JUBE
A Flexible, Application- and Platform-Independent Environment for Benchmarking

Sebastian Lührs
s.luehrs@fz-juelich.de
Jülich Supercomputing Centre

4th JLESC Workshop, Bonn, Germany, 2 - 4 December 2015
What is JUBE?

- Generic, configurable environment to run, monitor and analyse benchmarks in a systematic way
- Developed 2008, redesigned 2014
- Also usable for testing or production scenarios

```
cpus = 2048

cpus = 4096

cpus = 8192

opt = O1

opt = O3

create result

configuration

benchmark and input data

JUBE configuration

XML

platform specific configuration

JUBE configuration

XML

analyse output

compile

opt = O1

opt = O3

execute

#cpus = 2048

#cpus = 4096

#cpus = 8192

#cpus = 2048

#cpus = 4096

#cpus = 8192

35.13

18.14

7.68

automatic workflow creation and execution

opt

#cpus

time [s]

O1 2048 35.13
O1 4096 18.14
O1 8192 9.32
O3 2048 29.00
O3 4096 15.00
O3 8192 7.68

create result

```
Why JUBE?

Alternatives:

- **Manual benchmarking:**
  - Easy to use
  - Time-consuming
  - Very error-prone

- **Benchmark specific script solution:**
  - Optimized
  - Changes can be time-consuming
  - Portability problems

---

**JUBE provides a generic workflow and parameter handling environment, but also supports more flexible and specialised approaches.**
Key Concept: Workflow creation

- Dependency driven step structure
- Parameter based expansion of steps

![Workflow Diagram](image-url)
Key Concept: Directory and data handling

- Each parameter/step combination runs in a separate sandbox directory
- Automatic directory connectivity creation

```
cppflag = -O1

Source

substitution

Makefile

Makefile.in

directory

my_exe

compile

cores = 2048

compile

cores = 4096

compile

cores = 8192

compile

```

```
cppflag = -O3

Source

substitution

Makefile

Makefile.in

my_exe

compile

cores = 2048

compile

cores = 4096

compile

cores = 8192

compile

```
Key Concept: Platform independence

- Separation of platform dependent and independent configuration options
Example Benchmark Configurations

- **HPL**: High-Performance Linpack Benchmark
  
  www.netlib.org/benchmark/hpl

- **IOR**: (InterleavedOrRandom) I/O benchmark

  sourceforge.net/projects/ior-sio

- **mdtest**: Metadata test benchmark

  sourceforge.net/projects/mdtest

Publicly available and extensible JUBE configuration file repository:

https://github.com/FZJ-JSC/jube-configs
Outlook and open Tasks

- Extend JUBE benchmark configuration repository
- Extend job system interaction
  - Allow JUBE to monitor submitted jobs more easily
- Add another configuration file input format
  - Not everyone likes writing XML files by hand
Where to start?

Download and Tutorials:
- www.fz-juelich.de/jsc/jube
- Open Source (GPLv3)

Prerequisites:
- OS: Linux
- Python 2.6, Python 2.7, Python 3.2 (or a more recent version)

Contact:
- jube.jsc@fz-juelich.de