



# easybuild

building software with ease since 2009

SC14 BoF lightning talk

Getting Scientific Software Installed: Tools & Best Practices

19 November 2014

[jens.timmerman@ugent.be](mailto:jens.timmerman@ugent.be) — [easybuild@lists.ugent.be](mailto:easybuild@lists.ugent.be)



easybuild

Do you get these?

“Please install <software> on the HPC cluster?”



easybuild

Do you get these?

“Please install <software> on the HPC cluster?”  
By tomorrow. But preferably by yesterday.



- ▶ Scientists (rightly) focus on the science of the software they produce, not on build procedure(s), portability, ...
- ▶ This makes building/installing (lots of) scientific software painful: (more often than not) very time-consuming, error-prone, hard to get right, a lot of dependencies (that often suffer from the same problems, ...)





easybuild

# Common issues

- ▶ non-standard build tools
- ▶ incomplete build procedures, for example, no install step
- ▶ interactive scripts
- ▶ hardcoded parameters
- ▶ poor/outdated documentation
- ▶ a lot of work and effort duplication across HPC sites.



easybuild

# EasyBuild to the rescue

- ▶ EasyBuild is a software build and installation framework.
  - ▶ written in Python 2.x (2.4 and higher)
  - ▶ started in 2009, in-house for ~2.5 years, GPLv2 since 2012
  - ▶ stable API since EasyBuild v1.0 (November '12), latest is version 1.15.2
  - ▶ continuously enhanced and extended, thoroughly tested
  - ▶ release early, release often strategy (major version every 4-6 weeks)
  - ▶ development is highly community-driven

<http://hpcugent.github.io/easybuild>



easybuild

# EasyBuild community

lead developer (Kenneth Hoste)



Ghent University & VSC sites (Belgium)

University of Luxembourg

IMP/IMBA (Austria)

The Cyprus Institute

University of Basel (Switzerland)

Jülich Supercomputer Centre (Germany)

Bayer (Germany)

University of Auckland (New Zealand)

Kiev Polytechnic Institute (Ukraine)

Idaho National Lab (US)

Pacific Northwest National Lab (US)

UC Davis (US)

& (many?) more...

6th EasyBuild hackathon

18-20 June 2014, Vienna (Austria)

+ support from NVIDIA, TACC (Lmod), ...



easybuild

# Starting with EasyBuild

0. Easily install EasyBuild by bootstrapping it

```
$ curl -O http://hpcugent.github.io/easybuild/bootstrap_eb.py  
$ python bootstrap_eb.py <prefix>
```

1. Set module path, load EasyBuild module, basic configuration

```
$ export MODULEPATH=<prefix>/modules/all:$MODULEPATH  
$ module load EasyBuild  
$ export EASYBUILD_PREFIX=<prefix>
```

2. Example: build WRF & all deps using Intel compilers/libraries

```
$ eb WRF-3.5-ictce-5.3.0.eb --robot  
  
$ module av WRF  
WRF/3.5-ictce-5.3.0
```





easybuild

# Supported software (1.15.1)

483 different software packages (2,501 example builds)

a2ps ABAQUS ABINIT ABySS ACML ALADIN Alinea ALLPATHS-LG AMOS AnalyzeFMRI ANSYS ant APBS ARB argtable aria2 Armadillo arpack-ng ASE ATLAS Autoconf Automake bam2fastq BamTools Bash BayesTraits bbcp bbFTP bbftpPRO bc beagle-lib Beast BEDTools BFAST binutils biodeps BioPerl Biopython BiSearch Bison BitSeq BLACS BLAST BLAT BOINC Bonnie++ Boost Bowtie Bowtie2 BWA byacc bzip2 cairo CAP3 CBLAS ccache CCFits CD-HIT CDO CEM CFITSIO cflow CGAL cgdb cgmipich cgmipolf cgmipapich2 cgmivolf cgompi cgoolf Chapel CHARMM Circos Clang ClangGCC CLHEP CLooG Clustal-Omega ClustalW2 CMake Coreutils Corkscrew **CP2K** CPLEX CRF++ Cube CUDA Cufflinks cURL cutadapt CVS CVXOPT Cython DB DB\_File Diffutils DL\_POLY\_Classic Docutils **DOLFIN** Doxygen **EasyBuild** ECore ed Eigen ELinks ELPA ELPH Emacs EMBOSS EPD ErlangOTP ESMF ESPResSo evmix expat eXpress FASTA fastahack FastTree FASTX-Toolkit FCM FDTD\_Solutions Ferret FFC FFTW FIAT file findutils fixesproto flex FLTK FLUENT fmri FoldX fontconfig foss FRC\_align freeglut FreeSurfer freetype FSL g2clib g2lib GATE GATK gawk GCC gcccuda GD GDAL GDB Geant4 GEM-library GEMSTAT GenomeAnalysisTK GEOS gettext GHC Ghostscript gimkl gimpi GIMPS git GLib GLIMMER GLPK glproto gmacml GMAP GMP gmpich2 gmpolf GMT gmipapich2 gmivolf gnuplot gnutils Go goalf gompi gompic google-sparsehash goolf goolfc GPAW gperf gperftools Greenlet grep grib\_api GROMACS GSL gsl GTI guile gzip h4toh5 h5py h5utils Harminv HDF HDF5 HH-suite HMMER horton HPCBIOS\_Bioinfo HPCBIOS\_Debuggers HPCBIOS\_LifeSciences HPCBIOS\_Math HPCBIOS\_Profilers HPCG HPL HTSeq hwloc Hypre icc iccifort ictce ifort iimpi iigmpi imake imkl impi Infernal inputproto Inspector Instant intel iomkl iompi IOR lperf ipp IPython iqacml Isolnfer ispc itac JAGS Jansson JasPer Java Jellyfish Jinja2 JUnit kbproto LAPACK less lftp libcircle libctf libdrm libffi libgd libgtextutils libharu libibmad libibumad libibverbs libICE libidn Libint libint2 libjpeg-turbo libmatheval libpciaccess libpng libpthread-stubs libreadline libSM libsmm LIBSVM LibTIFF libtool libungif libunistring libunwind libX11 libXau libXaw libxc libxcb libXext libXfixes libXi libxml2 libXmu libXp libXpm libxslt libXt libyaml likwid Lmod Lua LWM2 lxml lynx LZO M4 MAFFT make makedepend Maple MariaDB Mathematica MATLAB matplotlib mc MCL mcpp MDP mdtest Meep MEME Mercurial Mesa Mesquite MetaVelvet MethPipe METIS MMSEQ Modeller Molden Molekel molmod Mothur motif MPFR mpi4py mpiBLAST MPICH MPICH2 MrBayes MTL4 MUMmer MUMPS MUSCLE MUST MUSTANG MVAICH2 NAMD nano NASM NCBI-Toolkit ncd4 **NCL** ncurses nview NEdit netaddr netCDF netCDF-C++ netCDF-C++4 netCDF-Fortran netcdf4-python netifaces netloc nettle **NEURON** nodejs ns numactl numexpr numpy NWChem O2scl Oases OCaml Oger OPARI2 OpenBabel OpenBLAS **OpenFOAM** **OpenFOAM-Extend** OpenIFS OpenMPI OpenPGM OpenSSL ORCA orthoncl otcl OTF OTF2 packmol PAML pandas PANDAsq PAPI parallel Paraview ParFlow ParMETIS ParMGridGen Pasha patch paycheck PCC PCRE PDT Perl **PETSc** petsc4py phonopy PhyML picard pixman pkg-config PLINK PnMPI popt PP PRACE PRANK Primer3 printproto problog protobuf pscom PSI psmi2 PyQuante pysqlite pyTables **Python** python-dateutil python-meep PyYAML PyZMQ QLogicMPI Qt qtop QuadProg++ **QuantumESPRESSO** R RAXML RCS RDP-Classifer RNAz ROOT Rosetta rSeq RSEQtools Ruby Sablotron SAMtools ScalAPACK Scalasca ScientificPython scikit-learn scipy SCons SCOOP Score-P SCOTCH SDCC SDPA sed segemehl setuptools Shapely SHRIMP SIBELia sickle Silo slalib-c SLEPc SOAPaligner SOAPdenovo SOAPdenovo2 SOAPec SPAdes Sphinx SQLite SRA-Toolkit Stacks stemming Stow Stride SuiteSparse SURF SWIG sympy Szip TAMkin Tar tbb TCC Tcl tclcl tcsh Tesla-Deployment-Kit texinfo Theano TiCCutils TIMBL TinySVM Tk TopHat Tornado TotalView TREE-PUZZLE Trilinos Trinity UDUNITS UFC UFL util-linux Valgrind VCFtools Velvet ViennaRNA Vim Viper vsc-base vsc-mypirun vsc-mypirun-scoop vsc-processcontrol VSC-tools VTK VTune WHAM **WIEN2k** wiki2beamer **WPS** **WRF** xbitmaps xcb-protocol XCrySDen xextproto XML XML-LibXML XML-Simple xorg-macros xproto xtrans XZ yaff YamCha YAML-Syck Yasm YAXT ZeroMQ zlib zsh zsync



easybuild

# Current status

currently (EasyBuild v1.15.2):

- ▶ supports GCC/Clang/Intel compilers & various MPI/BLAS/LAPACK/FFT libraries
- ▶ robust **framework** providing supporting functionality
- ▶ very **dynamic design**: plugin support for new compiler/MPI/software package
- ▶ **generates module files** (Tcl), supports using (C/)Tcl and Lmod module tools
- ▶ support for using a custom module naming scheme you define yourself
- ▶ **fully autonomous builds**, build logging, automagical dependency resolution, ...
- ▶ support for hierarchical module naming schemes

work in progress:

- ▶ improve integration with Lmod (e.g., support for generating Lua module files)
- ▶ better support for site customisation
- ▶ better error reporting
- ▶ support for more compilers, software applications and platforms
- ▶ new format for easyconfig files (build specs) to handle explosion of contributions

**The EasyBuild community drives most new features, so do get involved!**



easybuild

Want to know more?

**website:** <https://hpcugent.github.io/easybuild>

**GitHub:** [https://github.com/hpcugent/easybuild\[-framework|-easyblocks|-easyconfigs\]](https://github.com/hpcugent/easybuild[-framework|-easyblocks|-easyconfigs])

**PyPi:** [https://pypi.python.org/pypi/easybuild\[-framework|-easyblocks|-easyconfigs\]](https://pypi.python.org/pypi/easybuild[-framework|-easyblocks|-easyconfigs])

**mailing list:** [easybuild@lists.ugent.be](mailto:easybuild@lists.ugent.be)

**Twitter:** @easy\_build

**IRC:** #easybuild on freenode.net

**YouTube:** search for “EasyBuild intro” or “EasyBuild WRF”