



# Cetmodules 2.X Status and Highlights

Chris Green, FNAL  
*2021-02-23*

## Recap

- Progressing towards total replacement of UPS by Spack in experimental software ecosystem.
- cetmodules 1.X: UPS-free CMake-based build system based on cetbuildtools.
- Testing and improving Spack-related operations is difficult because (*e.g.*) experimental code must be forked to remove reliance on UPS for both building and use.

## cetmodules 2.X

- Remove the need for forking experimental software for building with Spack: retain the ability to build with / for UPS while also being buildable with Spack.
- Improve cetmodules to embrace “modern” (c. 2014) CMake paradigms:
  - Targets vs CMake variables with library filenames.
  - INTERFACE and OBJECT libraries and PUBLIC vs PRIVATE vs INTERFACE dependencies.
  - Components.
  - Handling of transitive dependencies.
  - Automatic generation of CMake configuration files.
  - Automatic generation of package checksums.
- Spack / UPS build compatibility relies on “Project variables,” managed by cetmodules, with configuration translated from `product_deps` to CMake by `setup_for_development` and `buildtool`—use of `buildtool` is now **required** for at least the CMake stage.
- cetmodules-using packages can use cetbuildtools-using dependencies via `mrbs` or UPS.

## Upgrading from cetbuildtools

- Initially: upgrading to cetmodules is *not necessary*—simply upgrading to cetbuildtools 8.X should be all that is required to use cetmodules-built UPS packages.
- Via mrb 5.X, can develop simultaneously an arbitrary mix of cetbuildtools / cetmodules-using packages.
- Incremental best practice improvements to use modern CMake features via cetmodules will reduce library size and dependencies.
- Build-ability with Spack will require:
  - Spack recipes.
  - Migration of configuration from `product_deps` to project variables set in the project's top-level `CMakeLists.txt` file. The latter takes precedence -> single point of maintenance.
- As changes are made to accommodate and ease building for Spack, retain the ability to build for UPS and use UPS-packaged dependencies.

## Status

- Addressing some remaining backward / forward compatibility niggles.
- Almost complete (hopefully this week) -> art suite 3.07.X
- Documentation in process (couple of weeks).

## Rollout

- Discussion of new and technical aspects with external projects / experiments where necessary.
- Technical help available for corner cases, issues, making best use of modern CMake paradigms, *etc.*
- Completion of documentation.
- artdaq-core, TRACE, nutools, LArSoft.