

## Editorial to the theme issue on multi-level modeling

Colin Atkinson<sup>1</sup> · Thomas Kühne<sup>2</sup> · Juan de Lara<sup>3</sup>

Received: 19 September 2016 / Accepted: 29 September 2016 / Published online: 13 October 2016  
© Springer-Verlag Berlin Heidelberg 2016

Multi-level modeling, i.e., the explicit use of multiple levels of classification in modeling, is a conservative extension of the well-established, traditional two-level object-oriented paradigm. Two-level object-oriented technology has been tremendously successful in both modeling (e.g., UML) and programming (e.g., Java). However, it has been shown that attempting to capture certain domains or systems with only two classification levels (i.e., objects and their types) results in accidental complexity that stems from an impedance mismatch between the subject at hand and the solution technology used to capture it [2]. Examples for domains that can be much more elegantly captured using multiple classification levels are biological taxonomies, process (meta-) modeling, enforced software architectures, and systems with dynamic type levels [4].

In particular, two-level technologies suffer from a simplistic type/instance dichotomy. Modeling elements must either play the role of a type or the role of an instance but not both. This means that types themselves cannot be treated as first class citizens of a system, i.e., as instances of (meta-) types, and thus cannot themselves be added dynamically to a system in a type-safe way.

### History of multi-level modeling

The use of multiple levels of classification arranged in a classification hierarchy has a long history. Success stories include compiler compilers, the EIA/CDIF standard, and the UML's four-layer architecture. These applications are based on the idea of language definition, i.e., they take a linguistic view to classification, and are referred to as cases of “metamodeling” [3].

Since 2001, however, a new type of organizing classification hierarchies that mirror logical classification levels in the domain has been receiving increasing attention. The underlying classification principle is known as “ontological classification.” Initially referred to as “multi-level meta-modeling,” this alternative strand of (meta-)modeling is increasingly being referred to as “deep modeling” in order to distinguish it from the historically more prevalent language-engineering driven “metamodeling.” A variety of approaches offering disciplines and supporting mechanisms, e.g., for “deep characterization,” have been proposed based on the general idea of modeling domains with minimal accidental complexity.

### Multi-level modeling today

In 2016, the third international workshop on multi-level modeling “MULTI 2016” was held, a testimony to the steady growth of a research community focused on furthering the state of the art in modeling with multiple domain classification levels. Multi-level modeling has been demonstrated to be an elegant replacement for many ad hoc constructs (e.g., stereotypes in the UML) and workarounds (e.g., the “Type Object” pattern), addressing real needs that were previously only supported by partial and inelegant solutions.

✉ Colin Atkinson  
atkinson@informatik.uni-mannheim.de

Thomas Kühne  
Thomas.Kuehne@ecs.vuw.ac.nz

Juan de Lara  
Juan.deLara@uam.es

<sup>1</sup> University of Mannheim, Mannheim, Germany

<sup>2</sup> Victoria University of Wellington, Wellington, New Zealand

<sup>3</sup> Universidad Autónoma de Madrid, Madrid, Spain

Researchers are, however, still continuing to propose new supporting foundations, mechanisms, and tools [1].

## In this issue

An indispensable prerequisite for understanding classification hierarchies as they occur in multi-level modeling is a thorough understanding of classification itself. Only once classification itself has been properly understood and formalized, and its historical usage patterns in other domains have been explored for their relevance to modeling, does it become possible to create a well-founded notion of multi-level modeling. In *Formalization of the classification pattern: survey of classification modeling in information systems engineering*, Partridge et al. investigate the properties of the classification pattern in particular by using the notion of a powerset. They examine the formal underpinnings of the powerset concept in the context of modern set theory and perform a literature survey on the formalization of classification. The latter reveals three major formalization strands which are then discussed in detail.

In multi-level modeling, classification is obviously the most fundamental inter-level relationship. However, it is also important to fully understand intra-level relationships and to consider the potential utility of further inter-level relationships. In *Toward a well-founded theory for multi-level conceptual modeling*, Carvalho and Almeida present a theory for multi-level modeling defined using first-order logic that includes structural inter-level relationships that aim at more precisely capturing classification patterns. Their additional, so-called cross-level, relationships are powertype, categorization, and partitioning. With respect to intra-level relationships, they cover both specialization and subordination.

The value users of multi-level modeling technology receive depends directly on the semantic richness of the supported mechanisms. The richer the semantics of multi-level mechanisms and the better adapted they are to their intended usage, the more their users will benefit. In *Dual deep modeling: multi-level modeling with dual potencies and its formalization in F-Logic*, Neuymayr et al. refine the existing notion of “potency”—a mechanism to support deep characterization—to the notion of “dual potency.” They present a multi-view approach to multi-level modeling based on the ANSI/SPARC architecture in which traditional, two-level models, at the external level, are extracted from a single, integrated multi-level model at the conceptual level. The key innovation in the approach is the inclusion of properties possessing dual potencies to allow the depth of characterization

to be expressed separately for the source and target. The proposed approach is formalized using F-Logic/Flora-2.

As valuable as a theoretically well-founded understanding of multi-level modeling is, the overall picture is incomplete without a demonstration that respective techniques can be applied at industrial scales. In order to stand a chance of wider adoption, multi-level modeling approaches need to prove themselves in industrial contexts as well as being internally consistent. In *An integrated multi-level modeling approach for industrial-scale data interoperability*, Igamberdiev et al. evaluate their OMLM and MULLER frameworks in a real-life industry project in the oil and gas domain. They use Flora-2 as a single underlying language to support an approach to multi-level modeling that aims to minimize the complexity of implementing, querying, and verifying multi-level models in the context of big data applications.

**Acknowledgements** We would like to thank Martin Schindler for his excellent support and the authors for making this theme issue possible by submitting their work. We are also indebted to the reviewers for their detailed reviews which in many cases directly led to considerable improvements.

## References

1. Multi-level modeling WIKI. <http://homepages.ecs.vuw.ac.nz/Groups/MultiLevelModeling/> (2014)
2. Atkinson, C., Kühne, T.: Reducing accidental complexity in domain models. *Softw. Syst. Model.* **7**(3), 345–359 (2008)
3. Kühne, T.: Editorial to the theme issue on metamodelling. *Softw. Syst. Model.* **8**(4), 447–449 (2009)
4. Lara, J.D., Guerra, E., Cuadrado, J.S.: When and how to use multilevel modelling. *ACM Trans. Softw. Eng. Methodol.* **24**(2), 12:1–12:46 (2014)



in 1990 and 1985, respectively, and his B.Sc. in Mathematical Physics from the University of Nottingham in 1983.

**Colin Atkinson** is a Full Professor of software engineering at the University of Mannheim. Before that he was an Associate Professor at the University of Kaiserslautern and an Assistant Professor at the University of Houston Clear Lake. His research interests include the use of model-driven and component-based approaches in the development of dependable computing systems. He received his Ph.D. and M.Sc. in Computer Science from Imperial College, London,



**Thomas Kühne** is an Associate Professor at Victoria University of Wellington, New Zealand. Prior to that he was an Assistant Professor at the Technische Universität Darmstadt, Germany, an Acting Professor at the University of Mannheim, Germany, and a Lecturer at Staffordshire University, UK. His research interests include model-driven development, metamodeling, and multi-level modeling. He received his Ph.D. and M.Sc. in Computer

Science from the Technische Universität Darmstadt, Germany in 1998 and 1992, respectively



**Juan de Lara** is an Associate Professor at the Computer Science Department of the Universidad Autónoma in Madrid, where he leads the Modeling and Software Engineering research group (<http://www.miso.es>). His research interests include meta-modeling, multi-level modeling, domain-specific languages, and model transformation. He received his Ph.D. and M.Sc. in Computer Science from the Universidad Autónoma in 2000 and 1996, respectively.

Software & Systems Modeling is a copyright of Springer, 2018. All Rights Reserved.