
Implementation of Custom Rotation Order in remoll GDML

Prakash Gautam

⟨ प्रकाश गौतम | پرکاش گوتم ⟩

University of Virginia

MOLLER Simulation Meeting

2024-02-20

Rotation Order: Current Implementation

- ▶ When geometry is imported with geometry we can specify the rotation of physical volume with rotation tag

```
<rotation unit="degree" x="45" y="25" z="90"/>
```

this always does the rotation $R_{x(45)} \rightarrow R_{y(25)} \rightarrow R_{z(90)}$.

- ▶ As far as I can tell there is not other way to specify custom rotation order in standard GDML parser of Geant4.
- ▶ The rotation order is hardcoded

```
// G4GDMLReadParamvol.cc#L1207 geant4.10.07.p04  
parameter.pRot = new G4RotationMatrix();
```

```
parameter.pRot->rotateX(rotation.x());  
parameter.pRot->rotateY(rotation.y());  
parameter.pRot->rotateZ(rotation.z());
```

Custom Rotation Order

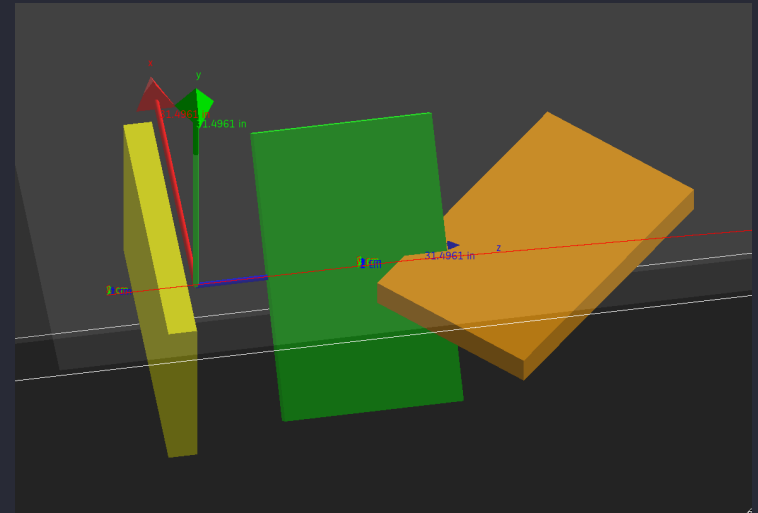
- ▶ I created a new tag that can be used in remoll GDML files `<rotation>` for ordered rotation.

```
<rotation unit="degree" x="45" y="25" z="90" order="zxy"/>
```

this would allow the rotation $R_{z(90)} \rightarrow R_{x(45)} \rightarrow R_{y(25)}$. The value of order parameter can be any of the six permutations of string “xyz”.

Example:

- ▶ Yellow: Unrotated
- ▶ Green: $R_{x(90)} \rightarrow R_{y(45)} \rightarrow R_{z(0)}$
- ▶ Orange: $R_{y(45)} \rightarrow R_{x(90)} \rightarrow R_{z(0)}$
- ▶ *there is a translation between them*



Remoll GDML Parser

- ▶ Remoll only parses the auxiliary tags in GDML by itself and the rest is done by standard G4GDMLParser.

```
<auxiliary auxtype="SensDet" auxvalue="coilDet"/>  
<auxiliary auxtype="DetNo" auxvalue="3001"/>  
<auxiliary auxtype="Color" auxvalue="magenta"/>
```

- ▶ In remollDetectorConstruction.hh there is an object of G4GDMLParser.

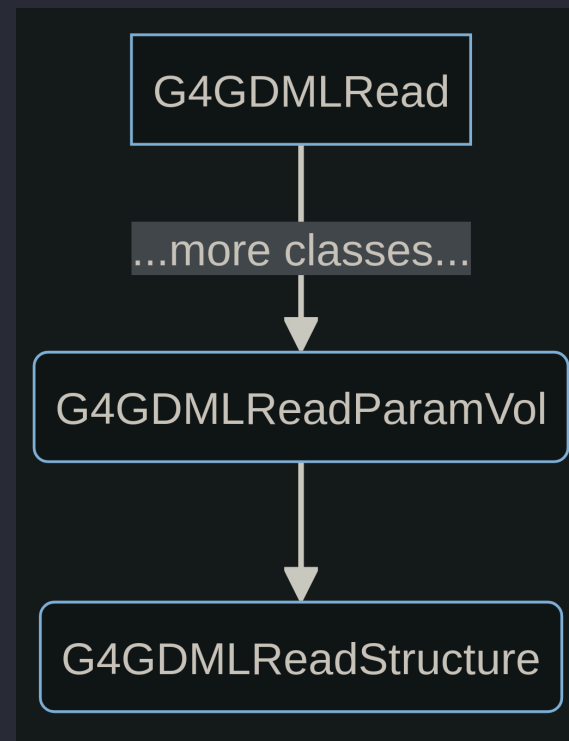
```
G4GDMLParser fGDMLParser; //remollDetectorConstruction.hh  
fGDMLParser.Read(fGDMLFile,fGDMLValidate); //remollDetectorConstruction.cc  
...  
...
```

- ▶ The constructor of G4GDMLparser takes an optional argument of G4GDMLReadStructure.
The object

```
G4GDMLParser();  
G4GDMLParser(G4GDMLReadStructure*);
```

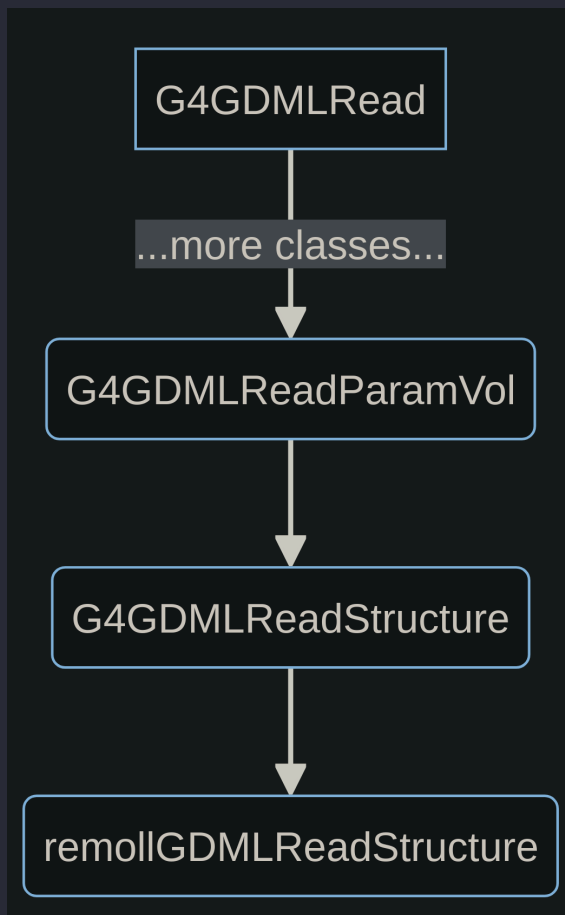
The hierarchy of class G4GDMLReadStructure

- ▶ G4GDMLReadStructure has two three virtual functions that can be overridden.
- ▶ G4GDMLReadParamvol reads the parameters of <phyvol> tag.
- ▶ G4GDMLReadStructure parses <structure> tags etc.



Creating a new tag in

- ▶ A new class `remollGDMLReadStructure` is derived from `G4GDMLReadStructure`
- ▶ It overrides some methods inside `G4GDMLReadParamvol` and some from `G4GDMLReadStructure`.
- ▶ In the standard Geant4 implementation the rotation angles are parsed as vector of three components so the order of rotation is not possible.
- ▶ The new implementation for `<rotation>` parses the angles and order parameter to create a rotation matrix, thus preserving order information for further rotation.



Summary

- ▶ Current GDMLParser from geant4 doesn't allow custom order rotation of `<phyvol>` and always does the order $R_x \rightarrow R_y \rightarrow R_z$.
- ▶ A new tag is now available `<orotation>` that behaves in the same way as the `<rotation>` tag except it also takes an optional argument `order` which can have values that are any permutation of string "xyz". (Defaults to "xyz").

```
<orotation unit="degree" x="45" y="25" z="90" order="zxy"/>
```

- ▶ Every instance of `<rotation>` can in principle be replaced by `<orotation>` without any impact.
- ▶ It has been tested by Xiang to change rotation order of Quartz tiles.
- ▶ More tests needed to make sure it doesn't affect any other functionality.
- ▶ A new header/source file pair will be added in remoll which can be compiled with the standard procedure without any further dependency.