

2005-10-25 LCD Weekly

NG: (showing event display)

Some tracks are being drawn in the wrong direction in vertexing code. (HelixSwimmer?)

JS: Didn't check the GUI. Probably a test is missing to cover this.

TJ: Looked like there was an incorrect abs().

NG: Improve the tests. Fix the code. etc.

JS: Related to fix for negative DOCA tracks failing. (last week)

Checked-in some code that wasn't in yet.

Display vertex and real tracks in WIRED after finding.

Now draws blue ellipsoids for each vertex.

NG: Request for discussion on tracker hits from TJ. May not happen because TN and NS cannot attend this meeting.

JS: Track fitter?

NG: Checked in simple Kalman-based fitter.

Some code to use as a starting point.

News from MC?

MC: Matching of tracks to clusters in the calorimeters. Any tool that is ready for this?

NG: Look at SM's code in contrib area for extrapolating cells, etc.

MC: Niels trying to get back into LCD, but busy with BaBar stuff.

TJ: Want to change arrays to positions in GeomConverter.

Will break a lot of code.

WL: Trouble reading StdHep in Mokka, generated by WHIZARD.

NG: No neighbors using GridXYZ.

GL: Started something on endcaps, but didn't understand what was going on. Put it aside.

NS: JS missed one fix in HelixSwimmer.

RP: Discuss tracker hit stuff?

NG: Yes.

TJ: Cell id contains information from the detector.

NS: Position given by cell id?

TJ: Yes.

Both TrackerRawData and TrackerPulse are 64-bits.

NS: ADC and charge, repeating each other?

TJ: In TrackerPulse, the ADCs are optional. The charge is the integrated charge of the pulse in arbitrary units. Quality (i.e. is well-shaped). Time.

TrackerData = set of ADC values for each of the pulses making up the data.

Merging hits - multiple SimTrackerHits into one TrackerPulse?

NS: Should have a list of SimTrackerHits, because the pulse is combined from the simulated hits.

NG: Or introduce a new class, e.g. MCTrackerPulse.

TJ: May add that to our classes.

RP: Need to be able to trace back to MCParticles.

Don't see any way to do it with current LCIO classes.

TJ: Adding pointers shouldn't be a problem by using LCRelation.

RawTrackerHits correspond to TrackerPulse or TrackerRawData?

NS: Would call TrackerPulse the RawHit?

TJ: SimTrackerHits will make RawPulses.

1-to-1 correspondence between TrackerHit and TrackerPulse?

NS: No. TrackerPulse should have a list of TrackerHits.

RP: TB says charge spready over several strips.

Could record individual strip pulse heights but this is going backwards, because clustering is already done.

NG: Just have a collection of SimTrackerHits.

RP: TB wanted to store RawHits rather than clusters.

NG: TrackerHit nominally has a list of RawHits. Is it TrackerPulses or TrackerRawData that TrackerHit points to? TrackerHit points to a list of TrackerPulses. This is the calibrated data.

NS: This is what you deal with in reconstruction.

TJ: Add cellid to the TrackerHit.

NG: Still want to know layer, detector, etc.

NS: Faster than trying to look at the coordinate.

TJ: Add to next version of LCIO.

NG: TrackerPulse = single CCD pixel or strip readout. This shouldn't be split.

RP: It should have a list of MCParticles?

NG: Use an LCRelation for this.

TJ: Our versions of these interfaces will include pointers to SimTrackerHit list.

RP: Covariance matrix where strips at 45 degree angle -> cartesian coordinates are a bit odd. Could smear by length and rotate, but this is a little odd.

NS: Will need to use sum of covariance matrix in x and y as measurement of error along strip. ? If defined covariant matrix in Cartesian coordinates, a transformation will be needed to do actual fitting or finding. My error matrix will not give covariance matrix in cartesian coordinates. Gives in direction perpendicular to the track direction. This can be transformed in x,y,z. Error in plane perpendicular to the track. At every point, it will be perpendicular to the track. It doesn't depend on direction, because multiple scattering will be the same answer. Scatter angle in the plane. It will project on any surface where measurements are done.

NG: Still some questions about how to use TrackerHit correctly.

RP: At some point, need to encode the strip or pixel numbers.

NG: Have not defined this yet.

TJ: Can use mechanism from calorimeter encoding. But how to deal with size of pixels or strips? Just make error solely based on size of strip or pixel?

NG: Will need to know orientation and length based on IDs. This system is to be defined.

RP: Type would distinguish a pixel from strip device or something else?

TJ: Type actually tells what type of hits are pointed to. Could also include other information but there is not enough room to store the cell id right now.

RP: It would be nice to whether dealing with pixel or strip device.

NS: If ADC counts are included, should have Digitizer object to store what was the scale, ADC separation, scale, etc. Need to describe the parameters of the ADC. Can store the result of the digitization or can feed parameters to digitization code. Just ADC counts are ambiguous.

TJ: Use info to interpret the ADC counts?

NS: Yes.

TJ: Can put parameters into the collection.

NS: Something like this is needed within event.

Example - What is conversion from charge to ADC counts?

RP: Several concepts.

Pitch, orientation of strips, etc.

TJ: Geometrical parameters should probably go into the Conditions DB.

NG: In current digitizer, read in parameters from properties file?

NS: Have CCD electronics specification.

TJ: Function of the detector, then it can live in the Conditions DB.

NS: But may have same detector, but may want to try different CCDs. Different noise levels, thicknesses, etc. As soon as digitized data is to be stored, need to characterize what made it. For CCDs, there are a large number of parameters.

NG: Can define tag that points to some properties files.