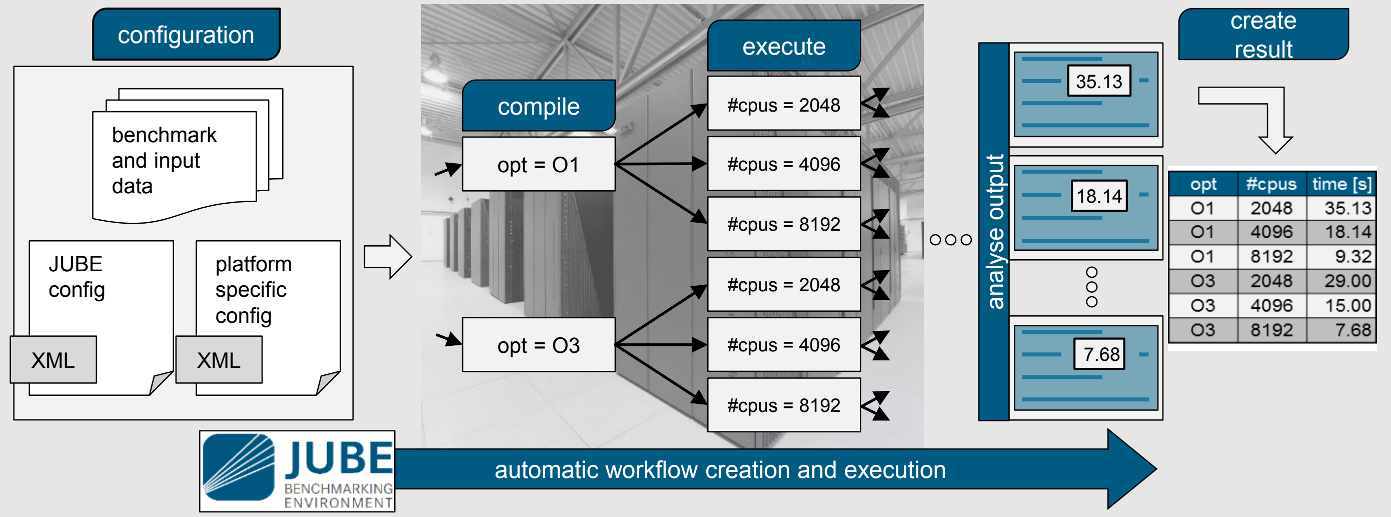
JUBE
BENCHMARKING
ENVIRONMENT

JÜLICH
FORSCHUNGSZENTRUM

JUBE

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



configuration

benchmark and input data

JUBE config XML

platform specific config XML

compile

opt = O1

opt = O3

execute

#cpus = 2048

#cpus = 4096

#cpus = 8192

#cpus = 2048

#cpus = 4096

#cpus = 8192

analyse output

35.13

18.14

7.68

create result

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

JUBE BENCHMARKING ENVIRONMENT

automatic workflow creation and execution

- JUBE considerably **reduces the cost** of configuring, running and analysing benchmark, test or production suites
- **Flexible and generic workflow management**
- XML-file based configuration allows a very **flexible parameterization**
- Use cases: Procurements, comparison of systems, monitoring effects of system and configuration changes, software testing

```
<jube>
 <benchmark name="bench" outpath="./runs">
  <parameterset name="compileset">
    <parameter name="execn">my_exe</parameter>
    <parameter name="cppflag">
     -O1,-O2
    </parameter>
  </parameterset>
  <fileset name="sources">
    <copy>src/*</copy>
  </fileset>
  <substituteset name="compilesub">
    <iofile in="Makefile.in" out="Makefile" />
    <sub source="#PROGNAME#" dest="$execn" />
  </substituteset>
  <step name="compile">
    <use>compileset,sources,compilesub</use>
    <do>make OPT=$cppflag</do>
  </step>
  <step name="execute" depend="compile">…
```

Key concepts and strategy

- Independence of applications and platforms
  ➡ **Provision of templates**
- Span a multi-dimensional parameter space
  ➡ **Substitution of placeholders**
- Workflow and directory management
  ➡ **Sandbox directory handling**
- No source code modifications and configuration files created once
  ➡ **Reproducibility**

Implementation and Availability

- Implemented in Python (compatible with Python 2.6, 2.7, 3.2 or any newer version)
- Current version: 2.1.4
- Open Source (GPLv3)
- Documentation and tutorial available online

Development

- High-level HPC queue management system configuration and communication options
- Implementation of additional workflow types (loops and groups)
- Benchmark run data reusage

Contact: jube.jsc@fz-juelich.de | Website: www.fz-juelich.de/jsc/jube

Member of the Helmholtz Association