JOHANNES KEPLER
UNIVERSITÄT LINZ

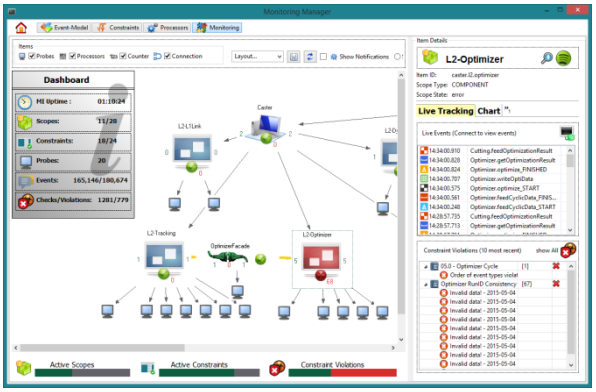
Project for a PR/Bachelor's Thesis

Priv.-Doz. Mag. Dr.
Rick Rabiser
CDL MEVSS, ISSE

T +43 732 2468 4363
F +43 732 2468 4345
rick.rabiser@jku.at
http://mevss.jku.at/rabiser

Demonstrator for the REMINDS Monitoring Framework

REMINDS (http://mevss.jku.at/reminds) [1] is a tool-supported framework for monitoring systems of systems at runtime. It comprises a flexible runtime monitoring infrastructure providing support for different roles and a requirements monitoring model covering the requirements to be monitored, the constraints checking adherence of a system's behavior to its requirements, the events and data produced by systems at runtime, and the probes instrumenting systems to intercept events and data at runtime.



So far we have applied REMINDS to monitor industrial automation systems. However, due to non-disclosure agreements we are not allowed to present all details when giving demonstrations. The goal of this thesis/project is thus to produce a demonstrator for REMINDS that allows demonstrating all of its capabilities using an open source yet realistic example.

More specifically, the goals of this thesis/project are:

- Select an example system of your choice (open source!)
  - Preferably, this is a large-scale system implemented using several different technologies.
  - We suggest something well-known such as a web browser or an office tool.
- Instrument the example system with software probes. REMINDS provides an API to develop such probes.
  - Probes are technology-specific. Depending on the technology the system to be instrumented is based on; different types of probes have to be developed. For instance, Java systems could be instrumented using Aspect-J, while C++ systems require directly modifying their code to instrument them.
  - The goal of this project/thesis is also to show that probes developed in different technologies can be integrated with REMINDS.
- Define an event model. Probes provide events and data at runtime, from the running and instrumented system. These are captured by REMINDS in an event model. You have to define such an event model using the REMINDS tool.
- Create a scope model using REMINDS that reflects the architecture of the monitored system, i.e., to group probes and constraints visually (see screenshot above).
- The main aim of monitoring is to check the behavior of a monitored system at runtime. Based on key requirements of the example system, you should define constraints to be monitored, e.g., to check event occurrence, event sequences, or event data.
  - We also suggest seeding some defects into the example system leading to constraint violations when monitored to be able to demonstrate how REMINDS detects and visualizes violations.
- Package and deploy your demonstrator. Also: produce a demo screencast/video.

[1] M. Vierhauser, R. Rabiser, P. Grünbacher, K. Seyerlehner, S. Wallner, and H. Zeisel, "ReMinds: A Flexible Runtime Monitoring Framework for Systems of Systems," Journal of Systems and Software, 2015 (http://dx.doi.org/10.1016/j.jss.2015.07.008).

JOHANNES KEPLER
UNIVERSITÄT LINZ
Altenberger Straße 69
4040 Linz, Österreich
www.jku.at
DVR 0093696