Hands-On on physics

« microdosimetry »

Sébastien Incerti

2015 Geant4-DNA Tutorial

Hiroshima, Japan

August 24-25

microdosimetry

- We are going to try the microdosimetry extended example which makes use of
  Geant4 & Geant4-DNA Physics processes and models
  - More tricky than « dnaphysics »: you need to construct your own PhysicsList class

- This is an extended example and it is located in
  $G4INSTALL/examples/extended/electromagnetic/dna

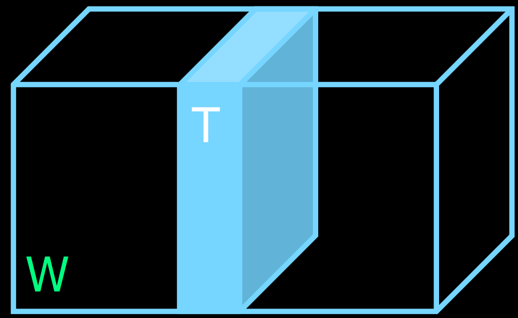
microdosimetry

- This example shows the multi-scale combination of
  - CONDENSED EM Physics processes
    - Geant4 Standard EM
  - DISCRETE EM Physics processes
    - Geant4-DNA
  - for protons/hydrogen
  - in two regions (the « World » and the « Target »)

- Useful when the user is interested in simulating efficiently high energy incident particles with Geant4 EM Standard Physics, providing a space phase input for Geant4-DNA simulations applied to much smaller volumes

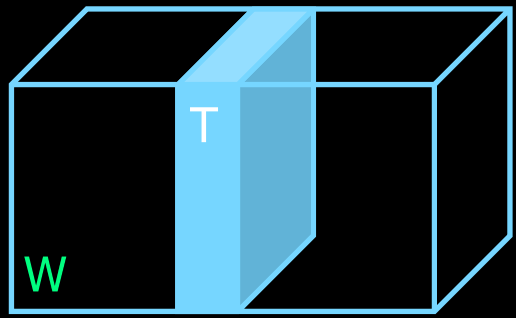
- Also shows how to create a process to kill particles below a certain energy threshold
  - G4ElectronCapture

Geometry



- A 50 μm thick « Target » volume placed in a 1 mm thick « World » mother volume
- Both contain liquid water only
- We would like to use, for electrons:
  - Geant4 EM Standard models in the « World »
  - Geant4-DNA models in the « Target » (below 1 MeV)

How to ?



- 3 main steps are needed
  - Create a Region for the « Target » volume in DetectorConstruction.cc
  - Define Physics for the « World » in PhysicsList.cc
  - Define Physics for the « Target » in PhysicsList.cc

1) Create a Region for the « Target » volume in DetectorConstruction.cc

```
fRegion = new G4Region("Target");

 G4ProductionCuts* cuts = new G4ProductionCuts();

 G4double defCut = 1*nanometer;
 cuts->SetProductionCut(defCut,"gamma");
 cuts->SetProductionCut(defCut,"e-");
 cuts->SetProductionCut(defCut,"e+");
 cuts->SetProductionCut(defCut,"proton");

 fRegion->SetProductionCuts(cuts);

 fRegion->AddRootLogicalVolume(logicTarget);
```

2) Define Physics for the « World » in PhysicsList.cc

- Let's consider the case of electrons
- In the loop over particles, we ACTIVATE Geant4 EM standard processes
  - These will be ACTIVE IN THE WORLD

```
if (particleName == "e-") {

  // STANDARD msc is active in the world
  G4eMultipleScattering* msc = new G4eMultipleScattering();
  pmanager->AddProcess(msc, -1, 1, 1);

  // STANDARD ionisation is active in the world
  G4eIonisation* eion = new G4eIonisation();
  eion->SetEmModel(new G4MollerBhabhaModel(), 1);
  pmanager->AddProcess(eion, -1, 2, 2);
```

Define Physics for the « World »

- then INACTIVATE Geant4-DNA processes using a G4DummyModel and the SetModel method.

They will be INACTIVE IN THE WORLD.

```
// DNA elastic is not active in the world
G4DNAElastic* theDNAElasticProcess = new G4DNAElastic("e-_G4DNAElastic");
theDNAElasticProcess->SetModel (new G4DummyModel(),1);
pmanager->AddDiscreteProcess(theDNAElasticProcess);
// DNA excitation is not active in the world
G4DNAExcitation* dnaex = new G4DNAExcitation("e-_G4DNAExcitation");
dnaex->SetModel (new G4DummyModel(),1);
pmanager->AddDiscreteProcess(dnaex);
// DNA ionisation is not active in the world
G4DNAIonisation* dnaioni = new G4DNAIonisation("e-_G4DNAIonisation");
dnaioni->SetModel (new G4DummyModel(),1);
pmanager->AddDiscreteProcess(dnaioni);
// DNA attachment is not active in the world
G4DNAAttachment* dnaatt = new G4DNAAttachment("e-_G4DNAAttachment");
dnaatt->SetModel (new G4DummyModel(),1);
pmanager->AddDiscreteProcess(dnaatt);
// DNA vib. excitation is not active in the world
G4DNAVibExcitation* dnavib = new G4DNAVibExcitation("e-_G4DNAVibExcitation");
dnavib->SetModel (new G4DummyModel(),1);
pmanager->AddDiscreteProcess(dnavib);
```

In addition (not mandatory), kill all very low energy electrons

```
// THE FOLLOWING PROCESS WILL KILL ALL ELECTRONS BELOW A SELECTED ENERY THRESHOLD

// Capture of low-energy e-

G4ElectronCapture* ecap = new G4ElectronCapture("Target", 5.1*eV);

pmanager->AddDiscreteProcess(ecap);
```

3) Define Physics for the « Target »

- INACTIVATE EM Standard processes for electrons in the « Target » Region below 1 MeV

  by registering corresponding EM Standard models to the G4EmConfigurator and use the

  SetActivationLowEnergyLimit and SetExtraEmModel methods

```
G4EmConfigurator* em_config = G4LossTableManager::Instance()->EmConfigurator();
 G4VEmModel* mod;

 mod = new G4UrbanMscModel();
 mod->SetActivationLowEnergyLimit(1*MeV);
 em_config->SetExtraEmModel("e-","msc",mod,"Target");

 mod = new G4MollerBhabhaModel();
 mod->SetActivationLowEnergyLimit(1*MeV);
 em_config->SetExtraEmModel("e-","eIoni",mod,"Target",0.0,100*TeV,
  new G4UniversalFluctuation());
```

Specify
the low enery limit
of activation you need

Specify particle name, process
name, model and region
name
and energy interval
(option if more than one
models are used)

Define Physics for the Target

- ACTIVATE Geant4-DNA processes in the TARGET Region by registering the corresponding Geant4-DNA models to the G4EmConfigurator and using the SetExtraEmModel method

```
mod = new G4DNAChampionElasticModel();
em_config->SetExtraEmModel("e-","e-_G4DNAElastic",mod,"Target",0.,1*MeV);

mod = new G4DNABornIonisationModel();
em_config->SetExtraEmModel("e-","e-_G4DNAIonisation",mod,"Target",11*eV,1*MeV);

mod = new G4DNABornExcitationModel();
em_config->SetExtraEmModel("e-","e-_G4DNAExcitation",mod,"Target",9*eV,1*MeV);

mod = new G4DNAMeltonAttachmentModel();
em_config->SetExtraEmModel("e-","e-_G4DNAAttachment",mod,"Target",4*eV,13*eV);

mod = new G4DNASancheExcitationModel();
em_config->SetExtraEmModel("e-","e-_G4DNAVibExcitation",mod,"Target",2*eV,100*eV);
```

Specify energy range of applicability

Don't forget to ALWAYS check at run time that EM standard and DNA processes are not active simultaneoulsy or that the processes that you want to activate are indeed really active if this is the case, tune your energy limits …

SUMMARY

Selecting limits of models

C++ G4EmConfigurator* em_config = G4LossTableManager::Instance()->EmConfigurator(); G4VEmModel* mod = new G4DNABornIonisationModel();	Displayed at initialization time	Used at run time for transport
em_config ->**SetExtraEmModel**("e-","e_G4DNAIonisation", mod,"Target",X*MeV,Y*MeV);	Emin = 0 MeV Emax = Y MeV	Emin = inner low limit of model (11 eV) Emax=inner max limit of model (1 MeV)
mod ->**SetActivationLowEnergyLimit**(W*MeV);	Emin = W MeV	Emin = W MeV
mod ->**SetActivationHighEnergyLimit**(Z*MeV);	Emax = Z MeV	Emax = Z MeV

Make sure that W is > than the model's inner low energy limit and that Z is < than the model's inner high energy limit

microdosimetry

- Copy the microdosimetry extended example to your local directory, create your build directory and compile microdosimetry

```
cd
cp –R $G4EXAMPLES/examples/extended/medical/dna/microdosimetry .
mkdir build-microdosimetry
cd build-microdosimetry
cmake ../microdosimetry
make
```

- Run microdosimetry

```
./microdosimetry -mac microdosimetry.in -mt 2 -out microdosimetry.root
```

2 protons of 5 MeV are shot

  - Results are saved in microdosimetry_t*.root files

- Results can be analyzed using ROOT

```
root plot.C
```

Output of microdosimetry

Numbering of processes from SteppingAction class

e-_G4DNAElastic	11
e-_G4DNAExcitation	12
e-_G4DNAIonisation	13
e-_G4DNAAttachment	14
e-_G4DNAVibExcitation	15
eCapture	16
proton_G4DNAExcitation	17
proton_G4DNAIonisation	18
proton_G4DNAChargeDec.	19
hydrogen_G4DNAExcitation	20
hydrogen_G4DNAIonisation	21
hydrogen_G4DNAChargeInc.	22
alpha_G4DNAExcitation	23
alpha_G4DNAIonisation	24
alpha_G4DNAChargeDec.	25
alpha+_G4DNAExcitation	26
alpha+_G4DNAIonisation	27
alpha+_G4DNAChargeDec.	28
alpha+_G4DNAChargeInc.	29
helium_G4DNAExcitation	30
helium_G4DNAIonisation	31
helium_G4DNAChargeInc.	32

Eg. 1 proton of 5 MeV