

# Wednesday

*Rob's editorial comments in italics.*

Rob showed his proposal for organizing hit information. See the two attached files.

There were a number of errors in the slides that have been fixed in the versions put at the above location.

The main point of the slides is that the class that holds a collection of hits on one sensor, also holds a reference to the appropriate geometry object. This gets around the need for an IDDecoder at this level of abstraction ( although it may still be needed upstream ).

Tim suggested that at the hit level a pixel detector should return a cellID that is a single int, not a pair. This is converted to (u,v) in local coordinates by a method of the geometry system. One good case to illustrate why this is a good idea is the case of a hexagonal close packed pixel detector, in which the transformation from a cellID, whether 1D or 2D, into (u,v) is complicated and should be hidden from the user. Another case is a strip detector with strips that are not simple rectangles, such as curved strips or phi strips that have a width that is a function of radius. So Rob will modify his proposed classes to be consistent with this idea.

In the new geometry model the geometry class for a pixel detector can return two measurement directions uhat and vhat, as unit vectors in the global coordinate system. The measured quantities (u,v) are offsets along these unit vectors.

People asked if Rob's Hit class could be a RawTrackerHit. This was left open.

There was a long discussion about what is needed for an IDDecoder vs IDHelper in the RawTrackerHit class. The upshot was to replace 3 classes with 2: Get rid of:

- long getCellId();
- IDDecoder get IDDecoder();
- SubDetector getSubDetector();

And replace with:

- (some smarter cell ID) getCellID();
  - DetectorElement getDetectorElement();
- where the smarter cell Id is to be defined. It is capable of identifying a single channel on the sensor defined by DetectorElement. There was disagreement about whether it should just be the channel ID on the given detector or whether it should be the full ID of the channel, including the components that identify the sensor. *I believe that it should be the former since we already have the information about the sensor available via the getSubDetector() method.*