CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

ETH zürich

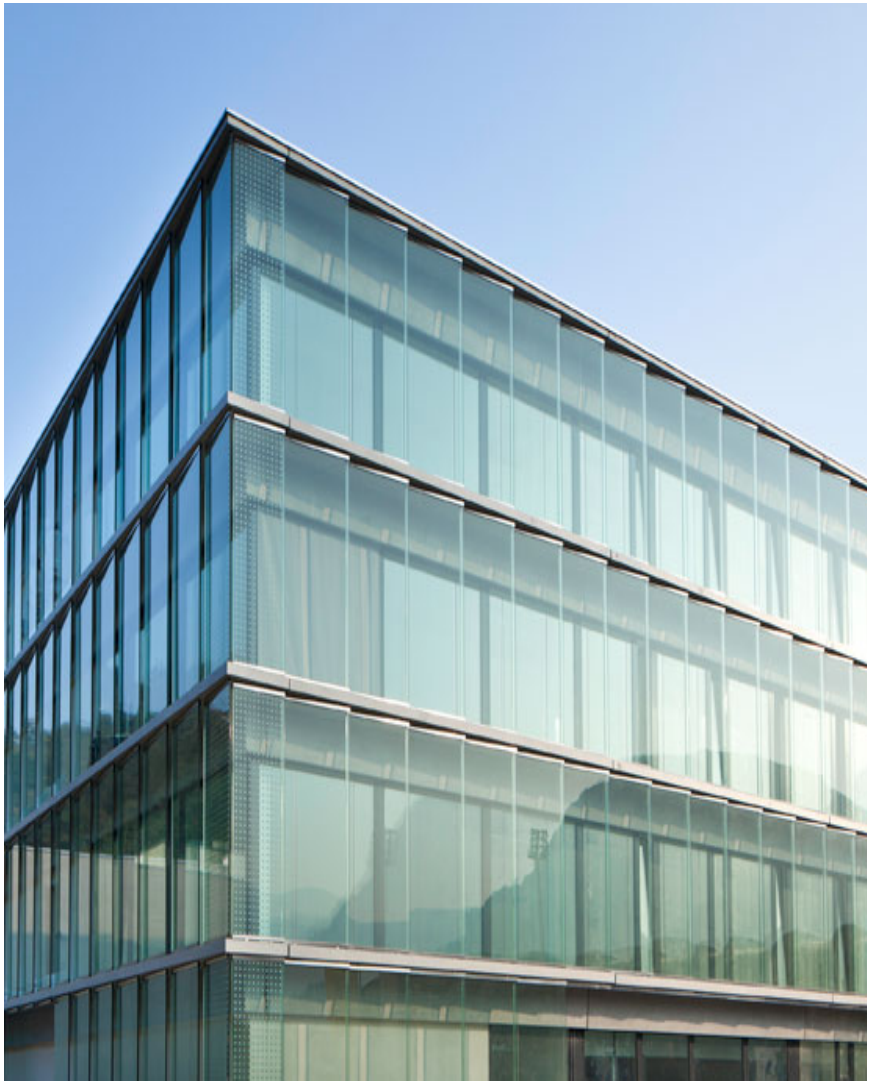
EasyBuild @ CSCS site update

3rd EasyBuild User Meeting
Jan 30th – Feb 1st 2018, Amsterdam

Guilherme Peretti-Pezzi
Theofilos Ioannis Manitaras

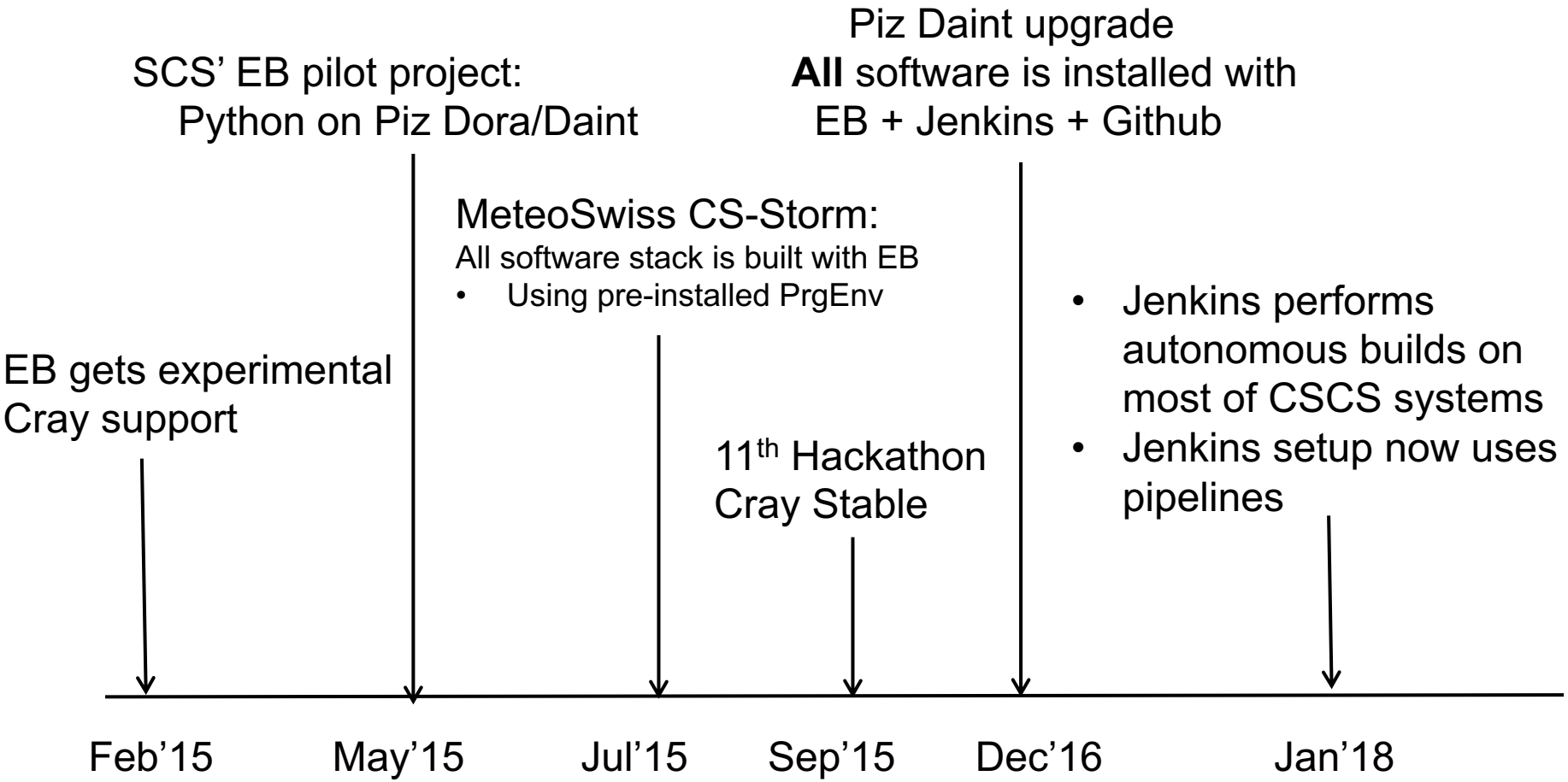
Scientific Computing Support (CSCS)

Outline



- **EB @ CSCS (Guilherme)**
  - **Timeline**
  - **Piz Kesch & Escha use case**
    - **Cray CS-Storm**
  - **Piz Daint**
    - **Cray XC**
- Using Jenkins with Easybuild at CSCS (Theo)
- Final remarks

EasyBuild timeline @ CSCS



EasyBuild Enhancements for Cray Systems

1. Support for external module files
2. Definition of Cray-specific toolchains
3. Custom easyblock for Cray toolchains

- Various smaller enhancements specific to the Cray environment

- **Thanks to Peter Forai & Kenneth Hoste**

Overview of CSCS HPC systems

System	Scope	Accelerators / node	Type
Piz Daint	User Lab	1 GPU	Cray XC50
Monch	PASC projects	0	NEC Intel IvyBridge
Escha	Meteo Swiss	16 GPU	Cray CS-Storm
Kesch	Meteo Swiss	16 GPU	Cray CS-Storm
Leone	Large Memory	1 GPU	HP DL 360 Gen 9

Piz Kesch & Escha use case (MeteoSwiss / Cray CS-Storm)

"Kesch" and "Es-cha" consist of identical systems (production and failover), each comprising:

Cray CS-Storm: 12 nodes

- 2 x Intel Haswell E5-2690v3 2.6 GHz 12-core CPUs per node
  - total of 24 E5-2690v3 processors
- 256 GB 2133 MHz DDR4 memory per node
  - total of 3 TB
- 8 NVIDIA® Tesla® K80 GPU devices per node
  - total of 192 GPUs

MeteoSwiss, the Swiss national weather forecasting service, hosts their dedicated production systems at Cray CS-Storm at CSCS, Lugano.

- EasyBuild software stack in production since September/2015
- (GCC initially provided by Cray was unable to assemble AVX2 instructions for haswell)

Piz Daint

Model	Cray XC50/XC40
XC50 Compute Nodes	Intel® Xeon® E5-2690 v3 (Haswell) @ 2.60GHz (12 cores, 64GB RAM) and NVIDIA® Tesla® P100 16GB
XC40 Compute Nodes	Intel® Xeon® E5-2695 v4 (Broadwell) @ 2.10GHz (18 cores, 64/128 GB RAM)
Login Nodes	Intel® Xeon® CPU E5-2650 v3 @ 2.30GHz (10 cores, 256 GB RAM)
Interconnect Configuration	Aries routing and communications ASIC, and Dragonfly network topology
Scratch capacity	6.2 PB (Lustre / Sonexion 3000)

Piz Daint

- #3 Top 500
  - #1 in Europe
  - 19.590 PFLOPS
- #10 Green 500
  - 10.398 MFLOPS/W

Available software on Cray using EasyBuild

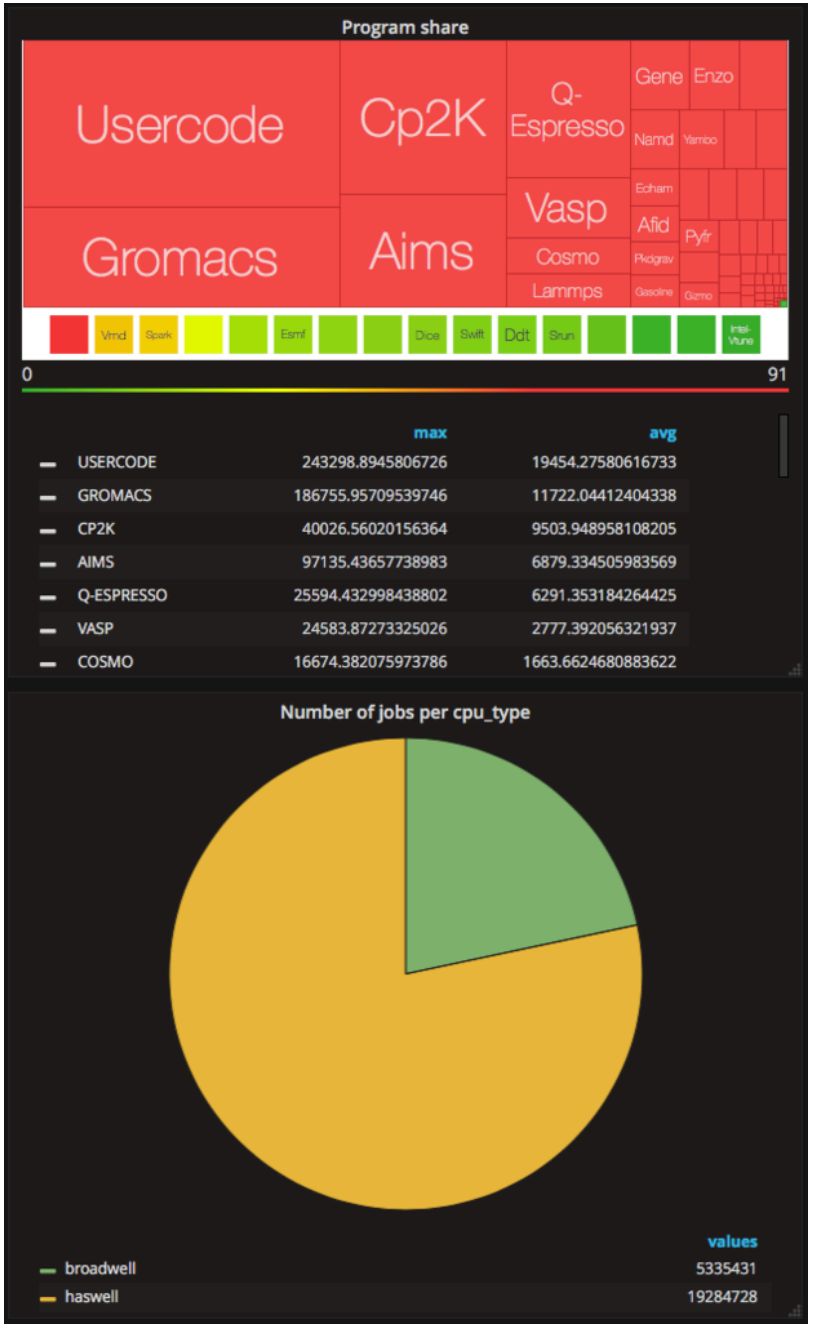
- Stock EasyBuild repository
  - Python, including
    - accelerated numpy + scipy, h5py, …
  - WRF
  - CP2K[*]
  - GROMACS[*]
  - Boost
  - GSL

[*] = GPU-enabled recipe available

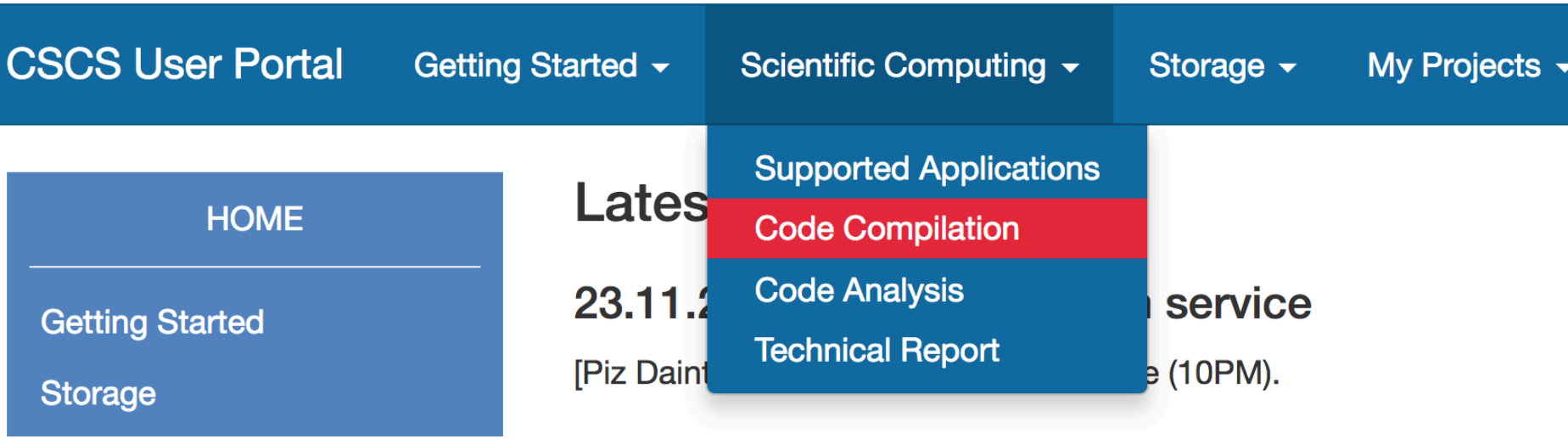
- CSCS Production Github repository
  - Amber[*]
  - CDO
  - CPMD
  - LAMMPS[*]
  - NCL
  - NCO
  - ParaView[*]
  - Octave
  - QuantumESPRESSO
  - R
  - Score-P/Scalasca/Extrae/Ipm
  - VASP[*]
  - Visit
  - VMD[*]
  - VTK[*]

  - TensorFlow[*], Theano[*]
  - CNTK[*], Caffe[*], Caffe2[*]

Apps and Users using xalt statistics



User instructions for EasyBuild (1)



User instructions for EasyBuild (2)

CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

ETH zü

CSCS User Portal | Getting Started | Scientific Computing | Storage | My Projects

SCIENTIFIC COMPUTING

Supported Applications

- ABCpy
- Amber
- CP2K
- CPMD

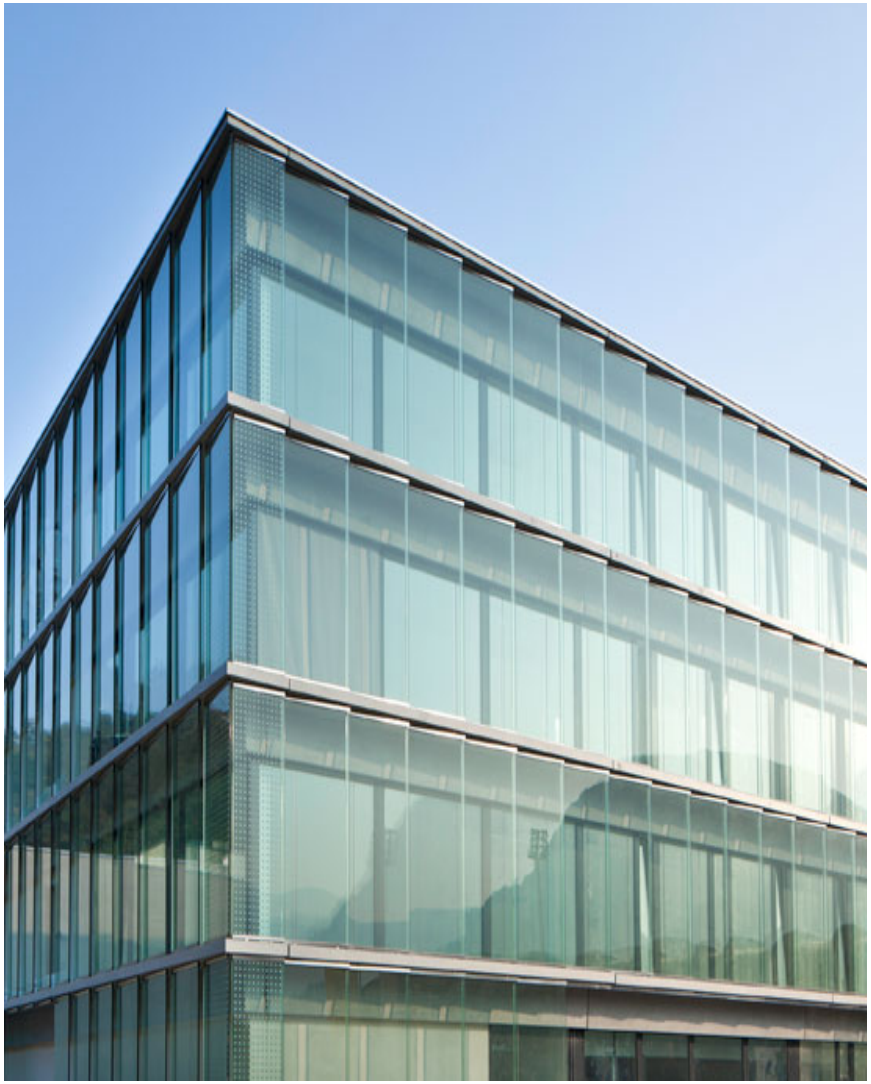
EasyBuild framework

Loading the environment

The EasyBuild framework is available at CSCS through the module `EasyBuild-custom`. This module defines the location of the EasyBuild configuration files, recipes and installation directories.

```
module load EasyBuild-custom
```

Outline



- EB @ CSCS (Guilherme)
  - Timeline
  - Piz Kesch & Escha use case
    - Cray CS-Storm
  - Piz Daint
    - Cray XC
- **Using Jenkins with Easybuild at CSCS (Theo)**
  - **Use of Jenkins with Easybuild at CSCS**
  - **Advantages of Jenkins pipelines**
  - **Demonstration of Testing pipeline**
  - **Demonstration of Regression pipeline**
- Final remarks

Jenkins at CSCS

- At CSCS Jenkins is the basic tool used for CI to test and deploy software packages which are defined as Easyconfigs and are built with Easybuild.

- Jenkins combined with Easybuild is used in three distinct projects:
  - Testing: tests a new Easyconfig submitted to Github.
  - Production: builds the software to be available as a module for users.
  - Regression: runs Easybuild for each installed Easyconfig.

- Recently we have migrated our Jenkins projects to `Pipelines` which offer:
  - Greater flexibility regarding the actions performed by Jenkins.
  - The `Jenkinsfile` of each project is version controlled.
  - Easily run the CI in parallel using all available resources.

Submitting a new PR

Contributing back

How to submit a pull request:

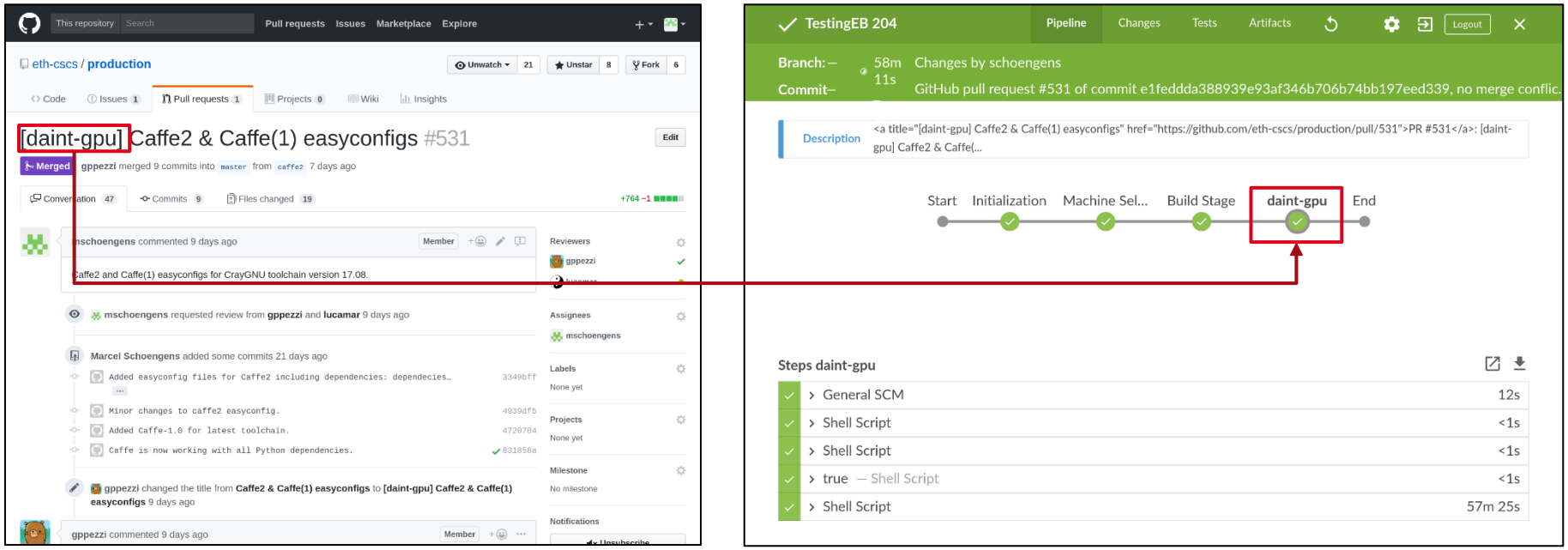
1. Add the EasyBuild configuration files to a **new branch**, including **all the required dependencies**
2. Create and **assign yourself a pull request** following this policy for the title:
   - the title **must match a supported system** in the list `daint dom kesch leone monch`, otherwise the build will fail immediately. The system names **have to be enclosed in square brackets** to be distinguished from the actual pull request title and be parsed by the corresponding Jenkins project.
   - if the title matches `WIP` ("Work In Progress"), then the test build will be aborted immediately, as work in progress is not supposed to be tested.
   - Dom and Piz Daint can test both software stacks `-gpu` and `-mc` at once:
     a. if the title matches only `${system}-gpu` or `${system}-mc`, only that software stack will be used:
        - `[dom-gpu] NAMD` will build using `-gpu`, `[dom-mc] NAMD` will use `-mc`.

     b. if the title matches both or none, then both will be used, one after another in a loop:
        - `[dom] NAMD` will build using both `-gpu` and `-mc` in a loop.
        - `[dom-gpu,dom-mc] NAMD` will do the same.
3. The CSCS Jenkins project `TestingEB` will test the build of new EasyBuild recipes with respect to the master. The corresponding pipeline of `TestingEB` is contained in the `jenkins/JenkinsfileTestingEB` script.
4. If the build is successful, you should **ask for a review**: the pull request will only be merged when approved.
5. In order to re-trigger the testing of the pull request without committing a change, add the comment `retest this please` which will notify the `TestingEB` Jenkins project.
6. (CSCS only) for production builds, please update the appropriate production build list here.

CI – Testing a new Easyconfig submitted to Github (1)

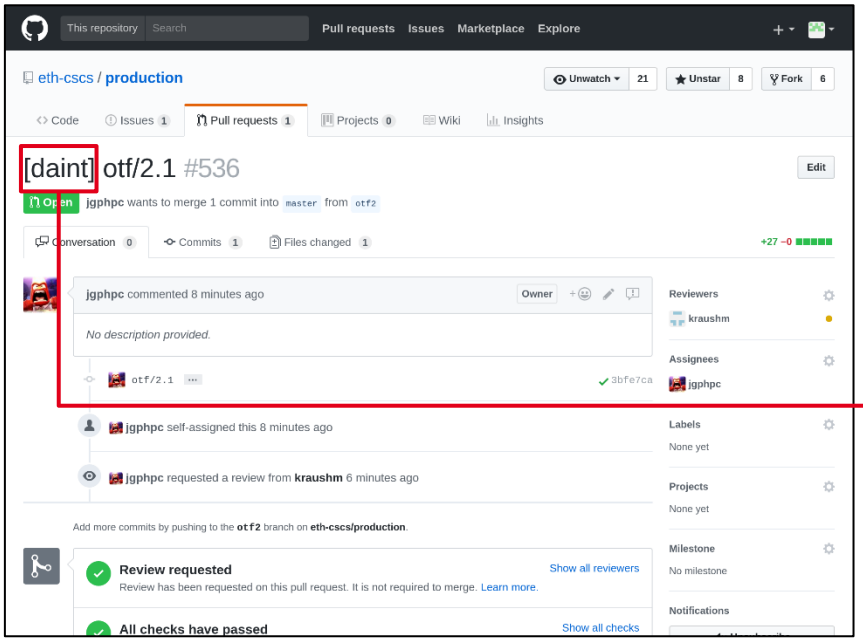
Github PR ([daint-gpu])

Triggered Pipeline

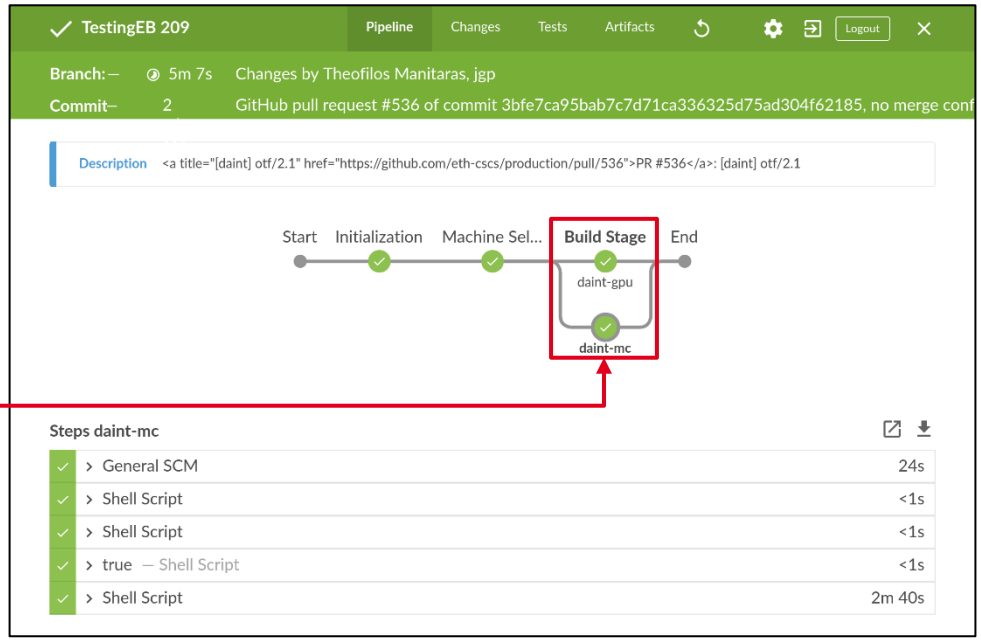


CI – Testing a new Easyconfig submitted to Github (2)

Github PR ([daint])

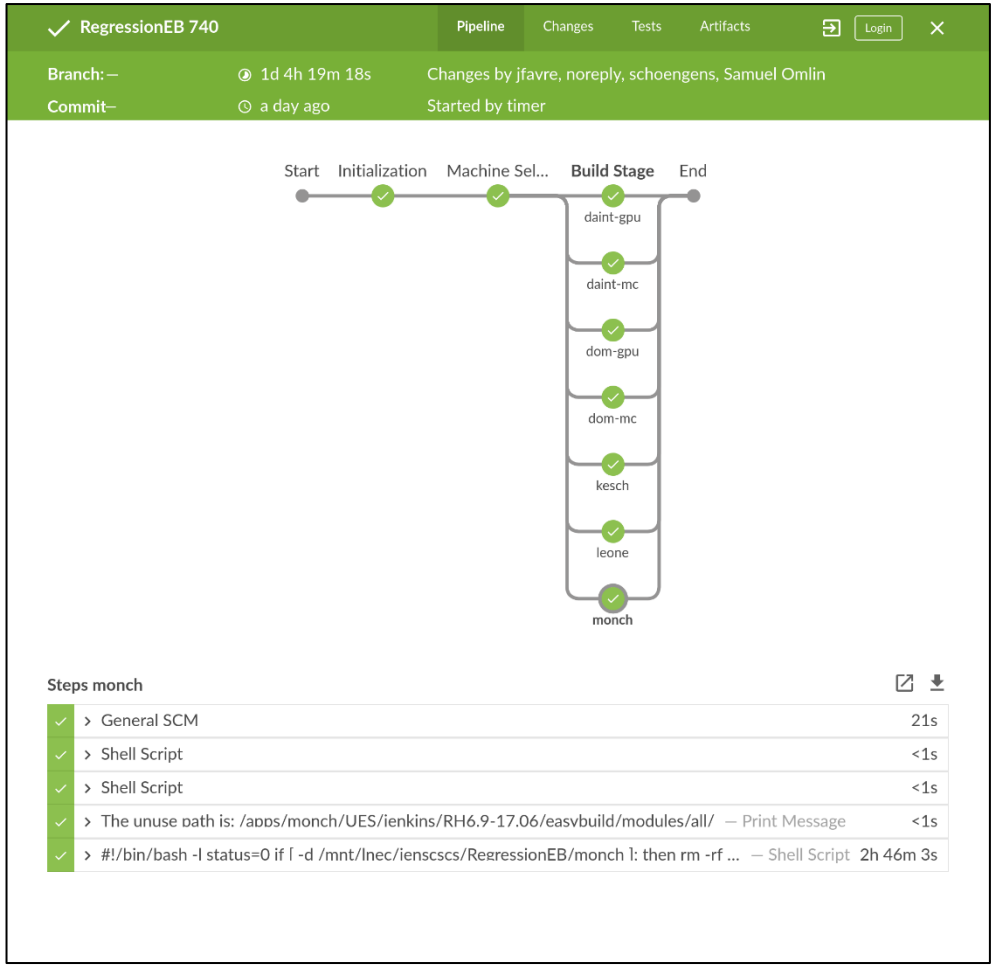


Triggered Pipeline

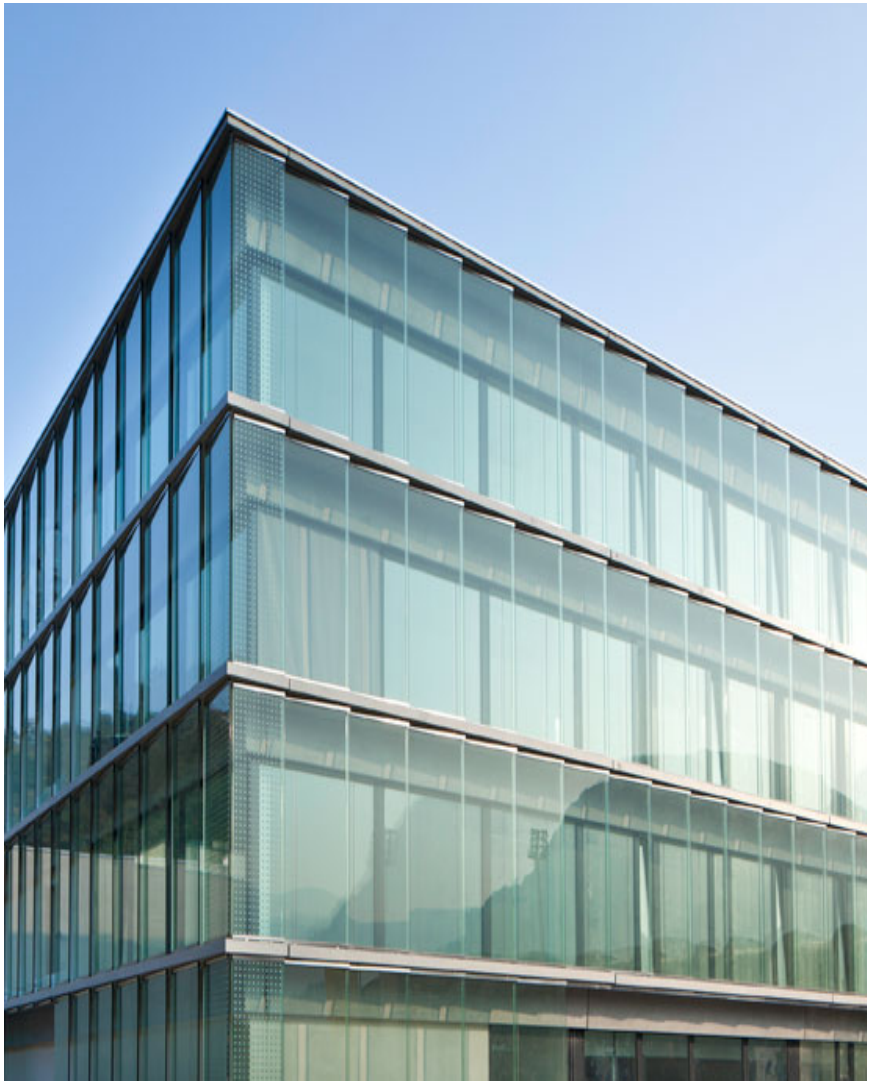


Regression Pipeline (All machines)

- Runs once a week and tests all Easyconfigs in production per machine in parallel.
- Ensure that each software package builds successfully.
- Through the Blue Ocean interface it is easy to inspect the console output of each machine.



Outline



- EB @ CSCS (Guilherme)
  - Timeline
  - Piz Kesch & Escha use case
    - Cray CS-Storm
  - Piz Daint
    - Cray XC
- Using Jenkins with Easybuild at CSCS (Theo)
- **Final remarks**

Final remarks

- Moving to EB at institution level takes time
  - Learning curve
  - Resistance to change
- EasyConfigs vs. EasyBlocks (my 5 cents)
  - EasyBlocks
    - (+) Reusable: Great for well packaged & stable software
    - (-) Too much overhead for bleeding edge software
    - (-) Reproducibility: how keep track of changes?
  - EasyConfigs
    - (-) Reuse by copy paste (=> duplication)
    - **(+++) Self contained recipes**

Final remarks (2)

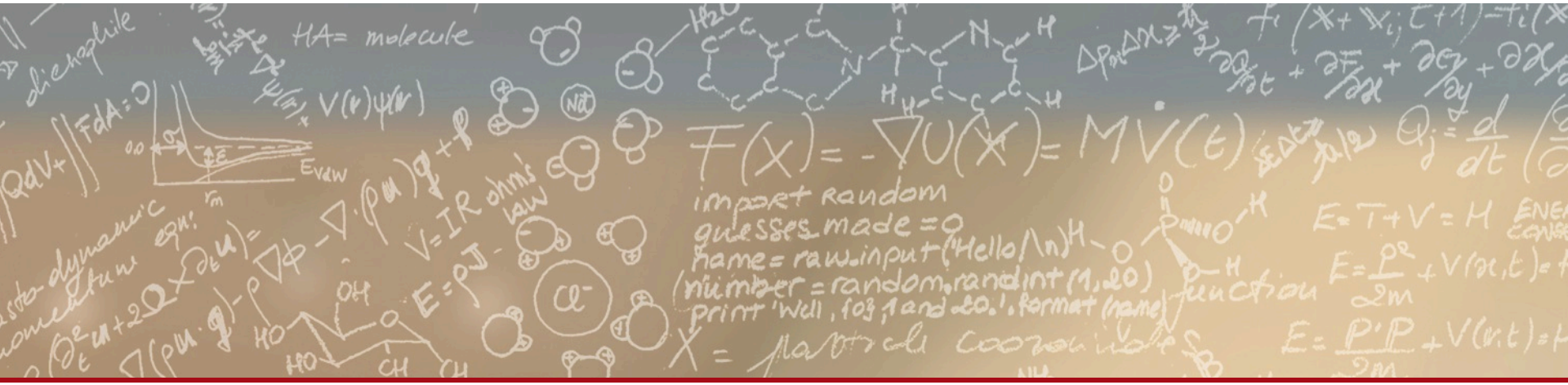
- Feature wish list
  - New command line options for dependencies
    - --try-dep-version ?
  - Have a separate log for command lines performed by EB
    - As complement to –extended-dry-run
  - Backup of custom easyblocks for reproducibility
  - External modules
    - Improve error reporting for missing modules
    - Generic/versionless entries on the metadata file

- Ongoing work
  - Deploying HPC OpenStack cluster with EB

Useful links for EasyBuild @ CSCS

- EasyBuild User Documentation at CSCS
  - https://user.cscs.ch/scientific_computing/code_compilation/easybuild_framework/

- Easyconfig files repositories
  - List of production builds performed by Jenkins
    - https://github.com/eth-cscs/production/tree/master/jenkins-builds
  - Custom easyconfigs:
    https://github.com/eth-cscs/production/tree/master/easybuild/easyconfigs
  - Custom easyblocks:
    - https://github.com/eth-cscs/production/tree/master/easybuild/easyblocks



Thank you for your kind attention