# High Impact Changes for Spack `develop` and 1.0

Chris Green, FNAL

*Spack Migration Team Meeting, 2024-12-12*

# Overview and Motivation

- Spack 1.0 is currently planned for June 2025.
- Todd's announcement describes some upcoming general high impact changes, with links to further details for each.
- This presentation will introduce those changes and discuss the implications and risks for us. Some of them will need to be monitored and possibly steered in order to ensure they satisfy our use cases.
- Some changes are previewed in release v1.0.0-alpha.2 as they have not yet been merged to develop.
- Other changes of importance are proposed, upcoming, or already merged to develop, and should also be monitored for suitability and disruption.

# Compilers as Build Dependencies

- Open PR Turn compilers into nodes #45189
- Also relevant: Slack comment from Harmen Stoppels:
  *... The TL;DR is I'm claiming making Spack aware of the C++98/C++11 ABI breakage is a waste of time right now.*
- This could be *very* good for us.
- It could also stop us in our tracks. Specifically, if the new paradigm is too rigid, it may prevent us using system libraries compatibly built with the system compiler in conjunction with libraries and applications built with a different compiler.

# Migrate Package Recipes to Their Own Repository

- Open Issue Discussion: Move built-in packages to a separate repository #47480
- This will change the way we work, and will necessitate some script changes (maybe).
- In general, this should be good for us.

# Include Source Provenance in `spec.json` and Package Hashes

- Open PR specs: include source provenance in spec.json and package hash #32312
- This could be straightforward and non-disruptive.
- Or it could vastly worsen the rebuild problem unless we rethink the way we work.

# "Stateless" Spack

- Draft PR Shared spack #47615
- This should be very good for us, especially for CVMFS.
- Need to keep an eye on it to make sure that it's everything we need, and not just *almost* everything we need.

# Environments

- Open Issue Discussion: Environments: Support fully including other environments and standardize include syntax #46973
  - Draft PR Support conditional and optional includes #46792
  - Draft PR Allowing environment variables to be set in a spack.yaml #47587
  - Draft PR Environments: support including other environment files #42442
  - Open PR Command: Add option to add/remove environment includes #44285
  - Merged PR Add configs for reusing specs from environments and enable reuse from included concrete environments by default #45139
  - Open PR Environments: refresh cached URL-included config on re-concretization #34199
- A *lot* of material here that we should understand the implications of, and steer and/or report problems with if necessary.