

# Scenario-Based Elicitation, Specification, and Comprehension of Transient Software Behavior

Sebastian Frank<sup>1</sup>, Alireza Hakamian<sup>1</sup>, Samuel Beck<sup>1</sup>, Christoph Zorn<sup>1</sup> and André van Hoorn<sup>2</sup>

<sup>1</sup>University of Stuttgart. Stuttgart, Germany

<sup>2</sup>University of Hamburg. Hamburg, Germany

## Context

Modern, microservice-based software systems are subject to frequent change while operating in production, leading to so-called transient behavior. Changes can arise from outside, e.g., workload peaks and changing user behavior, or inside the system, e.g., autoscaling and deployment of new (versions of) services. The satisfaction of a software system's users can be heavily affected if the system is not resilient to such changes. Therefore, it is crucial to specify resilience requirements and comprehend the system's transient behavior. However, the changes causing transient behavior often happen unexpectedly, making such requirements hard to elicit. Since transient behavior can be complex and difficult to anticipate, it is also challenging for software architects to create precise and quantifiable specifications.

## Objective

Our vision in the *DiSpel* project is to develop an interactive approach for continuous elicitation, specification, and refinement of resilience requirements at runtime in the domain of microservice-based software systems. As part of this vision, we aim to support stakeholders in eliciting and specifying precise and quantifiable resilience requirements by developing new (interactive) techniques and adopting established techniques from other domains, e.g., risk analysis, visualization, and human-computer interaction.

## Method

In our work, we use scenarios as described by the Architecture Tradeoff Analysis Method (ATAM) [1] as the foundation for specification, since they represent both essential parts of a resilience requirement: the incidents and the expected response. We aim to interactively transform the elicitation-friendly scenarios into more formal representations, e.g., Probabilistic Computation Tree Logic [2], in order to disambiguate them and make them quantifiable.

---

SSP'21: Symposium on Software Performance, November 9–10, 2021, Leipzig, Germany



© 2021 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

We use system traces to automatically extract architectural information and hazards as starting points for the elicitation process. Additionally, we combine visualizations of the architecture and transient behavior with assistance by chatbots. For the visualization and specification of the expected response, we rely on resilience curves — our recent works are based on Bruneau’s resilience triangle [3] as a simplification.

## Results

We have achieved the following complementary intermediate results towards our overall vision:

- Resirio [4] — a tool that is able to import execution traces and conducts a CHAZOPS-based [5] hazard analysis on the extracted architectural model. The results are presented in a graph-based architecture visualization intended to help users in interactive elicitation of resilience scenarios. In a conversation with a chatbot, the user can then specify the resilience scenario. We conducted a user study with participants from industry and academia to evaluate Resirio’s usability, effectiveness, and support. The evaluation shows that the developed prototype gives novice requirements engineers a foundation for fast requirements elicitation and that Resirio complements traditional requirements engineering approaches.
- TransVis [6] — a tool focused on the specification of expected responses and comprehension of transient behavior. TransVis presents a graph-based visualization of the system’s services that indicates potential violations. For each service, the violations and measurements can be visually investigated and compared against simple specifications based on Bruneau’s resilience triangle model. An integrated chatbot assists the user in her tasks, e.g., adding, deleting, and showing specifications. A user study revealed that the developed visualizations are effective for specifying and exploring transient behavior. However, the chatbot was rarely utilized by the participants.
- Currently, we are also developing concepts and editors for the interactive transformation of ATAM-based scenarios into more formal representations. While still under development, early versions of some of the planned prototypes are already available. This step is intended to extend the requirement elicitation workshop, which we presented in a previous work [7].

## Conclusion

While the evaluations of Resirio and TransVis showed that they can support users in the elicitation and specification of resilience requirements as well as comprehension of transient behavior, they also revealed some limitations. The chatbots were not well accepted by the participants — except for quick replies — and only simple scenarios can be specified. Therefore, we aim to enhance and extend our concepts to support the specification of more sophisticated scenarios.

## Talk Outline and Further Resources

In our talk, we will give an overview of the vision and current activities in the *DiSpel* project for elicitation and specification of resilience requirements. We present preliminary results, i.e., the concepts, prototypes, evaluation results, and lessons learned regarding Resirio and TransVis. Furthermore, we provide an insight into the currently developed editors.

## Acknowledgement

This research is funded by the Baden-Württemberg Stiftung (Orcas project) and the German Federal Ministry of Education and Research (Software Campus 2.0 – Microproject: DiSpel).

## References

- [1] R. Kazman, M. Klein, M. Barbacci, T. Longstaff, H. Lipson, J. Carriere, The architecture tradeoff analysis method, in: Proc. 4th IEEE Int. Conf. on Engineering of Complex Computer Systems (ICECCS), 1998, pp. 68–78.
- [2] H. Hansson, B. Jonsson, A logic for reasoning about time and reliability, Formal aspects of computing 6 (1994) 512–535.
- [3] M. Bruneau, S. E. Chang, R. T. Eguchi, G. C. Lee, T. D. O’Rourke, A. M. Reinhorn, M. Shinozuka, K. Tierney, W. A. Wallace, D. Von Winterfeldt, A framework to quantitatively assess and enhance the seismic resilience of communities, Earthquake spectra 19 (2003) 733–752.
- [4] C. Zorn, Interactive Elicitation of Resilience Scenarios in Microservice Architectures, Master’s thesis, University of Stuttgart, Germany, 2021.
- [5] J. Earthy, Hazard and operability studies as an approach to software safety assessment, in: IEE Computing and Control Division Colloquium on Hazard Analysis, 1992.
- [6] S. Beck, S. Frank, M. A. Hakamian, L. Merino, A. van Hoorn, TransVis: Using visualizations and chatbots for supporting transient behavior in microservice systems, in: 2021 Working Conference on Software Visualization (VISSOFT), IEEE, 2021, pp. 65–75.
- [7] S. Frank, A. Hakamian, L. Wagner, D. Kesim, J. von Kistowski, A. van Hoorn, Scenario-based resilience evaluation and improvement of microservice architectures: An experience report, in: 5th International Workshop on Formal Approaches for Advanced Computing Systems (FAACS@ECSA), 2021.