

Simulation Experiment Schemas – Beyond Tools and Simulation Approaches (Software Appendix)

Pia Wilsdorf, Julius Zimmermann, Marcus Dombrowsky,
Ursula van Rienen, Adelinde M. Uhrmacher

Abstract

This archive contains the software described in the paper “Simulation Experiment Schemas – Beyond Tools and Simulation Approaches” published at the Winter Simulation Conference 2019. The tool guides users through the simulation experiment specification process via a graphical user interface. The entered information is used in the generation of backend-independent experiment specifications in the JSON¹ format. The JSON-based specifications can be mapped to different simulation backends. Currently, two domains of modeling and simulation are supported: stochastic discrete-event simulation, and finite element simulation for electromagnetics. For both domains, basic simulation runs are supported as well as complex experiment designs such as sensitivity analyses based on full factorial designs.

1 General Information

Name: Pia Wilsdorf, Julius Zimmermann, Marcus Dombrowsky,
Ursula van Rienen, Adelinde M. Uhrmacher

Email: {pia.wilsdorf, julius.zimmermann, marcus.dombrowsky, ursula.van-rienen,
adelinde.uhrmacher}@uni-rostock.de

Affiliation: Institute of Visual and Analytic Computing, University of Rostock;
Institute of General Electrical Engineering, University of Rostock

Location: Albert-Einstein-Straße 22, 18059 Rostock, Germany; Albert-Einstein-Straße 2,
18059 Rostock, Germany

Funding: This research was supported by the German Research Foundation (DFG) via the research grants UH 66/18 'GrEASE' and CRC 1270 'Elaine'.

¹<http://www.json.org>

Date: January 2020

Type: Software

Language: English

Keywords: simulation experiment, schema, template, discrete-event simulation, finite element method

Rights: CC BY-SA 4.0

2 Archive Structure

Executable: contains the executable needed to start the GrEASE² experiment generation pipeline. It also includes *sessl-quickstart* to automatically execute SESSL³/ML-Rules⁴ experiments. Resources like the experiment schemas defined using JSON Schema⁵, and the experiment templates defined in the FreeMarker⁶ template language are part of the runnable JAR.

SourceCode: contains all the Java sources.

3 System Requirements

Java Runtime Environment 8

EMStimTools⁷ (for the finite element simulations)

SALOME⁸ (for the finite element simulations)

4 Generating Simulation Experiments

The *GrEASE Experiment Generator* can be started by running the JAR file *experiment-generation-0.0.1-SNAPSHOT-jar-with-dependencies.jar*.

In the GUI, the user first has to select the modeling and simulation domain in the *Schema* tab, i.e., either *Stochastic Discrete-Event* simulation or *Finite*

²<https://gepris.dfg.de/gepris/projekt/320435134>

³Ewald, R. and Uhrmacher, A.M., 2014. SESSL: A domain-specific language for simulation experiments. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 24(2), p.11. <https://doi.org/10.1145/2567895>

⁴Maus, C., Rybacki, S. and Uhrmacher, A.M., 2011. Rule-based multi-level modeling of cell biological systems. *BMC Systems Biology*, 5(1), p.166. <https://doi.org/10.1186/1752-0509-5-166>

⁵<https://json-schema.org/specification.html>

⁶<https://freemarker.apache.org/docs/index.html>

⁷<https://github.com/j-zimmermann/EMStimTools>

⁸<https://www.salome-platform.org>

Element Method simulation for electromagnetics. Further, the user may select the *Add Experiment Design* checkbox to load schemas for various types of simulation experiments such as sensitivity analysis or statistical model checking. Depending on the selections new tabs are generated:

- Stochastic Discrete-Event:
 - Model
 - Simulation
 - Observation
- Finite Element Method:
 - Geometric Model
 - Physical Model
 - Simulation
 - Observation
- Experiment Design
 - Experiment Design

In each tab different properties of the simulation experiment can be specified, as described in the corresponding paper. Figure 1 shows the *Model* tab for the domain of stochastic discrete-event simulation.

To generate a simulation experiment specification from the entered data, the user has to click the *Generate Experiment* button. First, a JSON experiment specification is generated. If there is information missing or specified in an incorrect format, the tool will present error messages. If no further errors are detected, the completed simulation experiment specification in the specification language of the target backend is presented. Currently, experiment generation is supported for the following backends:

- Stochastic Discrete-Event: SESSL/ML-Rules
- Finite Element: EMStimTools/YAML

In addition, the *GrEASE Experiment Generator* can automatically execute the generated experiments (this is currently only supported for SESSL/ML-Rules). Alternatively, the experiment specifications can be stored on the file system via the *Menu*.

Figure 1 summarizes the main GUI components of the GrEASE Experiment Generator.

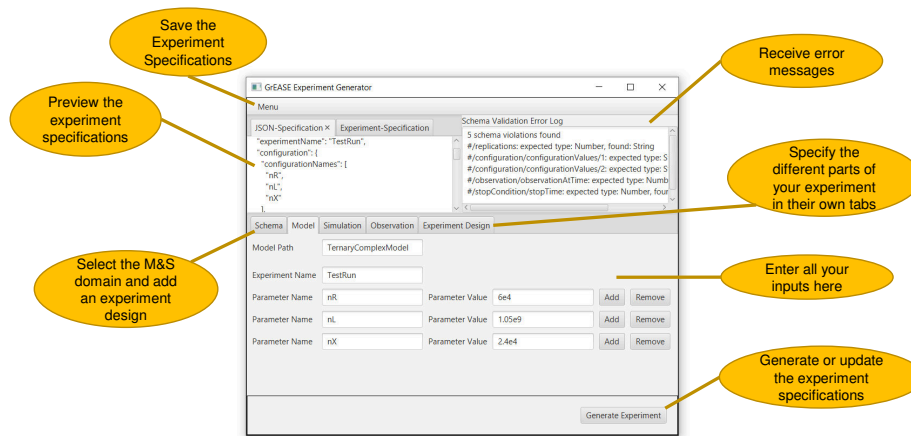


Figure 1: The GrEASE Experiment Generator guides users through the experiment specification process.