

Improving Enterprise Application Software Development Management With MODAF

Karolis Noreika^[0000-0003-4062-2149]

Institute of Data Science and Digital Technologies, Vilnius University,
Akademijos 4, Vilnius
karolis.noreika@mif.vu.lt

Abstract. The paper deals with the problem of misalignment between business strategy execution and enterprise application software development management. The underlying issues of the problem cause bad results in IT project delivery. The PhD research is based on an internal modelling paradigm that seeks to reveal and model the causality of the domain. The different approaches used to mitigate the problem, like Agile methods or Enterprise Architecture Frameworks are not resolving it completely, therefore a different approach is required. This paper presents steps to formalize an approach to ensure alignment between Agile software development management hierarchy and business management using enterprise architecture framework MODAF. The management transaction (MT) concept is used to reveal the informational content of links in the Agile software development management hierarchy. The MT defines the normalized (causal) content of the Agile items which are considered as self-managed activities. The models of MODAF are used to refine the detailed content of the causal interactions in the Agile hierarchy. The main contribution is a method to ensure every task done in Agile software development projects contributes to a strategic objective of the enterprise.

Keywords: Agile Software Development Management, MODAF, Management Transaction, Business and IT Alignment

1 Introduction

Information systems are considered an integral part of enterprise business operations. However, continuous changes in the business environment and various compliance, legal and market requirements, technical novelties, require constant improvement of existing business processes and this further requires changes to existing and creation of new information systems.

Enterprise application software (EAS) supports business processes over the big scope for interlinked business units and their processes. The development of such systems requires considerable business process knowledge. Managing required development changes via EAS projects is an integral part of the business development area in most enterprises. The development of EAS requires revealing the content of causal links of business activities and, develop EAS that fully supports business operations.

EAS must be aligned with the long-term goals of the organization to ensure that the business operates efficiently and have a competitive advantage. Businesses regularly review and define strategic business objectives to define their long-term goals and support vision and mission statements. However, the information integrity from strategic objectives level to a development task of individual EAS project is not ensured and this results in business and IT alignment problem.

Software project management researches from 2019 by KPMG, AIPM and IPMA [1] indicate that only up to 44% of EAS projects are likely to be delivered that meet the original goal and business intent, only up to 36% on budget and up to 30% on time. This research did not focus only on a single project management methodology: it covers both waterfall and Agile methods.

Using Agile methods for software development management is a way to manage business and IT alignment problem. Agile methods help to adapt to constantly changing business conditions when the environment is ambiguous and the scope of the requirements is highly likely to change over the period of the project. Agile methods also promote frequent feedback, ensuring the possibility to reach alignment between business and IT in a timely manner. "14th State of Agile" report shows, that usage of Agile methods are quite successful with business value delivered in 46% and customer/user satisfaction is achieved in 45% of Agile projects [2]. Although Agile methods improve the success of software development projects, using only Agile methods for software development management proves not to be enough to make sure development done during EAS development projects is contributing to strategic business objectives. This means that business and IT alignment problem is still not resolved sufficiently.

In this paper an approach is presented to support the links between strategic business objectives and the concepts of Agile themes, initiatives, epics, and user stories using the concept of management transaction and enterprise architecture frameworks. This helps to solve the problem of business and IT alignment and business strategy execution with EAS development projects.

Development of EAS also requires an understanding of causality in the particular subject domain. Causal knowledge is described by Zack as a „description of causal links among a set of factors <...> which provides a means for organizations <...> how best to achieve some goal" [3]. The awareness of the specific domain causality is the prerequisite for discovering deep knowledge (i.e. regularities, laws) in a given domain. Causal modelling is aimed at discovering causal dependencies of processes and information attributes in various real-world domains.

Enterprise Architecture (EA) is one of the most widely recognized methods to ensure business and IT alignment. Dahalin et al. states that EA provides "a blueprint for how an organization achieves the current and future business objectives using information technologies" [4]. Enterprise architecture solutions are used to align processes in the enterprise to the supporting information systems and their infrastructure.

This paper is structured as follows: in the second section, the research approach and overview of business and IT interaction modelling is presented, including related works. The third section explains the method to use management transaction to ensure business and IT alignment in an Agile environment using MODAF. The final section contains a summary of the paper and plans for future research.

2 Business and IT Interaction Modeling Overview

The research approach is based on the conceptual scheme presented in Fig. 1 and Fig. 2. External modelling of the system is considered a black-box approach, where the details of internal systems are not clear and are based on input-output results. Internal modelling includes some prior knowledge of the system attributes, its behavior and internal dependencies (Fig. 1). This approach is based on the “good regulator” theorem by Conant et al. [5] and internal model (IM) principle by Francis [6].

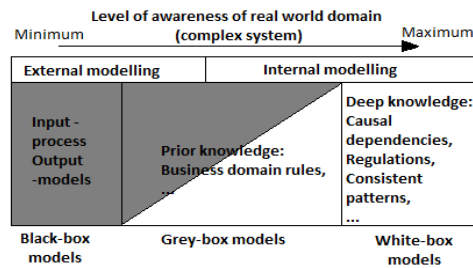


Fig. 1. Different real-world perception approaches

Business management node M1 in Fig. 2 depicts business management methods. Enterprise models node M2 corresponds to enterprise architecture and business process modelling methods. Enterprise activities node M3 depicts real-world domain activities linked by causality dependencies. Domain causality node (M4) represents business domain modelling frameworks aimed for causal dependencies discovery. The EAS block indicates that part of the business management functionality and enterprise activities are IT supported. The conceptual scheme shows that the study is based on an internal modelling paradigm that seeks to reveal and model the causality of the domain.

1. Business and IT alignment process based on the experience: M1 – (M4) – M3, where (M4) denotes the business manager's understanding of causality (experience, mental ideas), which is a mental model (not manifested in an expressive form), therefore marked in brackets.

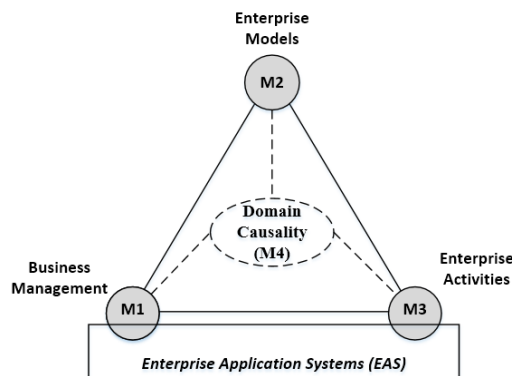


Fig. 2. Approach to Business and IT interaction modelling

2. Business and IT alignment process based on the Enterprise modelling and business process modelling methods (model-driven engineering(MDE) approach): M1 – M2 – (M4) – M3;

3. Business and IT alignment process, based on causal modelling. Domain causality model (M4) is created in expressive form and is implemented in the virtual environment to validate M1 – M2, M2 – M3, and M1 – M3 mapping (1):

$$\{M1 - (M4) - M2; M2 - (M4) - M3; M1 - (M4) - M3\} (1)$$

The business and IT alignment management activities in Fig. 2 can be organized in several ways. The following subsections explain the approach to business and IT interaction modelling as in Fig. 2 in more detail.

2.1 Business Management Methods for Business and IT Alignment

Business management based business and IT alignment methods are generally focused on communication and leadership. This set represents the M1 node as in Fig. 2 and illustrates the business management methods domain. Strategic management is generally based on management approaches, decisions of top leadership in the organization, and the ability to transfer these decisions from top-level throughout the hierarchy of departments and employees. Alkhafaji and Nelson state that “Strategic management is the process of assessing the corporation and its environment in order to meet the firm's long-term objectives of adapting and adjusting to its environment through manipulation of opportunities and reduction of threats” [7].

Interdependencies between different tasks cause coordination problems. An early definition of coordination can be found in Van de Ven et al., who defines coordination as “integrating or linking together different parts of an organization to accomplish a collective set of tasks” [8]. Taxen and Riedl provide a theory of conceptualization of coordination in the IS domain based on a neurobiological perspective [9].

Classical organization theory includes the scientific management approach, Weber's bureaucratic approach, and administrative theory [10].

These methods and theories mostly rely on communication between different levels of managers in an enterprise and are not sufficient to ensure that the information defined as strategic objectives of the enterprise is successfully transferred to a software development task to ensure business and IT alignment.

2.2 Agile Methods and Tools for Software Development Management

As in Fig. 2, Agile software development management is also considered part of business management methods. Over the years, the Agile frameworks such as Scrum, Extreme Programming (XP), ScrumBan, where Scrum is mixed with Kanban framework, Scaled Agile Framework, Spotify model or hybrid versions of these and other methods gained popularity because of adaptability to change and reduction of project costs [2].

EAS projects managed using Agile methods have distinct definitions that represent the hierarchy of tasks in an EAS project to ensure the business is aligned with the de-

velopment of enterprise application software. Agile software project management professionals and vendors use various namings for the elements in the hierarchical structure. Usually, there are four levels used to define the task size and each different level have a distinct definition. The levels in the hierarchy from themes to user stories are also known as TIES, standing for themes, initiatives, epics, and user stories [11].

The lowest level of granularity is defined by the term “user story”. User stories provide not too detail and also not too wide description of the business problem. It was originated by Connextra in the United Kingdom and popularized by Cohn [12]. User stories should span no longer than the single development iteration (1 to 4 weeks).

The next level in the Agile software project management hierarchy is Epic. Epic is bigger in terms of the development capacity required than the available capacity for development iteration [12], [13]. Epics should span no longer than 12 weeks.

The next level is an Initiative. Atlassian, the top vendor of tools for software development management, describes the initiative as “a collection of epics that drive toward a common goal” [14]. Initiatives provide business and IT perspective on achieving business goals and should not span more than 1 year.

Theme is the top level in Agile software project management. It is defined as “an aggregation of user stories to show business value delivered and to help with prioritization as well as show planned product delivery at a high level” [15]. Themes can be considered as the level between initiatives and strategic business objectives. Themes usually take up to 2 years.

The links between TIES structure elements and the strategic business objectives are currently defined by communication between project managers, Agile leaders, and business managers. To ensure business and IT alignment, an additional definition of coordination between business and IT management is required. Big amount of informational units needs to be coordinated during the Agile software development project to achieve the strategic business objectives. As an example, information elements distribution from a real-world Agile EAS development project is presented in Table 1.

Table 1. Agile software project management levels

Hierarchy level	Agile concept	Description	Number of objects
1	Strategic business objective	Short statement explaining long term business goal and having link to vision and mission of the company.	1
2	Theme	High level objective with time constraint and usually monetary goals that allows to achieve strategic business objective.	1-5
3	Initiative	Specific activity to contribute to achieving goals on the theme level.	5-15
4	Epic	Functionality description to achieve goals of linked initiative.	>300
5	User story	Task representing business need or problem.	>3000

Agile methods provide a good way to manage changes and provide transparency to the enterprise application software development process, but it does not on its own ensure that development tasks are aligned with strategic objectives of the enterprise.

2.3 Business and IT Interaction Modelling Methods

Enterprise modelling, business management modelling, and organizational systems theory are interrelated areas of research of complex systems with an active human component (goal-oriented behavior). Business and IT capabilities alignment has been a topic of research for a long time. A very early attempt to tackle the problem was the Business and IT strategical alignment model (SAM) by Henderson and Venkatraman [16]. SAM was proposed to represent business strategy alignment with IT strategy. SAM consists of four domain alignment perspectives (Business strategy, IT strategy, Organization infrastructure and processes, and Information Systems infrastructure and processes). Each domain alignment perspective focuses on a different aspect of the business and IT alignment. However, it is quite fuzzy and focuses mostly on the business management part of the problem. It should be noted that the Business and IT Strategic Alignment Model (SAM) is the early attempt to identify causal dependencies between two enterprise domains: business activities and IT-based activities.

One of the most well-known methods is Guidelines Regarding Architecture Alignment (GRAAL). It is based on the SAM. GRAAL is a conceptual framework providing a collection of concepts and relations among them [17]. Gerow et al. [18] identified 6 definitions of alignment, i.e.: Business alignment, Cross-domain alignment (business strategy to IT infrastructure and processes), Cross-domain alignment (IT strategy to business infrastructure and processes), Intellectual alignment, IT alignment, Operational alignment based on Henderson's and Venkatraman's SAM [16]. But these methods are conceptual and not adapted to be used in most popular enterprise architecture modelling tools.

Enterprise modelling perspective includes such approaches as Service Oriented Architecture (SOA) [19], enterprise architecture frameworks (DODAF, MoDAF, TOGAF [20]), business process modelling, and decision modelling techniques (UML, BPMN, DMN [21]). Also SOA based methods like BITAM by Chen et al. [22]. BITAM uses a twelve-step process for managing, detecting, and correcting the misalignment at the architecture level. Also, the SBISAF framework by Morkevicius et al. [23] is a SOA-based framework that has its implementation in MagicDraw CASE tool using UPDM enterprise modelling language and proved to significantly reduce the misalignment between business and IS in the enterprise model.

Object Management Group (OMG) provides a group of methods under the Business Process Model and Notation (BPMN) and Decision Model and Notation (DMN) category, including BMM, BPMM, BPDM, SBVR, WfMF, CMMN, and VDML [21]. These methods are consistent with the model-driven architecture (MDA) and model-based engineering (MBE) software development approach. Model-based methods are formalized and use structural modelling notations.

However, most of the suggested methods are not adapted to be used in most popular enterprise architecture modelling or software development management tools and take significant time to evaluate the misalignments. Models still need to be translated to project requirements or the requirements themselves should be adjusted manually. The Agile management concepts are also not taken into account, therefore a different method is required to address these issues.

2.4 Enterprise Architecture Modelling

Enterprise architecture (EA) modelling approach was mentioned for the very first time by Zachman [24] in 1987. Kim et al. describe it as “a holistic understanding of all aspects of a business, its drivers and surrounding environments, its business processes, organizational structures, information flows, IT systems, and technical infrastructures” [25]. A properly defined EA allows business to identify and utilize their capabilities in executing business strategy. Enterprise architecture framework models are developed using various different enterprise modelling languages like EEML, UEMML, Archimate. An approach to business and IT alignment in an Agile environment using concepts from Scrum, Scaled Agile frameworks and Archimate was presented in [26].

Business knowledge is captured as models using Enterprise Architecture frameworks, but causal knowledge should also be captured [27]. Causal modelling is suitable for discovering causal dependencies in various real-world domains. Two levels of enterprise causal knowledge modelling were introduced by Gudas. The first level is the presentation of the discovered causation using the Management Transaction (MT) framework. At the second level, a deep knowledge structure of MT is revealed in a more detailed framework called the Elementary Management Cycle (EMC) [28]. We use only MT framework as unified component of causal knowledge (deep knowledge) for an enterprise management model. MT is relevant to some enterprise goal (G) and captures knowledge on enterprise process (P), a feedback loop (F, P), informational input flow (A), informational output flow (V), and management function (F): $MT = (P, F, A, V)$. The conceptual structure of MT is presented in Fig. 3 (a). The conceptual causal model presented in Fig. 3 (b) reveals the internal steps and information flows of MT, also shows a feedback loop of information transferring in-between F and P.

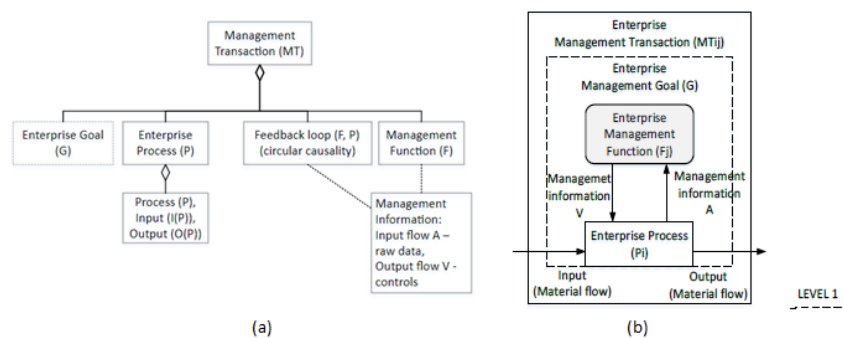


Fig. 3. Management Transaction: a) conceptual structure, b) causal model of MT

British Ministry of Defence Architecture Framework (MODAF) is an Enterprise Architecture framework that is used to model complex systems and uses the concepts of views (Strategic, Operational, System, Acquisitions, Technical Standards, and All views) together with a set of models that help model entire enterprise and its operations. MODAF is the most widely known and established framework. One of the key concepts is the “capability” concept. The strategic views are focused primarily on capability management. A capability is described by MODAF as a specification or classification of the ability of the enterprise [29]. Capabilities change over time, but the main underlying idea is that capabilities enable the organization to act and succeed in the preferred domain based on the skill set and abilities the enterprise has.

3 Mapping Agile Software Management Hierarchy to Enterprise Architecture Framework Concepts

Business strategy execution is a process of acting on the defined strategic plan in an effort to achieve strategic business goals that are derived from the vision and mission statement of the organization. It is an ongoing activity during the whole lifespan of the enterprise. Given the Agile software project management context including the TIES structure, when the business execution process goes from strategic objectives to themes, market researches are done to ensure the planned themes are what the market is expecting from the enterprise. Also, enterprises evaluate their capabilities, strengths to ensure these are utilized in order to achieve strategic objectives. Typically in the business execution process achieving a strategic objective takes effort from many various business functions like marketing, logistics, etc., and very often – IT. New or current improved IT solutions directly make an impact to achieve strategic objectives as customers need service on-demand with fewer approvals or any other delays in their experience.

When the business execution process goes from themes to initiatives, business function managers distribute the high-level tasks with capabilities evaluated to relevant department leaders. In the case of the IT department, this could be reducing costs through “sun setting” a list of legacy applications by either implementing features in more modern EAS or by optimizing the business process so that it does not rely on the legacy system. Once moving through the business strategy execution process from initiatives to epics this will be units of a smaller work effort like replacing a feature in some old legacy software with more modern tech stack tools. A single task to contribute to that is usually called a user story and reflects the business need from the stakeholder’s perspective. In the example case, this may be just moving some software components to a new tech stack. The described business strategy execution process, based on empirical research, is presented as a BPMN diagram in Fig. 4. It also includes the relevant models from MODAF to ensure business and IT alignment.

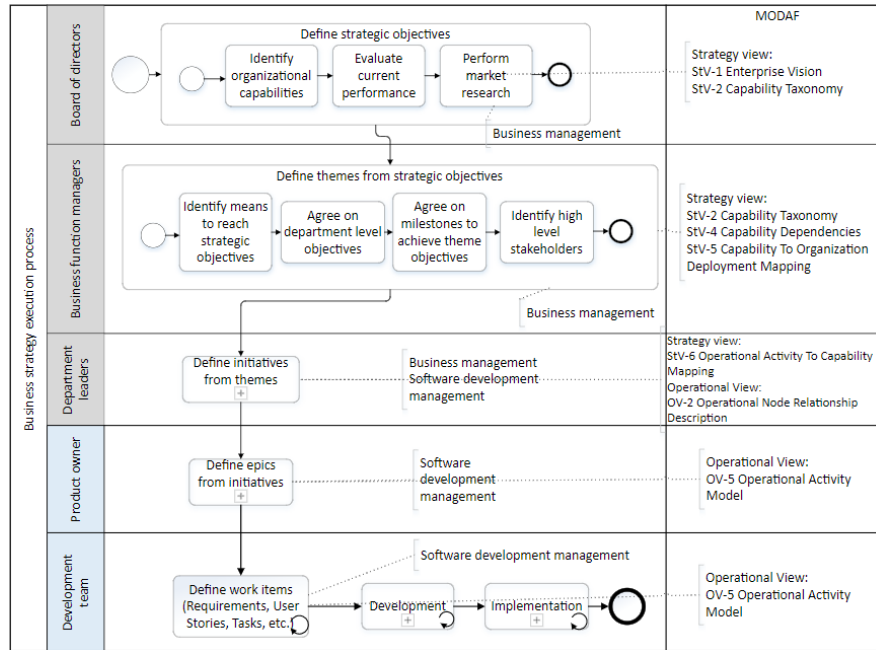


Fig. 4. Business strategy execution process and mapping to MODAF

The section of the diagram on the left represents the business strategy execution process in the context of Agile software project delivery. The views and models from MODAF on the right part ensure alignment between business in IT. Agile project management practices and tools do not provide a structured and formalized approach to define relations in the TIES hierarchy. Therefore we use MODAF views models to structure the Agile project management process. Using the strategy view models of MODAF provides the well-defined mapping of Agile software project management hierarchy to the business management process. Agile management concept mapping to MODAF products is presented in Table 2.

Table 2. Alignment of Agile and MODAF concepts

Agile hierarchy	MODAF products and concepts	
	View	Elements
Theme	StV -1	Capability
	StV-2	Capability dependence
Initiative Epic	StV-6	Operational Node
	OV-2	Operational Activity
		Operational Activity Flow
Epic User story Task Change request	OV-5	Operational Activity
		Operational Performer
		Operational Role
		Operational Activity Flow

“MODAF products and concepts” column in Table 2 shows that management transaction (MT) can be modeled in more detail. MT elements includes causality element, constraints for management transaction and the management information flows of A and V[33].

Table 3. Mapping of Agile concepts to MT and MODAF concepts

Agile items mapping to MT elements	MODAF concepts mapping to MT	MT elements
Theme (T): T := G;	StV concepts: Enterprise Phase (EP), Capability (C): (EP, C) := (G)	Goal (G)
Initiative (I): I := MT (F, P, (A,V));	OV concepts: Operational Node (N), Operational Activity (O), Operational Activity Flow (AF): (N, O, AF) := (F, P, (A,V));	Goal (G) Management function (F) Process (P) – (P1, P2, ...,Pn) Information flows (A,V)

The elements in the “MODAF products and concepts” column in Table 4 represent the logical links between different levels of the Agile software management hierarchy. They do not represent any software development realization. This mapping is used to define the interactions between different levels of Agile management hierarchy (as presented in Fig 4).

4 Summary and Future Works

Based on a review of direct literature and other similar reviews, it can be concluded that there are no uniform approach to solve the business and IT alignment problem when developing enterprise application software (EAS) projects. The paper presented the literature overview of three main research areas for solving business and IT alignment problem: business management, enterprise modelling, and Agile methods.

The mapping of Agile software management hierarchy items (themes, initiatives, epics, user stories) to the views (strategic, operational) of the enterprise architecture framework MODAF was presented. This allows to integrate the top-down and bottom-up transitions between layers of the Agile software management hierarchy. Management transaction is used to capture the causal knowledge between the different levels of the Agile software management hierarchy.

The aim of future research is to provide EAS project leaders an unbiased support tool to present them with the validation rules for EAS development task alignment to the strategic objectives of the enterprise. Each interaction between levels of the Agile management hierarchy needs to be modeled as a management transaction and any misalignment should be presented as a dashboard.

References

1. KPMG, AIPM, IPMA: The future of Project management: global outlook 2019 <https://www.ipma.world/assets/PM-Survey-FullReport-2019-FINAL.pdf>, last accessed 2021/07/12.
2. Digital.ai Software, Inc.: 14th Annual State of Agile Report, <https://stateofagile.com/#ufhi-615706098-14th-annual-state-of-agile-report/702749>, last accessed 2021/06/27.
3. Zack, M.H.: If Managing Knowledge is the Solution, then What's the Problem? In: Yogesh Malhotra (eds.) Knowledge Management and Business Model Innovation, Idea Group Publishing, doi: 10.4018/978-1-878289-98-8.ch002, (2001).
4. Dahalin, M.Z., Razak, A.R., Ibrahim, H., Yusop, I.N., Kasiran, K.M.: An enterprise architecture methodology for business–it alignment: adopter and developer perspectives. In Soliman S.K. (eds.) Communications of the IBIMA, vol. 2011, doi: 10.5171/2011.222028, (2011)
5. Conant, R.C., Ashby, R. W.: Every good regulator of a system must be a model of that system. *International journal of systems science* 1, pp. 89–97, (1970).
6. Francis, B.A., Wonham, W.M.: The internal model principle of control theory. *Automatica* 12(5), 457–465, (1976).
7. Alkhafaji, A., Nelson, R.A.: Strategic Management Formulation, Implementation, and Control in a Dynamic Environment. 1st edn. Routledge London, United Kingdom, (2003)
8. Van De Ven, A. H., Delbecq, A.L., Koenig Jr., R.: Determinants of Coordination Modes within Organizations. *American Sociological Review*, vol. 41, no. 2, 322–338, (1976)
9. Taxen, L., Riedl, R. Understanding Coordination in the Information Systems Domain. *Journal of Information Technology Theory and Application (JITTA)* 17, 5–40, (2016).
10. Onday, O.: Classical Organization Theory: From Generic Management of Socrates to Bureaucracy of Weber *International Journal of Business and Management Review*. Vol.4, No.1, pp. 87–105, (2016)
11. Prior, D.: Agile Planning With Ties With Tom Churchwell., <https://www.leadingagile.com/podcast/agile-planning-ties-tom-churchwell/>, last accessed 2021/07/03
12. Cohn, M.: *User Stories Applied: for Agile Software Development*. 1st edn. Addison Wesley, USA (2004).
13. Agile Alliance: Agile Glossary, <https://www.agilealliance.org/glossary/epic>, last accessed 2021/07/10.
14. Atlassian: Epics, stories, themes, and initiatives, <https://www.atlassian.com/agile/project-management/epics-stories-themes>, last accessed 2021/07/17.
15. McDonald, K.: *Beyond Requirements: Analysis with an Agile Mindset (Agile Software Development Series)*. 1st edn. Addison-Wesley Professional, USA (2015).
16. Henderson, J.C., Venkatraman, N.: Strategic alignment: A model for organization transformation via information technology” Working Paper 3223–90. Massachusetts Institute of Technology, 1990, 458 p. ISBN 9781245057264.
17. Van Eck, P., Blanken, H., Wieringa, R.: Project GRAAL: towards operational architecture alignment. *International Journal of Cooperative Information Systems*, 13(3), 235–255 (2004).
18. Gerow, J.E., Thatcher, J.B., Grover, V.: Six types of IT-business strategic alignment: an investigation of the constructs and their measurement, *European Journal of Information Systems*, 24(5), pp. 465–491, doi: 10.1057/ejis.2014.6 (2015)
19. The OMG: SOA, <https://www.omg.org/technology/readingroom/SOA.htm>, last accessed 2021/08/30

20. The OMG: The TOGAF Standard, Version 9.2, <https://www.opengroup.org/togaf>, last accessed 2021/08/20.
21. The OMG: Business Modeling Category - Specifications Associated, <https://www.omg.org/spec/category/business-modeling/About-business-modeling/>, last accessed 2021/07/06
22. Chen, H.M., Kazman, R., Garg, A.: Managing misalignments between business and IT architectures:a BITAM approach. *Journal of Science of Computer Programming*, 57(1), 5–26, (2005).
23. Morkevicius, A., Gudas, S., Silingas, D.: SBISAF: a service-oriented business and information systems alignment method. *Informatica*, 24(2), pp. 231–251, (2013).
24. Zachman J. A.: A Framework for Information Systems Architecture. *IBM Systems Journal*, Volume 26, Number 3 (1987).
25. Kim, H., Oussena, S., Essien, J., Komisarczuk, P.: Towards Event-Driven Enterprise Architecture. In Perez-Castillo, R., Piattini, M. (eds.) *Uncovering Essential Software Artifacts through Business Process Archeology*, pp. 285–311, IGI Global (2014).
26. Noreika, K.: Business Capabilities Utilization Enhancement Using Archimate for EAS Projects Delivery in an Agile Environment. In: Matulevičius, R., Robal, T., Haav, H-M., Maigre, R., Petlenkov, E. (eds.) *Joint Proceedings of Baltic DB&IS 2020 Conference Forum and Doctoral Consortium co-located with the 14th International Baltic Conference on Databases and Information Systems (BalticDB&IS 2020) CEUR Workshop proceedings*, vol. 2620, pp. 49–56, Estonia (2020).
27. Gudas, S.: Causal Modelling in Enterprise Architecture Frameworks. *Informatica*, pp. 1–35, doi: 10.15388/21-INFOR446 (2021).
28. Gudas, S.: *Foundations of the Information Systems' Engineering Theory*. Vilnius University, Vilnius (2012).
29. British Ministry of Defence: MOD Architecture Framework (MODAF), https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/36757/20100602MODAFDownload12004.pdf, last accessed 2021/07/15.