

DUNE Software News and Announcements

Tom Junk

DUNE Software Architecture and Management Meeting

November 3, 2023

Recent and Upcoming Events

- DOE Review of DUNE Software and Computing, July 13-14
 - <https://indico.fnal.gov/event/60441/>
 - Report available in DUNE-Doc-29392
- DUNE Collaboration Meeting, Santa Marta, Sep. 25-29
 - <https://indico.fnal.gov/event/58097/>
- DUNE Computing Project Management Meeting, Oct. 11-13
 - <https://indico.fnal.gov/event/61352/>
- DUNE Framework and Core Software Review, CERN Dec. 11-13
 - <https://indico.cern.ch/event/1340290/overview>
- DUNE Collaboration Meeting, CERN, Jan 22-26, 2024
 - <https://indico.fnal.gov/event/60987/>

From the DOE Reveiw

Using common tools from across FNAL/HEP and adding DUNE adaptations is a good strategy for sustainable software development. However, dependencies from and requirements for **external software deserves more attention**. US DUNE S&C may want to work with external developers of the tools DUNE depends on to understand and negotiate plans and deadlines and, if necessary, make the in-kind contributions to ensure tool capabilities are available for DUNE on time.

[emphasis added]

From the DOE Review

There are essential tools, such as visualization and end-analysis tools, that do not currently appear in DUNE S&C planning. It would be beneficial to conduct a complete survey of required software and develop a plan to identify responsible institutions within DUNE, to ensure all requirements are considered.

From the DOE Review

The development of the DUNE S&C framework appears to be on the critical path to delivering a system that can facilitate the physics goals of the collaboration. The current milestones do not provide time for redevelopment of elements that will rely on the DUNE framework, or refinement from physics analysis requirements and machine learning workflows that may arrive when DUNE collaborators begin widespread use.

Translation: Framework development may pinch us for time.

Spack Transition

- SL7 is EOL in June 2024. Next up: Alma Linux 9
- UPS will not be supported on AL9
- We will have to use Spack at that time; exact timeframe still tbd
- Some feature requests of the Spack developers are pending
- DUNE DAQ is already using Spack

- I run SL7 in a container on AL9 and it seems to work
 - The SL7 container that grid jobs run isn't enough to build dune software
 - I have a customized container image in `/dune/data2/users/trj/containers/trjsl7.tgz`

New Compiler Versions

- Old ones: dunesw versions v09_78_06d00 and earlier (but not too far back!)
 - **e20** GCC v9.3.0, -std=c++17
 - **c7** Clang v7.0.0, -std=c++17
- New ones: dunesw versions v09_78_06d00 and later
 - **e26** GCC v12.1.0, -std=c++17
 - **c14** Clang v14.0.6, -std=c++17
 - Still missing tensorflow

<https://cdcv.s.fnal.gov/redmine/projects/cet-is-public/wiki/AboutQualifiers>

Many thanks to Lynn Garren for providing branches for our repos to fix minor issues caught by the new compilers. Tom reviewed them, merged them, and fixed the remaining issues Lynn asked us to look at.

GArSoft v02_20_00 is also at e26 and c14. Found an issue with a ROOT header that was already fixed by ROOT but not in our version.

dune_pardata Maintenance

- Historically it's been Alex Himmel doing this, but he doesn't have time anymore.
- StashCache solves the same problem, but the versioning is different (i.e. you do it yourself with file and directory names – there's no UPS).
- Files in StashCache are in DUNE's persistent dCache space and exported via CVMFS. Files are owned by the dune account – minimizes chance of accidental loss and allows for future maintenance even if original owners leave.
- Recent additions for SNB flux files just went in to StashCache

nopayloadclient and dunenpc

- Items from Lino Gerlach
- Many thanks to Lynn Garren for adding an SSI build shim for nopayloadclient
- I had gotten started with it but got stuck
- SSI build shims to be replaced with Spack recipes.
 - hopefully this puts the burden on package developers for providing pluggable builds, but sometimes we have to modify components and even edit code.
 - Example – nopayloadclient comes with a copy of nlohmann_json
 - but so does DUNE-DAQ.
 - and ROOT! Look in \$ROOTSYS/include/nlohmann
 - Versions were not compatible
 - Lynn straightened this one out.

sand-reco

- Matteo Tenti has copied sand-reco from SAND's gitlab instance to DUNE's github organization as sand-reco
- Additions to make it UPS-comptible – a UPS directory with product_deps based on other DUNE products, and enough CMakeLists.txt file changes to get mrb to run.
- mrb does not build sand-reco products however – no libraries or executables in localProducts when mrb is run on it
- Tom has been experimenting with a private copy, getting mrb to build the executables inside of it, and found that there are a lot of compiler warnings which are treated as errors.

Geometry Refactorization

- Kyle and Erica have been working on reorganizing the geometry interface in LArSoft
- Needed to remove some assumptions and couplings
 - WireReadout is not really part of the detector geometry that GEANT4 needs to know about
 - Some detectors have pixels instead of wires

<https://github.com/LArSoft/larcoreobj/pull/19>

<https://github.com/LArSoft/larcorealg/pull/37>

and a PR for most of the repos in the dunesw stack – hold off for v09_82_00d00

To-Dos

- Keep up with PR review
- Make a release notes web page: Follow example of LArSoft and put on GitHub
https://github.com/LArSoft/larsoft/releases/tag/v09_81_00
- Spack/cetmodules. Remove things like `find_ups_product()` in CMakeLists.txt files. `find_package()` is not a direct drop-in replacement however (esp. ROOT libraries and I had a problem with gsl)
- HDF5 XRootD static interface
- Coding Guidelines web page – See the draft at
https://docs.google.com/document/d/1rRgGwGDOh5Z9XV0_R46M1RrHRimWPVjEeDMTp7KSJMk/edit
- Doxygen/LXR
- Memory Usage Task Force
- Straighten up fcl files – many duplicate `#includes` in `services_dune.fcl`

US-DUNE Software Management

From the DOE Review Recommendations, number 1:

1. US DUNE is encouraged to clearly delineate the US responsibility in the DUNE S&C system beyond hardware contributions. For software activities that may also be supported by international partners, the specific features and attributes that are a sole US responsibility should be identified. This should be done by December 2023.

Ken Herner has asked me to be a L2 manager for this effort, with the main job of identifying milestones and setting schedules for US DUNE software contributions.

Software work is easily transferred across international boundaries. Institutions like to develop algorithms and contribute to the physics. Less so for software integration and management.