

BENCHMARKING AND PARAMETER STUDIES WITH JUBE

28 JUNE 2019 | ILYA ZHUKOV, SEBASTIAN LÜHRS

PARAMETER STUDIES ARE A LOT OF MANUAL WORK

I have to **touch multiple files** to change all my parameters.

A longer **workflow** is needed to perform my run properly.

Each run should be executed **independently** and in a **reproducible** way.



I can't **remember the parametrization** I used in one of my runs.

I want to use a **HPC** system.

I want to **reuse as much as possible** when switching to a different platform or to a different application.

I need an **overview** about my parametrization and the results.

WHAT IS JUBE?

Generic, lightweight, configurable environment to run, monitor and analyse application execution in a systematic way

Can be used for:

- Benchmarking,
- Parameter studies,
- Testing,
- Profiling,
- Production scenarios

Alternatives

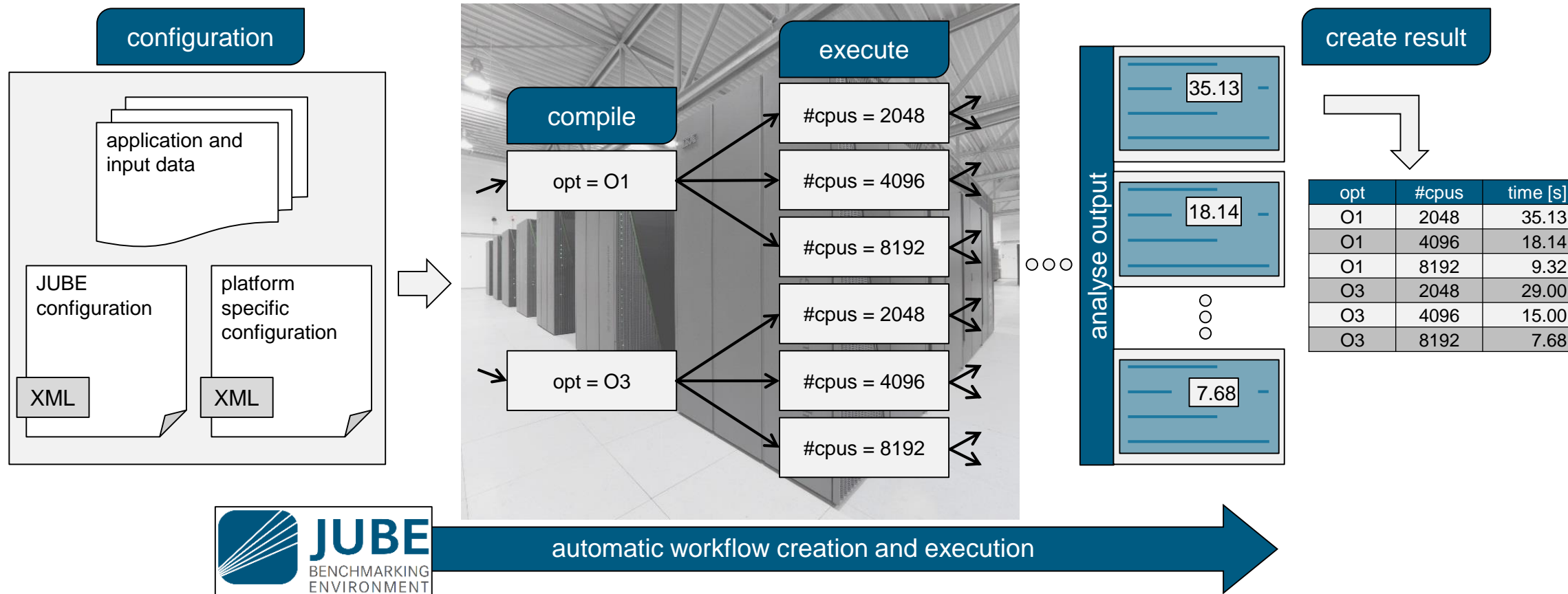
- or -

Are you a potential JUBE candidate?

- Manual execution
 - *Easy to use*
 - *Time-consuming*
 - *Very error-prone*
- Application specific script solution
 - *Optimized*
 - *Changes can be time-consuming*
 - *Portability problems*



WHAT IS JUBE?



JUBE AVAILABILITY

Download, Tutorials and Documentation:

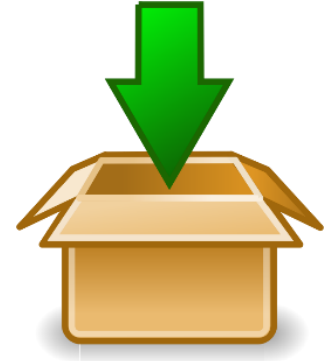
- **www.fz-juelich.de/jsc/jube**

Prerequisites:

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

Contact:

- jube.jsc@fz-juelich.de



COMMAND LINE ACCESS



Start a new benchmark run

- `jube run benchmark.xml`

Continue an existing benchmark run

- `jube continue benchmark_dir [--id <id>]`

Analyse the benchmark data

- `jube analyse benchmark_dir [--id <id>]`

Create and show result representation

- `jube result benchmark_dir [--id <id>]`

HELP?!



Online documentation and tutorial

- `www.fz-juelich.de/jsc/jube`

Info mode

- `jube info benchmark_dir [--id <id>] [--step <stepname>]`

Command line accessible glossary

- `jube help <keyword>`

Logs

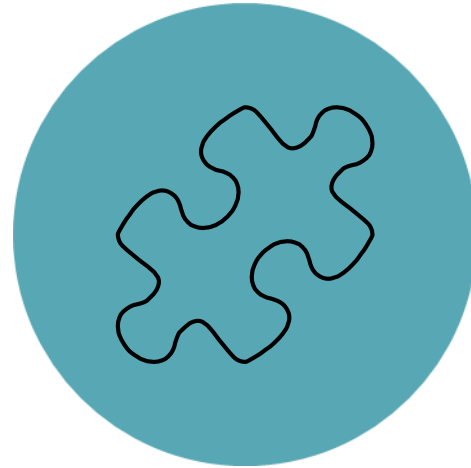
- `jube log benchmark_dir [--id <id>] [--command <cmd>]`

Debug mode

- `jube --debug run|continue|analyse|result ...`

Verbose mode

- `jube -v[vv] run ...`



HANDS-ON BT-MZ

WORKFLOW

- Download and decompress tarball
- Compile executable for specified compiler and MPI (and Score-P instrumentation)
 - **ii** – Intel and IntelMPI
 - **ip** – Intel and ParaStationMPI
 - **gp** – GCC and ParaStationMPI
- Execute scaling tests
- Analyse results
- Write report

HANDS-ON: PREPARATION



Copy prepared JUBE file

```
cp -rv /p/scratch/share/VI-HPS/examples/jube $SCRATCH/$USER  
cd $SCRATCH/$USER/jube
```

Load JUBE

```
module load JUBE
```

Examine JUBE file and adjust if necessary

```
vim jube-npb.xml
```

HANDS-ON: BT-MZ BENCHMARKING



Start a new benchmark run with Intel and IntelMPI and Score-P

```
jube run jube-npb.xml -t ii scorep
```

Continue an existing benchmark run

```
jube continue run/ [--id <id>]
```

Analyse the benchmark data

```
jube analyse run/ [--id <id>]
```

Create and show result representation

```
jube result run/ [--id <id>]
```