Verifying Dynamic Architectures using Model Checking and Interactive Theorem Proving

Diego Marmsoler[1]

With the emergence of mobile and adaptive computing, dynamic architectures have become increasingly important. In such architectures, components can appear and disappear, and connections between their ports can change, both over time [Br14]. Thus, the state space of such architectures is changing dynamically, which makes their verification challenging.

To address this problem, we propose an approach based on *model checking* and *interactive theorem proving* (Fig. 1). To this end, verification is split into two parts: component types and component integration. The restricted state space of single component types makes them amenable to automatic verification techniques, such as model checking, and since their implementation changes frequently, they benefit most from the fast feedback provided by such techniques. The correctness of the integration of verified components, on the other hand, requires axiomatic reasoning and thus it is best done using interactive theorem proving. The additional effort induced by such techniques is justified by the robustness of verification results at the integration level: they remain valid as long as components fulfill the specification of their types.

To implement the approach, we developed FACTum: a framework for the axiomatic specification of dynamic architectures. In FACTum, data types are specified using *algebraic specification techniques*. Component types are then specified by means of *state machines* and associated *assertions* about their behavior in terms of first order LTL formulae. Finally, component integration is specified by means of *architectural assertions*: first order LTL formulae over component variables, using dedicated predicates to denote component activation and interconnection. *Architecture diagrams* complement these techniques with a graphical notation to specify interfaces and certain activation/connection constraints. A FACTum specification comes with a formal, denotational semantics in terms of sets of *architecture traces* [MG16a, MG16b]. To support the specification process, we implemented FACTum in Eclipse/EMF. FACTum Studio [MG18] supports the development of FACTum specifications with rigorous type checking mechanisms. Moreover, it allows to generate corresponding NuSMV models [MD17] and Isabelle/HOL theories [Ma18b] from a FACTum specification. To further support the interactive verification of component integration, we developed a calculus to reason about dynamic architectures [Ma17b, Ma17c] and implemented it in Isabelle/HOL [Ma17a, Ma18a].

[1] Technische Universität München, Fakultät für Informatik, Bolzmannstraße 3, Deutschland, diego.marmsoler@tum.de, `https://marmsoler.com/`

doi:10.18420/se2019-52

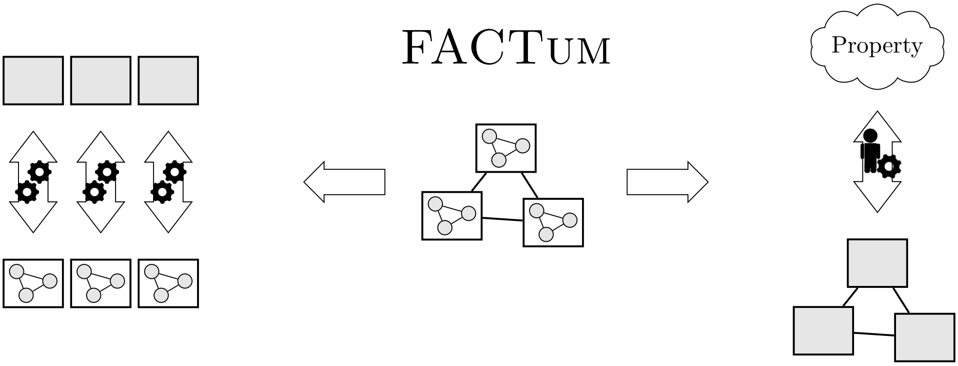

Abb. 1: Combined approach for the verification of dynamic architectures.

So far, we successfully applied the approach in four case studies [Ma18c]. First, we applied it to verify three well-known patterns for dynamic architectures: the *Singleton*, the *Publisher-Subscriber*, and the *Blackboard*. To demonstrate FACTUM's support for *hierarchical verification*, the Publisher-Subscriber pattern was modeled as an instance of the Singleton pattern and the Blackboard pattern as an instance of the Publisher-Subscriber. Thus, results obtained from the verification of lower-level patterns are automatically available to support the interactive verification of higher-level patterns. In another study, we applied the approach for the specification and verification of *Blockchain architectures*. The project consisted of roughly 3500 lines of Isabelle/HOL code, which demonstrates its feasibility for larger cases.

Keywords: Formal Methods, Dynamic Architectures, Interactive Theorem Proving, Model Checking, FACTUM

Literaturverzeichnis

[Br14] Broy, Manfred: A Model of Dynamic Systems. In (Bensalem, Saddek; Lakhneck, Yassine; Legay, Axel, Hrsg.): From Programs to Systems. The Systems Perspective in Computing, Jgg. 8415 in Lecture Notes in Computer Science, S. 39–53. Springer Berlin Heidelberg, 2014.

[Ma17a] Marmsoler, Diego: Dynamic Architectures. Archive of Formal Proofs, Juli 2017. `http://isa-afp.org/entries/DynamicArchitectures.html`, Formal proof development.

[Ma17b] Marmsoler, Diego: On the Semantics of Temporal Specifications of Component-Behavior for Dynamic Architectures. In: Eleventh International Symposium on Theoretical Aspects of Software Engineering. Springer, 2017.

[Ma17c] Marmsoler, Diego: Towards a Calculus for Dynamic Architectures. In (Hung, Dang Van; Kapur, Deepak, Hrsg.): Theoretical Aspects of Computing - ICTAC 2017 - 14th International Colloquium, Hanoi, Vietnam, October 23-27, 2017, Proceedings. Jgg. 10580 in Lecture Notes in Computer Science. Springer, S. 79–99, 2017.

[Ma18a] Marmsoler, Diego: A Framework for Interactive Verification of Architectural Design Patterns in Isabelle/HOL. In: The 20th International Conference on Formal Engineering Methods, ICFEM 2018, Proceedings. 2018.

[Ma18b] Marmsoler, Diego: Hierarchical Specication and Verication of Architecture Design Patterns. In: Fundamental Approaches to Software Engineering - 21th International Conference, FASE 2018, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2018, Thessaloniki, Greece, April 14-20, 2018, Proceedings. 2018.

[Ma18c] Marmsoler, Diego: A Theory of Architectural Design Patterns. Archive of Formal Proofs, März 2018. `http://isa-afp.org/entries/Architectural_Design_Patterns.html`, Formal proof development.

[MD17] Marmsoler, Diego; Degenhardt, Silvio: Verifying Patterns of Dynamic Architectures using Model Checking. In: Proceedings International Workshop on Formal Engineering approaches to Software Components and Architectures, FESCA@ETAPS 2017, Uppsala, Sweden, 22nd April 2017. S. 16–30, 2017.

[MG16a] Marmsoler, D.; Gleirscher, M.: On Activation, Connection, and Behavior in Dynamic Architectures. Scientific Annals of Computer Science, 26(2):187–248, 2016.

[MG16b] Marmsoler, Diego; Gleirscher, Mario: Specifying Properties of Dynamic Architectures using Configuration Traces. In: International Colloquium on Theoretical Aspects of Computing, S. 235–254. Springer, 2016.

[MG18] Marmsoler, Diego; Gidey, Habtom Kahsay: FACTUM Studio: A Tool for the Axiomatic Specification and Verification of Architectural Design Patterns. In: Formal Aspects of Component Software - FACS 2018 - 15th International Conference, Proceedings. 2018.