

Dear Lia,

thank you for your response to the discussion I brought up during the EVO meeting. I am sorry that this caused some confusion and I am glad that you wrote the explanation of the situation for the STT.

Please let me clarify my point a bit.

When I look at the STTPoint I see the following information stored in it:

- fx, fy, fz coordinates at wire center
- fx_in_local, ... local coordinates at entrance of act. volume
- fx_out_local, ... see above
- fx_wire_dir probably direction of wire (not explained in .h file)
- fxtot, fytot, fztot absolute coordinate of hit (to be deleted if I understand da cancellare correctly)

What I think you need in this file are the entrance and exit point in the global coordinate system of your detector and a unique identifier for your fired wire which allows you to extract the position of the wire out of the geoManager.

With these informations you can calculate all the rest.

So for me the most important information is fxtot, fytot, fztot which allows you to directly compare the result of your reconstruction with the MC information.

STTHit contains:

- fx, fy, fz coordinates at wire center
- fIsochrone radial distance from wire
- fRadial position along the circle in xy-plane (I assume that this is the position on the isochrone)
- fWireDirection (no explanation but it looks like fx_wire_dir from STTPoint)
- fRsim ???
- fRtrue ???
- fXint, fYint, fZint position of interaction

The information of fx, fy, fz and fWireDirection is available in both files, so this data can be reduced.

What I am missing is a STTDigi class. A digi file should contain all the information which you would get from your detector like the fired wire, the time, information of the pulse shape and the deposited charge and so on. All what your real detector would give you. On this data you would base your reconstruction on without accessing the MC data. This simulated data can later on be easily replaced by your real detector data and you can use the same reconstruction

code on simulation and real experiment. In your code digi data (like the isochrone) and reconstructed data (like fRadial and fXint,...) is mixed up which means that you always have to do the digitization and reconstruction in one task or your STTHit is partially empty.

In the MVD code we tried to avoid this to be able to test different reconstruction methods on the same digi data without doing the digitization again.

I hope this answer makes my point a bit more clear.

Ciao,

Tobias
