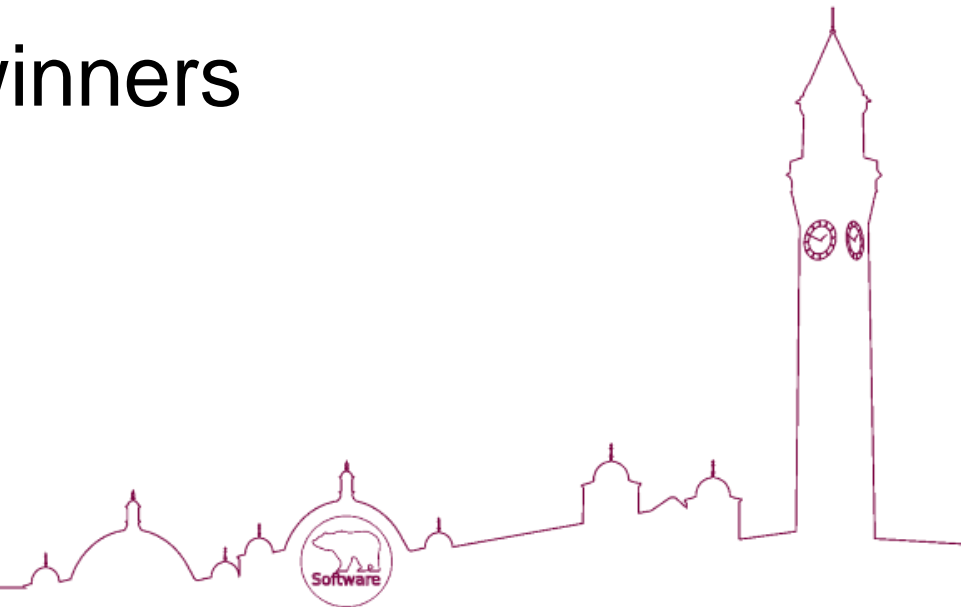# EasyBuild and POWER9

Thursday 30th January 2020
Simon Branford
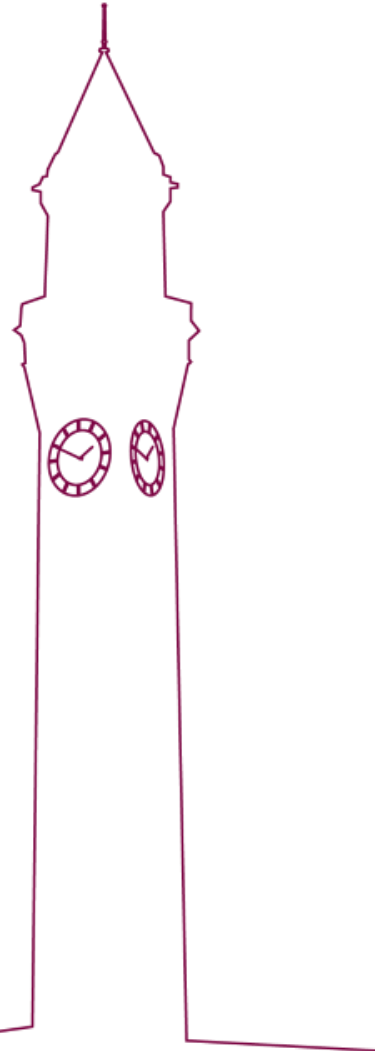
University of Birmingham

- Royal charter in 1900 (history back to 1828)
- Member of Russell Group
- 34835 students (2017/18) – 4th largest in UK
- >7000 staff
- 11 staff and alumni are Nobel prize winners
- £134.2M research income (2017/18)
- £3.5bn economic impact
- Campuses in Birmingham and Dubai

BEAR

☐ Birmingham Environment for Academic Research
- HPC, storage, high speed networking
- POWER9 AI Cluster
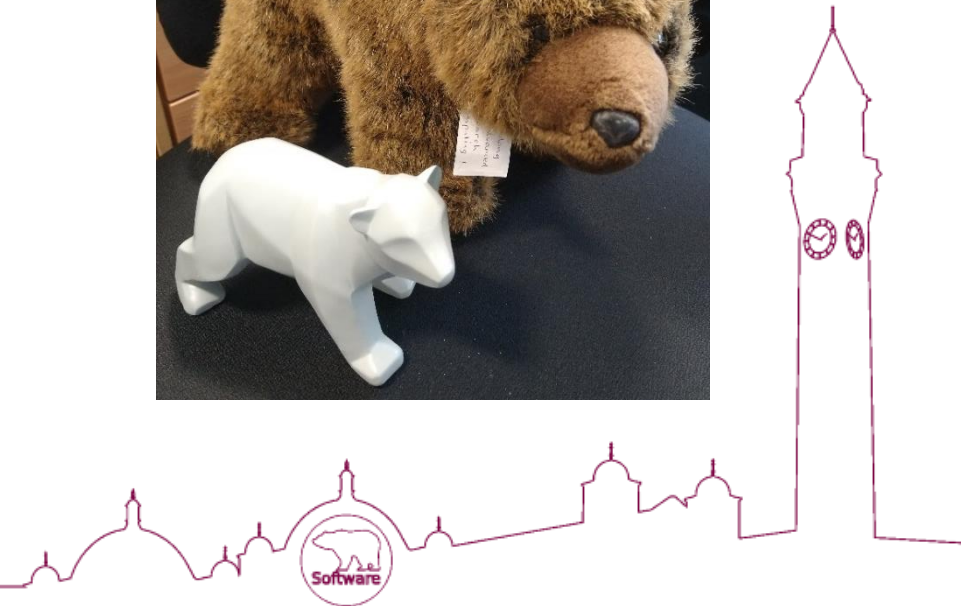
Advanced Research Computing

- Architecture, Infrastructure, and Systems
  - Systems design, service integration, hardware management, core service software maintenance, high speed networking, supplier management, 7 FTE + 1 worklink student
- Research Engagement Group
  - Training coordination, data management, evangelising services, 3.6 FTE
- Research Software Group
  - HPC software, user support, improving research software, 8 FTE + 1 worklink student

BlueBEAR and Our Users

- ☐ BlueBEAR is our Linux supercomputer
  - – Currently v3 – started in 2012
- ☐ Available for free to all researchers at the University
  - – We have lots of bioinformaticians
  - – Lots of HTC, some HPC
- ☐ Groups can purchase additional resources
- ☐ We do rolling updates
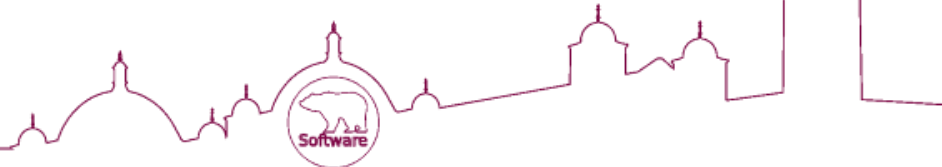 - – hardware and software
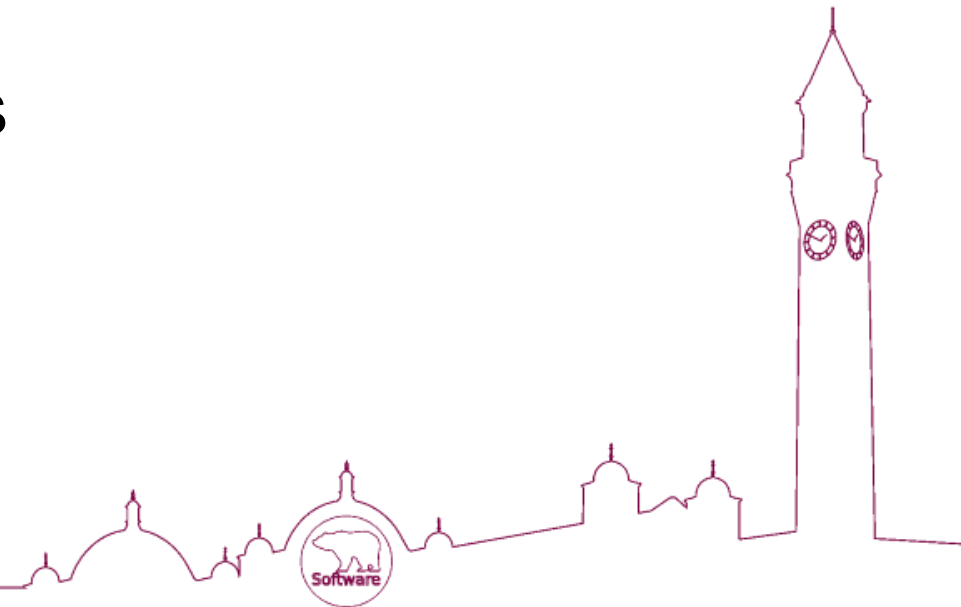
Hardware and OS

- ☐ BlueBEAR
  - – Intel (Centos 7.7)
    - ☐ Sandybridge
    - ☐ Haswell / Broadwell + GPUs (/ SkyLake + GPUs)
    - ☐ CascadeLake
  - – IBM (RHEL 7.6)
    - ☐ POWER9 + GPUs
- ☐ BEARCloud / CaStLeS VMs
  - – Intel CPUs (Centos 7.7 or Ubuntu 16.04)
   - ☐ NFS and no-Infiniband
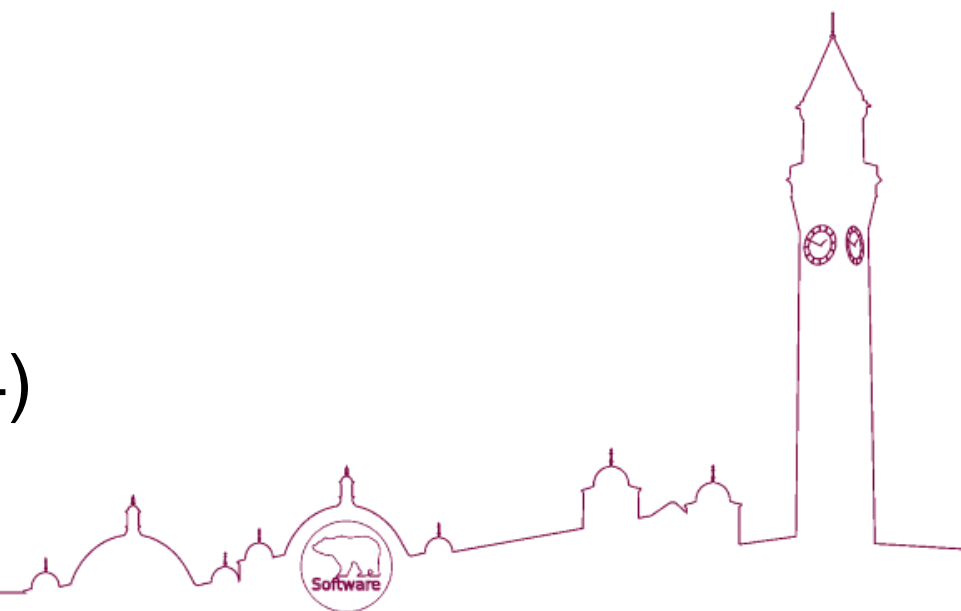
Available Software

- Application count: 1117
- Version count: 3203
- Version on architecture count: 11676

EasyBuild at Birmingham: 2016-2017

☐ Started using in 2016
☐ Early use was forcing EasyBuild to match existing module structure
☐ Lots of local customisations
☐ Made it difficult to:
  – pull from upstream
  – contribute back
☐ Making the job harder for us
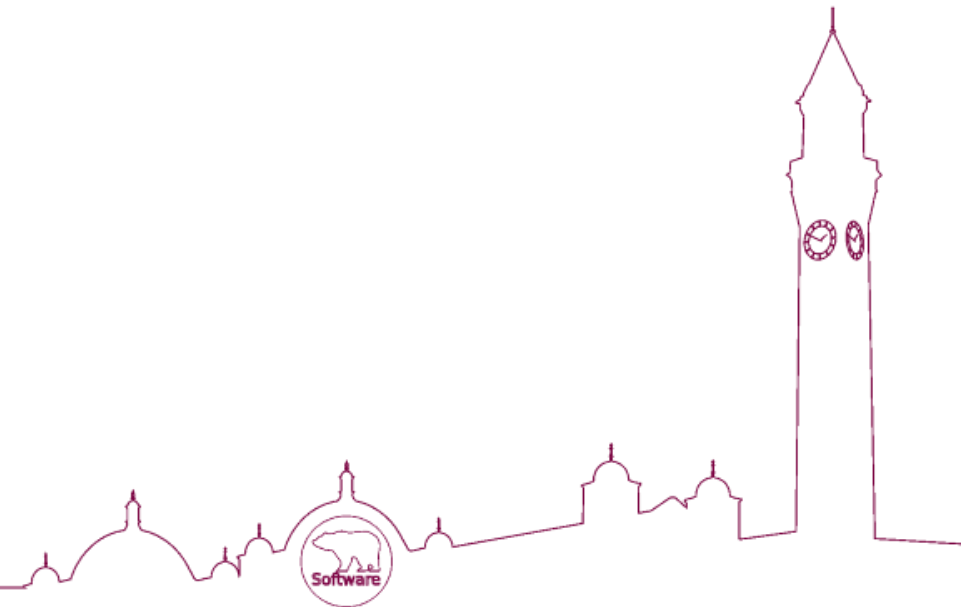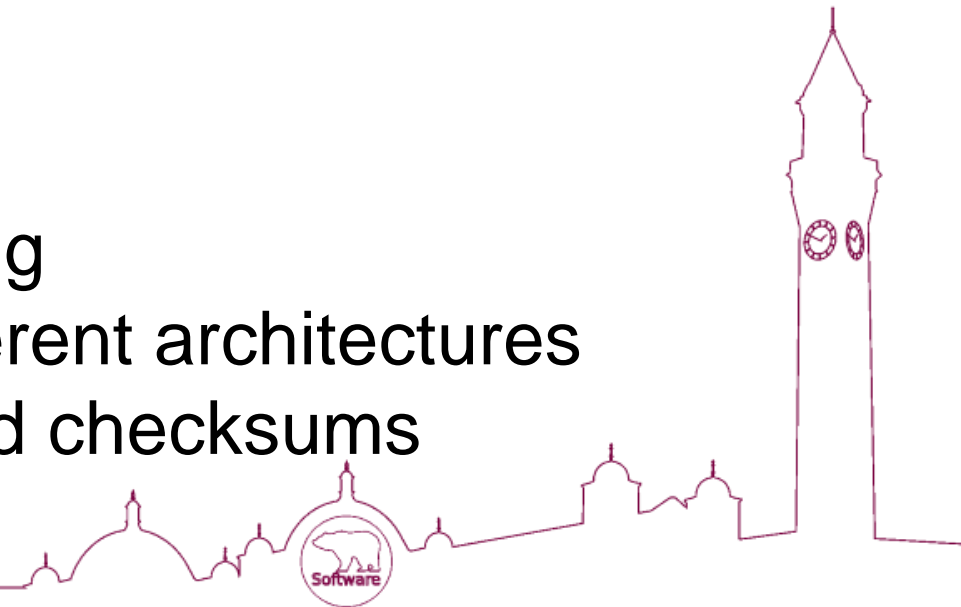
EasyBuild at Birmingham: 2017-2018

☐ Several new people joined ARC
   – Me being one of them!
☐ Decision to move closer to upstream
   – Standard module tree, with little local modifications
☐ We were getting closer to upstream...
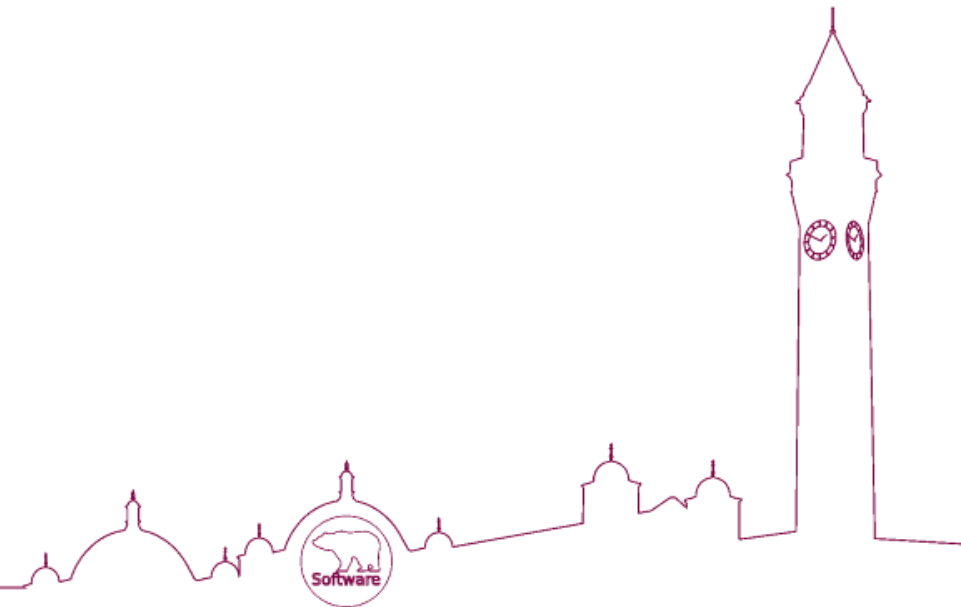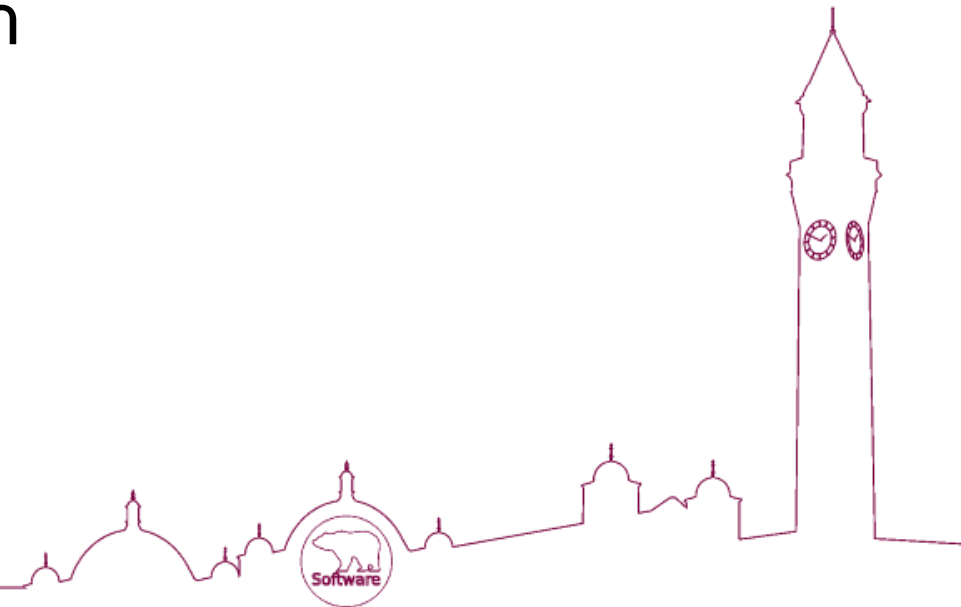
EasyBuild at Birmingham: POWER9

☐ Arrived late 2018 and more in 2019

☐ Desire to treat them the same as the rest of BlueBEAR

☐ EasyBuild - not working out of the box

  – our ecs repositories from then has lots of

```
import os  # noqa
arch = os.environ.get("BB_CPU")
if arch == ...
```

☐ Worked with the EasyBuild maintainers, e.g

  – ebs - alterations in what is done on different architectures

  – ecs - specify different dependencies and checksums

EasyBuild at Birmingham: 2020

☐ Moved from internal GitLab to using GitHub
  – https://github.com/bear-rsg/
    ☐ configs, blocks, and framework repositories
☐ Aim is to contribute more upstream
  – James contributed his first ec this month

EasyBuild at Birmingham: How we Work

- ☐ All installations are by request
  - – We decide on which toolchain to use for the installation
- ☐ Each six-monthly period - new branch for ecs, ebs, and framework
  - – Installations use that toolchain - so foss 2019b in 2019b
  - – Allows us to review local modifications
  - – Generally we have two active for installation
    - ☐ Currently 2019a and 2019b
  - – 2019b lives in the 2019b branches in our GitHub repositories
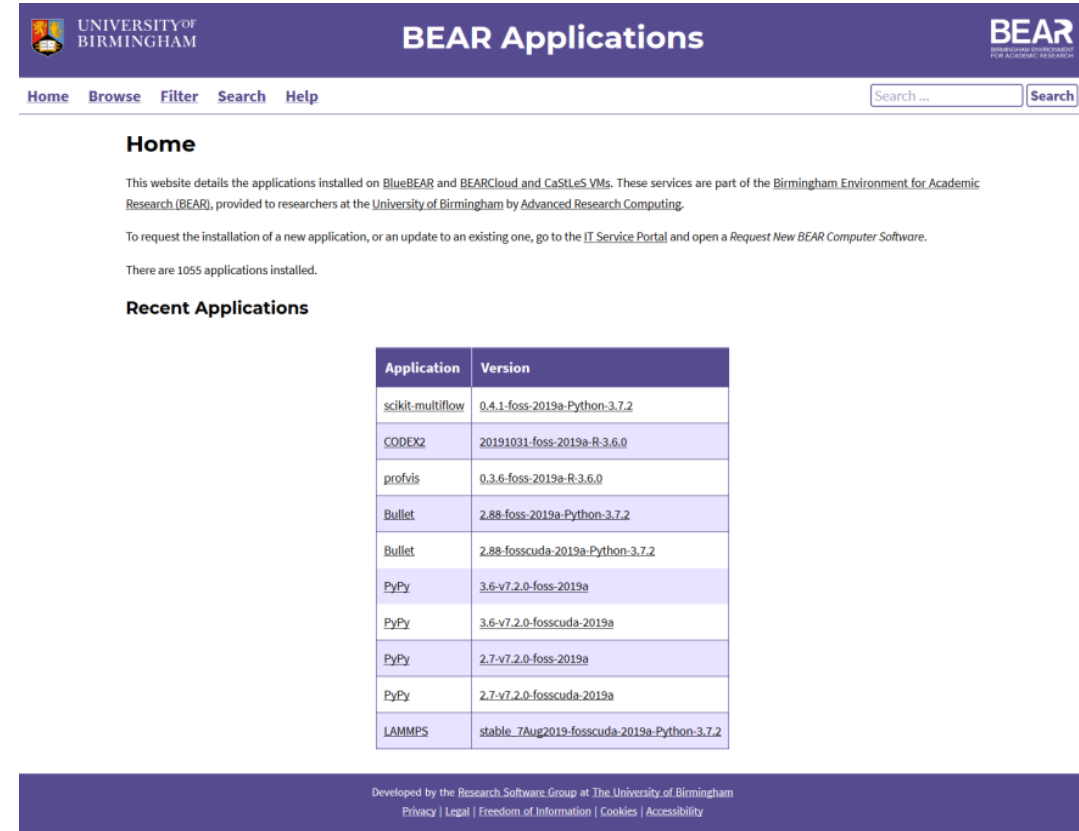- ☐ Each developer has their own EB environment

EasyBuild at Birmingham: Module setup

☐ BlueBEAR setup, so that the modules and software is then organised by six month period, architecture, and OS

  – `[path]/2019b/EL7-cascadelake/{modules,software}`

☐ Module paths are then set automatically on each system

  – User does not need to know what arch / OS they are on

☐ We have only a few users who self build software

  – We give them options, such as restrict what arch they use, build versions per arch, etc...
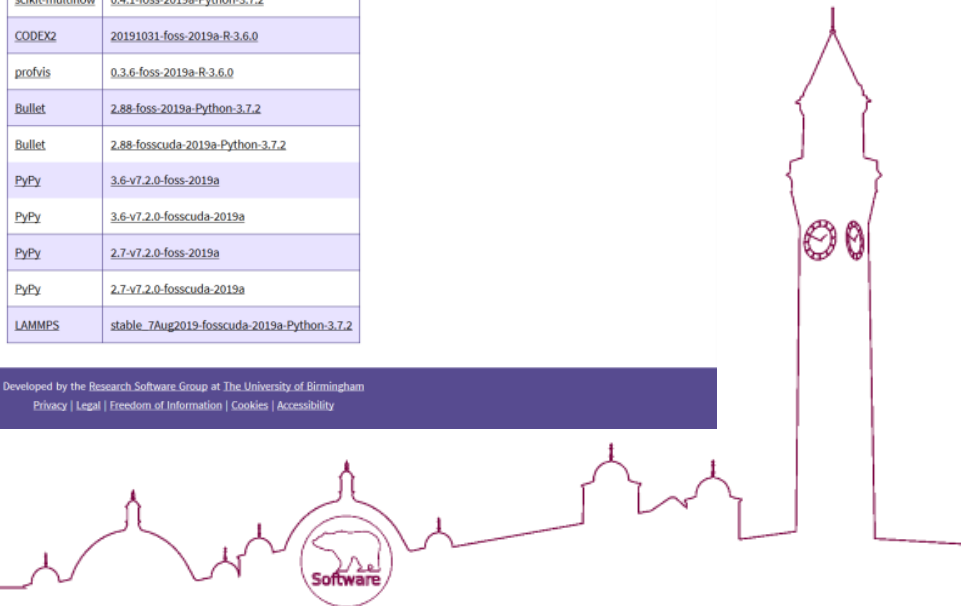
BEAR Applications Website

☐ bear-apps.bham.ac.uk

☐ Old solution was manually created pages in CMS

☐ Too many applications …
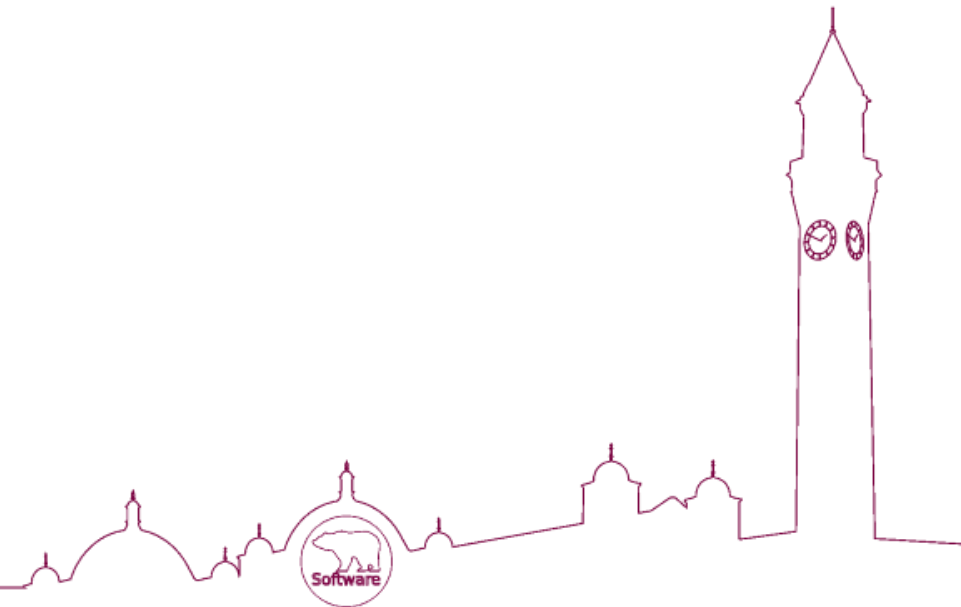
☐ Automatic
  – But allowing customisation

ReFrame

```
Successful test run 1580185837:
    *  "--performance-report" "-t" "nightly" "-r"
    * Branch: master, Hash: 573ffca49eb8bd0523a147a649247fce505b9798
[  PASSED  ] Ran 1341 test case(s) from 486 check(s) (0 failure(s))
```

Thanks

?