# Package Management With `Spack`

Daniel Topa
HII-TSD
daniel.topa@hii-tsd.com

October 2, 2024

**Abstract**

The package `spack` is a widely used and modern package management toolset born on the HPC and now exploited for personal computation. By design, `spack` allows user to unite environments under a compiler, a Python version, an MPI instance and manage the many variants. A simple example is provided which demonstrates how quickly the application can be downloaded and used. Next is a discussion of what `spack` does and who is using it. We conclude with links to articles and briefings which may be of interest to the new user.

## 1 Getting Started

### 1.1 Quick Example: `hwloc`

Consider an example build of the package, `hwloc`.

> The Hardware Locality (hwloc) software project. The Portable Hardware Locality (hwloc) software package provides a portable abstraction (across OS, versions, architectures, ...) of the hierarchical topology of modern architectures, including NUMA memory nodes, sockets, shared caches, cores and simultaneous multithreading. It also gathers various system attributes such as cache and memory information as well as the locality of I/O devices such as network interfaces, InfiniBand HCAs or GPUs. It primarily aims at helping applications with gathering information about modern computing hardware so as to exploit it accordingly and efficiently.

#### 1.1.1 Basic Steps

1. download spack
2. initialize spack
3. install hwloc

#### 1.1.2 Command Line Steps and Result

```
$ git clone https://github.com/spack/spack.git
$ source spack/share/spack/setup-env.sh
$ spack install hwloc
```

## 1.2 Install **spack**, build **hwloc**

```
dantopa@Xiuhcoatl-8.local:example $ git clone https://github.com/spack/spack.git
Cloning into 'spack'...
remote: Enumerating objects: 582107, done.
remote: Counting objects: 100% (1607/1607), done.
remote: Compressing objects: 100% (799/799), done.
remote: Total 582107 (delta 772), reused 1273 (delta 547), pack-reused 580500 (from 1)
Receiving objects: 100% (582107/582107), 197.03 MiB | 35.67 MiB/s, done.
Resolving deltas: 100% (273672/273672), done.
Updating files: 100% (11933/11933), done.

dantopa@Xiuhcoatl-8.local:example $ spack install hwloc
==> Installing gmake-4.4.1-ietaaa3kpwrzml6fhorys6hakqmisyf4 [1/8]
==> No binary for gmake-4.4.1-ietaaa3kpwrzml6fhorys6hakqmisyf4 found: installing from source
==> Fetching https://mirror.spack.io/_source-cache/archive/dd/dd16fb1d67bfab79a72f5e8390735c49e3e8e70b4945a15ab1f81ddb
==> No patches needed for gmake
==> gmake: Executing phase: 'install'
==> gmake: Successfully installed  gmake-4.4.1-ietaaa3kpwrzml6fhorys6hakqmisyf4
  Stage: 0.65s.  Install: 36.72s.  Post-install: 0.06s.  Total: 37.56s
[+] /Volumes/spacktivity/example/spack/opt/spack/darwin-sonoma-skylake/apple-clang-16.0.0/gmake-4.4.1-ietaaa3kpwrzml6f
==> Installing xz-5.4.6-hjg33x3qi6bqecwmlghxfezuddtwcjhw [2/8]
==> No binary for xz-5.4.6-hjg33x3qi6bqecwmlghxfezuddtwcjhw found: installing from source
==> Fetching https://mirror.spack.io/_source-cache/archive/91/913851b274e8e1d31781ec949f1c23e8dbcf0ecf6e73a2436dc21769
==> No patches needed for xz
==> xz: Executing phase: 'autoreconf'
==> xz: Executing phase: 'configure'
==> xz: Executing phase: 'build'
==> xz: Executing phase: 'install'
==> xz: Successfully installed xz-5.4.6-hjg33x3qi6bqecwmlghxfezuddtwcjhw
  Stage: 0.78s.  Autoreconf: 0.00s.  Configure: 28.71s.  Build: 12.71s.  Install: 3.55s.  Post-install: 0.27s.  Total:
[+] /Volumes/spacktivity/example/spack/opt/spack/darwin-sonoma-skylake/apple-clang-16.0.0/xz-5.4.6-hjg33x3qi6bqecwmlgh
==> Installing libiconv-1.17-oo6aigel5hcpcpfcvzlmit5mvbkzrrss [3/8]
==> No binary for libiconv-1.17-oo6aigel5hcpcpfcvzlmit5mvbkzrrss found: installing from source
==> Fetching https://mirror.spack.io/_source-cache/archive/8f/8f74213b56238c85a50a5329f77e06198771e70dd9a739779f4c02f6
==> No patches needed for libiconv
==> libiconv: Executing phase: 'autoreconf'
==> libiconv: Executing phase: 'configure'
==> libiconv: Executing phase: 'build'
==> libiconv: Executing phase: 'install'
==> libiconv: Successfully installed libiconv-1.17-oo6aigel5hcpcpfcvzlmit5mvbkzrrss
  Stage: 0.96s.  Autoreconf: 0.00s.  Configure: 54.41s.  Build: 11.15s.  Install: 1.82s.  Post-install: 0.22s.  Total:
[+] /Volumes/spacktivity/example/spack/opt/spack/darwin-sonoma-skylake/apple-clang-16.0.0/libiconv-1.17-oo6aigel5hcpcp
==> Installing zlib-ng-2.2.1-rjskn465o44z4n6q24dksiby2pd5lpm3 [4/8]
==> No binary for zlib-ng-2.2.1-rjskn465o44z4n6q24dksiby2pd5lpm3 found: installing from source
==> Fetching https://mirror.spack.io/_source-cache/archive/ec/ec6a76169d4214e2e8b737e0850ba4acb806c69eeace6240ed4481b9
==> No patches needed for zlib-ng
==> zlib-ng: Executing phase: 'autoreconf'
==> zlib-ng: Executing phase: 'configure'
==> zlib-ng: Executing phase: 'build'
==> zlib-ng: Executing phase: 'install'
==> zlib-ng: Successfully installed  zlib-ng-2.2.1-rjskn465o44z4n6q24dksiby2pd5lpm3
  Stage: 0.94s.  Autoreconf: 0.00s.  Configure: 10.03s.  Build: 6.46s.  Install: 0.37s.  Post-install: 0.06s.  Total:
[+] /Volumes/spacktivity/example/spack/opt/spack/darwin-sonoma-skylake/apple-clang-16.0.0/zlib-ng-2.2.1-rjskn465o44z4n
==> Installing pkgconf-2.2.0-7pmnvez4bcl4ydiuih3syxr4w6jlful6 [5/8]
==> No binary for pkgconf-2.2.0-7pmnvez4bcl4ydiuih3syxr4w6jlful6 found: installing from source
==> Fetching https://mirror.spack.io/_source-cache/archive/b0/b06ff63a83536aa8c2f6422fa80ad45e4833f590266feb14eaddfe1d
==> No patches needed for pkgconf
==> pkgconf: Executing phase: 'autoreconf'
==> pkgconf: Executing phase: 'configure'
==> pkgconf: Executing phase: 'build'
==> pkgconf: Executing phase: 'install'
==> pkgconf: Successfully installed  pkgconf-2.2.0-7pmnvez4bcl4ydiuih3syxr4w6jlful6
  Stage: 0.73s.  Autoreconf: 0.00s.  Configure: 11.10s.  Build: 2.28s.  Install: 0.64s.  Post-install: 0.06s.  Total:
[+] /Volumes/spacktivity/example/spack/opt/spack/darwin-sonoma-skylake/apple-clang-16.0.0/pkgconf-2.2.0-7pmnvez4bcl4yd
==> Installing libxml2-2.10.3-as2t7b3gziclpsms3fge2vyyhg7gwl5r [6/8]
==> No binary for libxml2-2.10.3-as2t7b3gziclpsms3fge2vyyhg7gwl5r found: installing from source
==> Fetching https://mirror.spack.io/_source-cache/archive/5d/5d2cc3d78bec3dbe212a9d7fa629ada25a7da928af432c93060ff5c1
==> Fetching https://mirror.spack.io/_source-cache/archive/96/96151685cec997e1f9f3387e3626d61e6284d4d6e66e0e440c209286
```

```
==> Moving resource stage
        source: /var/folders/f2/0qk5gn4x1rlczv63skzbp19h0000gn/T/dantopa/spack-stage/resource-xmlts-as2t7b3gziclpsms3f
        destination: /var/folders/f2/0qk5gn4x1rlczv63skzbp19h0000gn/T/dantopa/spack-stage/spack-stage-libxml2-2.10.3-a
==> Ran patch() for libxml2
==> libxml2: Executing phase: 'autoreconf'
==> libxml2: Executing phase: 'configure'
==> libxml2: Executing phase: 'build'
==> libxml2: Executing phase: 'install'
==> libxml2: Successfully installed libxml2-2.10.3-as2t7b3gziclpsms3fge2vyyhg7gwl5r
  Stage: 4.56s.  Autoreconf: 0.00s.  Configure: 18.28s.  Build: 12.18s.  Install: 1.78s.  Post-install: 0.12s.  Total:
[+] /Volumes/spacktivity/example/spack/opt/spack/darwin-sonoma-skylake/apple-clang-16.0.0/libxml2-2.10.3-as2t7b3gziclp
==> Installing ncurses-6.5-y4puwqifh7lcfoyme4xerqpyhy6wk5dd [7/8]
==> No binary for ncurses-6.5-y4puwqifh7lcfoyme4xerqpyhy6wk5dd found: installing from source
==> Fetching https://mirror.spack.io/_source-cache/archive/13/136d91bc269a9a5785e5f9e980bc76ab57428f604ce3e5a5a90cebc7
==> Applied patch /Volumes/spacktivity/example/spack/var/spack/repos/builtin/packages/ncurses/rxvt_unicode_6_4.patch
==> ncurses: Executing phase: 'autoreconf'
==> ncurses: Executing phase: 'configure'
==> ncurses: Executing phase: 'build'
==> ncurses: Executing phase: 'install'
==> ncurses: Successfully installed ncurses-6.5-y4puwqifh7lcfoyme4xerqpyhy6wk5dd
  Stage: 0.83s.  Autoreconf: 0.00s.  Configure: 1m 29.88s.  Build: 50.04s.  Install: 19.11s.  Post-install: 2.56s.  To
[+] /Volumes/spacktivity/example/spack/opt/spack/darwin-sonoma-skylake/apple-clang-16.0.0/ncurses-6.5-y4puwqifh7lcfoym
==> Installing hwloc-2.11.1-mfauw6yq45zhpldzh7ot5ns6tiisx4x2 [8/8]
==> No binary for hwloc-2.11.1-mfauw6yq45zhpldzh7ot5ns6tiisx4x2 found: installing from source
==> Fetching https://mirror.spack.io/_source-cache/archive/9f/9f320925cfd0daeaf3a3d724c93e127ecac63750c623654dca029850
==> No patches needed for hwloc
==> hwloc: Executing phase: 'autoreconf'
==> hwloc: Executing phase: 'configure'
==> hwloc: Executing phase: 'build'
==> hwloc: Executing phase: 'install'
==> hwloc: Successfully installed hwloc-2.11.1-mfauw6yq45zhpldzh7ot5ns6tiisx4x2
  Stage: 1.23s.  Autoreconf: 0.00s.  Configure: 43.66s.  Build: 7.48s.  Install: 2.53s.  Post-install: 0.26s.  Total:
[+] /Volumes/spacktivity/example/spack/opt/spack/darwin-sonoma-skylake/apple-clang-16.0.0/hwloc-2.11.1-mfauw6yq45zhpld
```

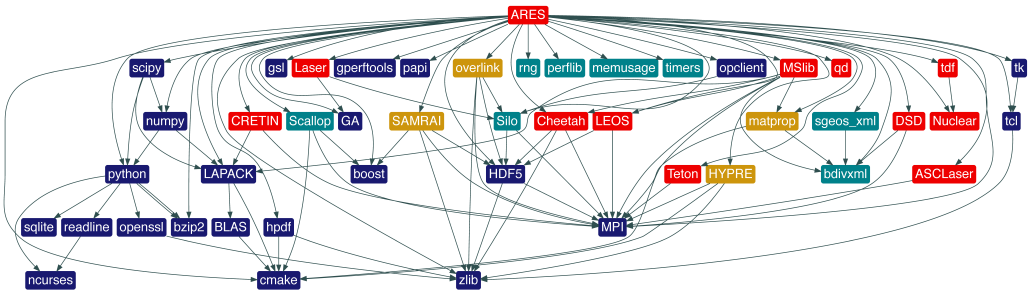


Figure 1: Sample dependency tree managed by `spack`.

## 1.3 Hardware Locality

The hardware locality application `hwloc` provides insight into the hardware configuration of the host machine. An example using `lstopo` is shown in figure 2.

## 1.4 How Does `spack` Work?

`spack` is a *download*, not an *installation*. It was born at Livermore out a desire to empower scientists to build their own software stacks. What started as a tool for people with local admin

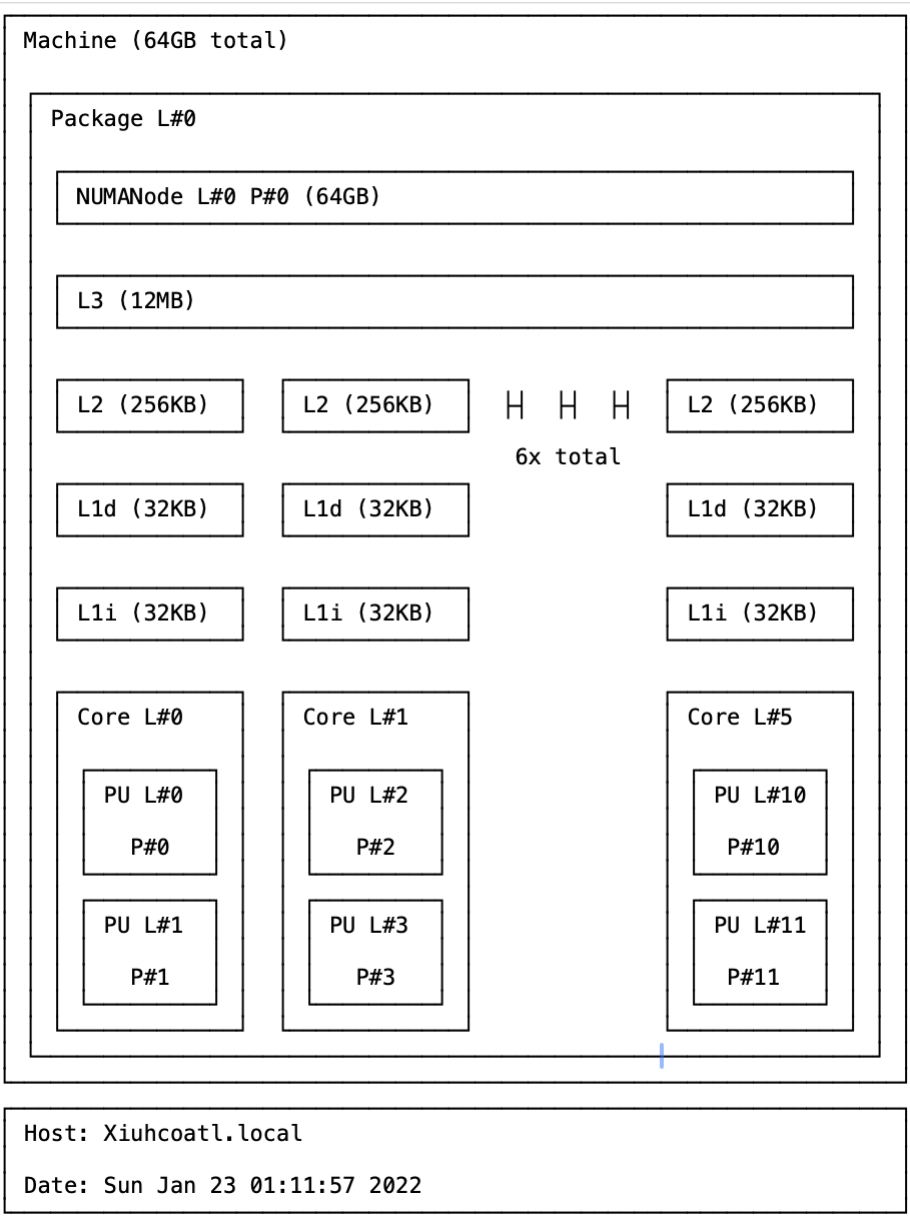


Figure 2: The application `hwloc` contains the utility `lstopo` which identifies the hardware configuration.

privileges over their machines is now a recognized tool used by the HPC support staffs world wide.

`spack` changes how developers interact with their uses. Instead of maintaining pages detailing install instructions for each hardware architecture and software environment, developers now maintain a single `spack` instance and utilize the issue tracking inherent in `GitHub`.

Whether the build system is autotools, make, cmake, ninja, etc., `spack` automate the process. A critical property is that `spack` build package creators use a standardized template which causes uniform performance of the builds

In essence, `spack` is a database managing dependencies, variants, and locations. Below is a sample tree diagram for a Livermore hydrocode showing the complexity managed by `spack`.

`spack` handles combinatorial complexity. For example, consider 4 compilers: Intel, GCC, PGI, NAG. For each compiler maintain 4 different versions; for example gcc 14.2.0, 13.3.0, 12.4.0, 4.8.5. Provide 4 MPI providers: OpenMPI, Cray-MPICH, MVAPICH, Intel-Parallel studio. Maintain 4 versions of each of those. Maintain 4 Python versions for each packages. This represents $4^5 = 1024$ instances, handled by `spack`.

### 1.5 `spack` Users, Platforms

`spack` is used extensively across HPC platforms and personal computing platforms, many of which are sampled below.

1. Windows 11, MacOS, ARM, Power8, Power9, x86-64, BlueGene
2. DOD HPCMP
3. Lawrence Livermore National Laboratory
4. Los Alamos National Laboratory
5. Oak Ridge National Laboratory
6. Argonne  National Laboratory
7. Intel
8. NCAR
9. CERN
10. Iowa State HPC
11. University of Wisconsin–Madison
12. UConn Storrs HPC
13. University of Michigan
14. NM State
15. Lehigh
16. Amazon Web Services
17. Azure

## 2 Probe commands in `spack`

There are many probe and diagnostic commands which help the builder understand the process and products. Two such commands are shown below.

## 2.1 Graph dependencies

```
$ spack graph openmpi
kpex76l openmpi@1.10.7%gcc
vuijyrm     hwloc@1.11.13%gcc
vlgsd6a         libxml2@2.10.3%gcc
7ffbqyf             libiconv@1.17%gcc
cejtv5p             pkgconf@1.8.0%gcc
ydjmqn5             xz@5.4.1%gcc
kgdj2w7         ncurses@6.4%gcc
cbup2u4     openssh@9.1p1%gcc
74ofkad         krb5@1.20.1%gcc
gw3muwr             bison@3.8.2%gcc
mbfdcbq                 m4@1.4.19%gcc
ytuafo5                     libsigsegv@2.13%gcc
fx3kvo3             diffutils@3.8%gcc
g7g5rxm             gettext@0.21.1%gcc
pirykzv                 bzip2@1.0.8%gcc
lij4icg                 tar@1.34%gcc
3tfa2za                     pigz@2.7%gcc
hnuj2am                     zstd@1.5.2%gcc
mf4yylc         libedit@3.1-20210216%gcc
pnhvhts         libxcrypt@4.4.33%gcc
cck5u3i             perl@5.34.0%gcc
duhpddy         openssl@1.1.1t%gcc
syyclam             ca-certificates-mozilla@2023-01-10%gcc
ggaig6s         zlib@1.2.13%gcc
lxwy7gr     perl@5.34.0%gcc
kzdyfxk     pkgconf@1.8.0%gcc
```

## 2.2 `spack info petsc`

The `spack` command `info` presents essential information about each package in the following order.

1. Dependencies
2. Homepage
3. Versions
4. Variants
   (a) build
   (b) link
   (c) run
5. License

The output starts with a brief description of the package and web site providing more information and a listing of available versions. Next is a list of variants and how to invoke them showing the user how to construct specific versions of the package – which will all be managed by `spack`. Users can easily specify whether to use `C` or `C++` for the build, whether to use double or single precision, whether to use `MPI`[1], whether to use OpenMP, and so on. The

[1] `spack` allows users to chose between many flavors of `MPI`

final sections lists dependencies for building, linking, and running. `spack` will build these as needed.

```
$ spack info petsc
Package:   petsc

Description:
    PETSc is a suite of data structures and routines for the scalable
    (parallel) solution of scientific applications modeled by partial
    differential equations.

Homepage: https://petsc.org

Preferred version:
    3.22.0    http://web.cels.anl.gov/projects/petsc/.../petsc-3.22.0.tar.gz

Safe versions:
    main      [git] https://gitlab.com/petsc/petsc.git on branch main
    3.22.0    http://web.cels.anl.gov/projects/petsc/.../petsc-3.22.0.tar.gz
    3.21.6    http://web.cels.anl.gov/projects/petsc/.../petsc-3.21.6.tar.gz

    3.11.1    http://web.cels.anl.gov/projects/petsc/.../petsc-3.11.0.tar.gz

Deprecated versions:
    None

Variants:
    X [false]                    false, true
        Activate X support
    batch [false]                false, true
        Enable when mpiexec is not available to run binaries
    build_system [generic]       generic
        Build systems supported by the package
    cgns [false]                 false, true
        Activates support for CGNS (only parallel)
    clanguage [C]                C, C++
        Specify C (recommended) or C++ to compile PETSc
    complex [false]              false, true
        Build with complex numbers
    cuda [false]                 false, true
        Build with CUDA
    debug [false]                false, true
        Compile in debug mode
    double [true]                false, true
        Switches between single and double precision
    exodusii [false]             false, true
        Activates support for ExodusII (only parallel)
    fftw [false]                 false, true
        Activates support for FFTW (only parallel)
    fortran [true]               false, true
        Activates fortran support
```

```
giflib [false]                false, true
    Activates support for GIF
hdf5 [true]                   false, true
    Activates support for HDF5 (only parallel)
hpddm [false]                 false, true
    Activates support for HPDDM (only parallel)
hwloc [false]                 false, true
    Activates support for hwloc
hypre [true]                  false, true
    Activates support for Hypre (only parallel)
int64 [false]                 false, true
    Compile with 64bit indices
jpeg [false]                  false, true
    Activates support for JPEG
knl [false]                   false, true
    Build for KNL
kokkos [false]                false, true
    Activates support for kokkos and kokkos-kernels
libpng [false]                false, true
    Activates support for PNG
libyaml [false]               false, true
    Activates support for YAML
memalign [none]               none, 16, 32, 4, 64, 8
    Specify alignment of allocated arrays
memkind [false]               false, true
    Activates support for Memkind
metis [true]                  false, true
    Activates support for metis and parmetis
mkl-pardiso [false]           false, true
    Activates support for MKL Pardiso
mmg [false]                   false, true
    Activates support for MMG
moab [false]                  false, true
    Acivates support for MOAB (only parallel)
mpfr [false]                  false, true
    Activates support for MPFR
mpi [true]                    false, true
    Activates MPI support
mumps [false]                 false, true
    Activates support for MUMPS (only parallel)
openmp [false]                false, true
    Activates support for openmp
p4est [false]                 false, true
    Activates support for P4Est (only parallel)
parmmg [false]                false, true
    Activates support for ParMMG (only parallel)
ptscotch [false]              false, true
    Activates support for PTScotch (only parallel)
random123 [false]             false, true
    Activates support for Random123
```

```
    rocm [false]                  false, true
        Enable ROCm support
    saws [false]                  false, true
        Activates support for Saws
    shared [true]                 false, true
        Enables the build of shared libraries
    strumpack [false]             false, true
        Activates support for Strumpack
    suite-sparse [false]          false, true
        Activates support for SuiteSparse
    sycl [false]                  false, true
        Enable sycl build
    tetgen [false]                false, true
        Activates support for Tetgen
    trilinos [false]              false, true
        Activates support for Trilinos (only parallel)
    valgrind [false]              false, true
        Enable Valgrind Client Request mechanism
    zoltan [false]                false, true
        Activates support for Zoltan

    when +rocm
      amdgpu_target [none]        none, gfx1010, gfx1011, gfx1012, gfx1013,
          gfx1030, gfx1031, gfx1032, gfx1033, gfx1034, gfx1035, gfx1036,
          gfx1100, gfx1101, gfx1102, gfx1103, gfx701, gfx801, gfx802, gfx803,
          gfx900, gfx900:xnack-, gfx902, gfx904, gfx906, gfx906:xnack-,
          gfx908, gfx908:xnack-, gfx909, gfx90a, gfx90a:xnack+, gfx90a:xnack-,
          gfx90c, gfx940, gfx941, gfx942
          AMD GPU architecture

    when +cuda
      cuda_arch [none]            none, 10, 11, 12, 13, 20, 21, 30, 32, 35,
          37, 50, 52, 53, 60, 61, 62, 70, 72, 75, 80, 86, 87, 89, 90, 90a
          CUDA architecture

    when +fortran
      scalapack [false]           false, true
          Activates support for Scalapack
      superlu-dist [true]         false, true
          Activates support for superlu-dist (only parallel)

Build Dependencies:
    blas  cuda       exodusii  giflib  gmp   hip      hipsolver  hsa-rocr-dev
       hypre  kokkos          lapack  libx11   llvm-amdgpu  metis  mmg   mpfr
        mumps     p4est            parmetis  python     rocblas    rocprim
       rocsolver  rocthrust  scalapack  sowing     suite-sparse  tetgen
       valgrind  zoltan
    cgns  diffutils  fftw      gmake   hdf5  hipblas  hipsparse  hwloc
       jpeg   kokkos-kernels  libpng  libyaml  memkind      mkl    moab  mpi
         netcdf-c  parallel-netcdf  parmmg    random123  rocm-core  rocrand
```

```
        rocsparse  saws       scotch     strumpack  superlu-dist  trilinos
        zlib-api

Link Dependencies:
    blas  cuda      fftw    gmake  hdf5  hipblas    hipsparse     hwloc  jpeg
           kokkos-kernels  libpng  libyaml      memkind  mkl  moab  mpi
        netcdf-c  parallel-netcdf  parmmg     rocblas    rocprim  rocsolver
        rocthrust  scalapack  sowing     suite-sparse  tetgen    valgrind
        zoltan
    cgns  exodusii  giflib  gmp    hip   hipsolver  hsa-rocr-dev  hypre
        kokkos  lapack          libx11  llvm-amdgpu  metis    mmg  mpfr  mumps
         p4est     parmetis         random123  rocm-core  rocrand  rocsparse
        saws       scotch     strumpack  superlu-dist  trilinos  zlib-api

Run Dependencies:
    None

Licenses:
    None
```

# 3 Learn More About Spack

## 3.1 Spack Awards

At the latest 2023 International Conference for High Performance Computing, Networking, Storage and Analysis `spack` recognized as the Best High Performance Computing (HPC) Programming Tool or Technology: Spack receives prestigious HPCwire award at SC23

## 3.2 Articles

1. Spack: A Flexible Package Manager for HPC Software
2. Mapping Out the HPC Dependency Chaos
3. HPX with Spack and Singularity Containers: Evaluating Overheads for HPX/Kokkos using an astrophysics application

## 3.3 Spack Documentation

1. The Spack package manager: bringing order to HPC software chaos
2. Overview
3. Getting Started
4. Basic Installation
5. Basic Usage
6. Tutorial
7. Packaging Guide
8. Documentation Home
9. GitHub Repo