Jens Knoop, Uwe Zdun (Hrsg.): Software Engineering 2016,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2016 93

A DSL-Based Approach for Event-Based Monitoring of Systems of Systems

Michael Vierhauser[1], Rick Rabiser[1], Paul Grünbacher[2], Alexander Egyed[2]

Abstract: Complex software-intensive systems such as systems of systems (SoS) need to be monitored at runtime to detect deviations from their requirements. In our earlier work [Vi15a] – summarized in this paper – we described our experiences of developing and applying an SoS monitoring approach based on a Domain-specific Language (DSL) in the domain of industrial automation software. More specifically, we have been developing a constraint DSL for industrial end users as well as an incremental constraint checker for event-based monitoring. Our evaluation demonstrates the expressiveness of our DSL and the scalability of the checker in an industrial scenario.

Keywords: Runtime Monitoring, Systems of Systems, Constraint DSL.

1 Summary

Many software-intensive systems today are systems of systems comprising heterogeneous and independently developed yet interrelated elements. As the full behavior of SoS emerges during operation only, system testing is not sufficient to determine compliance with requirements. Instead, the behavior of the systems and their interactions need to be continuously monitored and checked during operation to detect and analyze deviations from the expected behavior. Checks include the occurrence and order of runtime events (temporal behavior), the interaction of systems (structural behavior), or properties of runtime data (data checks).

Despite the wide variety of existing runtime monitoring approaches, most of these only support particular technologies or certain types of constraints and checks, impeding their application to SoS. In earlier work [Vi15a] we have described our experiences of extending an existing incremental consistency checker for design models [Vi10] to support event-based runtime monitoring of SoS [Vi15b]. Our work is motivated by an industrial case of monitoring a metallurgical plant automation system, an example of an SoS.

More specifically, we have been developing a domain-specific constraint language aiming at industrial end users, who often lack deep programming skills, to ease the definition of various types of constraints. Our DSL-based approach (cf. Fig. 1) allows incrementally checking constraints at runtime. This ensures that violations of requirements

[1] Johannes Kepler Unversität Linz, Christian Doppler Labor MEVSS, Altenberger Str. 69, 4040 Linz, michael.vierhauser@jku.at
[2] Johannes Kepler Unversität Linz, Institut für Software Systems Engineering, Altenberger Str. 69, 4040 Linz, paul.gruenbacher@jku.at

can be reported instantly to users monitoring an SoS. The approach further supports the definition and deployment of constraints at runtime, i.e., constraints can be added or modified without stopping the checker or the monitored systems.

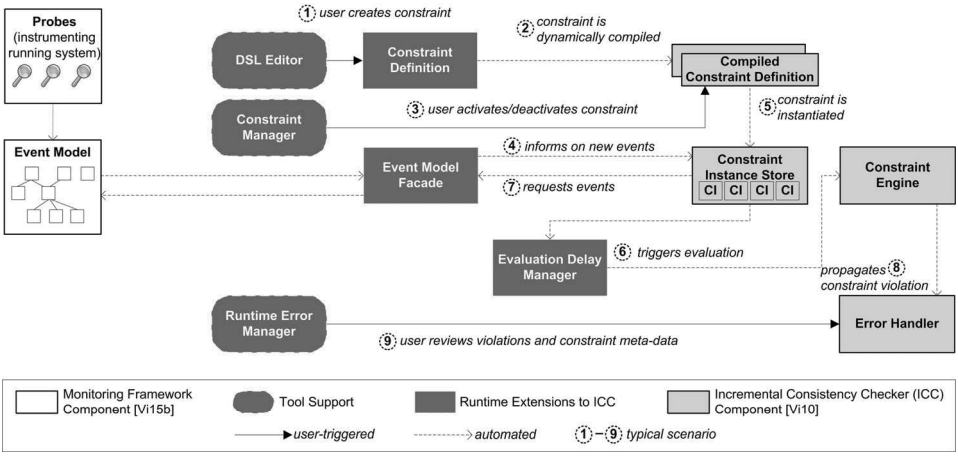

Fig. 1: Our DSL-based approach to define and check constraints at runtime [Vi15a].

In [Vi15a] we demonstrated the expressiveness of the DSL using real constraints from an industrial case. We further evaluated the scalability of our checker in an industrial monitoring scenario. Our experiences suggest designing an extensible constraint DSL in an iterative manner and keeping it as simple as possible. We also suggest keeping the mapping of the DSL to the checking engine flexible to gain independence of underlying checking technologies. Industrial scenarios demonstrate the need to add new or modify existing constraints even while the system and the monitoring infrastructure are running, e.g., if investigating newly emerging and therefore unforeseen issues.

References

[Vi15a] Vierhauser, M., Rabiser, R., Grünbacher, P., Egyed, A.: Developing a DSL-Based Approach for Event-Based Monitoring of Systems of Systems: Experiences and Lessons Learned. Proc. of the 30th IEEE/ACM Int'l Conf. on Automated Software Engineering, Lincoln, Nebraska, USA, ACM, 2015.

[Vi10] Vierhauser, M., Grünbacher, P., Egyed, A., Rabiser, R., Heider, W.: Flexible and Scalable Consistency Checking on Product Line Variability Models. Proc. of the 25th IEEE/ACM Int'l Conf. on Automated Software Engineering, Antwerp, Belgium, ACM, 2010, pp. 63-72.

[Vi15b] Vierhauser, M., Rabiser, R., Grünbacher, P., Seyerlehner, K., Wallner, S., Zeisel, H.: ReMinds: A Flexible Runtime Monitoring Framework for Systems of Systems. Journal of Systems and Software, 2015 (doi: 10.1016/j.jss.2015.07.008).