# Framework for Heterogeneous Modeling and Composition

Matias Ezequiel Vara Larsen, Julien Deantoni, Frédéric Mallet

▶ **To cite this version:**

## HAL Id: hal-01073202
## https://hal.science/hal-01073202

Submitted on 9 Oct 2014

# Framework for Heterogeneous Modeling and Composition [*]

Matias Ezequiel Vara Larsen[1]
, Julien DeAntoni[1]
and Frédéric Mallet[1]

Université Nice Sophia Antipolis,
I3S, INRIA, AOSTE
varalars@i3s.unice.fr, julien.deantoni@polytech.unice.fr, frederic.mallet@unice.fr

## Abstract

Embedded and cyber-physical systems are becoming more and more complex. They are often split into subsystems and each subsystem can be addressed by a Domain Specific Modeling Language (DSML). A DSML efficiently specifies the domain concepts as well as their behavioral semantics. For a single system, several models conforming to different DSMLs are then developed and the system specification is consequently heterogeneous, *i.e.*, it is specified by using heterogeneous languages. The behaviors of these models have to be coordinated to provide simulation and/or verification of the overall system. This thesis studies this coordination problem with the objective of enabling the execution and the verification of the heterogeneous systems, by relying on the behavioral composition of DSMLs. We are currently proposing a language named B-COoL (Behavioral Composition Operator Language) to specify the composition of DSMLs behavior. The aim of B-COoL is two-fold: to capitalize the coordination at the language level, and to enable the automatic generation of the coordination between models conforming to composed DSMLs.

## 1 Introduction

The development of embedded and cyber-physical systems is becoming more and more complex. Such systems involve the interactions of very different subsystems, *e.g.*, DSP, Sensors, Analog-Digital devices. The requirements of each subsystem are usually captured by using a Domain Specific Modeling Language (DSML). An heterogeneous system is then specified using several DSMLs. To simulate and verify heterogeneous systems, models conforming to different DSMLs have to be coordinated.

Existing approaches [3, 1] allow the user to hierarchically coordinate heterogeneous models. These approaches provide a unique abstract syntax and the behavioral semantics is brought by using a specific Model of Computation (MoC). For instance, in [5], Ptolemy [3] has been used to embed hierarchical finite state machines within different concurrent models of computations. Although these approaches are good for experimenting, they hide the coordination semantics in the tool. This limits the study and the tuning of the coordination. More importantly, they do not provide any specific language to help in the specification of the coordination.

Differently, coordination languages [7] allow the user to manually specify the coordination between models. Thus, the user can explicitly define the coordination between two models written in different languages. There are two kinds of coordination languages: *endogenous* and *exogenous*. Endogenous languages provide primitives that must be incorporated within a model

---

for its coordination. In exogenous languages such primitives are without a model. In any case, since the coordination depends on the coordinated models, it has to be redone for each pair of models.

There is a need to *capitalize* the specification of the coordination between models at the language level by defining a language dedicated to the composition of language behavior. Current structural composition languages [4, 6] allow the user to specify composition rules at the language level. These rules are then applied between models. Such composition rules rely on two major steps: *matching* and *merging*. The matching decides *what* elements of the syntax are selected and *when* (*i.e.*, under which condition) the merging is applied. The merging specifies *how* the selected elements are combined. Similarly, a behavioral composition language relies in a matching step that selects behavioral elements of models. However, the merging step is replaced by a *coordination step* that specifies how the selected elements must be coordinated.

We propose the Behavioral Composition Operator Language (B-COoL) to specify the behavioral composition of languages. B-COoL allows the user to define operators that define composition rules between elements of the semantics of one or more language(s). This specification can be used to automatically generate the coordination between two (or more) models. Currently, B-COoL relies on a behavioral interface of the languages in terms of Events as proposed in [2]. First experiments of coordination have been based on the use of CCSL as illustrated by the coordination of a Finite State Machine and a fUML model [8].

# References

[1] F. Boulanger and C. Hardebolle. Simulation of Multi-Formalism Models with ModHel'X. In *Proceedings of ICST'08*, pages 318–327. IEEE Comp. Soc., 2008.

[2] Benoit Combemale, Julien Deantoni, Matias Vara Larsen, Frédéric Mallet, Olivier Barais, Benoit Baudry, and Robert France. Reifying Concurrency for Executable Metamodeling. In Richard F. Paige Martin Erwig and Eric van Wyk, editors, *6th International Conference on Software Language Engineering (SLE 2013)*, Lecture Notes in Computer Science, Indianapolis, Etas-Unis, 2013. Springer-Verlag.

[3] J. Eker, J. W. Janneck, E. A. Lee, J. Liu, X. Liu, J. Ludvig, S. Neuendorffer, S. Sachs, and Y. Xiong. Taming heterogeneity – the Ptolemy approach. *Proc. of the IEEE*, 91(1):127–144, 2003.

[4] Franck Fleurey, Benoit Baudry, Robert France, and Sudipto Ghosh. Models in software engineering. chapter A Generic Approach for Automatic Model Composition, pages 7–15. Springer-Verlag, Berlin, Heidelberg, 2008.

[5] A. Girault, Bilung Lee, and E.A. Lee. Hierarchical finite state machines with multiple concurrency models. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 18(6):742–760, 1999.

[6] Dimitrios S. Kolovos, Richard F. Paige, and Fiona A. C. Polack. Merging models with the epsilon merging language (eml. In *In Proc. ACM/IEEE 9th International Conference on Model Driven Engineering Languages and Systems (Models/UML 2006*, 2006.

[7] George A. Papadopoulos and Farhad Arbab. Coordination models and languages. Technical report, Amsterdam, The Netherlands, The Netherlands, 1998.

[8] Matias Vara Larsen and Arda Goknil. Railroad Crossing Heterogeneous Model. In *GEMOC workshop 2013 - International Workshop on The Globalization of Modeling Languages*, Miami, Florida, USA, September 2013.