

# Automated Model Translations for Vehicular Real-Time Embedded Systems with Preserved Semantics

Saad Mubeen\*, Mikael Sjödin\*, Jukka Mäki-Turja\*<sup>†</sup>, Kurt-Lennart Lundbäck<sup>†</sup> and Peter Wallin<sup>‡</sup>

\*Mälardalen University, Sweden

<sup>†</sup> Arcticus Systems

<sup>‡</sup> Volvo Construction Equipment, Sweden

saad.mubeen@mdh.se

## Extended Abstract

Model-based development of software architecture for real-time embedded systems in modern vehicles has had a surge in the last few years. While the introduction of models into the development of real-time embedded systems has increased efficiency in some parts of the engineering process, the models are also cause of novel concerns. In particular, mismatch between structural and semantic assumptions in modeling languages used in different parts of the design process of such systems cause large problems when design artifacts are transformed between modeling languages.

In the industry, productivity is hampered by incompatible tools and file-formats, in conjunction with the need for non-trivial, manual and tedious translations between different model-formats. Moreover, these translations are done in ad hoc fashion making the result of the translation unpredictable and potentially with altered semantics. There is a strong need to investigate how to effectively and efficiently work with existing modeling languages for real-time embedded systems. A solution must entail possibilities to make tools inter-operable to allow automated (and semi-automated) translations between modeling languages and tools with preserved semantics.

In this paper, we present the work in progress on bridging the semantic gap that exists between such models that are used for the development of real-time embedded systems in the segment of construction-equipment vehicles. Benefits that are sought include; use of precise and unambiguous notations to describe complex features, preservation of timing properties after translations, faster turn-around times in early design-phases, possibilities to automatically derive test-cases and possibilities to automatically generate code.

### A. Main Goals and Contributions

We are investigating how research oriented and/or standardized component models intended for the automotive real-time embedded systems can be used together with component models actually used in the industry today to provide both a functional description of the system as well as providing an analyzable and a resource-efficient model.

**1) Model inter-operability:** Since different modeling languages support different types of expressions, it is often impossible to define a one-to-one mapping between different constructs in different languages. However, if designers choose to use a certain subset of the full expressiveness of a language, or choose to use a certain style of expression in the language, it can be possible to define unambiguous

mappings from those subsets or styles to other languages. Such subsets or styles can be expressed by a set of patterns. Our goal is to identify such patterns to allow expression of common solutions in a transformable style.

**2) Automated model translation:** Given above identified patterns that can be translated with preserved semantics, we need to find out which of these can be automatically translated and derive the corresponding translations.

**3) Identification of non-translatable models:** We also need to identify when a construct cannot be translated with preserved semantics. This functionality is useful both to find errors in the model, and to prevent erroneous translations with (potentially subtle) modifications to model semantics.

### B. Research Plan and Current Work

In particular, we attack the gap between functional models (expressed in standard languages as EAST-ADL [3] and/or proprietary languages such as Simulink or Statemate) and execution models (expressed in standard languages like AUTOSAR [2], TADL [4] and TIMMO [5] and/or proprietary languages like Rubus Component Model, RCM [1, 7, 6]) used for the development of software for real-time embedded systems. We plan to extract a large enough set of patterns from existing solutions at the partner companies and existing literature. Development of automated translations and also automatic detection of design-patterns that do not allow unambiguous transformation will be the second step. Based on these automated translations, it will be possible to assemble a seamless tool chain for the development of software for vehicular real-time embedded systems.

## References

- [1] Arcticus Systems. <http://www.arcticus-systems.com>.
- [2] AUTOSAR Overview, Rel.4.0, Ver.1.2.1., Rev.3, 2012.
- [3] EAST-ADL Domain Model Specification, D4.1.1.
- [4] Timing Augmented Description Language, Ver.2, D6.
- [5] TIMMO Methodology, Ver.2, D7. October 2009.
- [6] S. Mubeen, J. Mäki-Turja, and M. Sjödin. Support for Holistic Response-time Analysis in an Industrial Tool Suite: Implementation Issues, Experiences and a Case Study. In *19th IEEE Conference on Engineering of Computer Based Systems*, pages 210–221, April 2012.
- [7] S. Mubeen, J. Mäki-Turja, M. Sjödin, and J. Carlson. Analyzable modeling of legacy communication in component-based distributed embedded systems. In *37th Euromicro Conference on Software Engineering and Advanced Applications*, pages 229–238, Sep. 2011.