Spack - a package manager for HPC systems

Moritz Siegel

Why Package Manager?

Many installations of the **same** software but with different:

- Hardware
- Versions
- Compilers
- Flags
- Dependencies

Example MPI:

18 providers **x** 30 versions **x** 6 compilers **x** 22 flags **x** 31 dependencies =

= **2209680 variants** on VSC-5 alone!

Architecture: Spack Trees

We have 3 different **spack trees** for different architecture:

Cluster	CPU	GPU	Spack Tree
VSC-4	Intel	NO	**`skylake`**
VSC-5	AMD	NO	**`zen`**
VSC-5	AMD	YES	**`cuda-zen`**

All packages **only** work on their architecture!

Default Spack Trees

The current spack tree is shown on the left of the command line **prompt**:

```
skylake trainee00@l44:~$
zen trainee00@l55:~$
cuda-zen trainee00@l55:~$
```

On login the **spack tree** for the current architecture is set.

Switch Tree

All spack commands work on the **current tree**.

Type **`sky`**, **`zen`** or **`cuz`** to **switch** to a spack tree:

```
skylake trainee00@144:~$ zen
zen trainee00@144:~$ cuz
cuda-zen trainee00@144:~$ sky
skylake trainee00@144:~$
```

You are on the **same node `144`** the whole time.

Find Your Package

Find your package, like **openmpi**:

```
zen trainee00@144:~$ spack find openmpi
```

Use **"@"** to show specific **version** only:

```
skylake trainee00@144:~$ spack find openmpi@4.1.4
zen trainee00@155:~$ spack find openmpi@3
```

All **python** packages have a leading *"py-"* like in **py-numpy**.

All **R** packages start with *"r-"* like in **r-brew**.

Hash

Add **-l** to show the unique package **hash** too:

```
skylake trainee00@l44:~$ spack find -l openmpi@4.1.4
```

Use **any** spack command with this hash, and a **"/"** in front:

```
skylake trainee00@l44:~$ spack find -l /2vqdnay
```

Old Hashes

Use **spack search** to search for a **hash** of an old VSC installation:

```
zen trainee00@l55:~$ spack search /asdc2mk
```

You get some similar modules on the **current** spack tree.

Compiler

Use **"%"** to only show packages **compiled** with **intel**, **gcc**, **aocc**, etc:

```
skylake trainee00@144:~$ spack find openmpi %intel
zen trainee00@155:~$ spack find openmpi %aocc
cuda-zen trainee00@155:~$ spack find openmpi %gcc
```

Combine **"%"** and **"@"** to show specific **compiler version** only:

```
zen trainee00@155:~$ spack find openmpi %gcc@9
```

Flags

Use **`spack find -v`** to see the **flags** of a package:

```
skylake trainee00@l44:~$ spack find -v openmpi@4.1.4
```

Use **`"+"`** or **`"~"`** to show only packages that **have/not have** that flag:

```
cuda-zen trainee00@l55:~$ spack find openmpi +cuda
```

Dependencies

Use **spack find -d** to show the **dependencies** of a package:

```
skylake trainee00@144:~$ spack find -d /xt53foa
```

Use **"^"** to only show packages **A** that depend on **B**:

```
skylake trainee00@144:~$ spack find -d py-numpy ^intel-oneapi-mkl
zen trainee00@155:~$ spack find -d  py-numpy ^openblas
```

Load Your Package

Use **`spack load`** to load your package:

```
skylake trainee00@144:~$ spack load openmpi@4.1.4
```

Add the unique **hash** to load that **exact** package:

```
skylake trainee00@144:~$ spack load py-numpy/ygplffi
zen trainee00@155:~$ spack load py-numpy/dzcpm7y
cuda-zen trainee00@155:~$ spack load py-numpy/bctcmkr
```

All these packages are **numpy**, but built for different architectures.

List & Unload

Type **`spack find --loaded`** to list loaded packages:

```
skylake trainee00@144:~$ spack find --loaded
-- linux-almalinux8-zen3 / gcc@12.2.0 ---------------------------
o6ui7qb berkeley-db@18.1.40          4o2lzl4 libbsd@0.11.5
qta2z6t openssl@1.1.1s               sultos7 py-setuptools@63.0.0
...
```

Type **`spack unload`** to unload all packages.

Spack Modules

Every package installed with spack creates a module file:

```
<package>/<version>-<compiler>-<version>-<hash>
openmpi/3.1.6-aocc-4.0.0-k4glowk
openmpi/4.1.4-gcc-12.2.0-je3hmfk
openmpi/4.1.4-intel-2021.7.1-lex74to
```

The **hash** at the end, like `k4glowk` is the same as in spack.

Exercises

□ login to VSC-4/5.

□ switch between the three spack trees **zen**, **skylake** and **cuda-zen**.

□ find a **hdf5** variant compiled with `aocc` at **zen**.

□ find a **openmpi** with `cuda` support at **cuda-zen**.

□ find a **numpy** including `intel-oneapi-mkl` at **skylake**.

□ load any **openmpi** package, then list all the loaded packages.

□ find out what old package once was `/asdc2mk`.