Verifying Component and Connector Models against Crosscutting Structural Views (extended abstract)

Shahar Maoz
School of Computer Science
Tel Aviv University, Israel

Jan Oliver Ringert, Bernhard Rumpe
Software Engineering
RWTH Aachen University, Germany

Abstract: The structure of component and connector (C&C) models, which are used in many application domains of software engineering, consists of components at different containment levels, their typed input and output ports, and the connectors between them. C&C views, presented in [MRR13], can be used to specify structural properties of C&C models in an expressive and intuitive way.

This extended abstract reports on [MRR14], where we addressed the verification of a C&C model against a C&C view and presented efficient (polynomial) algorithms to decide satisfaction. A unique feature of our work, not present in existing approaches to checking structural properties of C&C models, is the generation of witnesses for satisfaction/non-satisfaction and of short natural-language texts, which serve to explain and formally justify the verification results and point the engineer to its causes.

A prototype tool and an evaluation over four example systems with multiple views, performance and scalability experiments, as well as a user study of the usefulness of the witnesses for engineers, demonstrate the contribution of our work to the state-of-the-art in component and connector modeling and analysis.

The structure of component and connector (C&C) models consists of components at different containment levels, their typed input and output ports, and the connectors between them. C&C models are used in many application domains of software engineering, from cyber-physical systems to web services to enterprise applications, as they offer a physically distributed computation model and a logically distributed development process [BR07].

In recent work [MRR13] we have presented *component and connector views*, as a new means to specify structural properties of component and connector models in an expressive and intuitive way. C&C views take advantage of novel abstraction mechanisms for hierarchy and connectivity, not present in comparable languages. These mechanisms allow different stakeholders to create views that express their partial knowledge about the structure of the system at hand, corresponding to different use cases, functions, or concerns.

In [MRR14] we considered specification and documentation usage scenarios. In the first, C&C views denote constraints derived from (partial) knowledge of the system under development. An architect is given these C&C views, describing mandatory, alternative, and negative structural properties, and is responsible for building a C&C model that satisfies them. In the second scenario, the views highlight design decisions and document how specific concerns are addressed using potentially crosscutting solutions in the model.

Unlike the views, a C&C model is complete: it includes all components and connectors, with all ports names and types. It is ready for implementation, e.g., for direct code generation. Thus, given a C&C model, one is interested in verifying whether it satisfies each of

the C&C views in its specification or documentation.

In [MRR14] we focused on the verification of C&C models against C&C views, and presented three contributions:

- First, we defined and implement an efficient (polynomial) algorithm for the structural verification of a C&C model against a C&C view.
- Second, we extended the verification algorithm to not only decide satisfaction, but also, importantly, to generate small model witnesses and short natural language texts that formally justify and explain the verification results to the engineer.
- Finally, we reported on an evaluation of our work over several example C&C model systems, taken from different sources and of different domains, and several C&C views specifications consisting of many views, both in terms of the performance and scalability of our algorithms, and in terms of its usefulness to engineers.

In [MRR13], we introduced C&C views and discussed the synthesis problem: given a C&C views specification, consisting of mandatory, alternative, and negative views, construct a concrete satisfying C&C model, if one exists. Synthesis is powerful, but it suffers from scalability limitations. In [MRR14], we complemented this previous work by focusing on the dual problem of verification.

As a concrete language for C&C models we use the textual ADL MontiArc [HRR12] developed with the MontiCore [KRV10] framework. The C&C views are defined as an extension to C&C models. Our implementation of C&C views verification, example C&C models, C&C views, and generators for synthetic examples are available from [www]. Details on the C&C views and C&C models language, the verification algorithm, and its correctness proves are provided in [Rin14].

References

[BR07]   Manfred Broy and Bernhard Rumpe. Modulare hierarchische Modellierung als Grundlage der Software- und Systementwicklung. *Informatik Spektrum*, 30(1):3–18, 2007.

[HRR12]   Arne Haber, Jan Oliver Ringert, and Bernard Rumpe. MontiArc - Architectural Modeling of Interactive Distributed and Cyber-Physical Systems. Technical Report AIB-2012-03, RWTH Aachen, february 2012.

[KRV10]   Holger Krahn, Bernhard Rumpe, and Steven Völkel. MontiCore: a framework for compositional development of domain specific languages. *STTT*, 12(5):353–372, 2010.

[MRR13]   Shahar Maoz, Jan Oliver Ringert, and Bernhard Rumpe. Synthesis of Component and Connector Models from Crosscutting Structural Views. In Bertrand Meyer, Luciano Baresi, and Mira Mezini, editors, *ESEC/SIGSOFT FSE*, pages 444–454. ACM, 2013.

[MRR14]   Shahar Maoz, Jan Oliver Ringert, and Bernhard Rumpe. Verifying Component and Connector Models against Crosscutting Structural Views. In Pankaj Jalote, Lionel C. Briand, and André van der Hoek, editors, *ICSE*, pages 95–105. ACM, 2014.

[Rin14]   Jan Oliver Ringert. *Analysis and Synthesis of Interactive Component and Connector Systems*. Number 19 in Aachener Informatik-Berichte, Software Engineering. Shaker Verlag, Aachen, Germany, December 2014.

[www]   C&C Views Materials. http://www.se-rwth.de/materials/cncviews/.