

Spack LIM, March 15, 2021

Connected: Andre Sailer, Ivan Razumov, Reiner Hauser, Graeme Stewart, Liz Sexton-Kennedy, Johannes Elmsheuser, Marco Clemencic, Valentin Volkl, Ben Couturier, Pere Mato, Gerardo Ganis, Marcin, Dmitri Konstantinov, Philipp James Elson,

Apologies: Oksana

Indico: <https://indico.cern.ch/event/1016908/>

Preliminary Evaluation of Spack as a new backend for LCG release

Slides by Ivan

Discussion Points:

- Build everything with spack or rely in system packages?
- ROOT dependencies: builtin or external?
- Target architecture? Haswell, Nehalem? Something else?
- Use rpath, runpath, remove information and use LD_LIBRARY_PATH?

The spack installation on /cvmfs/sw.hsf.org/ - HOWTO for Librarians and Lessons from Key4hep

Slides by Valentin

Questions:

Marco:

- What does adding all paths to relevant environment variables mean? Making rpath not necessary, or only those that are needed?
 - Valentin: Setting up LD_LIBRARY_PATH for runtime dependencies, e.g. Ctest for Gaudi.
 - Marco: IIUC, Spack uses the rpath during build time, so setting LD_LIBRARY_PATH is useless because rpath has precedence. Trying to understand if setup.sh just sets necessary LD_LIBRARY_PATHS for plugins. Would prefer to keep the default treatment that spack does
- Marco: How to define version for all packages, to fix versions as is done in the LCG stacks?
 - Valentin: Should be possible to, e.g., install a new version of podio on top of existing LCG stack, if spack basically unchanged

- Marco: Build everything with spack... I would again prefer to do what spack does. Can also build compilers with spack. Can benefit from the existing recipes in spack and build everything with spack instead of using system packages. Focus effort on getting coherent set of recipes.
 - Ivan: I hear you
- Root external dependencies:
 - Marco: Root should use external libraries as much as possible. Had the problem with the nlohman json library recently. Should have consistent set of libraries.
 - Gerri: Remember that ROOT said that they have 'tuned' builtins?
 - Marco: Should coordinate the recipes between ROOT and externals. Unless it is hidden inside a library and doesn't cause conflicts.
- Gerri: Valentin, did you build libc?
 - Valentin: used libc provided by gentoo-prefix and then continue with spack. Somethings are in different places, but seems like an interesting approach. Spack is investigating towards building from glibc upwards.
- Target architecture, different builds with different micro-architectures?
 - Marco: For LHCb, support starting from Nehalem as current minimum. Can give up relocation if whole stack can be build easily.
- Pere: how do you specify the target architecture?
 - Ivan: In the config file for spack.
 - Pere: Why are all packages in different paths containing architecture and compiler?
 - Valentin: Probably just cosmetic to organize packages.
- Reiner: For generic LCG releases and micro-architecture, use flags as proposed last time, and by RedHat, v2, v3 etc. Not 'Nehalem' etc.
 - Marco:
- Reiner: What about dbg vs. opt?
 - Ivan: For Cmake this is possible out of the box, but not for other types of Packages.
- Reiner: build on build machines and then install on CVMFS. How is this done? BuildCache?
 - Ivan: just copy from build machine to cvmfs
 - Reiner: RPM-like binary distribution?
 - Ivan: There is the build cache, have set it up but haven't tested it yet.
 - Reiner: Not yet using CVMFS for online.

Communication:

Spack channel in SPI Mattermost: <https://mattermost.web.cern.ch/spi/channels/spack4lcg>