

# HPFBU 2014 TOKAT KIŐ OKULU

## GEANT4 ÇALIŐMASI

Taylan YETKİN - [tyetkin@yildiz.edu.tr](mailto:tyetkin@yildiz.edu.tr)  
Adnan KILIÇ - [adnank@uludag.edu.tr](mailto:adnank@uludag.edu.tr)

# GEANT4' TEN ALGIC CEVABI ALMAK

UserSteppingAction() içerisinde adım adım alınan biriktirilir.

Bilgi UserEventAction::AddEnergy() fonksiyonu ile EventAction'a aktarılır.

Cevap, UserEventAction::EndOfEventAction() fonksiyonu içerisinde ekrana bastırılır.

Duyar Hacim ve Hit

Primitive Scorer Nedir

Analiz

## UserSteppingAction.cc

```
// energy deposit
G4double edep = step->GetTotalEnergyDeposit();
If (.....)
EventAction->AddEnergy(edep);
```

## UserSteppingAction.hh

```
private:
UserEventAction* EventAction;
```

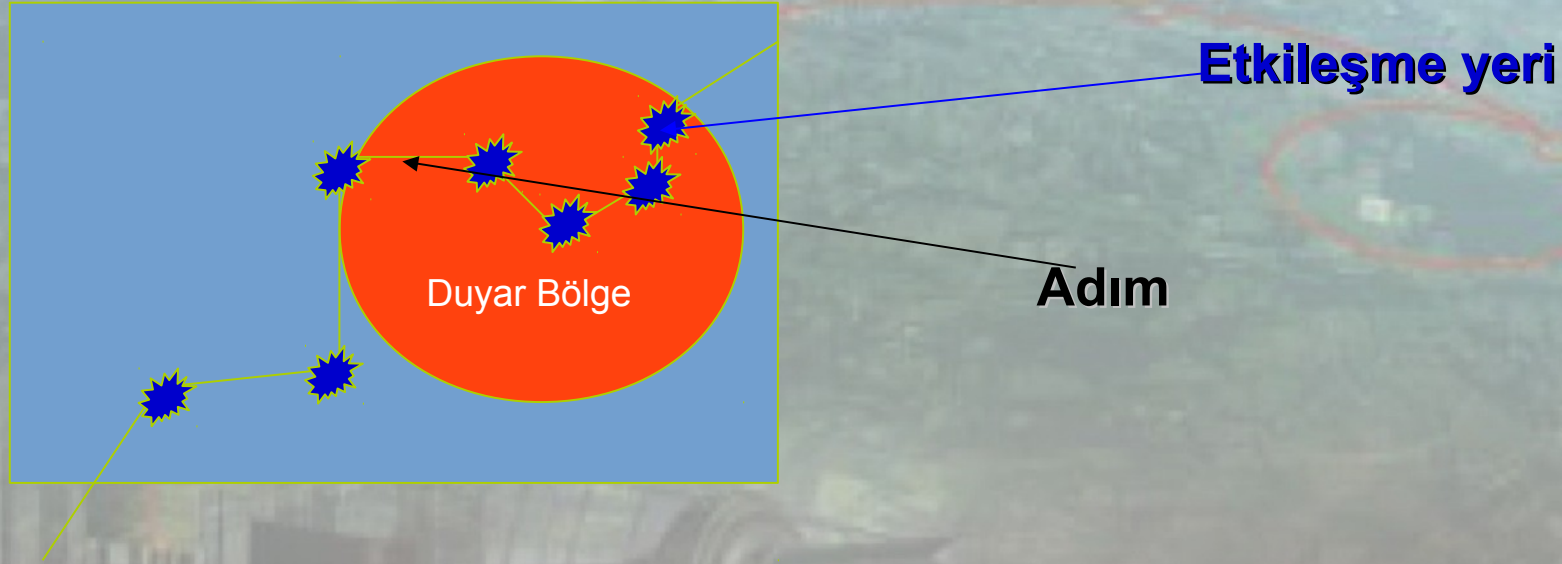
## UserEventAction.hh

```
private:
G4double fEdep;
AddEnergy(G4double de,) {
    fEdep += de;
}
```

## UserEventAction.cc

```
void UserEventAction::BeginOfEventAction(const G4Event* event)
{ // initialisation per event
  fEdep = 0.; }
void UserEventAction::EndOfEventAction(const G4Event* event)
{ G4cout<<"Depolanan Enerji :"<<G4BestUnits(fEdep, "Energy")<<G4endl;
}
```

# HIT

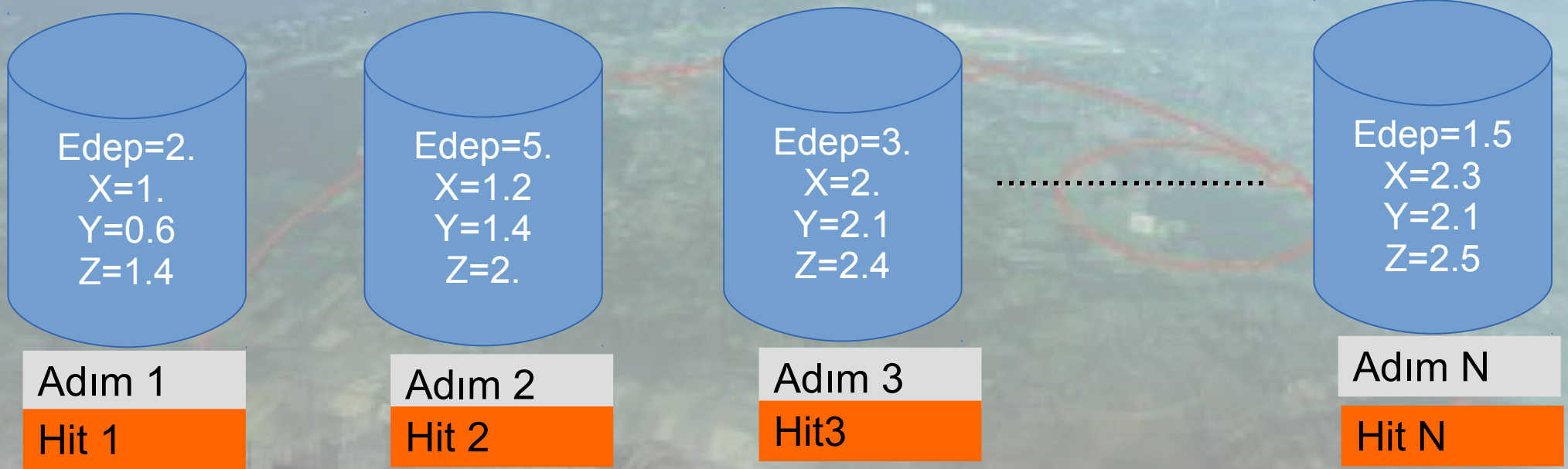


Bir hit, duyar algıç içerisinde gerçekleşen etkileşmenin bir fotoğrafıdır. Hit sınıfı, kullanıcı tarafından G4VHit temel sınıfından türetilen bir sınıftır. Türetilen Hit sınıfınızda istenilen değişken ve/veya bilgiler depolanabilir:

## Ne Gibi ?

- Konum, zaman ve bir adımdaki enerji kaybı
- Verilen bir track'ın momentumu, enerjisi, konumu, bulunduğu hacimi ve parçacığın tipi vb.

# VERİNİN KAYDEDİLMESİ



Her bir adımda **ProcessHits()** çağırılır.



Hit Biriktirisi =vector<Hit>

G4THitsCollection<UserHit>

**UserHit\*** bir dizi !

**Bir Hit başına bit Hit biriktirisi!**



EndOfEventAction()

G4Event::G4HCofThisEvent

//Duyar Algıç-Hit kullanımı)

examples/basic/B4/B4c -----İlk düzey

examples/basic/B5/ -----İlk düzey

# PRIMITIVE SCORER

G4MultiFunctionalDetector, G4VSensitiveDetector'den türetilmiş bir sınıftır.

Herbir G4VPrimitiveSensitivity herbir fiziksel hacim için bir fiziksel büyüklüğü biriktirir.

Bu yaklaşımda duyar algı ve hit sınıfları oluşturmaya ihtiyaç olmaz!

**İz uzunluğu:** GmG4PSTrackLength, GmG4PSPassageTrackLength

**Depolanan Enerji:** GmG4PSEnergyDeposit

**Akım/Akı:** GmG4PSFlatSurfaceCurrent, GmG4PSCylinderSurfaceCurrent, GmG4PSSphereSurfaceCurrent, GmG4PSPassageCellCurrent, GmG4PSFlatSurfaceFlux, GmG4PSCylinderSurfaceFlux, GmG4PSSphereSurfaceFlux, GmG4PSCellFlux, GmG4PSPassageCellFlux

**Diğerleri:** GmG4PSCylinderSurfaceCurrent, GmG4PSCylinderSurfaceFlux, GmG4PSFlatSurfaceCurrent, GmG4PSFlatSurfaceFlux, GmG4PSSphereSurfaceCurrent, GmG4PSSphereSurfaceFlux, GmG4PSTrackCounter

# SÜZGEÇ SCORER'LAR

Hangi tür parçacıkları score etmek istiyorsunuz ???

**Sadece yüklü/ yüksüz parçacıklar:** G4SDChargedFilter, G4SDNeutralFilter

**Belirli kinetik enerji aralığındaki parçacıklar:** G4SDKineticEnergyFilter

**Belirli parçacık türleri :** G4SDparticleFilter

**Belirli kinetik enerji aralığındaki parçacık türleri :** G4SDParticleWithEnergyFilter

<http://geant4.web.cern.ch/geant4/G4UsersDocuments/UsersGuides/ForApplicationDeveloper/html/Detector/hit.html>

Detaylı Bilgi için ziyaret edin....



# PRIMITIVE SCORER

//(primitive scorer'ların kullanımı)

examples/basic/B4/B4d -----İlk düzey

examples/extended/runAndEvent/RE02 -----orta düzey

examples/extended/runAndEvent/RE03 -----orta düzey

# ANALİZ

ROOT ile çalışmak için Analiz.hh isimli bir başlık dosyası oluştur.

```
#ifndef Analysis_h
#define Analysis_h 1

#include "g4root.hh"

#endif
```

## EventAction.cc

```
void EventAction::EndOfEventAction(const G4Event* event)
{
    .....
    // get analysis manager
    G4AnalysisManager* analysisManager = G4AnalysisManager::Instance();

    //fill Histogram
    // fill histograms
    analysisManager->FillH1(1, EnergyDeposit/MeV);
    // fill ntuple
    analysisManager->FillNtupleDColumn(0,EnergyDeposit/MeV);
    analysisManager->AddNtupleRow();
}
```

<http://g4course2010.web.cern.ch/g4course2010/task0/overview.html>

Alıştırma ve Kullanışlı Uygulamalar için ziyaret edin .....

Intro Task 0 Task 1 Task 2 Task 3 Task 4 Task 5 Task 6 Reference

Main Page Files

Task 0

### Overview of Geant4 tasks

Geant4 Users' Tutorial  
CERN, 15-19 February 2010

#### Geant4 Tasks

**Task 0 - Introduction to Geant4**

- how to compile and run the example programs
- introduction to basic UI commands
- introduction to visualization
- where to find help

**Task 1 - Geometry**

- using and defining materials
- constructiong a detector using solids and placements
- replica, GDML
- advanced features

**Task 2 - Primary Particles and User Interface**

- Particle gun
- HEPmc
- General Particle source

**Task 3 - Electromagnetic physics**

<http://hypernews.slac.stanford.edu/HyperNews/geant4/cindex>

Soru ve cevaplarınız geliştiricilerle paylaşmak için ziyaret edin .....

GEANT4 at hypernews.slac.stanford.edu Forum List by Category Not Logged In ([login](#))

# Geant 4

<a href="#">Forums by Category</a>	<a href="#">Recent Postings</a>	<a href="#">Member Info</a>	<a href="#">Overview</a>
<a href="#">Forums by Time Order</a>	<a href="#">Search in Forums</a>	<a href="#">Members List</a>	<a href="#">Contact Admin</a>
<a href="#">Request a New Forum</a>	<a href="#">Subscribe to Forums</a>	<a href="#">New Member</a>	

[Page Help](#)

---

**Category: Applications**

<a href="#">Educational Applications</a>	<a href="#">Industrial instruments</a>	<a href="#">Medical Applications</a>	<a href="#">Space Applications</a>
--	--	--------------------------------------	------------------------------------

---

**Category: Control of runs, events, tracks, particles**

<a href="#">Event and Track Management</a>	<a href="#">Multithreading</a>	<a href="#">Particles</a>	<a href="#">Run Management</a>
--	--------------------------------	---------------------------	--------------------------------

---

**Category: Experimental Setup**

<a href="#">Biasing and Scoring</a>	<a href="#">Fields: Magnetic and Otherwise</a>	<a href="#">Geometry</a>	<a href="#">Hits, Digitization and Pileup</a>
-------------------------------------	--	--------------------------	---

---

**Category: General matters**

<a href="#">Documentation and Examples</a>	<a href="#">HyperNews System Announcements</a>	<a href="#">Hypernews Testing</a>	<a href="#">Installation and Configuration</a>
<a href="#">User Requirements</a>			

---

**Category: Interfaces**

<a href="#">(Graphical) User Interfaces</a>	<a href="#">Analysis</a>	<a href="#">Persistency</a>	<a href="#">Visualization</a>
---	--------------------------	-----------------------------	-------------------------------

---

**Category: Physics**

<a href="#">Biasing and Scoring</a>	<a href="#">Electromagnetic Processes</a>	<a href="#">Fast Simulation, Transportation &amp; Others</a>	<a href="#">Hadronic Processes</a>
<a href="#">Physics List</a>	<a href="#">Processes Involving Optical Photons</a>		

# SANAL BİLGİSAYARDA YENİLEME

```
cd /hep/geant4data
rm -fv *tar.gz
unrar e ~/Desktop/LINUX_SHARED/*.tar.gz .
for f in ~/Desktop/LINUX_SHARED/*.tar.gz; do tar -xzvf $f ; done

rm -fv ~/Desktop/LINUX_SHARED/*.tar.gz

echo "#Geant4 env" >> $HOME/.bashrc
echo "source $HOME/hepWork/geant4.10.00/share/Geant4-10.0.0/geant4make/geant4make.sh" >>$HOME/.bashrc
echo "export Geant4_DIR=$HOME/hepWork/geant4.10.00/lib/Geant4-10.0.0" >> $HOME/.bashrc
```

# UYGULAMA ÖRNEĞİ

Kullanıcı etkileşimli mod



```
cd $HOME/hepWork/geant4.10.00/share/Geant4-10.0.0/examples
```

```
cd basic/B4/B4a
```

```
mkdir build
```

```
cd build
```

```
cmake ..
```

```
Make
```

```
./exampleB4a
```

```
idle>help
```

```
Idle>/gun/particle e-
```

```
Idle>/gun/direction 0 0 1
```

```
Idle>/gun/energy 1 GeV
```

```
Idle>/run/beamOn 10
```

```
Idle>/control/execute run2.mac
```

batch mod

```
./exampleB4a run2.mac  
veya  
./exampleB4a -m run2.mac
```

# Primitive Scorer Uygulaması

```
cd examples/B4d/
```

```
gedit exampleB4d.cc &
```

KOMUT TABANLI SCORER

```
#include "G4ScoringManager.hh"
```

```
int main() {
```

```
G4RunManager* runManager = new G4RunManager;
```

```
G4ScoringManager* scoringManager = G4ScoringManager::GetScoringManager();
```

UI komutları /score/ klasörü içerisinde...

```
/score/create/boxMesh boxMesh_1
#
/score/mesh/boxSize 5 5 7.5 cm
/score/mesh/nBin 30 30 30
#
/score/quantity/energyDeposit eDep
/score/filter/particle eMinusFilter e-
/score/filter/particle ePlusFilter e+

/score/close

/score/list

/vis/disable
/control/verbose 2
/run/verbose 1
/gun/particle e-
/run/beamOn 20
/vis/enable
#
# drawing projections
#
/score/drawProjection boxMesh_1 eDep
```



```
cd $HOME/hepWork/geant4.10.00/share/Geant4-10.0.0/examples
```

```
cd basic/B4/B4d
```

```
mkdir build
```

```
cd build
```

```
cmake ..
```

```
Make
```

```
./exampleB4d
```

```
idle>help
```

```
/scorer/ seçeneğini görmelisiniz!
```

```
Idle>/control/execute scorer.mac
```



# BİR PARÇA UYGULAMA

examples/basic/B4 (B4a, B4c veya B4d) üzerinde gerekli değişiklikleri yaparak,

Soğurucu tabaka Cu ve aktif tabaka Si

- 1) Cu malzemesi tanımla
- 2) Si malzemesi tanımla  
( NIST malzeme kütüphanesi ile değil!!!)
- 3) Bir Si aktif tabaka kalınlığı 500 um
- 4) Pb ile aynı radyasyon uzunluğuna sahip olunacak şekilde, soğurucunun Cu olması durumunda toplam soğurucu kalınlığını belirle
- 5) Bu Cu soğurucu kalınlığını uygun şekilde tabakalara ayır
- 6 ) Cu olması durumunda kalorimetrenin soğurucu ve aktif tabakalarında depolanan enerji dağılımlarını ayrı ayrı elde edin
- 7) Pb ve Cu enerji dağılımlarını aynı histogram üzerinde gösterin.

# GEANT4 MALZEME TANIMI

```
#include "G4Material.hh"  
...  
G4double density=1.390*g/cm3;  
G4double a=39.95*g/mole;  
G4double z=18.;  
G4String name;  
...  
G4Material* LAr = new G4Material(name="Liquid Argon", z, a, density);
```

```
#include "G4Element.hh"  
#include "G4Material.hh"  
...  
G4double a=1.01*g/mole;  
G4double z;  
G4String name,symbol;  
G4Element* H = new G4Element(name="Hydrogen", symbol="H",z=1.,a);  
  
a=16.0*g/mole;  
G4Element* O = new G4Element(name="Oxygen", symbol="O",z=8.,a);  
  
G4double density=1.000*g/cm3;  
G4int ncomponent,natoms;  
G4Material* H2O = new G4Material(name="Water",density,ncomponents=2);  
H2O->AddElement(H,natoms=2);  
H2O->AddElement(O,natoms=1);
```

```
#include "PrimaryGeneratorAction.hh"  
#include "PrimaryGeneratorMessenger.hh"
```

```
#include "G4Event.hh"  
#include "G4HEPEvtInterface.hh"
```

```
#include "G4ParticleGun.hh"  
#include "G4ParticleTable.hh"  
#include "G4ParticleDefinition.hh"
```

```
PrimaryGeneratorAction::PrimaryGeneratorAction()  
{  
  const char* filename = "PYTHIA_EVENT.data";  
  HEPEvt = new G4HEPEvtInterface(filename);  
  
  G4int n_particle = 1;  
  G4ParticleGun* fParticleGun = new G4ParticleGun(n_particle);  
  G4ParticleTable* particleTable = G4ParticleTable::GetParticleTable();  
  G4String particleName;  
  G4ParticleDefinition* particle  
    = particleTable->FindParticle(particleName="proton");  
  fParticleGun->SetParticleDefinition(particle);  
  fParticleGun->SetParticleMomentumDirection(G4ThreeVector(0.,1.,0.));  
  fParticleGun->SetParticleEnergy(7*TeV);  
  fParticleGun->SetParticlePosition(G4ThreeVector(0.*cm,0.*cm,0.*cm));  
  particleGun = fParticleGun;  
  
  messenger = new PrimaryGeneratorMessenger(this);  
  useHEPEvt = true;  
  nnprim = 0;  
}
```

```
PrimaryGeneratorAction::~~PrimaryGeneratorAction()  
{delete HEPEvt;  
  delete particleGun;  
  delete messenger;}  
void PrimaryGeneratorAction::GeneratePrimaries(G4Event* anEvent)  
{ if(useHEPEvt)  
  { HEPEvt->GeneratePrimaryVertex(anEvent); }  
  else  
  { particleGun->GeneratePrimaryVertex(anEvent); }  
}
```

# G4HEPEvtInterface

## PYTHIA6 DOSYASI