key4hep and spack

Juan Miguel Carceller

CERN

July 11, 2023

key4hep

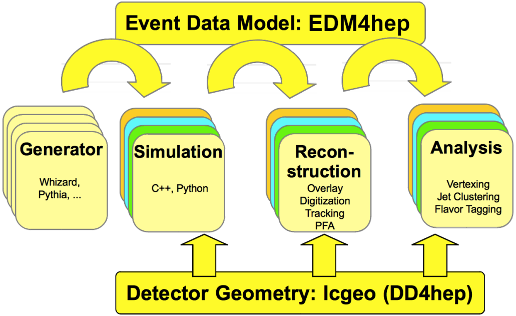
- Turnkey software for future accelerators

- Community with people from many different experiments: CEPC, CLIC, EIC, FCC, ILC, Muon Collider, etc.

- Share components to reduce maintenance and development cost and allow everyone to benefit from its improvements

- Complete data processing framework, from generation to data analysis

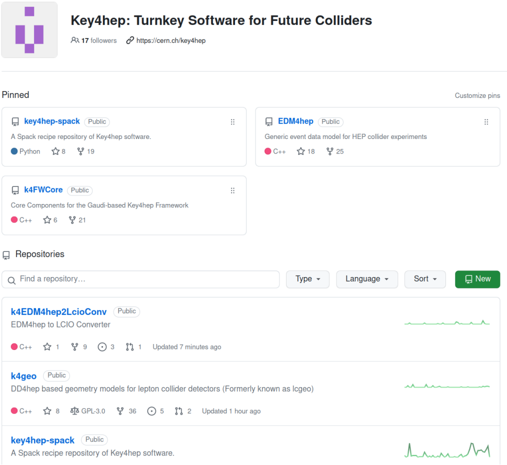
- R&D Programme, but at the same time we support users that rely on key4hep for their productions and other uses



key4hep: Software

- Many repositories under the key4hep organization
  - key4hep-spack for spack recipes
  - EDM4hep the Event Data Model used in key4hep
  - k4FWCore provides the interface between EDMs and Gaudi
  - k4MarlinWrapper allows to call Marlin processors
  - k4geo for detector models
  - . . .

github.com/key4hep



Key4hep: Turnkey Software for Future Colliders

17 followers https://cern.ch/key4hep

Pinned

Customize pins

key4hep-spack Public
A Spack recipe repository of Key4hep software.
● Python ☆ 8 19

EDM4hep Public
Generic event data model for HEP collider experiments
● C++ ☆ 18 25

k4FWCore Public
Core Components for the Gaudi-based Key4hep Framework
● C++ ☆ 6 21

Repositories

Find a repository…

Type ▾ Language ▾ Sort ▾ New

k4EDM4hep2LcioConv Public
EDM4hep to LCIO Converter
● C++ ☆ 1 9 3 1 Updated 7 minutes ago

k4geo Public
DD4hep based geometry models for lepton collider detectors (Formerly known as lcgeo)
● C++ ☆ 8 GPL-3.0 36 5 2 Updated 1 hour ago

key4hep-spack Public
A Spack recipe repository of Key4hep software.

key4hep

- Part of the CERN EP R&D Programme (Software)
- Team of $\sim$ 5 at CERN (recently +2 fellows +1 PhD student full time) actively working on key4hep
- More active developers from other labs like DESY or experiments like EIC
- More users at CERN and other places
- Also supported by HSF and AIDAInnova

Building key4hep: Summary

- Builds are done using spack

- Builds are deployed to cvmfs:
  - `/cvmfs/sw.hsf.org` for releases (every several months, on demand)
  - `/cvmfs/sw-nightlies.hsf.org` for nightly builds, every day

- A total of 502 packages in the last build

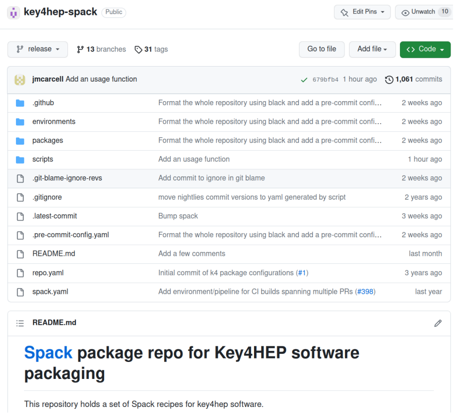
- A bash script that users can source is provided, so users don't need to use spack (but they can)

- Compilers are taken either from the system or provided under `/cvmfs/sw.hsf.org/contrib`

- Three flavours of builds: For CentOs 7, AlmaLinux 9 and Ubuntu 22.04, with more coming soon

Building key4hep: What do we build?

- Our starting point is github.com/key4hep/key4hep-spack
- Set of key4hep recipes (from the repositories under github.com/key4hep) plus repositories under other organizations iLCSoft, HEP-FCC, CEPC . . .
- Scripts for setting up the builds from cvmfs, for adding the latest version of each package to the nightlies environment. . .



jmcarcell Add an usage function ✓ 679bfb4 1 hour ago 1,061 commits

.github	Format the whole repository using black and add a pre-commit confi…	2 weeks ago
environments	Format the whole repository using black and add a pre-commit confi…	2 weeks ago
packages	Format the whole repository using black and add a pre-commit confi…	2 weeks ago
scripts	Add an usage function	1 hour ago
.git-blame-ignore-revs	Add commit to ignore in git blame	2 weeks ago
.gitignore	move nightlies commit versions to yaml generated by script	2 years ago
.latest-commit	Bump spack	3 weeks ago
.pre-commit-config.yaml	Format the whole repository using black and add a pre-commit confi…	2 weeks ago
README.md	Add a few comments	last month
repo.yaml	Initial commit of k4 package configurations (#1)	3 years ago
spack.yaml	Add environment/pipeline for CI builds spanning multiple PRs (#398)	last year

README.md

Spack package repo for Key4HEP software packaging

This repository holds a set of Spack recipes for key4hep software.

- About 75 packages that are not in upstream spack
- At the top there is a virtual package `key4hep-stack` that depends on a lot of other packages

Building key4hep: key4hep-stack

- Depends on most of the key4hep packages
- Added to the spec to ensure that the key4hep packages and other dependencies are built
  - For example development tools for users like jupyter
- Creates a `setup.sh` script with all the modifications done to the environment
- Users source this script (not directly) to get the environment

```
$ cat /cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-07-10/x86_64-almalinux9-gcc11.3.1-opt/key4hep-stack/2023-07-10-gnzgwf/setup.sh
export PERL5LIB=/cvmfs/sw-nightlies.hsf.org/key4hep/release...
export KEY4HEP_STACK=/cvmfs/sw-nightlies.hsf.org/key4hep/re...
export K4GEN=/cvmfs/sw-nightlies.hsf.org/key4hep/releases/2...
export DD4hepINSTALL=/cvmfs/sw-nightlies.hsf.org/key4hep/re...
export K4GEO=/cvmfs/sw-nightlies.hsf.org/key4hep/releases/2...
export DELPHES_DIR=/cvmfs/sw-nightlies.hsf.org/key4hep/rele...
export G4RADIOACTIVEDATA=/cvmfs/sw-nightlies.hsf.org/key4he...
export GI_TYPELIB_PATH=/cvmfs/sw-nightlies.hsf.org/key4hep/...
export PYTHONPATH=/cvmfs/sw-nightlies.hsf.org/key4hep/relea...
...
```

Building key4hep: How do we build?

- Builds are done in (docker) containers for CentOs 7, AlmaLinux 9 and Ubuntu 22.04; in Openstack instances provided by CERN that host Gitlab CI runners

- The spack environments can be found here and look like this:

```yaml
spack:
  view: false
  include:
  - ../key4hep-common/config-nightlies.yaml
  - ../key4hep-common/packages.yaml
  - ../key4hep-common/compilers.yaml
  repos:
  - ../..
  specs:
  - key4hep-stack+devtools
```

- `config-nightlies.yaml` contains the projections or paths where the packages will be found
- `packages.yaml` contains the variants that we use for several packages
- `compilers.yaml` contains the paths to the compilers (either the system nes or found in /cvmfs)

Building key4hep: How do we build?

- In addition, for the nightlies we have a script that adds the latest commit of each repository that we want to the specs, so it would look like:

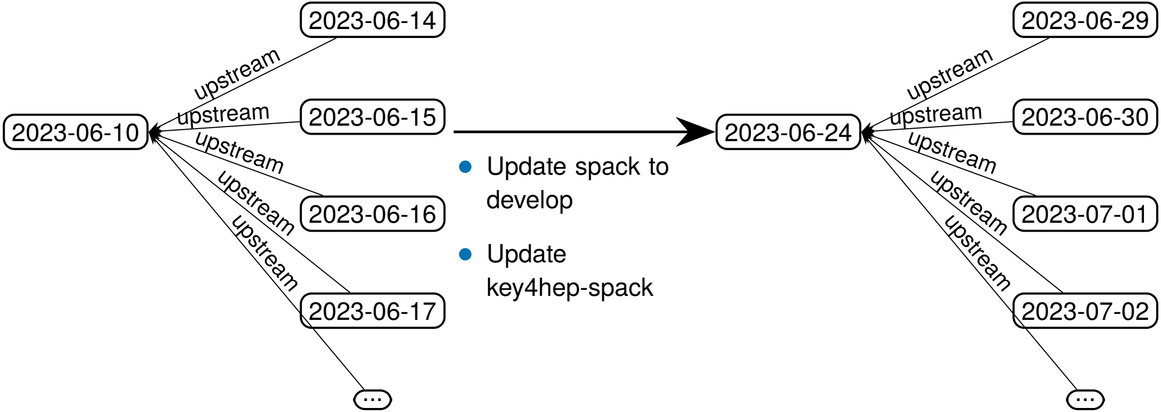
```
specs:
- key4hep-stack+devtools

- edm4hep@93effb8f84673ba13103fb7dcf7850cf7c8da719=develop
- podio@d8294d2a2aad3c737c7b9f07ad9e5b2d826f50b7=develop
- dd4hep@21df99ad97d1e4aad9aa444a9d2f190f64676f06=develop
- k4fwcore@94d12628c961eaf672f8332a53ea9569e8072d13=develop
- k4projecttemplate@72f6cbca810c3211b3c70a33abf75a1c341b8abd=develop
- k4simdelphes@430f6e4f54446447587bab3d21f98295b7feb015=develop
- k4clue@517b6c1cddef3ab38ed9a1d954f485019d784162=develop
- k4gen@a88e2b7ff10489af4ac4c1fd21ebfc76ea7eb51b=develop
- k4simgeant4@183d261ac61478cd5bad91faa029236ff6e5f86d=develop
- delphes@66b4866275f483dcd7394b4fe3f44914ecd56990=develop
- fccsw@07f9fcddbe71d777be25487dc7eb324682d41dcb=develop
- dual-readout@f386bde135c5874e3650c17c0d16781fa5d53457=develop
...
```

- =develop seems to be needed, when I tried without it spack wouldn't be

able to find that these commits come after the latest version and would try to install the latest version

Building key4hep: How do we build?

- For the nightlies, every ∼ two weeks the latest commit of spack is picked and a full build is done (builds from scratch)
- The rest of the days only a subset of repositories are built, the ones that have changed; these use the builds from scratch as upstream



- Update spack to develop
- Update key4hep-spack

Building key4hep: Build workflow

- When doing a full build and updating the spack commit, building never works
- Fixes are found and either a PR upstream or a workaround is made
- If a PR upstream, then the commit in the PR is cherry picked for the build
- It’s not too bad since we update frequently, there can’t be that many failures in two weeks worth of spack commits, but there is always at least one

Building key4hep: Deployment

- After a build is completed and tested it is deployed to cvmfs by rsyncing the build to the publisher node

- Several files are provided for each build for reproducibility:
  - `.spack-commit`: Which commit of spack was used to build this release
  - `.key4hep-spack-commit`: Which commit of `key4hep-spack` was used to build this release
  - `.cherry-pick`: A script that cherry picks the same commits as it happened during build time

.spack-commit

```
fde33e66be9c0b0c3f7b20c823447538455998f4
```

.key4hep-spack-commit

```
f6a53faaf9b82ab904f7d37a0a0a97eaeda5d2b7
```

.cherry-pick

```
git remote add jmcarcell https://github.com/jmcarcell/spack
git fetch jmcarcell
# add a recent version of madgraph
git cherry-pick 8cef8fc3140cb893250fdf2891f590b2993a704e --no-commit
git cherry-pick 2e8202d6c6a4f6a2e91b5841fac1cc5610cc6d7d --no-commit
# Fix _STAT_VER issues in xrootd, remove when
git cherry-pick 8578fc1327ba7b427cf891f4b6ba1817acf57449 --no-commit
```

Building key4hep: Result

```
$ ls /cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-06-24/x86_64-almalinux9-gcc11.3.1-opt
acts                         pmix
aida                         podio
aidatt                       popt
alpaka                       protobuf
assimp                       psimd
...                          ...
```

Inside each package there is a folder containing the version and the hash

```
$ cd acts
$ ls
23.2.1-hicgwc
```

And then the actual package is inside this folder:

```
$ cd 23.2.1-hicgwc
$ ls
bin  include  lib64
```

Building key4hep: User side

- We provide a script that checks the OS and sources the appropiate script

- With a `source /cvmfs/sw-nightlies.hsf.org/key4hep/setup.sh` , users on CentOs 7, Ubuntu 22.04 or AlmaLinux 9 get access to the stack

```
AlmaLinux/RHEL 9 detected
Setting up the latest Key4HEP software stack from CVMFS ...
 ...  Key4HEP release: 2023-07-10
 ...  Use the following command to reproduce the current environment:
 ...
         source /cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-07-10/x86_64-almalinux9-gcc11.3.1-opt/key4hep-stack/2023-07-10-gnzgwf/setup.sh
 ...
 ...  If you have any issues, comments or requests open an issue at https://github.com/key4hep/key4hep-spack/issues
```

- There is an equivalent `source /cvmfs/sw.hsf.org/key4hep/setup.sh`

- The script points the users to the actual `setup.sh` script that can be sourced to reproduce the current environment later

Issues

Issues: Build Caches

- We have been testing build caches

- A build was done for one day (for example 2022-05-20) and we tried to install in the location corresponding to a different day (2022-05-30)

- Relocation failed in several occassions:
  - git has binaries in non standard locations, but the RPATHS inside those were still pointing to the build location and not to the one where they were being installed
  - At least one package with symbolic links didn't have the symbolic links updated to the new locations

- At that point the build was broken and no more research was done, but it's possible there are more issues

Issues: Build Caches, Examples

- Git example, installed on 2023-05-30, rpath pointing to where the build cache was originally built

```
$ pwd
/cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-05-30/x86_64-almalinux9-gcc11.3.1-opt/git/2.40.0-bwhwxr/libexec/git-core
$ objdump -p git
...
RPATH                /cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-05-20/x86_64-almalinux9-gcc11.3.1-opt/git/2.40.0-bwhwxr/lib...
...
```

- Python example, broken symlinks:

```
$ ls -lah /cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-05-30/x86_64-almalinux9-gcc11.3.1-opt/python/3.10.10-plst3j/bin
total 102K
drwxr-sr-x. 2 cvmfs cvmfs  235 Jan  1  1970 .
drwxr-sr-x. 7 cvmfs cvmfs   70 Jan  1  1970 ..
lrwxrwxrwx. 1 cvmfs cvmfs    9 May 30 08:27 2to3 -> 2to3-3.10
-rwxr-xr-x. 1 cvmfs cvmfs  201 May 30 08:27 2to3-3.10
lrwxrwxrwx. 1 cvmfs cvmfs    8 May 30 08:27 idle3 -> idle3.10
-rwxr-xr-x. 1 cvmfs cvmfs  199 May 30 08:27 idle3.10
lrwxrwxrwx. 1 cvmfs cvmfs    9 May 30 08:27 pydoc3 -> pydoc3.10
-rwxr-xr-x. 1 cvmfs cvmfs  184 May 30 08:27 pydoc3.10
lrwxrwxrwx. 1 cvmfs cvmfs  121 May 30 08:27 python -> /cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-05-20/x86_64-almalinux9-gcc11.3.1-opt/python/3.10.10-plst3j/bin
lrwxrwxrwx. 1 cvmfs cvmfs   10 May 30 08:27 python3 -> python3.10
-rwxr-xr-x. 1 cvmfs cvmfs  27K May 30 08:27 python3.10
-rwxr-xr-x. 1 cvmfs cvmfs 3.1K May 30 08:27 python3.10-config
-rwxr-xr-x. 1 cvmfs cvmfs  65K Jan  1  1970 python3.10-gdb.py
lrwxrwxrwx. 1 cvmfs cvmfs   17 May 30 08:27 python3-config -> python3.10-config
lrwxrwxrwx. 1 cvmfs cvmfs  128 May 30 08:27 python-config -> /cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-05-20/x86_64-almalinux9-gcc11.3.1-opt/python/3.10.10-pls
```

Issues: Build Caches, Examples

- Git example, installed on 2023-05-30, rpath pointing to where the build cache was originally built

```
$ pwd
/cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-05-30/x86_64-almalinux9-gcc11.3.1-opt/git/2.40.0-bwhwxr/libexec/git-core
$ objdump -p git
...
RPATH                /cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-05-20/x86_64-almalinux9-gcc11.3.1-opt/git/2.40.0-bwhwxr/lib...
...
```

- Python example, broken symlinks:

```
$ ls -lah /cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-05-30/x86_64-almalinux9-gcc11.3.1-opt/python/3.10.10-plst3j/bin
total 102K
drwxr-sr-x. 2 cvmfs cvmfs  235 Jan  1  1970 .
drwxr-sr-x. 7 cvmfs cvmfs   70 Jan  1  1970 ..
lrwxrwxrwx. 1 cvmfs cvmfs    9 May 30 08:27 2to3 -> 2to3-3.10
-rwxr-xr-x. 1 cvmfs cvmfs  201 May 30 08:27 2to3-3.10
lrwxrwxrwx. 1 cvmfs cvmfs    8 May 30 08:27 idle3 -> idle3.10
-rwxr-xr-x. 1 cvmfs cvmfs  199 May 30 08:27 idle3.10
lrwxrwxrwx. 1 cvmfs cvmfs    9 May 30 08:27 pydoc3 -> pydoc3.10
-rwxr-xr-x. 1 cvmfs cvmfs  184 May 30 08:27 pydoc3.10
lrwxrwxrwx. 1 cvmfs cvmfs  121 May 30 08:27 python -> /cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-05-20/x86_64-almalinux9-gcc11.3.1-opt/python/3.10.10-plst3j/bin
lrwxrwxrwx. 1 cvmfs cvmfs   10 May 30 08:27 python3 -> python3.10
-rwxr-xr-x. 1 cvmfs cvmfs  27K May 30 08:27 python3.10
-rwxr-xr-x. 1 cvmfs cvmfs 3.1K May 30 08:27 python3.10-config
-rwxr-xr-x. 1 cvmfs cvmfs  65K Jan  1  1970 python3.10-gdb.py
lrwxrwxrwx. 1 cvmfs cvmfs   17 May 30 08:27 python3-config -> python3.10-config
lrwxrwxrwx. 1 cvmfs cvmfs  128 May 30 08:27 python-config -> /cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-05-20/x86_64-almalinux9-gcc11.3.1-opt/python/3.10.10-pls
```

Issues: Development

- How to set up a good local development environment is not solved in key4hep
- Question: use spack or not?
- Some people use spack to set up a local environment
  - It's not trivial to get started
  - Long time to get a working setup for the first time
- Other people compile manually and add the new paths to `LD_LIBRARY_PATH` and other env variables

Issues: Development

- Another issue we've found is to be able to install only what we want on top of an installation

  - Several files are provided for reproducibility:
    - `.spack-commit`: Which commit of spack was used to build this release
    - `.key4hep-spack-commit`: Which commit of `key4hep` was used to build this release
    - `.cherry-pick`: A script that cherry picks the same commits as it happened during build time
- Even when reproducing using those three files to reproduce exactly the environment that was used during the build, some hashes change (even for packages with the same commit that have not been changed for a long time), which will trigger compilation and installation.

- Example: We have a github CI workflow that is triggered in PRs, and builds the current package and dependent ones (to check that we don't break the build).
  - Almost always will build more than needed (but it's a few packages)

Summary

- We use successfully spack in key4hep to build our software stacks; plan on keep using it
  - Benefit a lot from maintenance of the recipes from the community
- System to build releases and nightlies automatically
- Some issues when we try to move beyond building