



CS5990 Masterarbeit Informatik
CS3990 Bachelorarbeit Informatik

Daniel Thoma, Malte Schmitz
{thoma, schmitz}@isp.uni-luebeck.de

winter 2018/19

September 21, 2018

Integrating jUnitRV and TeSSLa for Stream Runtime Verification (SRV)

This thesis is about integrating an existing Java Bytecode instrumentation solution with jUnit and the TeSSLa compiler and interpreter to build a Stream Runtime Verification tool for Java.

jUnitRV¹ is a tool extending the unit testing framework jUnit² by runtime verification capabilities. It provides a new annotation `@Monitors` listing monitors classes that observe test case execution and verify temporal assertions. In this way, jUnit's concept of assert-based verification is extended towards checking properties of complete execution paths.

TeSSLa³ is a temporal stream-based specification language designed for specifying and analyzing the behavior of cyber-physical systems, where timing is a critical issue. TeSSLa is ideally suited for stream runtime verification (SRV) and comes with tools supporting this goal. TeSSLa supports timestamped events natively and is hence suitable for streams that are both sparse and fine-grained.

The Idea of this thesis is to use TeSSLa as a monitoring engine for jUnitRV. The current implementation uses LTL based monitors, which should be replaced by TeSSLa. Since LTL is a logic the current monitor implementation checks if the trace of the program matches the given specification. The output of the monitor are boolean values of \mathbb{B}_4 . TeSSLa is a stream processing language which derives intermediate streams from the given input stream. The program trace can be used as input stream and de-

rived streams marked as output streams are the monitoring output.

Goals

- Understand the existing Java Bytecode instrumentation and the TeSSLa monitoring framework.
- Come up with a concept on how to integrate TeSSLa streams into jUnit using the existing instrumentation solution.
- Implement a well-designed and performant tool.

Requirements

- You have a solid background in software engineering, testing and verification, as well as theoretical computer science.
- You are an experienced programmer and fluent in Java and C.
- You are at least very interested in the Java internals, e.g. Java Runtime Environment, Java Bytecode, class loaders, ...

EU Horizon 2020 Project COEMS This work is supported in part by the European Horizon 2020 project COEMS, in which international academic and industrial partners from Norway, Romania, Austria and Germany collaborate. This thesis can be done in combination with an ERASMUS stay in Norway.

¹<https://www.isp.uni-luebeck.de/junitrv>

²<http://junit.org/>

³<https://www.tessla.io/>