

ロン (月) (コン(ヨ) ヨークタの

Software management with Spack

Trevor Khwam Tabougua

Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen Burckhardtweg 4, 37077 Göttingen

October 16, 2023

Learning objectives



- Understand the challenges of software installations in HPC.
- Formulate valid spack SPECS
- Install packages on the cluster using Spack.
- Manage different software versions.

Installing software on a Cluster



- Prepackaged software typically requires administrator (root) privileges
- admins cannot install all software required by users
- software installation very complex
- different tools may need different versions
- often trade reuse and usability for performance
- Compiling on the target system often yields better performance

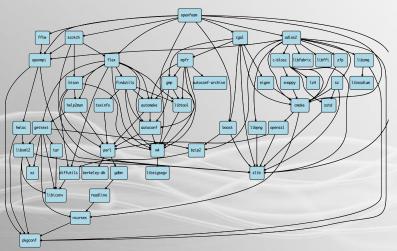
HPC Software is complex



- coexistence of several builds
- specific versions of compilers, MPI, libraries, dependencies
- often many dependencies

Openfoam dependency tree





HPC Software is complex



- specific versions of compilers, MPI, libraries, dependencies
- often many dependencies
- many compiling options
- users active on several clusters

Solution: Spack

Spack



- (月) (コト (ヨ) ヨー の()

- manages multiple builds
- takes care of dependency relationships
- drives package-level build systems
- wrapper around built systems (cmake/autotools/make etc.)
- targeted towards users, admins and developers



Spack basics



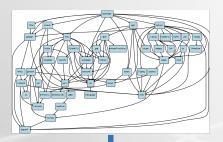
bunch of python scripts

- 98.2% according to github
- packages are maintained by the developers and the community
 - currently 5987 available packages
 - https://github.com/spack/spack
 - You can contribute!

Spack basics



- Dependency graphs are translated into a hash
 - unique identifier for each build (if all aspects of a build are identical - same hash)
- Spack uses RPATH and PATH to ensure dependencies are found





Spack on the SCC



ロン (月) (日) (日) 日 () ()

- most of the modules on the cluster are installed using Spack
- Spack itself is installed as a module spack-user
 - easily find installed software
 - reuse packages that is already installed
 - manage your own builds

How to use Spack?



- Load the spack-user module with module load spack-user
- Follow the instructions to activate the spack shell support
- Install supported software spack install SPEC
- Load the installed software spack load SPEC

SPEC syntax



SPECS specify the software configuration

- i.e. constraints YOU set for your installation
- optional: only specify what you need

\$ spack install mpileaks
\$ spack install mpileaks@3.3
\$ spack install mpileaks@3.3 %gcc@4.9.3
\$ spack install mpileaks@3.3 %gcc@4.9.3 +threads
\$ spack install mpileaks@3.3 cppflags="-O3 -g3"
\$ spack install mpileaks@3.3 target=cascadelake
\$ spack install mpileaks@3.3 ^mpich@3.2 %gcc@4.9.3

Unconstrained @ custom version % custom compiler +/-/~ build option Set compiler flags Set CPU architecture ^ dependency information

2 3 4

Example with nano



gadu102:61 23:23:18 -> spack Install nano () / opt/ss/rev[21.12/hasselL/gcc-93.36/kgconf-1.8.0-ktrb7r () / opt/ss/rev[21.12/hasselL/gcc-93.36/kgconf-1.8.0-ktrb7r () / opt/ss/rev[21.12/hasselL/gcc-93.36/kgconf-1.8.0-ktrb7r Installing nano-4.3-hyadazgaggd(tbtp4dmrgts3xg)rxky found: installing from source Using cached archive: / using visers/ttabudy.spack/0.17.1/cache/_source-cache/archive/0e/0e399729d105cb1a587b4140db5cf1b23215a0886a42b215efa98137164233 a6.tar.xz No patches needed for nano nano: Executing phase: / using cached archive: / using cache

- spack find -vdl nano
- spack install nano
- spack load nano
- nano –version

The Most Important Command



spack help

- overview over all spack commands
- spack help <command>
 - provides information about usage and available options for all spack commands
 - usage will help you to get familiar with spack

Basic Spack Commands



- (月) (コト (ヨ) ヨー の()

spack list

list of all available packages for spack
 spack find

list of all already installed packages
 spack compilers

list of all installed compilers

spack info

print information about a spack package

Let's install something!



Follow the Tutorial

- https://pad.gwdg.de/s/F8bHwyObF#
- Experiment with the various spack commands!
- Discuss with your group!
- Ask us if you've got any question or need assistance!
- Use our Spack cheat sheet!
 - https://pad.gwdg.de/s/QH2VaVqch##