`LArSoft` Librarians' meeting

Lynn Garren, Gianluca Petrillo, Erica Snider

May 20th , 2014

Today's agenda

1. `LArSoft` release status

2. Test and validation workshop

3. ClusterCrawler module redesign and algorithm interface proposal

4. Projects and issues update

LArSoft release status

Update by Lynn Garren in a moment...

Test and validation workshop

The workshop will take place at CDF, FNAL, on Tuesday, June 17th (afternoon) and Wednesday, June 18th (full day).
The proposed agenda draft includes:

- testing and use of the Continuous Integration System (following on from the morning breakout sessions)
- librarians demonstration of existing test/validation modules
- user coding of new test and validation modules
- user led discussion of algorithm improvements
- community discussion of future developments and needs from LArSoft

The latter two points are expected to be led by the participants from the experiments.

Mark Dykstra will be, in the next six weeks, putting together a full test path to be run via a scripted build system ("Jenkins").

Redesign of ClusterCrawler module

- the structure of ClusterCrawler module is currently not extensible to accommodate LBNE needs for disambiguation
- together with the experiments, a non-generic but still flexible solution has been agreed upon:
- the new module will have four slots for
  1. hit finding
  2. disambiguation
  3. cluster finding
  4. hit refining
- any of the appropriate algorithms should fit in each slot
- we'll ask Tyler about the disambiguation step and how it can be made into a relocatable algorithm
- we'll come out with a work plan soon

Project and issues update

Memory footprint for LBNE full detector geometry

Two two memory hogs have been identified:

1. in GEANT geometry *optimization*; why is that? the investigation is proving to be extremely painful... but still progressing
2. in Geometry service: wire information is largish, times 786000

Code to address the second problem has been committed:

- redundant information has been removed from `geo::Wire` (intention blessed by B. Rebel)
- ROOT structures replaced by CLHEP
- no loss of functionality nor interface change, but some less common operations might be slower
- memory usage of Geometry service dropped from 250 to 120 MB

Compression of raw digit (`raw::RawDigit`)

See the next talk...