Subject: Re: Detector materials with radiation length not equal to zero
Posted by Raghav Kunnawalkam on Thu, 15 Mar 2012 21:34:08 GMT
View Forum Message <> Reply to Message

Hi Mohammad

I followed it the way it was described in the panda detector case and when i run my macro, i am generating the branch RadLen and its corresponding leaves, but there is nothing inside them. They are all blank.
I tried to see what i am doing different from the panda example and i just cant make out the problem.

```
rad_complete_stt(Int_t nEvents = 1, TString  SimEngine ="TGeant3", Float_t mom = 7.24)
{

//set the name of output file and etc...

//load libraries


Double_t BeamMomentum   =15.0;

    // -----   Create simulation run   ----------------------------------------
    FairRunSim* run = new FairRunSim();
    run->SetName("TGeant3");       // Transport engine
   // fRun->SetBeamMom(BeamMomentum);

// here it does not let me set the beam momentum even though it is defined in the class FairRunSim. Maybe thats the issue here. i am running a beam whit no momentum.

    run->SetOutputFile(outFile);          // Output file
    FairRuntimeDb* rtdb = run->GetRuntimeDb();
    //   ---------------------------------------------------------------------


    // ----- Create media -------------------------------------------------

    run->SetMaterials("media.geo");//materials
      //---------------------------------------------------------------------- -

    // ----- Tell the detector to calculate the radiation length ----------

    run->SetRadLenRegister(kTRUE);

    //   ---------------------------------------------------------------------

    // -----   Runtime database   ---------------------------------------------


    Bool_t kParameterMerged = kTRUE;
    FairParRootFileIo* parOut = new FairParRootFileIo(kParameterMerged);
```

```
    //  FairParAsciiFileIo* parOut = new FairParAsciiFileIo();
    parOut->open(parFile.Data());
    rtdb->setOutput(parOut);
    rtdb->saveOutput();
    rtdb->print();


//create geometry, modules and detectors.


 // -----   Create PrimaryGenerator   --------------------------------------
    FairPrimaryGenerator* primGen = new FairPrimaryGenerator();
    run->SetGenerator(primGen);

    FairBoxGenerator* boxGen = new FairBoxGenerator(0, 10); // 13 = muon; 1 = multipl.
// the 0 here represents geantino.
    boxGen->SetPtRange(mom,mom); // GeV/c
    boxGen->SetPhiRange(0., 360.); // Azimuth angle range [degree]
    boxGen->SetThetaRange(0., 0.); // Polar angle in lab system range [degree]
    boxGen->SetXYZ(0., 0., 0.); // mm o cm ??
    primGen->AddGenerator(boxGen);

    // -----   Run initialisation   -------------------------------------------

    run->Init();

    // -----   Start run   ----------------------------------------------------

    run->Run(nEvents);
    //   ------------------------------------------------------------------------
    run->CreateGeometryFile("data/geofile_rad_length.root");

    // -----   Finish   -------------------------------------------------------
    timer.Stop();
    Double_t rtime = timer.RealTime();
    Double_t ctime = timer.CpuTime();
    cout << endl << endl;
    cout << "Macro finished succesfully." << endl;
    cout << "Output file is "    << outFile << endl;
    cout << "Parameter file is " << parFile << endl;
    cout << "Real time " << rtime << " s, CPU time " << ctime
    << "s" << endl << endl;
    //   ------------------------------------------------------------------------ end of file
}
```

The panda example is pretty much the same except geometry definitions, but there is something inside its RadLen folder. I just dont understand the mistake i am making here.
I have attached a picture of what my root file looks like and on the canvas you can see that there is nothing inside the leaf fradlen.

thanks a lot for all you help

Cheers
Raghav

File Attachments

1) Screen Shot 2012-03-15 at 5.32.44 PM.png, downloaded 916 times

