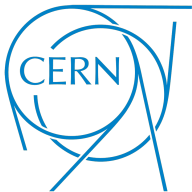


Building the Key4hep stack



Juan Miguel Carceller

CERN

May 9, 2023

News and changes

- Spack fork archived, not used anymore (some things may still need to be upstreamed) `/cvmfs/sw.hsf.org/spackages7`
- Builds done in containers for Centos 7, Almalinux 9 and Ubuntu 22.04
- New nightlies for the three OSES, new release only for Centos 7
- Users don't need to do anything, if you source from Centos 7 you get the Centos 7 nightly
- All in Gitlab CI, it's possible to build and deploy to cvmfs from the web

```
[jcarcell@lxplus901 ~]$ source /cvmfs/sw-nightlies.hsf.org/key4hep/setup.sh
AlmaLinux 9 detected
Setting up the latest Key4HEP software stack from CVMFS ...
... Key4HEP release: 2023-05-08
... Use the following command to reproduce the current environment:
...
... source /cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-05-08/x86_64-almalinux9-
gcc11.3.1-opt/key4hep-stack/2023-05-08-ygxo7/setup.sh
...
... If you have any issues, comments or requests open an issue at https://github.com/key
4hep/key4hep-spack/issues
```

```
[jcarcell@lxplus736 ~]$ source /cvmfs/sw-nightlies.hsf.org/key4hep/setup.sh
Centos 7 detected
Setting up the latest Key4HEP software stack from CVMFS ...
... Key4HEP release: 2023-05-03
... Use the following command to reproduce the current environment:
...
... source /cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-05-03/x86_64-centos7-gcc
12.2.0-opt/key4hep-stack/2023-05-03-q6pq42/setup.sh
...
... If you have any issues, comments or requests open an issue at https://github.com/key
4hep/key4hep-spack/issues
```

Requests

- Pytorch (without CUDA) is now being included in the nightlies
- matplotlib now can be used to display graphics
- ROOT is being built with FFTW support
- Being able to select different releases, nightlies since the script we have will always source the latest which is not what everyone may want. Only for nightlies, -l flag for listing the releases?:

```
source /cvmfs/sw-nightlies.hsf.org/key4hep/setup.sh -r 2023-05-08
```

- Not done yet:
 - Having 'clang-format' available. Currently, we don't build 'llvm' and that is a big package. But for Alma9 on lxplus for example 'clang-format' is available and it's not so old

```
[jcarcell@lxplus922 ~]$ clang-format --version  
clang-format version 14.0.6 (Red Hat 14.0.6-4.el9_1)
```

Nightlies

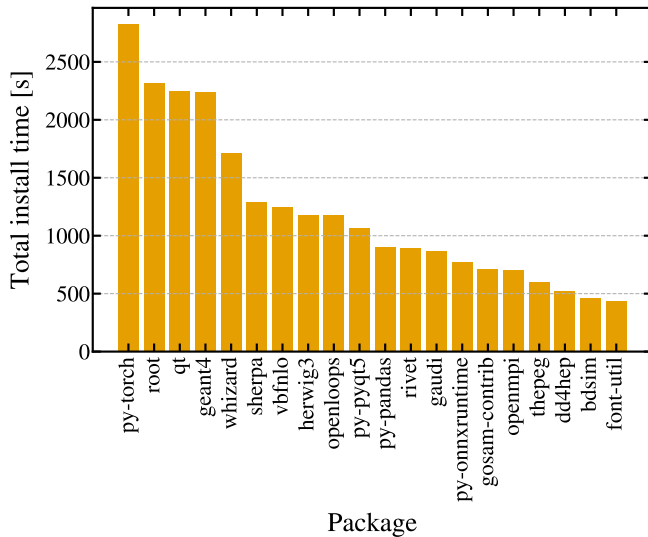
- Two build types depending if the build is done from scratch or not
- Builds from scratch will get the latest packages and let us know that the changes in spack don't break our builds - Updates every O(weeks)
- Daily builds that use as upstream the builds from scratch, they only build the packages that have changed (or those that depend on a package that has changed)
- **New:** Nightlies will be deleted after some time
- We have 100 GB and each build from scratch takes 20 GB (uncompressed)

Useful information

- Three new hidden files in each release
 - `.scratch`: If it's there, that means it's a build from scratch (all packages)
 - `.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
- All releases are under `/cvmfs/sw-nightlies.hsf.org/key4hep/releases`
- The latest one can always be found in `/cvmfs/sw-nightlies.hsf.org/key4hep/releases/latest`

```
[jcarcell@lxplus923 latest]$ ls -l
total 2
x86_64-almalinux9-gcc11.3.1-opt -> /cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-05-08/x86_64-almalinux9-gcc11.3.1-opt
x86_64-centos7-gcc12.2.0-opt -> /cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-05-03/x86_64-centos7-gcc12.2.0-opt
x86_64-ubuntu22.04-gcc11.3.0-opt -> /cvmfs/sw-nightlies.hsf.org/key4hep/releases/2023-05-02/x86_64-ubuntu22.04-gcc11.3.0-opt
```

Some stats



Nightly cleanup

- The following folders will be removed...
 - `/cvmfs/sw-nightlies.hsf.org/key4hep/spackages4` (42 GB)
 - `/cvmfs/sw-nightlies.hsf.org/key4hep/spackages5` (54 GB)
 - `/cvmfs/sw-nightlies.hsf.org/key4hep/spackages6` (88 GB)

Workareas

- Problem: Building a package requires people to mess with environment variables or use spack, both non trivial, esp. when multiple packages are involved
- Workareas could solve this
- Workflow:
 - Make a folder for the workarea (and source a script)
 - Clone all the packages you are going to work on
 - Execute `k4-workarea` to set the list of packages
 - Run `k4-build` when you want to build
 - The setup will take care of paths so for the packages you build it will pick up the local ones but for the ones you don't build it will pick the ones from the release or nightly