HPS
https://hps.vi4io.org

Laura Endter

Spack

Managing HPC-Sofware made easy

Learning Objectives

- Understand the difficulties concerning software installations in HPC.
- Install packages on the SCC using spack-user.
- Formulate valid spack SPECS.
- Manage different software versions.

Introduction

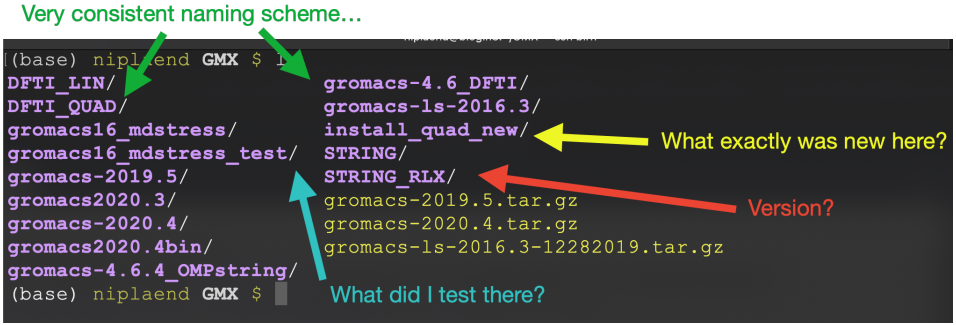


Figure: My personal gromacs mess.

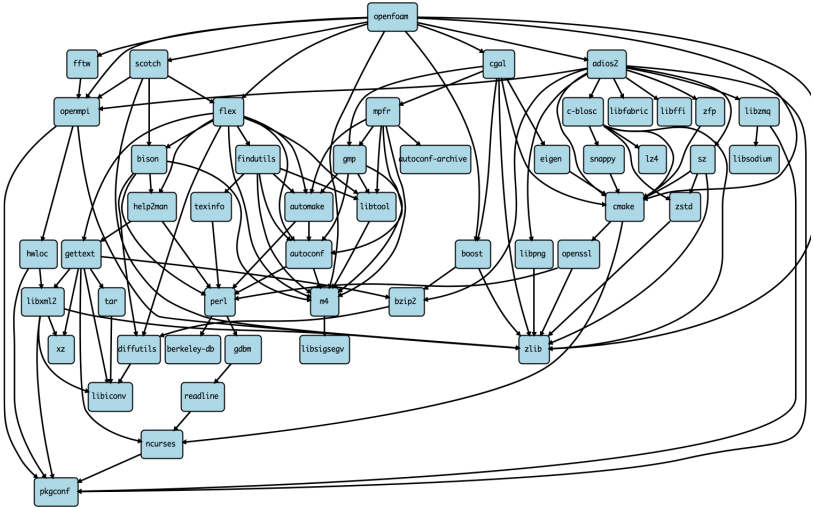
Installing software on a Cluster

- 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
- coexistence of several builds
- often many dependencies
- many compiling options
- users active on several clusters

Spack can help

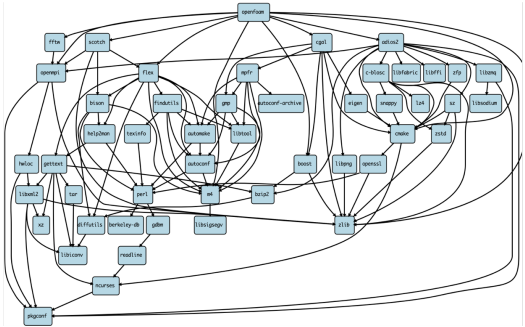
- manages multiple builds
- takes care of dependency relationships
- drives package-level build systems
- wrapper around built systems (cmake/autotools/make etc.)
- targeted towards users, admins and developers

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

- dependecy 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



#

openfoam-k5hyzc7e6meneq1dqfguw2nddghqfz52

Spack on the SCC

- 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

Note for DLR: This works the same on CARO.

Quick and Dirty Spack Installation

- Load the spack-user module with `module load spack-user`
- Follow the instructions to activate the spack shell support

```
> source $SPACK_USER_ROOT/share/spack/setup-env.sh
```

- 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`	Unconstrained
`$ spack install mpileaks@3.3`	@ custom version
`$ spack install mpileaks@3.3 %gcc@4.9.3`	% custom compiler
`$ spack install mpileaks@3.3 %gcc@4.9.3 +threads`	+/-/~ build option
`$ spack install mpileaks@3.3 cppflags="-O3 -g3"`	Set compiler flags
`$ spack install mpileaks@3.3 target=cascadelake`	Set CPU architecture
`$ spack install mpileaks@3.3 ^mpich@3.2 %gcc@4.9.3`	^ dependency information

Installation example

```
[gwdu101:8 14:13:50 ~ > spack install ffmpeg@4.4
==> Warning: Missing a source id for ffmpeg@4.4
[+] /opt/sw/rev/21.12/haswell/gcc-9.3.0/alsa-lib-1.2.3.2-bjqgyy
[+] /opt/sw/rev/21.12/haswell/gcc-9.3.0/libiconv-1.16-6npizd
[+] /opt/sw/rev/21.12/haswell/gcc-9.3.0/yasm-1.3.0-23xln7
[+] /opt/sw/rev/21.12/haswell/gcc-9.3.0/zlib-1.2.11-wwxahg
[+] /opt/sw/rev/21.12/haswell/gcc-9.3.0/diffutils-3.8-7usaan
[+] /opt/sw/rev/21.12/haswell/gcc-9.3.0/bzip2-1.0.8-l4ilvd
==> Installing ffmpeg-4.4-k5hyzc7e6meneqldqfguw2nddghqfz52
==> No binary for ffmpeg-4.4-k5hyzc7e6meneqldqfguw2nddghqfz52 found: installing from source
==> Warning: There is no checksum on file to fetch ffmpeg@4.4 safely.
==>   Fetch anyway? [y/N] y
==> Fetching https://ffmpeg.org/releases/ffmpeg-4.4.tar.bz2
==> No patches needed for ffmpeg
==> ffmpeg: Executing phase: 'autoreconf'
==> ffmpeg: Executing phase: 'configure'
==> ffmpeg: Executing phase: 'build'
==> ffmpeg: Executing phase: 'install'
==> ffmpeg: Successfully installed ffmpeg-4.4-k5hyzc7e6meneqldqfguw2nddghqfz52
  Fetch: 1.48s.  Build: 2m 13.17s.  Total: 2m 14.65s.
[+] /usr/users/lendter/.spack/0.17.1/install/haswell/gcc-9.3.0/ffmpeg-4.4-k5hyzc
```

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

`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#