

LArSoft - Idea #17307

Idea # 17279 (Assigned): Create an example about how to produce a custom ROOT tree out of LArSoft data

create a gallery example

07/31/2017 10:34 AM - Lynn Garren

Status:	Assigned	Start date:	07/31/2017
Priority:	Normal	Due date:	
Assignee:	Saba Sehrish	% Done:	0%
Category:		Estimated time:	12.00 hours
Target version:		Spent time:	0.00 hour
Experiment:	-		
Description			

History

#1 - 07/31/2017 10:35 AM - Lynn Garren

- Status changed from New to Assigned

- Assignee set to Gianluca Petrillo

#2 - 09/13/2017 11:45 AM - Gianluca Petrillo

- Estimated time set to 12.00 h

#3 - 10/12/2017 02:45 PM - Katherine Lato

- Assignee changed from Gianluca Petrillo to Saba Sehrish

#4 - 10/16/2017 03:32 PM - Thomas Junk

Some gallery examples are available in `dunetpc/dune/GalleryScripts`. There's a 3D spacepoint display, a `raw::RawDigit` display, FFT and channel correlation tools. Nothing DUNE-specific in fact.

#5 - 10/16/2017 03:35 PM - Thomas Junk

Oh, I didn't see the requirement here -- to make a custom ROOT tree, though they do make graphs, and changing them to make trees would be easy.

#6 - 10/16/2017 04:24 PM - Thomas Junk

Here's a gallery macro that writes a custom ROOT tree:

```
/dune/app/users/trj/augcollab/gallerydemo/gallery-demo/rdtree.C
```

#7 - 10/16/2017 04:42 PM - Gianluca Petrillo

My ambition would be an example of a tree which is not so flat.

It should not be so hard with ROOT 6, and I think it's good value for people trying to do a real analysis.

I am thinking of something as complex as having a branch with a STL vector of Track objects where:

```
struct Hit {
  geo::WireID wire;
  double      time;
  float       charge;
}; // struct Hit

struct Track {
  TVector3 startPoint;
  TVector3 endPoint;
  double   length;

  std::vector<Hit> hits;
}; // class Track
```

(using the horrible TVector3 may have some *sense* in such a tree)

Notes:

- the data structure is nested as opposed to flat (Track objects contain Hit objects)
- the data structures depend on LArSoft data types
- the example must also include how to read that tree
- there is no association between objects in different branches (i.e. no TRef): hits are *contained*. Creating that type of references is not going to be LArSoft- not gallery-specific.