Interface for Rapid Geometry Modelling and Exchange Between CAD Tools and Simulation Platforms Dedicated for HEP Experiments

Rimantas Naina[1], Jakub Hajduga[2], Tomasz Szumlak[2]

[1]Vilnius University

[2]AGH University of Krakow

GEANT4 TOOLKIT

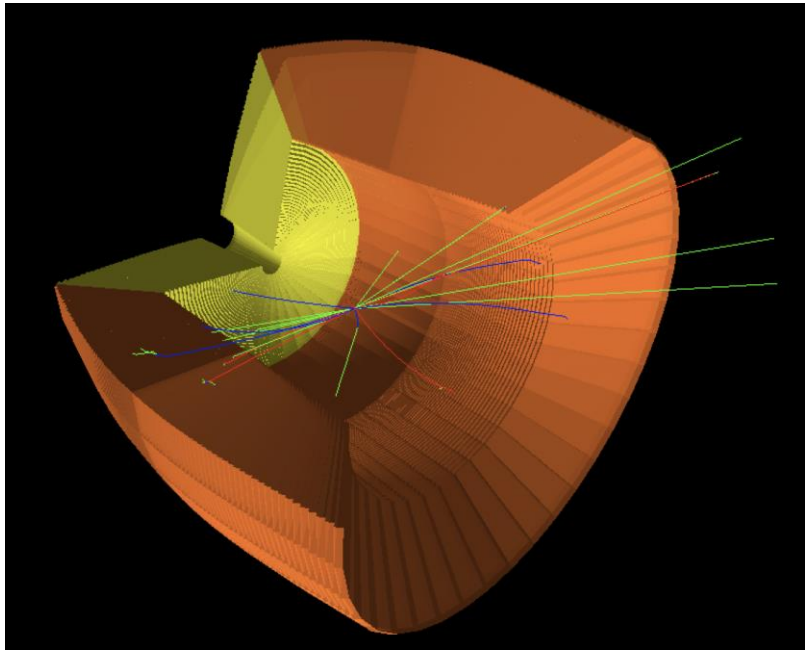
GEANT4 (GEometry ANd Tracking) is a toolkit designed for simulating the passage of particles through matter

Applications

- Simulate any detector setup
- Record physical quantities resulting from particle interactions with matter

Physics processes

- Electromagnetic, strong and weak interactions
- Wide energy range



WHERE IS GEANT4 USED?

High energy physics

- Simulating new detector setups
- Generating detector data to prepare for data analysis
- Used in data analysis to help make sense of the gathered data

Space science

- Evaluating the effects of space radiation on satellites

Medical physics

SIMPLE GEANT4 PROGRAM

GEANT4 is a toolkit which allows users to write their own simulation program.

Add a **physics list**
- Define what physics processes will be simulated.
- For example: include electromagnetic and weak interactions

Add **particle generator**
- Number of particles
- Type of particle

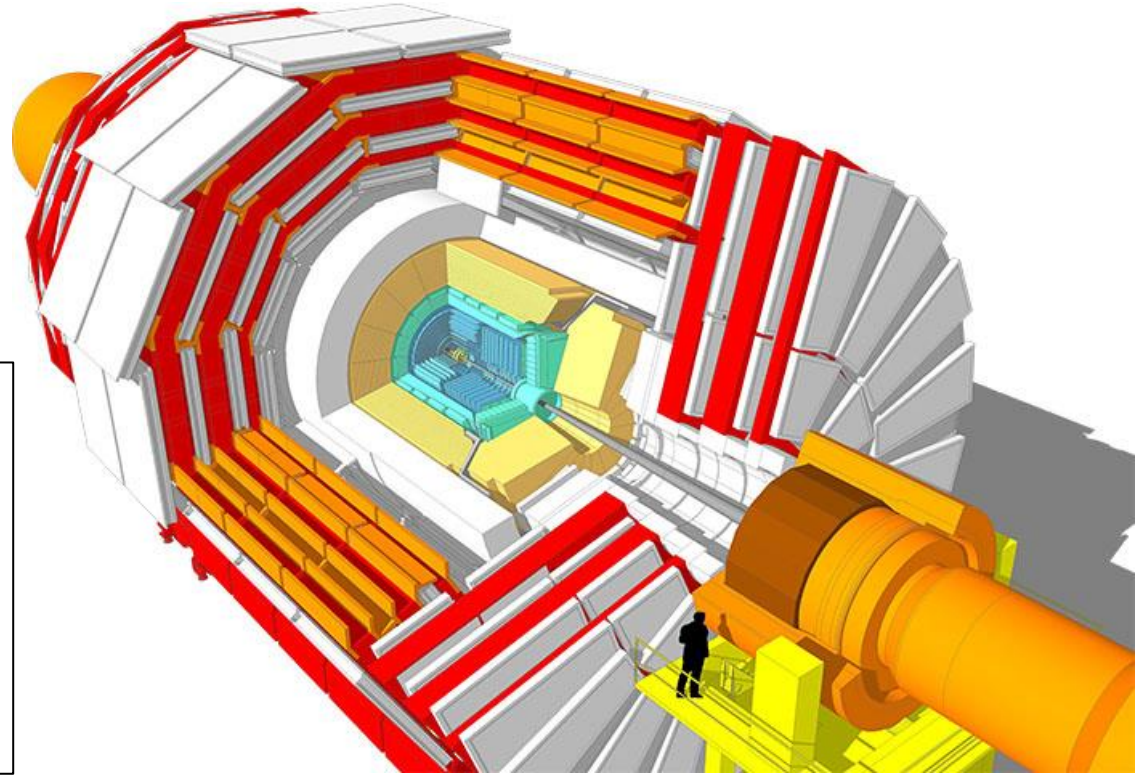
Add **detector geometry**
- Define the shape of the detector
- Material
- Position
- Rotation

DIFFICULTY WITH COMPLEX GEOMETRIES

GEANT4 allows the user to define any detector geometry in their simulation.

For large and complex detectors this can be very challenging.

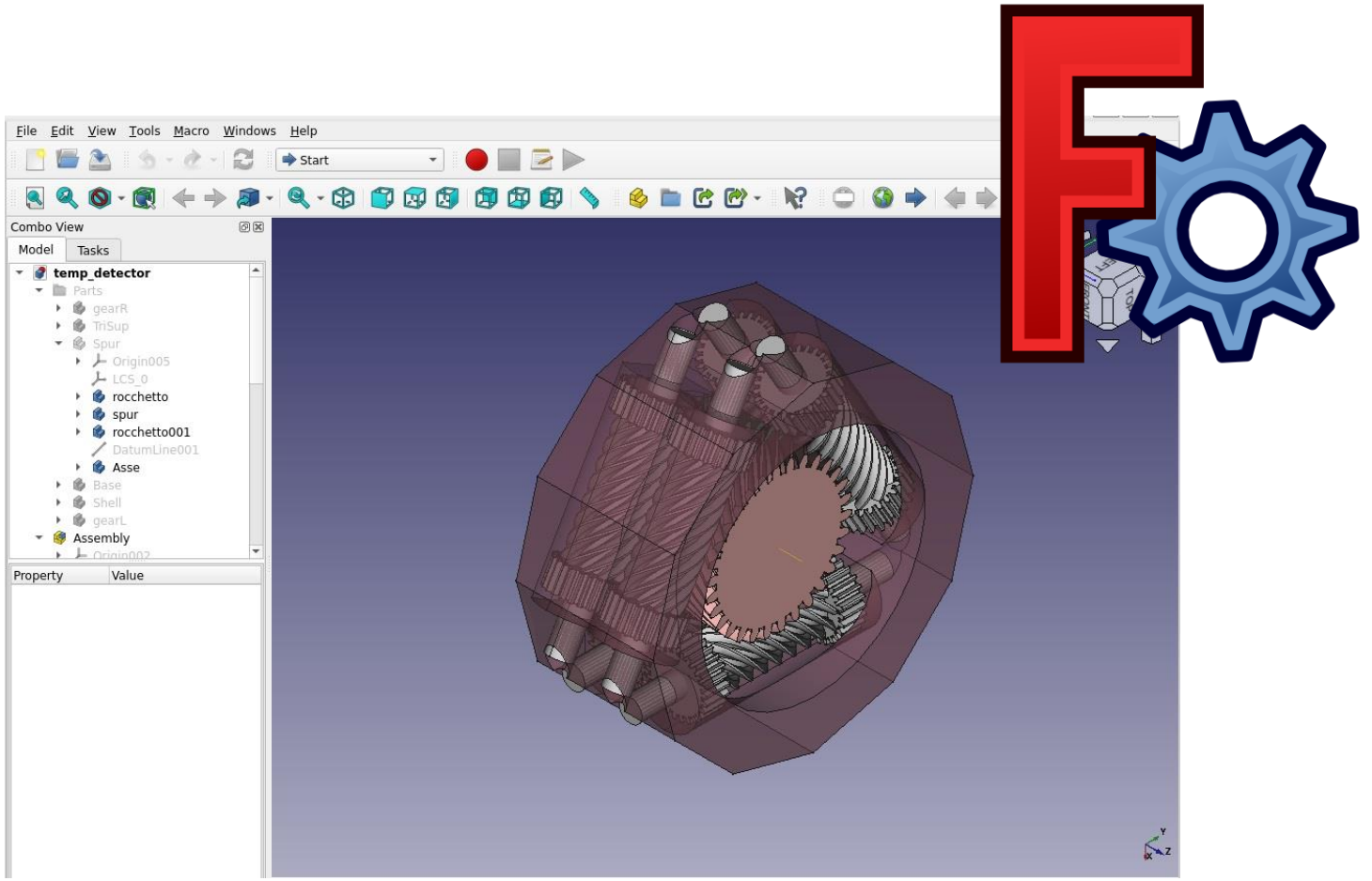
```
G4double innerRadius = 0.*cm;
G4double outerRadius = 60.*cm;
G4double hz = 25.*cm;
G4double startAngle = 0.*deg;
G4double spanningAngle = 360.*deg;

G4Tubs* trackerTube
  = new G4Tubs("Tracker",
               innerRadius,
               outerRadius,
               hz,
               startAngle,
               spanningAngle);
```



https://geant4-userdoc.web.cern.ch/UsersGuides/ForApplicationDeveloper/html/GettingStarted/geometryDef.html

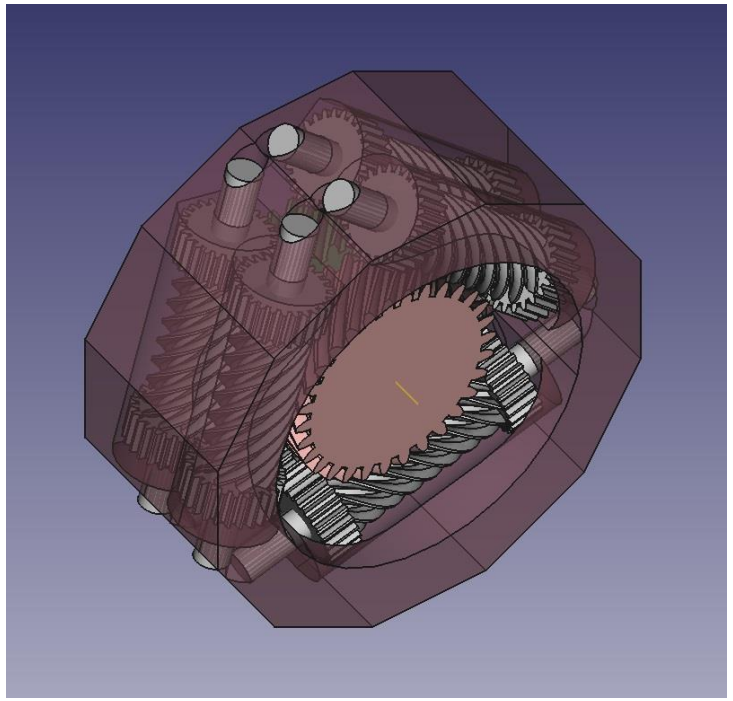
FREECAD

- Free software for creating 3D models.
- Written in C++ and python.
- Our goal is to make importing FreeCAD models into GEANT4 simulations easy.

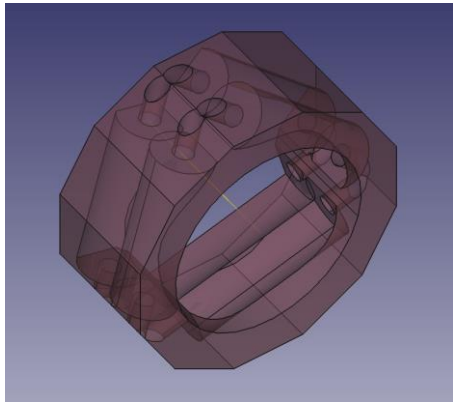


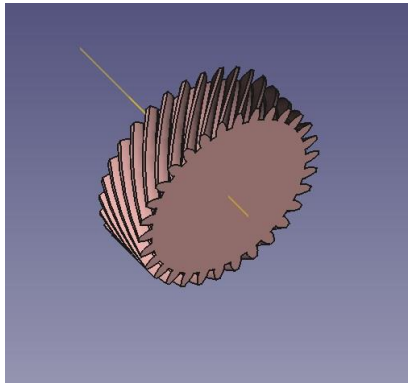
STL EXPORT

.FCStd



.STL





• • •

FREECAD OBJECT HIERARCHY

Detector

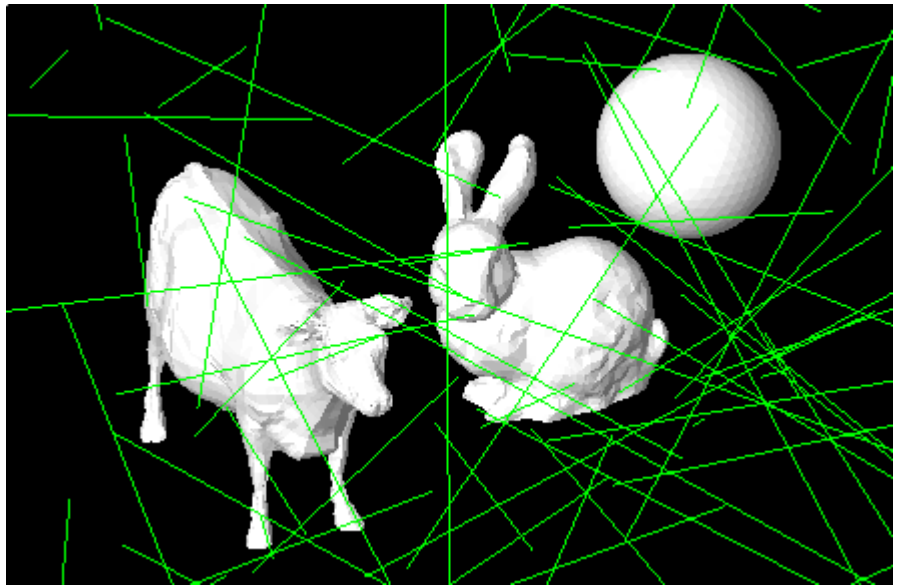
```
|
|-Part1
| |-Part11
| | |-Cylinder1
| | |-Cube
| | |-Part111
| | | |-Cylinder2
| | |-Part112
| |   |-Part1121
| |     |-Part11211
| |       |-Sphere
| |-Part12
|   |
|  ...
|
|-Part2
  |
...
```

- We need to get the relevant objects from the FreeCAD model and export them as STL files.
- Each shape in the FreeCAD model can have many layers
- Before exporting we need to get the correct position of the subshapes and we need to merge them into one shape

CADMESH

Once we have the detector geometry exported to multiple STL files we use CADMesh.

It allows us to import STL files into our GEANT4 simulations



```
auto mesh = CADMesh::TessellatedMesh::FromSTL(path);
G4VSolid* solid = mesh->GetSolid();
```

https://github.com/christopherpoole/CADMesh

TOML CONFIGURATIONS

Configurations about each part of the detector are saved in a TOML file.

- We can change the position, rotation and material of each part
- We can exclude a part from the simulation if we want
- Settings are automatically generated during STL export

```
[Parts]
#[Name, Position, Rotation, Material, STL_File]
elements = [
["Body", [200.0, 0.0, 0.0], [45, 0.0, 0.0], "G4_WATER", "/home/geant4/impress-u-agh/geometryWrapper/utils/../output/
["Body001", [0.0, 0.0, 0.0], [0.0, 0.0, 0.0], "G4_WATER", "/home/geant4/impress-u-agh/geometryWrapper/utils/../outpu
["Body002", [0.0, 0.0, 0.0], [0.0, 0.0, 0.0], "G4_WATER", "/home/geant4/impress-u-agh/geometryWrapper/utils/../outpu
["Body003", [0.0, 0.0, 0.0], [0.0, 0.0, 0.0], "G4_WATER", "/home/geant4/impress-u-agh/geometryWrapper/utils/../outpu
]
```

WHAT WE HAVE SO FAR

- We can easily export FreeCAD models to STL format
- We can import a single STL file into our simulation
- We can import multiple STL files into our simulation
- Position of each detector part can be adjusted in the TOML configuration file

WHATS NEXT

Interface for controlling detector geometry

- Simplify the process of exporting CAD models to STL format.
- Adjusting the position and material of the detector parts using the interface.

SUMMARY

- With this project we aim to simplify the detector design process for GEANT4 simulations
- We aim to create tools which would allow to easily import and control detector parts designed using CAD tools.

https://github.com/IMPRESS-U-AGH

rimantas.naina@ff.stud.vu.lt