

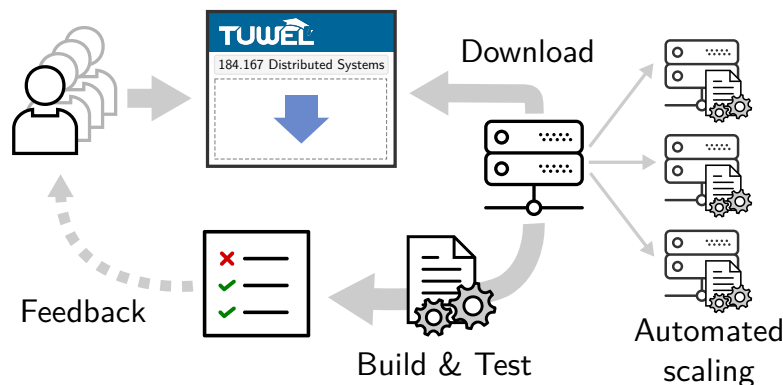
DSLabs Submission System

Scalable Automated Testing of Simple Distributed Systems Applications



FAKULTÄT
FÜR INFORMATIK
Faculty of Informatics

184.715 Project in Software Engineering & Internet Computing (12.0 EC)



Motivation

In the Distributed Systems Lab course, students create simple distributed applications in Java. Part of the grading process is to assert formal criteria of the submission and then running a set of automated tests. Currently, this is done in bulk after the submission deadline. A major problem with this approach is that, often, submissions do not adhere to formal criteria leaving individual submissions untested. This requires us to manually test the solution causing a large amount of effort and subsequent delays during the interviews. Instead, students should receive feedback on the state of their submission before the deadline.

The goal of the project is to create a system that automatically checks submissions and gives immediate feedback to students on the state of their submission. 49% of submissions happen on the day of the deadline. 14% of submissions (about 40-50) happen in the last *hour* before the deadline. A complete build and test cycle may take two to five minutes and consume a lot of resources. A single server creates a bottleneck in such a situation, and may lead to students not getting timely feedback on their submission. Using a vertical scaling approach may not be sufficient to accommodate this many test instances in parallel. To overcome this, a horizontally scalable system should be developed s.t. 40-50 test instances can run in parallel. To save resources, this scaling should be done on demand, i.e., elastically.

Work Description

- Design and implementation of a framework for automated testing of simple distributed applications written in Java
- Review of existing scalability approaches in continuous integration systems
- Design and implementation a system to elastically scale instances for isolated testing of the applications
- (Optional) TUWEL integration using the TUWEL API (creating an upload trigger)

Further Information

Start: Immediately, or during SS 2017

Basic Requirements: Excellent Java programming skills; basic understanding of scalability problems; knowledge of software engineering methods helpful

Contact:

Thomas Rausch
t.rausch@infosys.tuwien.ac.at
<http://dsg.tuwien.ac.at/staff/trausch/>

-3 -2 -1 0 1 2 3

Conceptual (Analytical)

-3 -2 -1 0 1 2 3

Empirical (Simulation)

-3 -2 -1 0 1 2 3

Practical (Implementation)

-3 -2 -1 0 1 2 3

Literature Work