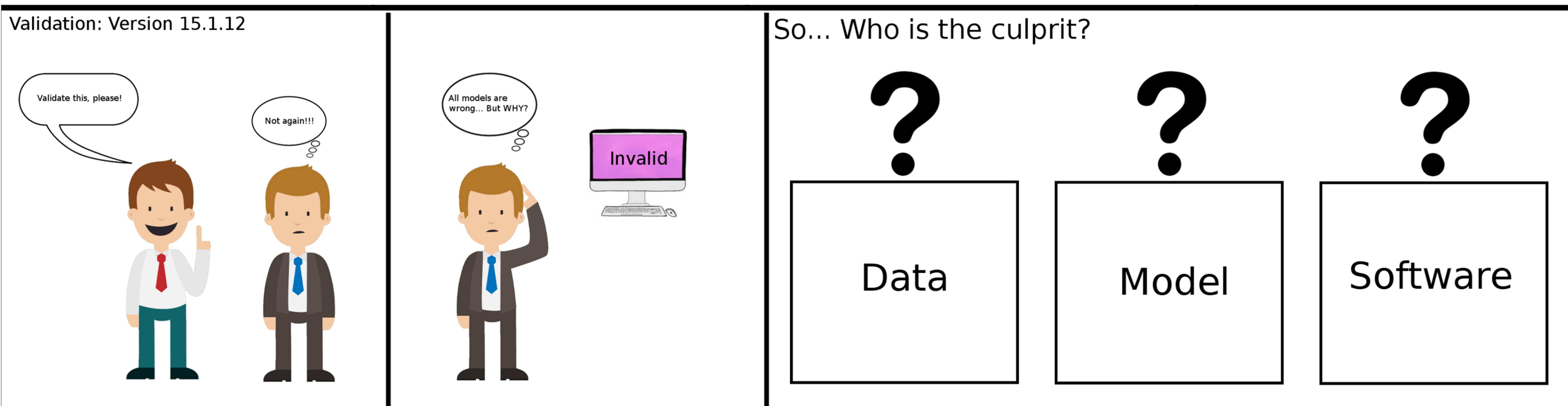


Automated Simulation Validation

B. Roungas^{@*}

[@]v.roungas@tudelft.nl, ^{*}Department of Multi-Actor Systems, Delft University of Technology, The Netherlands



HTML5
CSS3
JS
PHP
MySQL

+

=

Validation FOR DUMMIES

Metadata

Period: 01/06/2014 - 15/06/2014
Time: 16:00 - 19:00
Station: Ut
Direction: E

Difference in Average delays (seconds): 36.87

Kolmogorov-Smirnov p-value: 0

In the Kolmogorov-Smirnov test, a p-value less than 0.05 shows that the two samples are NOT distributed the same.

Chi-square p-value: 0.37989

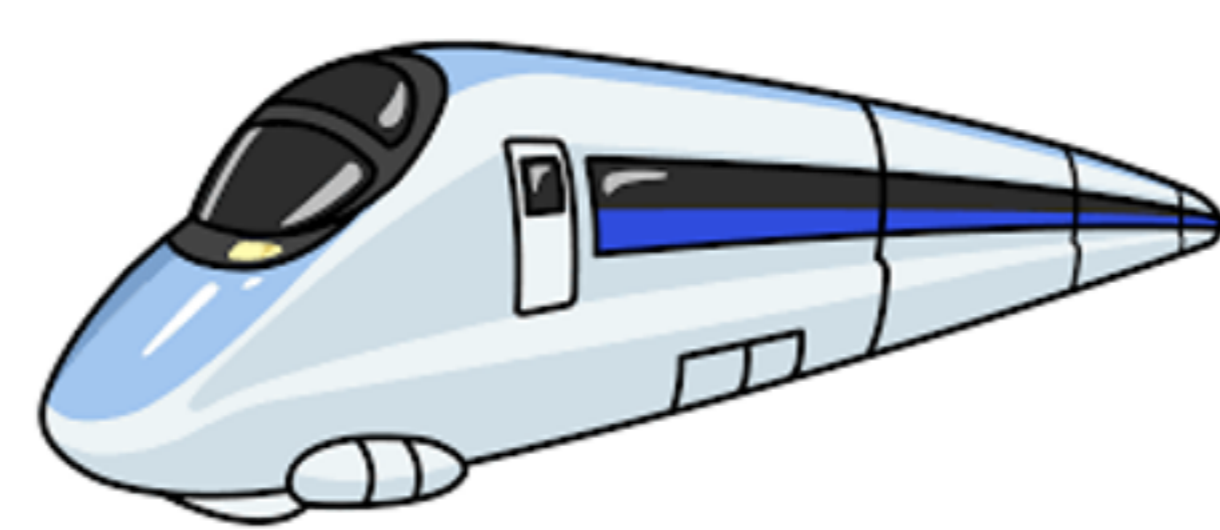
Average: 54.27
Standard Deviation: 63.66
of observations: 887
Max: 355 sec.
Min: -22 sec.

Aim

Building an automated or semi-automated procedure for validating simulation environments, similar to what unit testing is for verification. Particularly useful when the simulation environment is updated regularly.

Application

Case Study: Railway sector
Purpose: Punctuality & Safety



Data

Transformations

- **Data Cleaning:** Automatic match of train series and stations between simulated and operational data, Automatic exclusion of data with insufficient information.
- **Name & Unit Conventions:** Automatic name conversion of variables based on knowledge and content, Automatic conversion of minutes in seconds or calculation of delays based on planned and actual time.

Outlier Detection: Automatic detection and exclusion of unpredictable or suspicious delays.

Model

Requirements: Automatic checks of consistency between simulated and operational output.

Criteria for accepting Validity: Automatic input of acceptability criteria.

Type I & II error: Automatic mitigation of Type I & II error by multiple replications of the experiment.

Software

Reductio ad absurdum: Automatic detection of the common denominator among different models.

Conclusion

Front-end and back-end technologies, along with a statistical language can enable us building automated solutions for validating simulation data, models, and software environments.