DUNE Software Architecture and Management Status and Plans

Tom Junk

DUNE Software Architecture Meeting

February 2, 2024

Important Dates

- SL7 reaches end of life (EOL) June 30, 2024 -- RHEL 7 came out initially in 2014.
- CSAID wants to transition the dunegpvms to AL9 well in advance: March 20 was proposed by Seth Graham
- Lynn Garren pushed back and mid-May was settled on

- A date in the past – Jan 16 meeting with Steve White, Marc Mengel, Patrick Gartung and Kyle Knoepfel at which it was announced that there is a Spack installation of LArSoft in CVMFS.
- We had been waiting a long time for this, and now we can proceed to build and test dunesw in Spack

AL9 Hardware Resources

- dunebuild03.fnal.gov
- dunegpvm16.fnal.gov

My SLF7 Container

- I use this on my AL9 desktop
  /dune/data2/users/trj/containers/trjsl7.tgz
  unwind the tarball and do this:

  ```
  APPTAINER_BIND=/etc/hosts,/etc/localtime,/cvmfs,/dune,/run/user apptainer shell /path/to/unwound/container/trjsl7
  ```

- Tom has been attending the SL7 container task force meetings.
- Patrick Gartung has given instructions on how to start an interactive session using the batch worker node SL7 container (it's in CVMFS).
- That container lacks some RPMs needed for development. I added what was needed to run mrb builds in the container above.
- Mine lacks some token tools and the forge_tools debugger cannot allocate a Pseudo-Terminal for some reason

sandreco as a UPS product in CVMFS

- Matteo Tenti asked in July 2023 for sandreco to be tagged and installed in CVMFS for running on the grid.
- Prior workflows involved building the software in grid jobs (I think...).
- Software was located in a GitLab instance at INFN. Nowit is located in DUNE's GitHub area at https://github.com/DUNE/sandreco

  Matteo is the admin.
- Matteo has adjusted the CMakeLists.txt files and added a UPS directory with a product_deps (though some other files are missing)
- sandreco builds with mrb i, but it doesn't quite make a setuppable UPS product.

sandreco as a UPS product in CVMFS

- sandreco depends on edepsim and products edepsim depends on: geant4 and root.
- sandreco, when built, contains several standalone executables
  Analyze
  Digitize
  Display
  FastCheck
  Reconstruct
  EventDisplay
- The sandreco bin directory gets added to $PATH. These names are rather generic and may collide with other packages if it is meant to coexist with other software. "/usr/bin/analyze" (no caps) is a system command for database analysis.

sandreco as a UPS product

- The CMakeLists.txt files were set up in sandreco to put the executables in LocalProducts<tags>/bin but there's no sandreco directory or .version directory
- mrb mp finds nothing in the right places
- Tom tried rearranging the source tree to look more like the executable makers in dunecore but ran into compiler warnings.
- Would prefer not to have to rewrite software just to get it to install
- UPS is also vanishing with SL7. On AL9 we'll use Spack
- Tom chatted with Matteo at the January CM, and proposed a stopgap solution – handmade UPS product.

sandreco as a UPS product

- It took a lot less time than expected

    sandreco v01_00_00 –q e20:prof

is now available in CVMFS. Files were rearranged and a UPS table file added.

- No .cmake files provided, so building additional products on top of this one may not be as easy.
- Not too hard to update to a new version
    - Rebuild interactively, copy the product into a scratch area, copy the items in, and update version numbers. Install and test.
    - Stopgap until May. Will want a Spack recipe, but edepsim has to be Spack-ified first.
    - GArSoft needs edepsim too. Can call it edep-sim now (UPS is hyphen-phobic).

Spack Resources

- https://fifewiki.fnal.gov/wiki/Spack
- https://spack.readthedocs.io/en/latest/basic_usage.html
- n.b. Spack documentation on the web is versioned. Some commands evolve and reading the wrong docs can cause confusion.
- LArSoft uses Spack 0.21.2dev0
- Get this with
  spack --version
- see output of spack –help
- Kyle gave a talk on Spack development at the LArSoft coordination meeting last week
  https://indico.fnal.gov/event/63013/

Work plan for Spack transition

- Test functionality of LArSoft Spack installation using AL9
- Test functionality of modifying a local copy of a LArSoft package
- Build the dunesw stack in Spack in a private disk area using the LArSoft installed packages
- Check basic functionality and iterate until known workflows work (e.g. MC jobs, ProtoDUNE data reco, and other workflows).
- Rebuild and install dunesw in CVMFS
- Perform physics validation
  - Kirby is contacting physics groups to ask them to think about and produce validation plots and criteria
  - It's not just the Spack transition that can benefit from this (compilers..)
  - Missing-file errors should be fatal, but code that falls back to an option may behave non-reproducibly

Setting up LArSoft in Spack

- Log on to dunegpvm16.fnal.gov

```
source /cvmfs/larsoft.opensciencegrid.org/spack-packages/setup-env.sh
spack find larsoft
spack info larsoft
spack load larsoft@09.81.00
```

n.b. /cvmfs/larsoft.opensciencegrid.org/packages contains old stuff that is only a distraction (tutorial is a bit out of date).

n.b. loading larsoft does *not* set up gcc like the UPS one does. Compiler may be needed by interactive ROOT sessions.

gcc version is 12.2.0, one point release newer than e26 = gcc 12.1.0

spack list gives a long list of packages, not all of which can be loaded.

Little Things ...

- I noticed Spack puts things in ${HOME}/.spack, as well as in /tmp/<username>/spack-stage

- These can be issues if $HOME is not defined (actually I didn't try undefining it – could be it uses ~), or if ones Kerberos ticket has expired. Or if /tmp doesn't exist or is not writeable.

- They seem not to be too important – I deleted my .spack directory and spack just made it again.
- Raises reproducibility questions if a build or a setup depends on the contents of these directories. (I am told they don't)

LArSoft Spack Environment

```
<dunegpvm16.fnal.gov> spack env list

==> 4 environments

    critic-2-12-04-gcc-12-2-0-cxx17-prof-gcc-11-4-1  larsoft-09-81-00-gcc-12-2-0-cxx17-prof-gcc-11-4-1
    gcc-12-2-0-gcc-11-4-1                            nulite-3-15-04-gcc-12-2-0-cxx17-prof-gcc-11-4-1
```

So.. what's the compiler version?

```
spack env activate  larsoft-09-81-00-gcc-12-2-0-cxx17-prof-gcc-11-4-1
```

doesn't set up the compiler, but you have the option to load 11.4.1 or 12.2.0

I suspect the SciSoft team built gcc 12.2.0 using gcc 11.4.1

Executing the Work plan for Spack transition

1. Test functionality of LArSoft Spack installation using AL9

   e.g. lar -c eventdump.fcl <artrootfile>

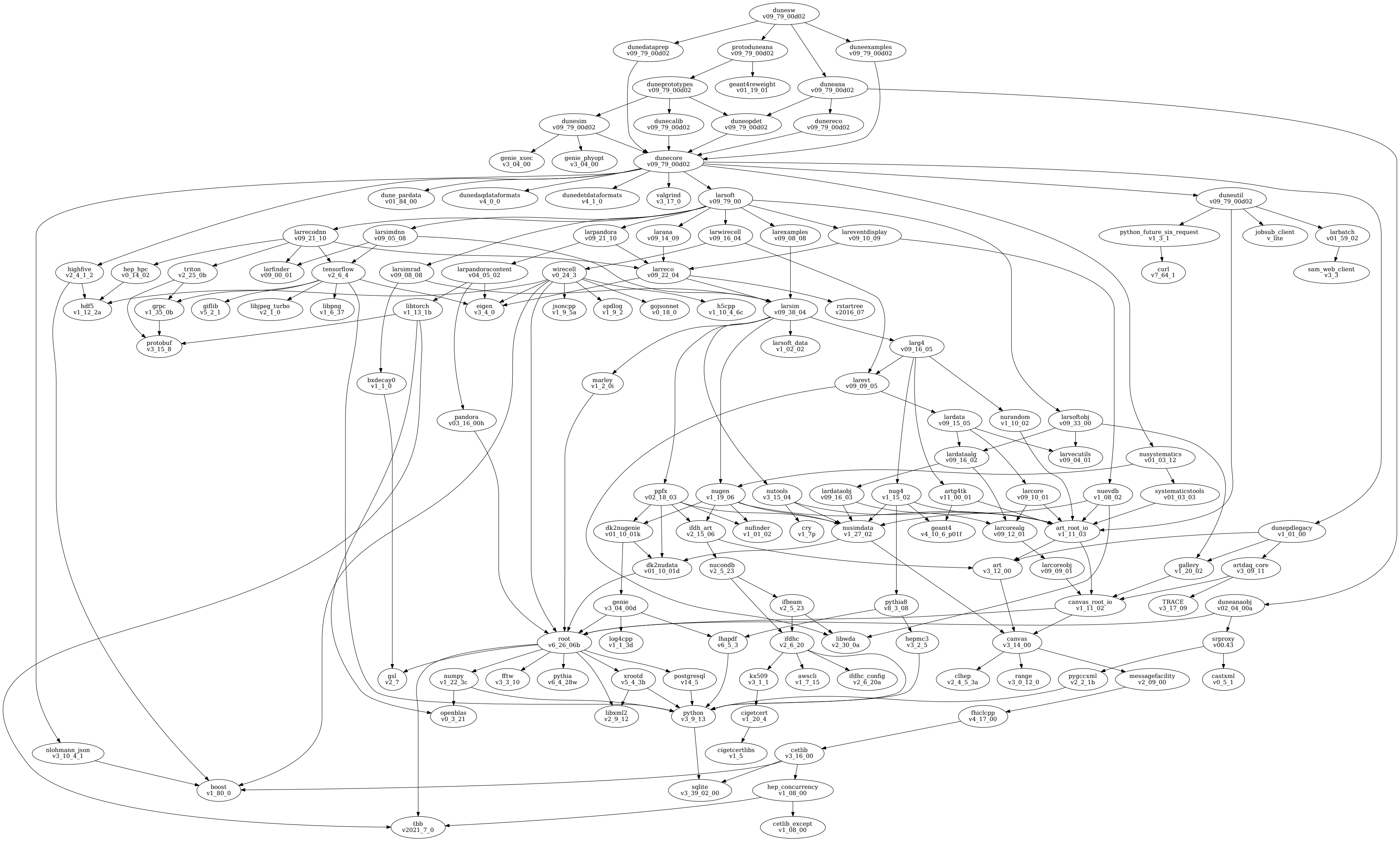
This failed. FHICL_FILE_PATH points to nonexistent directories. Feedback to scisoft given. This may have been remedied with symlinks as a workaround for v09.82.00.

1a. Or more basically, check to see if these framework executables work (no LArSoft needed)

```
lar --help
product_sizes_dumper
cetskelgen
```

- These latter work! Except cetsklgen -- missing some Perl module.

The dunesw tree



Special dunsw products

- dunepdlegacy is a standard mrb-built product, just not released every week. Should be converted along with the rest.
- dunedetdataformats and dunedaqdataformats
  currently header-only products with some scripts and other non-built items. Should not be too hard to install these.
  Need to set up include paths so builds of other packages work
- dune_pardata is also a bespoke repo
- Not dunesw products but still in our management remit:
  edepsim
  nopayloadclient
  sandreco
  garsoft

Other to-do items

- **Complete the style guide**

A start, but really it's just *art*-centric, and functionality-based, not style-based

https://docs.google.com/document/d/1rRgGwGDOh5Z9XV0_R46M1RrHRimWPVjEeDMTp7KSJMk/edit?usp=sharing

It contains prescriptions such as using FW_SEARCH_PATH to find files instead of absolute paths.

It has links to other style guides. (DAQ, LArSoft, SBN, Wire-Cell)

There is a script format_code that runs clang-format to tidy up style (spacing, indentation. Does not rename identifiers however)

Other To-Do Items

- Propose a list of responsibilities for the US-DUNE component of the Computing Consortium for Software operations
- Just about anyone in the world can work on DUNE Software – it does not require physical presence, though being plugged in to DUNE's physics and analysis efforts is a big help.
- Things we do and would like help with
  - Release management
  - Spack transition (and all other technology changes with software implications)
  - Pull request review
  - Coding guidelines
  - Training
  - Distribution of algorithms along with data to be FAIR
  - Interface with Fermilab's CSAID (FIFE, frameworks, LArSoft, computing...)