Dominik Bork, Robert Buchmann and Dimitris Karagiannis

Preserving Multi-view Consistency in Diagrammatic Knowledge Representation

Published in:

8th International Conference on Knowledge Science, Engineering and Management (KSEM) 2015, 28.10 – 31.10.2015, Chongqing, China, pp. 177-182, (http://ksem2015.swu.edu.cn/index.html)

Proceedings; Lecture Notes in Artificial Intelligence, LNAI 9403, Zhang, S., Wising, M., Zhang, Z. (Eds.), Springer

The final publication is available at: 10.1007/978-3-319-25159-2_16

Preserving Multi-View Consistency in Diagrammatic Knowledge Representation

Dominik Bork[1], Robert Buchmann[2], Dimitris Karagiannis[3]

[1,3]Faculty of Computer Science, University of Vienna, Austria
dominik.bork@univie.ac.at, dk@dke.univie.ac.at
[2]Faculty of Economic Sciences and Business Administration, Babes Bolyai University, Romania
robert.buchmann@econ.ubbcluj.ro

Abstract. Multi-view conceptual modeling provides means for representing, with diagrammatic means, the knowledge describing a "system under study" whose complexity cannot be captured in a single comprehensible representation. Typical examples are available in the field of enterprise modeling, where models are inherently layered or partitioned, a feature that must be enabled at meta-modeling level by means of abstraction and decomposition. Multi-view modeling must provide means for coping with the complexity of enterprise knowledge representations through consistency preservation techniques across multiple, interrelated views. The paper at hand formulates the conceptual functions fulfilled by multi-view modeling and provides a demonstrative implementation in the context of the Semantic Object Model enterprise modeling method.

Keywords: Multi-view modeling, diagrammatic knowledge representation, metamodeling

1 Introduction

Diagrammatic conceptual models *as means of knowledge representation* emerged at the intersection of *knowledge management* (seen as a specialization of intangible asset management) and *knowledge engineering* (seen as a specialization of artificial intelligence). They can be involved in the typical scenarios and stages addressed by the knowledge science, as identified by [1] (e.g., process mining/discovery, reasoning, organizational knowledge creation). One particular concern that distinguishes diagrammatic modeling from non-visual knowledge representation is a decomposition requirement**,** and a particular answer to this is **multi-view modeling**, where different types of models represent different facets of the same system. Their **consistency** must be preserved due to existing structural or semantic dependencies. Depending on the application domain, various meanings for the “view” and “multi-view modeling” notions are implied (cf. [2] for an overview). In the following, **multi-view modeling methods** are considered a specialized instantiation of the generic modeling method definition introduced by Karagiannis and Kühn [3]. **Conceptual viewpoints** may be conceived by defining metamodel partitions with different conceptual coverage, having structural or semantic dependencies between their concepts. We hereby reduce their consistency preservation strategies to two techniques defined on an conceptual level - **state translation** and **transition translation –** then we illustrate the two techniques by implementing them for the **Semantic Object Model (SOM)** enterprise modeling method [4]. The proof-of-concept modeling tool is hosted by the Open Model Initiative Laboratory[1] (OMiLAB) [5].

The remainder of the paper is structured as follows: Section 2 states the research challenges and discusses related works. In Section 3, the two proposed techniques are formulated as abstract patterns on a metamodeling level. These are instantiated in a modelling tool [6] implemented for an illustrative project-based case in Section 4. Conclusions are drawn in Section 5.

2 Research challenges and related works

The work started its investigation from trying to understand the rationale for which views are necessary in enterprise modeling (addressed by Section 1). Further on, we faced the following challenges when engineering new modeling methods and developing modeling tools for concrete cases: ***RC1.*** *What techniques are necessary on metamodeling level to preserve consistency in multi-view diagrammatic knowledge representation?* ***RC2.*** *How can these techniques be instantiated for concrete cases of enterprise view representation?*

While the multi-view concept is not an innovation in itself, in the existing literature it has been employed as means of facilitating separation of concerns in the contexts of enterprise architecture frameworks or software engineering. Standards such as ISO/IEC/IEEE 42010:2011 prescribe software architecture design methodologies that acknowledge the need for viewpoint specifications. Software engineering typically deals with a three-layered schema (physical, conceptual, external according to ANSI/X3/SPARC [7]) or a four-view schema (development, process, physical, logical, according to the 4+1 model [8]).

[9] investigates four domain-specific modeling language tools supporting view-based modeling from an industrial background and builds a taxonomy for view-based domain-specific modeling. However, the authors focus on the specification of the views and not on the consistent utilization of multi-view modeling. [10] developed multi-view modeling principles for the SOM method, contrasting diagram-oriented and system-oriented approaches. Recently, [11] characterized a set of multi-view modeling approaches in the context of embedded and cyber-physical systems. The work of [12] provides a multi-view method without considering consistency preservation strategies at metamodeling level, while [13] integrates views weakly by means of annotative semantic relations.

None of the identified approaches discusses multi-view modeling on a generic level taking into account not only the origin of the viewpoints – the decomposition

[1] OMiLAB SOM project page, http://www.omilab.org/web/som, last checked: 23.07.15

requirement – but also the processing of the multiple views by means of modeling operations and the inconsistencies raised by their execution.

3 Preserving consistency in multi-view models

Knowledge representation traditionally deals with rule processing, either in the form of **domain-specific heuristics** or in the form of **deductive/production rules**. The aim is to generate explicit representations of implicit knowledge or to check consistency of integrated representations. We transfer this desideratum to diagrammatic conceptual modeling and distinguish two consistency preservation techniques. Thus we also narrow down the generic multi-view modeling framework of [14] into:

- The viewpoint (**VP_j**, **VP_k**) is the metamodel part covered by a type of model. It contains only those concepts from the holistic enterprise metamodel which are relevant to the viewpoint according to the decomposition requirement;
- the view (**v_1**, **v_2**, …) is the model, thus an instance of a viewpoint;
- the instance function (**μ(Viewpoint, state)**) relates a state of the system under study to a view, also considering the viewpoint as a schema, hence delimiting the modeling constructs based on the viewpoint specification and the knowledge base describing the state to be modeled;
- the operation **Op()** alters the content of a model (**v**), shifting it in another state (**v`**);
- the consistency preserving function **T(),** which can be instantiated to **state translation (T_S)** and **transition translation (T_T).**

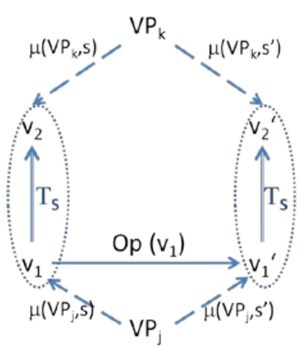


Fig. 1. T_S for *Views-by-generation*

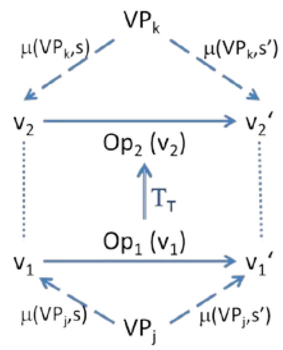


Fig. 2 T_T for *Views-by-design*

In the **views-by-generation** approach, **state translations (T_S)** specify the transformation of a complete view v_1 (reflecting a system state from viewpoint VP_j) into a new view v_2 (reflecting the same system state from a different viewpoint VP_k). At the time the transformation terminates, consistency must be ensured by the translation specification. Whenever the view v_1 is altered by applying some editing operation Op(), the state translation must be executed again on the output (v_1') to generate a new consistent view (v_2'). Hence, consistency preservation is decoupled from the editing operation Op(). On the metamodeling level, state translations typically have an algorithmic or formal rule-based implementation, triggered manually by the modeler.

In the **views-by-design** approach, **transition translations (T_T)** specify the transformation of the effects of a modeling operation Op_1, applied to v_1, into a semantically equivalent operation Op_2 that needs to be applied to v_2. Thus, transition translations integrate the multiple views in a dynamic manner.

4 Preserving consistency in the SOM modeling method

The SOM method [4] enables integrated modeling of enterprises by combining three layers. Each layer is further decomposed into multiple viewpoints. On the first layer, the **enterprise plan layer** is specified using the viewpoints **object system** and **target system**; hereby taking an external perspective on the enterprise. At the central layer, the **business process layer**, business processes (bp) are described from an internal perspective composing four viewpoints: **Interaction Schema (IAS), Task-Event Schema (TES), Object Decomposition Schema (ODS),** and **Transaction Decomposition Schema (TDS)**. The **business application system layer** specifies the resources from an internal perspective composing the viewpoints: **Schema of Task Classes (TAS)** and **Schema of Conceptual Classes (COS)**. On each layer different multi-view modeling approaches are utilized as discussed in the following.

4.1 Transition translations for SOM views-by-design modeling

Fig. 3 illustrates the integrated metamodel of the bp layer of SOM. The metamodels of IAS and TES, visualized using dashed boxes, are overlapping, e.g., the concept *Business Transaction* is part of both, IAS and TES. The ODS and TDS models solely visualize the hierarchical decompositions of *Business Objects* and *Business Transactions*, respectively.

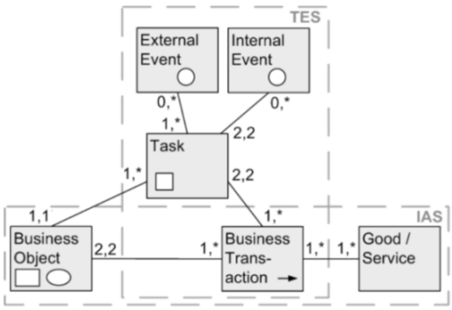
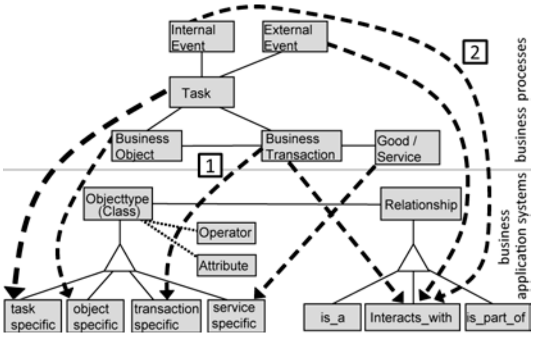


Fig. 3. Views-by-design in SOM modeling [4]

SOM also specifies the initial business process model the modeler interacts with simultaneously, following the views-by-design approach. SOM defines modeling operations a modeler applies in order to refine the initial model. These rules are specified formally [15] using the modeling constructs of the integrated metamodel. The effects of applying a modeling operation to one view are immediately transformed into semantically equivalent changes to all other affected views (i.e., transition translations). Changing the name of a business transaction in e.g., the TDS would immediately cause corresponding changes of the names of the dependent business transactions in the IAS and TES viewpoints.

4.2 State translations for SOM views-by-generation modeling

SOM utilizes a top down approach. Hence, modelers should start with the enterprise plan, then define the bp models, and finally specify the business application systems. Considering adjoining layers, state translations are utilized for views-by-generation modeling. Initial TAS and COS are generated using state translations, realized using metamodel-based model transformations. Whenever TAS and COS are generated, consistent is given. Changes to the derived TAS and COS are not considered in the business process layer, i.e., no transition translations are specified.



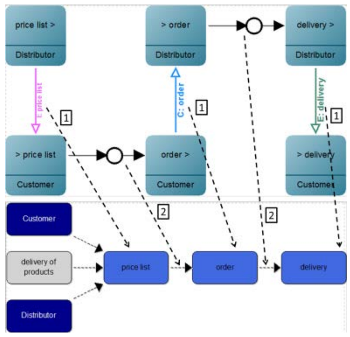


Fig. 4. Views-by-generation in in SOM modeling

Fig. 4 illustrates the specification for the state translations (on the left side) and exemplarily their application (on the right side) in the implemented SOM tool [6]. Generally, the transformations are based on a comprehensive SOM bp model consisting of four views. Due to limited space, the state translation between bp views and the COS view is exemplified. *Business Transactions* of the bp are transformed into *transaction-specific Objecttypes* in the COS (rule 1). Rule 2 transforms *internal events* into corresponding *interacts_with* relationships, connecting the transformed *transaction-specific Objecttypes* in order to preserve the process behaviour.

5 Conclusions and future work

The paper at hand introduced two consistency preservation techniques for multi-view knowledge representation through diagrammatic models – state translation and transition translation. The techniques were *evaluated through instantiation* in the SOM modeling tool which is regularly used for experimentation and didactic purposes at three universities. Future work focuses on a) the further application of the hereby introduced techniques for implementing an extensive corpus of established enterprise modeling methods; b) the integration of the presented techniques into MUVIEMOT, a conceptual modeling method for multi-view modeling tools [16]; c) the identification of multi-view requirements that cannot be supported with the presented techniques.

6 References

1. Nakamori, Y.: A Perspective of Knowledge Science. In: Proceedings of KSEM 2011, LNCS 7091, Springer, 171-182 (2011)
2. Kheir, A., Naja, H., Oussalah, M.C. and Tout, K.: Overview of an Approach Describing Multi-Views/Multi-Abstraction Levels Software Architecture. In: Evaluation of Novel Approaches to Software Engineering, Ed. by: Maciaszek, L. and Filipe, J., 140-148 (2013)
3. Karagiannis, D. and Kühn, H.: Metamodeling Platforms, In: 3rd International Conference EC-Web 2002 – Dexa 2002, Ed. by Bauknecht, K., Min Tjoa, A. and Quirchmayr, G., LNCS 2455, Springer, Aix-en-Provence, France, p. 182 (2002)
4. Ferstl O.K., Sinz E.J.: Modeling of Business Systems Using SOM. Handbook on Architectures of Information Systems. International Handbooks on Information Systems. Ed. by Bernus P., Mertins K., Schmidt G. Springer, Berlin, 347–367 (2005)
5. Karagiannis, D., Grossmann, W., Hoefferer, P.: Open Model Initiative: A Feasilbility Study, http://files.omilab.org/Open_Models_Feasibility_Study_SEPT_2008.pdf
6. Bork, D. and Sinz, E.J.: Design of a SOM Business Process Modelling Tool based on the ADOxx meta-modelling Platform, In: Pre-Proceedings 4th international Workshop on Graph-Based Tools Ed. by: de Lara, J. and Varro, D.. NL-Enschede, 89-101 (2010)
7. SPARC Relational Database Task Group: Final Report of the ANSI/X3/SPARC DBS-SG Relational Database Task Group. SIGMOD Rec. 12(4). Ed by. Brodie, M.L. and Schmidt, J.W., ACM, New York, NY, USA, 1-62 (1982)
8. Kruchten, P.: Architectural Blueprints - The “4+1“ View Model of Software Architecture. IEEE Software 12(6), 42-50 (1995)
9. Goldschmidt, T, Becker, S. and Burger, E.: Towards a Tool-Oriented Taxonomy of View-Based Modelling. In: Proceedings Modellierung 2012, Bamberg, 59-74 (2012)
10. Bork, D. and Sinz, E.J.: Bridging the Gap from a Multi-View Modelling Method to the Design of a Multi-View Modelling Tool. Enterprise Modelling and Information Systems Architectures 8(2): 25-41 (2013)
11. Persson, M. Törngren, M., Qamar, A., Westman, J., Biehl, M., Tripakis, S., Vangheluwe, H. and Denil, J.: A Characterization of Integrated Multi-View Modeling in the Context of Embedded and Cyber-Physical Systems. In Proceedings of the International Conference on Embedded Software, 1-10 (2013)
12. Masuda, H., Utz, W., Hara, Y.: Development of an Evaluation Approach for Customer Service Interaction Models. In: Proceedings of KSEM 2014, Springer, 150-161 (2014)
13. Fill, H.-G.: On the Social Network Based Semantic Annotation of Conceptual Models, In: Proceedings of KSEM’2014, Springer, 138-149 (2014)
14. Bork, D. and Karagiannis, D.: Model-driven Development of Multi-View Modelling Methods: The MUVIEMOT Approach, In: International Conference on Software Paradigm Trends. Ed. by Holzinger, Cardoso, Cordeiro, van Sinderen and Mellor, 11-23 (2014)
15. Bork, D. and Fill, H.-G.: Formal Aspects of Enterprise Modeling Methods: A Comparison Framework, Hawaii International Conference on System Sciences, 3400-3409 (2014)
16. Bork, D.: Using Conceptual Modeling for Designing Multi-View Modeling Tools, 21st Americas Conference on Information Systems (AMCIS) (2015)