

# LBNE Concerns for the LArSoft Stakeholders' Meeting

Tom Junk  
*Fermilab*

March 12, 2014

- **Memory Usage:**

- Simulation job has a high-water mark of 4 GB for the 10 kt FD.
- Better for the 4-APA mini-FD and the 35t prototype: less than 1 GB
- 2 GB taken before the first G4 step completes. Suspect much of it is in geometry
- Many thanks to Gianluca for spotting an inefficient vector resize! Eliminates a spike.
- Reconstruction: Already addressed. Do not put recob::Wire in the event. Go straight for hits.

But -- we lose some functionality this way. Would like to display recob::Wire and hits on the same event display. Possible with 35t and 4APA

Another mitigation -- zero-suppressing recob::Wire

## LBNE Code Management

Personnel are busy with many tasks:

Tom, Mike, Maxim, Liz, Brett, Qizhong are regular contributors to the software meetings

Many thanks to Erica and Gianluca and Mike for getting lbncode factorized!

Developers need to commit their code to the repository more frequently (even if it is not perfect)

## CVMFS

LBNE collaborators are eager to try this out.

Some confusion -- some thought that it was binary-only (we distribute source too this way I assume)

Event display is slow over remote X connections

## LBNE Code Concerns

Reconstruction code may not factorize easily into LBNE-specific and LBNE-agnostic pieces.

Stitching clusters together across APA gaps and across APA frames may require re-clustering.

Reducing the wire angle should help -- fewer ambiguities. Possible concern -- an ambiguity which can only be broken by information from the neighboring APA.

Already have this in determining which side of an APA activity is on: Special feature of LBNE.

Is factorizable disambiguation enough? Or does it have to be woven into event reconstruction at each step?

## LBNE Code Concerns

Code ownership and maintenance responsibility  
→ LBNE should take responsibility for maintaining its own code.

But we depend on shared code, which we are happy to use and contribute to (if we only had enough contributors with spare time).

Share in common maintenance? What if LBNE needs something specific? We should provide it.

Currently a low-FTE issue. Lots of names, though only partial effort from each.

## LBNE Online Concerns

artDaq + LArSoft:

Online monitoring: LArSoft (? or standalone) modules which can plug into simulation as well as the DAQ so we can prototype and test them.

Event filtering and triggering: Not clear if we need this -- it may be faster to write data than to process it (depends on how much CPU we have and how efficient we are at writing trigger/filter algorithms)

Online code management, repositories, and releases (not really a LArSoft issue).

# Computing Concerns Related to LArSoft

## Grid Jobs

- Virtual memory/job slot
- disk and tape -- being solved with the new dCache disk and SAM. Still a bit at the mercy of BlueArc
- Distributing non-data data:
  - Photon lookup libraries
  - GENIE cross section splines
  - G4 data files
  - Conditions database
  - smaller stuff (GDML, etc).
- Non-FermiGrid OSG sites -- presumably CVMFS will make it work.
- data handling: ifdh, xrootd, SAM