IceCube CVMFS Software and Spack

David Schultz - WIPAC, UW-Madison
Benedikt Riedel - University of Chicago
Outline

About IceCube

Our usage of Spack

Experiences
The IceCube Neutrino Observatory

Design and construction

Detector Design
- 1 gigaton of instrumented ice
- 5,160 light sensors, or digital optical modules (DOMs), digitize and time-stamp signals
- 1 square kilometer surface array, IceTop, with 324 DOMs
- 2 nanosecond time resolution

Detector Construction
- 7 seasons of construction, 2004-2011
- 28,000 person-days to complete construction, or 77 years of continuous work
- 4.7 million pounds of cargo shipped, 1.2 million of which was the drill
- 48 hours to drill and 11 hours to deploy sensors per hole
- 4.7 megawatts of drill thermal power with 200 gallons of water per minute delivered at 98 °C and 1,000 psi

IceCube Lab
Amundsen-Scott South Pole Station
IceTop
Enhanced Hot Water Drill
firm drill

2450 m
1450 m
50 m

Antarctic bedrock

86 strings of DOMs, set 125 meters apart
60 DOMs on each string
DOMs are 17 meters apart

Explorer
IceCube Requirements

Common software base for collaboration

- ~35 common software packages
- ~25 common python extensions
- ~5 custom software packages
- Single view of all packages
Advantages of Spack

- Don’t have to maintain install scripts for common software - shared by community
- gcc compiler package available
  -- requested by many users
- Automatically handles multi-platform installs
- Packages written in Python, our std language
How IceCube uses Spack

Build script:

1. Install spack and add private repo
2. `spack install` a compiler
3. `spack install` for each package
4. `spack view` symlinking all packages
5. Repeat 2-4 on each OS
Problems encountered

Figuring out dependencies

- Make sure packages use the exact version of a dependency we build

openblas@0.3.0 cpu_target=SSE_GENERIC
virtual_machine=True

suite-sparse@5.2.0 ^openblas@0.3.0 cpu_target=SSE_GENERIC
virtual_machine=True

- `spack spec` concretization shows what to expect
Problems encountered

Packages that need modification

- Lack of options: Root, many others
  -- We need a non-X11 build
  -- [https://github.com/spack/spack/pull/8428](https://github.com/spack/spack/pull/8428)

- Build errors: gnuplot, glib, …
  -- Was not happy with python build scripts
  -- Modified package to not build docs or tests
Problems encountered

Unexpected happenings

- Package hashes change when updated
  - *git pull* on spack was a bad idea - forced rebuild of a large number of packages
  - This created a failure in dependency resolution (two packages with the same specs but different hashes), forcing a delete and start over approach
Good things

https://github.com/spack/spack/pull/7152
(merged 2 weeks ago)

- Fixes a bug with views and python packages
  - Previously: needed to globally activate package, or set PYTHONPATH
  - Now: adding python package to view “just works”
Wishlist

- External config files:
  -- https://github.com/spack/spack/pull/6903

- Easier setup for external packages
  -- Maybe a command to auto-register a package?

- Not having to specify the full spec of a dependency if it’s already installed
Conclusions

IceCube is now using Spack for common software

Spack had some unexpected difficulties, but nothing that could not be worked around

Overall, happy to get majority of package installs from upstream / others