

Mathew Muether

Aug 9, 2024

TMS Geometry Update Discussion



TMS Geometry Revision

- The core of the existing TMS Geometry was built many years ago.
- It was designed to be simple and configurable.
 - Built from 5 primary objects replicated and placed in mother volumes of air.
- The base design that we have heading into the PDR is different from the current geometry and we should update to reflect the new design.

Existing Geometry Review

- <https://github.com/DUNE/dunendggd>
- The current TMS geometry is defined in 2 files
 - TMS.cfg – User config file

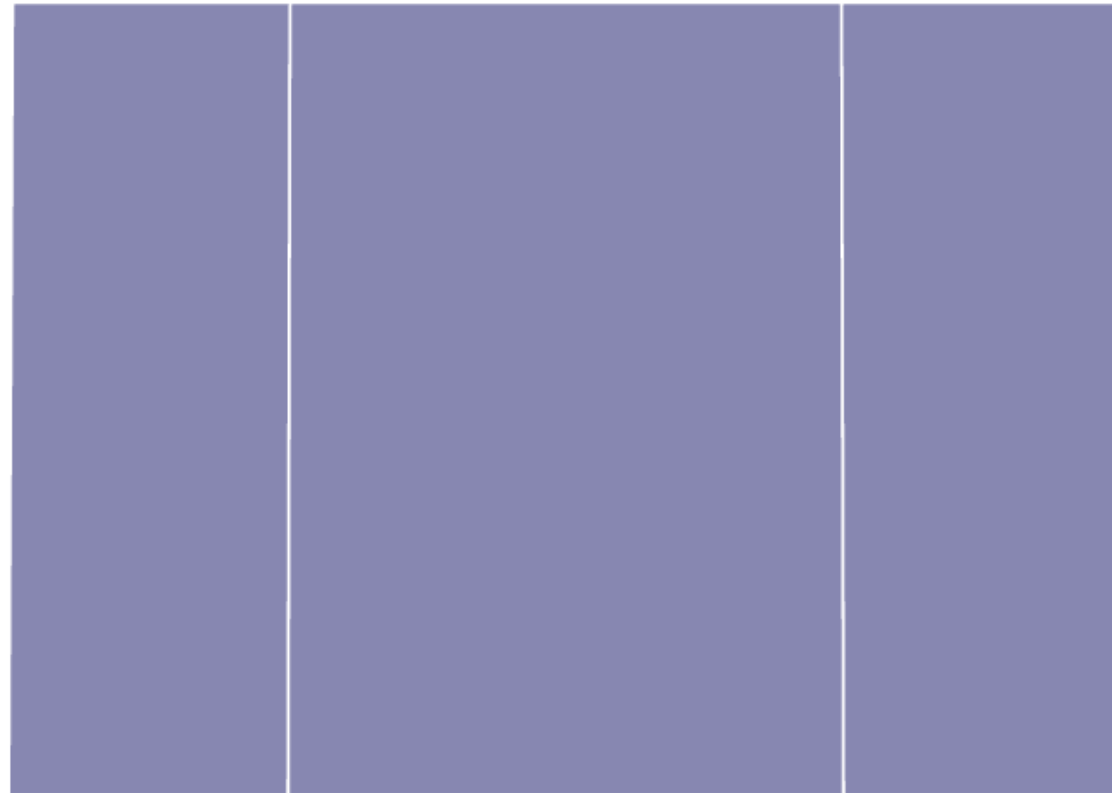
```
1 [TMS]
2 class = duneggd.SubDetector.TMS.tmsBuilder
3 mat = "SteelTMS"
4 thinbox1Dimension = [Q('1.749m'),Q('5.022m'),Q('0.015m')]
5 thinbox2Dimension = [Q('3.498m'),Q('5.022m'),Q('0.015m')]
6 gapPosition = [Q('0.020m'),Q('0m'),Q('0m')]
7 BFieldUpLow = '(0 T, -1 T, 0 T)'#(0 T,3 T,0 T)
8 BFieldUpHigh = '(0.0 T, -1 T, 0 T)'#(0.0 T, 5.5 T, 0 T)
9 BFieldDownLow = '(0 T, 1 T, 0 T)'#(0 T, -3 T, 0 T)
10 BFieldDownHigh = '(0.0 T, 1 T, 0 T)'#(0.0 T, -5.5 T, 0 T)
```

- TMS.py – Source code

The steel structure

- Thinbox1 and ThinBox2 are the thin side and central steel pieces.
- These box all each have a configurable field assigned to them.
- Thickbox1 and Thickbox2 are the thick versions.
- Two box1s are place around box 2 to form thin and thick “layers” (held in a box of air)

```
4 thinbox1Dimension = [Q('1.749m'),Q('5.022m'),Q('0.015m')]
5 thinbox2Dimension = [Q('3.498m'),Q('5.022m'),Q('0.015m')]
6 gapPosition = [Q('0.020m'),Q('0m'),Q('0m')]
7 BFieldUpLow = '(0 T, -1 T, 0 T)'#(0 T,3 T,0 T)
8 BFieldUpHigh = '(0.0 T, -1 T, 0 T)'#(0.0 T, 5.5 T, 0 T)
9 BFieldDownLow = '(0 T, 1 T, 0 T)'#(0 T, -3 T, 0 T)
10 BFieldDownHigh = '(0.0 T, 1 T, 0 T)'#(0.0 T, -5.5 T, 0 T)
```



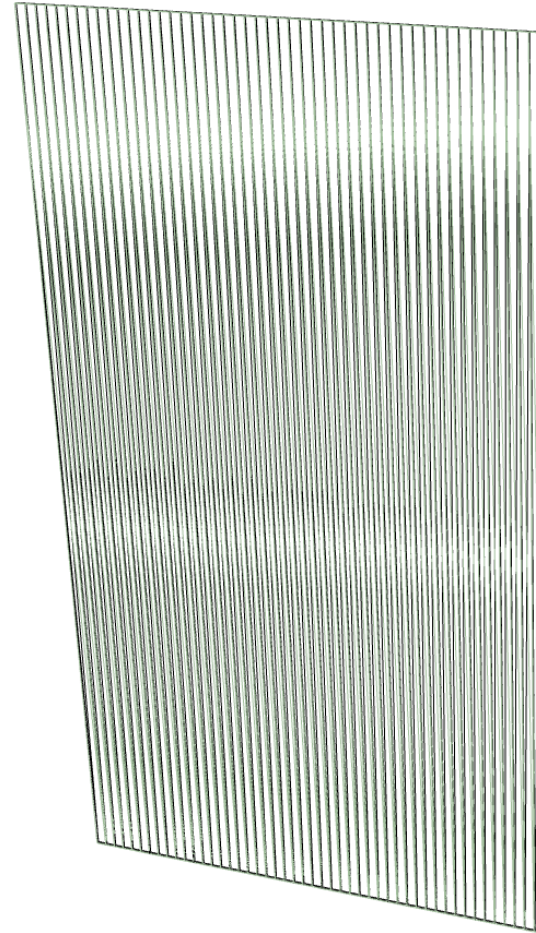
The Module

```
scinBox = geom.shapes.Box( 'scinbox'+self.name,  
    dx = 0.5*Q("0.03542m"),  
    dy = 0.5*Q("3.096m"),  
    dz = 0.5*Q("0.01m"))
```

```
scinBox_lv = geom.structure.Volume( 'scinBoxlv'+self.name, material='Scintillator', shape=scinBox)
```

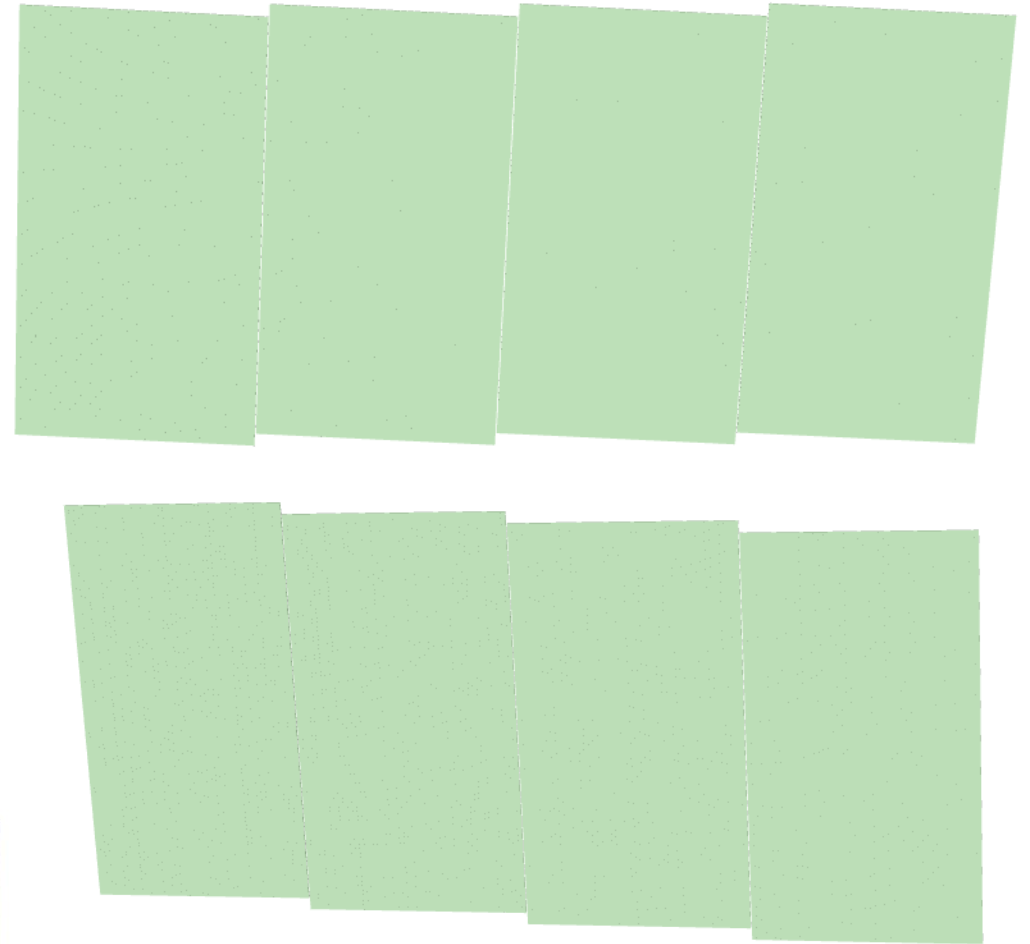
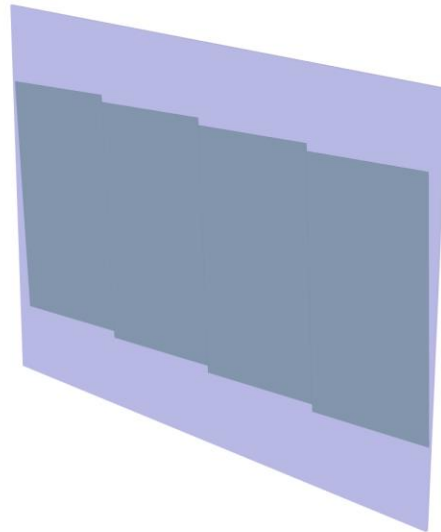
- We define a module which is made from clones of a box of scintillator (a bar).
- 48 bars are positioned in an air volume to form a module.

```
ModuleBox = geom.shapes.Box( 'ModuleBox',  
    dx = 0.5*Q("0.03542m")*48, # 0.04*42  
    dy = 0.5*Q("3.096m"),  
    dz = 0.5*Q("0.01m"))
```



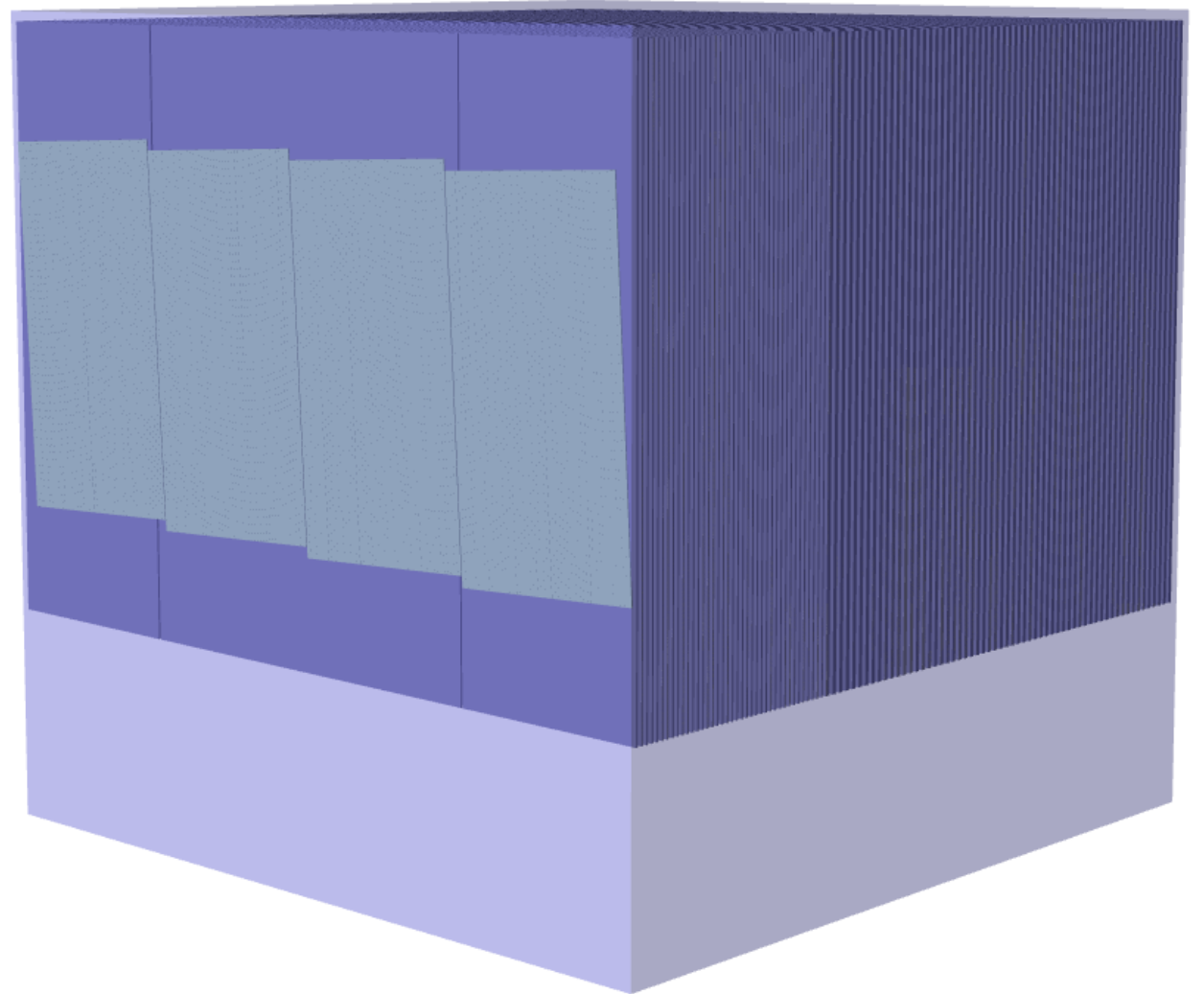
The Module Layer

- 4 modules are placed in a larger air volume side by side to form a Module layer.
- At the moment there are 2 types: left and right rotated (hard coded 3 degree rotations).



Building TMS

- Steel layers and module layers are then placed U(Thin)V(Thin)...U(Thick)... into a mother Air volume to form the current TMS model.
- This is then placed into full hall geometry for simulations.



Recommendations

- Update existing steel and scintillator dimension to match current PDR design.
- Divide source code into multiple parts in a TMS specific folder in the geometry package.
 - Tms.py – primary builder to assemble all the bits and bobs.
 - TMSCassette(U,V,Y).py (replaces the modules that exist)
 - We can have a few like a TMSCassetteSimple.py and TMSCassetteEng.py.
 - Can have other sub builders are needed.
 - TMSSteelEng.py, TMSSteelSimple – Steel structure builders with appropriate sub builders, one a full engineering match and the other a simple configurable version for optimization studies.
 - TMSSupport.py – Support and infrastructure builder.
- Remove hard coded values from Simple Models and expand config options to include modification of rotations, use of field maps,