MESSY Geometry Issues

Joanne Bogart   3 Oct. 2001

# Components of a Tower

For purposes of geometry description, a tower is an envelope volume with transverse dimensions = tower pitch. Height is CAL stay-clear ht. + TKR stay-clear ht. + TKR-CAL gap.

Volumes included in a tower are one CAL, one TKR, and that part of the grid (flange and web) contained in the envelope.
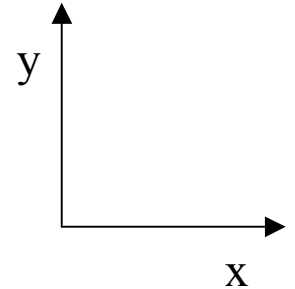
But we're only allowed to use simple volumes, so...

Option A: the per-tower web and per-tower flange are each described as 4 separate pieces, and another 4 pieces of each are needed to go around the 16 towers.

Z-slice of Towers (CAL + web)

CAL

web

y

x
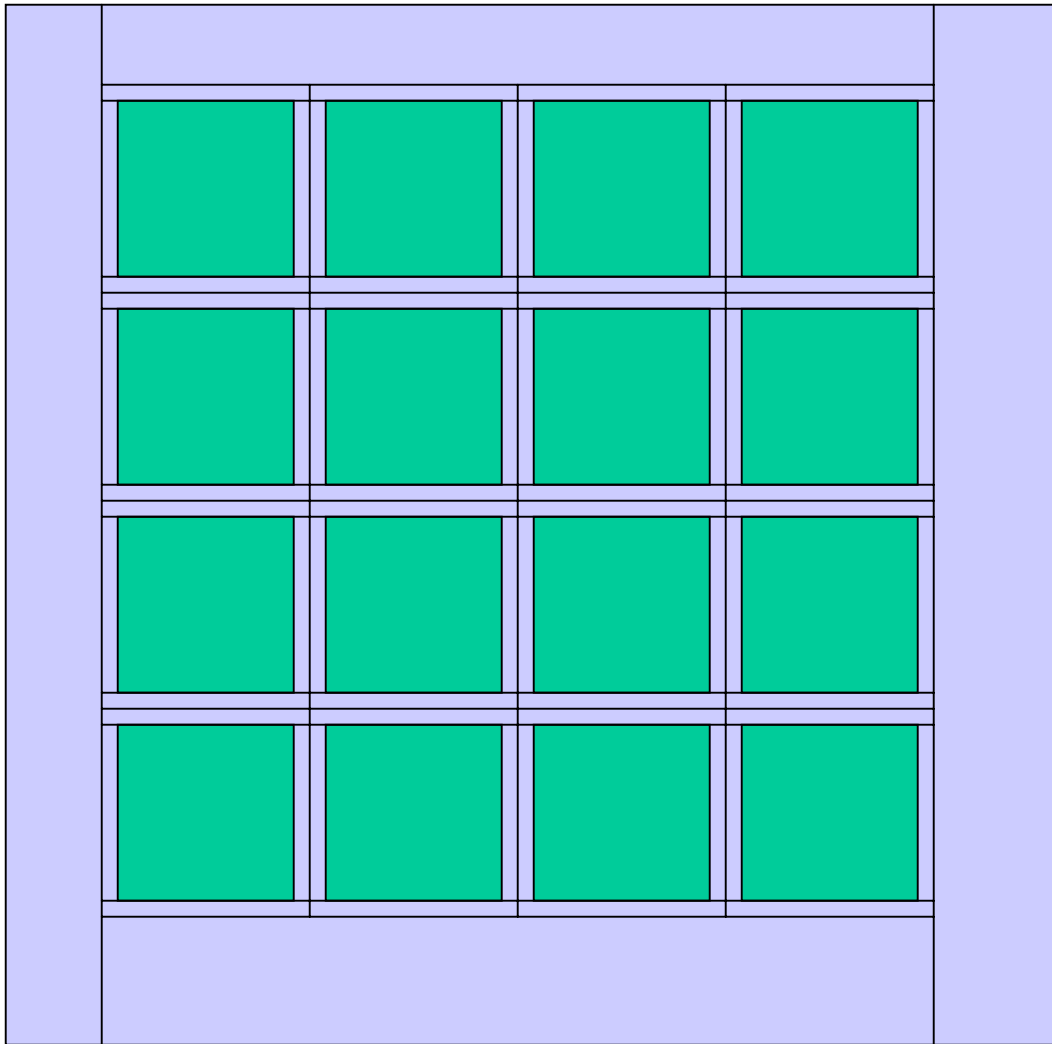
not to scale!

A Z-slice through the flange would look similar, but cut-outs are smaller and contain no material.

Alternatives?

Option B: Nest the towers inside a solid volume of grid material. Would also have to nest some boxes of vacuum, just about as awkward to define as grid pieces and even less natural.

Option C: Make use of boolean volumes as G4 defines them. Can then form unions, intersections, subtractions of simple volumes as long as they're made of same material. In particular, can readily describe a solid block with cut-outs as one volume.

It would be straightforward to add booleans to the set of volumes allowed by the XML description and understood by detModel, but what about Gismo?

Identifiers & Artificial Pieces

If we go with Option A, a tower can have 10 possible child volumes:

- A tracker

- A calorimeter

- Any of 4 grid web pieces

- Any of 4 grid flange pieces

Identifiers & Artificial Pieces

Since each of these is declared as a separate volume to the simulator, each must* have a distinct identifier, in particular the field which encodes "tower component" should have a distinct value.

But no other application is likely to care about this level of detail for non-sensitive material. Hit identifiers could treat the 4 web pieces per tower as one, or maybe even combine all web pieces from all towers.

The existing code for identifier conversion won't do this, but could made to without much additional work.

*or does it? I might not be understanding how simulators work.

More Criteria

Efficiency: Does one volume definition strategy (pieces, nesting, booleans) lead to significantly faster simulations? In particular, if booleans are a winner for G4, should we consider implementing them for Gismo?

Maintainability: Is there a significant difference in the complexity of description, depending on which strategy is used?