Subject: Re: Segmentation Violation when simulating events with run_sim1.C
Posted by Marius Mertens on Wed, 04 Feb 2009 13:47:34 GMT
View Forum Message <> Reply to Message

Hi Stefano,

these are the contents of my modified version of the macro.

Toggle Spoiler

```
// Macro for running Cbm  with Geant3  or Geant4 (M. Al-Turany , D. Bertini)
// Modified 22/06/2005 D.Bertini

Mrun_sim1_PndFlatGen(Int_t numevents, TString basefile)
{
  TStopwatch timer;
  timer.Start();
  gDebug=0;
  // Load basic libraries
  gROOT->LoadMacro("$VMCWORKDIR/gconfig/rootlogon.C");
  rootlogon();

  CbmRunSim *fRun = new CbmRunSim();

  // set the MC version used
  // ------------------------

  fRun->SetName("TGeant4");
  // Choose the Geant Navigation System
  // fRun->SetGeoModel("G3Native");

  //fRun->SetOutputFile("testsimu10.root");
  fRun->SetOutputFile(basefile);


  // Set Material file Name
  //----------------------

  fRun->SetMaterials("media_pnd.geo");

  // Create and add detectors
  //-------------------------

  CbmModule *Cave= new PndCave("CAVE");
  Cave->SetGeometryFileName("pndcave.geo");
  fRun->AddModule(Cave);

  CbmModule *Magnet= new PndMagnet("MAGNET");
  Magnet->SetGeometryFileName("FullSolenoid.root");
  fRun->AddModule(Magnet);

  CbmModule *Dipole= new PndMagnet("MAGNET");
  Dipole->SetGeometryFileName("dipole.geo");
```

```
  fRun->AddModule(Dipole);


  CbmModule *Pipe= new PndPipe("PIPE");
  //Pipe->SetGeometryFileName("pipebeamtarget.geo");
  fRun->AddModule(Pipe);

  /*
    CbmDetector *Stt = new CbmStt("STT",kTRUE);
    Stt->SetGeometryFileName("stt24.geo"); // 14 = 1 solo 2layer pablo01.geo 3 layers stt24
    fRun->AddModule(Stt);
  */
  /*
    CbmDetector *Stt= new CbmStt("STT", kTRUE);
    Stt->SetGeometryFileName("straws_axial.geo");
    fRun->AddModule(Stt);
  */
/*
  CbmDetector *Stt= new PndStt("STT", kTRUE);
  Stt->SetGeometryFileName("straws_skewed_blocks_35cm.geo");
  fRun->AddModule(Stt);
*/

  CbmDetector *Tpc = new PndTpcDetector("TPC", kTRUE);
  Tpc->SetGeometryFileName("tpc.geo");
  fRun->AddModule(Tpc);

  CbmDetector *Mvd = new PndMvdDetector("MVD", kTRUE);
  //Mvd->SetGeometryFileName("MVD_v1.0_woPassiveTraps.root");
  Mvd->SetGeometryFileName("MVD_v1_2addDisks.root");

  fRun->AddModule(Mvd);

  PndEmc *Emc = new PndEmc("EMC",kTRUE);
  Emc->SetGeometryFileNameDouble("emc_module1245.dat","emc_module3new.root "); // if
you want to use new geometry for FwEndCap
  fRun->AddModule(Emc);

  CbmDetector *Tof = new PndTof("TOF",kTRUE);
  Tof->SetGeometryFileName("tofbarrel.geo");
  fRun->AddModule(Tof);

  CbmDetector *Drc = new PndDrc("DIRC", kTRUE);
  Drc->SetGeometryFileName("dirc.geo");
  fRun->AddModule(Drc);

/*
 CbmDetector *Muo = new PndMdt("MDT",kTRUE);
  Muo->SetGeometryFileName("muopars.root");
  fRun->AddModule(Muo);
*/
```

```
  CbmDetector *Dch = new PndDchDetector("DCH", kTRUE);
  Dch->SetGeometryFileName("dch.root");
  fRun->AddModule(Dch);

  // Create and Set Event Generator
  //-----------------------------

  CbmPrimaryGenerator* primGen = new CbmPrimaryGenerator();
  fRun->SetGenerator(primGen);

  PndFlatParticleGenerator pndFlatGen;
pndFlatGen.SetPDGType(211);
//pndFlatGen.SetMultiplicity(10);
pndFlatGen.SetPRange(1, 1, 1);
pndFlatGen.SetPhiRange(0, 359.8, 0.1);
pndFlatGen.SetThetaRange(20, 20, 1);

primGen->AddGenerator(&pndFlatGen);


/*
    //EvtGen Generator
    CbmEvtGenGenerator* evtGen = new
    //CbmEvtGenGenerator("/home/ralfk/Pandaroot/pandaroot/macro/mvd/output.evt ");
    //CbmEvtGenGenerator("
/private/mertens/fairsoft/cbmsoft/pandaroot/pgenerators/EvtGen/lambdalam
bdabar_15gev_100k.evt ");
    CbmEvtGenGenerator(eventfile);
    primGen->AddGenerator(evtGen);
*/

  /*  // Box Generator
      CbmBoxGenerator* boxGen = new CbmBoxGenerator(13, 200); // 13 = muon; 1 = multipl.
      //  boxGen->SetPRange(1.,1.1); // GeV/c
      boxGen->SetPtRange(1.,1.); // GeV/c
      boxGen->SetPhiRange(0., 360.); // Azimuth angle range [degree]
      boxGen->SetThetaRange(0., 90.); // Polar angle in lab system range [degree]
      boxGen->SetXYZ(0., 0., 0.); // mm o cm ??
      primGen->AddGenerator(boxGen);
  */
  /*
      // proton 2212 pi+ 211 pi- -211
      Double_t randx, randy;
      for (Int_t n =0; n<10; n++){
        randx= gRandom->Gaus(0,1);
        randy= gRandom->Gaus(0,1);
        CbmParticleGenerator* partGen = new CbmParticleGenerator(2212, 1, 0.3*randx,
0.3*randy, 0.3);
        primGen->AddGenerator(partGen);
      }
  */
```

```
  // Ion Generator
  //CbmIonGenerator *fIongen= new CbmIonGenerator(79, 197,79,1, 0.,0., 25, 0.,0.,-1.);
  //  primGen->AddGenerator(fIongen);


  //
  fRun->SetStoreTraj(kFALSE);

  PndMultiField *fField= new PndMultiField();

  PndTransMap *map= new PndTransMap("TransMap", "R");
  PndDipoleMap *map1= new PndDipoleMap("DipoleMap", "R");
  PndSolenoidMap *map2= new PndSolenoidMap("SolenoidMap", "R");
  fField->AddField(map);
  fField->AddField(map1);
  fField->AddField(map2);

/**
 // New field oct.2008
  PndTransMap  *tmap= new PndTransMap("Trans1", "R");
  PndDipoleMap *dmap1= new PndDipoleMap("Dipole1", "R");
  PndDipoleMap *dmap2= new PndDipoleMap("Dipole2", "R");

  PndSolenoidMap *smap1= new PndSolenoidMap("Solenoid1", "R");
  PndSolenoidMap *smap2= new PndSolenoidMap("Solenoid2", "R");
  PndSolenoidMap *smap3= new PndSolenoidMap("Solenoid3", "R");
  PndSolenoidMap *smap4= new PndSolenoidMap("Solenoid4", "R");

  fField->AddField(tmap);
  fField->AddField(dmap1);
  fField->AddField(dmap2);

  fField->AddField(smap1);
  fField->AddField(smap2);
  fField->AddField(smap3);
  fField->AddField(smap4);
*/




  //magnetic field
  /*  PndConstField *fMagField=new PndConstField();
      fMagField->SetField(0.,0.,20.); // values are in kG
      fMagField->SetFieldRegion(-50, 50,-50, 50, -100, 100);// values are in cm
(xmin,xmax,ymin,ymax,zmin,zmax)
      fField->AddField(fMagField);
  */
  fRun->SetField(fField);
```

```
  fRun->Init();
/*
  // -Trajectories Visualization
  // ----------------------------
  CbmTrajFilter* trajFilter = CbmTrajFilter::Instance();
  // Set cuts for storing the trajectpries
  trajFilter->SetStepSizeCut(0.01); // 1 cm
  //     trajFilter->SetVertexCut(-2000., -2000., 4., 2000., 2000., 100.);
  //     trajFilter->SetMomentumCutP(10e-3); // p_lab > 10 MeV
  //     trajFilter->SetEnergyCut(0., 1.02); // 0 < Etot < 1.04 GeV
  trajFilter->SetStorePrimaries(kTRUE);
  trajFilter->SetStoreSecondaries(kTRUE);
*/
  //
  //   // Fill the Parameter containers for this run
  //   //-----------------------------------------
  //
  CbmRuntimeDb *rtdb=fRun->GetRuntimeDb();
  Bool_t kParameterMerged=kTRUE;




  //if a field is used save the parameters in the RTDB
  /*
 PndSolenoidPar* Par1 = (PndSolenoidPar*) rtdb->getContainer("PndSolenoidPar");
 if ( map2 ) {  Par1->SetParameters(map2); }
 Par1->setChanged();
 Par1->setInputVersion(fRun->GetRunId(),1);

 PndDipolePar* Par2 = (PndDipolePar*) rtdb->getContainer("PndDipolePar");
 if (map1 ) {  Par2->SetParameters(map1); }
 Par2->setInputVersion(fRun->GetRunId(),1);
 Par2->setChanged();

 PndTransPar* Par3 = (PndTransPar*) rtdb->getContainer("PndTransPar");
 if (map ) {  Par3->SetParameters(map); }
 Par3->setInputVersion(fRun->GetRunId(),1);
 Par3->setChanged();
  */

  PndMultiFieldPar* Par = (PndMultiFieldPar*) rtdb->getContainer("PndMultiFieldPar");
  if (fField) {  Par->SetParameters(fField); }
  Par->setInputVersion(fRun->GetRunId(),1);
  Par->setChanged();

  CbmParRootFileIo* output=new CbmParRootFileIo(kParameterMerged);
  output->open("testparams.root");
  rtdb->setOutput(output);
```

```
  rtdb->saveOutput();
  rtdb->print();
  // Transport nEvents
  // ----------------1

  Int_t nEvents = numevents;
  fRun->Run(nEvents);

  timer.Stop();

  Double_t rtime = timer.RealTime();
  Double_t ctime = timer.CpuTime();
  printf("RealTime=%f seconds, CpuTime=%f seconds\n",rtime,ctime);

  cout << " Test passed" << endl;
  cout << " All ok " << endl;
  exit(0);

}
```

If it works for you, it might indeed well be something with the geometry file of the MVD, which is the only really different input parameter I have found until now.

Best regards,

Marius
