

MatchBox: A Framework for Dynamic Configuration of Service Matching Processes

Marie Christin Platenius¹ Svetlana Arifulina² Wilhelm Schäfer³

This submission is based on our publication published in the proceedings of the 18th Int. ACM Sigsoft Symposium on Component-Based Software Engineering [PSA15].

1 Motivation

The increasing popularity of paradigms like service-oriented computing and cloud computing is leading to a growing amount of service providers offering software components in the form of deployed, ready-to-use services (Software as a Service, SaaS) [14, 20]. In order to discover and select software services, intermediaries apply service matching approaches for determining whether the specification of a provided service satisfies the requester's requirements. There are already lots of different service matching approaches considering different service properties (structural, behavioral, and non-functional properties). However, each of these approaches alone is not enough to provide a high matching result quality (e.g., accurate matching results) [BOR04].

Thus, such approaches should be combined into a more holistic approach leading to more accurate matching results. However, this combination is a manual, error-prone procedure where many design decisions are made. Furthermore, this procedure has to be repeated frequently depending on the context, e.g., to consider different requesters or markets.

2 Comprehensive Matching Processes

To cope with the challenge of combining service matching approaches, we propose our framework MatchBox. MatchBox supports reusing existing matching approaches and combining them in a model-driven way based on a reconfigurable model of the matching process (see a simplified version in Figure 1). Using this model and well-defined descriptions of the matching components, MatchBox takes care of control and dataflow.

The modeled processes can be executed automatically on a set of requested and provided service specifications as inputs. For this purpose, the matching process model is interpreted

¹ Paderborn University, Heinz Nixdorf Institute, Software Engineering Group, Zukunftsmeile 1, 33102 Paderborn, m.platenius@upb.de

² Formerly: Paderborn University, Databases and Information Systems, Zukunftsmeile 1, 33102 Paderborn, Svetlana.Arifulina@upb.de

³ Paderborn University, Zukunftsmeile 1, 33102 Paderborn, wilhelm@upb.de

and complex matching results that visualize and aggregating the results of all performed matching steps are returned to the user.

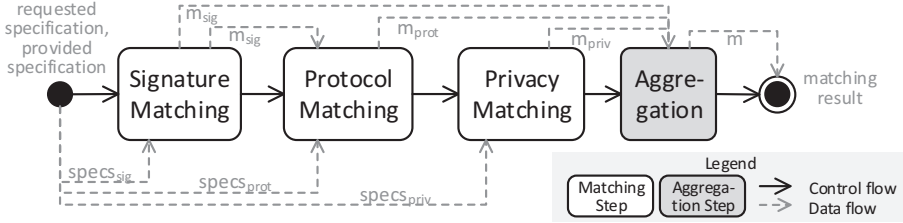


Fig. 1: Example Matching Process, based on [PSA15]

As a case study, we let computer science students integrate eleven matchers into MatchBox. The results show that the MatchBox framework remains flexibility (in terms of the variational scope of matching approaches) and reduces effort (in terms of lines of code and needed time) of matching component integration at the same time.

3 Fuzzy Matching Processes

The MatchBox framework can also be applied within the scope of Fuzzy Matching [PI16], which is a concept to deal with imprecise and incomplete specifications. Such specifications arise from the fact that requesters can have vague requirements, while providers are not always able and willing to specify details about their services. Fuzzy matching approaches being integrated into MatchBox matching processes deliver not only matching degrees as results but also an estimation of how uncertain the matching result is due to such imperfect specifications. Thereby, the risk of a wrong decision is made visible such that MatchBox' comprehensive matching results support the challenge of service selection even better.

Acknowledgements: This work was supported by the German Research Foundation (DFG) within the Collaborative Research Center On-The-Fly Computing" (CRC 901).

References

- [BOR04] Becker, Steffen; Overhage, Sven; Reussner, Ralf: Classifying Software Component Interoperability Errors to Support Component Adaption. In: Proc. of the 7th Int. Symp. on Component-Based Software Engineering (CBSE'04). volume 3054 of LNCS. Springer Berlin Heidelberg, pp. 68–83, 2004.
- [PI16] Platenius, Marie Christin: Fuzzy Matching of Comprehensive Service Specifications. PhD thesis, Paderborn University, 2016.
- [PSA15] Platenius, Marie Christin; Schäfer, Wilhelm; Arifulina, Svetlana: MatchBox: A Framework for Dynamic Configuration of Service Matching Processes. In: Proc. of the 18th Int. ACM SIGSOFT Symp. on Component-Based Software Engineering (CBSE'15). ACM, pp. 75–84, 2015.