Panel Discussion: How to Build a Perfect Enterprise Modeling Method

Jolita Ralyté [1], Dominik Bork [2], Manfred A. Jeusfeld [3], Marite Kirikova [4] and Janis Stirna [5]

[1] *University of Geneva, CUI, Ballette bât. A, Route de Drize 7, Carouge, 1227, Switzerland*
[2] *TU Wien, Business Informatics Group, Favoritenstrasse 9-11, 1040 Vienna, Austria*
[3] *University of Skövde, Box 408, 54128 Skövde, Sweden*
[4] *Riga Technical University, Kalku 1, Riga, LV-1658, Latvia*
[5] *Stockholm University, Postbox 7003, Kista, SE 16407, Sweden*

Abstract

This paper reports on the plenary panel discussion on "How to Build a Perfect Enterprise Modeling Method" held at the PoEM 2021 conference. The panel was charged with finding a pathway to the perfect enterprise modeling method. The panelists have a background in enterprise modeling and method engineering. So, the question is: Can method engineering help with designing a better enterprise modeling environment? The contributions in this paper should be regarded as a snapshot of our viewpoints at the time of the panel.

Keywords

Enterprise Modeling, Method Engineering, Method Requirements, Conceptual Modeling, Enterprise Architecture

1. Introduction

Enterprise Modeling (EM) is a well-known approach to formally describing an enterprise with a set of models reflecting different views of the enterprise such as its organizational structure, business goals, services, products and processes, information system (IS) and its components, IT infrastructure, security aspects, etc. The models are abstractions of the enterprise knowledge. They provide a medium to share a common understanding between various stakeholders and to conduct the progression of the enterprise.

An EM method is a conceptual tool, facilitator and guide that has various applications in the life of an organization. It supports the implementation of its Enterprise Architecture [1], the strategic planning, re-organization and digital transformation [2], the development and evolution of information systems and services in concordance with the organization's development strategy [3], the design of new products and services to be delivered to its customers [4], the development of digital twins [5], and dealing with various business challenges [6].

A typical EM method includes several modeling languages, one per different modeling view, and provides guidance (and sometimes a supporting tool) to build the models. It aims to be generic in the sense that it can be applied to any enterprise. However, the life of today's enterprise is not a long quiet river. It has to cope with diverse challenges of digital transformation, business agility and even technology disruption. The needs for EM evolve at the same pace as organizational contexts, business requirements and technologies evolve. Therefore, how could one method satisfy all organizations in all situations?

A large number of EM methods and languages have been proposed by academia and industry [6], but changing EM requirements make the task of building, tailoring and extending EM methods more than relevant today.

PoEM'21 Forum: 14th IFIP WG 8.1 Working Conference on the Practice of Enterprise Modelling, November 24–26, 2021, Latvia
EMAIL: jolita.ralyte@unige.ch(J. Ralyté); dominik.bork@tuwien.ac.at (D. Bork); manfred.jeusfeld@his.se (M.A. Jeusfeld); Marite.Kirikova@rtu.lv (M. Kirikova); js@dsv.su.se (J. Stirna)
ORCID: 0000-0001-8561-3567 (J. Ralyté); 0000-0001-8259-2297 (D. Bork); 0000-0002-9421-8566 (M.A. Jeusfeld); 0000-0002-1678-9523 (M. Kirikova); 0000-0002-3669-832X (J. Stirna)

The aim of the panel held at the PoEM 2021 conference was to explore the Method Engineering (ME) perspective of Enterprise Modeling (EM) – to elicit the requirements and highlight the challenges of ME for building a perfectly adapted and efficient EM method. The panelists were invited to answer the following three questions and sub questions:

1. **What are the requirements for a perfect EM method?** Does one method can fit all organizations and all situations? Does the enterprise context matter when building an EM method? How to make the EM easy and efficient?
2. **Which ME approach for building an EM method?** What works and what does not? How to learn from past failures and/or successes? Should we collect patterns and anti-patterns of ME for EM?
3. **What expertise and effort are required to build/use an EM method?** How to reduce the effort and how to gain the expertise?

The panel involved Dominik Bork, Manfred Jeusfeld, Marite Kirikova and Janis Stirna as the panelists, while Jolita Ralyté was the moderator of this session. The reminder of this paper summarizes the discussion: the views of the panelists and ant the inputs from the audience regarding the three main questions described above. In section 2 we elicit the requirements that an EM method should satisfy to seek the perfection. In section 3 we discuss method engineering challenges while in section 4 we reflect on the expertise and effort required to create EM methods. Section 5 concludes the paper by summarizing the outcome of the panel.

2. Requirements for a perfect EM method

From the Method Engineering (ME) point of view, and especially from the situational ME [7] point of view, one method cannot fit all situations and all organizational contexts. The first step of building a new method consists in specifying the requirements for this method [8]: defining its scope, required functionalities and quality aspects in accordance with the organizational context and situational factors. Functional method requirements specify various types of models to be produced, validated and managed. They are necessary but not sufficient to reach method perfection. Perfection, as a concept, is usually related to quality. In the following subsections we mainly discuss the quality requirements and situational aspects.

2.1. Quality requirements for an EM method

An enterprise modeling method typically covers multiple perspectives (data, process, goal, event, etc.) for multiple levels (business, application software, network and infrastructure, etc.). Such a multi-perspective, multi-layer modeling method first has to fulfill the typical quality requirements for multi-perspective modeling methods:

- *Minimal redundancy*: The constructs used in the different perspectives / layers should have a minimal overlap to avoid redundancy in the models created for the perspective / layer. A certain amount of redundancy is unavoidable to be able to express cross-references. Further, some redundancy may be useful to detect conflicts in views expressed by different stakeholders / modelers.
- *Cross-notational constraints*: When one modeling language makes references [9] to constructs used in another modeling language (used for another perspective / layer), then the EM method should contain suitable constraints for using these cross-references.
- *Consistency*: Changes in one model involving concepts that have cross-notational links to other models should be translated into semantically equivalent changes that an EM modeling tool should automatically apply to all affected models in order to retain an overarching model representation of the enterprise [10].

Specifically for EM, we identify further requirements for the perfect EM method:

- *Ontological grounding / semantics*: The constructs should be related to an agreed-upon ontology of artefacts occurring in enterprises, such as resources, processes, events, and so forth.

For example, a construct for a task should be declared to be equivalent to some concept in the ontology.

- *Detection of modeling flaws*: The perfect EM method should have advanced tools to detect flaws in models as well as "undesired model patterns".
- *Extensibility*: The core constructs of the EM method should be extensible by domain-specific constructs, ideally defined as specializations of the core constructs. The new constructs shall not violate the semantics of the core constructs that they are subclasses of.
- *Multi-level abstractions*: Enterprise models describe artefacts at different abstraction levels, e.g. physical infrastructure objects such as machines, computers. Databases like product catalogs refer to product models (further grouped into product types and categories). There may be business processes about the creation of product/service instances. Other business processes are about creating product/service models. Hence, the perfect EM method should be able to deal with artefacts at any of these abstraction levels.

Perfection of an EM method, consisting of procedures, meta-models, and actually constructed models, thus, depends on the quality of enterprise (meta-)models. K.S. Andersen, under supervision of J. Krogstie, has amalgamated the following systems of criteria for evaluating enterprise model quality [11], which are inherently important also for EM methods:

- *Principles of modelling* from [6]: *Completeness*: the degree to which all relevant facts from the domain are included in the model; *Correctness*: how well the model conforms to the rules of the modelling technique; *Flexibility*: the degree to which the model can adapt to changes in the modelling domain; *Integration*: to which degree of consistency between the different sub-models that constitute the model; *Simplicity*: the degree of minimal use of modelling constructs for the presenting knowledge in the model; *Understandability*: the extent to which the concepts and structures of the model can be understood by the stakeholder; and *Usability*: the ability of the model to be used for its intended purpose.
- *Guidelines of modeling* from [12] that contain such general principles as the *principle of construction adequacy* for being able to have a consensus that a model is correct compared to reality; *the principle of language adequacy* for a modeling context; the *principle of economic efficiency* ensuring that the cost of creating and maintaining a model does not exceed the value gained from it; the *principle of clarity* dealing with comprehensibility and explicitness of models; *the principle of systematic design* requiring every view be connected to the model and not creating disagreements between other views; the *principle of comparability* – models should be comparable both on model and meta-model level.
- *SEQUAL* (semiotic quality framework) [13] – a systems modelling reference model for evaluating the quality of conceptual models.

The strive for perfection can be also considered in terms of useful and used in practice. This in turn requires two key points to consider: (1) the EM method facilitating the production of high-quality models and (2) satisfied stakeholders [6]. While the former is addressed by introducing various guidelines and recommendations for modeling, the latter is achieved by recommendations and guidelines, and often even plain common sense, for organizing modeling sessions and projects. How to organize modeling and requirements workshops has been discussed, for example, in [14] [15]. In addition to such guidelines there are also a number of requirements to consider.

The EM method is to be defined and recognizable by its stakeholders. This means that it is possible for its users to say what the method is and how to use it, including how to acquire and introduce it in an organization. In this regard the method should support these actions e.g. by offering pathways and integration points with other methods and approaches. In doing so it would relieve the stakeholders from the burden to think how to do things they are already doing according to the new method. At the same time the EM method should also require certain discipline and rigor of working. Without this there is a risk that over time its use will degenerate into purely performative actions without deeper analytical meaning. In such cases only the name of the method would be left. Such situations have been observed in practice even by commercially successful method users. One of the reasons for this is the lack of understanding of what methods are, what in terms of skills and resources is needed to use methods, what they can actually help the organization to achieve, and what they cannot.

A common requirement observed in practice is that the EM method should be *easy to use* and, as formulated by potential stakeholders, "uncomplicated". In this regard it is worth noting that these are subjective criteria related to one background knowledge and pre-existing skill set. It is often so that things we know how to do seem easy to us. Hence, EM methods should relate to other methods and management techniques, even only to provide reference points. Method engineers should also think about how easy their method can be communicated to potential users and how they will understand them.

EM methods have to be *customizable*. They should offer customization and integration possibilities, e.g. in their meta-models and supporting tools. It should also welcome such customization and extensions, for example, by providing guidelines and consultations by method developers. At the same time, it should also be stated that not every aspect of the method should be open for customization and not everybody should be doing that. A way of dealing with this aspect in practice would be to strive for a more dedicated role of method expert or manager on an organizational level.

Last but not least, a perfect method would be supported with the tools that automatically could represent the existing IT architecture and reuse knowledge represented and captured by IT systems of an enterprise. Another issue is a visibility of available models and other forms of representation. One of the interesting approaches proposed in this regard is having a cockpit where the different views on an enterprise are available [16].

2.2. Taking into consideration the organizational context

The enterprise context should also be considered when choosing and using EM methods. For example, [17] and [18] discussed various aspects of situational contingencies named intentional and situational factors influencing EM method and tool adoption and use. In short, the choice of EM method and tool and their use depends on the intentions the company wants to pursue e.g. does it want to use the method without the support of external experts, and the situation it is in, e.g. how mature it is in terms of using modeling methods in general. These we call intentional and situational factors.

Regarding the context of an enterprise and the situations we can distinguish between features that are context and situation independent and the ones (e.g. language adequacy) that may be tailored to the size of the enterprise, business area, geographical location, cultural environment, political environment and other contextual and enterprise specific factors. One aspect that has a generic importance is the capability of an EM method to support continuous improvement in an enterprise [19]. To support continuity, the method must be applicable iteratively, and the results of its application should be reusable for further iterations. Respecting continuous changes in enterprises the method itself may need to have continuous improvements that make it conforming to new strategic and business needs of an enterprise. To have a continuous evolution and application of EM methods, a sophisticated tool support is necessary for capturing the modeling results, maintaining them, and simultaneously letting to incorporate changes in meta-models.

3. Method Engineering for EM

The aim of method engineering is to develop theories and foundations for building and adapting new methods. EM methods are within its scope. Situational method engineering [7] affirms that the method to be effective must perfectly adapt to the situation of the organization or the project which uses it. In this section we highlight the challenges and requirements that method engineering should take into account to support the engineering of situation-specific and adaptable EM methods.

The underlying question is whether the diversity of enterprises is so high that enterprise-specific methods should be developed on the fly. If so, then it may be hard to compare enterprise models of different enterprises. Such comparisons are almost impossible due to the confidentiality of the models. Still, there should be a common core of constructs that the EM community may agree upon, roughly along the lines of Zachman Framework [20] and TOGAF [21]. A particular challenge for an EM method is the knowledge acquisition. A meta model defining the constructs is not sufficient. One also needs to specify how to extract the models from sources such as interviews, existing information systems, documentation, and so forth. Another open challenge is the specification of analysis tools for enterprise

models: error detection, quality assessment, dependency tracing, simulations may be some of the analysis services to be offered. When enterprise models are used for what-if scenarios, then the method engineering approach should include constructs on how to map the to-be enterprise model back to the enterprise systems that implement the model, e.g. the ERP system of the enterprise.

To cater for the diversity of enterprises, one may also dynamically extend a core EM method by new constructs specific for the enterprise. The wheel does not have to be re-invented many times. A core EM method can be reused. The method engineering approach is then by principle incremental. Like for schema evolution in databases, changes to the EM method may require adapting the models created with the old version of the EM method. It would be desirable to allow the method engineer to restrict changes to those that would not require extensive changes to existing models.

One approach to adapt enterprise modeling methods to specific contexts and enterprises that was also generically implemented on a state-of-the-art metamodeling platform has been proposed by Bider et al. [22].

Patterns and anti-patterns are a proven tool to incorporate empirical knowledge into an engineering activity such as software engineering. In the context of enterprise modeling, such patterns could express for example how a software architecture supports a process pattern like “separation of concerns”. Patterns and anti-patterns can also be used to analyze enterprise models that are mined from other enterprise systems such as ERP systems. The downside is that patterns are only useful in a specific context (e.g. legal framework of a given country). It is inherently difficult to explicitly represent such a context.

The development of EM should be based on the principles of method engineering. A key aspect of method development is modularization; to offer a component-based method and guidance on how to use and combine the components in various method application contexts. This approach was used, for instance, in the development of 4EM and CDD. More about the CDD development and the choices made in this process is available in [23]. The component-based approach also offers the possibility to develop method extensions that do not need to be planned or even envisioned in the early stages of method development.

A key aspect of method development is tool support. Methods without any tool support are hard to introduce in practice. Hence, in this regard, the method development team should have an early strategy. While it is, of course, possible to develop a new tool from scratch this approach is undoubtedly costly and time consuming. Instead, strategies such as the implementation of method support in a customizable meta-tool should be considered as more efficient because despite their drawbacks such as being locked into a vendor’s tool and the functionality of that tool.

Method engineering for EM methods should foremost explicitly specify the intended stakeholders and their purposes of using the method to be developed. This makes sure that the method is neither over-engineered by focusing on edge cases, nor that it fails in covering the domain and supporting its purposes adequately. Moreover, EM method engineering should aim to follow procedures that are flexible and iterative in nature. The approach proposed by Frank [24] accounts to these requirements to a great extent by using sample models (or even informal sketches) as early as possible in the method engineering lifecycle. Explicating early design decisions by means of examples enables the method engineering team to involve other stakeholders and to integrate their feedback as early as possible. Eventually, the proposed EM method needs to undergo empirical evaluation in order to show its usability, ease of use, and value in the intended domain by the intended stakeholders.

The method engineering community should take care that the method can be supported by a dedicated tool or a set of tools. These tools would be expected to capture the stakeholder knowledge and generate useful representations of this knowledge. A good example of such generation capabilities is the CORE tool from Vitech corporation [25]. This tool, for instance, on the basis of a functional block diagram can create an IDF0 diagram or an activity diagram. This way the effort of model creators is saved and additional means for quality checking are available. Similar model generation approach could be also applied to models included in EM methods.

Respecting that the EM methods are applied by stakeholders whose knowledge may differ and is evolving, the method engineering for EM methods has to consider also supporting models and processes for knowledge management [26]. Another important issue is that the engineering method considers not only the method per se but also the software solution(s) that will support the method.

The above issues have been relatively well-understood in the EM community as well as they have attracted some albeit insufficient attention in publications. What has been discussed and researched considerably less is what kind of project management and project team a successful ME project should have. For example, many method developments have some origins in the academic world, while tool development needs considerable industrial backing that cannot be easily established in academic environments. Methods and tools need to be applied and the experiences of the applications channeled back into the development. This puts forward the need for the method development team to consist of researchers and practitioners. Another issue is that for a successful method to be existing and having impact in the practice, it needs to be used and updated over a long time. Hence, the more commonly seen constellations of teams centered on an externally funded research grant of 3-5 years need to plan how to work after the initial project ends.

4. Expertise and effort required to build/use an EM method

The key areas of expertise for building and maintaining EM methods professionally are method engineering and conceptual modeling as well as, if we consider that most methods require it, knowledge of modeling tools and method implementation in tools. Without competence in these areas the method developers will most likely end up repeating many mistakes of the past and most likely end up only with a research prototype method that focuses on the central aspects of the envisioned method, but will lack the comprehensiveness and detail needed for a method to be used in practice. For example, developing a core of a modeling language with meta-modeling can be done relatively quickly, after that a simple graphical language can be constructed and tested on a few example cases. While in a research project this would count as the first step of validation, it is still far from an industrially applicable method. To achieve the latter, one would need to apply it to a reasonable number of real-life projects or cases (probably no less than 10) and develop an extensive method guide, populated with examples, experience-based recommendations, patterns, and etc. This takes considerable time and effort and, moreover, it cannot be done with only a few people in the method development team. In addition, there are the ongoing tasks such as keeping a website up-to-date.

EM method development needs to be seen as a long-term endeavor. Two main stages can be considered – (1) the initial development work towards creating the first reasonably complete and useful method, and (2) applying the method and at the same time continuously improving it. The latter stage also requires the method developers to engage in activities in which only few of them have extensive expertise. These are marketing and sales of methods, tools and related consulting services. An aspect that has not been discussed in this regard is branding. Would it be relevant to think of modeling methods as brands, and, if so, what then can be done to increase brand recognition and awareness? The EM community associated with the PoEM conference mainly represents researchers who write academic publications and books. The reach of these activities outside the community however seems limited. Positive examples of other activities with potentially higher impact are, for example, the NEMO summer school [27].

In order for EM methods to be usable and actually used in practice, two things are of paramount importance. On the one hand, EM methods need to be supported by appropriate modeling tools. On the other hand, the value of enterprise models needs to be equal or higher compared to the efforts involved in their production.

To account for the heterogeneous stakeholders involved in enterprise modeling, corresponding tools need to be modern, easy to deploy, and efficient to use. The provision of application systems through the browser/cloud is nowadays considered a commodity and stakeholders involved in enterprise modeling projects should not be expected to install a software in advance or to get experienced with entirely new software systems and user interfaces. The uptake of web technologies and the provision of modeling tools through the web or as extensions to widely adopted systems like VS Code enables great potential for the EM community [28].

In order to increase the value of producing enterprise models, the EM community should also be open to incorporate methods and techniques from data science and artificial intelligence (for a recent discussion and selected advances, see [29][30][31][32]). Intertwining enterprise modeling with these fields of research not only generates manifold potentials for intelligent or assisted modeling [33][34],

it also ensures that the EM community works on the big challenges and remains attractive to students and young researchers.

A perfect EM method should also minimize the effort to create an up-to-date and consistent enterprise model. Many enterprises already have a number of software systems in use that maintain a partial view of the enterprise: workflow management systems contain process models, UML modeling tools may contain data models and other models, ERP systems contain not only data but also process models. Process models can be mined from event log files. The organizational structure of an enterprise may be stored in an ERP system or static documents. Such sources should be harvested automatically to minimize the manual modeling effort. While the initial investment in the necessary import filters is rather high, the automatic execution allows to keep the enterprise models up-to-date with minimal human effort, much like the ETL process in a data warehouse. The perfect EM method should automatically generate abstractions of the harvested models.

A positive example in terms of EM method and tool usage is ArchiMate language [35] and Archi tool [36]. Both are flexible and easy to teach. As a result, students who learn these methods at universities are freely using them at their workplaces. From this we can learn that overall education in enterprise modeling matters, as well as availability of freely and openly accessible tools. Simplicity is an important factor in the use of any tools, however, we cannot afford to oversimplify in enterprise modeling. Therefore, the skills for enterprise modeling will be possible to develop only gradually and by obtaining knowledge in several areas such as conceptual modeling, the use of modeling tools, facilitation skills, etc.

5. Conclusion

"Perfect is just a vision. We will never reach it, but we can work towards it. It keeps us sharp and critical and longing for more and better. It aligns goals of people. This is also how an organizational vision works" (Geert Poels, panel attendee). This quote from the panel chat suggests that while a perfect EM method is probably not reachable, it is still worth striving in that direction.

The aim of the panel presented in this paper was to challenge participants to express their understanding of what a perfect EM method should be and to identify the method engineering challenges to build such a method. Below we summarize the main findings from the panel discussion.

Value proposition. First of all, the method engineer should state the value proposition of the method. An EM method should be useful: fit to the purpose and fit to the user and his/her concerns. As the modeling task requires time and effort, the obtained results should be worth of it. Henderik Proper, a panel attendee, calls it a "Return on Modeling Effort". In general, an EM method is expected to support multi-perspective and multi-layer modeling.

Strong theoretical foundation. An EM method should be theoretically well-founded: based on an agreed ontology and formally defined using meta-models and/or formal languages. That concerns abstract and concrete syntax as well as the way of working to create and use models. This can be done with the help of method engineering approaches and tools and quality reference models.

Adaptability. The context and needs of EM vary from one organization to another and even evolve within an organization over time. Therefore, an EM method should be customizable and extensible by context-specific and/or domain-specific constructs as well as empirical knowledge captured in patterns or anti-patterns.

Method engineering. Many method engineering approaches and techniques exist. Which one to use for building an EM method? Keeping in mind that the method should fit the organizational context and needs, which evolve over time, we see two possibilities: (1) building on the fly or (2) tailoring of a core EM method? Both approaches are based on the principle of modularization. A method is viewed as a composition of method components, which cand be added and/or removed depending on the situation and current needs. That supports continuous method improvement. Component-based approaches also support incremental method development where method extensions can be done at any time, they do not need to be planned in advance. To build an EM method, both, method engineering and EM knowledge and expertise are required.

Automation. A tool support is necessary at both levels: method engineering and method application. A method engineering platform is expected to help formally define methods, potentially by reusing and

adapting existing method components, and to support their continuous evolution. Similarly, to make the modeling task efficient, an EM method should be supported by advanced tools, easy to deploy and to use. The tools must not only help to create models but also verify and validate their consistency and quality, and establish cross-model and cross-layer links and dependencies. Hence, the evolution of an EM method implies the evolution of the tools that support it.

Learning. No method or tool is going to completely replace human knowledge and experience, especially that of the business domain. Training is essential to gain expertise and efficiency in in the use of EM methods and tools.

Success. Finally, how can we measure the success of a method? Is the fact that the method is used in practice sufficient to say that it is successful? No, this is only a minimum criterion for appreciating the method. To reach the success an EM method should deliver the promised value to the stakeholders and possess most of the quality aspects highlighted in this paper.

6. References

[1] M. Lankhorst (ed), Enterprise Architecture at Work: Modelling, Communication and Analysis. The Enterprise Engineering Series, Springer, Berlin, Heidelberg 2017, https://doi.org/10.1007/978-3-662-53933-0.

[2] B. van Gils, H.A. Proper, Enterprise Modelling in the Age of Digital Transformation. In: Robert Buchmann A., Karagiannis D., Kirikova M. (eds) The Practice of Enterprise Modeling, PoEM 2018, LNBIP vol 335, Springer, Cham. pp. 257-273, 2018.

[3] W. Opprecht, J. Ralyté. and M. Léonard, M, Towards a Framework for Enterprise Information System Evolution Steering. In: Frank U., Loucopoulos P., Pastor O., Petrounias I. (eds) The Practice of Enterprise Modeling, PoEM 2014, LNBIP vol 197, Springer, Berlin, Heidelberg, pp 118-132, 2014.

[4] K. Sandkuhl, J. Stirna, F. Holz, Modeling Products and Services with Enterprise Models. In: Grabis J., Bork D. (eds) The Practice of Enterprise Modeling, PoEM 2020, LNBIP vol 400. Springer, Cham, pp. 41-57, 2020. https://doi.org/10.1007/978-3-030-63479-7_4

[5] U.V. Riss, H. Maus, S. Javaid, C. Jilek, Digital Twins of an Organization for Enterprise Modeling. In: Grabis J., Bork D. (eds) The Practice of Enterprise Modeling, PoEM 2020, LNBIP vol 400, Springer, Cham., pp 25-40, 2020. https://doi.org/10.1007/978-3-030-63479-7_3

[6] K. Sandkuhl, J. Stirna, A. Persson, M. Wißotzki, Enterprise Modeling: Tackling Business Challenges with the 4EM Method, 1st ed. 2014. The Enterprise Engineering Series. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-43725-4

[7] B. Henderson-Sellers, J. Ralyté, P. Ågerfalk and M. Rossi, Situational Method Engineering. Springer, 2014. https://doi.org/10.1007/978-3-642-41467-1

[8] J. Ralyté, R. Deneckère, C. Rolland, Towards a Generic Model for Situational Method Engineering. In Proceedings of the 15th International Conference on Advanced Information Systems Engineering, CAISE 2003, Eder J. and Missikoff M. (eds). LNCS vol. 2681, Springer-Verlag, pp. 95-110, 2003.

[9] H.W. Nissen, M.A. Jeusfeld, M. Jarke, G.V. Zemanek, H. Huber, Managing Multiple Requirements Perspectives with Metamodels. IEEE Software, 13, 2, pp. 37-48, 1996.

[10] D. Bork, R.A. Buchmann, D. Karagiannis. Preserving Multi-view Consistency in Diagrammatic Knowledge Representation. In International conference on Knowledge Science, Engineering and Management, Springer, Cham, pp 177-182, 2015.

[11] K.S. Andersen, Exploring Model Quality in Enterprise Modeling - A Case Study, Master Thesis, NTNU, 2016. https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/2415315

[12] D.L. Moody, Metrics for Evaluating the Quality of Entity Relationship Models. In: Ling TW., Ram S., Li Lee M. (eds) Conceptual Modeling – ER '98. ER 1998. LNCS vol. 1507. Springer, Berlin, Heidelberg, 1998. https://doi.org/10.1007/978-3-540-49524-6_18

[13] J. Krogstie, Model-Based Development and Evolution of Information Systems. Springer London, 2012.

[14] J. Stirna and A. Persson, Enterprise Modeling: Facilitating the Process and the People. Springer, Cham, 2018.

[15] A. Zavala, B.H.Hass, The Art and Power of Facilitation: Running Powerful Meetings. Management Concepts, Vienna, 2008.
[16] D. Jugel, Modeling Interactive Enterprise Architecture Visualizations: An Extended Architecture Description. Complex Systems Informatics and Modeling Quarterly, CSIMQ, no. 16, pp. 17–35, 2018. https://doi.org/10.7250/csimq.2018-16.02
[17] A. Persson, Enterprise Modelling In Practice: Situational Factors and Their Influence on Adopting a Participative Approach. Ph.D. thesis, Department of Computer and Systems Sciences, Stockholm University, 2001.
[18] J. Stirna, The Influence of Intentional and Situational Factors on Enterprise Modelling Tool Acquisition in Organisations. PhD Thesis, Department of Computer and Systems Sciences, Royal Institute of Technology, Stockholm, Sweden, 2001. ISSN 1101-8526
[19] A.V. Gumerov, M.K. Biktemirova, S.M. Nuryyakhmetova, R.E. Moiseev, A.B. Nikolaeva, R.R Kharisova, V.P. Rukomoinikova, Quality Functions Modeling of Industrial Enterprises Products. International Review of Management and Marketing, Vol. 6 (1): 165-169, 2016.
[20] The Concise Definition of The Zachman Framework by: John A. Zachman, https://www.zachman.com/about-the-zachman-framework
[21] The TOGAF® Standard, https://www.opengroup.org/togaf
[22] I. Bider, E. Perjons, D. Bork. Towards On-The-Fly Creation of Modeling Language Jargons. In proceedings of the 17th International Conference on ICT in Education, Research and Industrial Applications. Integration, Harmonization and Knowledge Transfer. Volume I: Main Conference, PhD Symposium, and Posters, 2021. CEUR-WS Proceedings Vol-3013, pp. 142-157, 2021. http://ceur-ws.org/Vol-3013/
[23] J. Stirna, J. Zdravkovic, J. Grabis, K. Sandkuhl, Development of Capability Driven Development Methodology: Experiences and Recommendations. In: Poels G., Gailly F., Serral Asensio E., Snoeck M. (eds) The Practice of Enterprise Modeling. PoEM 2017. LNBIP vol 305, Springer, Cham, 2017. https://doi.org/10.1007/978-3-319-70241-4_17
[24] U. Frank, Domain-Specific Modeling Languages: Requirements Analysis and Design Guidelines. In: Reinhartz-Berger I. et al. (eds) Domain Engineering. Springer, Berlin, Heidelberg, 2013. https://doi.org/10.1007/978-3-642-36654-3_62013.
[25] Vitech corporation, CORE: A Proven Legacy, a Pioneer in MBSE, 2021, URL: http://CORE Software – Vitech (vitechcorp.com)
[26] K. Lace, M. Kirikova, The Models for Knowledge Acquisition in PMI Specific Requirements Engineering. In: Serral E., Stirna J., Ralyté J., Grabis J. (eds) The Practice of Enterprise Modeling. PoEM 2021. LNBIP vol 432. Springer, Cham, 2021. https://doi.org/10.1007/978-3-030-91279-6_3 2.
[27] D. Bork, R.A. Buchmann, D. Karagiannis, M. Lee, E.T. Miron, An Open Platform for Modeling Method Conceptualization: The OMiLAB Digital Ecosystem. Communications of the Association for Information Systems 44: 673-697, 2019.
[28] P.L. Glaser, D. Bork. The bigER Tool - Hybrid Textual and Graphical Modeling of Entity Relationships in VS Code. In 25th International Enterprise Distributed Object Computing Workshop, EDOC Workshop 2021, pp. 337 - 340, IEEE Xplore, 2021.
[29] N. Shilov, W. Othman, M. Fellmann, K. Sandkuhl, Machine Learning-Based Enterprise Modeling Assistance: Approach and Potentials. In: Serral E., Stirna J., Ralyté J., Grabis J. (eds) The Practice of Enterprise Modeling. PoEM 2021. LNBIP vol 432. Springer, Cham, 2021. https://doi.org/10.1007/978-3-030-91279-6_2.
[30] J.C. Recker, R. Lukyanenko, M. Jabbari Sabegh, B. Samuel, A. Castellanos, From representation to mediation: a new agenda for conceptual modeling research in a digital world. MIS Quarterly: Management Information Systems, 45(1), 269-300, 2021.
[31] W. Maass, V.C. Storey, R. Lukyanenko, From Mental Models to Machine Learning Models via Conceptual Models. In: Augusto A., et al (eds) Enterprise, Business-Process and Information Systems Modeling, LNBIP vol 421, pp 293-300. Springer, Cham, 2021.
[32] D. Bork, A. Garmendia, M. Wimmer, Towards a Multi-Objective Modularization Approach for Entity-Relationship Models. In proceedings of ER Forum, Demo and Posters 2020, CEUR-WS Vol-2716, pp. 45-58, 2020. http://ceur-ws.org/Vol-2716/

[33] G. Mussbacher, B. Combemale, J. Kienzle, S. Abrahão, H. Ali, N. Bencomo, M. Búr, L. Burgueño, G. Engels, P. Jeanjean, J.-M. Jézéquel, T. Kühn, S. Mosser, H.A. Sahraoui, E. Syriani, D. Varró, M. Weyssow, Opportunities in Intelligent Modeling Assistance. Software and Systems Modeling, 19(5): 1045-1053, 2020.
[34] C. Feltus, Q. Ma, H.A. Proper, P. Kelsen, Towards AI Assisted Domain Modeling. In proceedings of the International Conference on Conceptual Modeling, ER 2021, LNCS vol 13012, Springer, Cham, pp. 75-89, 2021.
[35] Archimate Specification, https://www.opengroup.org/archimate-home
[36] Archi: The Open Source modelling toolkit for creating ArchiMate models and sketches. https://www.archimatetool.com/