Requirements Engineering for Model-Based Enterprise Architecture Management with ArchiMate

Dominik Bork[1(✉)], Aurona Gerber[2,3], Elena-Teodora Miron[1],
Phil van Deventer[2], Alta Van der Merwe[2], Dimitris Karagiannis[1],
Sunet Eybers[2], and Anna Sumereder[1]

[1] Research Group Knowledge Engineering, University of Vienna,
Waehringer Street 29, 1090 Vienna, Austria
{dominik.bork,elena-teodora.miron,dimitris.karagiannis,
anna.sumereder}@univie.ac.at
[2] Department of Informatics, University of Pretoria,
Hatfield, Pretoria 0083, South Africa
{aurona.gerber,phil.vandeventer,alta.vandermerwe,sunet.eybers}@up.ac.za
[3] CSIR Center for AI Research (CAIR), Brummeria, Pretoria, South Africa

Abstract. The role of information systems (IS) evolved from supporting basic business functions to complex integrated enterprise platforms and ecosystems. As a result, enterprises increasingly adopt enterprise architecture (EA) as a means to manage complexity and support the ability to change. We initiated a study that investigates the pivotal role of enterprise architecture management (EAM) as an essential strategy to manage enterprise change and within this larger context, specifically how the ArchiMate modeling language can be enhanced with capabilities that support EAM. This paper reports on the evaluation of an EA modeling tool (TEAM) which has been enhanced with EAM capabilities. The evaluation was performed by a focus group of enterprise architects that attended a workshop and applied the tool to an EAM case study. The evaluation results, requirements as well as a conceptualization for further development are presented and are of value for both, enterprise architecture researchers and enterprise architects.

Keywords: Enterprise architecture management · ArchiMate
Requirements engineering · Focus group

1 Introduction

"The digitization of our society changes the way society work, communicate and collaborate." [1] Similarly, digitization or digital transformation changes the way enterprises create value. Traditionally, enterprises created value by selling products or by providing services to customers with direct and simple business models. The digital transformation significantly changed these business models

(e.g., toward platform ecosystems [2]), customer involvement (e.g., value co-creation [3]), and product/service systems [4]. These changes are either driven or supported by information systems and therefore directly influence the enterprise architecture (EA). Thus, it is of utmost interest for enterprises to manage their EA as well as to manage their enterprise using EA, collectively termed *enterprise architecture management* (EAM) [5,6].

The Open Group Architecture Framework (TOGAF) and the ArchiMate [7] modeling language are widely adopted EA standards. However, both have limited support for corporate EA management because of the sole focus on the methodological and modeling language aspects of EA, respectively. Supporting these standards with computerized modeling environments creates opportunities to support EAM by for instance exploiting conceptual models as knowledge base for advanced management support [8]. Our study therefore investigates how EA modeling with proper tooling supports enterprise architecture management.

Adopting the action design research paradigm that incorporates evolutionary design with short evaluation/feedback loops [9], we implemented a first prototype of the TOGAF-based Enterprise Architecture Management (TEAM) modeling tool[1] that implements the Archimate 3.0.1. standard [7]. This paper reports on an evaluation/feedback loop of TEAM that used a carefully designed focus group. The focus group introduced eight EA experts to both EAM as well as the TEAM tool using a case study in a workshop scenario. In depth feedback was collected from the EA experts on the functionality of the tool, as well as input on how a modeling platform could support the two focus areas of EAM namely: (1) managing the EA of an enterprise, and (2) managing the enterprise using EA. This feedback was consolidated into advanced requirement themes for the second prototype version of TEAM.

The remainder of this paper is structured as follows: foundations are presented in Sect. 2 and in Sect. 3 the research design for the evaluation of TEAM is discussed. Section 4 consolidates the results by means of a set of requirement themes for advanced EAM. Finally, Sect. 5 concludes the paper.

2 Foundations

2.1 Enterprise Architecture Management

Enterprise Architecture Management (EAM) is a relatively recent perspective within the domain of EA. EAM is broadly defined as "*management practice that establishes, maintains and uses a coherent set of guidelines, architecture principles and governance regimes that provide direction for and practical help with the design and the development of an enterprise's architecture in order to achieve its vision and strategy*" [6]. In the 80's John Zachman, often described as the father of EA, adopted a systems engineering approach to develop the Zachman Framework for Enterprise Architecture or Zachman Framework[TM] (ZFEA) [10]. The

---

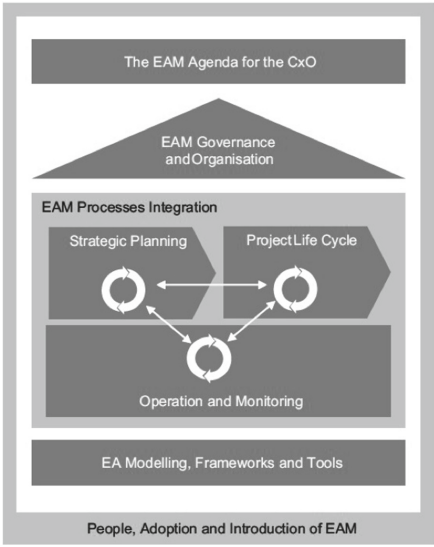[1] The tool is freely available on the OMiLAB TEAM project site at: http://austria.omilab.org/psm/content/team/info, last visit: 08.05.2018.

Fig. 1. EAM building blocks [6].

Fig. 2. ArchiMate 3.0 framework [7].

ZFEA had as primary goal the specification of a universal set of descriptive representations from different views for enterprises as socio-technical systems [10,11]. Originally, EAM was thus focused on the development of the enterprise architecture itself in an attempt to manage the complexity of modern enterprises [6, p. 13].

In the 90's the focus of EAM shifted from modeling the enterprise towards *alignment* of the different aspects within an enterprise [6, p. 14]. To assist with this alignment, several EA frameworks were proposed and EAM literature discussed various enterprise alignment aspects e.g. the execution of strategy through business-IT alignment [12–15]. Lapalme [16] summarized the EAM notions of the time by describing three schools of thought related to EA namely: (1) *Enterprise-wide IT platform (EIT)*, concerned with effective enterprise strategy execution and operation through IT-Business alignment; (2) *Enterprise (E)*, concerned with effective enterprise strategy implementation through execution coherency; and (3) *Enterprise-in-environment (EiE)*, concerned with fostering organizational learning by aligning the various facets of the enterprise such as governance structures and IT capabilities [16].

The most recent developments in EAM include the use of EA for strategic business management [6,17]. This strategic EAM standpoint incorporates all the previous EAM perspectives but specifically adopts the extended view that EA is a *management philosophy* and *executive management and governance function* that should, for instance, be used to manage holistic and sustainable enterprise transformation, alignment and integration [6, p. 57]. Given this perspective, EAM is a multidimensional function that influences all aspects of an enterprise,

including its organizational culture, communication practices and operations. Ahleman et al. [6, p. 42] proposed a model that depicts the essential EAM building blocks. As is shown in Fig. 1, the main and outside container for EAM indicate the 'soft factors' that are important within an organization. Stakeholder buy in into EAM is crucial when, for instance, altering organizational culture and changing individual behavior. Figure 1 furthermore depicts the role of EAM as a chief executive officer agenda at the top. The EAM governance and organization role deals with the manner in which EAM is institutionalized within an organization. Furthermore, the integration of EA into organizational processes includes the embedding of EAM into strategic planning, project life cycles and organizational operations and monitoring, which all have to do with the day-to-day operations of an enterprise. EAM building blocks have to include EA frameworks, modeling and tools, which represent and include the existing body of knowledge and best practices regarding enterprise architecture [6, p. 42]. Since ArchiMate is one of the dominantly used EA languages, conceptual modeling methods in general and ArchiMate in particular are briefly introduced in the following to establish a theoretic foundation for the rest of the paper.

2.2 ArchiMate, TOGAF and Conceptual Modeling Methods

ArchiMate is a standard of the Open Group that describes an enterprise architecture modeling language [18]. ArchiMate was originally developed by a team from Telematica Institute in the Netherlands to model an EA within and across business domains [19]. ArchiMate adopts a layered view on an enterprise depicted in the ArchiMate Framework where the core entities of an enterprise are categorized along two dimensions (layers and aspects) as shown in Fig. 2. In addition, ArchiMate adopts a service-oriented model where each layer provides services to the layers above it. ArchiMate focuses on specifying a modeling standard for enterprise architecture. By contrast, TOGAF, the Open Group Architecture Framework specifies guidelines for designing, planning, implementing, and governing an enterprise information technology architecture [14]. When the implementation of ArchiMate is discussed, it is often done within the TOGAF approach to provide the context of an enterprise architecture project [20].

Any conceptual modeling methods such as ArchiMate facilitates the management of complexity by applying abstraction. According to [21], modeling methods are composed of *modeling language*, *modeling procedure*, and *mechanisms & algorithms*. A modeling language can be further decomposed into: *syntax*, the available language elements; *notation*, the graphical representation of syntactic elements; and *semantics*, the meaning of the syntactic elements. The modeling procedure describes steps and results of utilizing a modeling method in order to create valid models. Lastly, mechanisms & algorithms define the model processing functionality that is provided by the modeling method (e.g., simulations and queries).

Conceptual modeling methods are used to create abstract representations of some part of the real world for *"human users, for purposes of understanding and communication"* [22]. This traditional view is still valid, however, nowadays

conceptual models are also viewed as a formalized knowledge base that enables machine processing and intersubjective understanding [23]. Conceptual modeling methods therefore not only target the best abstraction level for a specific domain by means of a metamodel, but also the enrichment of the modeling language with proper functionality to increase the value of the models. This approach to conceptual models is adopted by OMiLAB, the platform used for the development of TEAM, which is discussed in the next section.

2.3   The Open Models Laboratory (OMiLAB)

The Open Models Laboratory (OMiLAB, www.omilab.org) is an open platform for the conceptualization of modeling methods, combining open source and open communities with the goal of fostering conceptual modeling. OMiLAB constitutes a high number of international contributors [24]. Almost 50 different modeling methods have already been successfully conceptualized within OMiLAB [25], such as Multi-Perspective Enterprise Modeling (MEMO) [26] and SOM [27]. A more comprehensive view on successful conceptualizations within OMiLAB is given in [25]². The TEAM tool was implemented as a project within OMiLAB.

3   Research Design: Focus Group Evaluation

As stated, we report on the evaluation of the first prototype version of the TEAM modeling tool. In order to obtain the in depth feedback required, we adopted a *focus group* (FG) as research method. A FG is a qualitative research method that is effective when collecting data about the opinions of people or how they think, feel, or act regarding a specific topic [28]. The method is particularly useful for collecting data in complex scenarios where specialized knowledge is required. Using a FG for data collection in our evaluation of TEAM was therefore applicable because EAM has an extensive scope and we were particularly interested in the opinions of the participants (EA experts and practitioners) regarding EAM requirements when using TEAM. As a prerequisite, the FG needs to be designed in such a way that participants are able to provide high-quality, in-depth feedback. We therefore designed the FG as a workshop specifically aimed at EA experts and practitioners with several years of experience, and we included carefully developed feedback mechanisms that triangulate in order to collect data. Because the experience of the participants varied, we created a baseline by introducing the necessary background in the workshop. The workshop was structured as follows:

1. **Session 1: Enterprise Architecture Management:** During this session the theory, history and focus of EAM were introduced, followed by the focus areas of EAM namely (1) managing the EA of an enterprise; and (2) managing the enterprise using the EA.

---

² Full method repository is available at http://austria.omilab.org/psm/tools, last visit: 08.05.2018.

2. **Session 2: ArchiMate and TEAM:** This session consisted of two parts namely: (1) an overview of Archimate (most participants were familiar with ArchiMate and TOGAF); and (2) an introduction to the TEAM tool.
3. **Session 3: Focus Group Case Study:** In this session a detailed case study was introduced where participants were guided to use the TEAM tool. For more details of the case study, see Sect. 3.1.
4. **Session 4: Focus Group Feedback:** In this session the participants were asked to give high-level feedback on the TEAM tool, EAM and further development, especially given their experience, see Sect. 4.

The data was collected from eight workshop participants, of which seven were established EA specialists either working full-time as enterprise architects within organizations or as EA consultants responsible for projects initiating EA at various levels within organizations. The group included: (a) professional consultants and trainers who specialized in EA and ArchiMate; (b) professional users who employ EAM frameworks and tools in their respective enterprise or public administration and who are in charge of the EAM management; as well as (c) academics who research and teach EAM at graduate and post-graduate level but with previous experience in EA implementation. The next section presents the case study which was used to evaluate the TEAM tool.

3.1   Focus Group Case Study

The *Charlie's Auto Repair Shop* case study was employed to evaluate prototype one of TEAM. After an introduction to TEAM the experts were asked to model each of the three parts of the case. 45 min was allocated for each modeling task and 15 min were used for discussion. A final one hour long session was dedicated to discussing: (a) the quality and eventual shortcomings of the case itself given EAM; (b) the completeness and accuracy of the mapping between the ArchiMate standard and the tool; (c) usability of the current, and requirements for future versions of the TEAM tool; and (d) usefulnesses of the TEAM tool functionality for EAM.

Case Description. In line with the idea that EA and its management play a pivotal role in enterprise transformations, the case study's focus is on the transformation of a traditional car repair SME into a car repair-as-a-service business - strongly reliant on IT and the business opportunities enabled by it. *Charlie's Car Repair Shop's* original business model focused on providing parts and specialized repair for old-timers. Information technology played a marginal role in the back office of the business for administrative and bookkeeping activities. A management change triggered the modification of the business model. The assets of the old business - repair facility and machinery, spare parts, and mechanical expertise - will now be leveraged with the support of IT to realize a car repair-as-a-service business model where old-timer owners can book the assets to work on their cars. The customers will be charged usage-based fees for the different service components.

The underlying motivation is to monetize the old-timer owner's love and knowledge about cars. These persons are known to the repair shop as having two characteristics important for the repair-as-a-service business model. They tend to be financially well off and are able to invest in the costly maintenance and repairs. Moreover, they care about a particular car and also have a lot of knowledge about its mechanics.

Following a general introduction to the case, the first part of the case study detailed the new strategy defining *goals*, the *expected outcomes* as well as the *necessary capabilities*. The second part then derived exemplary *business services* to be offered to the clients, *technology services* as well as *business processes* necessary for the provisioning of the new services. The identified services were also linked to their *technology assets* like software and hardware. Lastly, the third part described the *physical elements* which establish the "execution environment" for the services, like repair spaces, repair machinery etc. These physical elements were linked to the previously defined technology assets.

In alignment with the ArchiMate 3.0.1 standard and following the TOGAF framework, the case includes also Internet of Things and physical assets - thus expanding the EAM space considered in previous versions of the standard.

Exemplary Case Solutions. TEAM provides the full spectrum of the ArchiMate 3.0.1 modeling language. The language concepts are grouped into the ArchiMate 3.0.1 layers - called *model types*: *strategy layer*, *business layer*, *application layer*, *technology layer*, *physical layer*, *implementation/migration layer*, *motivation layer*, and *analysis model*. While each of the model types contains only the concepts specific to it, e.g. a *business service class* is included in the business layer, the analysis model is a container of all ArchiMate 3.0.1 classes thus allowing a top to bottom model for the whole EA. For purposes of this case study participants were encouraged to use the analysis model type. Increasing readability within the model is achieved by using the *grouping class* to graphically compose objects which also belong semantically together (seen in Fig. 4 by the dotted boxes).

In the first part, *"The Strategy"*, the participants needed to cognitively differentiate between a *goal* and an *outcome* as well as between a *capability* and a *resource*. To ease the identification of the correct ArchiMate concepts, cues are provided in the case description, for example *"...the need to build up the auto shop's IT Operations and Management capabilities"*, which points the participant to the concept to use, i.e., *capability* and its name *IT Operations and Management*. One solution to the first part of the case study is represented in Fig. 3.

Business services need to support the goals defined for the new strategy. On their part they must be aided by appropriate business processes as well as technology services. For example, the newly instituted *Repair space rental* service triggers a newly defined business process which in itself employs the *Billing* technology service. In addition, not shown in the case, one could include a *Rental space booking* application running on a web-based client-server hardware
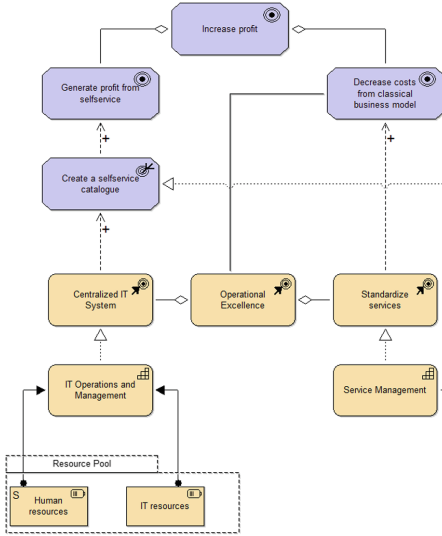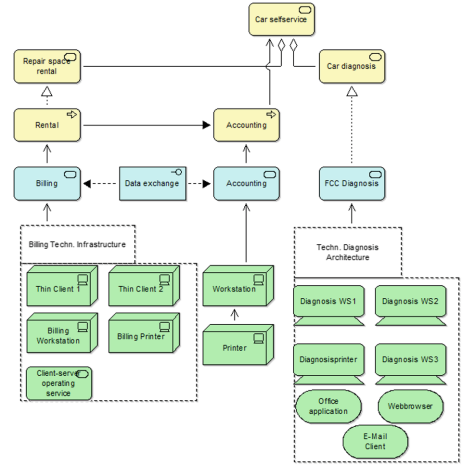
Fig. 3. Strategy model in TEAM.

Fig. 4. Services and processes model in TEAM.

which allows customers to book their repair slots on-line. For practicing purposes and due to limited time, only an excerpt of the services and processes involved was discussed in the case study. One possible solution of the second part is presented in Fig. 4.

The new business model also triggered changes on equipment level (see Fig. 5). While previously the machinery necessary for mechanical repairs did not need any ICT, now, with the time-dependent billing of usage, each machine must be able to "identify" at least the client to be billed as well as the start and end time of the rental. To this end, car repair machines must be equipped with card reading devices and enabled to transmit the necessary information to the *Billing application* and ultimately to the *Billing technical service*.

The new language concepts available in ArchiMate 3.0.1 on the strategy - and on the physical layer enable the enterprise architects to create a comprehensive model stack on which different analytics can be applied, both at design but also at "run" time, thus enabling the enterprise architect's management capabilities.

4   Evaluation and Advanced Requirements for EAM

The evaluation feedback was obtained during the focus group case study and feedback sessions. During the case study session where the tool was used by the participants, feedback was obtained through interaction and discussion with the participants, as well as through documented observations by the research team when supporting the participants.

4.1   Focus Group Case Study Evaluation Results

The participants were asked to evaluate the TEAM tool, EAM and further development, especially given their experience. Questions were prepared to guide the feedback. During the session, the discussion was recorded and transcribed. All feedback is described in the following and collated into the requirement themes reported in Sect. 4.2.

Workshop participants easily found their way through the first two parts of the case study as it used familiar concepts and terminology. The third part, which relies heavily on new modeling constructs defined in ArchiMate 3.0.1, required a bit more working time.

TEAM was easily understood and handled by the participants. They remarked positively on the intuitive use of modeling concepts and their graphical representation. Moreover, participants found it useful that the use of connectors was limited by the tool only to those allowed according to ArchiMate 3.0.1.

4.2   Advanced Requirements for EAM

For eliciting the requirements, we analyzed the focus group feedback from the workshop participants from both the case study and the feedback sessions and condensed the feedback into four advanced EAM requirement themes. Each requirement theme is described using the aspects: *Rationale*, detailing the ratio behind it; *Metamodel Requirements*, describing the requirements on metamodel level; *Implementation*, indicating how the requirement theme should be implemented in a modeling tool; and *Execution*, exemplifying the execution by the modeler. Finally, we indicate in Sects. 4.3 and 4.4 how these requirements could be incorporated conceptually into the next versions of the TEAM tool.

Theme 1 - Information Management

Rationale: It is reasonable to have the possibility of attaching comments and
  descriptions to the ArchiMate concepts. The generic nature of these attributes
  enables the modeler to document further properties - besides solely the name
  - for each concept. Moreover, such meta data can be used for analysis as well
  as possible future developments. For example, the descriptions can reveal,
  which additional attributes might be required.
Metamodel Requirements: Two new attributes, termed *Description* and *Notes*
  of datatype *string*, should be introduced into the TEAM metamodel. They
  should be provided for each ArchiMate concept.
Implementation: The two attributes shall be adding to the metamodel and their
  values should be stored with the models. Visualization and editing of these
  attributes shall be enabled.
Execution: The modeler shall be able to see and edit description and notes in
  the properties of each modeled concept.

Theme 2 - Lifecycle Management

Rationale: When dealing with ICT, lifecycle management plays a vital role. Questions like "until when are software systems supported with updates?", or"when becomes a certain component invalid?" are crucial for EAM. There should be different kinds of dates in the various layers. For example, the application layer components should have attributes for licenses, which can be outdated or invalid. Time elements in the model should offer possibilities regarding queries and a kind of lifecycle management in the model.

Metamodel Requirements: In general, there should be one time attribute for nearly all ArchiMate concepts. In addition, the attributes purpose and name should vary from layer to layer, as there are specific requirements and types of dates. A valid until date should be used for application layer concepts.

Implementation: The new attributes should be visualized to the modeler for editing. Additionally, two *queries* should be realized that enable the modeler to efficiently list in-/valid application components of the current model.

Execution: The modeler should define a date at the beginning of the query execution. The tool then lists all instances that fulfill the query criteria. It should be possible to click on the elements in the list to navigate directly to the corresponding instance in the model.

Theme 3 - Responsibility Management

Rationale: The assignment of responsibilities should enforce a higher level of engagement and ease EAM. Thus, technology layer components should be assigned to actors/roles in the business layer. To its end, a visualization functionality shall be realized that displays the connections between the components on the different layers.

Metamodel Requirements: To combine business and technology layer, semantic links between concepts of those two layers should be added. Such semantic links might be realized as references or pointers that are specified at the corresponding metamodel classes.

Implementation: Reference attributes between technology and business layer should be added for selected elements of the two layers. Furthermore, a functionality shall be provided that generates, starting from a technology layer model, the list of corresponding actors/roles of the business layer.

Execution: The modeler shall be enabled to edit the specific reference attributes in order to semantically link concepts of the two layers. Moreover, the modeler shall be enabled to generate the list of responsibilities. All list items shall enable direct navigation to the corresponding instances in the models.

Theme 4 - Business Continuity Management

Rationale: In today's fast changing business models, built on top of complex ecosystems, failure and service unavailability is inevitable. Enterprises therefore aim to establish a business continuity management (BCM) strategy. Conceptual modeling and modeling tools can play a vital role in BCM [29,30]. A

prerequisite for managing crisis events is to be aware of the mutual effects different EA instances have on each other. A semantic link between business and technology models should be established. The goal is to identify the impact of a technology element (e.g., function, process, interface, event, service) on a business layer element of the same type.

Metamodel Requirements: Especially concerned are function, process, interface, collaboration, interaction, event and service of the technology and the business layer. A reference attribute, which relates the elements of these two layers shall be added.

Implementation: 'Influence on' reference attributes shall be used to define relationships between elements of the technology layer and the business layer.

Execution: The reference attributes shall be editable by the modeler, thereby enabling the efficient specification of relationships. Moreover, a functionality shall be realized that queries the models for these attribute values and lists all relationships. This functionality shall be parameterizable by the type of concepts interested in. The modeler may e.g., parameterize a certain business function to be out of order and receive a list of technology components related to this function.

4.3   Conceptualization of Modeling Tools with ADOxx

Meta modeling platforms are used for the development of modeling tools. They raise the abstraction level of modeling tool development to a more elaborate level that is adequate for method engineers. The goal is to enable also non-programmers to realize their modeling tools. This is achieved by providing a rich set of pre-configured functionality the user then only needs to adapt to his/her domain. Moreover, users can benefit from existing tool developments on a certain platform.

ADOxx [31] is a meta modeling platform that has been successfully used in research and industry. The aim of the platform is to raise the abstraction level of modeling tool development to a less implementation-specific level [32]. ADOxx takes care of all domain-independent and non-functional requirements like model management, user management, storage, and user interaction. What is left to be done by the tool engineer is according to [33]: (1) configure the specific modeling language by referring it's concepts to the meta concepts of the platform; (2) provide a proper visualization for the concepts and combine concepts into logical clusters, i.e., model types; and (3) realize additional functionality like model transformations, model queries, or simulations.

4.4   The TEAM Tool

Figure 6 visualizes a screenshot of the TEAM modeling tool realized with the ADOxx metamodeling platform. TEAM realizes all layers of ArchiMate 3.0.1 following the TOGAF framework, as well as the requirement themes described in Sect. 4. This enables TEAM to do basic ArchiMate modeling and TOGAF support as well as acting as a facilitator for EAM. Besides the modeling palette,
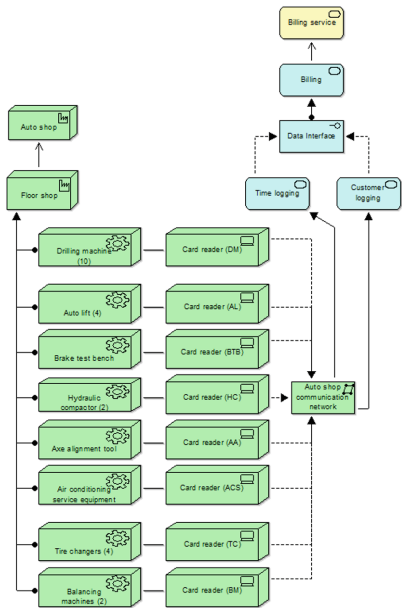
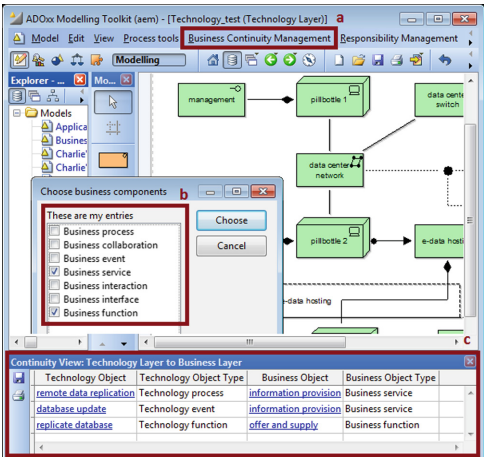**Fig. 5.** Equipment model in the TEAM tool.

Fig. 6. Executing model queries in the TEAM tool.

listing the available ArchiMate language concepts of the currently opened model on the left side, the tool also comes with an intuitive context menu that features the model queries - e.g., for the lifecycle management - and the additional functionality - e.g., for the business continuity management.

At the top of Fig. 6, indicated by the letter 'a' is the menu bar implemented for the business continuity management and responsibility management. When clicking on 'a', the modeler is presented a multi-select box (see Fig. 6 'b') where he/she can de-/select the ArchiMate concepts he/she is interested in, thereby parameterizing the query. After confirming the selection, TEAM executes the query and visualizes the query result window (see Fig. 6 'c') on the bottom). The results window lists the relationships between the selected business object type instances and the technology objects of the currently opened models (in Fig. 6 *Business service* and *Business function* were selected).

5   Conclusions and Future Research

This paper reported on an action design science research project that targeted the identification and conceptualization of requirements for an advanced enterprise architecture management approach that integrated the TOGAF framework with the ArchiMate 3.0.1 modeling language. The data was collected using a workshop focus group design where in-depth feedback was obtained during tool

use in a case study and a feedback session. The feedback was obtained from eight EAM experts and practitioners and was condensed into a set of requirement themes for advanced EAM. Finally, the realization of these requirements with the ADOxx metamodeling platform as a project within the Open Models Laboratory (OMiLAB, www.omilab.org) was briefly illustrated.

Intuitive usage of the modeling tool was evaluated positively by the focus group. Results for the modeling tasks differed. The case study showed, that practitioners were able to create good models for commonly used ArchiMate layers like application and technology. By contrast, support by the moderators was necessary to achieve good results for the new ArchiMate 3.0 layers like motivation. Focus group participants expressed a strong need to support managers and enterprise architects not only with a methodology like TOGAF and an existing language like ArchiMate, but also with a full-fledged modeling environment. Based on the expert feedback, the paper specified requirement themes for advancing model-based EAM. Consequently, EAM has the ability to emerge from being limited to IT experts towards becoming a management tool fostering efficient business operations and the ability to change. This paper finally introduced a first prototype of the TEAM tool, aiming for a tool-based application of advanced EAM.

This research also comes with some limitations. The number of experts was quite low, however we ensured a homogeneous set of participants in the workshop and the discussion. Moreover, some feedback might be biased by the tool that has been used. It is important to differentiate in future design cycles more clearly between the conceptual approach and the tool support.

In future research we will extend the case study with tasks, that utilize some of the advanced features. This extended case study shall then be used to evaluate the second TEAM prototype - eventually leading to a mature modeling environment for advanced EAM. Moreover, we will consider to extend the functionality, e.g., with semantic technologies as proposed in [34,35] and mechanisms for ensuring consistency between the multiple ArchiMate layers [36–38].

References

1. Zimmermann, A., Jugel, D., Sandkuhl, K., Schmidt, R., Schweda, C., Moehring, M.: Architectural decision management for digital transformation of products and services. CSIMQ (6), 31–53 (2016)
2. Tiwana, A., Konsynski, B., Bush, A.A.: Research commentary - platform evolution: coevolution of platform architecture, governance, and environmental dynamics. Inf. Syst. Res. **21**(4), 675–687 (2010)
3. Prahalad, C.K., Ramaswamy, V.: Co-creation experiences: the next practice in value creation. J. Interact. Mark. **18**(3), 5–14 (2004)
4. Mont, O.K.: Clarifying the concept of product-service system. J. Clean. Prod. **10**(3), 237–245 (2002)

5. Matt, C., Hess, T., Benlian, A.: Digital transformation strategies. Bus. Inf. Syst. Eng. **57**(5), 339–343 (2015)
6. Ahlemann, F., Stettiner, E., Messerschmidt, M., Legner, C. (eds.): Strategic Enterprise Architecture Management. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-24223-6
7. The Open Group: The open group: ArchiMate 3.0.1 specification (2017). 07 Nov 2017
8. Pittl, B., Bork, D.: Modeling digital enterprise ecosystems with ArchiMate: a mobility provision case study. Serviceology for Services. LNCS, vol. 10371, pp. 178–189. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-61240-9_17
9. Sein, M.K., Henfridsson, O., Purao, S., Rossi, M., Lindgren, R.: Action design research. MIS Q. **35**(1), 37–56 (2011)
10. Zachman, J.: A framework for information systems architecture. IBM Syst. J. **26**, 276–292 (1987)
11. Zachman, J.A.: The Concise Definition of The Zachman Framework by: John A. Zachman (2008)
12. IFIP-IFAC Task Force: GERAM: Generalised Enterprise Reference Architecture and Methodology, Version 1.6.3. Technical report March, Integration, IFIPIFAC Task Force on Architectures for Enterprise (1999)
13. Simon, D., Fischbach, K., Schoder, D.: An exploration of enterprise architecture research. Commun. Assoc. Inf. Syst. **32** (2013). Article 1
14. The Open Group: TOGAF, an Open Group standard (2017)
15. de Vries, M., van der Merwe, A., Gerber, A.: Towards an enterprise evolution contextualisation model. In: Proceedings of the First International Conference on Enterprise Systems, Cape Town, South Africa, pp. 1–12. IEEE (2013)
16. Lapalme, J.: Three schools of thought on enterprise architecture. IT Prof. **14**(6), 37–43 (2012)
17. Ross, J.W., Weill, P., Robertson, D.: Enterprise Architecture as Strategy: Creating a Foundation for Business Execution. Harvard Business Press, Brighton (2006)
18. OMG: ArchiMate 3.0.1 Specification. The Open Group, June 2016. http://pubs.opengroup.org/architecture/archimate3-doc/
19. Lankhorst, M. (ed.): Enterprise Architecture at Work: Modelling, Communication, and Analysis. Springer, Berlin/ New York (2005). https://doi.org/10.1007/3-540-27505-3
20. Vicente, M., Gama, N., da Silva, M.M.: Using ArchiMate and TOGAF to understand the enterprise architecture and ITIL relationship. In: Franch, X., Soffer, P. (eds.) CAiSE 2013. LNBIP, vol. 148, pp. 134–145. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-38490-5_11
21. Karagiannis, D., Kühn, H.: Metamodelling platforms. In: Bauknecht, K., Tjoa, A.M., Quirchmayr, G. (eds.) EC-Web 2002. LNCS, vol. 2455, p. 182. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-45705-4_19
22. Mylopoulos, J.: Conceptual modelling and Telos. In: Loucopoulos, P., Zicari, R. (eds.) Conceptual Modelling, Databases, and CASE: an Integrated View of Information System Development, pp. 49–68. Wiley, New York (1992)
23. Bork, D., Fill, H.G.: Formal aspects of enterprise modeling methods: a comparison framework. In: 2014 47th Hawaii International Conference on System Sciences (HICSS), pp. 3400–3409. IEEE (2014)
24. Bork, D., Miron, E.T.: OMiLAB - an open innovation community for modeling method engineering. In: Niculescu, A., Negoita, O.D., Tiganoaia, B. (eds.) 8th International Conference of Management and Industrial Engineering (ICMIE 2017), pp. 64–77 (2017)

25. Karagiannis, D., Mayr, H.C., Mylopoulos, J.: Domain-Specific Conceptual Modeling. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39417-6

26. Bock, A., Frank, U.: Multi-perspective enterprise modeling—conceptual foundation and implementation with ADO*xx*. In: Karagiannis, D., Mayr, H., Mylopoulos, J. (eds.) Domain-Specific Conceptual Modeling, pp. 241–267. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39417-6_11

27. Ferstl, O.K., Sinz, E.J., Bork, D.: Tool support for the semantic object model. In: Karagiannis, D., Mayr, H., Mylopoulos, J. (eds.) Domain-Specific Conceptual Modeling, pp. 291–310. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39417-6_13

28. Freitas, H., Oliveira, M., Jenkins, M., Popjoy, O.: The focus group, a qualitative research method. J. Educ. **1**(1), 1–22 (1998)

29. Benaben, F., et al.: A conceptual framework and a suite of tools to support crisis management. In: Proceedings of the 50th Hawaii International Conference on System Sciences (2017)

30. Rejeb, O., Bastide, R., Lamine, E., Marmier, F., Pingaud, H.: A model driven engineering approach for business continuity management in e-health systems. In: 2012 6th IEEE International Conference on Digital Ecosystems Technologies (DEST), pp. 1–7. IEEE (2012)

31. ADOxx.org: ADOxx Metamodelling Platform (2018) https://www.adoxx.org/live/home, Accessed 11 Mar 2018

32. Efendioglu, N., Woitsch, R., Utz, W.: A toolbox supporting agile modelling method engineering: ADOxx.org modelling method conceptualization environment. In: Horkoff, J., Jeusfeld, M.A., Persson, A. (eds.) PoEM 2016. LNBIP, vol. 267, pp. 317–325. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-48393-1_23

33. Bork, D., Sinz, E.J.: Design of a SOM business process modelling tool based on the ADOxx meta-modelling platform. In: Pre-proceedings of the 4th International Workshop on Graph-Based Tools. University of Twente, Enschede, pp. 90–101 (2010)

34. Gerber, A., der Merwe, A.V., Kotze, P.: Towards the formalisation of the TOGAF contenet metamodel using ontlogies. In: The 12th International Conference on Enterprise Information Systems (2010)

35. Buchmann, R.A., Karagiannis, D.: Enriching linked data with semantics from domain-specific diagrammatic models. Bus. Inf. Syst. Eng. **58**(5), 341–353 (2016)

36. Awadid, A., Bork, D., Karagiannis, D., Nurcan, S.: Toward generic consistency patterns in multi-view enterprise modelling. In: Twenty-Sixth European Conference on Information Systems (2018, in press)

37. Bork, D., Buchmann, R., Karagiannis, D.: Preserving multi-view consistency in diagrammatic knowledge representation. In: Zhang, S., Wirsing, M., Zhang, Z. (eds.) KSEM 2015. LNCS (LNAI), vol. 9403, pp. 177–182. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-25159-2_16

38. Karagiannis, D., Buchmann, R.A., Bork, D.: Managing consistency in multi-view enterprise models: an approach based on semantic queries. In: Twenty-Fourth European Conference on Information Systems (ECIS) (2016). Research Paper 53