

Zastosowanie pakietu Geant4 w fizyce jądrowej

Aleksandra Fijałkowska

Wydział Fizyki, Uniwersytet Warszawski

aleksandra.fijalkowska@fuw.edu.pl

9 stycznia 2019

Geant4 umożliwia importowanie elementów geometrii z rysunków technicznych zapisanych w plikach STL.

Jeden plik może zawierać jedną bryłę, która po zaimportowaniu będzie służyła za obiekt typu **G4Solid**, czyli wyłączenie kształt, bez informacji o jej innych cechach typu materiał, kolor, lokalizacja itp.

Najprostsza metoda importu plików do projektu wykorzystującego Geant4 jest skorzystanie z biblioteki CADMesh (<https://github.com/christopherpoole/CADMesh>).

<https://github.com/christopherpoole/CADMesh> Szukaj

Why GitHub? Business Explore Marketplace Pricing

Search Sign in Sign up

christopherpoole / CADMesh

Watch 12 Star 37 Fork 32

Code Issues 11 Pull requests 0 Projects 0 Insights

Join GitHub today

GitHub is home to over 28 million developers working together to host and review code, manage projects, and build software together.

Dismiss

Sign up

A CAD file interface for GEANT4

cadmesh geant4 assimp tetgen monte-carlo-simulation radiation

172 commits 3 branches 8 releases 1 contributor MIT

Branch: master New pull request Find file Clone or download

christopherpoole Checking that the closing of a token/block is being done with a token... Latest commit 635e857 on 8 Nov 2017

cmake/Modules	Added a cmake module to find tetgen (from here: https://raw.githubusercontent.com/christopherpoole/CADMesh/master/cmake/Modules/tetgen.cmake)	a year ago
examples	No need to cast the filename from a string to a char array.	a year ago
external	Updated assimp to version 4+.	a year ago
include	Specifying entires for the Unknown file type enum in the extension, st...	a year ago
src	Checking that the closing of a token/block is being done with a token...	a year ago
tests	Using the new close surface method in the unit tests.	a year ago
.gitignore	Ignoring vim *.swp files.	4 years ago

Biblioteka CADmesh oferuje wygodną instalację z wykorzystaniem CMake. Przytaczając rozdział Installation w README:

```
# CADMesh latest release
git clone https://github.com/christopherpoole/CADMesh.git
git checkout v1.1

# OR download a release from
https://github.com/christopherpoole/CADMesh/releases,
and extract the download somewhere.

cd cadmesh/

# Fetch the external dependancies (optional), running
# cmake will result in this happening anyway.
# git submodule update --init --recursive

mkdir build
cd build

cmake .. -DCMAKE_INSTALL_PREFIX:PATH=<optional install location>
make
sudo make install
```

Od czasu CMake preferuję instalację oprogramowania w katalogach lokalnych, nie systemowych, ale wybór należy do Państwa.

Aby wprowadzić biblioteki CADMesh do kompilacji należy postąpić analogicznie do dołączenia jakichkolwiek innych bibliotek (np. ROOT). CMakeList należy uzupełnić o linie:

```
...  
find_package(cadmesh)  
include_directories(${CADMESH_INCLUDE_DIRS})  
  
...  
target_link_libraries(ProjectName ${cadmesh_LIBRARIES})
```

Jeśli CADMesh jest zainstalowany w „niestandardowej” lokalizacji uprzednio należy ją dodać do zmiennej LD_LIBRARY_PATH. Można to zrobić przez ręcznie, wykonując uprzednio napisany skrypt, lub w pliku .bashrc

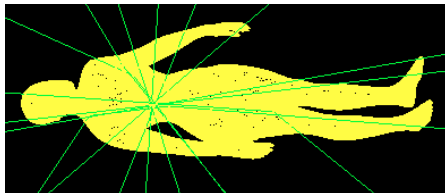
```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/path_to_CADMesh/build/lib
```

```
#include "CADMesh.hh"
....
CADMesh* fantomMesh = new CADMesh("filemane.stl", //nazwa pliku
                                   1*cm, //units
                                   G4ThreeVector(0*cm, 0*cm, 0*cm), //position
                                   false); //reverse

//robi obiekt G4VSolid z ,,siatki''
G4VSolid* fantomSolid = fantomMesh->TessellatedMesh();
```

Posiadając obiekt typu G4VSolid można z nim postępować tak jak z każdym innym kształtem.

```
void DetectorConstruction::ConstructHumanFantom()  
{  
    G4NistManager* man = G4NistManager::Instance();  
    G4Material* water = man->FindOrBuildElement("G4_WATER");  
    CADMesh* fantomMesh = new CADMesh("../Female_Human_Body_lowRes.stl",  
                                       1*cm,  
                                       G4ThreeVector(0*cm, 0*cm, 0*cm),  
                                       false);  
    G4VSolid* fantomSolid = fantomMesh->TessellatedMesh();  
    G4LogicalVolume* fantomLogVol = new G4LogicalVolume(fantomSolid,  
                                                         water,  
                                                         "fantomLogVol");  
    G4VisAttributes* fantomVisAtt = new G4VisAttributes(G4Colour::Yellow());  
    fantomLogVol->SetVisAttributes(fantomVisAtt);  
    G4ThreeVector pos(0,0,0);  
    new G4PVPlacement(0, pos, fantomLogVol, "fantom", worldLogic, 0, 0);  
}
```



CADMesh jest projektem „żywym”, stale rozwijanym przez autora biblioteki. Podzieliłam się z Państwem doświadczeniami, które nabyłam ok 2 lat temu. Jeśli napotkacie Państwo jakieś problemy proszę nie wahać się ich zgłaszać.

- ▶ Kompilacja, zachęcam do zwracania uwagi nie tylko na błędy, ale również na ostrzeżenia kompilatora
- ▶ Działanie, program powinien się uruchamiać i działać „zgodnie z planem”
- ▶ Krótki dokument opisujący ideę projektu – co było celem projektu (np. wyznaczenie wydajności, funkcji odpowiedzi itp.), jak wygląda układ detekcyjny, co jest częstką pierwotną i jak jest ona generowana, co jest wynikiem symulacji
- ▶ Wynik działania kodu - wykresy lub liczby
- ▶ Ewentualne skrypty, wykorzystane do wyznaczenia pożądanej wielkości