

easybuild

Past, present & future

3rd EasyBuild User Meeting

Jan 30th - Feb 1st 2018

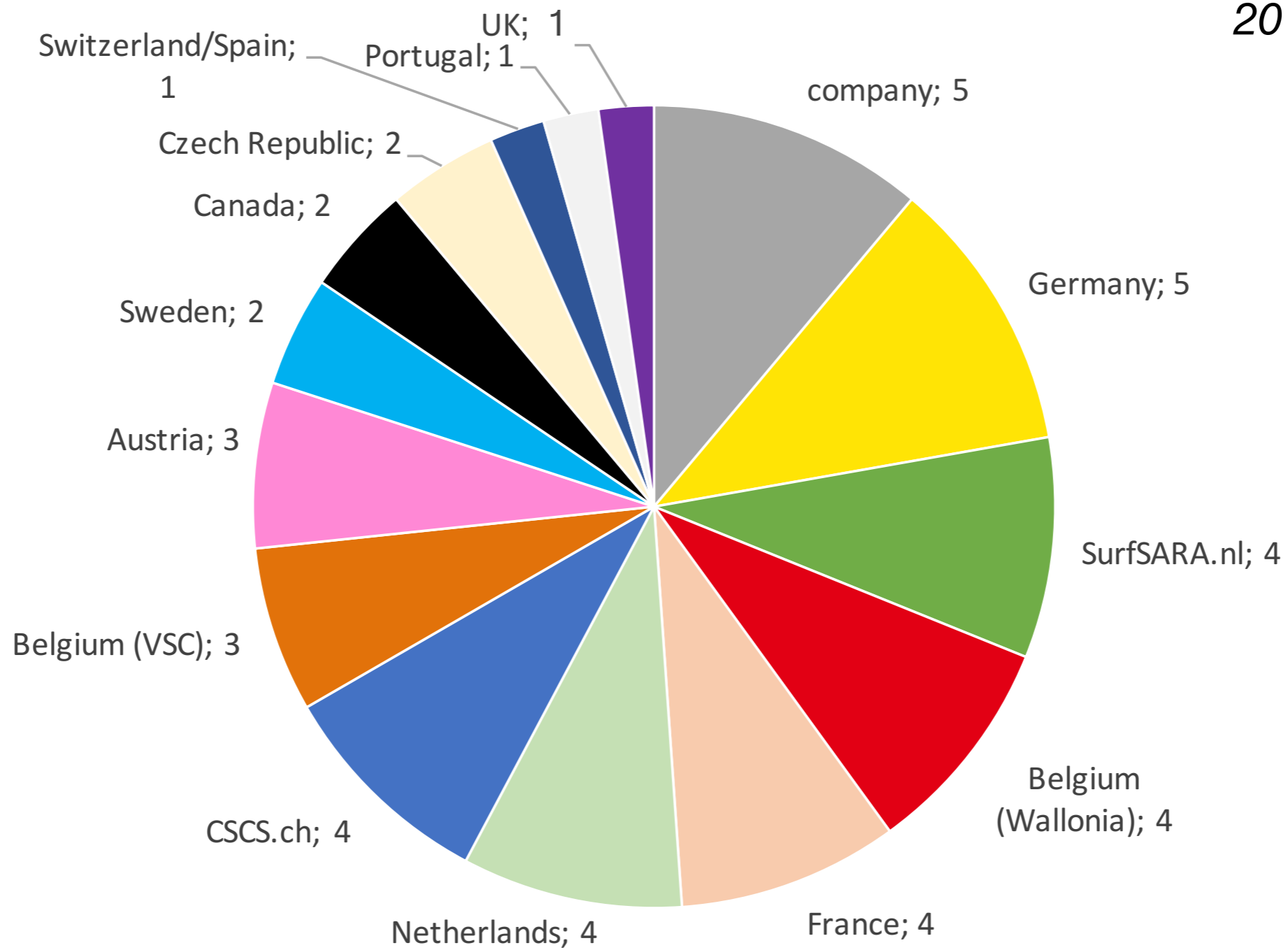
SURFsara - Amsterdam (.nl)

<https://github.com/easybuilders/easybuild/wiki/3rd-EasyBuild-User-Meeting>

45 attendees, 10+ countries

2016 (Ghent): 19

2017 (Jülich): 35



Agenda (Tuesday)

[9.30am] Welcome & coffee

[10.00am] **EasyBuild: past present & future** (*Kenneth Hoste, HPC-UGent*)

[11.30am] **Modules v4** (*Xavier Delaruelle, CEA.fr*)

[12.00] (*lunch*)

[1.00pm] **site presentations**

- SurfSARA (*John Donners*)
- JSC (*Damian Alvarez*)
- University of Muenster (*Holger Angenent*)

[2.00pm] datacenter tour (group I)

[2.00pm] hands-on session

[3.00pm] (*coffee break*)

[5.00pm] **Recent developments in Spack** (*Todd Gamblin, LLNL - USA*) **[remote]**

Agenda (Wednesday)

[9.00am] **Nix as HPC package management system** (*V. Reis, B. Bzeznik - INRIA*)

[9.30am] **EasyBuild + Nix + CVMFS @ ComputeCanada** (*B. Oldeman, ComputeCanada*)

[10.00am] **Introduction to uDocker** (*Luís ALves, LIP, Portugal*)

[10.30am] (*coffee break*)

[11.00am-12.00] hands-on session

[12.00] (*lunch*)

[1.00pm] **site presentations**

- CSCS (*Teo Ioannis*)
- CEA.fr (*Laurent Nguyen*)
- RUG.nl (*Fokke Dijkstra*)
- Bright Computing (*Panos Lampropoulos*)

[2.00pm] datacenter tour (group II)

[2.00pm] hands-on session

[3.00pm-3.30pm] (*coffee break*)

[4.00pm] **Update on Lmod + XALT** (*Robert McLay, TACC*) **[remote]**

[5.00pm] **Update on Singularity** (*Gregory Kurtzer, Sylabs Inc.*) **[remote]**

Agenda (Thursday)

[9.00am] **site presentations**

- Experiences with RPATH support in EasyBuild
(*Pablo Escobar, sciCORE/UniBas/SIB*)
- ...
- ...

[10.00am] (*coffee break*)

[10.30am] hands-on session

[12.00] (*lunch*)

[1.00pm] datacenter tour (group III)

[1.00pm] hands-on session

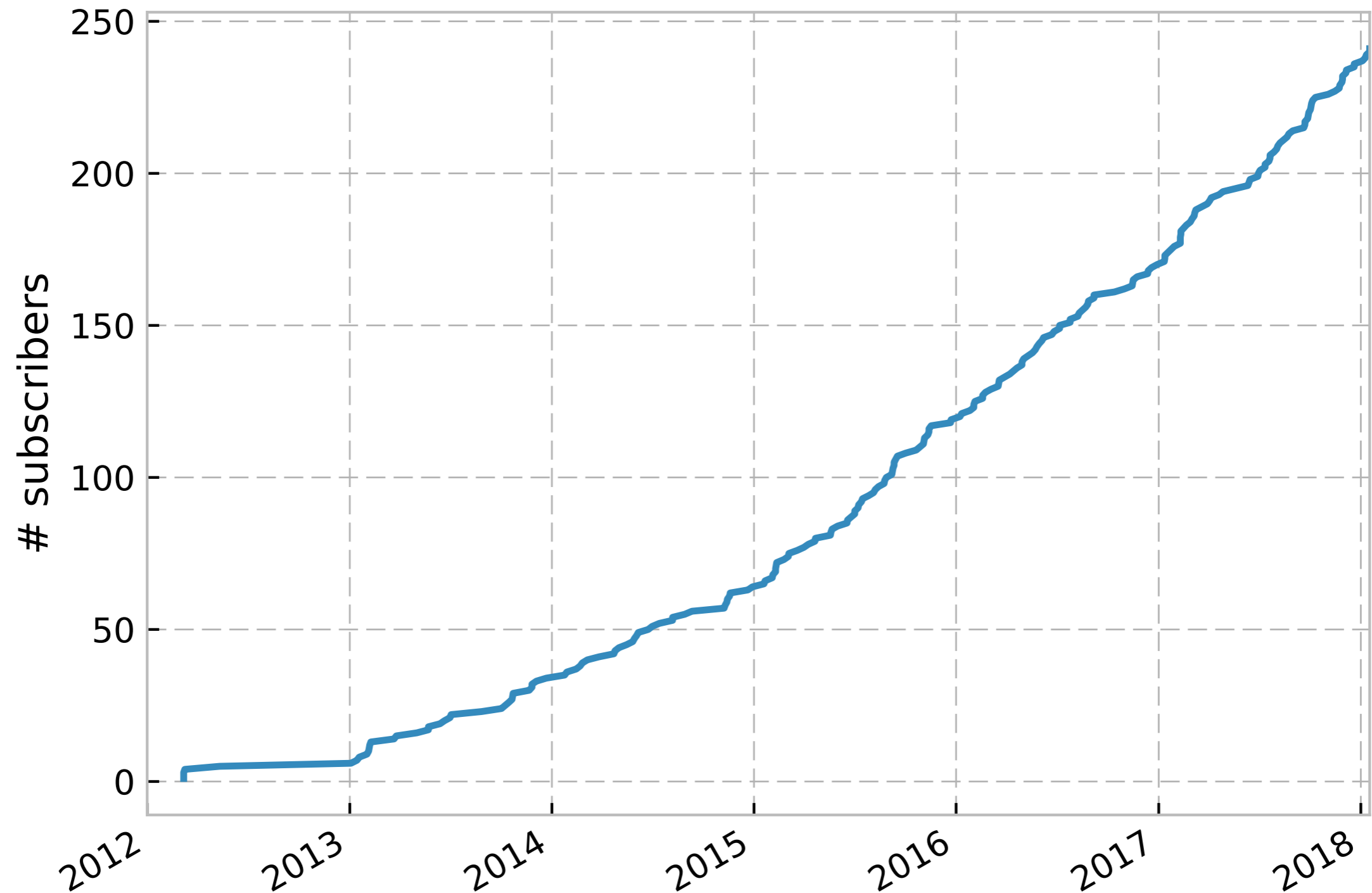
[2.00pm] **Guix@HPC** (*Ricardo Wurmus*)

[2.30pm] **Guix for complex software environments** (*Pjotr Prins*)

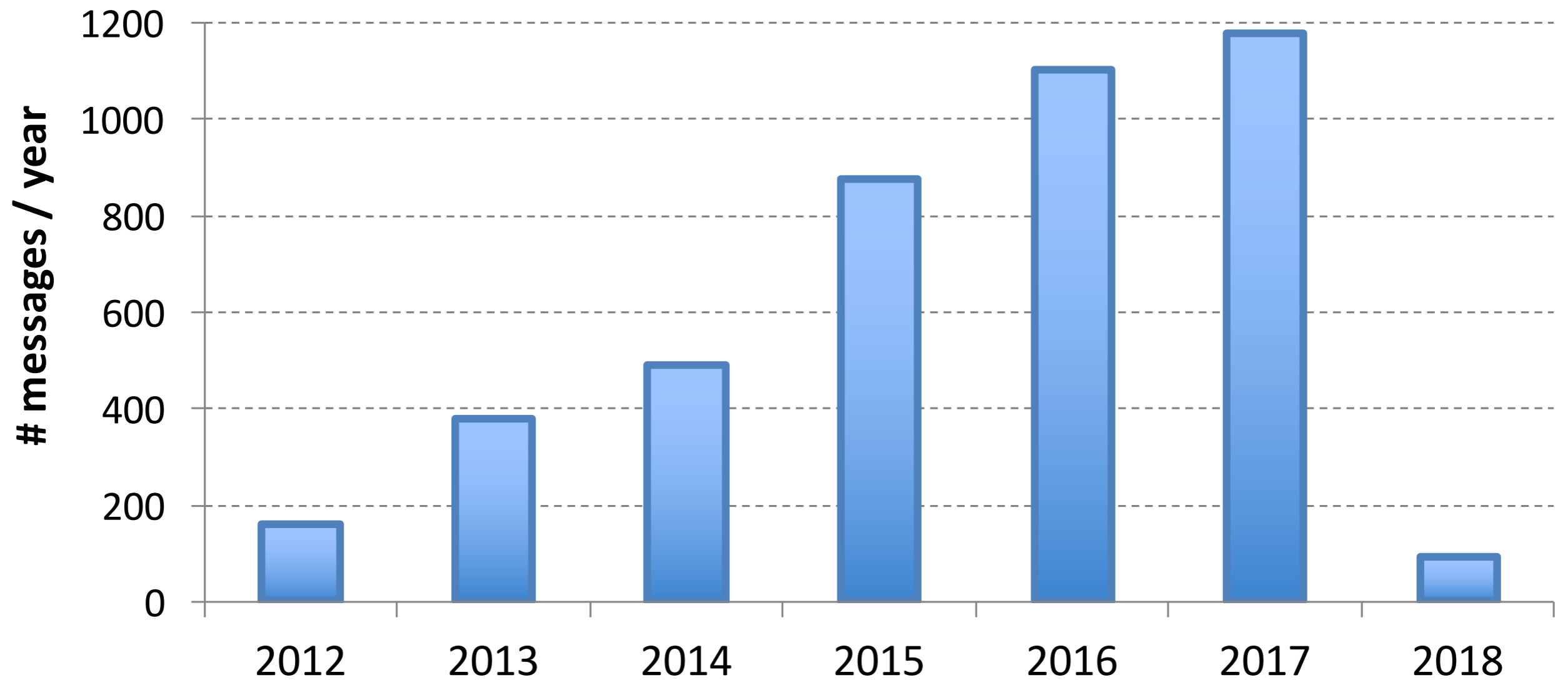
[3.00pm] (*coffee break*)

[3.30pm] hands-on session

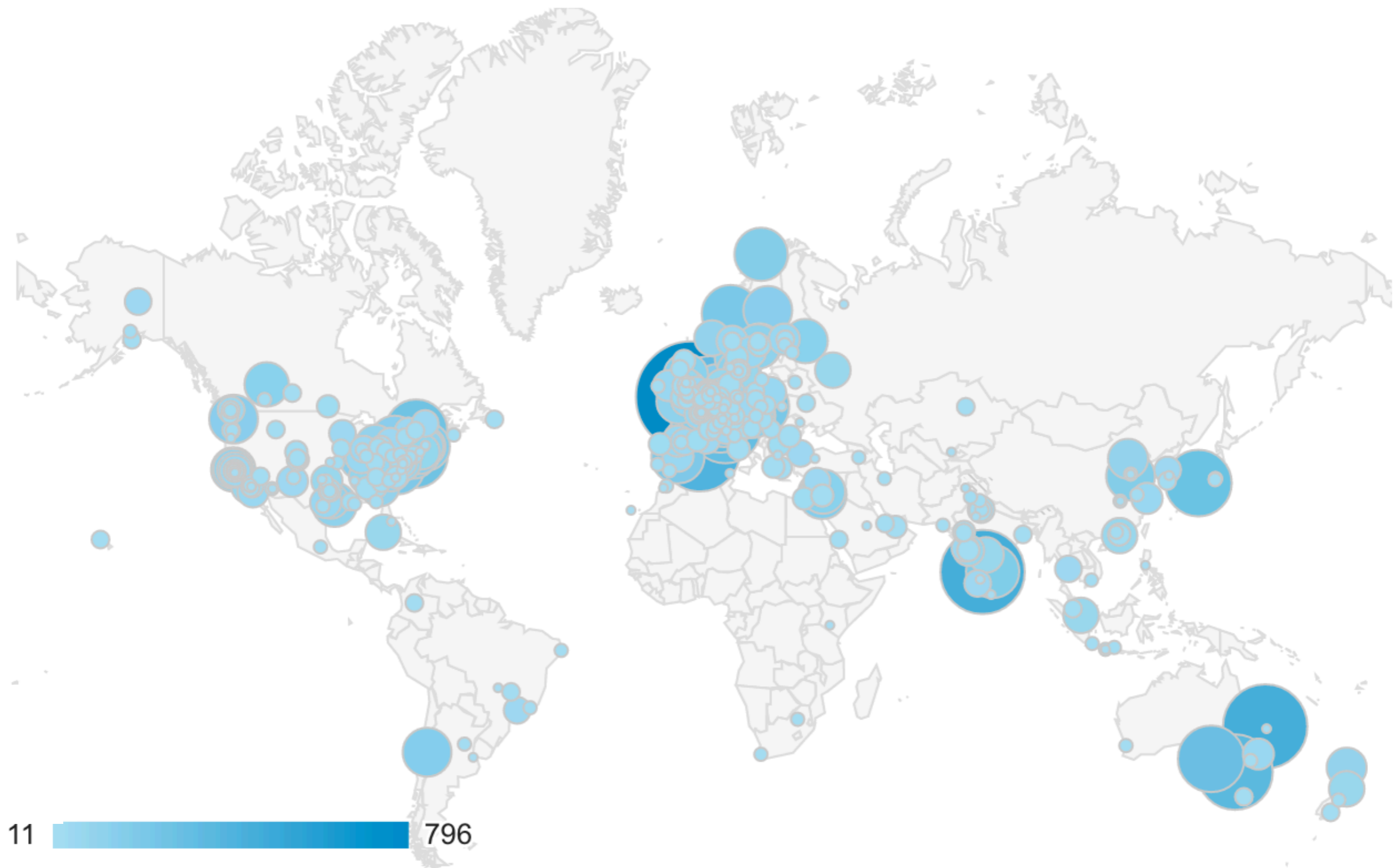
EasyBuild mailing list subscribers



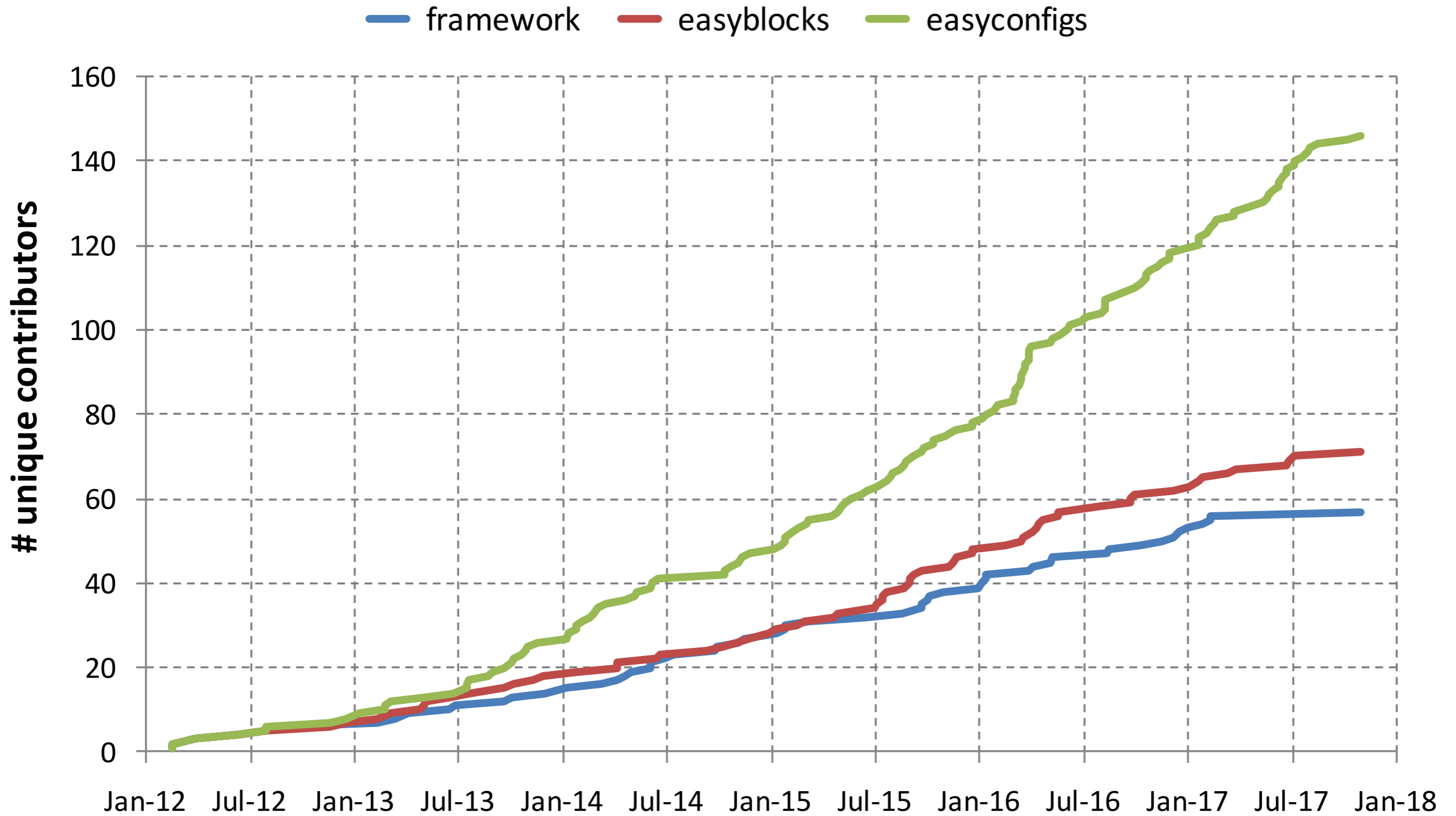
EasyBuild mailing list volume of messages



Cities from where easybuild.readthedocs.io was accessed at least 10 times in 2017



EasyBuild contributors



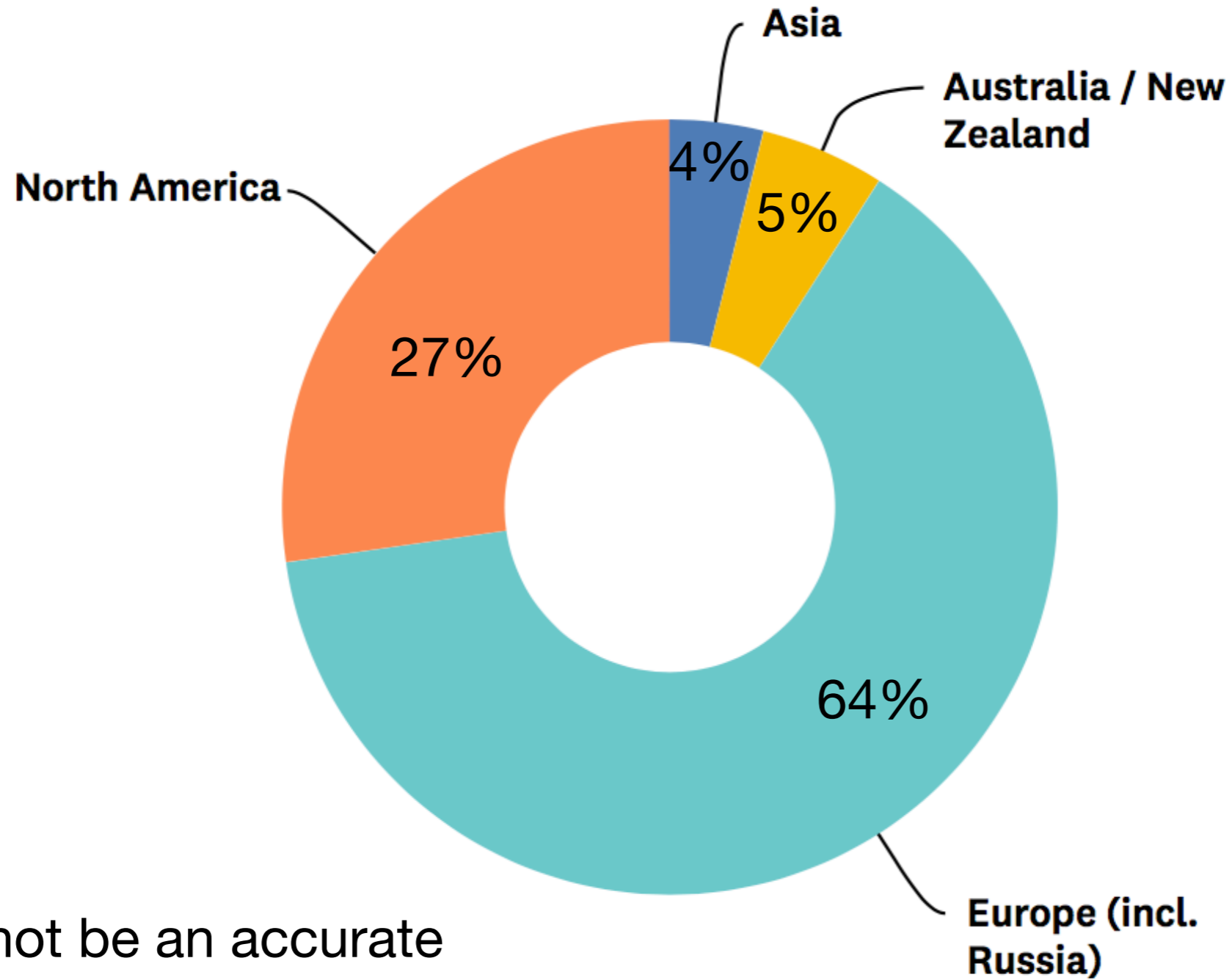


easybuild User Survey

- 77 respondents
- was open for (only) 2 weeks
- only (officially) announced via EasyBuild mailing list, to try and get *relevant* responses (actual EasyBuild users)
- aim was to try and get a better view on the community



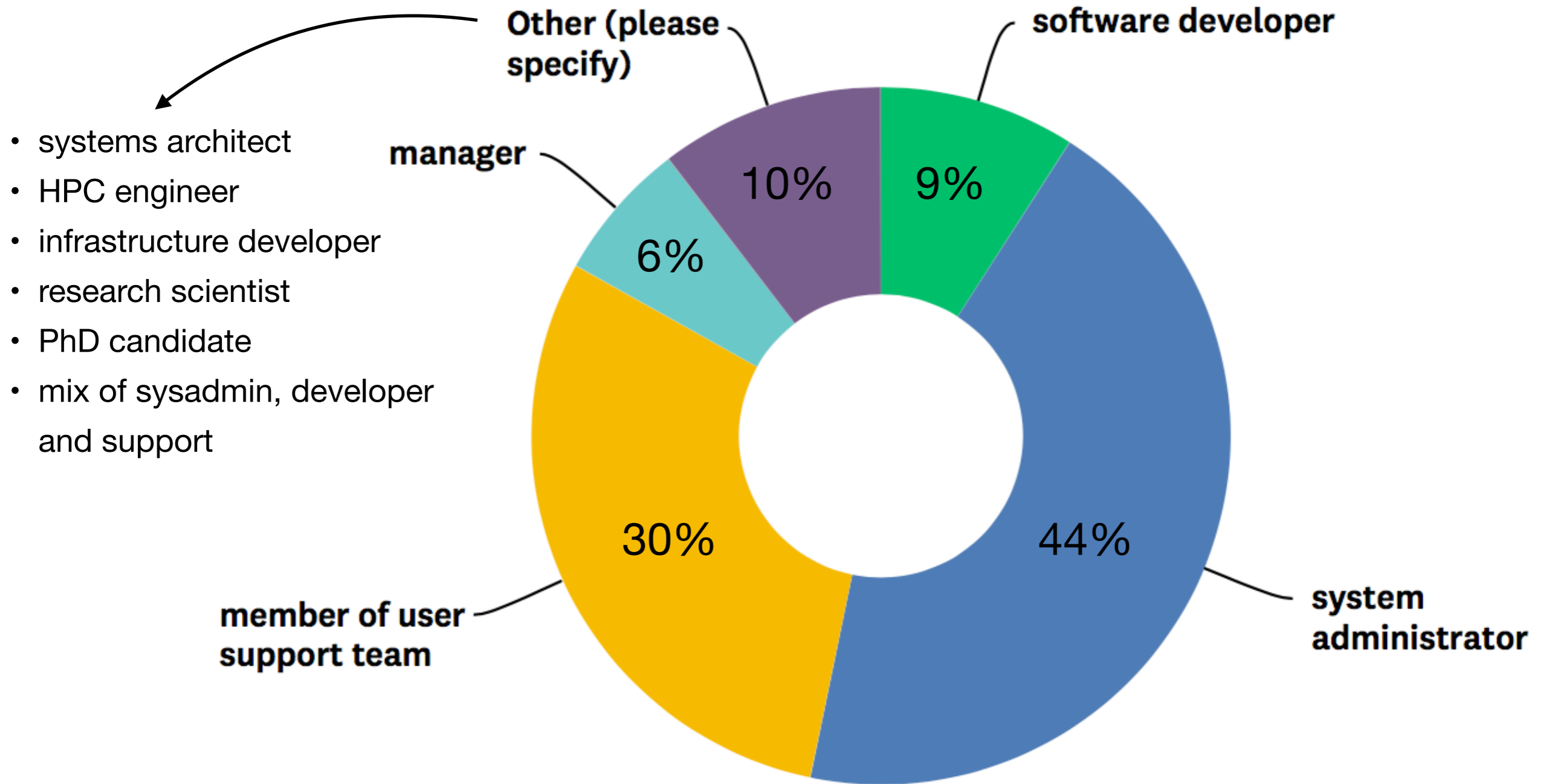
easybuild User Survey



This may not be an accurate representation of the community...

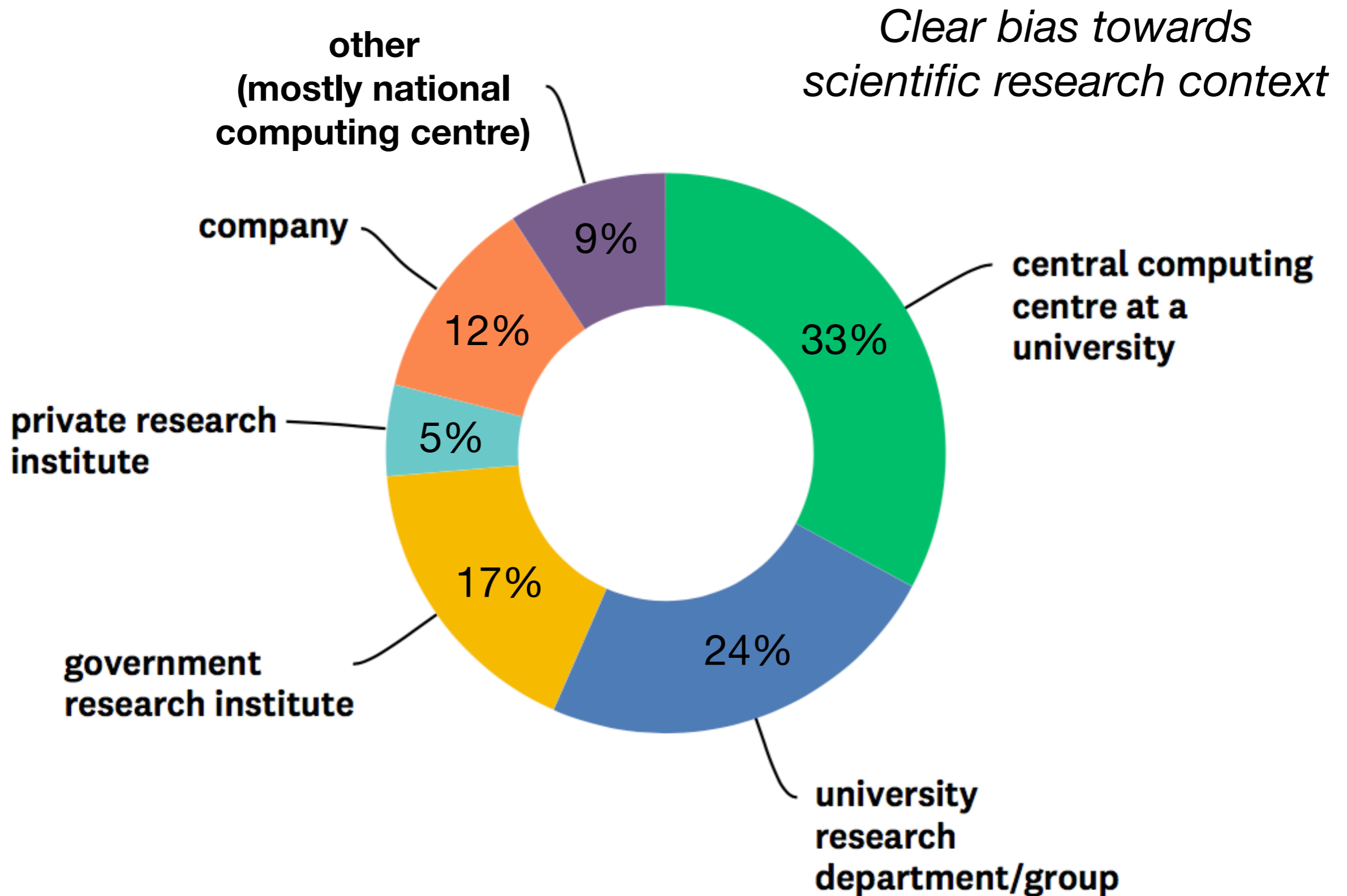


easybuild User Survey

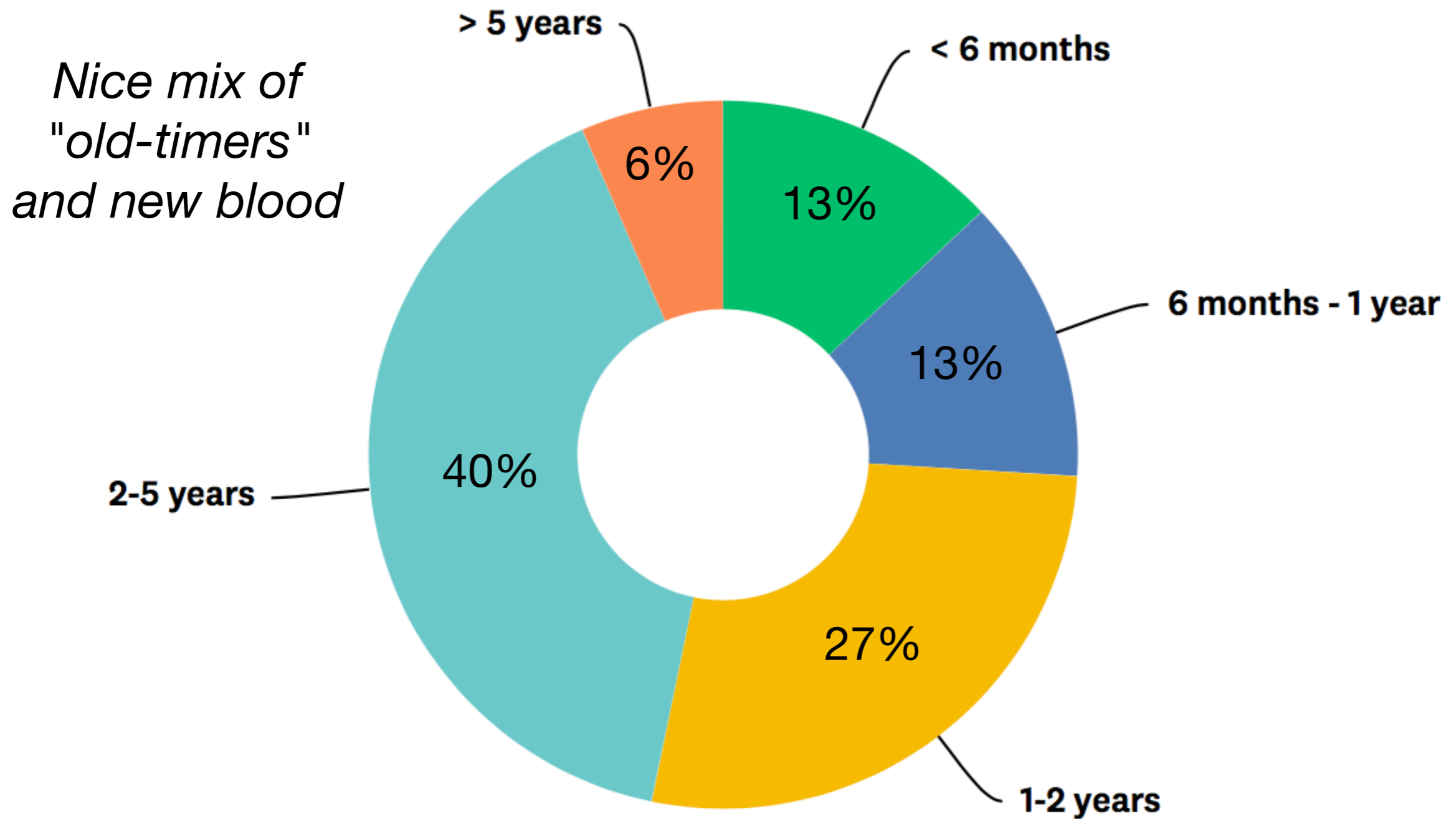




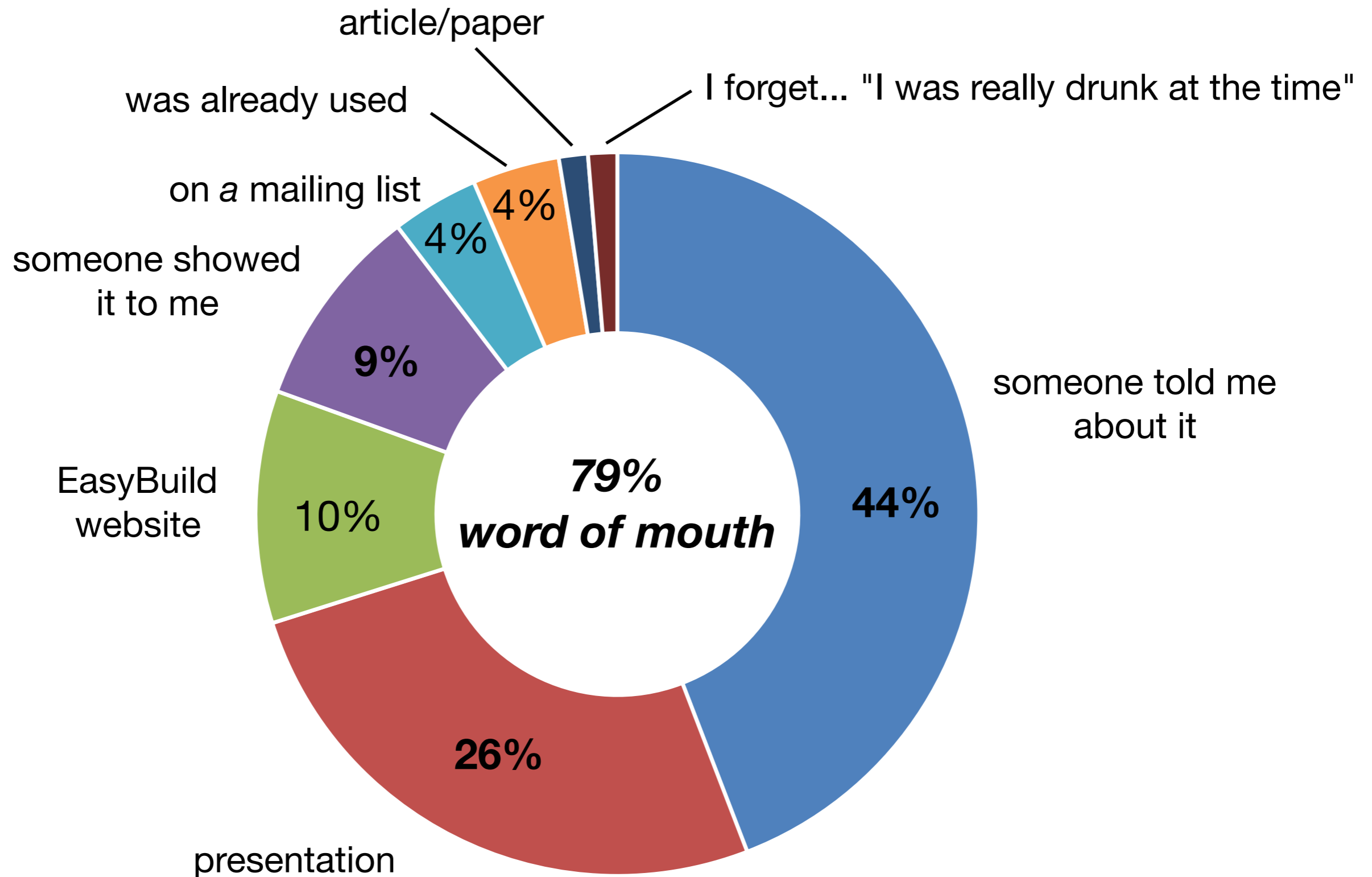
easybuild User Survey



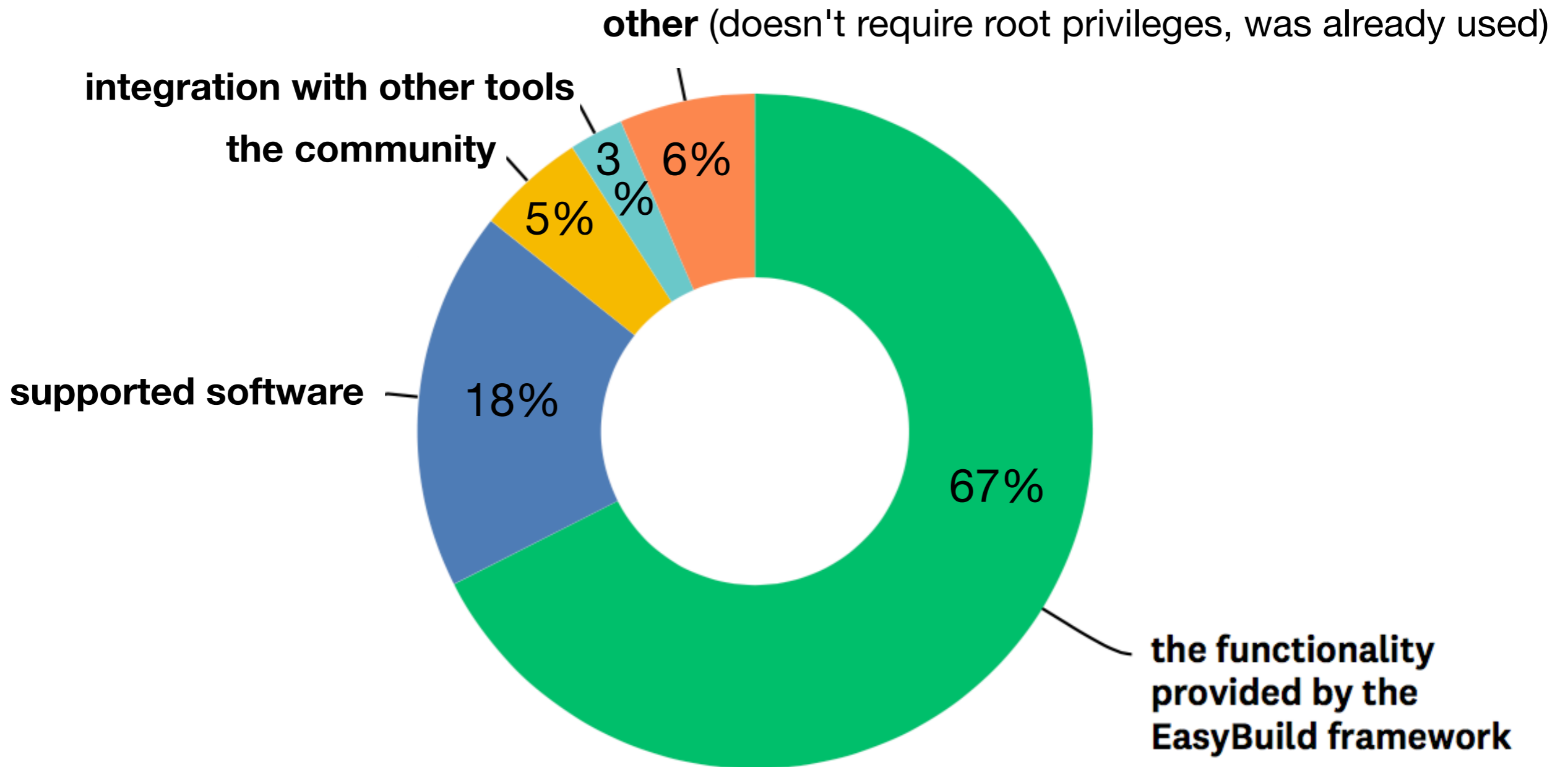
How long have you been using EasyBuild?



How did you first learn about EasyBuild?

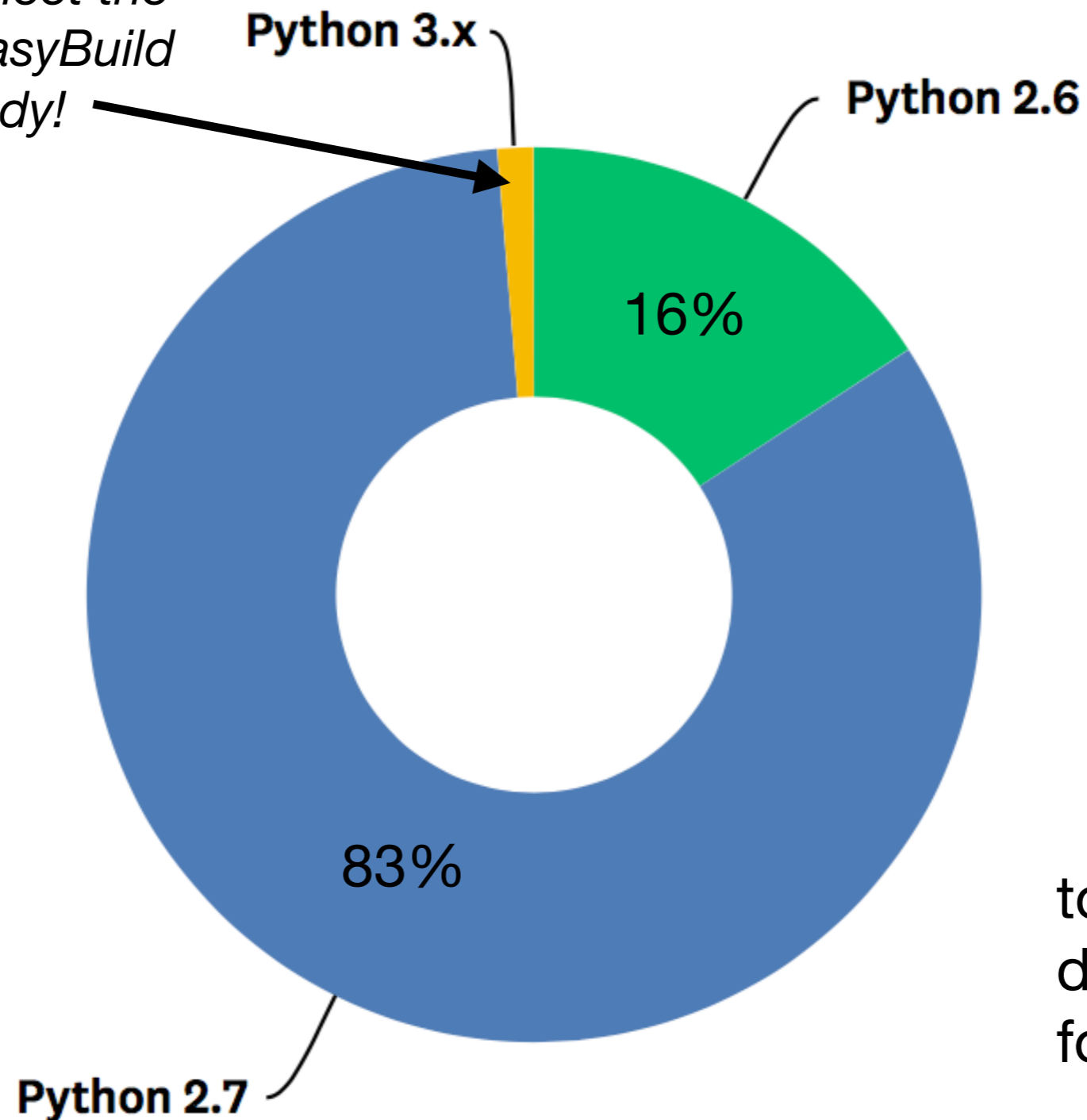


What aspect of EasyBuild convinced you?



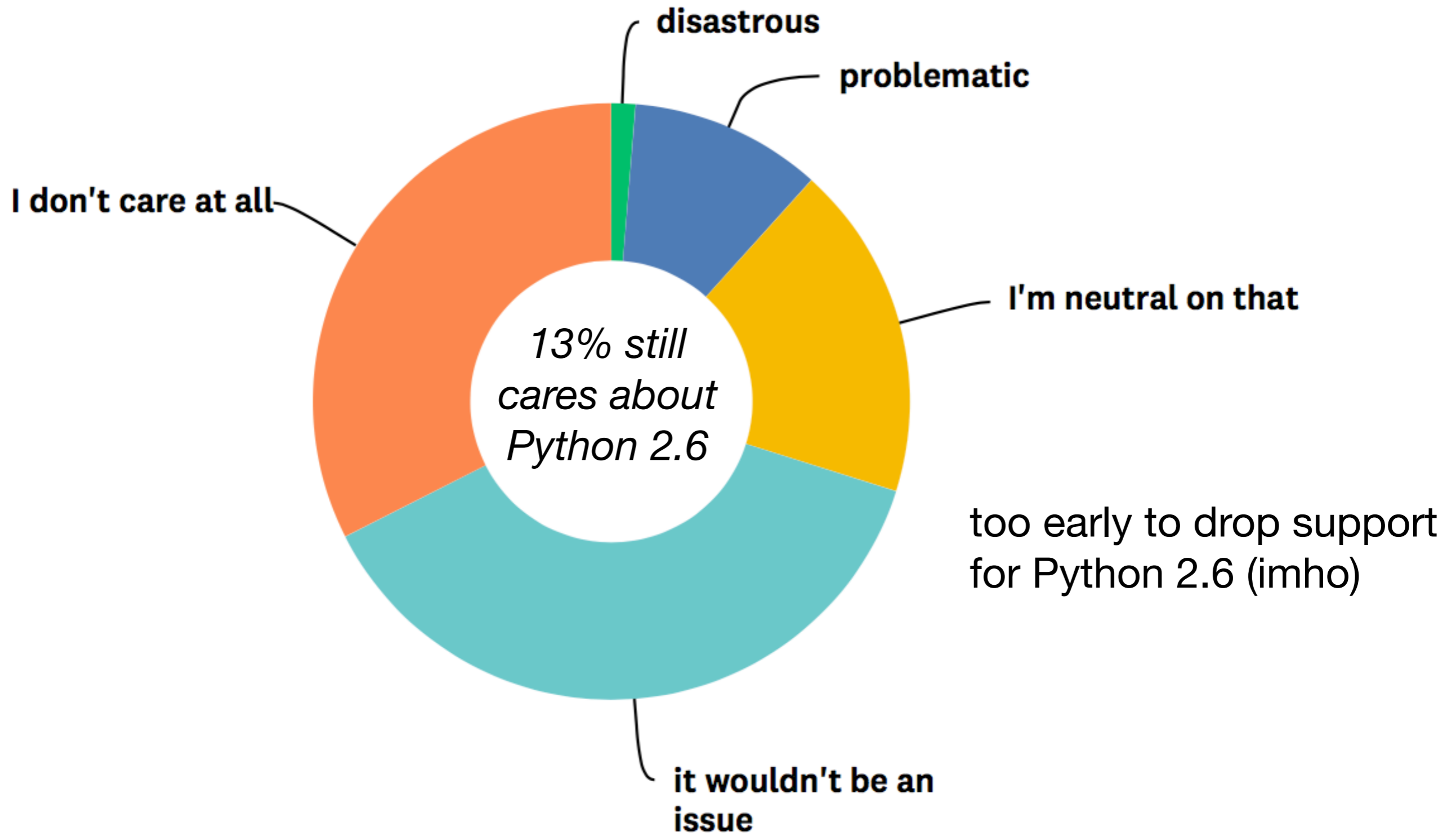
What Python version do you use to run EasyBuild?

I would really like to meet the person who ported EasyBuild to Python 3 already!

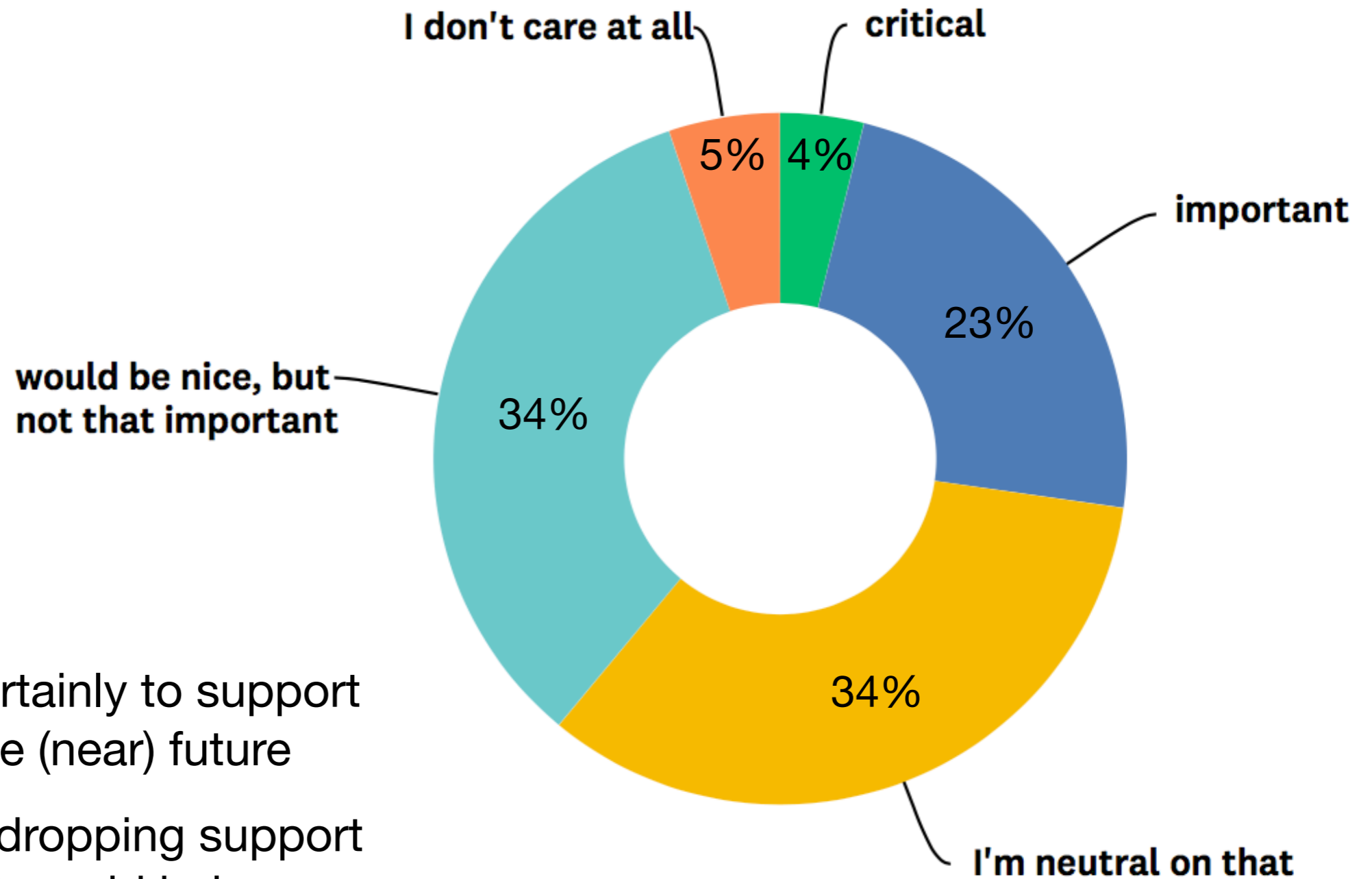


too early to drop support for Python 2.6?

How troublesome would it be for you if EasyBuild becomes incompatible with Python 2.6?

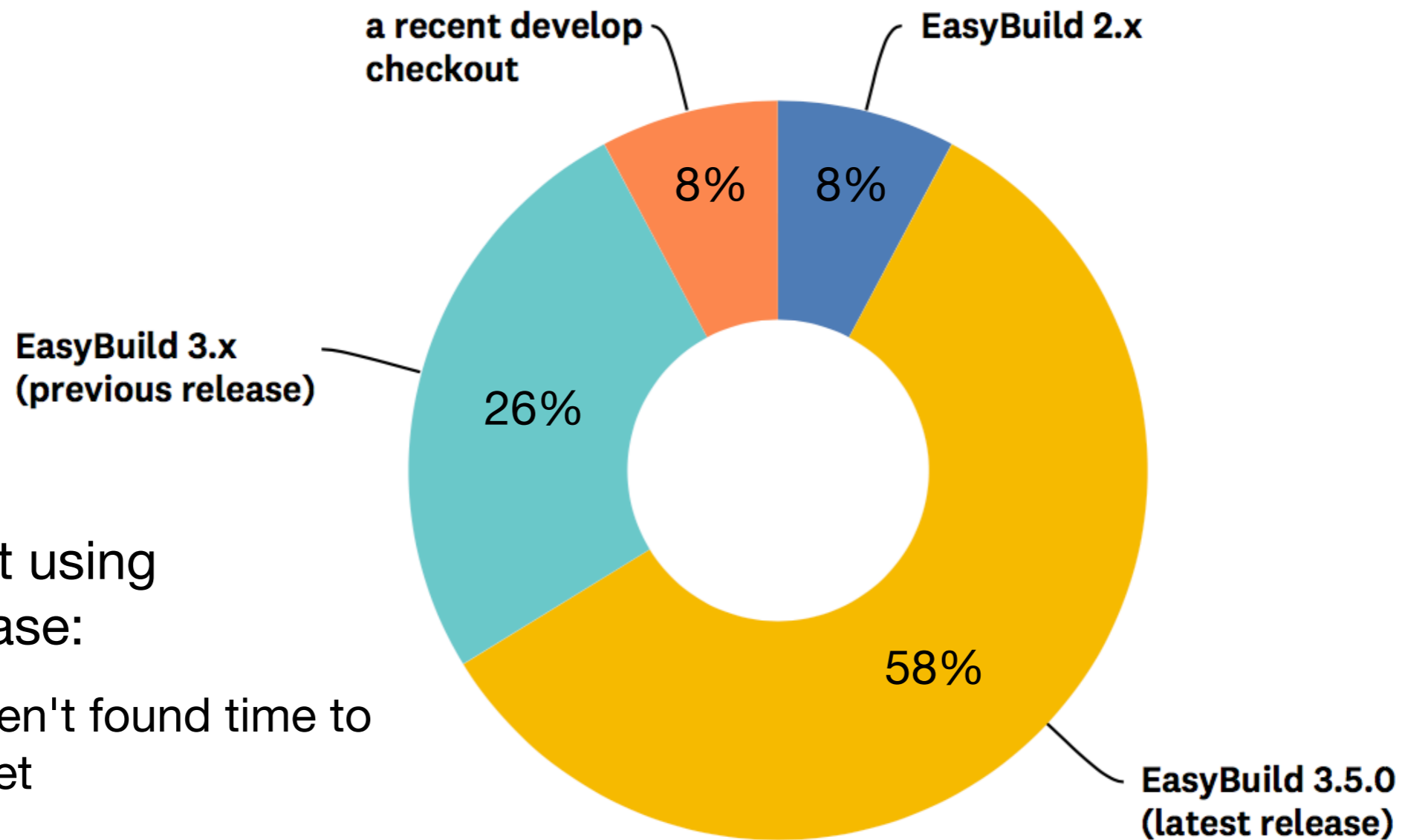


How important is it to you that EasyBuild becomes compatible with Python 3?



- intention is certainly to support Python 3 in the (near) future
- being able to dropping support for Python 2.6 would help
- vsc-base needs to be ported first...

Which EasyBuild version do you use?

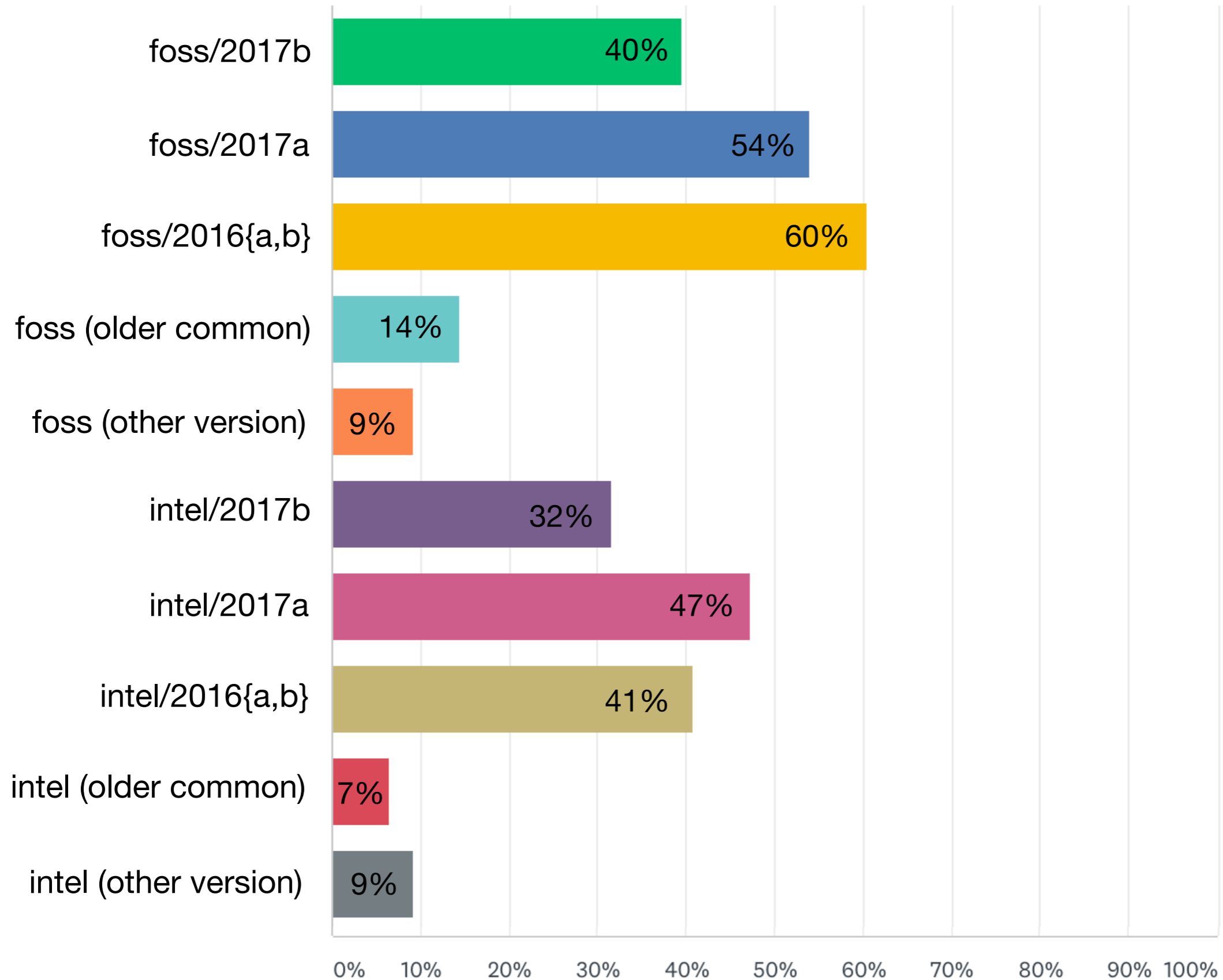


People not using latest release:

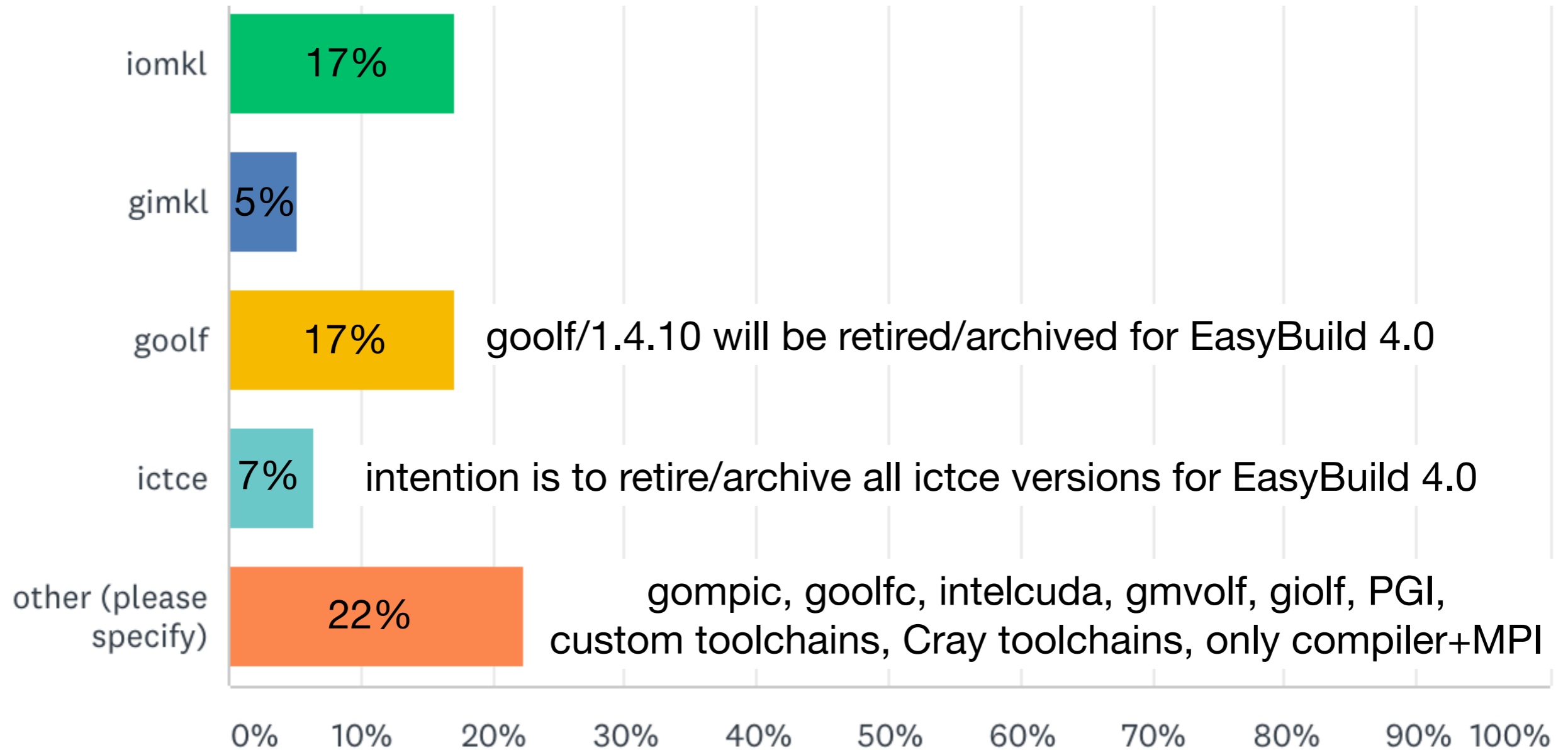
- 50% haven't found time to update yet
- 37% say current version works fine
- 10% prefer living on the edge using `develop`
- for 3% update failed (1 person)

High-quality releases are really important!


Common toolchains are used actively



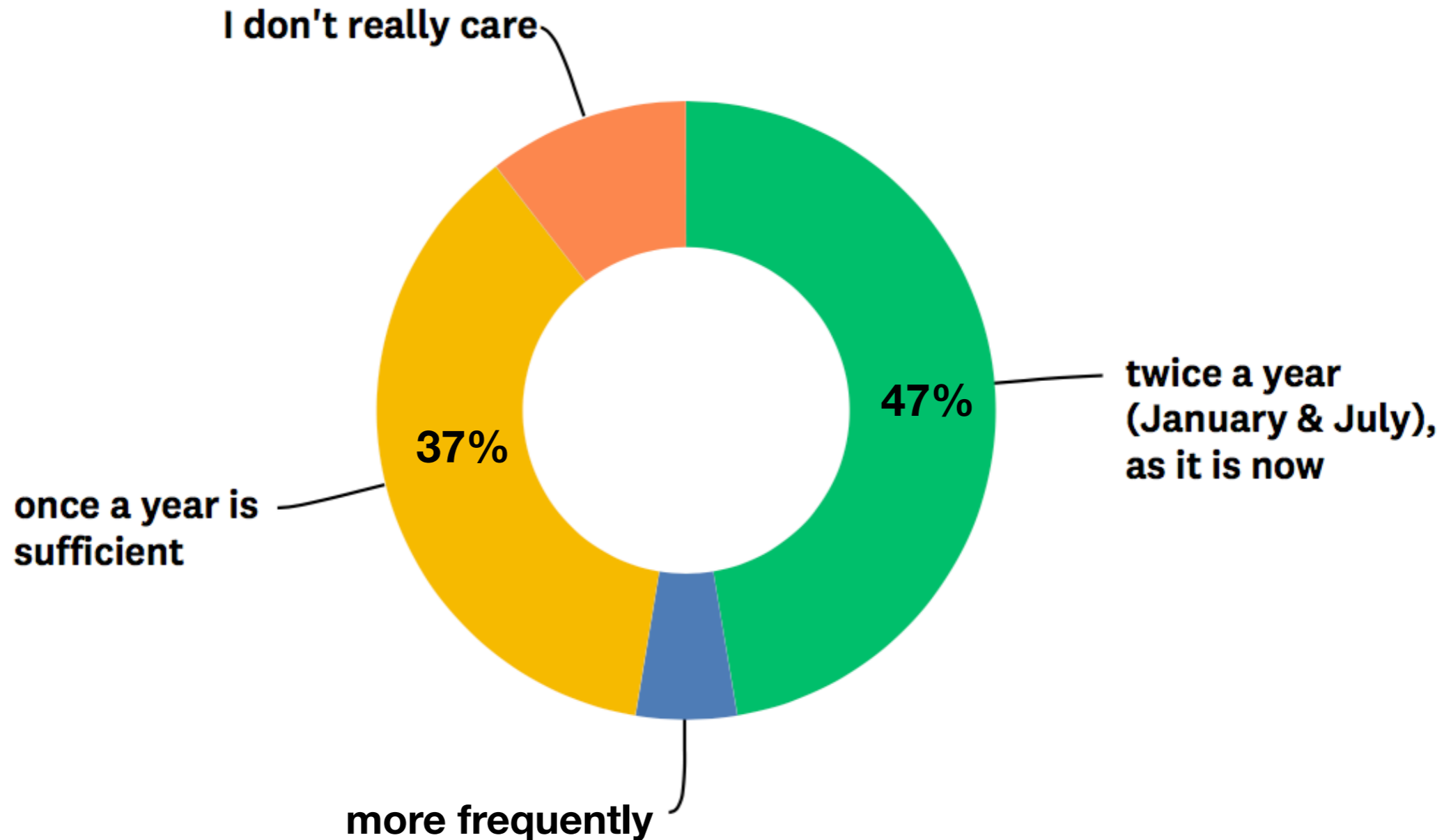
Other toolchains are less commonly used



Side step: toolchain woes...

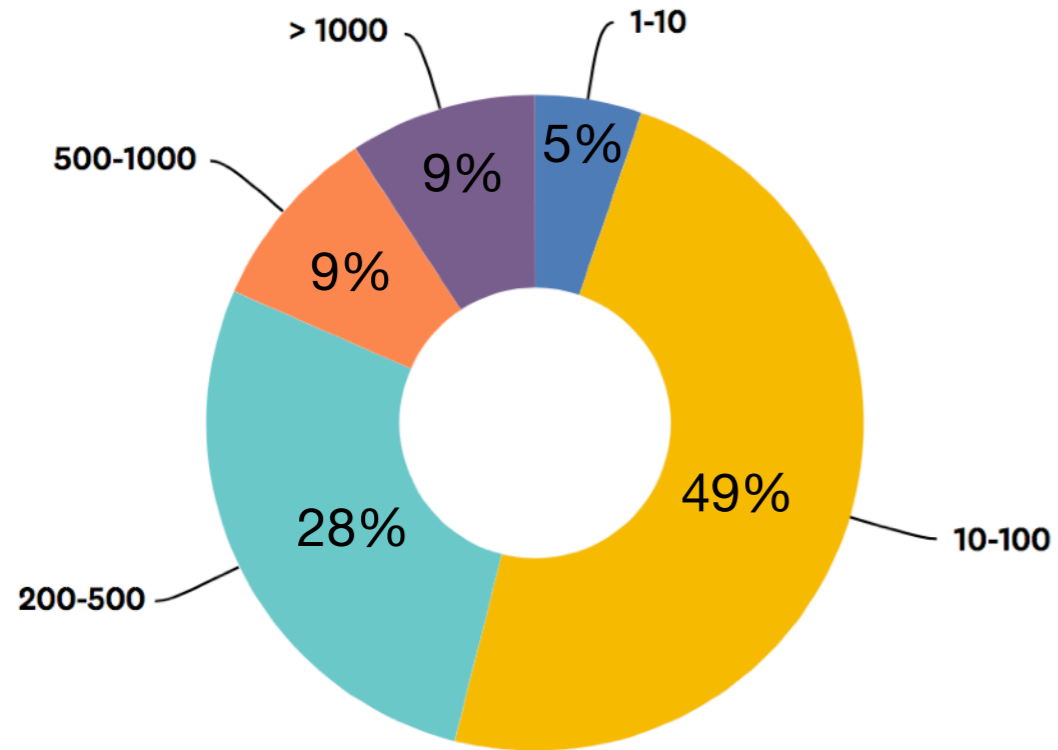
- Is the `intel` toolchain *really* worth all the trouble?
 - compilation issues due to picky/quirky compilers
 - runtime issues (see recent problems related to `glibc`)
 - licensing cost
 - is there actually a (significant) performance gain over GCC?
- some concern about sustainability of OpenBLAS
 - v0.2.21 planned for Aug'17 never actually happened
 - no releases since then...
- time to promote `gimkl` or `gomkl` to common toolchain?
 - 
 - (GCC + Intel MPI + Intel MKL)
 - (GCC + Open MPI + Intel MKL)

How frequently should the common toolchains be updated?

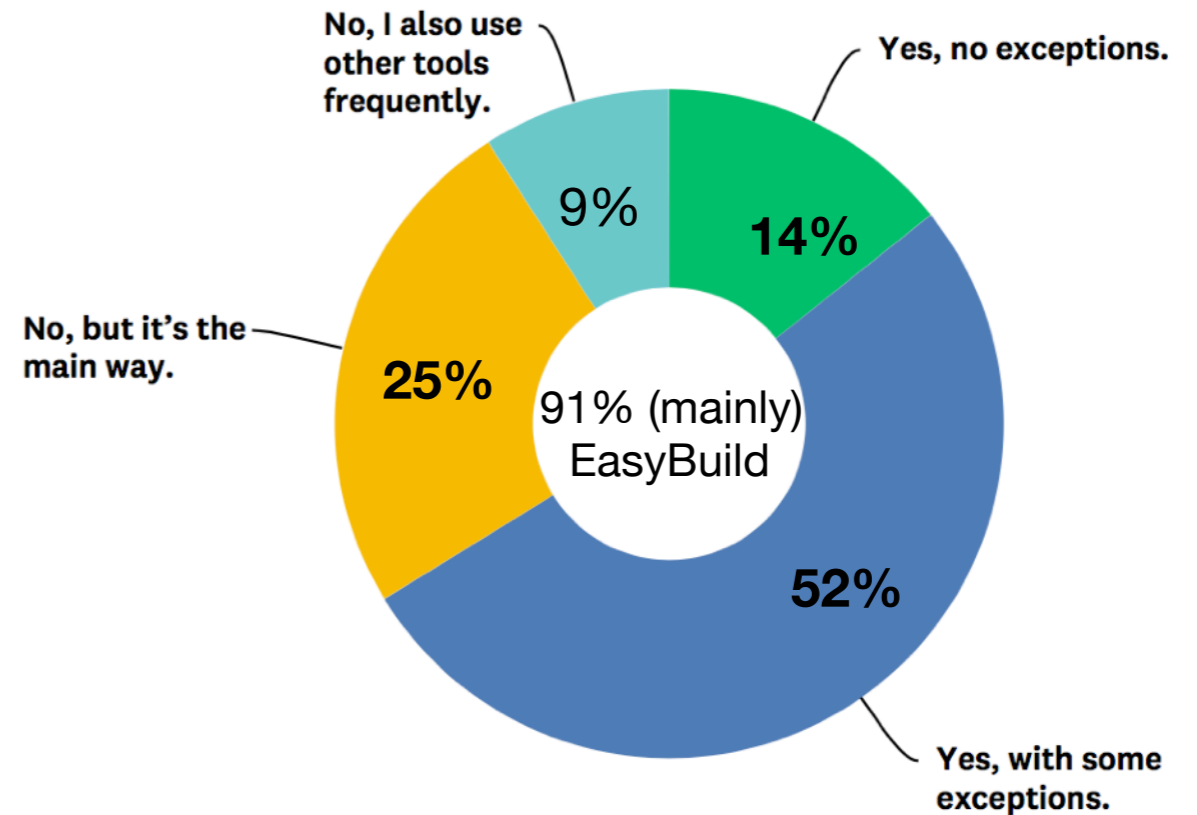


Conclusion: stick to twice a year?
(people can ignore common toolchain updates once a year)

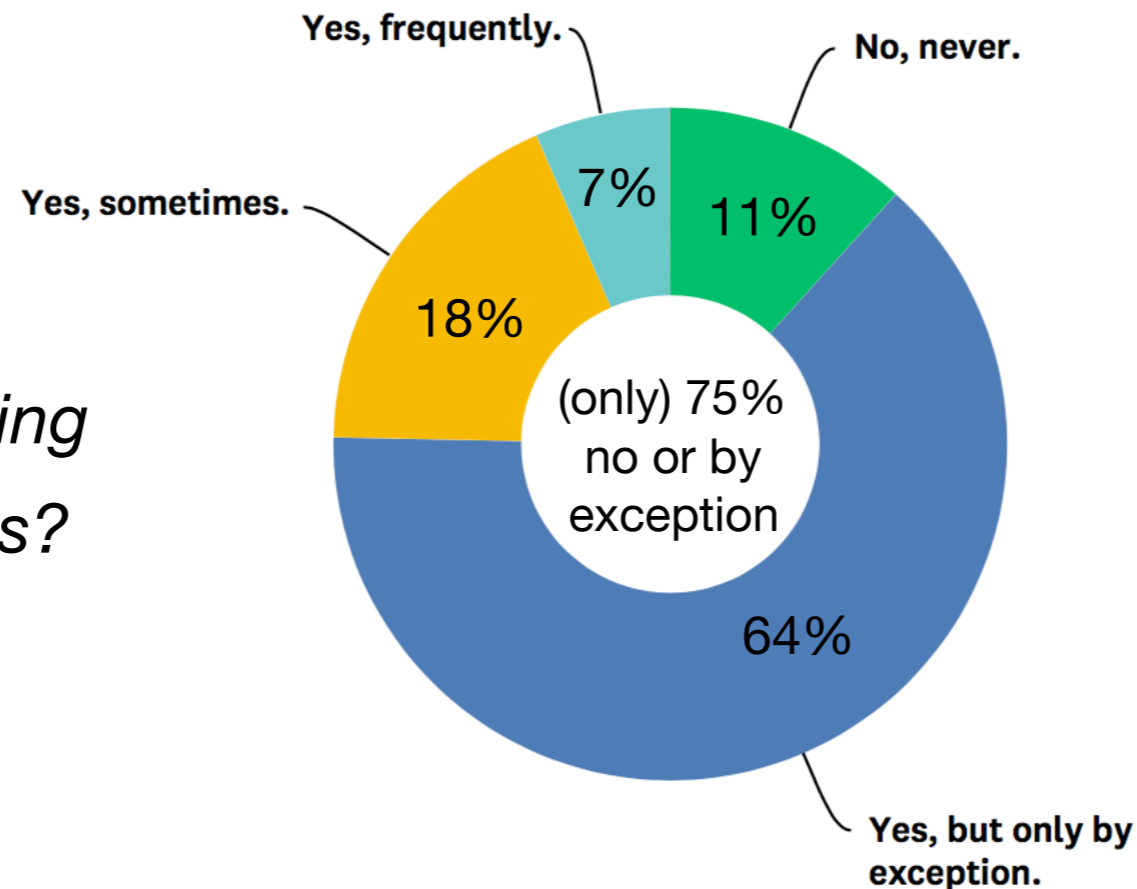
software installations in last year



Do you only install software with EasyBuild?

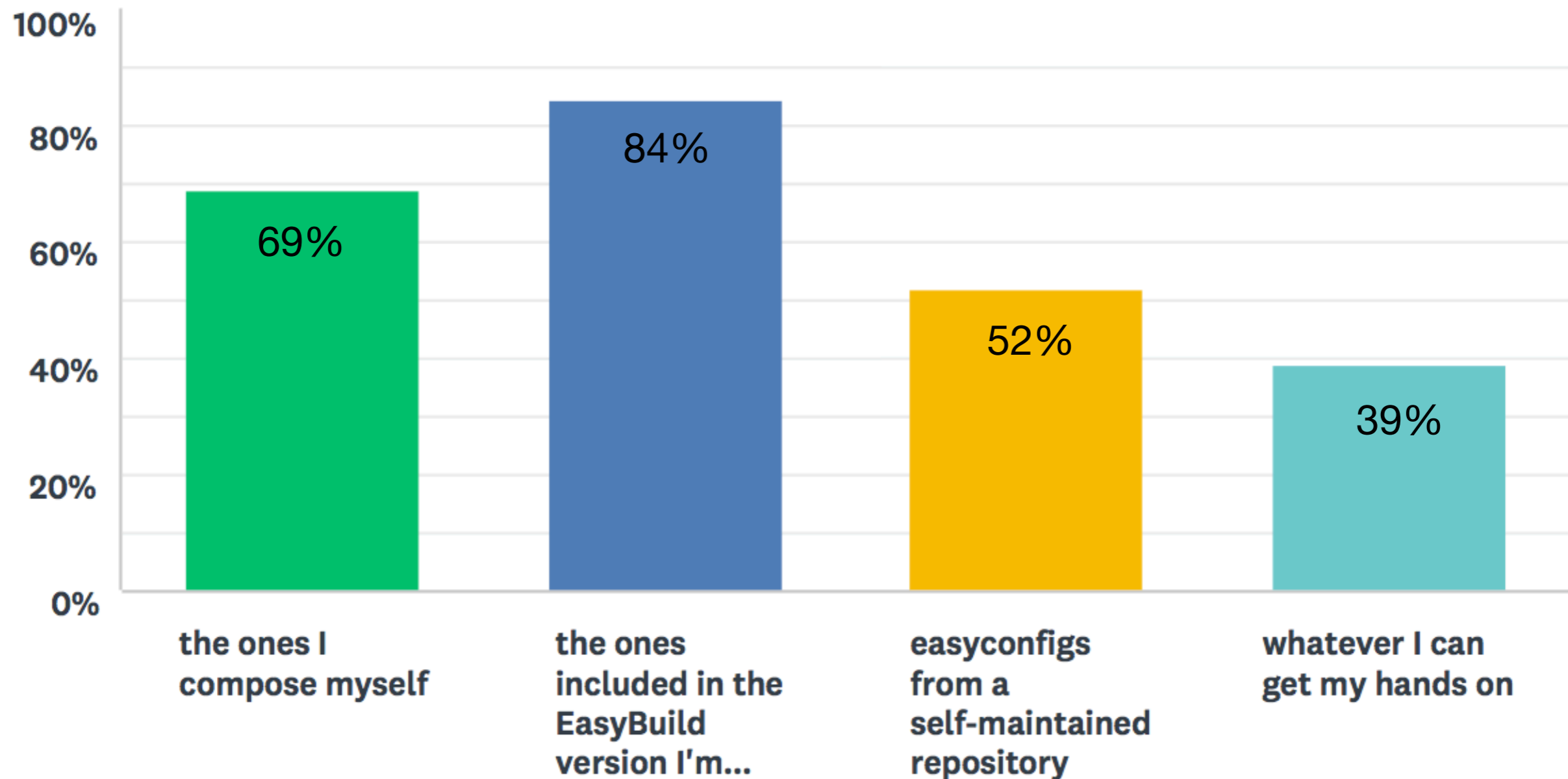


Do you still install software manually?



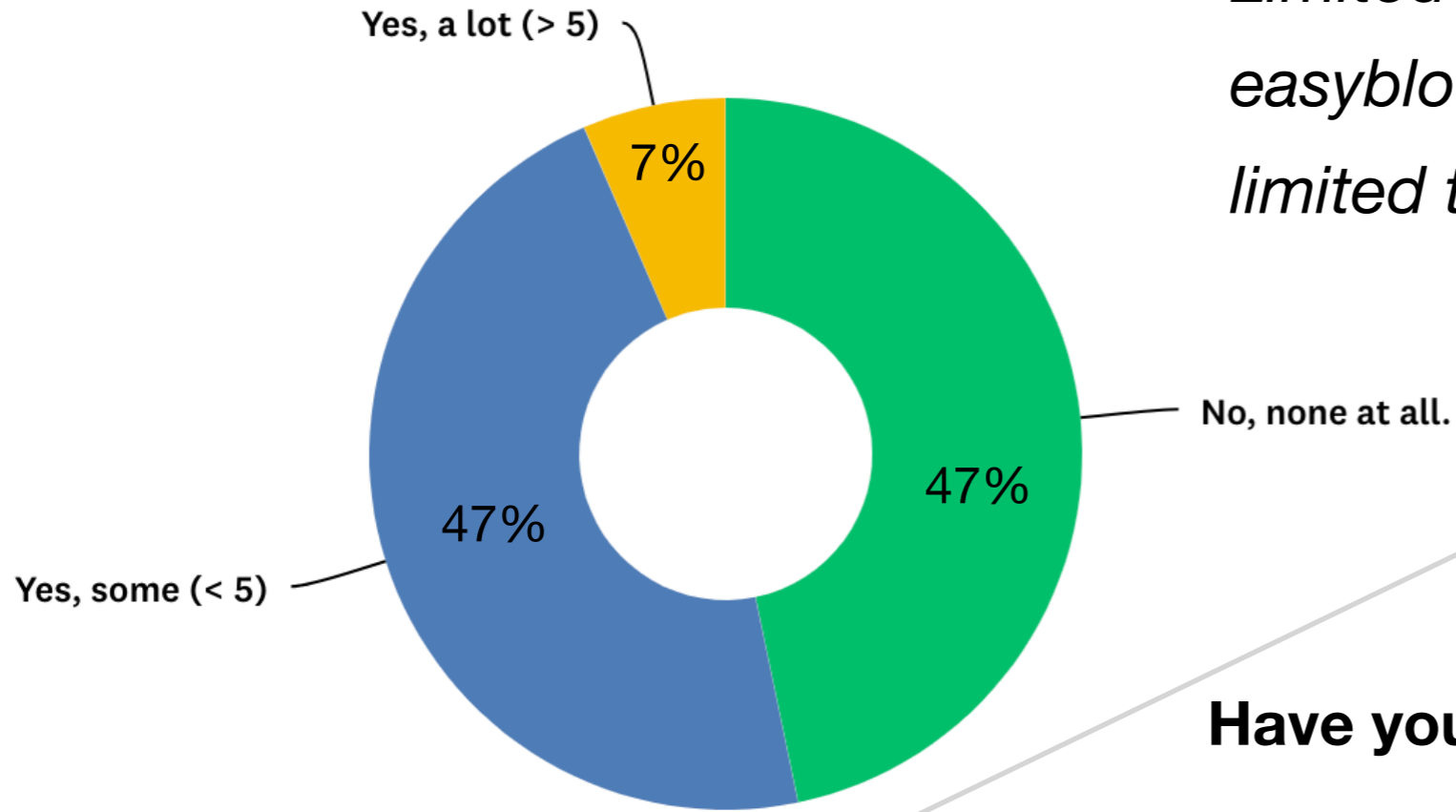
Reasons for still doing manual installations?

Which easyconfig files do you use?



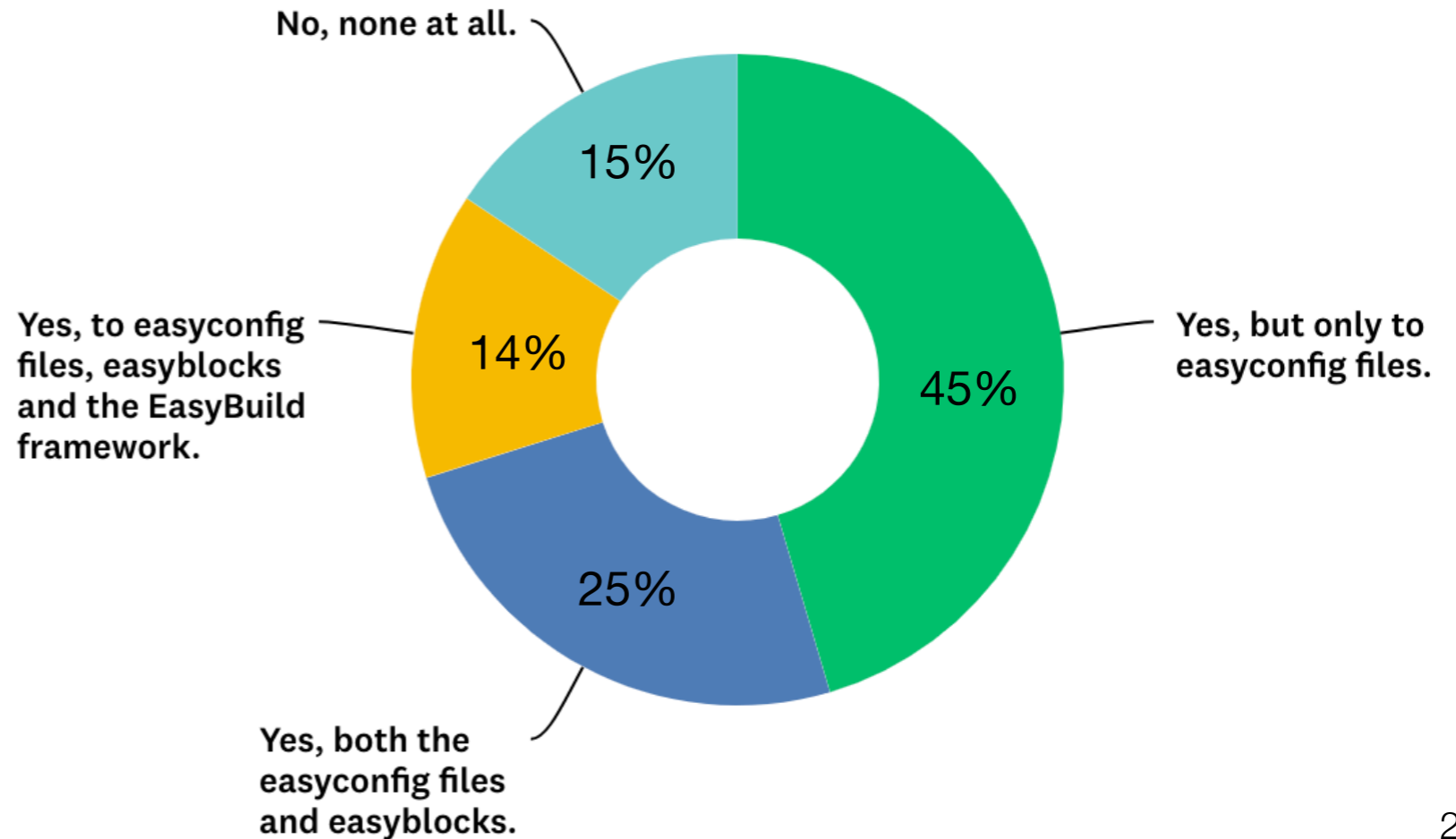
- easyconfigs included with EasyBuild should be up-to-date & high-quality...
- quite a lot of people have their own easyconfigs repository

Do you use custom easyblocks?



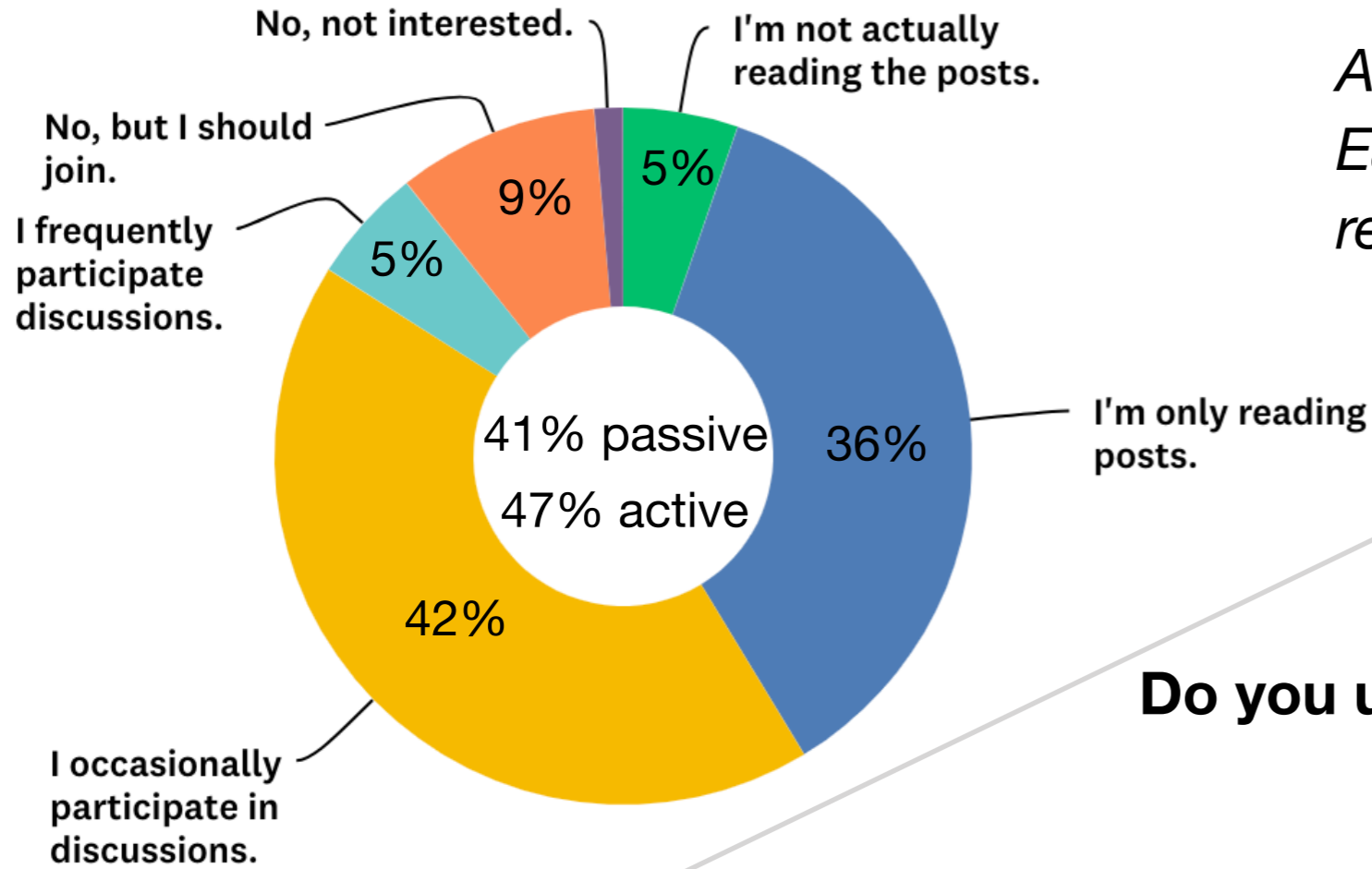
Limited use of custom easyblocks, probably mostly limited to licensed software?

Have you made site-specific customisations?



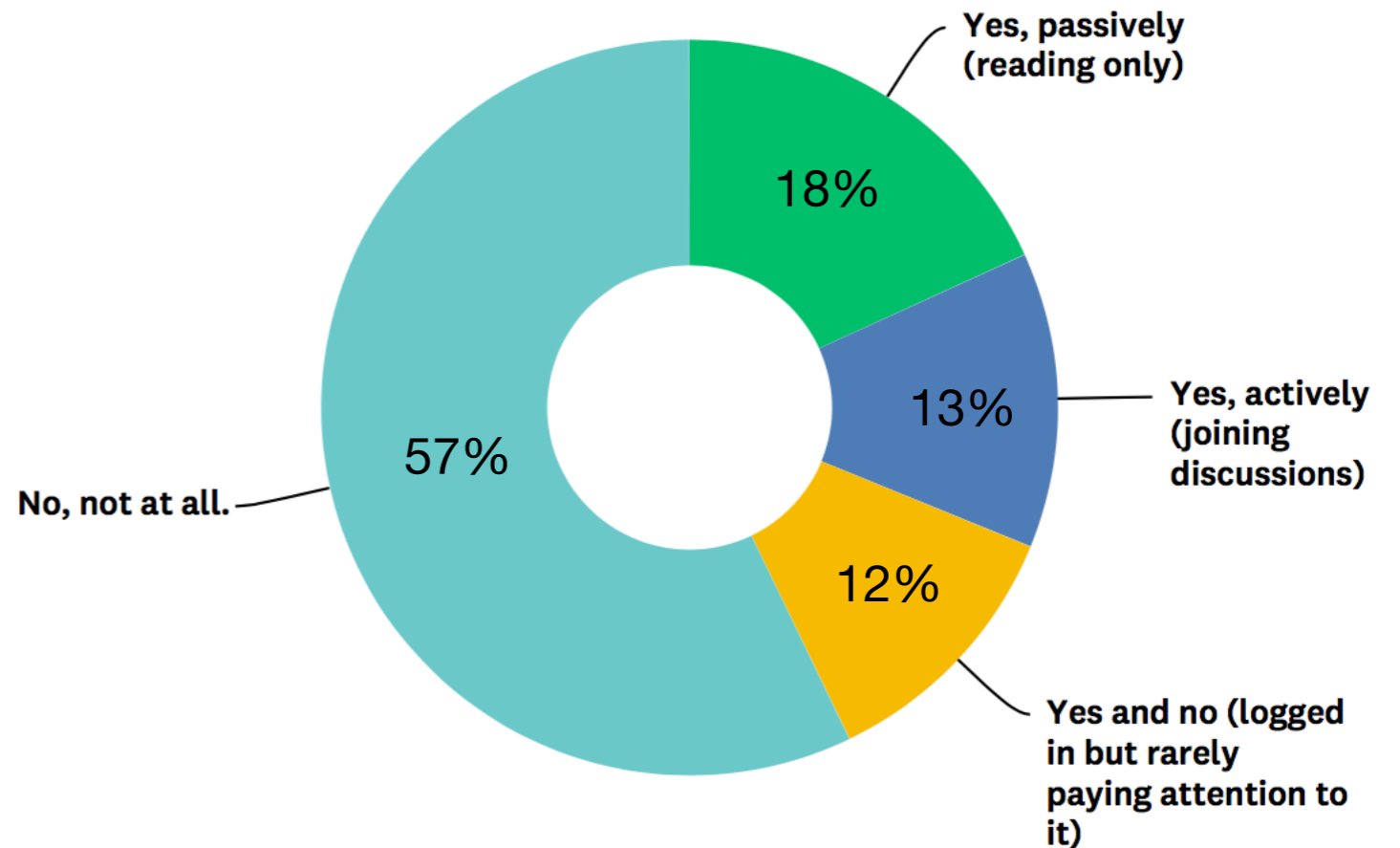
Better support needed for making site-specific customisations easier?

Are you subscribed to the EasyBuild mailing list?



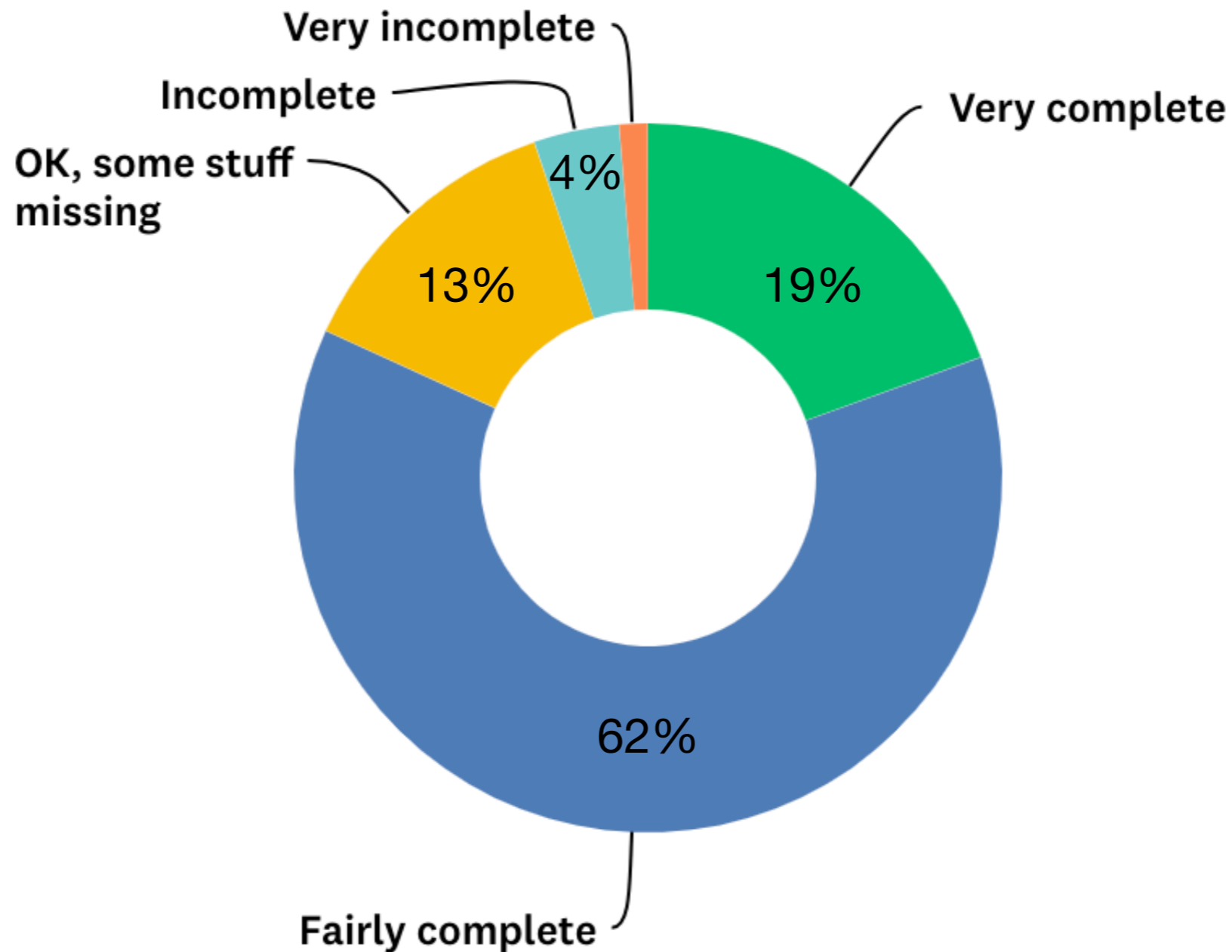
A significant part of the EasyBuild community remains mostly silent.

Do you use the EasyBuild IRC/Slack channel?



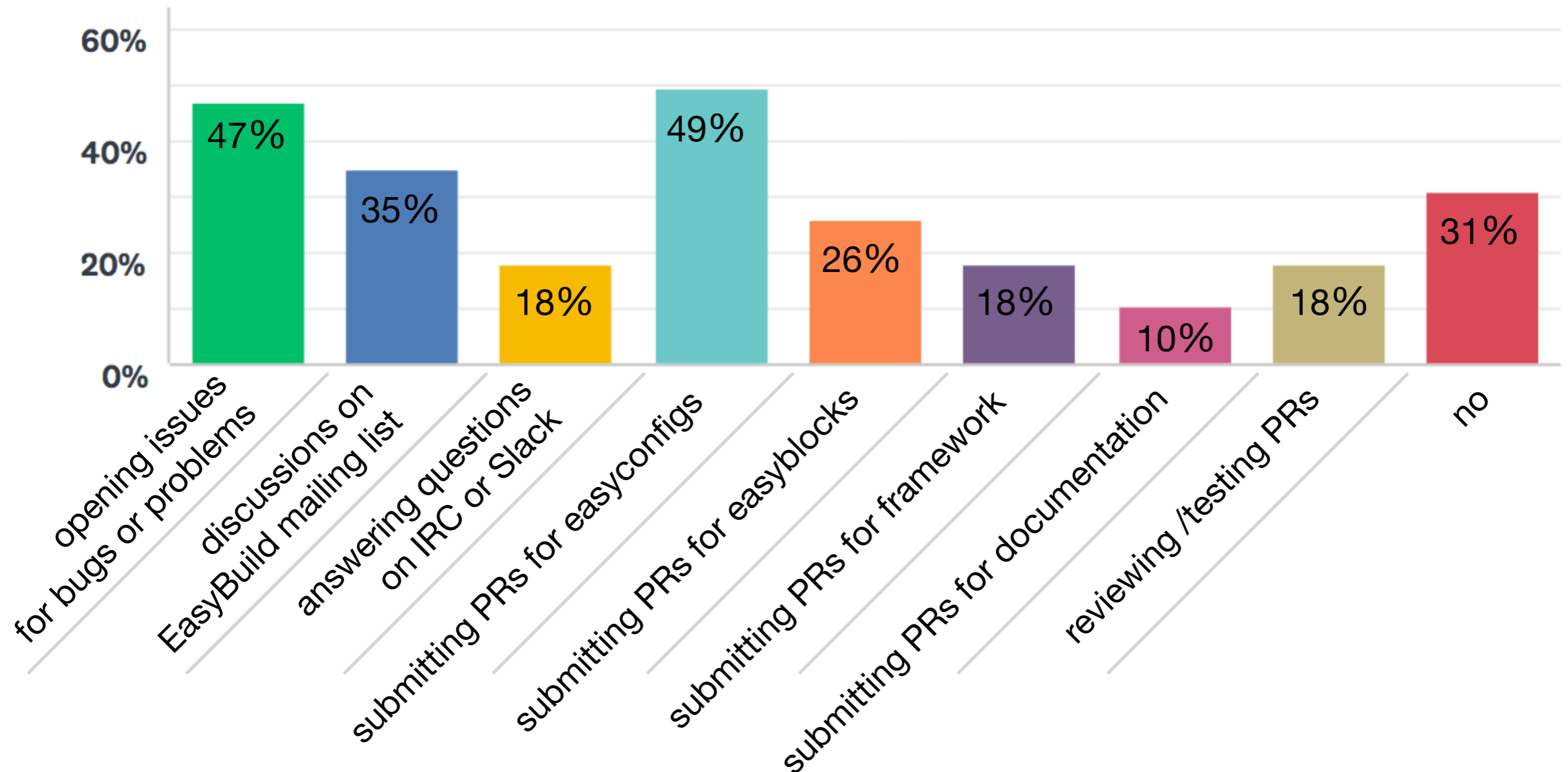
Only a small minority actively engages with the EasyBuild community through a chat platform, so not the right place to make any 'big' decisions.

How complete is the EasyBuild documentation?



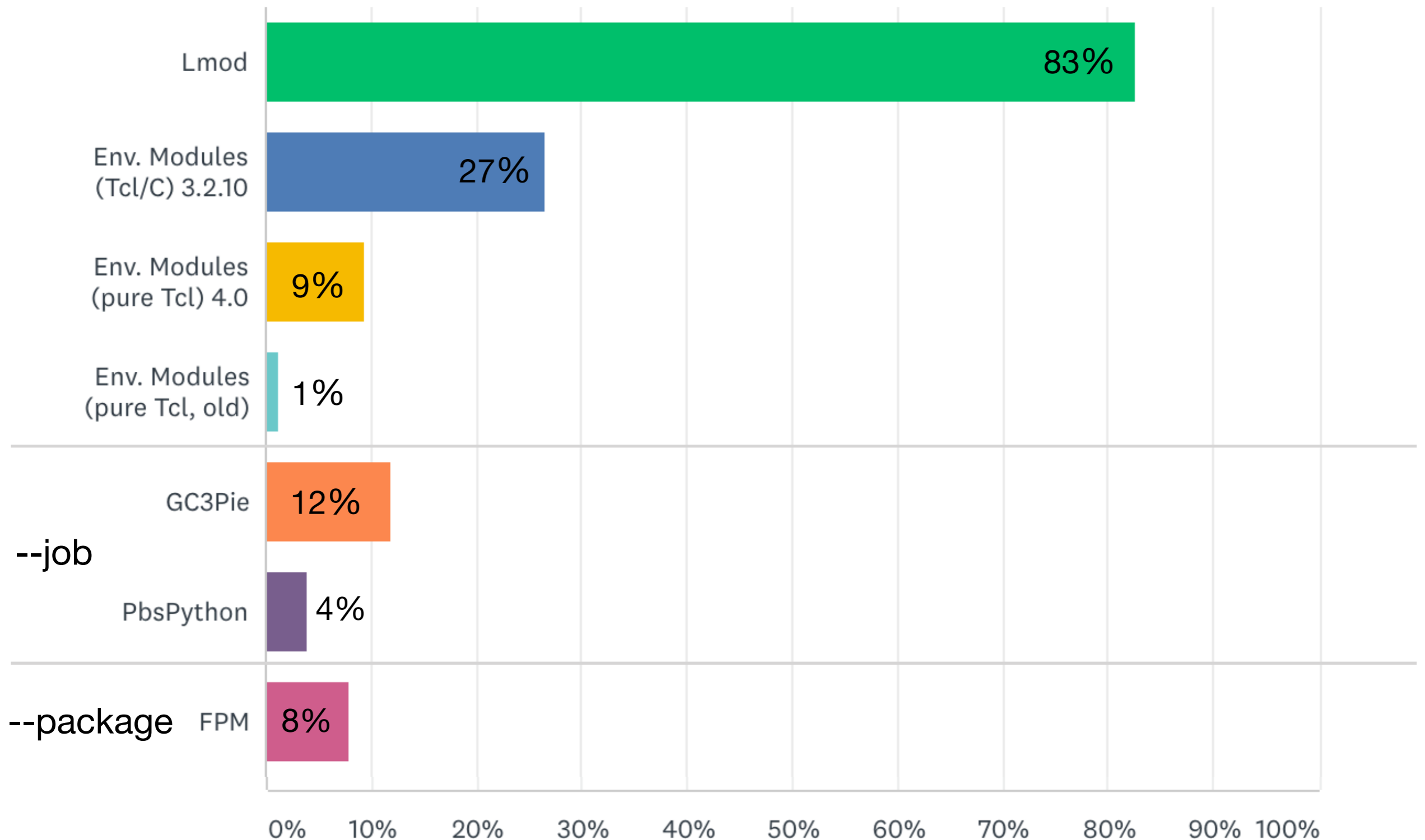
documentation is good enough for (large) majority of people?
(I disagree, some important stuff is (still) not documented yet)

Do you actively contribute back to EasyBuild?

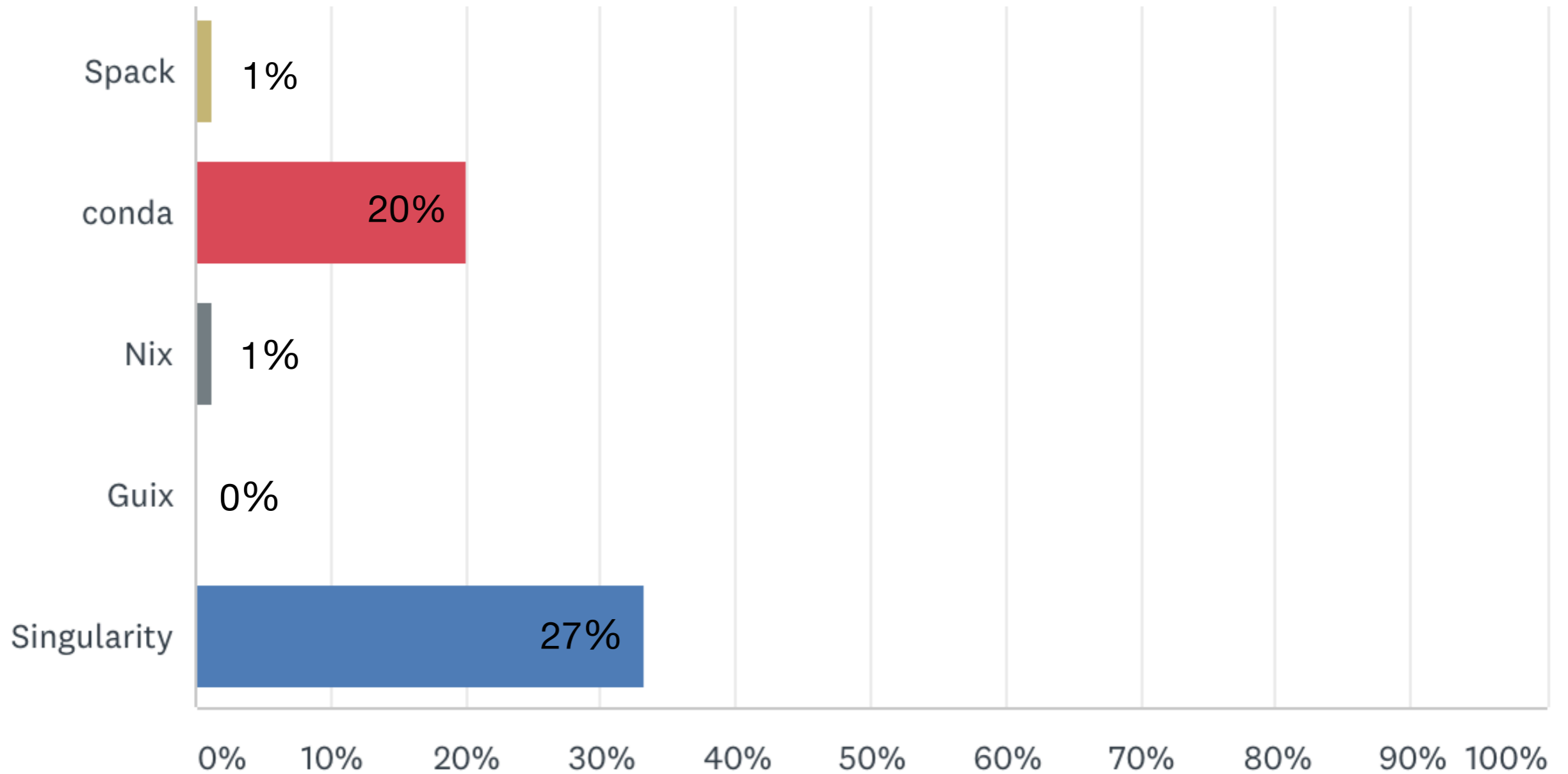


- almost 50% contributes easyconfig files, opens issues for bugs/problems
- only 10% contributes to documentation...
- 31% does not contribute back at all

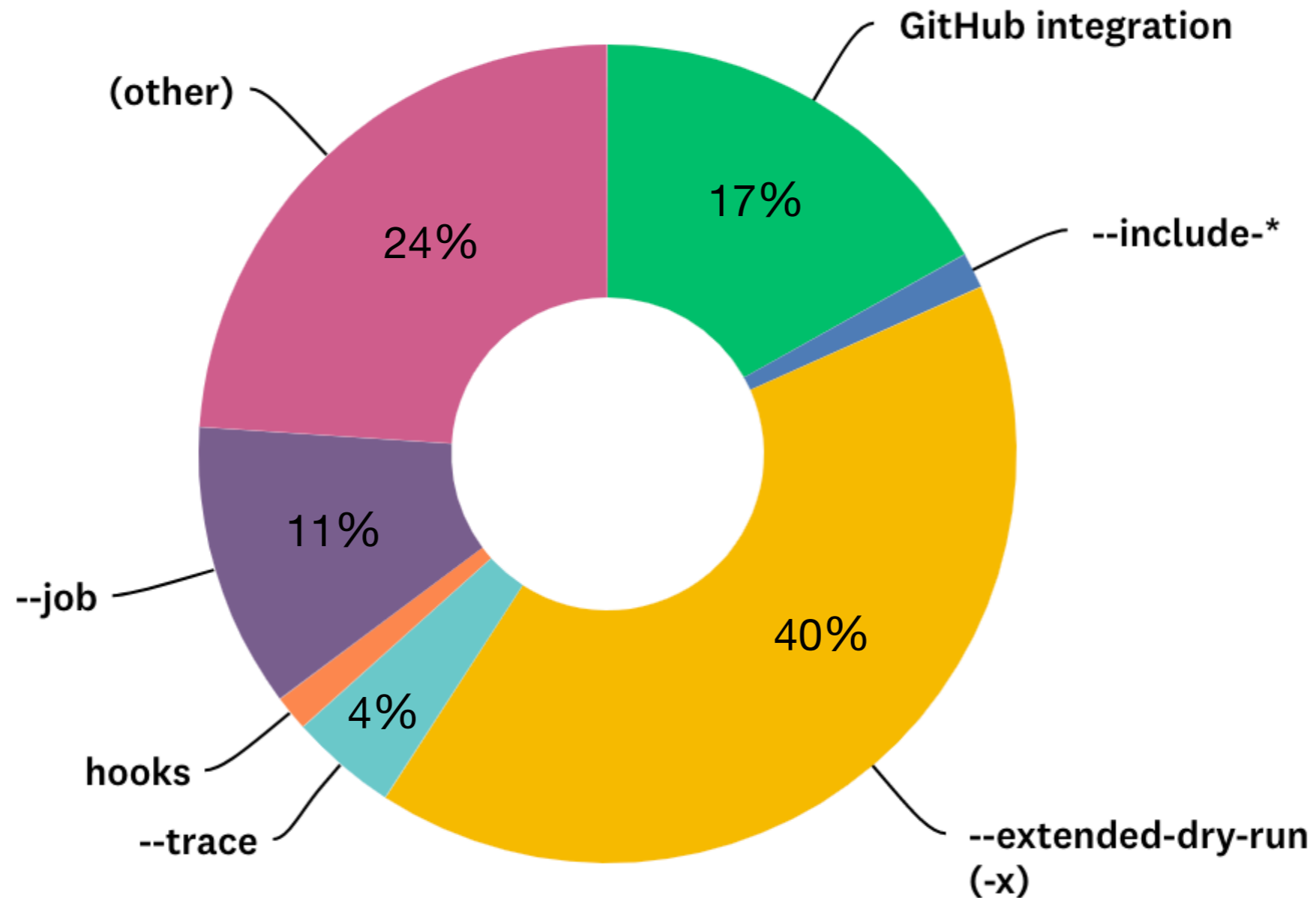
Do you use any other tools in combination with EasyBuild?



Do you use any other tools in combination with EasyBuild?



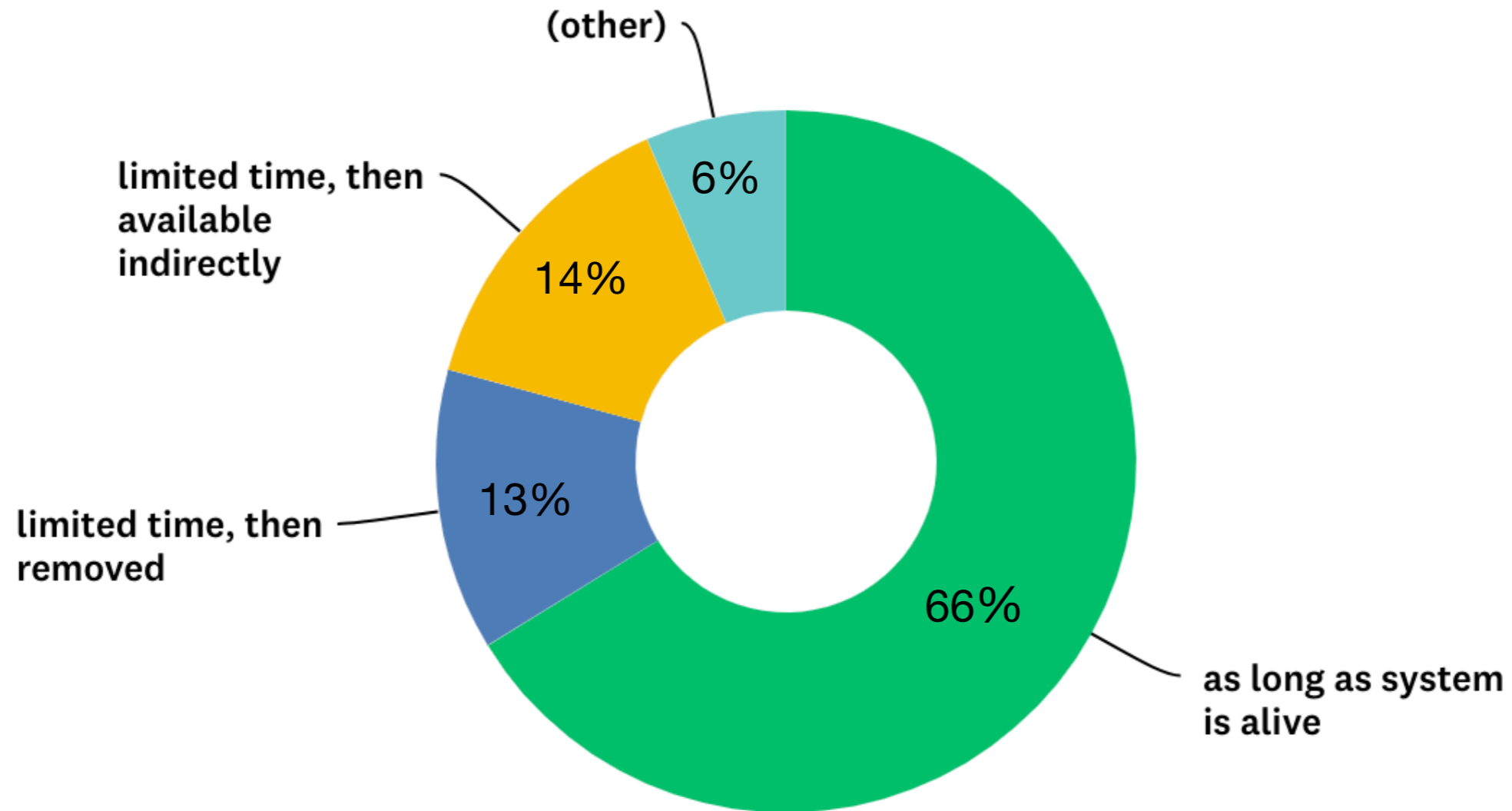
What is your favourite EasyBuild feature?



other:

- --try-*
- --robot
- --module-only
- installation logs
- RPATH support
- all of the above

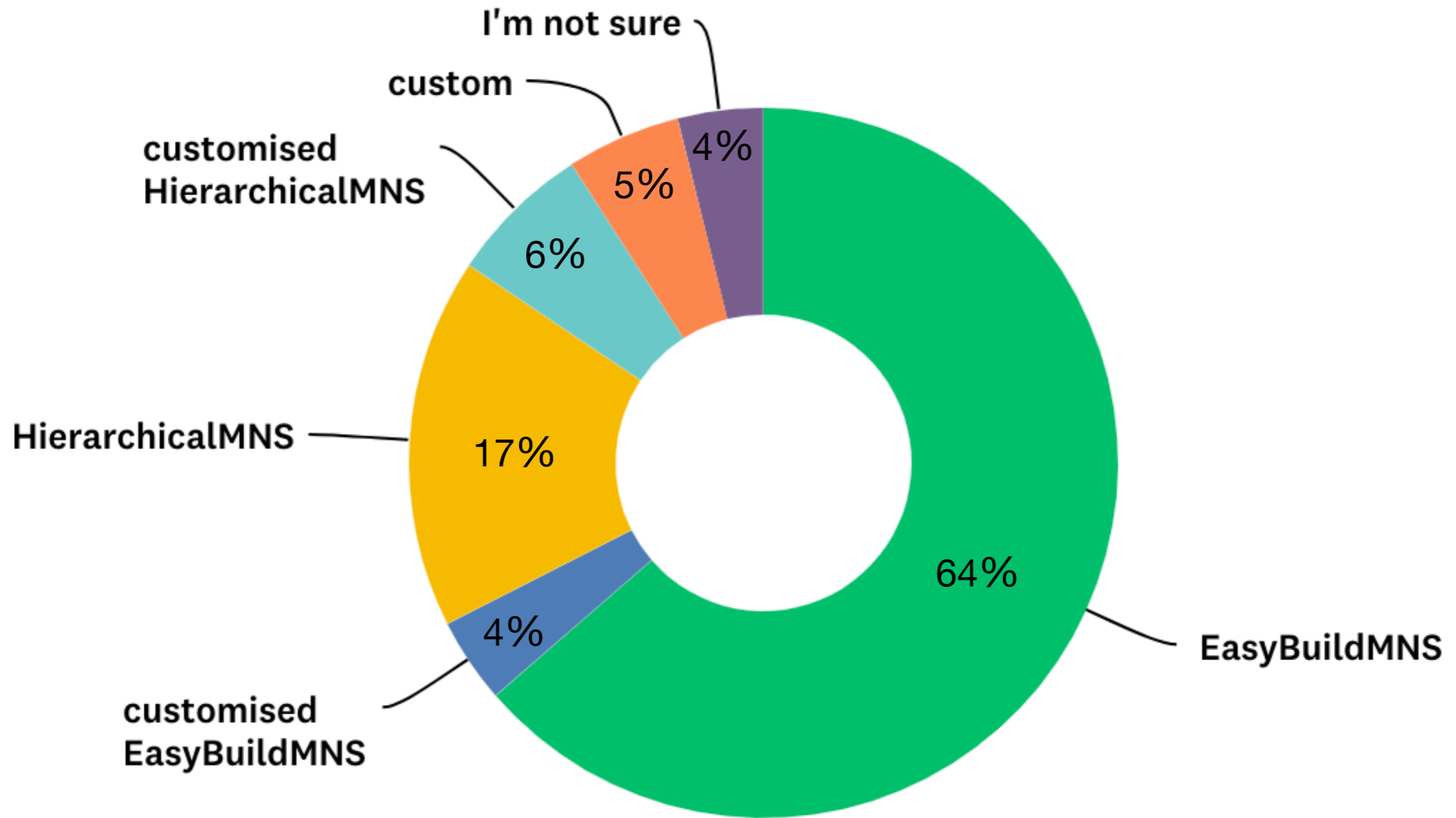
How long are modules installed with EasyBuild available to users?



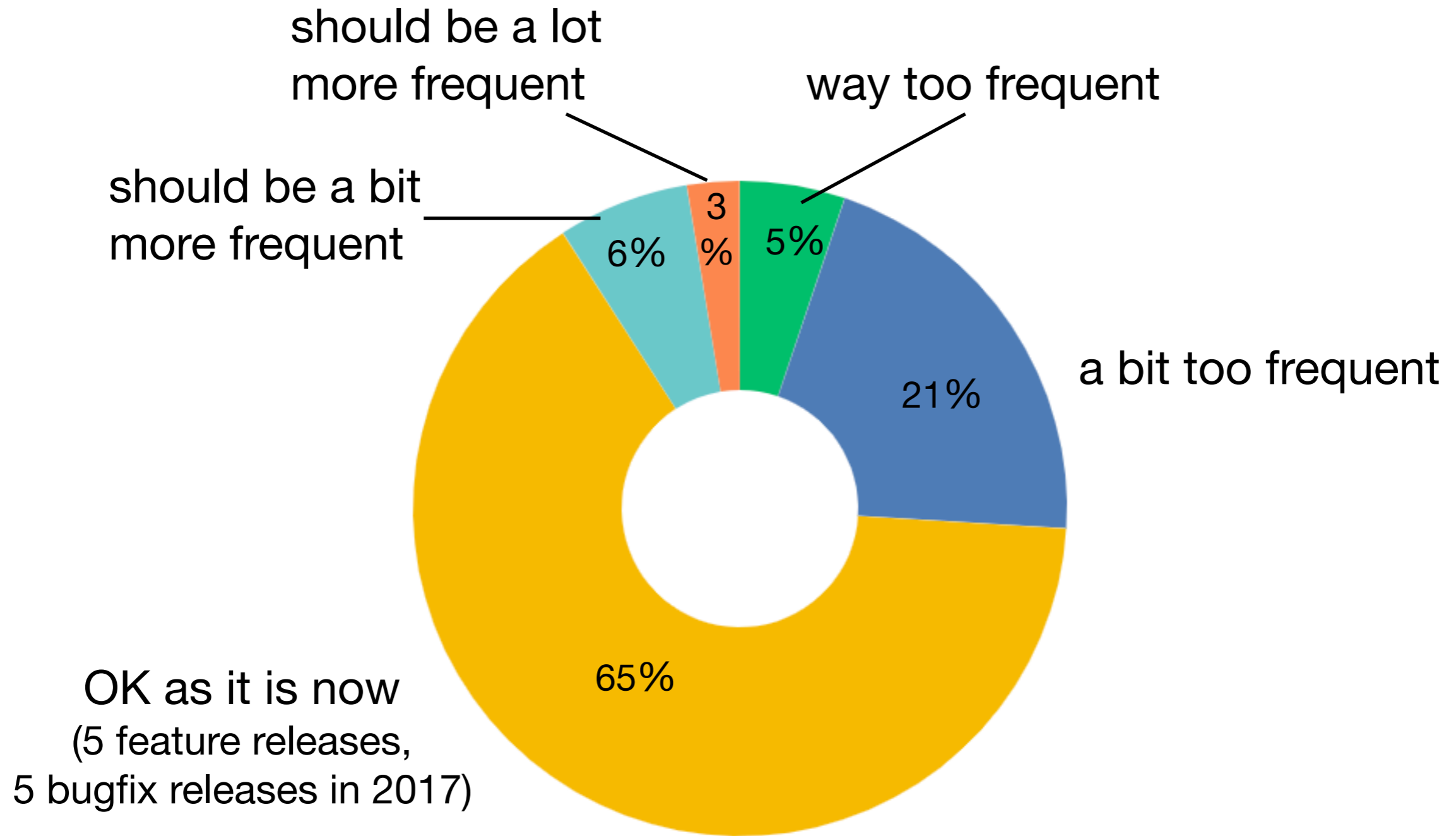
other:

- removed when it can't be supported anymore
- as long as toolchain is supported
- based on N-month review of usage
- undecided

Which module naming scheme do you use?

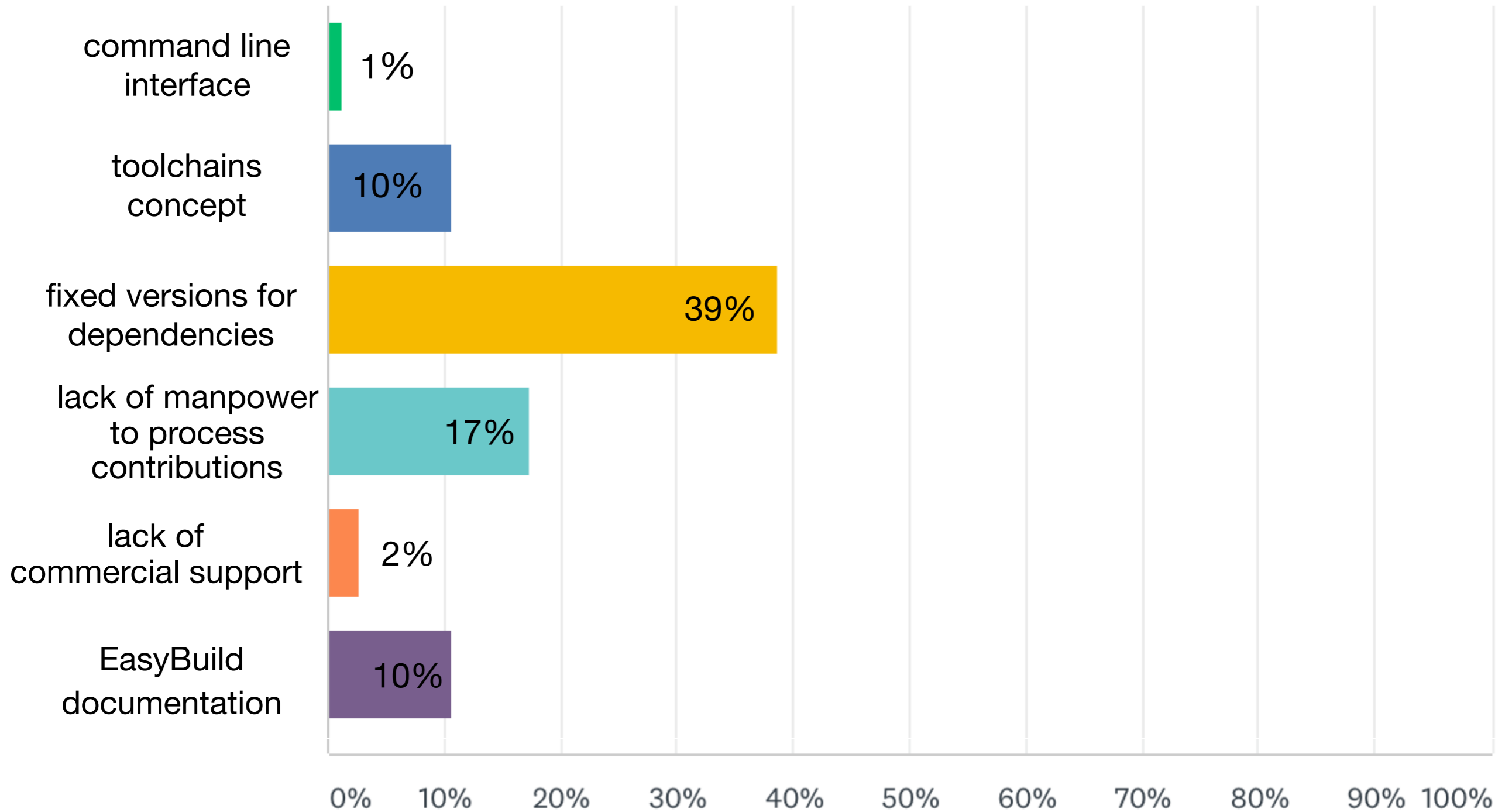


How do you like the frequency of EasyBuild releases?

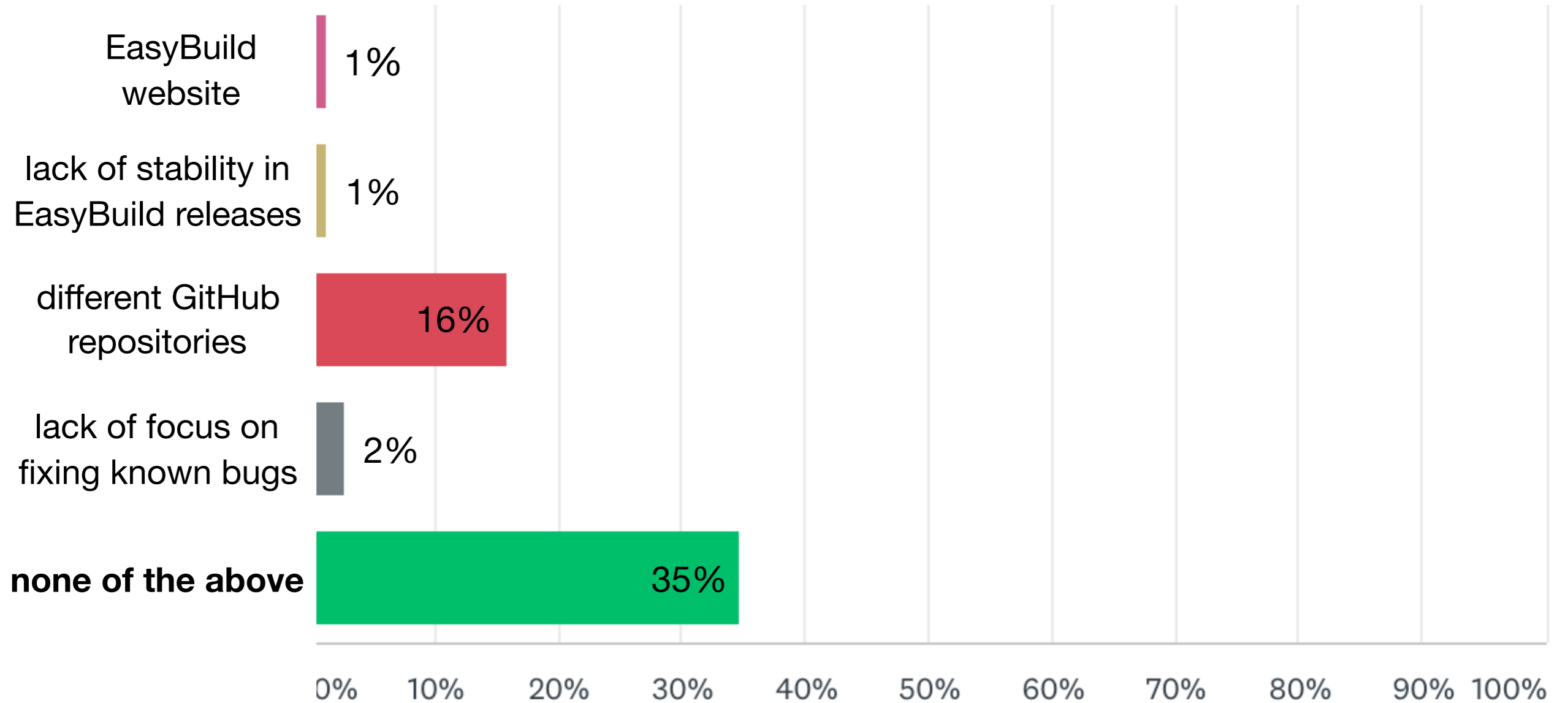


Conclusion: doing well, a bit less frequent may be OK

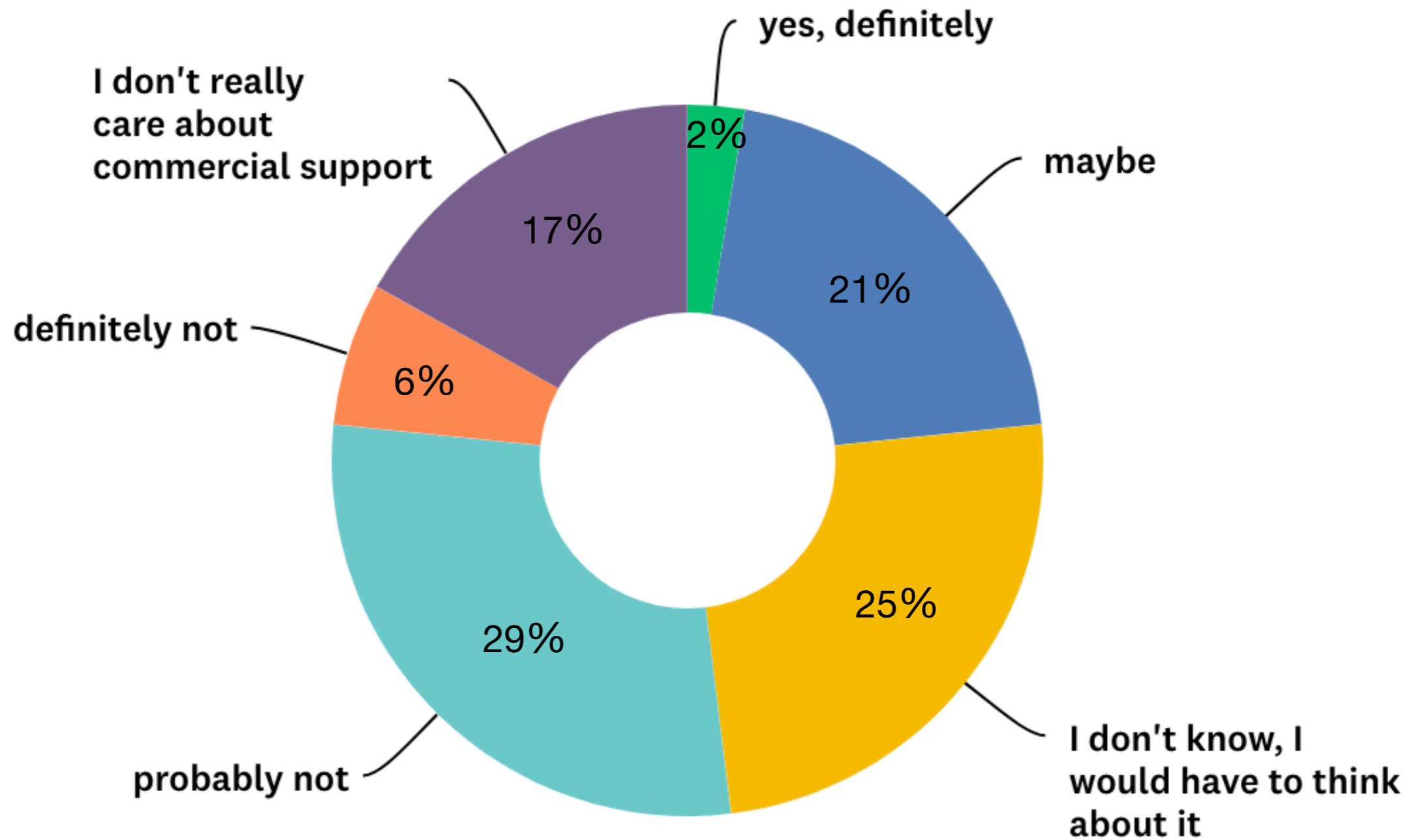
Which parts of EasyBuild do you *not* like?



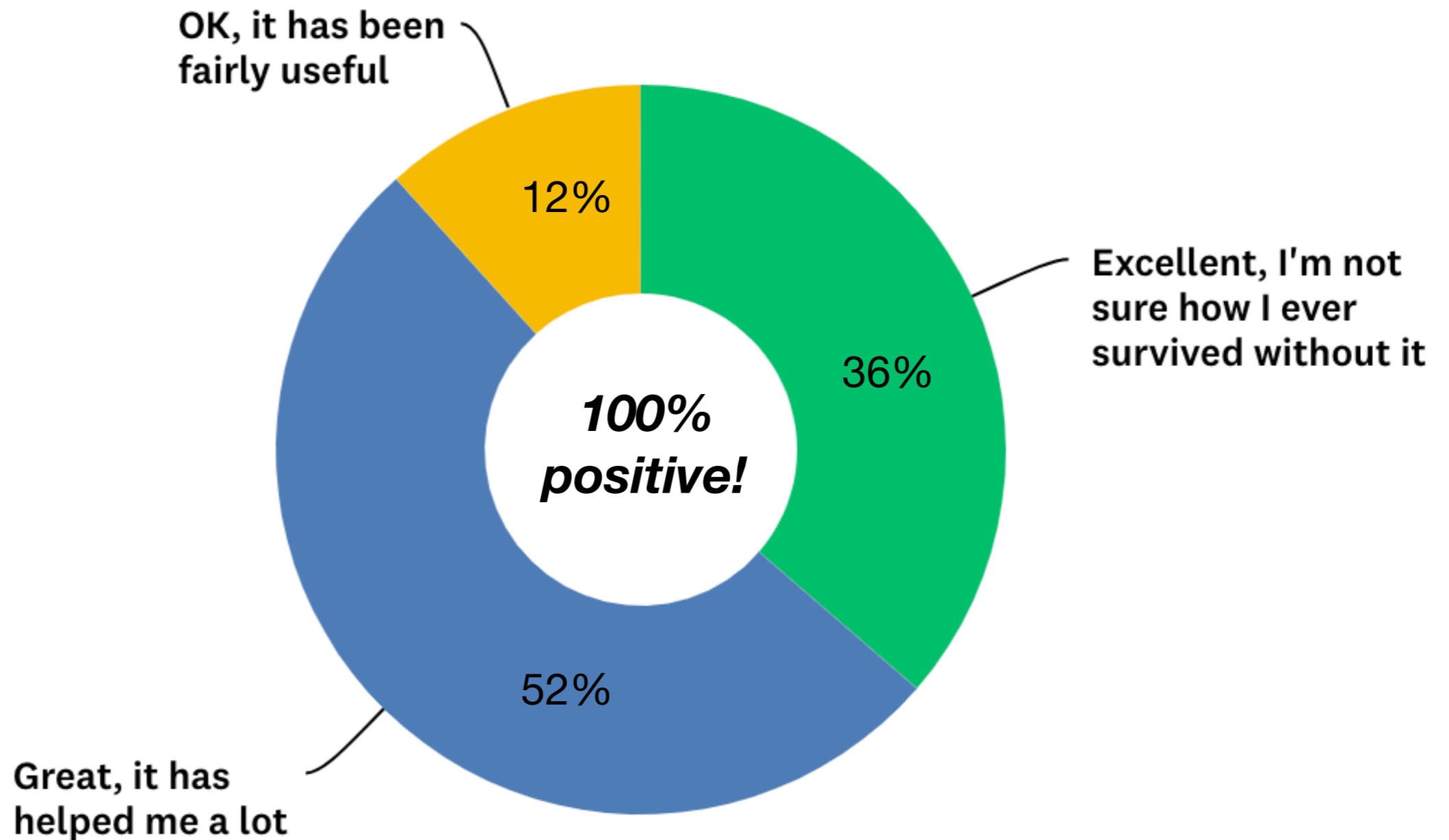
Which parts of EasyBuild do you *not* like?



If a commercial support option would be available, would you consider purchasing it?



How would you rate the overall quality of EasyBuild?



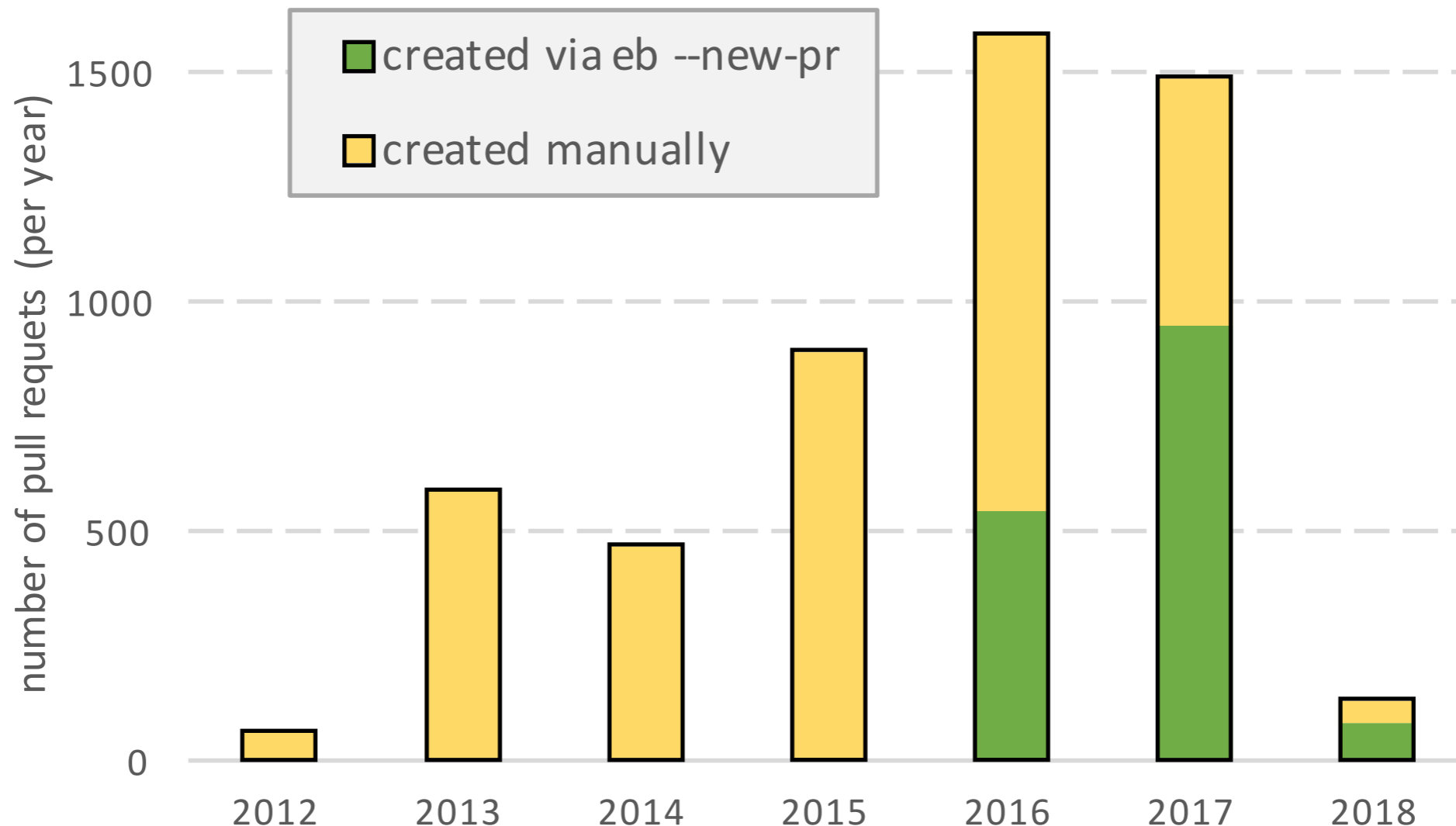
Note: not a single respondent picked "could be better" or "pretty bad"

Any additional comments?

- "Keep up the good work! Amazing software !"
- "I don't know what I would do without Easybuild :)"
- "EasyBuild is great! Keep doing the amazing work!"
- "thank you for making my job easier :)"
- "free beer for contributors!"

Impact of GitHub integration in EasyBuild

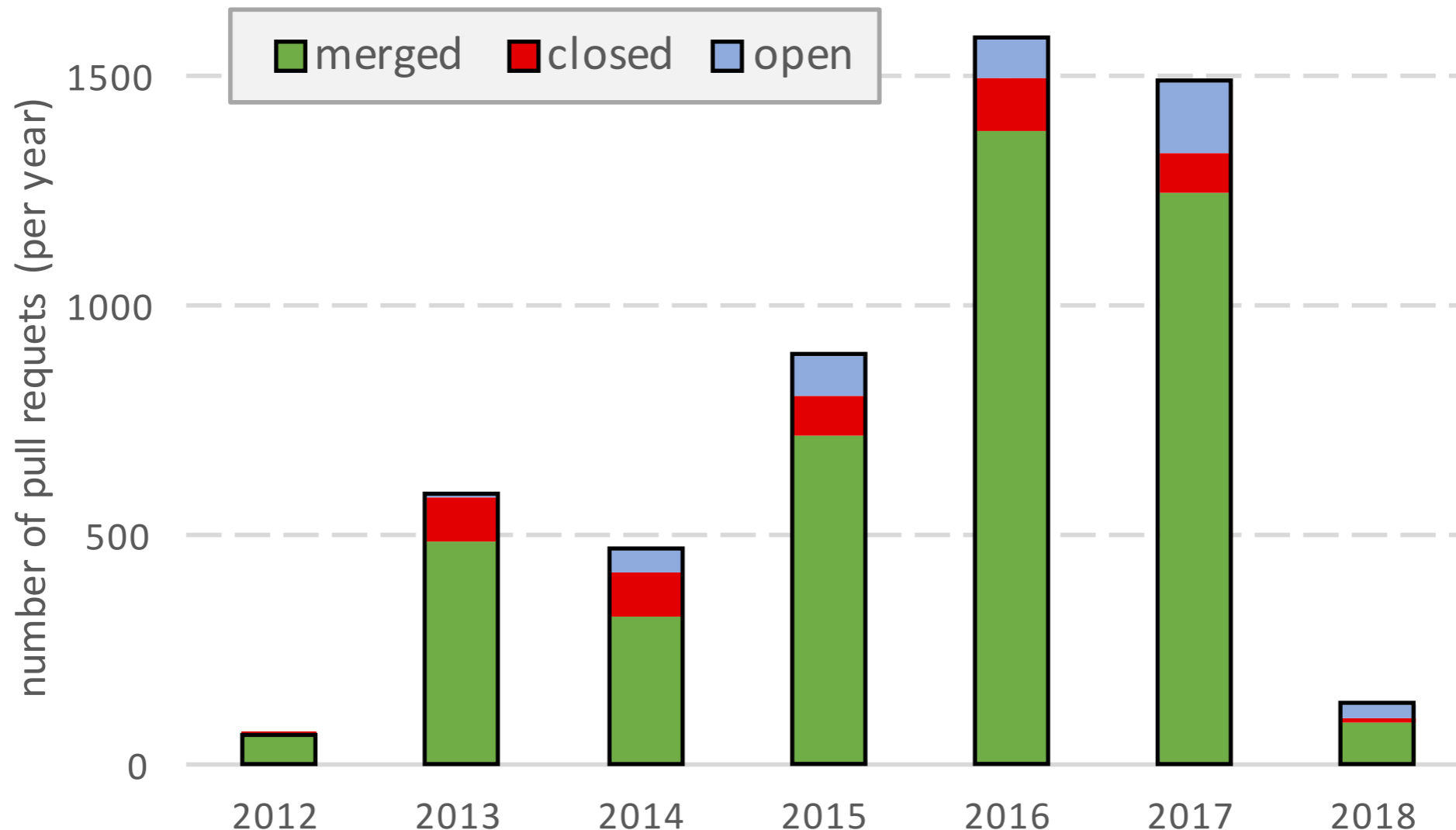
<https://github.com/easybuilders/easybuild-easyconfigs/pulls>



- > 500 PRs opened using `eb --new-pr` in 2016 (34%)
- almost 1000 PRs in 2017 (63%)
- 80% more PRs in 2016/2017 compared to 2015


Impact of GitHub integration in EasyBuild

<https://github.com/easybuilders/easybuild-easyconfigs/pulls>



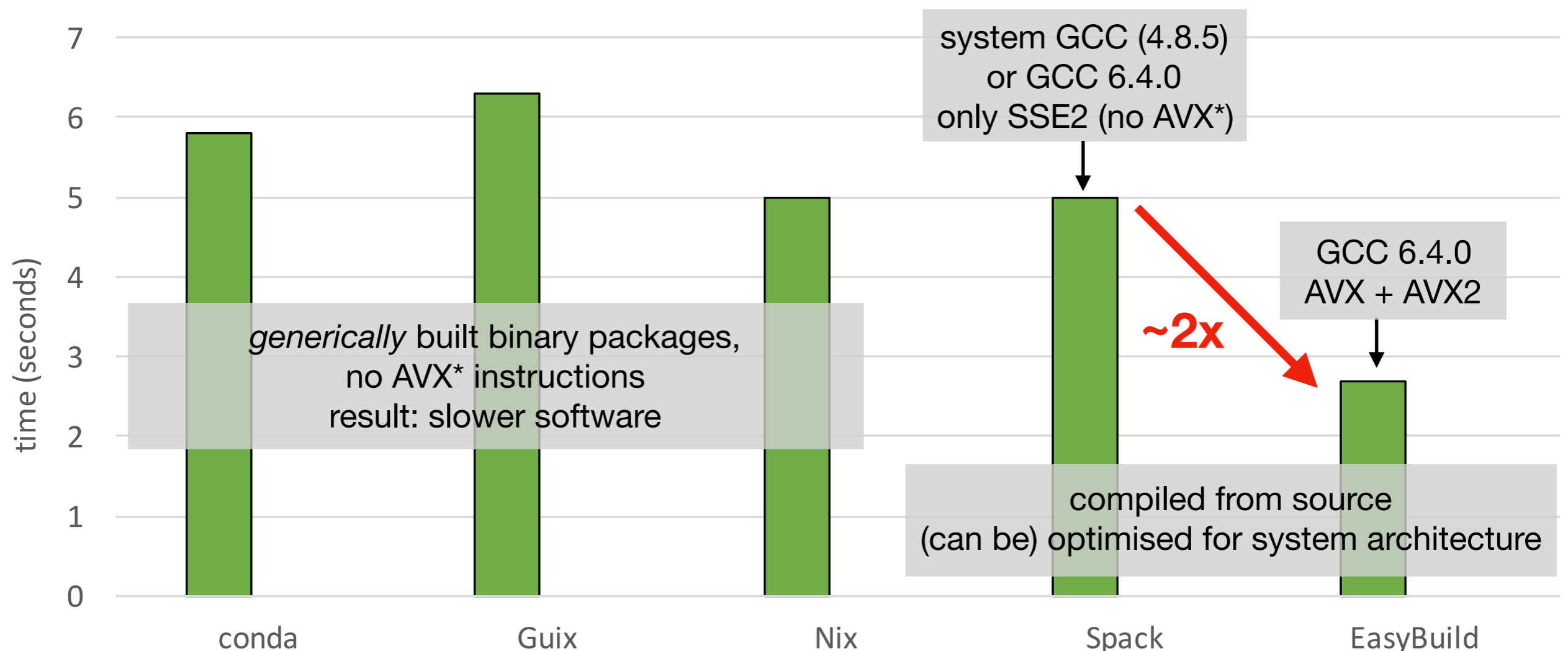
- `eb --merge-pr` has helped to keep pace with incoming PRs
- ~85% of PRs gets merged (usually a good reason for the ones that are not)
- would have been difficult without automation due to amount of PRs

Comparison of installation tools

	 CONDA	 easybuild	 Guix	 Nix	 Spack
platforms	Linux, macOS, Windows	Linux, Cray	GNU/Linux	Linux, macOS, Unix	Linux, macOS, Cray
implementation	Python 2/3, YAML	Python 2	Scheme, Guile	C++, Nix (DSL)	Python 2/3
supp. software	> 3,500	> 2,000	< 6,500	> 13,000	> 2,300
releases, install & update	★★★★☆	★★★★☆	★★★☆☆	★★★☆☆	★★☆☆☆
documentation	★★★★☆	★★★☆☆	★★★★☆	★★★★☆	★★★★☆
configuration	★★★★☆	★★★☆☆	★★☆☆☆	★★☆☆☆	★★★★☆
usage	★★★☆☆	★★★☆☆	★★★☆☆	★★★☆☆	★★★☆☆
time to result	★★★★★	★★☆☆☆	★★★★★	★★★★★	★★☆☆☆
performance	★★☆☆☆	★★★★★	★★☆☆☆	★★☆☆☆	★★★☆☆

Performance of FFTW installation

- single-core test from http://micro.stanford.edu/wiki/Install_FFTW3
 - N0, N1 set to 8192 to obtain sufficiently 'long' run times
- timings are for *default installations* (no tweaking)
- test system: CentOS 7.4, Intel E5-2680v3 (Haswell-EP) 2.5GHz

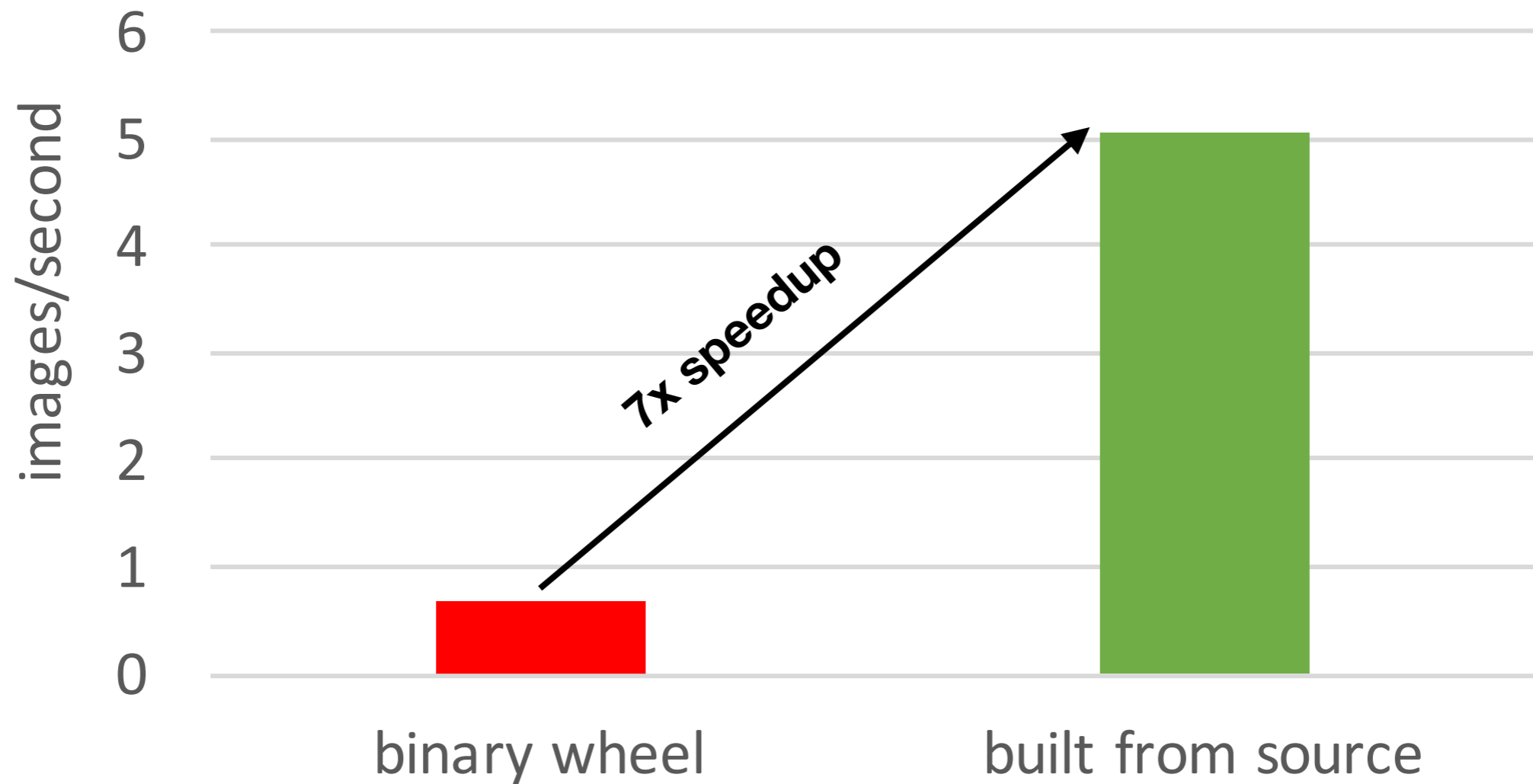


Future work

- WIP
 - support for building TensorFlow from source (both CPU/GPU)
 - `eb --update-versions`
 - Singularity integration
- limiting # of diff. software versions per common toolchain
- frequently requested features
 - `eb --continue`
 - `eb --remove`
 - more flexible dependency version specs
 - support for multiple active module naming schemes



ResNet-50 on Intel Haswell (CPU only)



Singularity integration

- see recent work by Shahzeb Siddiqui (Pfizer)
- <https://github.com/easybuilders/easybuild-framework/pull/2332>
- working towards support in EasyBuild for:
 - generating Singularity definition files
 - building Singularity images with EasyBuild
- concept can easily be extended to Docker & co too
- example:

```
eb OpenMPI-2.1.2-GCC-6.4.0-2.28.eb --singularity  
--singularity-bootstrap shub:... --buildimage ...
```