**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

**Bachelor's Thesis (or Project for a PR or Master Thesis)**

A "Minimal" Runtime Monitoring Client

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.
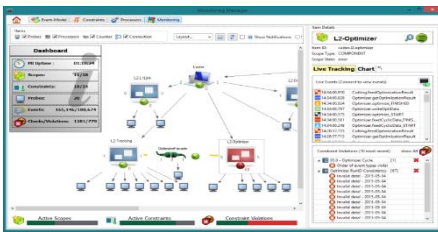
So far we have implemented various views for visualizing events and event data as well as constraint violations and diverse editors to manage the monitoring process.

The current monitoring client is an Eclipse SWT 3.8-based Rich-Client Application.

The goal of this project is to design, implement and test a simpler, more light-weight monitoring client that only provides the basic features covering most practical scenarios. Particularly, the following capabilities should be provided:

- Receive events and data from the REMINDS monitoring server.
- Visualize incoming events and data in relation with the monitored system components (instrumented by probes).
- Visualize constraint violations detected by the REMINDS constraint server.
- Visualize trends and statistics about monitored events and constraints.
- Support simple configuration (e.g., addresses and ports used for communication with REMINDS servers and other settings).

The project involves:
- Negotiating the exact requirements with the REMINDS core development team.
- Investigating/finding potential technologies for developing the monitoring client
  - It must be possible to interact with monitoring server via JMS, Socket, M-Socket or RMI
  - It must run on a Windows machine (be it in a Browser or as a Desktop application)
  - It must be a current, state-of-the-art technology that is still maintained and that is open source.
- Design a modular architecture that will allow extending the monitoring client in the future with as little effort as possible.
- Develop a prototype of the minimal monitoring client.
- Create a showcase demonstrating the capabilities of the minimal monitoring client.

[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, 2016 (http://dx.doi.org/10.1016/j.jss.2015.07.008).