Installing Scientific Software on HPC with EasyBuild

14th International SuperComputing Camp
https://sc-camp.org

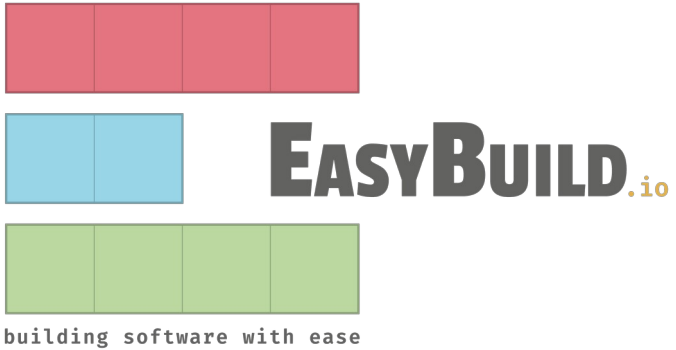
SC-Camp 2023

Cartagena, Colombia
May 14-20, 2023

Xavier Besseron
University of Luxembourg

LuXDEM Research Team
https://luxdem.uni.lu

Outline



- Part 1: Short Introduction to EasyBuild
- Part 2: EasyBuild at work

This tutorial is based on EasyBuild tutorials available at http://tutorial.easybuild.io/.

uni.lu UNIVERSITÉ DU LUXEMBOURG

Short Introduction to EasyBuild

What is EasyBuild?

- Software **build** and **installation** framework
- Targeted to **scientific software**
- Tailored towards **High Performance Computing** systems
- Focus points: **Performance**, **Reproducibility**, **Collaboration**

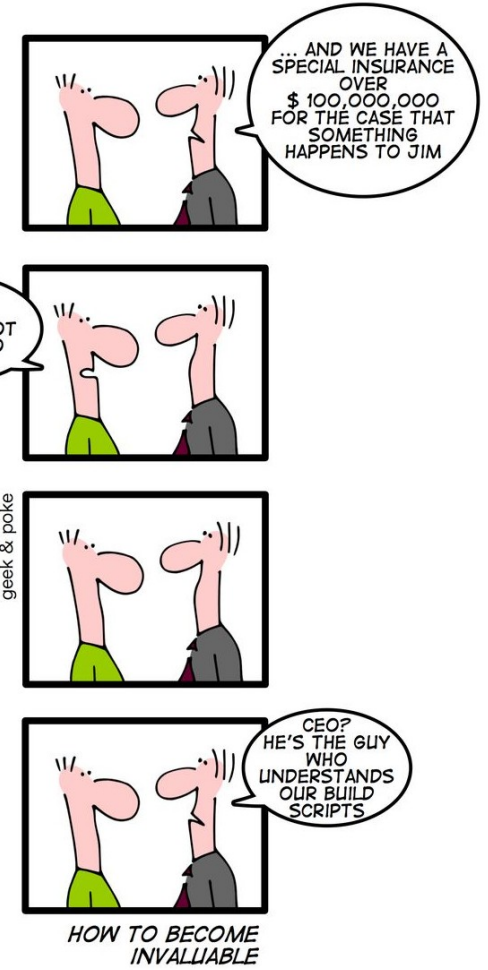
Key features

- Fully autonomous installation
- No admin privileges are required
- Highly configurable

Objective

→ **Empower scientists to self-manage their software stack on HPC**

Alternative to EasyBuild → **Spack** https://spack.io/



Supported software

EasyBuild supports 2995 different software packages (incl. toolchains, bundles):

```
ABAQUS    AlphaFold    Amber    ANSYS    Blender
Boost    Caffe    Cantera    CGAL    COMSOL    Eigen
FastQC    GROMACS    gmsh    HDF5    Horovod    Hypre
Julia    JupyterLap    LAMMPS    MATLAB    matplotlib
METIS    MOOSE    MUMPS    NAMD    Nektar++    netCFD
OpenBLAS    OpenFOAM    OSPRay    pandas    PAPI
ParaView    PETSc    preCICE    PyTorch    R    RSEM
Rstudio-Server    SAGE    SAMTools    Scalasca
scikit-learn    scipy    Score-P    SQLite    STAR-CCM+
TensorFlow    Valgrind    Vampire    VASP    WRF
Yade    …
```

Full list at https://docs.easybuild.io/version-specific/supported-software

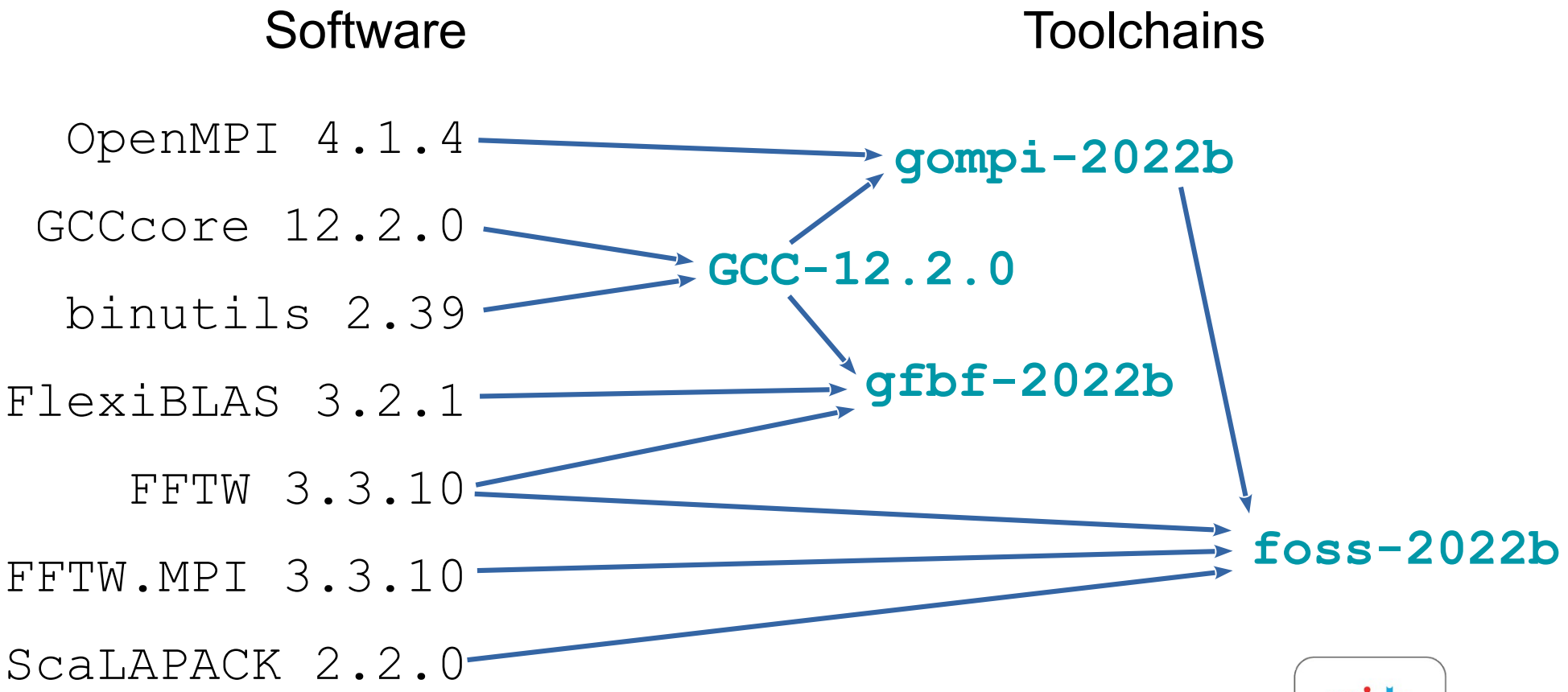
Toolchains

- Set of **compilers** and **libraries** used to build a software
- Typically
  - compilers for C, C++ and Fortran, and possibly CUDA
  - libraries for MPI, linear algebra, fast fourier transform, etc.
- Combined hierarchically to form bigger toolchains
- Defined/released twice a year → *2019a, 2019b, 2020a, 2020b, 2021a, ...*

Common toolchains

- **System** toolchain: OS compilers and libraries, used to bootstrap other toolchains
- **FOSS** toolchain consists of open source components
  → *GCC, Open MPI, FlexiBLAS with OpenBLAS, ScaLAPACK, and FFTW*
- **Intel** toolchain based on Intel software
  → *Intel C, C++ and Fortran compilers, Intel MPI, and Intel MKL libraries*

Example of toolchain: **FOSS 2022b**



EasyConfig file → recipe to build a software

Naming convention: `<name>-<version>-<toolchain><versionsuffix>.eb`

- **<name>** represents the software name;
- **<version>** represents the software version;
- **<toolchain>** represents the toolchain: `<toolchain name>-<toolchain version>`
  → omitted for the system toolchain
- **<versionsuffix>** is an optional name to represent the variants

→ *To keep in mind to search for easyconfig files*

Examples

`GCC-12.2.0.eb`

`HDF5-1.12.2-gompi-2022a.eb`

`HDF5-1.12.2-GCC-11.3.0-serial.eb`

`OpenMPI-4.1.4-GCC-12.2.0.eb`

`PyTorch-1.12.1-foss-2022a.eb`

`PyTorch-1.12.1-foss-2022a-CUDA-11.7.0.eb`

Software Installation and Environment Modules

EasyBuild will

- install each software in its own sub-directory

  → under **`<install prefix>/software/`**

- create a module file to use the software

  → under **`<install prefix>/modules/`**

- The installation prefix can be customized using the environment variable `EASYBUILD_INSTALLPATH`
- Default install prefix in `~/.local/easybuild/`
- See `https://docs.easybuild.io/configuration/`

Question?

EasyBuild at work

Practical session!

Have a look at EasyBuild documentation https://docs.easybuild.io/
or `eb --help` to understand the meaning of command line parameters.

Getting started

- Connect to Guane

```
$ ssh USERNAME@toctoc.sc3.uis.edu.co
$ ssh guane
```

- Get a computing node

```
$ srun -N 1 -n 2 -c 1 --time 2:0:0 --pty bash
```

Installing EasyBuild

- Install your own version of EasyBuild
- More details on https://docs.easybuild.io/installation/

```
$ pip3 install --user easybuild
```

- Check the installation

```
$ which eb
$ eb --version
$ eb --help
```

Configuring EasyBuild

- Installation path

```
$ export EASYBUILD_INSTALLPATH=~/sccamp2023/easybuild
```

  - Software will be installed under `~/sccamp2023/easybuild/software`
  - Modules will be installed under `~/sccamp2023/easybuild/modules`

- Check the configuration

```
$ eb --show-config
```

  - Many configuration variables → see `eb --help`
  - Can be specified via a configuration file, environment variables or command line parameters
  - See `https://docs.easybuild.io/configuration/`

Searching for available software

- Search for PyTorch

```
$ eb -S PyTorch
$ eb -S ^PyTorch.*.eb
$ eb -S ^PyTorch.*foss-2022.*.eb
```

- Search for Scikit-Learn

```
$ eb --search scikit-learn
$ eb --search scikit-learn.*2022b
```

- Search for OSU Micro Benchmark

```
$ eb -S OSU
$ eb -S OSU-Micro-Benchmarks.*2022b
```

Checking dependencies and installations

- What is required to install PyTorch?

```
$ eb PyTorch-1.12.1-foss-2022a.eb -D
$ eb PyTorch-1.12.1-foss-2022a.eb -x
```

- What is required to install Scikit-Learn?

```
$ eb scikit-learn-1.2.1-gfbf-2022b.eb -D
```

- What is required to install OSU Micro Benchmarks?

```
$ eb OSU-Micro-Benchmarks-6.2-gompi-2022b.eb -D
```

Building on top of other installations

- Rely on software installed at another location

```
$ ls -l /home/xbesseron/sccamp-2023/easybuild/
$ ls -l /home/xbesseron/sccamp-2023/easybuild/modules/
$ ls -l /home/xbesseron/sccamp-2023/easybuild/software/

$ module use /home/xbesseron/sccamp-2023/easybuild/modules/all/
```

- Check again what is required to install Scikit-Learn

```
$ eb scikit-learn-1.2.1-gfbf-2022b.eb -D
```

- Check again what is required to install OSU Micro Benchmarks

```
$ eb OSU-Micro-Benchmarks-6.2-gompi-2022b.eb -D
```

Installing a software

- Install OSU Micro Benchmarks

```
$ eb OSU-Micro-Benchmarks-6.2-gompi-2022b.eb
```

- Install Scikit-Learn

```
$ eb scikit-learn-1.2.1-gfbf-2022b.eb
$ eb scikit-learn-1.2.1-gfbf-2022b.eb --robot
```

Using an installed software

- Update the MODULEPATH

```
$ module use ~/sccamp2023/easybuild/modules/all
$ module avail
```

- Load the module for OSU Micro Benchmarks

```
$ module avail OSU
$ module show OSU-Micro-Benchmarks/6.2-gompi-2022b
$ module load OSU-Micro-Benchmarks/6.2-gompi-2022b

$ echo $EBROOTOSUMINMICROMINBENCHMARKS
$ which osu_latency
```

- Load the module for OSU Micro Benchmarks

```
$ mpirun -n 2 osu_latency
$ mpirun -n 2 osu_bibw
```