Fake Tile Shingling May 2007

Initial proposal at C&A Meeting: http://www.slac.stanford.edu/exp/glast/ground/LATSoft/vol1/geometry/notes/GeoApril2007.pdf See Slide 7

When Joanne modeled the tiles as depicted without changing anything else Joanne realized there would be a gap along the edge where the sides come together for rows 1 and 3 so she modified the design by making the width of these rows greater. You can see some more pictures at http://www.slac.stanford.edu/exp/glast/ground/LATSoft/geometry/pictures/ACD/newSides/

From Alex, for our information, in reality the corner gaps between the adjacent sides tiles are 4mm in nominal.

Checking Changes

Ran detCheck's dumpIds application to compare the output using xmlGeoDbs v1r39p3 against v1r40 (contains changes for side tile shingling). The initial output provides a count of materials, constants and detectors - the only difference between that portion of the output is in the number of constants. It is now 771 rather than 753.
The rest of the output is volumeIdentifiers - performing a diff on the output shows that the only changes are to the side tiles and demonstrates that the build order of the side tiles has changed from bottom to top, now to top to bottom. This is confirmed by Joanne. A visual scan of the dumped Ids suggests the volIds are as they should be and that the placement and dimensions of the side tiles has changed in a sensible way. Here is the old dumpId output and here is the new dumpId output.

Next up: Modifying side ribbons to follow contours of new side tiles. Here's a picture (see slide 3). There are additional pretty pictures in that link as well, including the reworking of the top ribbons.

Updates to the top and side Ribbons

The Side ribbons are now segmented to follow contours of the fake tile shingling.

The volIds for the ribbons were updated to provide for the new segments in the x top and side ribbons. The ribbon id's are formed with 6 fields (in xmlGeoDbs v1r39p3):

1. [0] field = fLATObjects, value = eLATACD(=1)
2. [1] field = fACDFace, value in the range [0,4]. The negative X face is 1, then continue counterclockwise in numbering the remaining side faces.
3. [2] field = fACDCmp, value = eACDRibbon (= 41)
4. [3] field = fMeasure, value = eMeasureX (resp. eMeasureY) for ribbons whose short dimension is parallel to x-axis (resp. y-axis).
5. [4] field = fRibbon, values in the range [0, 3].
6. [5] field = fSegment. Value is always 0 for side faces. For x-measuring top ribbons (whose continuation runs along +Y and -Y faces) range is in [1, 5] for normal segments under top tiles. Value = 6 for extra little pieces of ribbon behind vertical part of top bent tiles. y-measuring top ribbons are modeled with a single volume, fSegment = 1.

So, in particular, the side ribbons are /1/<face (1-4)>/41/<meas-direction>/<ribbon # (0-3>/0

For the new segments along the sides, the fSegment takes on a value of 1-7 rather than 0

Update for the top X ribbons
(the so-called "X ribbons" since they measure X = run perp. to X axis) more realistic by adjusting the lengths of the under-tile pieces and connecting them with little vertical pieces as Alex has done in his model*. This would mean adding still more sensitive volumes. I would leave the association between ids and existing ribbon segments under top tiles as is; that is, they would have id
/1/0/41/eMeasureX/ribbon# (0-3)/ segment# (1-5)
for large horizontal segments and
/1/0/41/eMeasureX/ribbon# (0-3)/ segment# (6-7)
for the little vertical pieces behind the bent parts of top tiles; you can see one of these in the upper left corner of the png.
The new little vertical segments would have segment# 8, 9, 10 and 11.
X Ribbons

Y Ribbons

Running detCheck on xmlGeo v1r39p3 and v1r42
IdMap from v1r39p3 and v1r42

Updates for AcdDigi

The introduction of additional fSegment values does not require any changes to idents or AcdDigi. In those contexts we are strictly interested in detectors, not so much individual volumes. AcdIds are detector-level identifiers and the volume ids when used in an AcdId ctor, are stripped of any segmentation information. AcdDigi searches the geometry during initialization to locate all ACD detectors. This search uses the volume names to pick out volumeIds of interest. This bit does have to be modified, due to the new naming organization of the ribbons.