

10 things you didn't know yet about



<https://www.ugent.be/hpc>

Kenneth Hoste (HPC-UGent)

kenneth.hoste@ugent.be



Vlaanderen
is computing

<https://www.vscentrum.be>

5th EasyBuild User Meeting
Barcelona - Jan 31st 2020



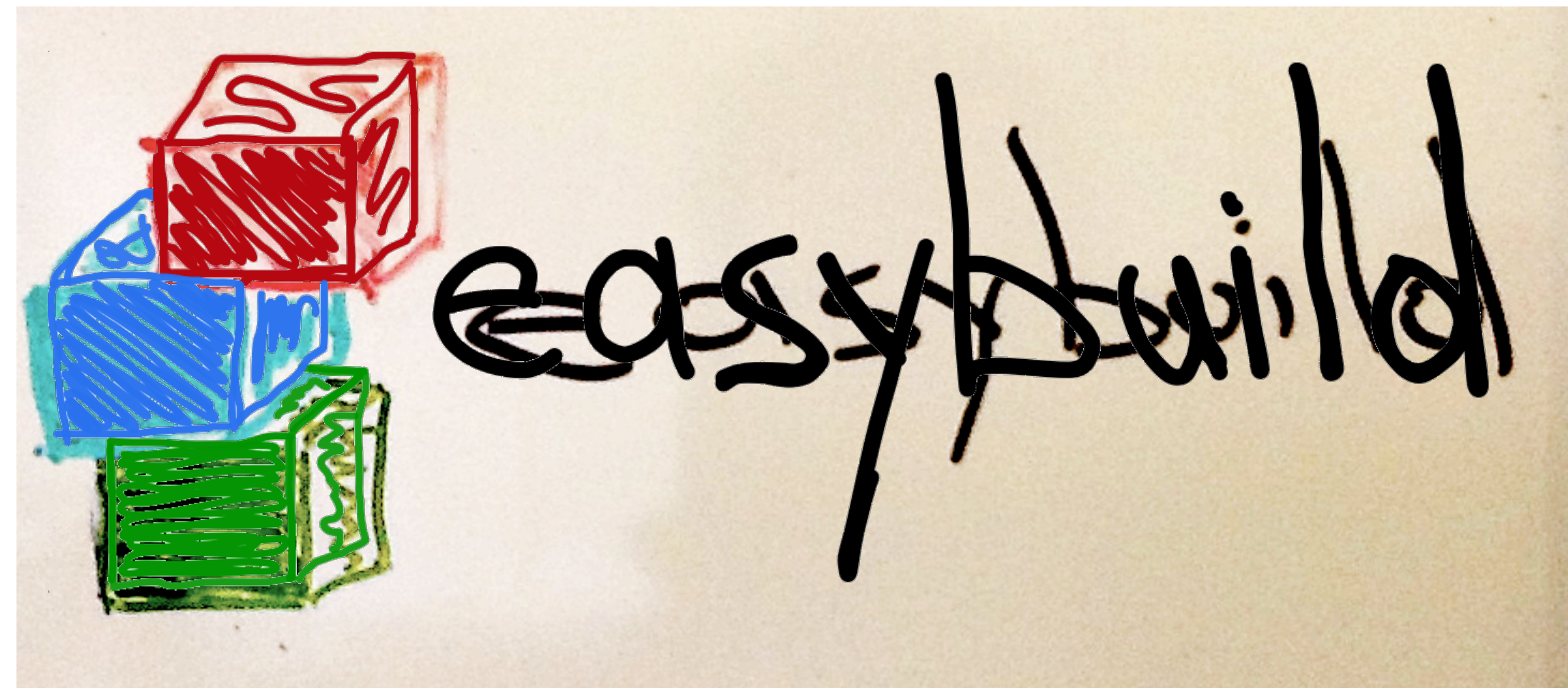
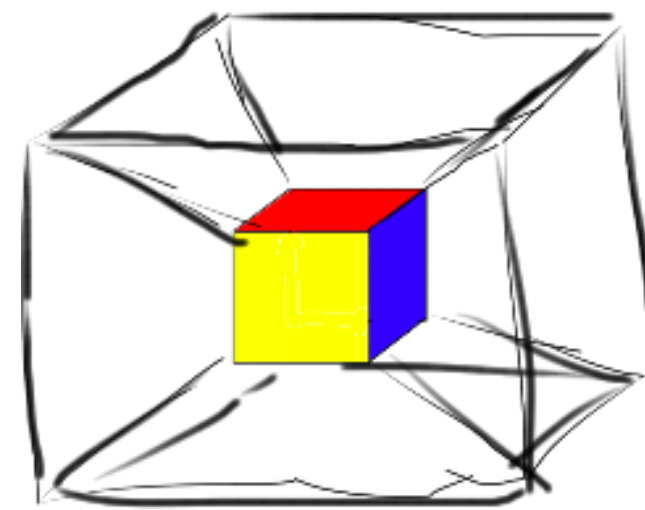
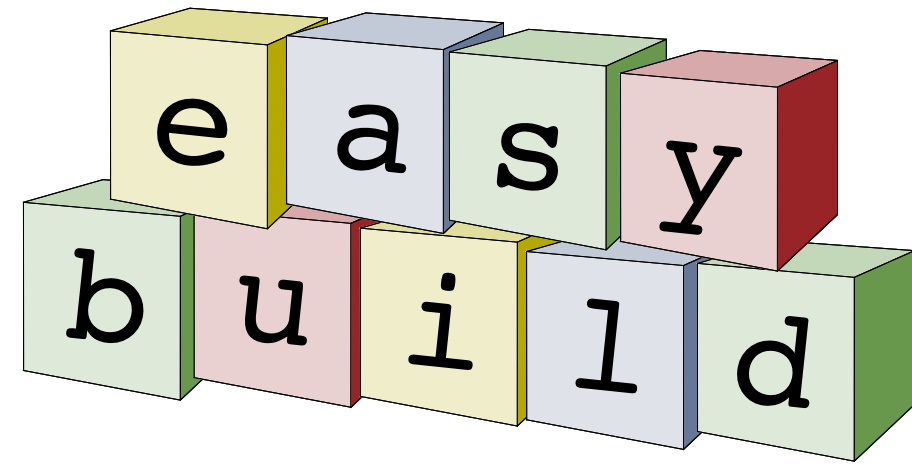
<https://github.com/easybuilders/easybuild/wiki/5th-EasyBuild-User-Meeting>

Confession: I did not create EasyBuild

- EasyBuild was created in summer of 2009
- by HPC-UGent tech lead Stijn De Weirdt
- handed over to me early 2011
- it took me a while to figure out ***what*** it did, before I could start to try and understand *how* it worked
- it took several re-designs and a couple of student interns before we were comfortable in making it public (Apr. 2012)



The EasyBuild logo is (literally) a whiteboard sketch



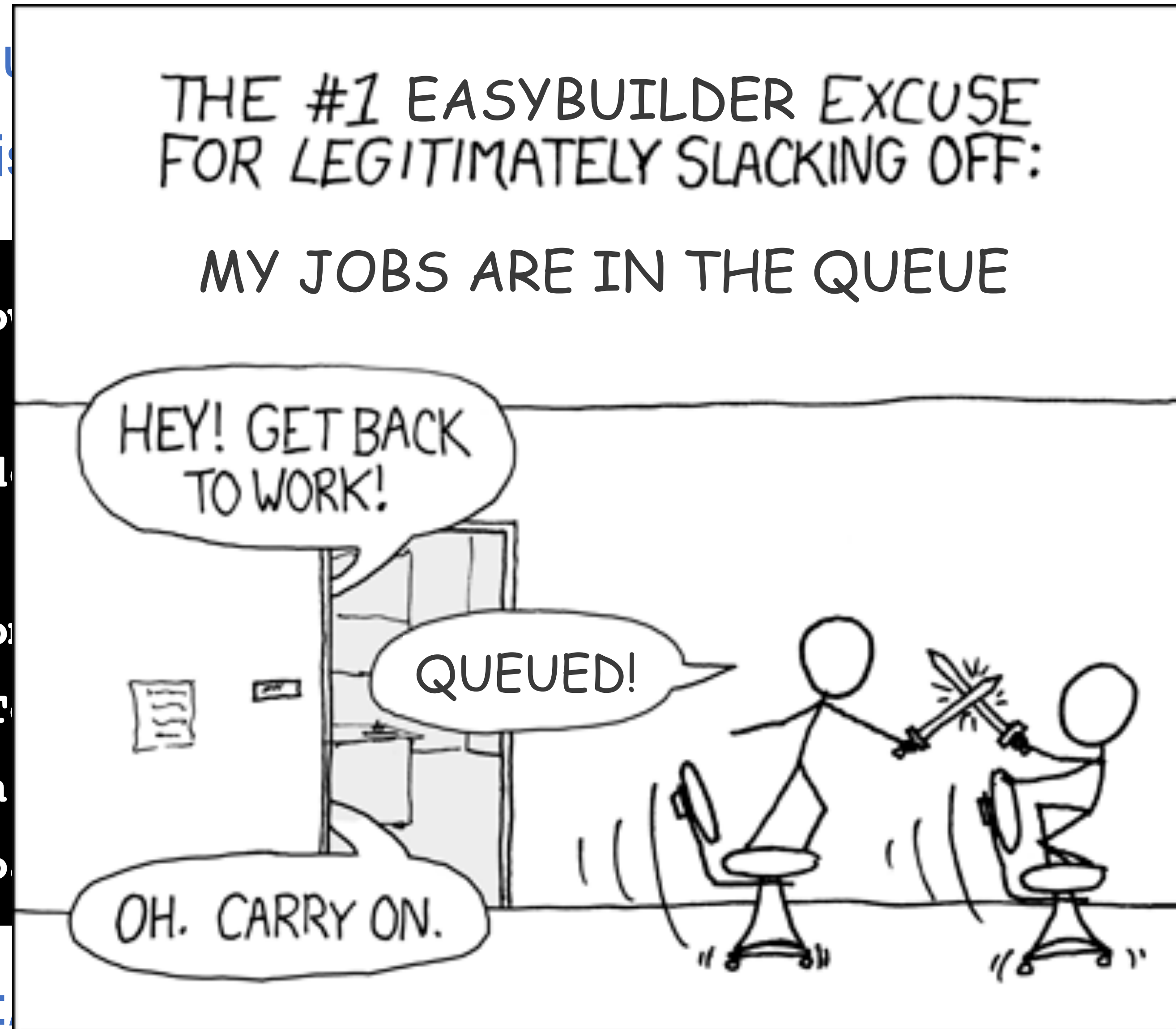
(some Belgian beer may have been involved)

Distributed software installation

You can let EasyBuild
effectively parallelis

```
$ eb TensorFlow
...
== resolving dependencies
...
== List of submitted jobs:
521812; ...; TensorFlow/2.1.0-fosscuda
== Submitted jobs:
```

see <https://easybuild.github.io/docs/job.html>



"eb --robot --job",
odes.

```
eb --robot
```

```
.0_231):
1 (TensorFlow/
```

[_jobs.html](https://easybuild.github.io/docs/job.html)

Use EasyBuild to install Spack, (don't) use Spack to install EasyBuild

```
$ eb Spack-0.12.1.eb
```

```
...
```

```
== building and installing Spack/0.12.1...
```

```
...
```

```
== COMPLETED: Installation ended successfully (took 18 sec)
```

```
$ module load Spack/0.12.1
```

```
$ spack install easybuild@4.1.1
```

```
...
```

```
==> Error: SyntaxError: Update Spack with 'git pull', and start over.
```

Use EasyBuild to install Spack, it's the one true way.



ETH zürich

CSCS User Portal Getting Started ▾ Scientific Computing ▾ Storage ▾ Tools ▾ My Projects

SCIENTIFIC COMPUTING

Scientific Applications

Amber
CP2K
CPMD
GROMACS
LAMMPS
NAMD
Quantum ESPRESSO
SIRIUS
VASP

Data Science

ABCpy
Dask

Spack Package Manager

Installation

CSCS provides a set of [configuration files](#) and installation recipes enabling users to get started with [Spack](#) on [Piz Daint](#). These files define the basic configuration for the Cray modules, as well as for Spack recipes, environment modules and installation directories. Additionally, CSCS also provides an EasyBuild recipe to configure and install the development version of Spack.

On Piz Daint, the development version of Spack can be installed using EasyBuild:

```
module load daint-gpu
module load EasyBuild-custom
eb Spack-daint-git-develop.eb
```

This should install Spack in the folder `$EASYBUILD_INSTALLPATH/software/Spack/daint`. Please note that the variable `$EASYBUILD_INSTALLPATH` is defined by the `EasyBuild-custom` module.

The Spack module is accessible as follows:

```
module use $EASYBUILD_INSTALLPATH/modules/all
module load Spack
```

Spack Package Manager

Installation
Build your Program
CSCS Configuration Files

[Back to top](#)

EasyBuild detects typos in easyconfigs

```
$ eb spack.eb
```

```
== temporary log file in case of crash /tmp/eb-Wrow3w/easybuild-5jCkYv.log
```

```
ERROR: Failed to process easyconfig /home/tgamblin/spack.eb:
```

```
You may have some typos in your easyconfig file: buildotps -> buildopts
```

Partial command line options

```
# you don't have to use the full option name
```

```
$ eb --install-latest-eb-release
```

```
...
```

```
== EasyBuild/4.1.1 is already installed (module found), skipping
```

```
# just enough characters to uniquely identify the option is enough
```

```
$ eb --install-la
```

```
...
```

```
== EasyBuild/4.1.1 is already installed (module found), skipping
```


Customising EasyBuild with hooks

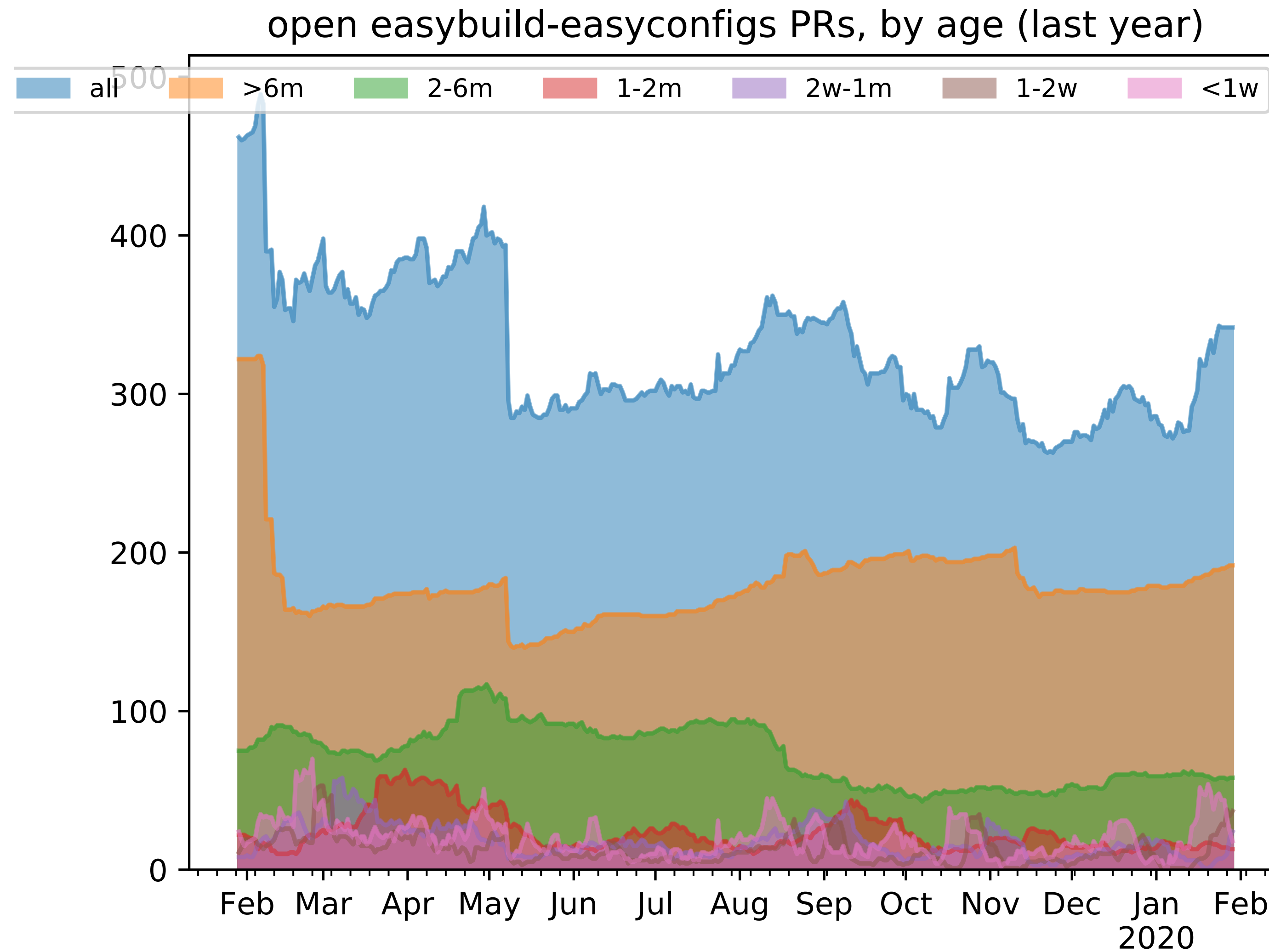
You can customise EasyBuild's behaviour as needed via a single Python module implementing hooks.

```
1  def pre_configure_hook(self, *args, **kwargs):
2
3      """Example pre-configure hook to replace --with-verbs with --without-verbs for OpenMPI."""
4
5      if self.name == 'OpenMPI' and '--with-verbs' in self.cfg['configopts']:
6
7          self.log.info("[pre-configure hook] Replacing --with-verbs with --without-verbs")
8
9          self.cfg['configopts'] = self.cfg['configopts'].replace('--with-verbs', '--without-verbs')
```

```
$ eb --hooks=$HOME/my_eb_hooks.py OpenMPI.eb
...
== configuring...
== Running pre-configure hook...
```

see <https://easybuild.readthedocs.io/en/latest/Hooks.html>

Maintainers vs contributors



We are always looking for additional maintainers...

eb --trace (eb -T)

Show executed commands in output of 'eb' command.

(pro tip: `export EASYBUILD_TRACE=1`)

```
$ eb bzip2-1.0.6.eb --trace
```

```
...
```

```
== building...
```

```
>> running command:
```

```
[started at: 2020-01-30 14:44:37]
```

```
[output logged in /tmp/eb-8bC8Jr/easybuild-run_cmd-k0snTG.log]
```

```
make -j 4 CC=gcc CFLAGS='-Wall -Winline -O3 -fPIC -g $(BIGFILES)'
```

```
>> command completed: exit 0, ran in 00h01m02s
```


eb --trace (eb -T)

Show executed commands in output of 'eb' command.

(pro tip: `export EASYBUILD_TRACE=1`)

```
$ eb bzip2-1.0.6.eb --trace
...
== sanity checking...
>> file 'bin/bunzip2' found: OK
>> file 'bin/bzip2' found: OK
>> file 'include/bzlib.h' found: OK
>> file 'lib/libbz2.a' found: OK
...
```

eb --missing-modules (eb -M)

Show quick overview of what is missing for installing
a specific (set of) easyconfig(s).

```
$ eb --from-pr 9795 --missing-modules
```

```
...
```

```
3 out of 52 required modules missing:
```

- * MMseqs2/10-6d92c-intel-2019b (MMseqs2-10-6d92c-intel-2019b.eb)
- * BLAST+/2.9.0-iimpi-2019b (BLAST+-2.9.0-iimpi-2019b.eb)
- * OrthoFinder/2.3.8-intel-2019b-Python-3.7.4 (OrthoFinder-2.3.8-intel-2019b-Python-3.7.4.eb)

Using EasyBuild as a Python library

```
from easybuild.tools.filetools import read_file, write_file
from easybuild.tools.modules import modules_tool
from easybuild.tools.options import set_up_configuration

# set up EasyBuild configuration
set_up_configuration()

modtool = modules_tool()

# do useful stuff here, using functions provided by EasyBuild...
```


EasyBuild's best feature: the community !

