Towards a Multi-Level Model of Enterprise Architecture Modeling Notations

Ahmed Hussein[1] and Simon Hacks[2][a]
[1]*KTH Royal Institute of Technology, Stockholm, Sweden*
[2]*Stockholm University, Stockholm, Sweden*

Keywords: Enterprise Architecture Modeling, Multi-Level Model, Meta-Meta-Model, Taxonomy.

Abstract: Over the past few decades, the field of enterprise architecture (EA) has grown, and many EA modeling notations have been proposed. In order to support the different needs, the different notations vary in the element types that they provide in their metamodel. This abundance of elements makes it difficult for the end-user to differentiate between the various elements and complicates the model transformation between different EA model notations. Therefore, this research analyzes existing EA frameworks and their modeling notations and extracts common properties. First, we performed a literature review to identify common EA frameworks and their modeling notations. Second, based on the found notations' concepts, we create a taxonomy based on their similarities that leads to a multi-level model of EA notations.
Our results showed that The Open Group Architecture Framework, ArchiMate, Department of Defense Architecture Framework, and Integrated Architecture Framework are the most used EA frameworks. Those frameworks served as input for a multi-level model comprising the common concepts of the different modeling notations.

1 INTRODUCTION

Although digital transformation offers organizations great potential for improvement, it comes with additional challenges such as increasing complexity of IT systems. Other challenges are business-IT alignment and changing environments. Moreover, the stakeholders have different needs that should be addressed by both the business and IT (Hacks et al., 2017). These challenges can be addressed by means of enterprise architecture (EA), which can help to address all stakeholders' concerns (The Open Group, 2019), and the implementation fosters organization's success (Zachman, 1987) and balances IT efficiency and business innovation (Shah and Kourdi, 2007).

The idea behind EA has originated since the 1960s (Kotusev, 2016b). Through the years, the concept of EA has evolved, producing a holistic view to align strategy, operations, and technology of the organization in order to achieve its business goals. Using EA ensures better alignment with business objectives and consistency across the organization, allows recognizing potential improvements (Steen et al., 2004) and provides a general understanding of the enterprise (Franke et al., 2009). Moreover, it improves return on existing investments while minimizing the risk of future investments (The Open Group, 2018).

To assess the current state of the organization and ease the EA's evolution, EA models are defined. An EA model is a representation of the whole's enterprise EA that consists of multiple views and architectural models (The Open Group, 2019; Department of Defense, 2010). The goal of an EA model is to capture the most important components of an enterprise and the relations between them so that the current state can be visualized, reviewed, and future desired states can be planned (Department of Defense, 2010; Hacks and Lichter, 2018a; Hacks and Lichter, 2018b).

In the past decades, many researchers have been attracted to the field of EA, and a plethora of different EA modeling notations has been proposed (The Open Group, 2019; The Open Group, 2018; Department of Defense, 2010; Federal CIO Council, 2013; MOD, 2012; Wout et al., 2010). Unfortunately, one can recognize that notations get more and more extensive, including more and more different element types to satisfy the different needs. For example, ArchiMate (The Open Group, 2019) includes more than 50 different

[a] https://orcid.org/0000-0003-0478-9347

Hussein, A. and Hacks, S.
Towards a Multi-Level Model of Enterprise Architecture Modeling Notations.
DOI: 10.5220/0011706700003467
In *Proceedings of the 25th International Conference on Enterprise Information Systems (ICEIS 2023) - Volume 2*, pages 466-473
ISBN: 978-989-758-648-4; ISSN: 2184-4992

elements in the latest specification. ArchiMate 3.0.1 introduced more than 15 new elements from its predecessor, such as “Events”, “Outcome”, and “Capability” and ArchiMate 3.1 has added ”Value stream”.

One problem with introducing more elements is that it will be difficult to create a concise dictionary of elements definitions. According to ArchiMate, the “Value Stream” is defined as “a sequence of activities that create an overall result for a customer, stakeholder, or end user” (The Open Group, 2019). This definition is very similar to the “Business process” element, which is already included in the previous versions. Despite the similarity in the definition, they are intended to be used differently. The “Value stream” describes the organization’s business model and value proposition, while “Business process” describes the operating model (The Open Group, 2019). This similarity between the different elements will mean that an end user will have to go through the full specifications in order to determine what element to use.

This abundance of elements can make it difficult for the end-user to differentiate between the uses of the different elements to specify what elements match their needs. Moreover, it hinders the transformation of the respective EA models between the different notations. In the context of the Unified Modeling Language (UML), Leroux at al. (Leroux et al., 2006) argue that having a limited number of elements (while allowing for extension) typically makes the notation richer and easier to use. We argue that the same can be applied to EA modeling notations.

Accordingly, we analyze existing EA modeling notations and determine their common properties. These properties include architectural elements, their attributes, and the relationships between them (Franke et al., 2009). Additionally, we present the common elements in layers a multi-level model (Frank, 2014; Atkinson and Kühne, 2001) that is capable of describing the different notations. This yields a better understanding of the concepts behind the different elements and how they interact with each other.

In order to achieve this, we address the following research questions:

- *What are the most commonly used EA modeling frameworks in scientific research?*[1]
- *What are the common properties of EA modeling that are present in the different notations?*

[1]It is criticizable to opt for scientific research here as it does not necessarily reflect the reality in organizations. However, to the best of our knowledge, there are no reliable statistics about the usage of EA frameworks in organizations available, while it is the case for scientific research on EA frameworks (e.g., (Barbosa et al., 2019)).

To answer these questions, we discuss next the different methods that we applied in our research. Afterwards, we present the multi-level model as the result of the first iterations considering different EA modeling frameworks. This is followed by a discussion on the findings and the limitations of our work. Before our conclusion, we shed light on related work.

2 METHOD

In order to answer the research questions, we performed our research in four phases:

- Firstly, we studied the relevant literature about EA, and the provided metamodels of the commonly used EA frameworks. A literature review has been conducted following the guidelines of Kitchenham and Charters (Kitchenham and Charters, 2007). The outcome of this step was a set of EA modeling frameworks in which each of them provides a publicly available metamodel.
- Secondly, the most commonly used EA frameworks were selected, and their metamodels were studied and analyzed in order to come up with a list of the entity types that are supported in the selected modeling frameworks.
- After that, we followed Nickerson et al. (Nickerson et al., 2013) to classify the elements into a taxonomy that we used to determine the multilevel model. The classification was based on the similarities between the different EA frameworks and was performed in multiple iterations to define characteristic group elements.
- Finally, the elements and the taxonomy were reported, presented, and discussed.

2.1 Literature Review

To identify the primary studies, we searched in ACM Digital library, Google Scholar, IEEExplore, ScienceDirect, and Springer Link. The search terms were “*enterprise architecture management*”, “*enterprise architecture framework*”, “*enterprise architecture modeling*”, “*enterprise architecture implementation*”, and “*developing enterprise architecture*”.

Each of the search term produced tens of thousands of results for each of the electronic sources. For each source, the results were sorted by relevance to the search term (an estimation of how much a specific document in the results is related to the search query (Zhang et al., 2018)). After that, the top 20 papers were selected resulting in a total of 500 studies.

Table 1: Most used EA frameworks.

EA framework	Number of studies
TOGAF	40
ArchiMate	37
Zachman	22
DoDAF	15
FEAF	12
IAF	4
Gartner	3
MODAF	3
TEAF	2
GERAM	2

To select only the relevant studies, the following inclusion and exclusion criteria were defined:

- Inclusion criteria:
  1. Studies in English.
  2. Studies with full text available.
  3. Research articles, conference proceedings, and book chapters.
  4. Studies with focus on EA frameworks.
  5. Studies aiming to apply concepts from, address limitations of, or critically evaluate an EA framework.
- Exclusion criteria:
  1. Studies not in English.
  2. Secondary studies, i.e., excluding any SLR.

In addition to the inclusion and exclusion criteria, the following criteria were set to assess the quality of the selected studies.

1. Is the goal of the study stated and explained?
2. Is the context in which the study was performed described?
3. Is the choice of the selected EA framework(s) motivated?
4. Is the direction for future work discussed?
5. Is the validity and reliability of the study discussed?

After applying the inclusion and exclusion criteria and ignoring duplicate studies, 78 studies have been included, conducted between 2004 and 2021. After synthesizing the collected data, we identified TOGAF as the most used EA framework, followed by ArchiMate. Table 1 shows the top 10 used EA frameworks.

Next, each framework was analyzed to identify their EA modeling capabilities, based on their specifications (The Open Group, 2019; The Open Group, 2018; Department of Defense, 2010; Wout et al., 2010; Federal CIO Council, 2013; Bernus et al., 2003; MOD, 2012). The Zachman framework, and Gartner Framework are considered proprietary frameworks (Hadaya et al., 2020) so they were excluded. Additionally, the Treasury Enterprise Architecture Framework (TEAF) was also excluded as we could not find a formal specification.

On the other hand, the Generalised Enterprise Reference Architecture and Methodology (GERAM) provides a set of requirements that other EA frameworks should meet. Thus, the framework is actually considered to be a meta-framework and is intended to aid the development of EA frameworks (Bernus et al., 2003; Bernus et al., 2015). Moreover, many studies have analyzed and mapped existing EA frameworks to GERAM, such as (Noran, 2003a; Noran, 2003b; Noran, 2005; Williams and Li, 1999; Saha, 2004; Chaharsooghi and Ahmadi Achachlouei, 2011).

While FEAF focuses on EA modeling, it does not provide a metamodel. The framework instead provides a set of reference models. Accordingly, we do also neglect this framework.

Following EA modeling frameworks were identified as relevant for our multi-level model of EA modeling notations: (1) TOGAF, (2) ArchiMate, (3) DoDAF, (4) IAF, (5) MODAF.

2.2 Multi-Level Model Development

To the best of our knowledge, there is no concrete approach to develop a multi-level model based on different sources. However, the challenge of developing a multi-level model that relies on common properties is similar to the challenge to develop a taxonomy for a certain domain, which is the process of classifying or categorizing objects into different groups. Accordingly, we want to classify the elements that are available in the selected EA framework metamodels based on the similarities between those frameworks.

There are multiple approaches to generate a taxonomy (Nickerson et al., 2013; Harst et al., 2021; Usman et al., 2017). We followed Nickerson et al. (Nickerson et al., 2013) to generate our multi-level model, as it is a widely used methodology in information systems literature with over 700 citations.

Nickerson et al. (Nickerson et al., 2013) first define a basis for the choosing characteristics, i.e., meta-characteristic and end conditions. In our case, the meta-characteristic is defined and to generate the characteristics of the taxonomy: "Structural elements similarities between the different metamodels" to address our second research question.

Next, one defines ending conditions of the iterations:

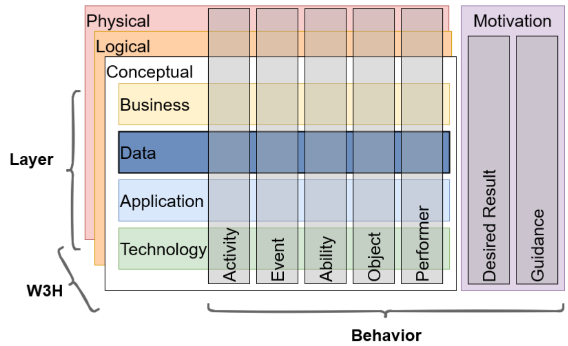

Figure 1: Taxonomy dimensions.

- All elements of the EA framework are examined.
- For all dimensions, each characteristic has at least one element classified under it.
- All dimensions and characteristics are unique.
- In the last iteration, no modification, additions, or deletions has been made to the dimensions nor the characteristics.
- The taxonomy is comprehensive, extensible, and concise.

In total, we performed three iterations. The first iteration included elements from TOGAF and ArchiMate, and each of the following iterations added elements from one additional EA framework (DoDAF and IAF, respectively).

3 A MULTI-LEVEL MODEL OF EA MODELING NOTATIONS

To provide a basis for our multi-level model, we created a taxonomy within three iterations based on the found aspects in the considered frameworks. The taxonomy is comprised of the three dimensions, "Layer", "Behavior", and "W3H" (Contextual (Why), Conceptual (What), Physical (With what), Logical (How)), and respective classifying items (cf. Figure 1).

Table 2 shows the result of the created taxonomy for the common elements in the different frameworks. Each column in the table represents a characteristic of the taxonomy, and the characteristics are grouped by the dimensions mentioned above. An element (row) can be classified in one and only one of the characteristics of each dimension (marked with an x). The final taxonomy result shows that within the common elements of the four EA frameworks, most elements focus on describing the business aspects, activities, and concepts when looking at the "Layer", "Behavior", and "W3H" dimensions respectively.

The taxonomy provides a common abstraction for the different frameworks. After the taxonomy was created, the closest concepts (elements) in the different frameworks were grouped together to find the common elements that are shown in Table 2. This was done by comparing the different definitions of the concepts by one of the researchers and regular discussion rounds on the constructed taxonomy between both of the researchers.

Table 3 provides some examples for mapping the closest concepts in the different EA frameworks, where closeness means the similarity between the concept descriptions determined by the researchers. Even though some concepts are not explicitly available in a framework, some more granular elements fall under it. For example, TOGAF does not explicitly define an abstract "Performer" element, but there are multiple elements available in TOGAF that can be classified as "Performers". On the other hand, DoDAF uses the same term to describe both physical and logical application components, while other frameworks provide separate elements.

The hierarchical structure of the common elements is visualized in a figure[2]. At the top, elements are divided into Behavior, Resource and Motivation elements, i.e., using the Behavior dimension. Then, elements are divided into different layers, Business, Data, Application, and Technology layers.

When creating the multilevel model for the common elements, we used the aforementioned hierarchical structure to assign elements to their corresponding level. Elements at the lowest level are assigned to $M2$ [3] and with each level we increase the model level until we reach $M6$ for the top abstract element. Our figure[4] shows the multilevel model for the generic common elements. On the other hand, we visualize[5] the elements from the different layers.

4 DISCUSSION

The first research question of this study aimed at identifying the most used EA modeling frameworks in the relevant literature. Our findings support the claims of previous studies (Cameron and McMillan, 2013; Mo-

[2] https://github.com/simonhacks/iceis2023/blob/main/Elements\%20Structure.png

[3] $M0$ and $M1$ has not been assigned to any of the elements, since they typically refer to the real-world user objects and the elements of a specific EA model, respectively.

[4] https://github.com/simonhacks/iceis2023/blob/main/multilevel_model_generic.png

[5] https://github.com/simonhacks/iceis2023/blob/main/multilevel_model_layers.png

Table 2: Classification of the common elements.

	Layer					Behavior							W3H			
Element	B	D	A	T	G	Ab	Ac	E	O	P	DR	G	Ct	Cp	L	P
Role	x									x				x		
Actor	x									x						x
Data		x							x					x		
Business Object	x								x					x		
Business Service	x						x							x		
Application Service			x				x							x		
Technology Service				x			x							x		
Business Event	x							x						x		
Goal					x						x		x			
Objective					x						x		x			
Organization Unit	x									x						x
Physical Application Component			x							x						x
Physical Technology Component				x						x						x
Contract	x								x					x		
Business Process	x						x							x		
Logical Business Component	x						x								x	
Logical Application Component			x				x								x	
Logical Technology Component				x			x								x	
Principle					x							x	x			
Requirement					x							x	x			
Constraint					x							x	x			
Course of Action	x						x							x		
Capability	x					x								x		
Project					x		x							x		

Layer: [B] Business, [D] Data, [A] Application, [T] Technology, [G] Generic.
Behavior: [Ab] Ability, [Ac] Activity, [E] Event, [O] Object, [P] Performer, [DR] Desired Result, [G] Guidance.
W3H: [Ct] Contextual, [Cp] Conceptual, [L] Logical, [P] Physical.

hamed et al., 2012; Kotusev, 2016a; Barbosa et al., 2019) that TOGAF is the most used EA framework, often in combination with ArchiMate.

The goal of the second research question was to identify common architectural elements in EA modeling frameworks. The taxonomy dimensions cube was inspired by the ArchiMate layers as well as the IAF content framework abstraction levels. The "Layer" dimension is influenced by TOGAF, IAF, and Archimate, since DoDAF does not provide a classification for the elements into different layers. Similarly, the "Behavior" dimension is influenced by the aspect areas of ArchiMate and the elements from DoDAF. Our dimension has more granular characteristics than the ArchiMate aspect areas, mainly due to the high level abstract elements available in DoDAF; such as "Activity" and "Desired Result" which group multiple concepts together. The last dimension is what we call "W3H" which is derived from how IAF, and to some extent TOGAF, to group elements into conceptual, logical, physical, and contextual elements.

The four EA frameworks share a total of 24 elements in addition to 13 abstract elements. TOGAF has the lowest number of elements defined in its metamodel compared to the other EA frameworks. TOGAF content metamodel only defines a total of 38 elements, while ArchiMate defines 59 elements, IAF and DoDAF has more than 80 elements defined. Additionally, the business architecture has more elements in common between the different frameworks. This can be traced back to TOGAF and IAF; since DoDAF does not provide an elements' layers taxonomy, and ArchiMate has a relatively close number of elements in business, application, and technology layers

Moreover, we noticed the ambiguity of terms in EA modeling as different terms in different EA frame-

Table 3: An Extract from mapping the closest concepts in TOGAF, ArchiMate, DoDAF and IAF.

Concept	**TOGAF**	**ArchiMate**	**DoDAF**	**IAF**
Business Service	Business Service	Business Service	Service	Business Service
Business Process	Process	Business Process	Process	Business Service
Logical Application Component	Logical Application Component	Application Function	System	Logical IS Component
Physical Application Component	Physical Application Component	Application Component	System	Physical IS Component
Performer	***	Active Structure	Performer	***
Business Object	***	Business Object	Information	Business Object
Driver	Driver	Driver	N/A	Business Driver

[N/A] Concept not available in the EA framework.
[***] Concept not explicitly available, but multiple elements fall under it.

works might refer to the same concept (cf. Table 3). For example, IAF uses the term "Business Service" to describe both internal processes and externally provided services. According to IAF, the business service element can be used on different levels of details depending on the goal of the architecture (Wout et al., 2010). This adds to the findings of previous studies that showed the deviations in both the definition of EA (Saint-Louis et al., 2017) and what is considered an EA framework (Franke et al., 2009).

Our work has some limitations. First, we focused only on analyzing the common elements used in EA modeling. Even though we tried to describe these elements in a multilevel model, the relationships used in the model were not extracted using the same method we used to extract the elements due to complexity. However, in a next iteration, we plan to address this.

Additionally, different other frameworks such as GERAM (Generalised Enterprise Reference Architecture and Methodology) or MODAF were not included when developing the taxonomy and analyzing the common elements, due to time limitations. Including the additional elements might result in the generation of more dimensions in the taxonomy or identifying more abstract elements. Moreover, to ensure that no additional dimensions are added to the taxonomy in the last iteration, more iterations were needed, i.e., more frameworks and more time.

Finally, it would been beneficial to include the input from practitioners. That would help to define the common elements from the end users' points of view.

5 RELATED WORK

Before concluding our work, we focus on related literature about EA frameworks and, specifically on previous literature reviews on EA and EA frameworks. Finally, we discuss research on metamodeling in EA.

Hadaya et al. (Hadaya et al., 2020) proposed a methodology to evaluate EA frameworks based on 14 criteria. Their goal was to make it easier for EA practitioners to select the framework that addresses their needs. They first performed a systematic literature review (SLR) on EA framework evaluation criteria. They concluded that previous EA literature does not define comprehensive evaluation criteria. Therefore, they developed their own evaluation criteria containing features like architecture layers taxonomy, architecture taxonomy aspect areas, metamodel completeness and complexity, or development process. Next, they tested it with a set of six EA frameworks and different domain experts, who perceived most of the proposed criteria to be usable, relevant, and correct.

Zhou et al. (Zhou et al., 2020) performed a SLR on EA visualization methodologies. The results show that there is no comprehensive methodology for EA modeling. Thus, they proposed a method for EA visualization that aims to address the shortcomings of the previous methods. Moreover, Zhou et al. (Zhou et al., 2020) found that the most used EA frameworks were TOGAF and DoDAF with more than 60 respectively 20 studies using them. Additionally, they reported that ArchiMate was the most used modeling language, with more than 40 papers using it.

Saint-Louis et al. (Saint-Louis et al., 2017) investigated the deviations in EA definitions. They examined 145 definitions and concluded that there are considerable differences in the definitions. Frank et al. (Franke et al., 2009) suggested a meta framework for EA frameworks to examine if the content of a specific EA framework satisfies practitioners' needs. They divided the content of EA frameworks into "architectural governance" and "EA modeling concepts". As part of the study, the content of multiple frameworks has been examined and classified. Comparably, Cameron and McMillan (Cameron and McMillan, 2013) investigated the usage and characteristics of various EA frameworks to provide a method for selecting the right EA for an organization. Another study that aims to help practitioners select the EA framework that satisfies a specific criterion, is the study performed by Urbaczewski and Mrdalj (Urbaczewski and Mrdalj, 2006), which compared five frameworks. The study provides guidelines for comparing EA frameworks based on the viewpoints and the aspects they support.

In the field of EA, modeling is considered to be a focus of many EA frameworks (Franke et al., 2009). Therefore, EA frameworks usually define a metamodel in order to define a structure for the architectural artifacts and ensure consistency when using them to develop an EA (The Open Group, 2018; Department of Defense, 2010; Wout et al., 2010). Additionally, it allows architecture tools vendors to create easier to use tools by adding consistency and interoperability to the tools they create (The Open Group, 2011). The metamodel of an EA framework is defined by defining the allowed content in an architectural model. The content consists of the entity types (architectural elements), element attributes and the relationships between the different elements (Franke et al., 2009; The Open Group, 2019; The Open Group, 2018; Department of Defense, 2010; Wout et al., 2010). Some frameworks even define their graphical representation of it content (The Open Group, 2019; Atkinson and Kühne, 2020), while others only define the allowed entities and the relations textually (The Open Group, 2018; Department of Defense, 2010; Wout et al., 2010).

Atkinson and Kühne (Atkinson and Kühne, 2020) suggested improving the ArchiMate modeling standard through the use of multi-level modeling. The study used deep modeling to describe ArchiMate and then discusses the potential improvements using the concepts of deep modeling in EA. They claimed that deep modeling can simplify the usage of the ArchiMate language and improve its expressiveness.

6 CONCLUSION

In this work, we performed an analysis of the most used EA frameworks (TOGAF, ArchiMate, DoDAF, and IAF) in the EA literature and extracted the core elements of EA modeling. Then, a taxonomy of elements was created based on the frameworks' elements. The created taxonomy helped in identifying what concepts are available in the different EA frameworks. Next, we identified the common elements that are available in the different EA frameworks. Finally, the common elements are presented in both hierarchical structure and in a multi-level model that shows how the different elements can interact with each other.

We believe that the findings will be interesting for EA researchers and practitioners working on developing and maintaining EA modeling frameworks. Existing EA modeling frameworks might use the list of common elements identified in this study as their core elements, while using the rest of the elements just on demand. Additionally, practitioners can use the taxonomy to identify what element types are provided by the different EA modeling frameworks, thus increasing their ability to identify matching EA modeling frameworks.

For future work, we plan to demonstrate our work on real world cases in which different EA notations are used and need to be mapped to each other. A possible case could be the analysis of EA quality attributes and to abstract the definition of respective queries while still keeping their semantics (Smajevic et al., 2021).

REFERENCES

Atkinson, C. and Kühne, T. (2001). The essence of multi-level metamodeling. In *Proceedings of the 4th International Conference on The Unified Modeling Language, Modeling Languages, Concepts, and Tools*, pages 19–33, London, UK. Springer.

Atkinson, C. and Kühne, T. (2020). A deep perspective on the archimate enterprise architecture modeling language. *Enterp. Model. Inf. Syst. Archit. Int. J. Concept. Model.*, 15:2:1–2:25.

Barbosa, A., Santana, A., Hacks, S., and Stein, N. v. (2019). A taxonomy for enterprise architecture analysis research. In *21st International Conference on Enterprise Information Systems*, volume 2, pages 493–504. SciTePress.

Bernus, P., Nemes, L., and Schmidt, G., editors (2003). *GERAM: The Generalised Enterprise Reference Architecture and Methodology*, pages 21–63. Springer Berlin Heidelberg, Berlin, Heidelberg.

Bernus, P., Noran, O., and Molina, A. (2015). Enterprise architecture: Twenty years of the geram framework. *Annual Reviews in Control*, 39:83–93.

Cameron, B. H. and McMillan, E. (2013). Analyzing the current trends in enterprise architecture frameworks. *Journal of Enterprise Architecture*, 9(1):60–71.

Chaharsooghi, K. and Ahmadi Achachlouei, M. (2011). Developing life-cycle phases for the dodaf using iso15704 annex a (geram). *Computers in Industry*, 62(3):253–259.

Department of Defense (2010). DoD Architecure Framework version 2.02.

Federal CIO Council (2013). Federal enterprise architecture framework v2.

Frank, U. (2014). Multilevel modeling - toward a new paradigm of conceptual modeling and information systems design. *BISE*, 6(6):319–337.

Franke, U., Hook, D., Konig, J., Lagerstrom, R., Narman, P., Ullberg, J., Gustafsson, P., and Ekstedt, M. (2009). Eaf2- a framework for categorizing enterprise architecture frameworks. In *2009 10th ACIS International Conference on Software Engineering, Artificial Intelligences, Networking and Parallel/Distributed Computing*, pages 327–332.

Hacks, S., Brosius, M., and Aier, S. (2017). A case study of stakeholder concerns on eam. In *2017 IEEE 21st International Enterprise Distributed Object Computing Workshop (EDOCW)*, pages 50–56. IEEE.

Hacks, S. and Lichter, H. (2018a). A probabilistic enterprise architecture model evolution. In *2018 IEEE 22nd International Enterprise Distributed Object Computing Conference (EDOC)*, pages 51–57. IEEE.

Hacks, S. and Lichter, H. (2018b). Towards an enterprise architecture model evolution. In *Workshops der INFORMATIK 2018-Architekturen, Prozesse, Sicherheit und Nachhaltigkeit*. Köllen Druck+ Verlag GmbH.

Hadaya, P., Leshob, A., Marchildon, P., and Matyas-Balassy, I. (2020). Enterprise architecture framework evaluation criteria: a literature review and artifact development. *Service Oriented Computing and Applications*, 14(3):203–222.

Harst, L., Otto, L., Timpel, P., Richter, P., Lantzsch, H., Wollschlaeger, B., Winkler, K., and Schlieter, H. (2021). An empirically sound telemedicine taxonomy – applying the cafe methodology. *Journal of Public Health*.

Kitchenham, B. and Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering.

Kotusev, S. (2016a). Enterprise architecture is not togaf. *British Computer Society (BCS)*.

Kotusev, S. (2016b). The history of enterprise architecture: An evidence-based review. *Journal of Enterprise Architecture*, 12:29–37.

Leroux, D., Nally, M., and Hussey, K. (2006). Rational software architect: A tool for domain-specific modeling. *IBM Systems Journal*, 45(3):555–568.

MOD (2012). Mod architectural framework.

Mohamed, M. A., Galal-Edeen, G. H., Hassan, H. A., and Hasanien, E. E. (2012). An evaluation of enterprise architecture frameworks for e-government. In *2012 Seventh International Conference on Computer Engineering Systems (ICCES)*, pages 255–260.

Nickerson, R. C., Varshney, U., and Muntermann, J. (2013). A method for taxonomy development and its application in information systems. *European Journal of Information Systems*, 22(3):336–359.

Noran, O. (2003a). An analysis of the zachman framework for enterprise architecture from the geram perspective. *Annual Reviews in Control*, 27(2):163–183.

Noran, O. (2003b). *A Mapping of Individual Architecture Frameworks (GRAI, PERA, C4ISR, CIMOSA, ZACHMAN, ARIS) onto GERAM*, pages 65–210. Springer Berlin Heidelberg, Berlin, Heidelberg.

Noran, O. (2005). A systematic evaluation of the c4isr af using iso15704 annex a (geram). *Computers in Industry*, 56(5):407–427.

Saha, P. (2004). Analyzing the open group architecture framework from the geram perspective.

Saint-Louis, P., Morency, M. C., and Lapalme, J. (2017). Defining enterprise architecture: A systematic literature review. In *2017 IEEE 21st International Enterprise Distributed Object Computing Workshop (EDOCW)*, pages 41–49.

Shah, H. and Kourdi, M. (2007). Frameworks for enterprise architecture. *IT Professional*, 9:36 – 41.

Smajevic, M., Hacks, S., and Bork, D. (2021). Using knowledge graphs to detect enterprise architecture smells. In *IFIP Working Conference on The Practice of Enterprise Modeling*, pages 48–63. Springer.

Steen, M., Akehurst, D., ter Doest, H., and Lankhorst, M. (2004). Supporting viewpoint-oriented enterprise architecture. In *Proceedings. Eighth IEEE International Enterprise Distributed Object Computing Conference, 2004. EDOC 2004.*, pages 201–211.

The Open Group (2011). *TOGAF® Version 9.1*. TOGAF Series. Van Haren Publishing.

The Open Group (2018). *The TOGAF® Standard, Version 9.2*. TOGAF Series. Van Haren Publishing.

The Open Group (2019). *Archimate ® 3.1 Specification*. The Open Group series. Van Haren Publishing.

Urbaczewski, L. and Mrdalj, S. (2006). A comparison of enterprise architecture frameworks. *Issues in information systems*, 7(2):18–23.

Usman, M., Britto, R., Börstler, J., and Mendes, E. (2017). Taxonomies in software engineering: A systematic mapping study and a revised taxonomy development method. *Information and Software Technology*, 85:43–59.

Williams, T. J. and Li, H. (1999). *PERA and GERAM—enterprise reference architectures in enterprise integration*, pages 3–30. Springer US, Boston, MA.

Wout, J. v., Hartman, H., Hofman, A., Stahlecker, M., and Waage, M. (2010). *The Integrated Architecture Framework Explained: Why, What, How*. Springer-Verlag, Berlin, Heidelberg, 2. aufl. edition.

Zachman, J. A. (1987). A framework for information systems architecture. *IBM Systems Journal*, 26(3):276–292.

Zhang, J., Liu, Y., Ma, S., and Tian, Q. (2018). Relevance estimation with multiple information sources on search engine result pages. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM '18, page 627–636, New York, NY, USA. Association for Computing Machinery.

Zhou, Z., Zhi, Q., Morisaki, S., and Yamamoto, S. (2020). A systematic literature review on enterprise architecture visualization methodologies. *IEEE Access*, 8:96404–96427.