Explainability in enterprise architecture models

Elena Romanenko[1]

[1]Free University of Bozen-Bolzano, Universitätsplatz 1 - piazza Università, 1, Bozen-Bolzano, Italy – 39100

Abstract
Providing explainability to enterprise architecture models is a highly important task. The paper reveals why user-neutrality is a strong limitation of the existing approaches. This work presents the initial stage of the research.

Keywords
Explainability, enterprise architecture models, complexity management, user modelling

1. Introduction

Explanation could play an important role in building trust in information system's decision. Thus, there is a need for understanding how well a system's decisions are grounded, especially in cases when the derived decisions may significantly affect humans' lives, e.g., in the domains of medicine or law [1].

According to Arrietta et al., explainability should be considered as an interface between humans and the system, which is comprehensible to humans [1]. The authors suggested to place the audience in the centre when explaining the model, and to consider different categories of users. It can be claimed that there is no such thing as a universal explanation, but there is a *need for personalised explanation to a given user.* One of the approaches to information management personalization is user profiling. According to [2] *ontologies can be used for modelling user context.* Enterprise architecture modelling is a domain, where reaching the right level of information granularity is very important for several reasons. Firstly, due to the fact that such models can grow very fast, especially for large enterprises. Secondly, because these models are always used by different types of users, from stakeholders and managers to programmers. One of The Open Group ®[1] standards is the ArchiMate® Specification[2]. The ArchiMate modeling language for enterprise architectures enables to describe, analyze, and visualize the relationships among business domains in an unambiguous way. However, according to [3, p. 59], in ArchiMate, "semantics was explicitly left out". This situation led to some initiatives to use ontologies for enterprise architecture analysis, e.g. [4]. Hence, the problem of providing explainability to enterprise architecture models for different users can be reduced to *personalised complexity management in large models according to the user profile.*

CHItaly 2021 Joint Proceedings of Interactive Experiences and Doctoral Consortium, July 11–13, 2021, Bolzano, Italy
eromanenko@unibz.it (E. Romanenko)
0000-0002-8139-5977 (E. Romanenko)

[1] 1https://www.opengroup.org
[2] https://publications.opengroup.org/c197

The rest of the paper is organized as follows. Section 2 defines a notion of explainability. We describe an enterprise architecture model and personalization in such models in Section 3. The state-of-the-art of complexity management in large models is given in Section 4. Finally, Section 5 concludes the paper.

2. Defining Explainability

In the last few years, the domain of Explainable Artificial Intelligence (XAI) has become a subject of intense study. Large organizations announced their work in that direction. IBM proclaimed explainability together with fairness and robustness as three pillars for building trustworthy AI pipelines [3]. Also the US Defense Advanced Research Projects Agency (DARPA) assigned the highest priority to the XAI project, planning to complete it in 2021 [5]. However, most of the research so far is done in the field of explainable Machine Learning, with the focus on Deep Neural Networks, see, e.g., [6].

There is still some terminology variation on what is meant by *explainability in an information system.* As it was mentioned previously, according to Arrietta et al., it is an interface between humans and the system, while *interpretability* is defined as an ability to explain a phenomenon in understandable terms to a human [1]. Erasmus et al. go even further and define interpretation as “something one does to an explanation with the aim of producing another, more understandable one” [7], i.e., *interpretation is understood as an operator leading to another, more preferable explanation.* The authors claimed, that a complex explanation is an explanation no less, and the main issue is in the user’s capacity to understand such explanation. Another approach is considered in [8], where interpretability should lead to explainability of the system, while the latter is referred to as “the understanding the human user has achieved from the explanation”.

Consequently, *explainability strictly depends on the user and user’s competencies*, thus, the user profile should be taken into account when providing explanations, and this idea is reflected in the literature. Gunning and Aha considered the “user’s mental model” that directly affects user comprehension [5]. Arrieta et al. [1] suggested placing the audience in the centre when explaining the model and distinguished the following categories of users: (1) domain experts, users of the model; (2) regulatory entities/agencies; (3) managers and executive board members; (4) data scientists, developers; (5) users affected by the model’s decisions.

However, there is no common agreement on which groups of users should be selected and while some authors follow the suggested profiles, e.g., in [9], some others lay emphasis on other groups (see [6]). Although these groups could be adapted to different information systems, such approach is still not very convenient, because (i) these groups are preselected and fixed, (ii) the users can have different competencies even within the same group and (iii) some user’s characteristics may change in time [10].

Another approach to adaptivity and personalization support that can be found in the literature is ontology-based user profiling [2], [11]. Instead of extracting user stereotypes, one can develop a user profile ontology, that would be able to exhibit users’ characteristics. *Hence, an explainable information system should provide different users with an explanation of the system results according to the right level of user’s competencies and goals reflected in the user’s ontology.*

[3] https://developer.ibm.com/technologies/artificial-intelligence/articles/the-ai-360-toolkit-ai-models-explained/

3. Importance of Explainability in Enterprise Architecture Models

Enterprise Architecture (EA) models are used by different categories of users and should provide the right level of information granularity for all of them. According to Lankhorst et al. EA is "a coherent whole of principles, methods, and models that are used in the design and realisation of an enterprise's organisational structure, business processes, information systems, and infrastructure" [3, p. 3]. EA models are used by professionals with different background, goals and competencies, namely stakeholders, architects, developers, and sometimes also regulatory agencies representatives, hence, should inherently reflect their views to the given enterprise, *including guiding managers in designing business processes and developers in building applications according to business objectives and policies* [ibid. p. 4].

As it was previously mentioned, the ArchiMate language is an international standard for EA modelling. Despite the facts that (i) the resulting architecture should be coherent by definition and (ii) the number of elements in such models may grow fast for large organizations, the language is missing formal semantics [3, p. 59] and *complexity management in such models is usually done manually by architects with so-called views over the model.* These views are aimed at explaining the content of the model according to the different predefined roles of users.

Imagine a company working on the flights' aggregation service. Assume that on the motivation layer the enterprise architect formulated the following principle: "Search results should contain only relevant information". At the application level this principle may lead to two services: (1) rates comparison service; (2) travel conditions information service. Given two programmers working separately on each of these services, e.g. applying the microservices development approach, who are interested in the information and decisions regarding one service only, the architect would have to manually create two separate views. It could be done by the means of modern modelling tools and proper viewpoints, but this approach is not scalable.

Kang et al. claimed that lack of semantics in EA models is a source of communication problems, because EA components are defined in natural language and could be misunderstood [4]. Since there is a necessity to support enterprise architects in the development process, there are several initiatives for bringing formal semantics into EA models. Gampfer et al. even noticed, that "the focus of EA research has shifted from understanding EA in the early years to managing EA today" [12]. Some of these initiatives are based on providing (meta-)ontologies, e.g., [4], while others attempt to verify the models with methods of formal logic, e.g., [13].

4. Complexity Management Approaches

To the best of our knowledge so far there is no suggested approach to complexity management of EA models that could provide viewpoints according to the user profile. However, *EA models can be considered as a special class of conceptual models.*

For quite some time both ontological complexity management and complexity management in conceptual modelling have been areas of intensive research. Investigations are mostly done on the following tasks:

1. *modularization:* the process of fragmenting a model into several parts;

2. *view extraction:* the process of computing a self-contained portion (closure) of a model that results from a particular traversal of links starting at a central concept or concepts, defined by the user (see, e.g., [14]);
3. *model abstraction or summarization:* the process of producing a reduced version of the original model by omitting details and concentrating on the semantic context [15].

The techniques that are used to solve the above tasks can be roughly classified into *traversal-based* and *logic-based approaches.* Some of the approaches from the first group, e.g. [14], assumes that the engineer has a deep understanding of the given ontology, while others are able to deal with the task in an automatic way [15], yet leading to the deterministic results without any obvious way for adapting them to the user profile.

Logic-based approaches are usually grounded on the notion of *forgetting*. According to Yizheng Zhao [16] forgetting is a non-standard reasoning service that creates views by eliminating concepts and roles from description logic-based ontologies while preserving all logical consequences up to the remaining symbols. Comparing these approaches to the traversal-based approaches one can decide that they are more 'user-centric', since the user decides what exactly she wants to forget, however, such approaches also (i) require an understanding of the ontology and (ii) are deterministic up to the given seeds.

In the above-mentioned example, our architect could be interested not only in the automatic generation of two viewpoint-based models but also in adapting those models to the knowledge level of each programmer, incl. providing additional information about the system if needed.

5. Conclusions

According to Erasmus et al. increasing the complexity of a phenomenon does not make it less explainable [7]. Given a complicated EA model, there is a need for automatic complexity management, but widely used visual language cannot provide it. A "better system" should rather present results tailored to characteristics of each user, rather than being intent on some "typical" person [10]. The research is on the initial stage, however, it is expected, that user profile can be described with the help of the ontology, which should be able to reflect not only long-term user characteristics, such as area of interest or level of expertise, but also relatively short-term ones, e.g., current goal.

Acknowledgments

This paper and the research behind it would not have been possible without the exceptional support and valuable comments of my supervisors, Prof. Diego Calvanese and Prof. Giancarlo Guizzardi.

References

[1] A. B. Arrieta, N. Díaz-Rodríguez, J. D. Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, F. Herrera, Explainable artificial intelli-

gence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible ai, Information Fusion 58 (2020) 82–115. doi:10.1016/j.inffus.2019.12.012.

[2] A. Katifori, M. Golemati, C. Vassilakis, G. Lepouras, C. Halatsis, Creating an ontology for the user profile: Method and applications, in: Proceedings of the First IEEE International Conference on Research Challenges in Information Science (RCIS), 2007, pp. 407–412.

[3] M. Lankhorst, Enterprise Architecture at Work: Modelling, Communication and Analysis, 3rd ed., 2013. doi:10.1007/978-3-642-29651-2.

[4] D. Kang, J. Lee, S. Choi, K. Kim, An ontology-based enterprise architecture, Expert Systems with Applications 37 (2010) 1456–1464. URL: https://www.sciencedirect.com/science/article/pii/S0957417409006368. doi:10.1016/j.eswa.2009.06.073.

[5] D. Gunning, D. Aha, Darpa's explainable artificial intelligence (XAI) program, AI Magazine 40 (2019) 44–58. URL: https://ojs.aaai.org/index.php/aimagazine/article/view/2850. doi:10.1609/aimag.v40i2.2850.

[6] G. Ras, M. van Gerven, P. Haselager, Explanation Methods in Deep Learning: Users, Values, Concerns and Challenges, Springer International Publishing, 2018, pp. 19–36. doi:10.1007/978-3-319-98131-4_2.

[7] A. Erasmus, T. Brunet, E. Fisher, What is interpretability?, Philosophy & Technology Online (2020). doi:10.1007/s13347-020-00435-2.

[8] A. Rosenfeld, A. Richardson, Explainability in human–agent systems, Autonomous agents and multi-agent systems 33 (2019) 673–705. doi:10.1007/s10458-019-09408-y.

[9] A. Heuillet, F. Couthouis, N. Díaz-Rodríguez, Explainability in deep reinforcement learning, Knowledge-Based Systems 214 (2021) 106685. URL: https://www.sciencedirect.com/science/article/pii/S0950705120308145. doi:10.1016/j.knosys.2020.106685.

[10] E. Rich, Users are individuals: individualizing user models, International Journal of Human-Computer Studies 51 (1999) 323–338. URL: https://www.sciencedirect.com/science/article/pii/S1071581981603122. doi:10.1006/ijhc.1981.0312.

[11] A. Sieg, B. Mobasher, R. Burke, Learning ontology-based user profiles: A semantic approach to personalized web search, IEEE Intelligent Informatics Bulletin 8 (2007) 7–18.

[12] F. Gampfer, A. Jürgens, M. Müller, R. Buchkremer, Past, current and future trends in enterprise architecture — a view beyond the horizon, Computers in Industry 100 (2018) 70–84. URL: https://www.sciencedirect.com/science/article/pii/S0166361517306723. doi:10.1016/j.compind.2018.03.006.

[13] P. Szwed, Evaluating Efficiency of ArchiMate Business Processes Verification with NuSMV, volume 243 of *Lecture Notes in Business Information Processing*, 2015, pp. 179–196. doi:10.1007/978-3-319-30528-8_11.

[14] N. F. Noy, M. A. Musen, Traversing Ontologies to Extract Views, 2009, pp. 245–260. doi:10.1007/978-3-642-01907-4_11.

[15] G. Guizzardi, G. Figueiredo, M. M. Hedblom, G. Poels, Ontology-based model abstraction, in: Proceedings of the 13th International Conference on Research Challenges in Information Science (RCIS), 2019, pp. 1–13. doi:10.1109/RCIS.2019.8876971.

[16] Y. Zhao, Automated Semantic Forgetting for Expressive Description Logics, Ph.D. thesis, University of Manchester, 2018.