Enterprise Modelling in the Age of Digital Transformation

Hend*erik* A. Proper
Luxemburg Institute of Science and Technology, Belval, Luxemburg
e.proper@acm.org

Bas van Gils
Strategy Alliance, Amersfoort, The Netherlands
bas.vangils@strategy-alliance.com

An IFIP SELECT paper for IT professionals based on the research paper:

van Gils B., Proper H.A. (2018) Enterprise Modelling in the Age of Digital Transformation. In: Buchmann R., Karagiannis D., Kirikova M. (eds) The Practice of Enterprise Modeling. PoEM 2018. Lecture Notes in Business Information Processing, vol 335. Springer, Cham

Publisher's information: https://link.springer.com/chapter/10.1007/978-3-030-02302-7_16#citeas

1 Introduction

The world around us is increasingly becoming a digital world. '*Being digital*' [9] seems to be the new norm. As such, it is safe to speak about the digital transformation of society (see e.g. [2, 17, 13]). As a result, the number of (human/computer) components that are interconnected and behave (semi)autonomously has also increased to the point where we are confronted with highly complex systems [15] that are hybrids between human and digital actors.

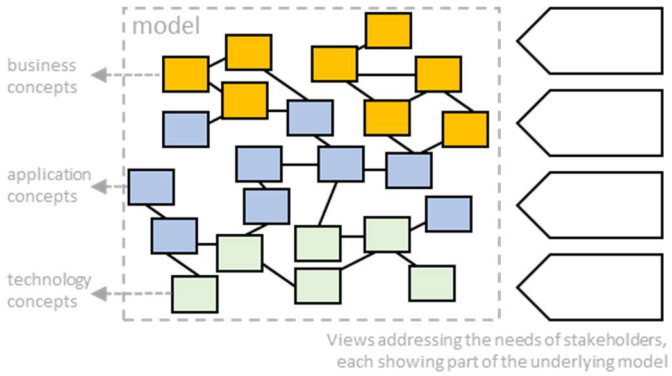
It is essential for enterprises to manage the resulting complexity, and we believe that *modelling* is a key component to successful digital transformation [12]. In the past, models have been used to create a visual *map* of the past/present/future of the enterprise (e.g. [16]).

It is, therefore, only logical to try and continue to do so in the digital age. At the same time, this ambition brings about many challenges. In terms of the Cynefin framework [15], one can say that for *complicated* systems it *may* be possible (even though it may be at high cost) to create an accurate model of such systems, whereas for *complex* systems, models can only function as humble (yet useful) *hypothesis* of the workings of these systems. The very nature of complex systems is that they cannot be fully understood. Yet, modelling helps stakeholders to communicate about these systems in a meaningful way especially when trying to influence the behaviour of this system.

In our paper [6], presented at the PoEM (Practice of Enterprise Modelling) working conference, we focused on the question if the ArchiMate enterprise (architecture) modelling language lives up to the challenges of the digital age.

As a brief introduction to (the use of) ArchiMate, consider the following example which is inspired by a project in the financial industry in which one of the authors of this paper participated. This organisation employs over 800 people in various locations and offers a broad range of financial services to clients. There were two main drivers for documenting the architecture of the organisation using ArchiMate. First, there was a regulatory requirement to map out the interplay between processes, systems, and data. Second, there is a strategic driver to build an improved *platform for execution* for core processes on new IT and a sound understanding of current practices are required. For this organization, we developed an extensive model that lists: (a) products and services, linked to (b) processes, roles and responsibilities, (c) data and data flows, connected to processes and systems, (d) the application landscape on which these processes run, and (e) the supporting

infrastructure. All at a high level of abstraction. The power of ArchiMate lies in the fact that we can create different *views* to address the specific *concerns* of individual *stakeholders*. In this case, we made views for process owners (showing what will happen to their process in the transition), system owners (showing what is expected of them), the corporate data management office (showing where data lives and how it flows), the compliance department (showing a wide variety of aspects) and many others. Being able to generate these views from a single underlying model was the key to success. The above diagram illustrates this way of thinking.



The ArchiMate language was actually the result of a research project, which ran from 2002 to 2004 [4], and involved three industrial partners [1] and five research partners [2]. Soon after the completion of the project, the ArchiMate language was transferred to The Open Group, where it became one of its standards [1]. When ArchiMate was developed, we were clearly only at the start of the transition to the digital age. This resulted in a first version of the language that attempted to unify a business/application/technology perspective by – rather forcefully – imposing a language structure that uses three layers.

Meanwhile, ArchiMate [1, 7] is rapidly becoming a/the leading industry standard for enterprise (architecture) modelling [3] and has, as such, a key role to play in the coordination of enterprise transformations [12].

Since its initial development, the ArchiMate language has evolved. Several issues have been fixed along the way, while new concepts, additional layers, as well as several extensions have been added. This makes it fair to consider the question, now the digital age has clearly taken hold, to what extend ArchiMate meets the enterprise (architecture) modelling needs of the digital age. This resulted in our paper [6], which focused on the question if the current version of the ArchiMate language indeed lives up to the challenges of the digital age.

In this short paper, we focus on the question of how to apply the results of the latter paper. In doing so, we take the increasing digital nature and complexity of society/enterprises into account, as well as the need to use models to attempt to deal with this complexity.

2 Modelling: challenges and recommendations

In [6], we have extensively studied the *aptness* of the ArchiMate language in light of the modelling challenges associated with digital transformation. In doing so, we have identified eleven main challenges, and grouped them in three main categories:

Expressiveness of the language – (1) objects should be allowed to play operand and operant roles, (2) clear separation between objects that represent "things" in the real world, and objects representing information about the real world, (3) ability to deal naturally with the duality of human and digital actors, (4) ability to specify if objects can, should, and/or are allowed, to be

[1] ABN AMRO Bank, Stichting Pensioenfonds ABP, and the Dutch Tax and Customs Administration (Belastingdienst)

[2] Telematica Instituut, Ordina Institute, Centrum voor Wiskunde en Informatica (CWI), Leiden Institute for Advanced Computer Science (LIACS), and the Radboud University

[3] The support for this claim lies in the steady growth of the number of certified professionals as well as the popularity of the ArchiMate topic on Google trends.

uniquely identified, and what the expected reliability is, (5) ability to specify modalities on relationships, (6) ability to capture (potential) value(s) of products and services, and how this results in value co-creation between providers and consumers of services by way of resource integration, and (7) capture design decisions and their motivation, at different levels of specificity with regards to implementation decisions.

Managing the spectrum of modelling concepts – (8) a way to manage the set of modelling concepts, balancing the needs of domain, and purpose, specificity, the need for standardisation, and comprehensibility of the modelling language, (9) provide a structure that allows for a consistent use of abstractions across relevant aspects of the enterprise, and (10) how to ground enterprise models in terms of natural language like verbalisations, without losing the advantages of having compact notations (as well).

Enterprises are in motion – (11) How to capture the motion of an enterprise, it terms of its current and desired affairs.

These are serious challenges. It is important to keep in mind that modelling languages, and ArchiMate is no exception in this, are subject to the same laws of nature to which all other (natural) languages are. More specifically, languages will *evolve* to meet the needs of people who speak/use the language[4] [3].

In [6], we have proposed the following to improve ArchiMate in light of the above mentioned challenges: (A) make the language design more modular, (B) rely on grounded enterprise modelling [11], (C) add more semantic precision, (D) improve the way abstraction/layers are used, (E) accommodate value co-creation, (F) capture design decisions, and (G) add support to manage constant change.

Both the *challenges* and the *recommendations* were based on two sources: (1) a broad knowledge of the available literature, both practical and scientific, combined with (2) our extensive (combined) experience in teaching and using the language in practice.

3 Examples of the challenges

In this section, we provide two examples to illustrate the challenges as well as our proposed direction for the further evolution of the ArchiMate language. Both examples are inspired by real world, practical situations in which the ArchiMate modelling language was used.

3.1 Adding detail; one step at a time

As a first example, imagine a consultant having been asked to craft a model of the *current* situation in an organisation that operates within a certain industrial domain (e.g. finance, manufacturing, logistics, etc.). Also imagine that this consultant has never worked in the specific industry, while having to work with architects and stakeholders with very little background in the ArchiMate language. In practice, this happens more often than one might think.

Experience shows that, in situations like these, it helps to break down the modelling challenge into manageable pieces [10]. A method that we often use is to first brainstorm all the components that should go in the model – preferably using sticky notes – and later decide what type of concept they are, and how they are related. From a modelling perspective, this requires the capability to deal with the fact that the (meta) *type* of a concept used in a model is increasingly refined during the modelling process [10].

[4] In a recent interview for [5], Stijn Hoppenbrouwers introduced the verb 'languaging' to stress this dynamic nature of language.

The architecture of the ArchiMate language [8] does involve the stepwise refinement of the (meta) types of the concepts and relations used in an ArchiMate model. The ArchiMate standard [1] (and more importantly, most ArchiMate tooling) only allows modellers to use the 'leaf' (meta) types, such as *Business role*, *Application component*, *Artefact*, etc. Even though this might be seen as a choice that might, at first, simplify the language, during an actual modelling process this choice proofs to cause problems.

The underlying issues pertain to challenge (8), but also has some elements related to challenges (4) and (11). We believe that the solution lies in a combination of using (B) and (D), in other words *grounded enterprise modelling* [10] and improving the way in which *abstraction/layers* are used

As discussed in [10], we also consider it to be a good way to teach learners (both students and practitioners) how to use ArchiMate, by starting with the main building blocks and add more detail/sophistication as you move along.

The 'solution' that we now use in absence of these mechanisms is to start with brown paper/sticky notes for a first cut on the model. When this is completed, we translate the sticky notes to an ArchiMate model in a computer system. The main disadvantage is that we lose the connection between what we model with the stakeholders and what we will eventually show them through our analyses. It is our experience that this increases the chance of a communications breakdown.

3.2 Human/machine interaction

As a second example, consider situations where human and computer/robot/machine systems collaborate to create products and services. This is no longer confined to the realm of physical manufacturing anymore (e.g. where robots assist human workers in creating complex devices). The same ideas have been implemented in e.g. the hospitality services where robots such as *Pepper* which (or should we say: who?) collaborates with human actors in airport terminals [14].

This example pertains to challenge (3), while also being related to challenge (2). We suggest that the solution of this type of challenge lies in the direction of (D).

In the context of ArchiMate, the most straight-forward way to model the collaboration between *Pepper* and human actors in achieving a task, is to use the modelling *Collaboration* construct. This construct exists in each of the ArchiMate layers but is intended to *only* aggregate elements *from that same layer*. Aggregating an element from two different layers (e.g. an actor from the business layer and, say, equipment from the technical layer) is not presently allowed in ArchiMate.

In the absence of a 'proper' solution in order to model the collaboration between human/business actors and more technical/infrastructure actors, we used a simple work around: we created a *grouping* that is composed of a *business actor* and a *device* and assigned these to a *process*. In essence we created our own version of the *collaboration* concept in this way.

4 Conclusion

In this short paper we considered two examples of challenges facing the ArchiMate language in the context of the digital transformation, while also providing practical suggestions on how to deal with these in practice.

We firmly believe in continuing to evaluate and evolve the language in light of challenges of practitioners in the field. To do this well, we would like to invite practitioners to create a corpus of cases by sharing real-world experiences of using enterprise models (in ArchiMate, and/or other enterprise modelling languages) to solve digital transformation challenges. Ideally, published cases include a factual account and evaluation of the fitness of the language to solve the case at hand.

References

[1] I. Band, T. Ellefsen, B. Estrem, M.-E. Iacob, H. Jonkers, M. M. Lankhorst, D. Nilsen, H. A. Proper, D. A. C. Quartel, and S. Thorn. *ArchiMate 3.0 Specification*. The Open Group, 2016.

[2] S. J. Berman. Digital transformation: opportunities to create new business models. *Strategy & Leadership*, 40(2):16-24, 2012.

[3] M. Bjeković, H. A. Proper, and J.-S. Sottet. Enterprise modelling languages - just enough standardisation? In B. Shishkov, editor, *Business Modeling and Software Design - Third International Symposium, BMSD 2013, Noordwijkerhout, The Netherlands, July 8-10, 2013, Revised Selected Papers*, volume 173 of *LNBIP*, pages 1-23. Springer, Heidelberg, Germany, 2014.

[4] H. Bosma, H. W. L. ter Doest, and M. Vos. Requirements. Technical Report ArchiMate Deliverabe D4.1, TI/RS/2002/112, Telematica Instituut, December 2002.

[5] B. van Gils. *A gentle introduction to Data Management - a thorough overview, balancing academic rigor and practical relevance*. Van Haren Publishing, forthcoming.

[6] B. van Gils and H. A. Proper. Enterprise modelling in the age of digital transformation. In R. A. Buchmann, D. Karagiannis, and M. Kirikova, editors, *The Practice of Enterprise Modeling - 11th IFIP WG 8.1. Working Conference, PoEM 2018, Vienna, Austria, October 31 - November 2, 2018, Proceedings*, volume 335 of *LNBIP*, pages 257-273. Springer, Heidelberg, Germany, 2018.

[7] M. M. Lankhorst, S. J. B. A. Hoppenbrouwers, H. Jonkers, H. A. Proper, L. van der Torre, F. Arbab, F. S. de Boer, M. Bonsangue, M.-.E. Iacob, A. W. Stam, L. Groenewegen, R. van Buuren, R. J. Slagter, J. Campschroer, M. W. A. Steen, S. F. Bekius, H. Bosma, M. J. Cuvelier, H. W. L. ter Doest, P. A. T. van Eck, P. Fennema, J. Jacob, W. P. M. Janssen, H. Jonkers, D. Krukkert, D. van Leeuwen, P. G. M. Penders, G. E. Veldhuijzen van Zanten, and R. J. Wieringa. *Enterprise Architecture at Work - Modelling, Communication and Analysis*. The Enterprise Engineering Series. Springer, Heidelberg, Germany, 4th edition, 2017.

[8] M. M. Lankhorst, H. A. Proper, and H. Jonkers. The anatomy of the ArchiMate language. *International Journal of Information System Modeling and Design*, 1(1):1-32, 2010.

[9] N. Negroponte. *Being Digital*. Vintage Books, New York, New York, 1996.

[10] H. A. Proper, M. Bjeković, B. van Gils, and S. J. B. A. Hoppenbrouwers. Towards a Multi-Stage Strategy to Teach Enterprise Modelling. In D Aveiro, G. Guizzardi, S. Guerreiro, and W. Guédria, editors, *Advances in Enterprise Engineering XII - 8th Enterprise Engineering Working Conference, EEWC 2018, Luxembourg, May 28 - June 1, 2018, Proceedings*, volume 334 of *LNBIP*, pages 181-202. Springer, Heidelberg, Germany, 2018.

[11] H. A. Proper, M. Bjeković, B. van Gils, and S. J. B. A. Hoppenbrouwers. Towards grounded enterprise modelling. In C. Debruyne, H. Panetto, G. Weichhart, P. Bollen, I. Ciuciu, M.-E. Vidal, and R. Meersman, editors, *On the Move to Meaningful Internet Systems. OTM 2017 Workshops - Confederated International Workshops, EI2N, FBM, ICSP, Meta4eS, OTMA 2017 and ODBASE Posters 2017, Rhodes, Greece, October 23-28, 2017, Revised Selected Papers*, volume 10697 of LNCS, pages 141-151. Springer, Heidelberg, Germany, 2018.

[12] H. A. Proper, R. Winter, S. Aier, and S. de Kinderen, editors. *Architectural Coordination of Enterprise Transformation*. The Enterprise Engineering Series. Springer, Heidelberg, Germany, 2018.

[13] J. Ross, C. M. Beath, and M. Mocker. *Designed for digital: How to architect your business for sustained success*. Management on the Cutting Edge. MIT Press, 2019.

[14] C. Singh. Humanoid robot named pepper will help passengers at New Zealand airport, October 2019.

[15] D. J. Snowden and M. E. Boone. A leader's framework for decision making. A leader's framework for decision making. *Harvard Business Review*, 85(11)(11):68-76, November 2007.

[16] J. M. Tribolet, P. Sousa, and A. Caetano. The Role of Enterprise Governance and Cartography in Enterprise Engineering. *Enterprise Modelling and Information Systems Architectures*, 9(1):38-49, June 2014.

[17] G. Westerman, D. Bonnet, and A. McAfee. *Leading digital - Turning technology into business transformation*. Harvard Business Review Press, 2014.