Information Systems Students' Impressions on Learning Modeling Enterprise Architectures

1st Ville Seppänen
Faculty of Information Technology
University of Jyvaskyla
Jyvaskyla, Finland
ville.r.seppanen@jyu.fi

2nd Mirja Pulkkinen
Faculty of Information Technology
University of Jyvaskyla
Jyvaskyla, Finland
mirja.k.pulkkinen@jyu.fi

3rd Toni Taipalus
Faculty of Information Technology
University of Jyvaskyla
Jyvaskyla, Finland
toni.taipalus@jyu.fi

4th Jarkko Nurmi
Faculty of Information Technology
University of Jyvaskyla
Jyvaskyla, Finland
jarkko.s.nurmi@student.jyu.fi

***Abstract*—This Full Research Paper presents enterprise architecture (EA) modeling tools utilized in an educational context. EA is a well–known and a commonly used approach for organizational development aiming to improve the alignment of business operations and information technology. This high level design of information technology (IT) driven business operations lays the foundations on lower level technical activities such as the design and implementation of application programs and features, system boundary interfaces, database distribution and data pipes, and system recovery. Organizations' architectures are made visible by creating EA artefacts, such as business process diagrams, data models and development roadmaps for the betterment of a holistic understanding and future planning of organizational IT solutions. It follows that IT students as future IT professionals need to understand the high level organizational IT landscape in order to understand, for example, software interface design, feature prioritization, and the evaluation of suitable technologies. Although EA is one of the core competency areas of the academic information systems graduate curriculum, the means of teaching EA are seldom discussed, and studies specifically focusing on modeling EA are lacking. In this paper, we report our experiences on teaching a practical course on EA, and our findings based on data collected from students who took the course. By discussing our findings in relation to a widely acknowledged competency model for graduate degree programs in information systems as well as prior research, we conclude that it is possible to effectively teach the modeling of some of the most essential EA artefacts with different tools. Perhaps most importantly, our findings show that modeling tools that are strict in EA standard conformance are perceived easier to learn and use by students, than merely illustrative tools with lenient or nonexistent conformance checks.**

***Index Terms*—computing, education, enterprise architecture, modeling, information system**

I. Introduction

Designing and managing enterprise architecture (EA) is one of the seven high-level competency areas of IS 2010 Curriculum Guidelines for Undergraduate Degree Programs in Information Systems (IS 2010) and EA is one of the seven core courses recommended to be common to all Information Systems programs [1], including the MSc in IS [2]. Students should develop expertise in high level design and management of IT capabilities and be able to plan the alignment of these capabilities with general organizational goals [1]. EA "focuses on organizational level issues related to planning, architecting, designing, and implementing IT-based solutions" [1], thus providing an integrative view to themes of other information systems core courses, which include, for example, IT Infrastructure, Systems Analysis and Design, and Data and Information Management.

EA offers a holistic approach for managing different dimensions of an organization, such as its strategic goals, business activities, software systems, databases, and technology infrastructures. By making these components and their relationships visible by modeling, EA can support the co-operation of different stakeholder groups, and foster the use of common language in the activities of planning, decision-making, and development [3]. EA is widely used in both public and private organizations for strategy formation and aligning the business capabilities with the supporting IT resources [4].

This paper reports teaching experiences from a practical EA course, which we ran for second year students of the three year Bachelor Program in IS. The course aims to meet the requirements for Enterprise Architecture competency area in the IS 2010 curriculum. One of the focal practice related learning objectives of the course is to introduce students to the creation of EA models (EA artefacts) with EA modeling tools. Some previous studies have discussed the use of modeling tools designed specifically to support the learning of modeling languages, often Unified Modeling Language (UML), but we are not aware of such learning tools available for EA modeling. Thus, the primary objective of this research is to find out *how students perceive, as a part of their learning experience, the use of three different EA modeling tools, which are mainly aimed for professional use*.

Learning EA related competencies should equip students with skills they need as, e.g., system architects or junior consultants, who analyze a client's business, and prepare models of the current and the target state of the architectural dimensions in, e.g., EA development projects. Sometimes the modeling tool to be used in a project is defined by the client organization, thus there should be little threshold for switching between tools. Further learning outcomes in the course are knowledge of the mainstream modeling languages for EA modeling, (e.g., ArchiMate and Business Process Model and Notation, BPMN), as well as common principles for modeling, deepening prior knowledge in basic systems development, and gaining understanding and practical skills on how a modeling tool with a model database works. Secondary goal of our research is, therefore, to find out *to what extent practical skills in EA modeling can be developed in a given setting*. The main driver behind the objectives is curriculum development [5], [6] with specific focus on tools used; our results provide implications that more professional tools — although the learning curve is steeper — equip students with skills to understand the EA process, not just the modeling. This, in turn facilitates lifelong learning and meta-skill development, rather than superficial learning of different tools.

The rest of this paper is structured as follows. The next Section discusses the theoretical background of our study covering three themes. First, the general theme of modeling EA is addressed. Second, the industry standard EA modeling language ArchiMate is be discussed. Third, prior research on the subject of teaching modeling and systems design is reviewed. Section III introduces the research setting of this study and the course from which the data were collected for this research. The three modeling tools used in the course, namely Microsoft Visio Professional, Archi, and Arter Arc are briefly introduced, before further clarifying the process of data collection. Section IV presents the results of our study, which are then discussed in Section V along with practical recommendations. Finally, Section VI concludes the paper.

II. Theoretical Background

A. Modeling Enterprise Architecture

Modeling structural and dynamic components of an organization is a key prerequisite of EA management. The models created to describe these structures and their relationships are referred to as EA artefacts [7]. The EA artefacts describe how the different components of an organization are related to each other and work as an interconnected whole. The EA artefacts can be created for the purposes of the operational management by detailing an organization's current structural elements or for the strategic change management by representing possible future-state designs and the transition plan between the current and target states. The EA artefacts support the dialogue between the different stakeholders by proving a common language for the planning and development activities. Modeling an EA is usually a laborious and time-consuming task, and needs acumen regarding both organizations' business functions and IT, as well as expertise on modeling complex and intertwined systems [8].

Niemi and Pekkola [9] have studied the use of EA artefacts. They identified different use cases for the artefacts, that include, for example, defining and planning solutions, designing and implementing solutions, executing solution acquisitions, planning solution updates, supporting development projects, and supporting strategic planning. Based on a large dataset collected from practitioners, Bischoff et al. [10] categorize EA artefacts into the four classes according to how intensively they are used in practice and to what degree the external pressure from stakeholders increases the use intensity: EA superstars, EA pressure beneficiaries, EA shelf-warmers, and EA annoyances. We utilized their results in the research design of this study as explained in Section III.

B. ArchiMate Modeling Language

A majority of organizations practicing EA use some method or a framework to steer the process and the content of their EA work. According to a survey [11], the most commonly used framework in industry is The Open Group Architecture Framework (TOGAF). ArchiMate modeling language, also an Open Group standard, provides a TOGAF compliant modeling notation for all the EA domains. ArchiMate is widely used in both public and private organizations around the world, and it is supported by a majority of EA modeling tools. As a semi–formal modeling language, ArchiMate provides a coherent and visually uniform representation for the EA artefacts covering different components of an organization and their interdependencies. As such it aims at enabling communication among stakeholders and guides the complicated change processes of architectural structures [12]. The architecture viewpoints (i.e., layers) since ArchiMate 3 have been Strategy, Business, Application, Technology, Physical, Motivation, and Implementation & Migration. The architecture elements belonging to the layers are divided into three structural aspects, namely Active, Behavior, and Passive. The elements of Active Structure layer (e.g., a Business Actor and an Application Component) are entities that perform some behavior. The Behavior elements (e.g., a Business Process and an Application Function) are units of activity that is performed by the Active Structure elements. A behavior is performed upon Passive Structure elements (e.g., a Business Object and a Data Object). ArchiMate includes 3 modeling elements in the Strategy layer, 13 modeling elements in the Business layer, 9 in the Application layer, 13 in the Technology layer, 4 in the Physical layer, 10 in the Motivation layer, and 5 in the Implementation & Migration layer. In addition, there are generic elements to represent grouping and location of other elements. The 11 relationship types between elements can be categorized into four sets. Structural relationships represent static construction or composition between elements. Dependency relationships describe how elements support other elements and Dynamic relationships are used to model behavioral dependencies between elements. There are also Other relationships for representing the cases of specialization and association.

C. Teaching Modeling and Systems Design

Extant research on teaching systems modeling is somewhat limited, and especially studies on EA modeling education are lacking. As noted by Tambouris et al. [13], the EA implementation is a complex multidisciplinary process, and as such it requires highly skilled personnel, i.e., enterprise architects with diverse competencies. Tambouris et al. [13] proposed the EA Competence Framework to support the development of EA education programs and materials. Among the other capabilities required from enterprise architects, such as business acumen, technological knowledge and people skills [14], they also emphasize the expertise on modeling techniques and business modeling. However, according to Seppänen et al. [8], one of the key factors still limiting successful EA implementations is the lack of practical EA modeling skills.

UML is widely used in the Computer Science and Information Systems curricula for the activities of systems design and modeling, and the majority of previous modeling education research focuses on UML teaching in universities. As the ArchiMate language has derived a number of concepts from UML, and straightforward analogies can be defined also for other concepts, research on UML education carries some relevance for this study as well. Siau and Loo [15] studied problems encountered by university students in learning and applying UML and identified two problem categories. The first category was traced to the inherent problems in UML. This category includes problems related to ambiguity and inconsistency of UML diagrams, semantics of diagrams, and excessive number of constructs making notations ambiguous and difficult to remember. Problems in the second category, peripheral issues in learning UML, relate to prior knowledge of learners and the lack of prior knowledge in structural analysis and design, lack of good study materials, as well as problems such as user-unfriendliness of CASE (computer-aided software engineering) tools. While a number of studies over the decades [16]–[20] have shown that the use of modeling tools can facilitate teaching of modeling language concepts, it has also been argued that these tools often are confusing to beginners, and difficult to learn as their focus lies on professional development instead of educational aspects [21].

On the other hand, studies [21], [22] have indicated some promise regarding the use of modeling tools specifically designed for the educational purposes but the evidence of their positive effect on the learning outcomes is currently limited, and studies on teaching EA are remarkably few and far between. Teaching requirements elicitation in relation to the EA has been discussed [23], [24], and Steenkamp et al. [25] reported their experiences on giving an EA specification course that was based on a real-world business case. Prior studies specifically focusing on the EA modeling cannot be currently found, but arguably curriculum development benefits from the use of appropriate tools.

III. Research Setting

A. The Course

The research setting was built around the course *Enterprise Architecture in Practice*, which aims at offering students a practically oriented introduction to the concept of EA and its applications in the design and development of organizations' business and IT. The majority of approximately 50 students who enrolled in the course during the Spring semester 2019 were completing their second year of the Information Systems Bachelor Program. This effectively means that the students had completed basic system development courses, and had knowledge and skills in system design (e.g., designing UML class and sequence diagrams).

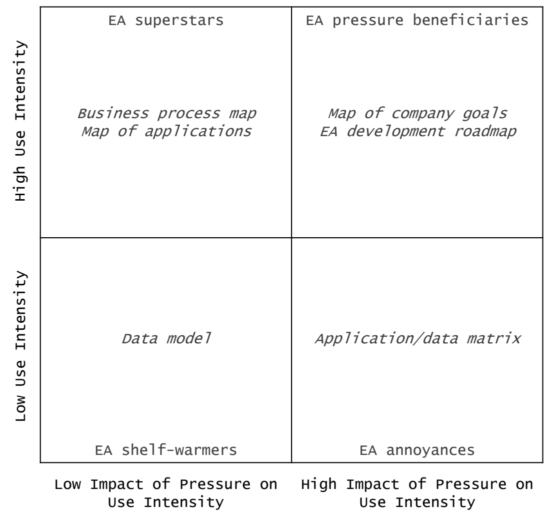

Fig. 1. Modeling exercises classified according to Bischoff et al. [10]

The course consisted of lectures and guided modeling exercises in computer classrooms. To complete the course, the students were required to pass an exam, for which they prepared with lectures and some reading materials (e.g., the course book [26]), and individually conducted a practical work on case-based EA modeling. Prior to starting the modeling exercises in a classroom, the students followed ten hours of lectures, first to introduce the EA basic concepts, the significance of EA to the IT management and IS planning. Also, TOGAF framework and the Architecture Development Method (ADM) were discussed. Next, six hours of modeling-focused lectures were given. These covered ArchiMate 3.0.1 elements and relationship types in detail, and provided examples of different EA artefacts for different architecture layers, and a two-hour lecture on business process modeling using BPMN 2.0. The modeling lectures were based on the lecturers' experiences on EA modeling, the Open Group's documentation on the ArchiMate standard, and the book *Enterprise Architecture at Work -– Modeling, Communication and Analysis* [26], which was also used as the textbook in the course.

During the classroom exercises, the students were asked to create six different EA artefacts with three modeling tools discussed in the next section. The artefacts included 1) Business process map, 2) Map of applications, 3) Map of company goals, 4) EA Development roadmap, 5) Data model, and 6) Application/data matrix (Fig. 1).

These artefacts were chosen for the modeling exercises because together they provide an encompassing view of different EA artefacts, and could accentuate the modeling tools' different approaches, capabilities and limitations. In their study on EA artefact use intensity among enterprise architects, and the pressure from the stakeholders affecting the use intensity, Bischoff et al. [10] classified artefacts 1) and 2) as EA superstars with high use intensity with low impact of pressure on use intensity. Artefacts 3) and 4) were classified as EA pressure beneficiaries with high use intensity and high impact of pressure on use intensity. Artefact 5) presents EA shelf-warmers, which have low use intensity and low impact of pressure on use intensity. Finally, artefact 6) was classified by Bischoff et al. [10] as an EA annoyance with low use intensity but high impact of pressure on use intensity.

B. The Modeling Tools

The three modeling tools the students used in the course were Microsoft Visio Professional, an open source ArchiMate modeling tool Archi, and a business process and EA modeling tool Arter Arc. These tools were chosen for the course because they offer three drastically different approaches to modeling. According to the State of Enterprise Architecture 2018 survey [27], Microsoft Visio is the second most commonly used tool among the survey respondents while Archi shared the third place. Arter Arc was chosen because of its easily customizable metamodel-founded approach and secondly, it is a Software-as-a-Service implementation, which is an increasing trend also for EA modeling tools.

Microsoft Visio is a general-purpose diagramming tool that practically allows users to model effectively anything. It emphasizes the visual representation of the models and supports a number of different modeling notations, such as UML, BPMN, and ArchiMate, which are available as stencil packs. The stencils merely provide the graphical notation to be used in the models. Microsoft Visio does not require the models to comply with the syntax of the modeling languages that the stencils represent, nor can it validate the model's compliance to the standard. Microsoft Visio's functionality can be extended by linking diagrams with relational data stored and maintained in a relational database or an Excel worksheet. Of the three tools discussed in this paper, Microsoft Visio is the most permissive one as it sets no limitations regarding the modeling. As such, it does not provide the user with any guidelines or language-related help. However, Microsoft Visio "has found favor with many architects due to its broad availability and familiarity" [27].

Archi is an open source modeling tool specifically aimed for the ArchiMate modeling. It provides a comprehensive built-in ArchiMate documentation, model validation functionality, and tools such as Magic Connector which help users in creating models that comply to the ArchiMate standard. Archi has a large userbase and a relatively active user community contributing feedback and ideas for the developers.

In terms of modeling capabilities, Arter Arc can be regarded as the most advanced tool of the three. Its approach to modeling is quite different from the more visually-oriented tools such as Microsoft Visio. The EA components are first created by entering their characterizing information textually,

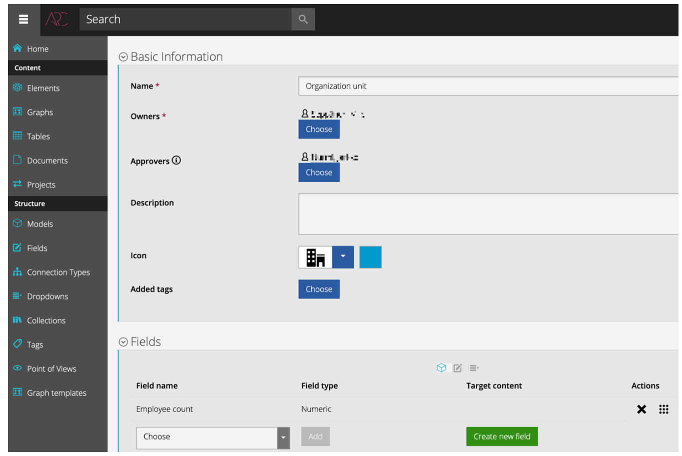

Fig. 2. Defining a conceptual architecture component with Arter Arc

and by defining their relationships with other components (Fig. 2). Then, if deemed useful for the purpose, users can draw a visual model to which these components are linked. Arter Arc also allows users to easily modify and extend the underlying metamodel to which the models adhere.

We summarize the characteristics of these three modeling tools with regard to the three aspects (Table I). First, *Compliance to ArchiMate Specification* denotes that how compliant the tool is in regard to the syntax of the modeling language. Each tool supports the use of ArchiMate's graphical presentation. However, Microsoft Visio does not enforce any requirements towards its correct use. Neither does Arter Arc, which practically allows users to create any kind of graphical illustrations without necessitating that the modeling language's elements and their relationships adhere to the standard. However, with Arter Arc, it is possible to define an underlying metamodel, which then, for example, limits the use of relationship types to only those that are allowed to connect certain types of modeling elements. Archi, on the other hand, is built on the ArchiMate metamodel and therefore allows a user to only create models that adhere to the standard (although there are some errors in the current implementation). Second, *Assisting Features* refers to the features that assist users in creating correct and expressive models, and also help users in creating syntactically correct and semantically meaningful models. Microsoft Visio provides no help in relation to modeling language specific aspects, but it has a plenty of features that support creating visually expressive diagrams. Archi, on the other hand, provides a wide array of assisting features for creating both syntactically correct and easily interpretable models. It features a comprehensive built-in documentation for each ArchiMate element and relationship type, the Magic Wand tool automatically suggests relationship types allowed between the certain element types, and the tool is able to validate the model and notifies the user about the errors (e.g., syntax errors) and the possible problems (e.g., duplicate elements) (Fig. 3). Arter Arc provides a user with no assistance for the language-specific diagramming but if

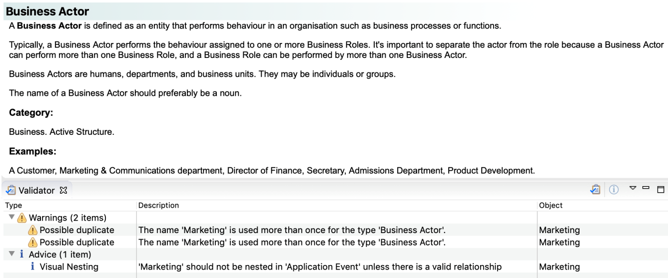

Fig. 3. Built-in ArchiMate documentation and model validator output in Archi

TABLE I
CHARACTERISTICS OF THE MODELING TOOLS

	Compliance to ArchiMate specification	Assisting features	Modeling freedom
Microsoft Visio	Low	Moderate	High
Archi	High	High	Low
Arter Arc	Moderate	Moderate	Moderate

a user wants to create models that adhere to a predefined metamodel, the tool offers a number of helpful features. For example, Arter Arc suggests suitable targets to connect the architecture components using the relationships types defined in a metamodel.

Third, *Modeling Freedom* refers to the possibility to create any kind of model whether or not the model would adhere to the modeling language specification. While adherence to the modeling standards is in many cases arguably an important requirement, there are also use cases for EA artefacts where it might be beneficial to use free-formed models or diagrams that are more easily approachable for non-EA-savvy audiences. In this regard, Microsoft Visio scores a high mark as a generic purpose diagramming tool. Archi, on the other hand, only allows creating ArchiMate-standard compliant models. Arter Arc, depending on how it is used, allows both free-formed diagramming and metamodel-guided modeling approaches.

For each of the modeling tools used in the course, the students were given short instructional video tutorials. The length of the tutorials for Microsoft Visio, Archi and Arter Arc were 8, 11 and 27 minutes, respectively. The videos explained the basic workflow and logic of the tools by showing how to create a simple hierarchical representation of an organization structure (i.e., an organization chart). This example utilized the ArchiMate Business Actor element (e.g., an organization unit), the composition relationship (e.g., a marketing division has sub-departments) and the attributes to detail the elements (e.g., department's employee count; cf. Fig. 2). The varying lengths of the video tutorials do not imply that a longer video would have been more thorough or detailed, but rather reflect how quick it was to create the example model, and to explain the required steps while doing so.

C. Research Data

The research data was obtained from 25 students who volunteered to answer the survey questionnaire regarding their experiences on creating the six different EA artefacts by using the three different modeling tools. The students were asked to fill in a questionnaire soon after they had finished practicing with each tool. Each student used the tools in randomly selected order as we considered possible that the accumulating experience on creating the EA artefacts would affect the impressions on the tools used later during the exercises. The questionnaire contained 18 five-level Likert-items on the scale from 1 (I fully disagree) to 5 (I fully agree). The items are presented in detail in the next section. In addition, the students were asked to give each tool an overall grade from 1 (the worst) to 5 (the best) and were able to write free comments. The students were allowed to answer the questionnaire anonymously.

IV. RESULTS

Table IV presents the means and standard deviations of the students' evaluations that were given with the survey questionnaire assessing their perception of the various characteristics of the tools. The survey items are categorized into four groups. The first group, *Ease of learning the modeling tool*, contains the three items related to the understandability and learnability of the tool. In this regard, the students quite unanimously agreed that Archi was the easiest to learn, and also that the logic of Archi was the easiest to understand (combined mean 3.96). On the other hand, Arter Arc was considered to be the most difficult to master (combined mean 2.64). This is understandable due to its unique and more time-consuming approach to creating models, coming from a different intended tool usage, as previously discussed.

The second group, *Ease of creating EA artefacts*, addressed the ease of creating different EA artefacts and the ease of modifying previously created artefacts. In practice, the latter is an important feature as it facilitates that the EA artefacts remain up-to-date. Again, Archi scored the highest marks for almost every item in this group (combined mean 3.62) while Arter Arc got the lowest average score (combined mean 3.11). It appears that the students considered creating Data Model and Application / Data Matrix types of artefacts somewhat more difficult than the other models they were asked to create during the exercise sessions. The take-away from this observation could be that there might be a need for more emphasis on modeling in the first semesters of the Bachelor program. Data modeling is not an EA-specific, but rather generic learning target in IT studies. A basic understanding might support the creation of data models, however, with this experiment we cannot discern whether it is the technicalities of the tool usage, or the basic understanding of data models and data modeling behind the result.

The third group of evaluation items, *Support for understanding different EA artefacts*, addressed the artefacts' understandability and capability to convey intended information. Overall, Archi and Arter Arc scored the highest marks with

TABLE II
MEANS AND STANDARD DEVIATIONS OF MODELING TOOL ASSESSMENTS

Ease of learning the modeling tool	Microsoft Visio	Archi	Arter Arc
The logic of this tool was easy to understand.	3.05 (1.05)	4.09 (0.75)	2.62 (1.39)
This tool was easy to learn.	3.05 (1.05)	4.05 (0.72)	2.54 (1.27)
I am able to use this tool well after completing the exercises.	3.09 (0.61)	3.73 (0.55)	2.77 (1.24)
Mean	3.06	3.96	2.64
Ease of creating EA artefacts	Microsoft Visio	Archi	Arter Arc
Creating Business Process Map was easy with this tool.	3.50 (0.91)	3.64 (0.95)	3.31 (1.03)
Creating Map of Applications was easy with this tool.	3.14 (1.17)	3.95 (0.95)	2.92 (1.04)
Creating Map of Company Goals was easy with this tool.	3.55 (1.22)	4.09 (0.92)	3.23 (1.09)
Creating EA Development Roadmap was easy with this tool.	3.41 (1.10)	4.00 (0.87)	3.08 (0.86)
Creating Data Model with this tool was easy.	2.82 (0.85)	2.95 (0.95)	2.69 (1.03)
Creating Application / Data Matrix was easy with this tool.	2.55 (0.91)	3.05 (0.79)	2.92 (0.95)
It was easy to modify the models with this tool.	3.09 (0.92)	3.68 (0.99)	3.62 (1.04)
Mean	3.15	3.62	3.11
Support for understanding different EA artefacts	Microsoft Visio	Archi	Arter Arc
This tool helped me understanding the meaning of the models.	3.45 (0.96)	3.91 (0.81)	3.46 (0.88)
The models created with this tool conveyed intended information content.	3.41 (0.91)	4.09 (0.53)	3.62 (0.96)
The models created with this tool were easy to understand.	3.64 (1.00)	4.05 (0.74)	4.08 (0.49)
This tool would support the communication between different stakeholders.	2.55 (0.86)	1.95 (1.21)	2.31 (1.18)
Mean	3.26	3.50	3.37
Support for learning creating EA artefacts	Microsoft Visio	Archi	Arter Arc
This tool supported modeling.	3.41 (0.91)	4.05 (0.65)	3.69 (0.75)
This tool supported my creativity while modeling.	3.41 (0.73)	3.77 (0.61)	3.69 (0.75)
This tool supported creating ArchiMate compliant models.	2.86 (0.94)	3.73 (1.16)	3.00 (1.15)
This tool helped me modeling correctly.	3.05 (1.05)	3.82 (0.80)	3.31 (0.75)
Mean	3.18	3.84	3.42
Overall grade of the tool	3.23 (0.87)	4.09 (0.61)	3.46 (0.88)

little differences. For the ability to support communication between the different EA stakeholders (e.g., business and IT), the otherwise well-evaluated Archi was assessed to perform quite poorly in this regard (combined mean 1.95). This is probably due to its strong orientation towards the ArchiMate modeling, which students considered not being an optimal medium for communicating with business stakeholders. EA's capability to support communication and coordination in different transformation efforts is considered as one of the key success factors [28]–[30], while studies report that this often fails [8], [31]. This leads to the question, whether it is possible to combine in one tool an optimal user experience for such different stakeholder groups as modelers working on the EA practicalities, and the business managers making the EA related decisions. The learning experiment seems to convey this difficulty in EA practices.

Finally, the fourth group, *Support for learning creating EA artefacts*, contains the items that directly contribute towards learning the modeling and use of the ArchiMate language. Again, Archi was evaluated as being the strongest performer (combined mean 3.84), which was expected, since it provides built-in ArchiMate documentation and the tools for model validation. After completing the classroom exercises, the students were allowed to freely choose the modeling tool (not limited to the three options discussed in this paper) that they would use while working on the modeling assignment mandatory to pass the course. More than two thirds of the students chose to use Archi.

While filling the survey questionnaire, the students were also able to give open comments. The following three comments (translated to English) summarize the opinions regarding the tools. Microsoft Visio Professional: "*I have used Visio in many companies I have worked in. It appears to be licensed everywhere around and therefore is the de facto modeling tool. It is a decent tool, for example, for creating simple business process models but for the ArchiMate modeling there seem to be better solutions available.*" Archi: "*[...] user-interface is nice and simple. The tool was quick to pick up and learn (as it has a similar structure as UMLet) and finding the correct modeling elements was always easy. I especially liked the Hints tool because it gave me more information about the modeling elements and their properties while working on a model.*" Arter Arc: "*This is a complicated tool and difficult to use. It took a lot of learning before I was able to create a model and the amount of work was double compared to the other tools. However, after all this work was done, the end results were clear and very good.*"

V. DISCUSSION

EA is one of the seven high-level competency areas of IS 2010 Curriculum Guidelines for Undergraduate Degree Programs in Information Systems. IS graduates should be able to design, deploy and maintain an EA, which necessitates crafting an effective, usable and communicable set of

EA artefacts. Many EA artefacts, however, are regarded as challenging to create and maintain [32]. In the practice, EA is considered requiring seniority, broad understanding of both the business and the IT contexts, and requisite communication and coordination skills. All these appear demanding learning targets. For learning, students need to examine the purpose and concepts of EA, and apply them in practical use cases that are used in communicating the issues for both business and IT stakeholders. While education-oriented EA modeling tools are not available, there are both commercial and freeware solutions that can be used in teaching and learning EA. In our course, we used three different tools with slightly different profiles:

- A mainstream general modeling tool, fully compatible with a broad set of other office applications likely familiar to the students.
- A free tool with modeling functionality, but not a model database or other functionalities to be used in the management of EA, already familiar to the students from earlier modeling exercises.
- A professional EA tool, with modeling functionality, a model repository as well as multi-user support, and strengths in the model and architecture management functionalities.

Microsoft Visio Professional — for EA modeling equipped with the ArchiMate stencils actively used by the EA professional community [27] — is familiar to many IS students from their previous modeling and diagramming exercises. Despite this, Visio did not inspire a lot of enthusiasm. However, the comments also point to the fact that anyone working with modeling in diverse settings is likely to encounter models created with Visio, and are possibly required to modify them and maybe create some more models with this tool. Therefore, we would consider MS Visio a "required element" in the modeling choreography for an EA course. We are also glad to see that this tool is already broadly known.

While it is not education-oriented per se, compared to, for example, some of the UML modeling tools available, a number of Archi's features are suited for educational purposes, and seem to support the learning of modeling business and technology environments, especially if the industry-standard ArchiMate modeling language is used. However, due to its current technical limitations (e.g., the lack of database repository, and multi-user support) it obviously has its shortcomings when considering more complex and demanding use cases akin to real-world EA modeling problems. What's more, for enterprises managing their EA, maintaining an EA repository is an important concern. Modeling may be a task outsourced to IT providers or consultancies, but for the management of EA, the enterprise should have the capability to maintain a knowledge base on its own EA. The benefits of EA are often dependent on the quality and accessibility of the related information by the enterprise itself. We would consider Archi, or a similar free tool, a very good choice for learning purposes to practice the skills on diverse EA models and in learning the craft. However, as the only choice it would leave out some significant EA specific elements such as modelling and managing models specific to EA. A plus is of course that students can freely download the tool and practice on their own.

Finally, the professional tool Arter Arc, providing excellent multi-user support, version control, rollback features for the models, and versatile capabilities for metamodel customization, unsurprisingly appeared more difficult to learn than the other tools. This reflects the learning effort EA requires; together with modeling comes the need to understand the underlying metamodel, as well as model management. Despite the challenges in learning, the students appreciated Arter Arc's capabilities, and appeared to be genuinely interested in its approach to modeling, which necessitated and provoked more profound understanding of the problem area. The aspect of support for the EA management, which is the strength of the Arter Arc tool, was not part of the experiment and it would require a different experimental setting to examine this phenomenon. In line with the status and the targets of this course, our study focused on the practical modeling and the understanding of individual models. It is crucial that the students, who are in the beginning of their learning path towards enterprise architecting, get to experiment with a professional tool. This conveys the complexity of EA on one hand, but on the other, the potential power of a tool-supported, organized EA information base. This also sets further learning targets.

Strong preference of the free Archi could indicate that the tool indeed supports the task at hand, i.e. modeling. Archi may lack features that would be necessary in a real enterprise practice, but as this is the first introduction to EA modelling, a simple tool lowers the threshold for learning. Maintaining a model repository with version control and providing search and access to the model repository also count in the EA work, and should be considered only as further learning targets. Practical experience with professional tools helps students adjust to work environments, and hopefully inspires students to continue in this learning area.

Long term work naturally makes more steadfast with modelling, and the use of a specific tool develops an overall ability to understand languages and generalize this knowledge into different business domains, languages, and tools. The target to have an overall competency in EA modelling combined with a fluent use of diverse tools needs more practical work from students. What teaching could contribute here is the requirement to *use two tools at minimum*. With two tools, the generic and tool specific elements of the modelling task are conveyed and a critical evaluation of tools can be developed. The role of tools in EA work is significant, and even more than a language, the tool may be defining how the EA work is done. Furthermore, even though *using more realistic tools or environments* in teaching makes learning more difficult [33], it facilitates skills required in the students' future work environments. This, in turn, helps students *understand a phenomenon rather than a tool*, supporting the view of deep learning.

VI. Conclusion

Enterprise architecture is a crucial conceptual blueprint in facilitating a holistic understanding of an organization's information technology alignment with respective business processes, and modeling enterprise architecture is a crucial high level skill of future professionals. In this study, we set out to describe and evaluate three tools used in enterprise architecture modeling. Our results show, among other findings, that despite strict conformance checks, students found Archi easier to learn and use than Arter Arc and Microsoft Visio. Furthermore, Archi was seen to more effectively facilitate the process of learning modeling than the other tools. We concluded that it is crucial for information technology students to understand the high level view, and to learn to model enterprise architecture landscapes with more than just one tool in order to facilitate deep understanding of modeling languages and convention.

References

[1] H. Topi, J. S. Valacich, R. T. Wright, K. Kaiser, J. Nunamaker, J. C. Sipior, and G. J. de Vreede, "IS 2010: Curriculum guidelines for undergraduate degree programs in information systems," *Communications of the Association for Information Systems*, vol. 26, 2010. [Online]. Available: doi.org/10.17705/1cais.02618

[2] "MSIS 2016: Global competency model for graduate degree programs in information systems," Tech. Rep., may 2017. [Online]. Available: doi.org/10.1145/3129538

[3] S. Aier, C. Riege, and R. Winter, "Classification of enterprise architecture scenarios - an exploratory analysis," *Enterprise Modelling and Information Systems Architectures - An International Journal*, vol. 3, no. 1, pp. 14–23, 2008. [Online]. Available: doi.org/10.18417/emisa.3.1.2

[4] F. Rahimi, , J. Gøtze, C. Møller, and and, "Enterprise architecture management: Toward a taxonomy of applications," *Communications of the Association for Information Systems*, vol. 40, pp. 120–166, 2017. [Online]. Available: doi.org/10.17705/1cais.04007

[5] C. Nygaard, T. Højlt, and M. Hermansen, "Learning-based curriculum development," *Higher Education*, vol. 55, no. 1, pp. 33–50, 2006.

[6] P. Wolf, "A model for facilitating curriculum development in higher education: A faculty-driven, data-informed, and educational developer–supported approach," *New Directions for Teaching and Learning*, vol. 2007, no. 112, pp. 15–20, 2007.

[7] R. Winter and R. Fischer, "Essential layers, artifacts, and dependencies of enterprise architecture," in *2006 10th IEEE International Enterprise Distributed Object Computing Conference Workshops (EDOCW'06)*. IEEE, 2006. [Online]. Available: doi.org/10.1109/edocw.2006.33

[8] V. Seppänen, K. Penttinen, and M. Pulkkinen, "Key issues in enterprise architecture adoption in the public sector," *Electronic journal of e-government*, vol. 16, no. 1, 2018.

[9] E. Niemi and S. Pekkola, "Using enterprise architecture artefacts in an organisation," *Enterprise Information Systems*, vol. 11, no. 3, pp. 313–338, 2015. [Online]. Available: doi.org/10.1080/17517575.2015.1048831

[10] S. Bischoff, S. Aier, and R. Winter, "Use it or lose it? the role of pressure for use and utility of enterprise architecture artifacts," in *2014 IEEE 16th Conference on Business Informatics*. IEEE, jul 2014. [Online]. Available: doi.org/10.1109/cbi.2014.56

[11] B. H. Cameron and E. McMillan, "Analyzing the current trends in enterprise architecture frameworks," *Journal of Enterprise Architecture*, vol. 9, no. 1, pp. 60–71, 2013.

[12] H. Jonkers, M. M. Lankhorst, H. W. L. ter Doest, F. Arbab, H. Bosma, and R. J. Wieringa, "Enterprise architecture: Management tool and blueprint for the organisation," *Information Systems Frontiers*, vol. 8, no. 2, pp. 63–66, feb 2006. [Online]. Available: doi.org/10.1007/s10796-006-7970-2

[13] E. Tambouris, M. Zotou, E. Kalampokis, and K. Tarabanis, "Fostering enterprise architecture education and training with the enterprise architecture competence framework," *International Journal of Training and Development*, vol. 16, no. 2, pp. 128–136, 2012. [Online]. Available: https://doi.org/10.1111/j.1468-2419.2012.00400.x

[14] M. Boster, S. Liu, and R. Thomas, "Getting the most from your enterprise architecture," *IT Professional*, vol. 2, no. 4, pp. 43–51, 2000. [Online]. Available: doi.org/10.1109/6294.869382

[15] K. Siau and P.-P. Loo, "Identifying difficulties in learning UML," *Information Systems Management*, vol. 23, no. 3, pp. 43–51, jun 2006. [Online]. Available: doi.org/10.1201/1078.10580530/46108.23.3.20060601/93706.5

[16] P. J. Burton and R. E. Bruhn, "Using uml to facilitate the teaching of object-oriented systems analysis and design," *Journal of Computing Sciences in Colleges*, vol. 19, no. 3, p. 278–290, 2004. [Online]. Available: dl.acm.org/doi/10.5555/948835.948863

[17] G. Engels, J. H. Hausmann, M. Lohmann, and S. Sauer, "Teaching UML is teaching software engineering is teaching abstraction," in *Satellite Events at the MoDELS 2005 Conference*. Springer Berlin Heidelberg, 2006, pp. 306–319.

[18] C. Starrett, "Teaching UML modeling before programming at the high school level," in *Seventh IEEE International Conference on Advanced Learning Technologies (ICALT 2007)*. IEEE, jul 2007. [Online]. Available: doi.org/10.1109/icalt.2007.234

[19] B. T. Mynatt and L. M. Leventhal, "A CASE primer for computer science educators," *ACM SIGCSE Bulletin*, vol. 21, no. 1, pp. 122–126, 1989. [Online]. Available: doi.org/10.1145/65294.71200

[20] Y.-C. Ho, "To what extent will case tools assist users in the systems development?" in *Proceedings of the 1992 ACM SIGCPR conference on Computer personnel research - SIGCPR*. ACM Press, 1992. [Online]. Available: doi.org/10.1145/144001.144040

[21] E. Ramollari and D. Dranidis, "Studentuml: An educational tool supporting object-oriented analysis and design," in *Proceedings of the 11th Panhellenic Conference on Informatics*, 2007, pp. 363–373.

[22] D. Dranidis, "Evaluation of studentuml: an educational tool for consistent modelling with uml," in *Proc. Informatics Education Europe II Conference*, 2007, pp. 248–256.

[23] G. Regev, D. C. Gause, and A. Wegmann, "Experiential learning approach for requirements engineering education," *Requirements Engineering*, vol. 14, no. 4, pp. 269–287, jun 2009. [Online]. Available: doi.org/10.1007/s00766-009-0084-x

[24] G. Regev, L. Regev, Y. Naïm, J. Lang, and A. Wegmann, "Teaching an ethnographic approach to requirements elicitation in an enterprise architecture course," in *Proceedings of the 1st International Workshop on Socio-Technical Perspective in IS Development (STPIS'15)*, vol. 1374, no. CONF. CEUR Workshop Proceedings, 2015, pp. 5–19.

[25] A. L. Steenkamp, A. Alawdah, O. Almasri, K. Gai, N. Khattab, C. Swaby, and R. Abaas, "Enterprise architecture specification case study," *Journal of Information Systems Education*, vol. 24, no. 2, p. 3, 2019.

[26] M. Lankhorst, *Enterprise Architecture at Work*. Springer Berlin Heidelberg, 2017. [Online]. Available: doi.org/10.1007/978-3-662-53933-0

[27] D. Carr and S. Else, "State of enterprise architecture survey: Results and findings," *EAPJ Journal*, no. MAy, 2018.

[28] R. Abraham, "Guidelines for architecture models as boundary objects," in *The Enterprise Engineering Series*. Springer International Publishing, 2017, pp. 193–210.

[29] P. Weill, J. W. Ross, and D. C. Robertson, "Enterprise architecture as strategy," *Massachusetts: Harvard Business School Press. STMIK AMIKOM Yogyakarta*, pp. 6–7, 2006.

[30] T. Tamm, P. B. Seddon, G. Shanks, and P. Reynolds, "How does enterprise architecture add value to organisations?" *Communications of the Association for Information Systems*, vol. 28, 2011. [Online]. Available: doi.org/10.17705/1cais.02810

[31] J. L. G. Dietz and J. A. P. Hoogervorst, "Enterprise ontology in enterprise engineering," in *Proceedings of the 2008 ACM symposium on Applied computing - SAC '08*. ACM Press, 2008. [Online]. Available: doi.org/10.1145/1363686.1363824

[32] S. Kotusev, "Eight essential enterprise architecture artifacts," *British Computer Society (BCS)*, 2017. [Online]. Available: http://www.bcs.org/content/conWebDoc/57318

[33] T. Taipalus, "The effects of database complexity on SQL query formulation," *Journal of Systems and Software*, vol. 165, p. 110576, 2020.