
particleGun->SetParticleEnergy(50.*MeV);
particleGun->SetParticlePosition(G4ThreeVector(-20.0*cm,0.*cm,0.*cm));
particleGun->GeneratePrimaryVertex(anEvent);
```



Commands of Geant4

- Start a run
 - Idle> /run/beamOn 1000
 - 1000 events are generated
- Execute macro file
 - Idle> /control/execute vis.mac
- Change view point
 - Idle> /vis/viewer/set/viewpointThetaPhi 45 45
- Finish Geant
 - Idle> exit

Problem

- Execute these example programs
- Modify ExN00
 - Change material of calorimeter
 - Scintillator → C, Fe, H₂O, Pb etc.
 - Change size and shape of calorimeter
 - Box → Tube etc.
 - Change primary particle
 - “e-” → “pi-”, “kaon-”, “proton”
 - Energy