HSF Packaging Working Group Update

Graeme A Stewart

HOW2019 Workshop at JLab, 2019-03-21
HSF Packaging Group

- **Raison d’être:**
  - ...common issues, tools, and approaches to building and distributing the software stacks used by HEP experiments - these are big stacks, e.g. 400 packages in LCG release used by ATLAS, LHCb, SWAN
  - Continues to be a very active HSF group

- **Published an HSF Technical Note** looking at the main build tools
  - Community and FOSS - main advantages and disadvantages summarised
  - Still valid, but tools evolve and change over time

- **Last year...**
  - We worked on a Use Cases document that elaborates the main requirements and desiderata for packaging in the HEP community
    - From librarians to developers
  - We also worked on some Test Drive instructions
    - Allowing people to rapidly test some of the different packaging solutions we know
  - Presentation at CHEP2018 summarising our work
TODOs for 2019

- Get main prototypes into a state where they solve most of the problems at a production level
  - “Continuous Delivery” system in place (build, make binary package, deploy to CVMFS, deploy to container)
    - Recall that CVMFS and containers are key features of the landscape now
  - We want these to aim at providing meaningful and testable solutions to the packaging problem for HEP
  - We have rolling updates and use the packaging group lists for discussion and sharing ideas and solutions

- Longer term goals and thoughts
  - CHEP2019 paper
  - Update of the packaging group technical note
CONDA

- Very popular package system
  - Language agnostic, Multi-platform, Multi-architecture, Binary installs
  - Data science and machine learning community use this extensively

- Deep rewrite in the last few years
  - Now supports multiple environments
    - These can be enabled and disabled easily, independent but share common components if they exist
    - Builds quite deeply, having moved away from system libraries for internal consistency

- Official Anaconda packages (1400) supplemented by Conda Forge community support for 6000 more packages
  - Other channels: BioConda, AstroConda

- Conda package recipes easy to write in YAML

- Addition of ROOT packaging in Conda Forge (Chris Burr, Henry Schreiner) makes it very attractive for HEP analysis users

- Not clear it can so easily deliver full suite of production stacks to experiments
Nix

- Package manager based around deep rebuilds and a functional description of packages and their dependencies
  - Excels at build consistency and reproducibility
  - 40k Nixpkgs
- Support for multiple environments, with shared components
- Chris Burr was at NixCon in London, good survey of Nix community
  - Quite a popular solution in HPC world
- LHCb have a prototype with a significant number of projects building with Nix
- Main drawbacks for HEP are
  - Fixed build locations (not relocatable)
    - This is rather awkward for deployment (cvmfs) and development; not great for worldwide collaborations
  - Quite particular functional DSL for package descriptions
- Compute Canada do use Nix to build a base on which non-Nix development rests
Spack

- Package manager developed at LLNL, initially for HPCs
  - Builds rather deeply, but can be instructed to take system packages
  - Supports caches and relocation of binary artefacts
  - Strong support for scientific computing and from different science communities
  - Package descriptions written in Python
- We have had good contact with the development team for the last few years
  - FCC stack has been built using Spack for quite a few years
- Some really key features that support HEP use cases are in final development
  - Spack Chains allows different spack instances to build on top of each other
  - Better env setup for developers working from source added last year (spack setup)
- A few HEP people were added as developers in Spack GitHub
  - Help get HEP packages merged faster (review of these was slow in the past), e.g., ROOT
- We have our own channel in the Spack Slack workspace
Spack - Packaging Projects

● FNAL MVP
  ○ Provide builds to neutrino community, replacing outdated tools
  ○ SpackDev project provides an easy development environment setup on top of Spack built packages
  ○ MVP (minimum viable product) can be tested, feedback encouraged
  ○ A lot of useful HEP packages contributed (geant4, vecgeom, clhep, genie, log4cpp, lhapdf, ifdhc, …)
  ○ Discussion document on missing features and problem areas prepared for the community
    ■ e.g. sensitivity of package hashes to recipe changes

● CERN FCC/LCG Stack
  ○ Extending use of Spack from “higher level” FCC packages to deeper into the LCG stack
  ○ About 200 packages targeted
  ○ Will have a summer student to work on this

● SuperNEMO
  ○ In the process of migrating from homebrew to Spack
  ○ First results very encouraging - full report in next week’s meeting
Conclusions

- Packaging remains a de facto area where working together as a community helps.
- Group has remained active, informing the community and providing mutual support.
- Prototypes are advanced and providing real solutions.
  - Often now we are in the nitty-gritty and diabolical details.
- Whether “one size fits all” remains to be seen.
- My personal take:
  - Spack the most promising solution for production software builds.
  - CONDA well liked and supported by end users - could be a great way to provide analysis software.