



Towards a full Spack / Spackdev – based build ecosystem for *art*-based experiments

Chris Green, FNAL, 24 January 2018



Goals

- A maintainable, sustainable build and development ecosystem for (at least) experiments using the art framework.
- Replace the current system of:
 - making “relocatable UPS” packages of external packages using ad-hoc build scripts, relying on use of e.g. LD_LIBRARY_PATH for relocate-ability (not viable with MacOS / SIP); and
 - “cetbuildtools” and ”MRB,” home-grown CMake-based systems for developing packages singly and together, respectively, which rely on the environment to find UPS packages.

with:

- External packages built using Spack recipes; and
- Experiment packages developed using “cetmods” (“de-UPSed” cetbuildtools) and Spackdev.

Progress

- A set of 25 milestones which, when achieved, will demonstrate the layered functionality required to replace the existing system, including:
 - The ability to build and use a guaranteed coherent set of packages (dependent versions, compiler, etc).
 - The ability to relocate an already-built set of packages for use on a different machine and/or directory tree.
 - The ability to check out and develop a code base spanning multiple packages, interacting with pre-built external and experiment-authored packages as appropriate.
 - Eventual sign-off from all stakeholders and retirement of the old system.

Progress

- Addition of, “buildcache” functionality to Spack, per previous reports from Patrick.
- Identification of possible trouble points which will require investigation and possible further code development:
 - Making available a fully-coherent release along all the axes we care about: compiler, MPI implementation, debug vs profile, etc., python version, etc.: Spack load / Lmod has limitations.
 - Avoiding unnecessary duplication of builds of non-compiler-specific packages: binary-only, data-only, C-only / Python-only / C & Python, etc. Could be multiple 10s of GiB in wastage.

Current work

- De-UPS-ing cetbuildtools.
- Refining milestones and reviewing / assigning priorities to requirements in case of technical issues.
- Implementing milestones.