FAKULTÄT FÜR INFORMATIK

DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

Master′s Thesis in Information Systems

Practices in Agile Architecture Governance: Multiple Case Studies in Large Organizations

Niklas Reiter

FAKULTÄT FÜR INFORMATIK

DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

Master's Thesis in Information Systems

Practices in Agile Architecture Governance: Multiple Case Studies in Large Organizations

Ansätze zur agilen Architektur-Governance: Mehrere Fallstudien in großen Organisationen

Author:	Niklas Reiter
Supervisor:	Ömer Uludağ, M. Sc. Fakultät für Informatik Technische Universität München
Examiner:	Prof. Dr. Florian Matthes Fakultät für Informatik Technische Universität München
Date:	December 7, 2018

I confirm that this master's thesis is my own work and I have documented all sources and material used.

Munich, December 7, 2018 Niklas Reiter

Abstract

Over the past two decades, enterprise architecture management has established itself as an important management function for large IT organizations by aligning both business and IT with strategy and goals. As a governance body, the focus of enterprise architects has been primarily on reducing the operating costs of software applications, managing the complexity of the application landscape, and improving enterprise-wide transparency so far. Driven by external factors, such as changing customer demands, regulatory changes, and technological advancements, a paradigm shift can be observed in large IT organizations. This shift entails fundamental changes in the way large IT organizations work, such as responding to changes over following plans or customer collaboration over contract negotiation. However, the traditional mindset of enterprise architects often hinders the transformation to lean-agile approaches due to their primary raison d'être and thus contradicts the new way of working per se. A reorientation of enterprise architects is therefore essential to enable the digital transformation.

As little research has been conducted on this issue, we fill this gap by providing a multiple-case study on the new role of enterprise architects in supporting large-scale agile development. We present an embedded multiple-case study of four leading German companies from the automotive, information technology, insurance and retail sectors, describe typical challenges faced by enterprise architects in large-scale agile development, and provide a set of recommendations for action for addressing them. In total, we conducted 64 semi-structured interviews with 21 persons from the three stakeholder groups: agile team, enterprise architect and management. The results indicate that the lack of capacity of enterprise architects hampers the support of agile teams and that large IT organizations lack appropriate scaling options at portfolio and organizational levels. In addition, the findings reveal that the enterprise architects' value contribution for supporting agile teams has not yet arrived at the team level, leading to significant acceptance issues. Based on our observations, we provide a collection of recommendations for action to improve the collaboration between enterprise architects and agile teams, such as the collaborative establishment of architecture principles and guidelines, the technical support of agile teams through build platforms and architectural spikes, and the establishment of communities of practice for architecture that replace traditional and tedious architecture boards.

Contents

1. Introduction

This chapter contains the following sections. Firstly, the motivation of this master's thesis is described in Section 1.1. Secondly, the objectives and the respective research questions are presented in Section 1.2. Following this, Section 1.3 describes the underlying research approach.

1.1. Introduction

The increasingly rapid and frequent changes in market conditions, technical and regulatory changes and the necessity to reduce costs, force large IT organizations carry out complex business transformations at more and more frequent intervals [119]. For this reason, a paradigm shift can be observed in large IT organizations. This change brings about fundamental changes in the way large IT organizations work, such as responding to changes in follow-up plans or working with customers to negotiate contracts [57, 142]. IT plays an important role here, as it acts as an enabler for transformation and represents an important success factor for the delivery of customer value on time. However, in order to achieve or adequately support this, IT must continuously align itself with the business and strategic IT processes such as requirement management, application lifecycle management and enterprise architecture management, and also software development must be more flexible [14, 26, 54, 86]. Not only must the software development within the teams follow the agile method, but the Enterprise Architecture Management (EAM) must do so as well [55]. However, this appears to be very difficult, as the traditional EAM is typically driven top-down, while the agile development is driven bottom-up on the one hand, and on the other hand, because EAM focuses on long-term planning and agile development is short-term oriented.

That is why both disciplines must adapt and ensure close collaboration between different stakeholder groups [53]. However, the challenge of how to successfully combine both disciplines remains, as the traditional way of thinking of enterprise architects (EA) and the transformation to agile approaches is often hindered. A new orientation of EAs is therefore essential to enable digital transformation [55, 88, 89]. According to Ruhbart [117] getting the necessary buy-in from architects, developers, and other stakeholders and increasing the awareness of the added value of EAM at all organizational levels is mandatory. In order to meet these challenges, a holistic and explicit picture of the company, the proper management of the enterprise architecture and the right governance mechanisms such as architecture principles and circles are required [131]. Enterprise architecture can briefly be described as a model of the enterprise's most important elements and their relations [5], whereas EAM is the process of creating and using enterprise architecture [106]. Three main functions such as documentation, analysis and planning and design of the enterprise architecture [79] are among others responsible for the success of IT strategy, products and IT governance [78]. Over the past two decades, EAM has established itself as an important management function for large IT organizations by aligning both business and IT with strategy and goals [57, 113, 119]. As a governance body, EA have so far focused on reducing operating costs of software applications, managing the complexity of application landscapes and improving enterprise-wide transparency. However, this has changed. The focus has shifted towards enabling the agile transformation [107]. This is currently taking place in large IT organizations and is leading to some challenges. On the one hand, the cooperation between the EAM and agile software development is still unclear, and on the other hand, the question which new responsibilities the EAM must have in order to act as an enabler of agile transformation has not been answered yet.

This master's thesis fills this research gap by providing a multiple embedded case study on agile architecture governance in four leading German companies from the automotive, information technology, insurance and retail sector. Based on four research questions, this thesis examines the field of tension between agile and traditional EAM and analyzes the status quo of agile principles, circles, the role and the value contribution of the EAM. Thereby typical challenges for EAs in the context of large-scale agile development are described. Subsequently, recommendations for action are derived. In total, 64 semi-structured interviews with 21 persons from the three stakeholder groups: agile team (AT, consisting of developers, a product owner (PO), and a scrum master (SM)), EA, and management have been conducted. Initial results indicate that the lack of capacity of enterprise architects hampers the support of agile teams and that large IT organizations lack appropriate scaling options at portfolio and organizational levels. In addition, the findings show that the enterprise architects' value contribution for supporting ATs has not yet arrived at the team level, which leads to significant acceptance issues. Based on observations, this thesis provides a collection of specific measures to improve the collaboration between enterprise architects and ATs, such as the collaborative establishment of architecture principles and guidelines, the technical support of ATs through architectural spikes, and the establishment of communities of practice and center of excellence for architecture that replace traditional and tedious architecture boards. In addition, based on the results of the case study, an initial EAM model is derived and presented.

1.2. Objectives

The aim of this master's thesis is the analysis of the EAM in an agile environment. Thereby, the suitability of architecture principles and architecture circles in an agile environment is examined. At the same time, it will be examined where architectural decisions are discussed and adopted and which stakeholders are involved. In the next step, tasks and responsibilities of the enterprise architecture management in the agile environment will be analyzed in more detail. The final goal of this master's thesis is to find out how the value contribution of the EAM can be measured. On the basis of the results, recommendations for action and a to-be EAM model with necessary artifacts, roles and a description of the interaction between the EAM process and the respective organizational level are derived. Based on the aforementioned objectives, the following research questions arise:

Research question 1 (RQ1): What is the motivation behind the definition of architectural principles, which architecture principles are suitable for an agile environment and who is responsible for their definition and compliance?
The objective of the first research question is to identify architectural principles suitable for an agile environment. Firstly, to get a better overview, goals and drivers for defining architecture principles should be identified. In the next step, the process and stakeholders responsible for creating and managing architectural principles should be examined. Finally, this research question aims to analyze compliance mechanisms, measures, and problems implementing architecture principles.

Research question 2 (RQ2): What kind of architectural decisions are made in an agile environment and how are they categorized, documented and communicated through the entire organization?
The objective of the second research question is to identify and classify architectural decisions and to investigate, in which circles they are discussed and documented. Firstly, it should be analyzed how architectural decisions are categorized and documented. Secondly, it should be examined at what organizational level and by which role an architectural decision is made to provide a better understanding of how agile environments affect various organizational levels. Finally, the architecture circles practiced in the organizations should be identified and analyzed to find out how architectural concerns are discussed and communicated.

Research question 3 (RQ3): Which tasks, responsibilities, problems, and solutions does the enterprise architecture team have in an agile environment and what is needed for an agile EAM to be successful?
The objective of the third research question is to identify the tasks and responsibilities of an enterprise architecture team in an agile environment. Firstly, the role and responsibility of the EAs and their change in the agile environment should be analyzed. Secondly, specific characteristics and the associated advantages and disadvantages of the EAs should be identified. Thirdly, in order to find out how EAs actually work, which artifacts are and should be provided by the EAs to the ATs in an agile environment is important to know. Fourthly, the change in the way the EAs works is questioned in order to draw conclusions about the extent of changes due to agile transformation. Finally, the problems of EAs in an agile environment and the way of solving these should be identified.

Research question 4 (RQ4): How effective is the enterprise architecture team and what are the expectations of ATs regarding collaboration and artifacts / models provided by the EAs and what approaches exist to measure the value contribution of EAM?
The objective of the fourth research question is to identify the effectiveness of the EAM. To assess this, the first step is to find out which information EAs need from ATs in order to provide suitable models. This raises the question of which architectural models are currently available and which should be made available in the future. In order to understand the entire process and the status quo, further aspects of as-is and to-be should be investigated. These include an AT's expectations on the availability and communication of an EA, the involvement of ATs in architectural processes, and the form and frequency of feedback and support of EAs. Afterwards, criteria are identified to measure current and future value contribution. Finally, it should be analyzed whether the role of EAs is recommended or not and how high the recommendation level is.

1.3. Approach

This master's thesis utilizes case study research methodology [121, 150] as the underlying research approach to address the objectives and research questions formulated in Section 1.2. A case study is a suitable research methodology for this master's thesis, as it studies contemporary phenomena in its natural context [121]. The case study research process comprises four phases (see Figure 1.1) as proposed by Runeson and Höst [121].

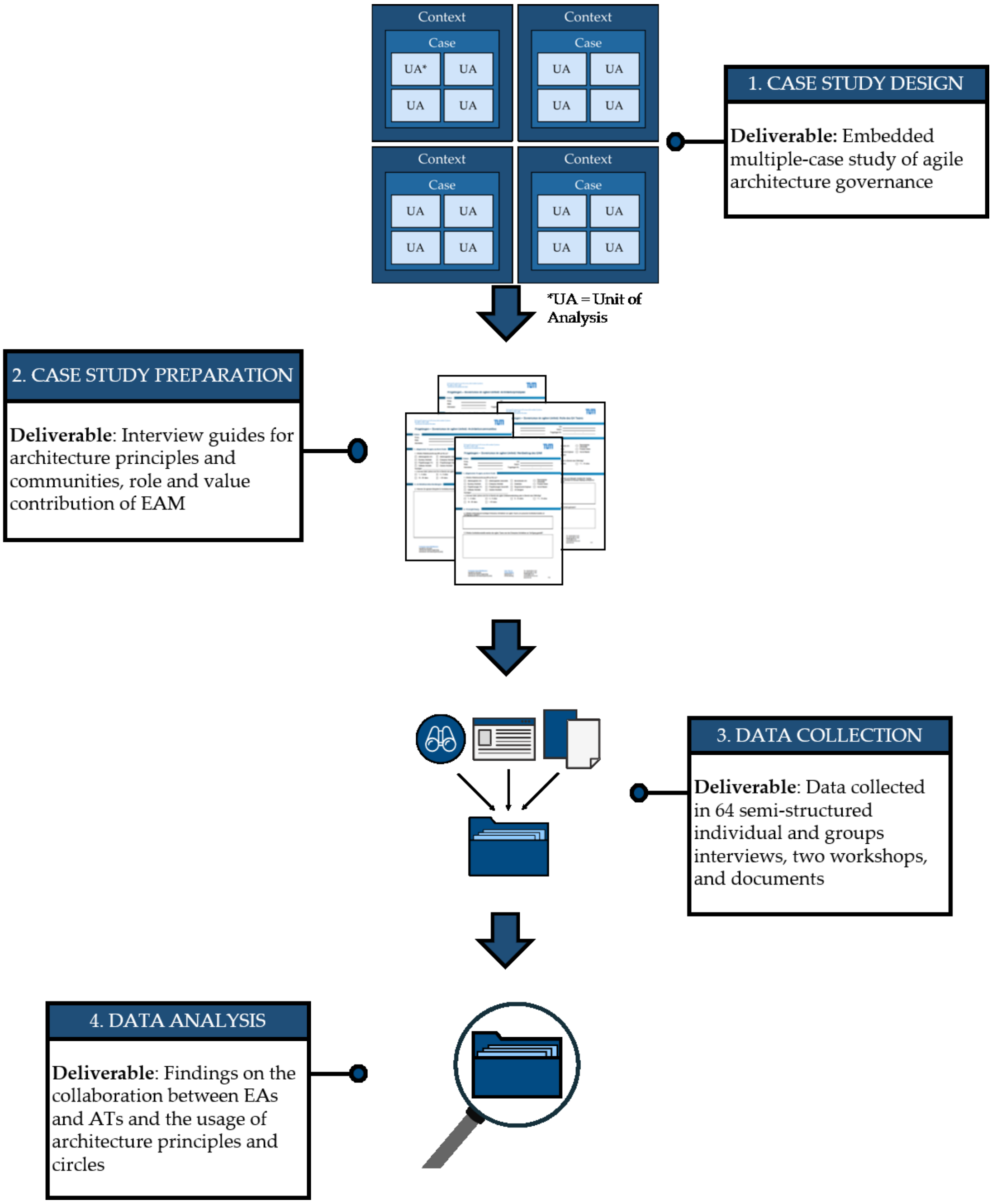

Context
Case
UA*
UA
Context
Case
UA
1. CASE STUDY DESIGN
Deliverable: Embedded multiple-case study of agile architecture governance
*UA = Unit of Analysis
2. CASE STUDY PREPARATION
Deliverable: Interview guides for architecture principles and communities, role and value contribution of EAM
3. DATA COLLECTION
Deliverable: Data collected in 64 semi-structured individual and groups interviews, two workshops, and documents
4. DATA ANALYSIS
Deliverable: Findings on the collaboration between EAs and ATs and the usage of architecture principles and circles

Figure 1.1.: Overview of the multiple-case study approach

The first phase is concerned with the selection of the case study's design. This master's thesis employs the embedded multiple-case study design (Type 4 as described by Yin [150]), since multiple units of analysis are investigated within four organizations. The cases were purposefully selected because the studied companies are currently in the midst of major agile transformation phases and their traditionally well-established EAM functions face new challenges while collaborating with ATs. The embedded multiple-case study consists of four units of analysis, addressing the four previously formulated research questions in Section 1.2. The second phase is concerned with the selection of appropriate means for the subsequent data collection process in the third phase. First-degree data collection techniques were applied by conducting semi-structured interviews with three stakeholder groups, namely ATs, enterprise architects, and managers, and two scaling agile practices workshops [134] with case organizations. Moreover, this master's thesis uses already available documents of the case organizations as a third-degree data collection technique [84]. The purposeful selection of three different data sources facilitates the triangulation of data sources [129]. In the last phase, workshop protocols, interviews, and documents were coded using a deductive approach [30]. Thereby, the semi-structured questionnaires have guided the a priori list of codes. After initial coding, codes were refined and consolidated by merging related ones and removing duplicates. Subsequently, the concepts were related to the formulated research questions.

The remainder of this master's thesis is organized as follows. Chapter 2 covers a collection of theoretical foundations, such as governance in general as well as corporate and IT governance, enterprise architecture management, agile enterprise architecture management and architecture principles and circles. Chapter 3 highlights several publications in the field of agile and large-scale agile development that emphasize the importance of architecture and related governance mechanisms. Chapter 4 presents the findings of the embedded multiple-case study on architecture principles and circles applied in an agile environment and the role and value contribution of the EAM for supporting ATs. Subsequently, chapter 5 delineates the main findings and limitations of this master's thesis. Chapter 6 concludes the master's thesis with a summary of results and a brief outlook for further investigations.

2. Foundations

This chapter describes the theoretical foundations of this master's thesis. The goal is the establishment of a common understanding of terms and concepts. Firstly, governance is described in Section 2.1. In particular, the areas of corporate governance (see Section 2.1.1), IT governance (ITG) (See Section 2.1.2) and agile IT governance (AITG) (see Section 2.1.3) are described. Secondly, a definition of EAM (EAM) (see Section 2.2) and EAM in the agile environment (see Section 2.2.3) is given. Thirdly, EAM and EAM in the agile environment are compared and the biggest differences are highlighted (see Section 2.2.4).

2.1. Governance

Governance is generally referred to as a control and regulation system. With the help of structures for operational and organizational structures, companies are given an internal framework for action. This includes both material requirements and procedural elements such as transparency and reporting obligations. The aim of the control and regulation system is to improve the management of an organization through better achievement of objectives [43]. In the following, corporate governance, ITG and agile ITG are presented and distinguished in detail. In addition, COBIT [66], TOGAF [140] and ITIL [1] are described as particularly relevant frameworks for establishing and ensuring an ITG in companies [69, 72].

2.1.1. Corporate Governance

Corporate governance generally describes the legal and factual regulatory framework for the management and supervision of a company [43]. The focus here is on the goal of reconciling the interests of the stakeholders involved. In its narrowest definition, corporate governance only has the task of protecting the shareholders from being overprivileged by the employed managers of a company. Theoretical modelling shows that the concrete design of corporate governance systems varies according to the desired and granted say of the interest groups [46]. Gillan & Starks and Shleifer & Vishny define corporate governance as a system of laws, rules and factors that control the operation of a company [45, 128]. Regardless of the definition used, the mechanisms of corporate governance are often subdivided into internal and external areas.

Figure 2.1 shows the relationship between the two areas. The left side contains the basics of internal control. In this case, the management, acting as a shareholder represen-

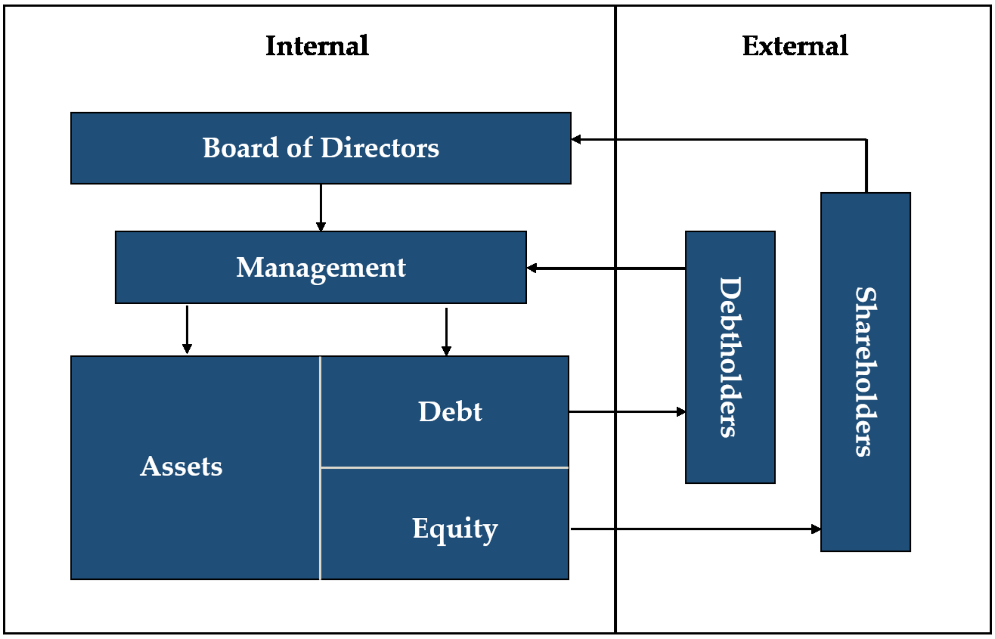


Figure 2.1.: Corporate governance and the balance sheet model of the firm [46]

tative, decides in which assets to invest and how these investments are to be financed. The board of directors is at the head of the internal control system and is responsible for advising and monitoring the management and for hiring, dismissing and compensating the executive board [70]. The right side of the diagram introduces elements of external governance resulting from the need to raise capital. In practice, a separation between investors and capital providers exists. This separation creates the need for corporate governance structures [46].
As early as 1999, the Organization for Economic Cooperation and Development (OECD) began to develop proposals and guidelines for the preparation of relevant structures for legislative and regulatory initiatives and for companies. The most recent version dates from 2015 and is today's international benchmark [37].

In summary, corporate governance is primarily concerned with the problems of delegation of tasks, competencies and responsibilities and methods for solving them [133]. Corporate governance regulations have the fundamental task of restricting the scope and motivations of the actors for opportunistic behavior by means of suitable legal and factual arrangements [99, 128].

2.1.2. IT Governance

IT governance (ITG) is an essential part of corporate governance and consists of management, organizational structures, processes and roles that ensure that information technology (IT) optimally supports corporate goals and strategy. As an important part of corporate management, it creates the prerequisites as well as a framework for optimal support of the corporate goals [53, 106]. Furthermore, it institutionalizes best practices for planning, acquiring, implementing, and monitoring IT operations and transformation [140]. ITG depends strongly on the organizational structure of the IT of a company. There is a difference between an IT department in which the specialist departments also have decision-making rights and an outsourced IT department with a responsible manager. Nevertheless, making sure, that ITG is not an end in itself, but makes a significant contribution to corporate success is the responsibility of the executive board and top management [65, 68]. Therefore, they have to make sure that IT is a regular item on their agenda and is addressed in a structured manner [32]. To guide, review and amend the aligned enterprise and IT strategies the board has to establish an adequate communication and work in close partnership with other boards [32].

Due to the fact that ITG and IT management are often mistakenly used as synonyms, a distinction is important. While IT management is concerned with the planning, organization, control and management of IT resources, ITG focuses on the responsibility of IT and its ability to contribute to achieving corporate goals [138, 143]. IT management, for example, answers questions about how to use IT solutions. ITG, on the other hand, clarifies which frameworks and conditions must be created by the management in order to be able to guarantee the goals and the success of the company [11].

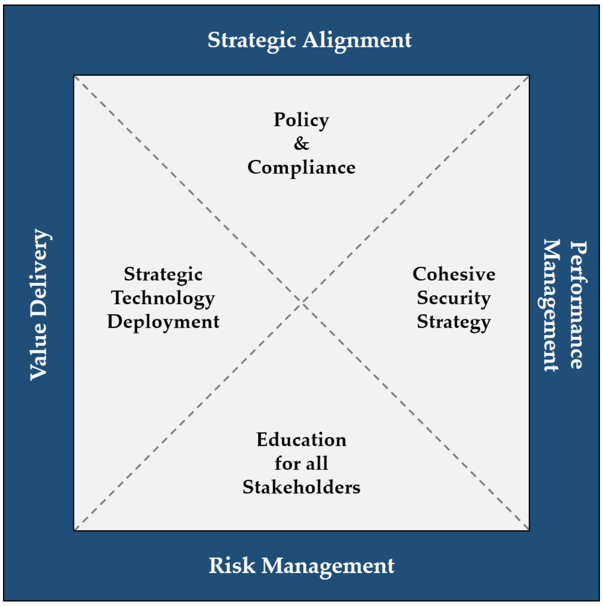


Figure 2.2.: ITG objectives [102]

ITG objectives (see Figure 2.2) in general can be divided into four main areas: strategic alignment, performance management, risk management and value delivery [102]. In order to implement these objectives in the right way, the enterprise needs to have a strong policy, compliance practices and performance and risk management processes citemoeller2013executive. In addition, an overall understanding and support of management for the value delivery is essential [53, 102]. Thereby, governance provides a structure that enables the IT department to meet the requirements of management and corporate strategy. These aspects of governance increase the company's success and minimize the risks arising from IT [53, 106]. Furthermore, the complete IT infrastructure should be aligned in such a way that the strategic goals of the company are in focus [106]. The costs and benefits of IT are constantly under review and being weighed up. Resources continuously are being optimized in order to mitigate possible risks [65]. In order to achieve the objectives mentioned, companies make use of frameworks [69, 72] to establish ITG properly. Particularly relevant frameworks for establishing and ensuring an ITG in companies are:

- Control Objectives for Information and related Technology (COBIT) (see Figure 2.3) was developed by ISACA [93]. The latter is currently available in its fifth edition [66]. In the current version it integrates five essential elements of existing reference models (e.g. ITIL, PRINCE2, TOGAF) and thus presents itself as a higher-level framework for "Governance and Management of Corporate IT". Among other things, the framework is based on five "Key Principles" as well as five governance and 32 IT management processes, which are combined in a process reference model [44, 66].

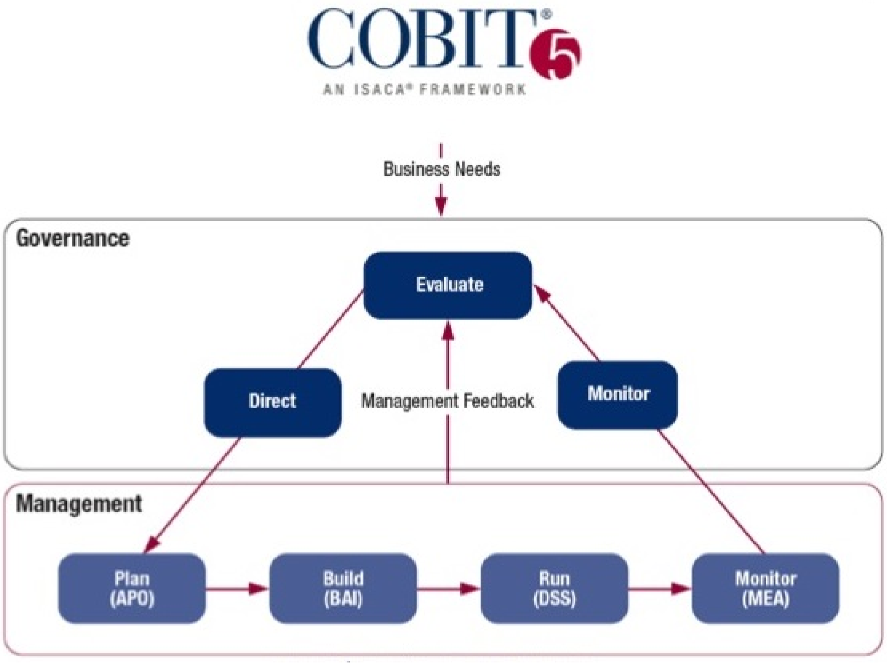


Figure 2.3.: Overview IT Governance - COBIT 5

- The Open Group Architecture Framework (TOGAF) (see Figure 2.4) provides an Architecture Governance Framework as a guideline for successful architecture governance. It defines architecture governance as ”the practice and orientation by which enterprise architectures and other architectures are managed and controlled at an enterprise-wide level.” For example, this includes the introduction of a system to ensure compliance with internal and external standards and the development of practices that ensure the accountability of stakeholders [140].

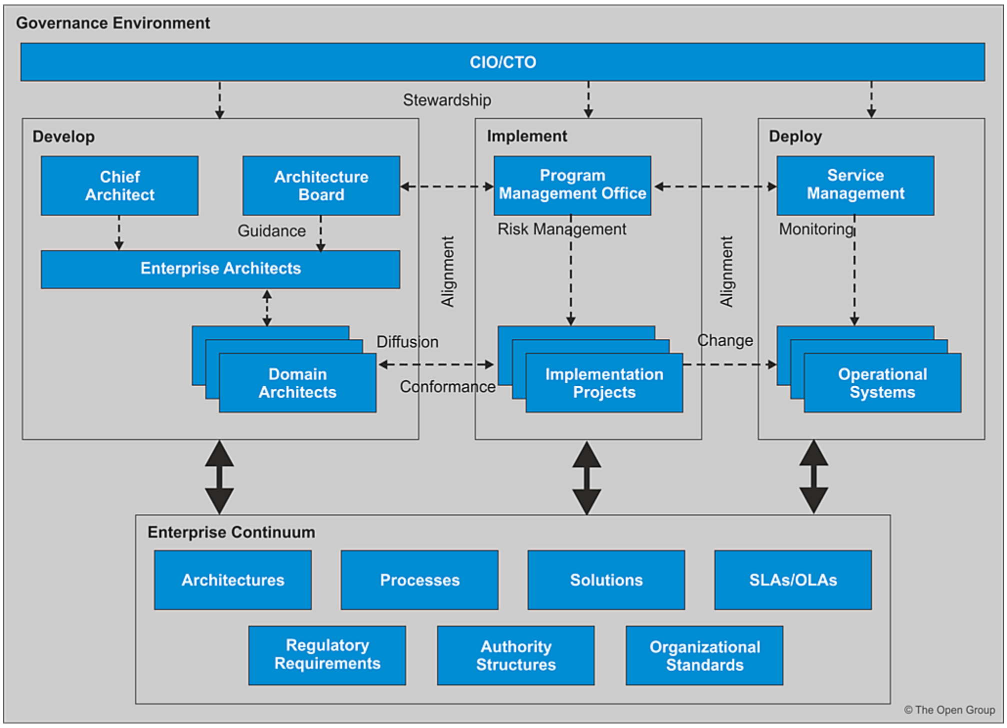


Figure 2.4.: Overview Architecture Governance Framework (Organizational Structure) - TOGAF [140]

- Information Technology Infrastructure Library (ITIL) (see Figure 2.5) focuses on service management and has been developed by the British CCTA since 1989 [1]. It provides detailed guidance on how processes or procedures should be designed. Furthermore, it focuses on defining how to plan, design and implement effective service management processes [98].

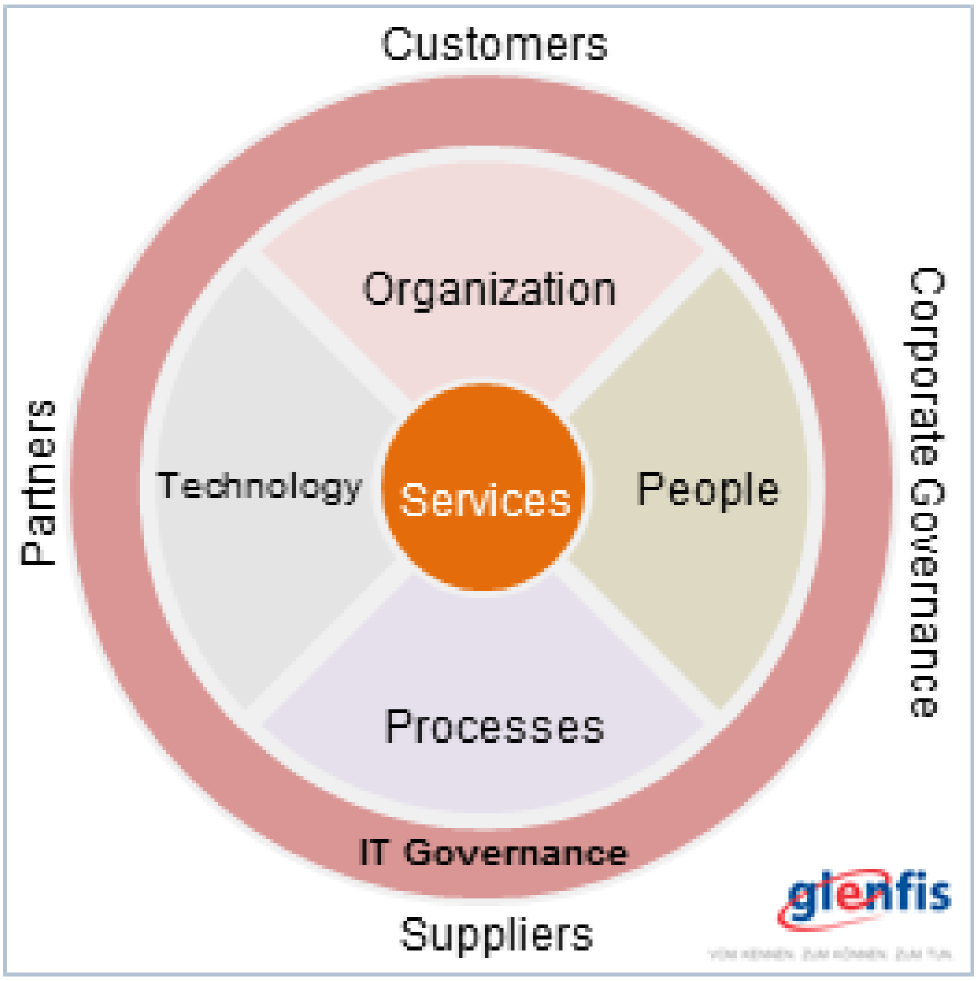


Figure 2.5.: Overview IT Governance and Service Management - ITIL [98]

In practice, COBIT, ITIL and TOGAF are often combined since COBIT covers both sides, governance (Evaluate, Direct and Monitor) and management (Align, Plan and Organize) (see Figure 2.6). It focuses on what should be covered in processes and procedures and is placed on a higher layer than ITIL and TOGAF. The ITIL and TOGAF, on the other hand, cover the management areas of IT only.

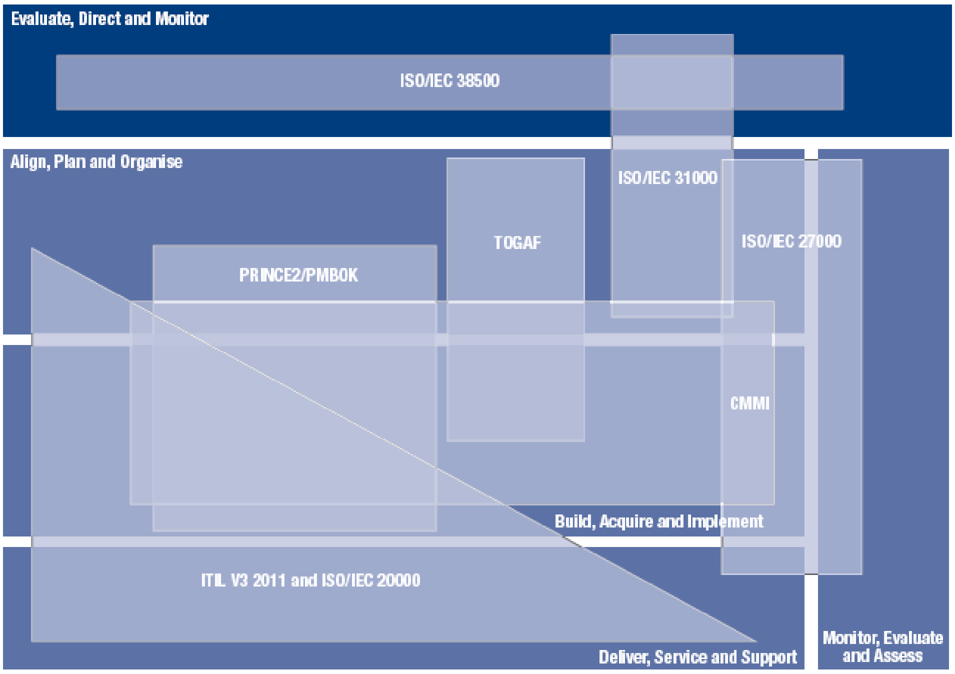

Evaluate, Direct and Monitor
ISO/IEC 38500
Align, Plan and Organise
ISO/IEC 31000
TOGAF
ISO/IEC 27000
PRINCE2/PMBOK
CMMI
Build, Acquire and Implement
ITIL V3 2011 and ISO/IEC 20000
Deliver, Service and Support
Monitor, Evaluate and Assess

Figure 2.6.: Overview COBIT ITIL [98]

In summary, ITG as a part of corporate governance consists organizational structures, frameworks, processes and roles that ensure that information technology (IT) optimally supports corporate goals and strategy [53, 106]. Furthermore, it institutionalizes best practices for planning, acquiring, implementing, and monitoring IT operations and transformation [140].

2.1.3. IT Governance in the Agile Environment

Due to frequent changes in market conditions, technical and regulatory changes, and the need to reduce costs, companies are forced to perform complex business transformations at ever shorter intervals. As a result, large IT organizations must be able to act flexibly and in a timely manner [7]. In order to respond to changing needs and deliver customer value in shorter time frames, business, governance, IT and therefore IT governance must be adapted to agility [54, 71, 77]. The research shows that the importance of agile governance has been recognized by the scientific community [27, 71, 92, 115], but has not been sufficiently highlighted to date, and that there is no uniform definition of the term. Research also shows that the role of autonomy and authority in the integration of agile ITG has not yet been sufficiently investigated [62]. An adaptation of the ITG to agility is required, especially in the context of agile software development [59]. Some authors point out that the traditional governance approach is driven top-down and thus does not fit in with the self-responsibility and self-organization presupposed by agile working methods [91, 92, 115]. Therefore, distinguishing between top-down and bottom-up driven processes is important here. The former refers to processes that are defined, controlled and monitored by higher roles, i.e. by roles that are further up the corporate hierarchy and primarily communicate "top-down", i.e. from top to bottom [83]. The latter describes the opposite (bottom-up). Here, processes are defined on the levels that are directly affected. Communication takes place from bottom to top [83]. In particular, the question arises as to how the balance between control and autonomy between teams and different roles within a company can be achieved. Uludağ et al. [136] suggest that the balance between centralized and decentralized architectural decisions plays an essential role and could have a negative effect on the duration of the decision-making period and thus reduce the benefit [136].

2.2. Enterprise Architecture Management

Despite several publications within the last 31 years, there is still no uniform definition of enterprise architecture and EAM [5, 14, 23, 87, 124]. Enterprise architecture and EAM are sometimes mistakenly used as synonyms [14, 20, 23], even though they differ significantly. Whereas enterprise architecture can briefly be described as a model of the enterprise's most important elements and their relations [5], EAM is the process of creating and using enterprise architecture [106]. Due to the holistic view and the associated higher degree of abstraction, one must clearly distinguish from information and software architecture. Whereas the latter gives developers detailed information about how the software needs to be built, enterprise architecture is rather high level and concentrates on the relevant business and technical structures of the company [52]. In the following, enterprise architecture and EAM will be described in detail.

2.2.1. Enterprise Architecture

Enterprise architecture has gained considerable attention from both, academia and industry [4, 15, 75, 81, 82, 123, 147]. It extends the classic IT architecture with enterprise specific elements [147]. Generally, enterprise architecture can be described an as a model that represents the elements of a company with regard to its strategy, its organization and its IT landscape including applications and infrastructure. Furthermore, it describes the interaction of elements of IT and the economic activity of a company [5]. In doing so, it focuses on the current and future state of architecture (see Figure 2.7) [5, 22, 56, 140].

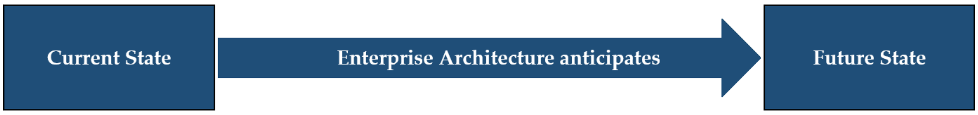


Figure 2.7.: Two Key States of enterprise architecture [126]

It covers various different facets of the enterprise ranging from business-related aspects to more IT-related aspects (see Figure 2.8). Cross-cutting aspects such as goals, projects and standards exert influence or realize the design of the architecture [21]. Goals, for example, describe the desired results, whereas projects are referred to as the ”implementors” of architectural change [21]. This includes business, organizational, application, information, infrastructure and data aspects [100]. The use of this image should enable the understanding of the elements of an enterprise architecture that contribute to its benefits, costs, time and risk in its environment [63]. Based on this understanding, the enterprise architecture is transformed into a target state [95, 103] and then implemented by the enterprise architecture team. In addition to Aier, Rieger and Winter, the ISO standard ISO/IEC/IEEE 42010:2011 is often cited [14, 19, 23, 63, 114, 140, 148]. The standard defines the general requirements for the enterprise architecture. Furthermore, it defines enterprise architecture as a ”fundamental organization of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution”. Principles are often at a higher level of abstraction. Their task is to determine the main features of the architecture in as consistent language as possible and thereby provide a common ground to establish common cross-team architectural standards. For example, principles that guide the development of an enterprise architecture such as profit, continuity or innovation [19]. Since archi-

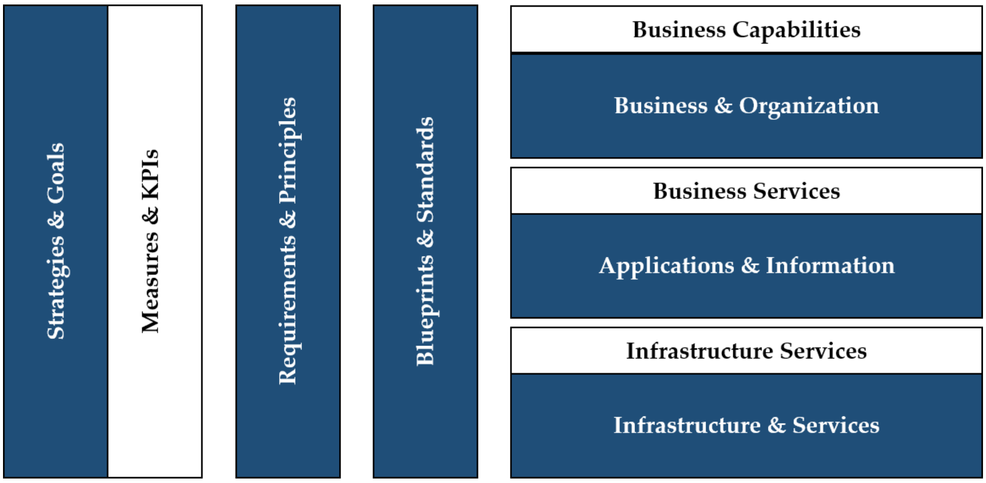


Figure 2.8.: Conceptual structure of an enterprise architecture [21]

tecture principles are a key artifact of the enterprise architecture and a major part of the case study in this master's thesis, a detailed definition is given in Section 4.2. An additional important part of enterprise architecture are patterns that can be applied to the management of enterprise architecture to document proven solutions to recurring problems in a given context [17]. The trade-off between specification and freedom is particularly important here. On the one hand, architectural development should not be hindered, but on the other hand, the number of technological inherited burdens should be reduced. In summary, enterprise architecture is a model of the enterprise's artifacts and their dependencies and aims to provide a picture of the holistic structure of an organization [34]. It serves as a basis for the application of principles that guide the design, engineering and implementation of a company. Overall, the process of creating the enterprise architecture with all its artifacts, models and objectives lies within the responsibility of the management discipline EAM [16]. That being said, enterprise architecture forms the core of EAM [52].

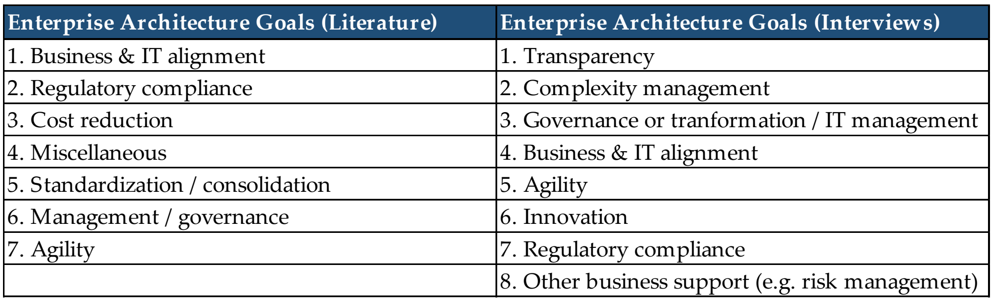
2.2.2. Enterprise Architecture Management

EAM is an essential part of IT management and includes all processes for documentation, analysis, quality assurance, planning, control of the further development of the IT landscape and the business architecture. It provides tools to master the complexity of the IT landscape and to further develop the latter in a targeted, business-oriented manner. EAM can be described as the process of creating and using enterprise architectures [106]. More precisely, EAM provides the content foundation for planning and controlling IT because in the enterprise architecture, a company's business and IT structures are collected and related to each other. Taking this into account, the diverse information needs of the various stakeholder groups can be satisfied. In addition, well-founded input can be provided for decisions [52]. This is referred to as a systematic and holistic approach to understand, communicate, design and plan the business and technical structures within the company. It makes it possible to master the complexity of the IT landscape as well as to develop the IT landscape strategically and in a business-oriented manner [2]. Furthermore, EAM creates transparency over the IT landscape with so-called business capability maps. It shows the interaction with the business architecture, promotes business IT alignment and supports the strategic and tactical planning and control of IT [114, 106].

Research primarily describes three important objectives of EAM as the following:

- IT Cost Reduction [14, 40, 42],
- Business and IT Alignment [14, 40, 41, 95, 106] and
- Reducing the Complexity of the IT Landscape [14, 53, 95].

In 2011, Lange and Mendling [80] identified further EAM objectives (see Table 2.1). Both literature research and semi-structured interviews with 16 industry experts were conducted. It should be noted that the most frequently mentioned objective is uppermost.



Enterprise Architecture Goals (Literature)	Enterprise Architecture Goals (Interviews)
1. Business & IT alignment	1. Transparency
2. Regulatory compliance	2. Complexity management
3. Cost reduction	3. Governance or tranformation / IT management
4. Miscellaneous	4. Business & IT alignment
5. Standardization / consolidation	5. Agility
6. Management / governance	6. Innovation
7. Agility	7. Regulatory compliance
	8. Other business support (e.g. risk management)

Table 2.1.: Enterprise architecture goals (literature and interviews)

The objective business and IT alignment is trying to align the business requirements with the underlying IT implementation [80]. Because of shorter response times and higher cost efficiency, it aims to provide a strategic competitive advantage [80, 101]. The cost reduction aims at reducing business process related and IT-related costs. To reduce the complexity of the enterprise architectures, standardization standardization and consolidation are important. This should lead to better decision-making, particularly in connection with management and governance [80]. Due to rapid change and competitive pressure, a quick adaptation to new market changes is important. This can be remedied with the help of an increased agility and thus a higher process and IT flexibility [80]. In addition, enterprise architecture targets support various other business functions, such as risk management in accordance with external regulations [80].

In order to achieve the goals mentioned above, EAM activities and procedures are necessary [149]. The entry point for new EAM activities and IT measures is requirements management [149]. New measures in particular must be recorded and documented in a consistent form [149]. Therefore EAM has to be integrated with other management functions, see Figure 2.9. Figure 2.9 shows processes for the management of application landscapes. EAM forms the bond for the cohesion with the other involved processes and thus represents an integrated overall process. Enterprise architecture makes it possible to recognize the interaction of different elements such as business processes, infrastructure elements, projects, IT architectures and strategies. With the help of enterprise architecture documentation the identification of correlations, implications and relations between the elements is subsequently possible. Hence, the EAM's task is to document, manage and control the interaction of these different elements on an architectural level. This is where requirements management takes effect. Figure 2.9 shows the interlocking of the requirements management with the IT project lifecycle (Define & document measure phase). Here, the goal is not to develop a comprehensive requirement analysis for a software requirement specification according to IEEE 830-1998 [29]. The goal is to prepare and facilitate the portfolio decision in the best possible way.

A requirement is documented in semi-structured text form. The documentation also requires the specification of a contact person and which strategies and goals are pursued or supported with the requirement. Thus, the understanding why an application system version has been developed and which goals were associated after creation of the project portfolio is already possible [149]. If the elements of the enterprise architecture that need to be changed are already known before a requirement is defined, this must also be documented directly. Through the documentation, searching for the same or similar associated architectural elements or goals is easily possible. In addition, bundles can be created. Ultimately, it depends on the decision of a committee (e.g. a steering committee). The committee decides whether a request is rejected, accepted or delegated back. If the request is accepted, the project portfolio management takes over the requirement [149].

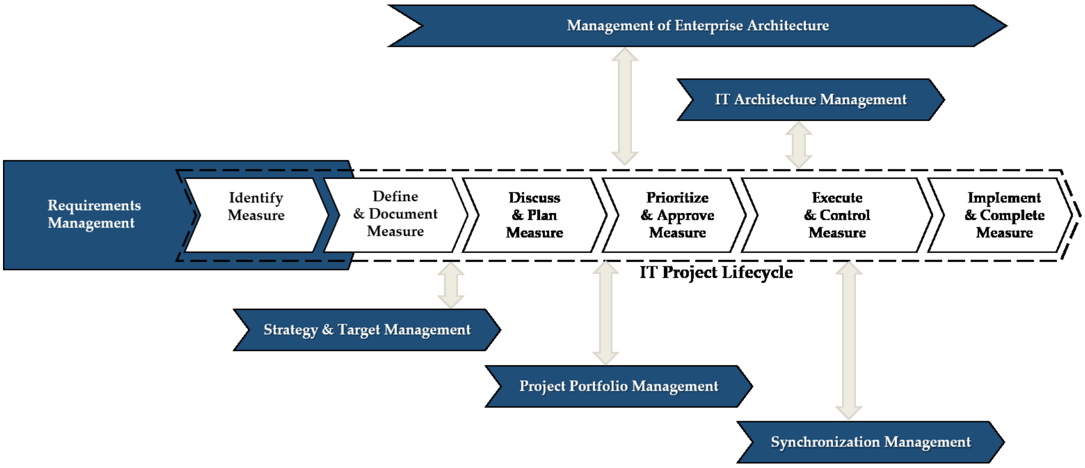

Management of Enterprise Architecture
IT Architecture Management
Requirements Management
Identify Measure
Define & Document Measure
Discuss & Plan Measure
Prioritize & Approve Measure
Execute & Control Measure
Implement & Complete Measure
IT Project Lifecycle
Strategy & Target Management
Project Portfolio Management
Synchronization Management

Figure 2.9.: Processes for the Management of Application Landscapes [149]

A new requirement is taken up and processed at the beginning of the requirement management phase. After the requirement has been properly documented, it will be evaluated uniformly as an IT measure. Once the committee has approved a requirement, a more detailed description is created, as the elements of the affected planned and target landscape are essential for the further processing of the project portfolio and synchronization management. With the help of the detailed description, a new project proposal is created in project portfolio management. This is then compared with other proposals from the project portfolio and at the same time included in the synchronization plan of synchronization management. Based on this proposal, a draft of the plan landscape is then created. After defining the IT project life-cycle, the measure must then be coordinated and planned (Coordinate Measure & Plan Phase). In addition, the management of the enterprise architecture must check whether this measure is compatible with the existing plan and target landscape. Thereby the goals of the target landscape and dependencies of the plan landscape are considered. Once an IT measure has been discussed and planned, the prioritization and approval phase begins (prioritize and approve measure phase). In the event of an approval by the project portfolio management, a new plan landscape is created. This is followed by the execute and control measures phase. First, a detailed project plan is created in this phase which includes a detailed synchronization plan. The IT architecture management supports the implementation of the measure through principles which, for example, have an influence on the software architecture. In order to be able to react to changes during the implementation, a continuous coordination between synchronization management and the actual measure takes place during the entire process. Finally, in the phase commissioning and complete measure, the changes generated by a measure are transferred from the planned landscape to the actual landscape.

To sum up, many scientists identified, that problems often arise during the introduction/implementation of EAM. Especially the process of achieving the above-mentioned objectives is difficult [52, 80]. The main problems include unenforceable requirements, lack of support by the leadership, incorrect focus or implementation stages that are too large and the resulting poor integration in the organization. This can be resolved by a systematic and step-by-step approach that takes best practices into account [52]. Even today there is still uncertainty about the way of measuring a successful enterprise architecture or a successful EAM. Scientists from the Technical University of Munich are providing an initial approach. They recommend to define so-called key performance indicators (KPIs), which enable EAs to plan, forecast, and evaluate enterprise architecture activities with regard to their contribution to target fulfillment [100]. In general, EAM can therefore be seen as an approach for creating, managing and using the artifacts provided by enterprise architecture [14, 106]. This means that EAM is the process of creating an enterprise architecture [106].

2.2.3. Enterprise Architecture Management in the Agile Environment

The EAM faces great challenges similar to software development due to the increasingly rapid and frequent changes in market conditions, technical and regulatory changes and the urgent need to reduce costs [7]. While agile principles and values are used in software development to counteract these challenges with greater flexibility and the ability to deliver results more quickly, EAM is still facing a radical change. However, EAM and agile development methods provide both decision frameworks, but differ fundamentally [56]. Where EAM, for example, focuses on a future vision for the entire company, the focus of agile development is on project implementation. Enterprise architecture is about where the company must stand tomorrow. This is precisely why a rethinking towards the ”Agile EAM” must take place. Especially in practice, agile principles and values are required to increase the efficiency of EAM [53]. While a few experts emphasize the mismatch between both disciplines [14, 105], the majority of scientists consider agile methods well suited for EAM [10, 53, 56]. In addition, the research shows, that there is no uniform definition of the term ”agile EAM”.

Scientists provide various principles and techniques to counteract the previously described challenges of the EAM and thus ensure greater agility of the EAM [10, 53, 56]. In contrast to the push principle (top-down), the pull principle (bottom-up) of Scrum as described in [47] should be applied in agile EAM. Thus, product development is not determined by irrelevant requirements of the team, but rather by the actually relevant requirements of the stakeholders [18]. This means that the enterprise architecture model should focus primarily on fulfilling the interests of the different stakeholders. To make this possible, a common vision between producer and consumer of the information is created on a daily basis [12]. In this case, especially wrong scoping is a big challenge. While agile methods rely on principles [47] from the Lean development of Toyota [110] to solve the problem, for example, an agile EAM should solve problems that arise during conception and planning through close daily cooperation and regular delivery of products [125] immediately. The latter also enables an early amortization of the EAM and confirms the importance of the agile EAM in the company [18]. Moreover, agile methods are based on commitment, focus, openness, respect and courage [125]. This is encouraged by the strict avoidance of disruptions during product creation [125]. An agile EAM should also pay strict attention to not disrupt teams during product creation, if possible. Through the large number of components involved and the associated network of dependencies a traditional EAM faces great administrative burdens. With the help of autonomous self-operated [110] and self-organized [12] teams, the administrative burden of an agile EAM can be reduced [18]. Furthermore, through the incremental and iterative character of agile development and, with that included, the possibility to react to changes quickly, an agile EAM is able to deliver different artifacts in less time [28].

In the following, agile EAM will be described by one of the most advanced approaches to date. Roth et al (2014) presents an iterative process of EAM [120] which will be used as the theoretical basis in this master's thesis.
Figure 2.10 shows the design of an organization-specific agile EAM practice based on theoretical concepts often found in current enterprise architecture frameworks which are extended by agile principles [120]. It thereby illustrates an iterative process to increase the agility of EAM.

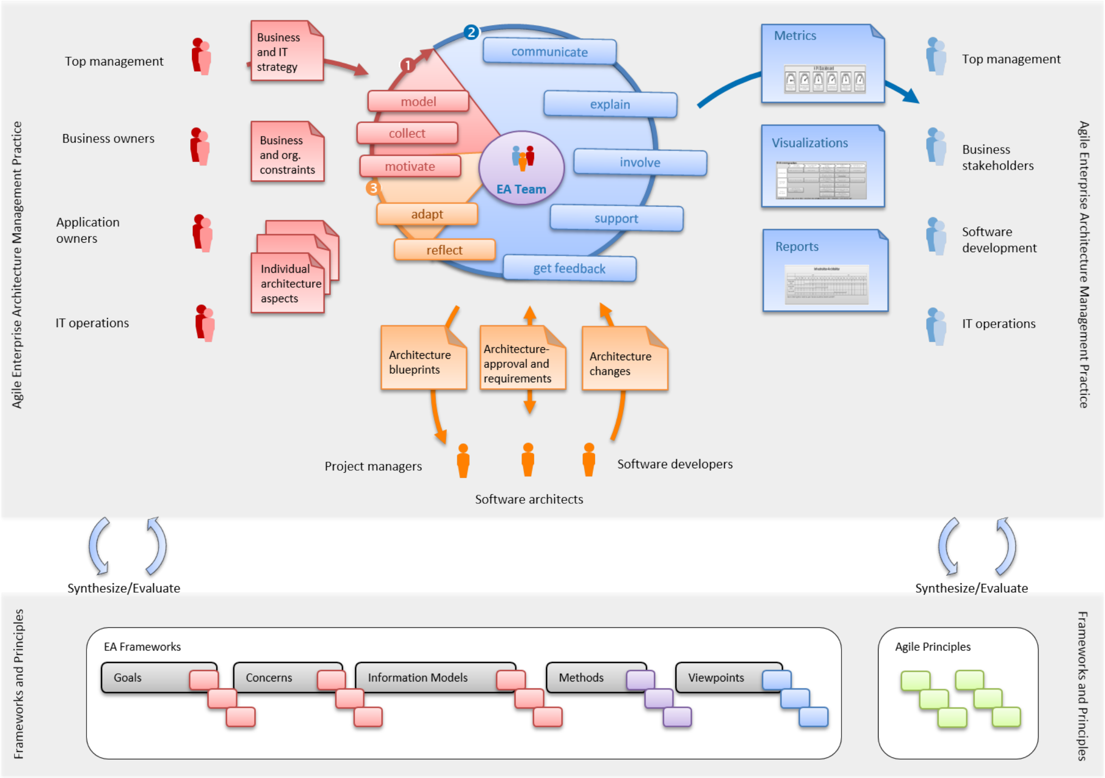


Figure 2.10.: Iterative Process of the EAM Function [120].

In the first step (model, collect, motivate), initial data is collected for the design of the enterprise architecture model. According to Roth et al. and Hanschke, social aspects such as support from top management and proactive stakeholder engagement are of great importance as a factor for successful EAM [6, 14, 53, 120]. Stakeholders should include not only business and application owners but also technical IT staff. For example, stakeholder groups such as top management and strategy office can define a business and IT strategy as the starting point for the EAM [120]. In addition, Roth et al. point out that the EAM often fails in practice due to over-modelling of the enterprise architecture. Therefore, the focus should be on important value-adding aspects [6, 120]. In the second step (communicate, explain, involve, support, get feedback), how the enterprise architecture team should explain the added value to the stakeholders is described. At this point,the involvement of stakeholders at all levels is important. This includes top management, business stakeholders, software developers, architects and project managers [120]. Since the EAM primarily pursues mid to long-term goals, the costs and benefits of the EAM should always be justified. This should be implemented with explicit communication and explanation of the enterprise architecture model to the entire organization. Artifacts such as reports, metrics and visualizations play an important role in this process. With the help of the enterprise architecture team, some

of the stakeholders might be able to solve their individual problems. As a result, collecting information might be justified without having to argue for it [120].
In the third step (adapt, reflect), the enterprise architecture team should analyze the feedback collected in step two and, based on this, adapt existing EAM functions or suggest changes to processes. Here, there is an option, for example, that the enterprise architecture team must first obtain the approval of the solution architects for architectural changes [120].

In summary, the aim of the agile EAM should be to focus on an iterative and incremental approach and also focus strongly on the needs of ATs and stakeholders [117]. Therefore, enterprise architecture should provide a common language for business and IT and be integrated with other management functions [120]. It must be visualized in such a way that makes it accessible to different stakeholders [101]. The development of a common infrastructure, common guidelines and valuable artifacts for the reuse of project teams would be a first starting point. Also, collaboration and communication should be in the center of action. Feedback cycles should come to the fore and should not be neglected. The exchange is the basic prerequisite for successfully managing a development. In addition, the role of the classical architect should be changed to a more advisory and supportive role [117]. Otherwise, EAs who are not actively involved in development teams run the risk of being perceived as ”ivory towers” and thus ignored [6, 148]. Another important point is concentration and providing the essential artifacts for both sides. Needless to say, a rough data model is of great importance for the AT, but at the beginning it would perhaps be more important to lead the team through all dependencies in detail and to show what experience has already been gained in this area [117].

2.2.4. Enterprise Architecture Management vs. Enterprise Architecture Management in the Agile Environment

EAM has proven to be an efficient instrument for the holistic alignment of business and IT in many organizations (see Section 2.2.2). Due to the increasingly rapid and frequent changes in market conditions, technical and regulatory changes and the urgent need to reduce costs, EAM and the way software is being developed must adapt to the changing environment and become more agile [7, 13]. Whereas in software development the use of agile methods such as Scrum and Extreme Programming is becoming more and more popular, EAM, on the other hand, is trying to become more flexible and collaborative [54]. Especially in practice, agile principles and values are required to increase the efficiency of EAM. So far, no scientific paper compares EAM with agile EAM. Since the focus on agile methods is the main difference between agile EAM and traditional EAM, the distinction will be based on the comparison between EAM and agile methods.

Figure 2.11 shows the tension between EAM and agile methods.

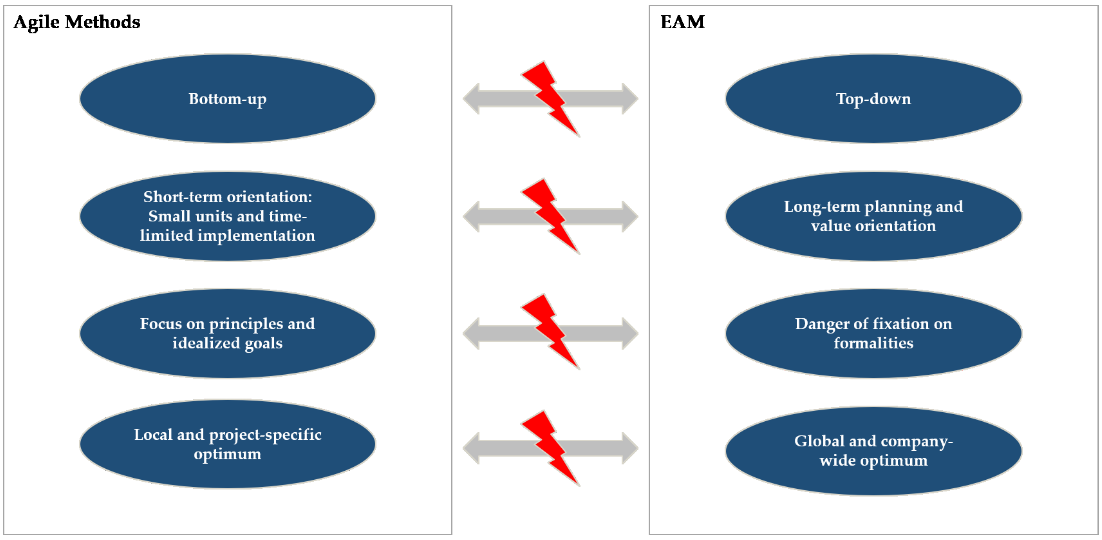

Agile Methods
Bottom-up
Short-term orientation: Small units and time-limited implementation
Focus on principles and idealized goals
Local and project-specific optimum
EAM
Top-down
Long-term planning and value orientation
Danger of fixation on formalities
Global and company-wide optimum

Figure 2.11.: Field of tension between EAM and Agile Methods

An important aspect of this tension is the difference between top-down architecture and bottom-up architecture. Top-down architecture, which is typically an enterprise architecture, works towards a long-term vision, while bottom-up architecture addresses questions that arise from projects - questions that should be answered quickly [56, 117]. The alignment of top-down and bottom-up architectures can improve and accelerate decision making at project levels that support the long-term vision. Important to note is that, in order to function properly, both architectural forms are required, top-down as well as bottom-up [56, 120].

In traditional EAM, the team usually starts with a business goal focusing on a global and enterprise wide optimum. The team plans for the long-run while aiming for value orientation [54]. Once the business goal is clear, the team must work out the requirements and changes that are necessary in the long-run (people, process, technology, application, data). Finally, the team has created an extremely specific plan about how things need to be done with very strict guidelines. At this point, the EAM runs the danger of fixation on formalities [14]. Once everything has been figured out, an implementation plan will be created. Eventually, this plan will be executed. The approach is top-down, structured and planned with a focus on the long term.

In agile, on the other hand, the business goal is in sight and the approach is always evolutionary. Agile methods focus on principles and idealized goals, using an iterative approach that takes one step at a time and always delivers working solutions [14]. In addition, there is always the possibility for reorientation and adaption. Whereas EAM focuses on a global and enterprise wide optimum, agile concentrates on a local and project specific optimum [54]. Since the environment can be defined freely, this often leads to an unclear understanding of roles [53]. An EA has the big picture in mind while a developer knows how to produce the software in detail and expects as few restrictions as possible [7]. This requires a rethinking of the role of the EA, the developer and other stakeholders at all levels of the company [145]. The aim should be to end the perception of architects as obstacles to development. Instead, the understanding should be promoted that the time it takes to deal with the architectural considerations of the company increases the overall value of a project for the organization [7]. Subsequently, the possibility to bring the added value of an EA's architectural decision to an AT, increases the understanding and leads to a higher intrinsic motivation of the AT [7].

2.3. Architecture Principles

In the following section, the status quo of research in the field of architecture principles is presented first. Secondly, a definition of architecture principles is given. Thirdly, the importance and added value of architecture principles in the context of EAM will be described. It should be noted that architectural principles and architectural guidelines are treated as synonyms in this master's thesis.

In literature, there are several different definitions of architecture principles [109, 132, 140, 151]. While questions of design representation such as meta-modelling and notation are intensively discussed in enterprise architecture, questions of design activity are often neglected here. Comparing the number of enterprise architecture publications with the number of principles show, that the latter is very limited. This is hard to imagine, because a variety of authors think that architectural principles are an essential and central element of enterprise architecture (cf. [8, 13, 24, 31, 50, 74, 104, 111, 112, 118, 127, 132, 139, 140, 141]) and describe them as the essence of architecture [33, 35, 61]. However, a precise definition of the concept of principles and the mechanisms and procedures needed to make them an effective regulatory instrument is still missing.

As early as 1990, Richardson et al. [118] provided a first definition of architectural principles. They defined them as philosophies that guide and govern the development of architecture [118]. This is similar to the definition given by Hoogervorst in 2004. Hoogervorst described architecture principles as "a consistent set of design principles and standards that guide design" [60]. However, the understanding of architectural principles is not consistent. Interestingly, a distinction between architecture principles and business and IT principles is made in most cases [48, 85, 123]. Greefhorst et al., Schekkerman und Lindström consider architecture principles as an integral part of architecture and describe business and IT principles as guidelines for managing IT. Among many other scientists, Op't Land and Proper go even further, and state that architecture principles are the key to ensure effective enterprise architecture [111]. Furthermore, the definition of architecture principles is part of many frameworks. Both TOGAF [140] and IEEE [109] describe principles as possibilities to guide the design and development of systems. They fill the gap between high-level requirements and concrete design decisions. In addition, they ensure that the enterprise architecture is future-oriented and that fundamental decisions are documented in a uniform language, thus considerably facilitating communication [109, 140]. Malan and Bredemeyer [94] take a different approach by focusing on the process of architectural development and discussing the three overall principles: ”Minimalist Architecture Principle”, Decisions With Teeth Principle” and ”Connect The Dots Principle” [94]. The description of the individual principles is interesting. While the ”Minimalist Architecture principle” states that a decision concerning the enterprise architecture should be delegated as far down as possible, the ”Decisions With Teeth Principle” defines that only rules and guidelines, which can and should be enforced in the enterprise, are included [94]. Finally, the ”Connect-the-Dots-Principle” states that an architectural decision must be based on a comprehensible relationship to the corporate strategy [94]. In this master's thesis, architectural principles are understood as the provision of a design space that ensures that companies develop their architecture within a specific framework [51, 60, 67, 111, 131, 137].

Enterprise architecture management (EAM) has become one of IT executives' top priorities [90]. According to ANSI/IEEE STD 1471-2000 principles are artifacts of the enterprise architecture that guide architecture's design and evolution [3, 36, 146]. Since EAM is the process of creating and using enterprise architecture [106] architecture principles are an essential and central element of enterprise architecture therefore of substantial importance for the EAM (cf. [8, 13, 24, 31, 50, 51, 74, 111, 112, 127, 139, 140, 141]). This is confirmed in particular by Weill and Ross [143] and Haki and Legner. The former call principles one of five key IT decisions that make IT a strategic asset in the context of IT governance. The latter conducted a study among experts to clarify how enterprise architecture principles can be turned into an effective means to shape the enterprise architecture. Based on the results of the study, Haki und Legner [51] confirm that enterprise architecture principles are an integral part of EAM and guide enterprise architecture design and evolution. The most important reason to use enterprise architecture principles is that they act as an enabler to reach enterprise architecture goals and benefits and keep consistency of the overarching architecture [51]. Furthermore, they close the gap between strategic goals and actual design decisions that have an impact on implementation [48], and, due to their ability to offer less room for interpretation and more insight into implementation and compliance, are considered to be the best governance instrument [49]. Principles focus on the most important requirements and enable companies to decide on what to design and govern in a top-down approach and what they would like to leave up to emergence. This is a major benefit, because it provides flexibility and therefore supports agility [48]. However, in order to achieve the added value mentioned, according to Greefhorst and Proper, both, the implementation of an effective process for adhering to the architecture that is carried out several times from start to finish of a project and the establishment of a general architectural governance framework which defines clear roles and responsibilities and how a communication via boards work are necessary [49].

3. Related Work

The literature research identified several publications in the field of agile and large-scale agile development that emphasize the importance of architecture and related governance mechanisms such as circles or architecture principles for agile enterprises. Often different approaches exist. The majority is concerned either with the agile creation, implementation and improvement of enterprise architecture or with the collaboration of EAs and agile software development teams. At the XP Conference 2010, the question *"Architecture and Agility – How Much Design Is Sufficient for Different Problem Classes?"* was raised. It was rated by scientists as the second most burning question [38]. This emphasizes the relevance of this research area [135].

Ambler [6] discusses typical problems with enterprise architecture approaches in practice. Among other things, he addresses the problem of a lack of awareness of the EAM. According to Amber, project teams, for example, do not know that there is an enterprise architecture and think that this is an outdated concept [6]. In addition, Ambler [6] introduces an agile model driven development approach (AMDD) at the enterprise level. This iterative approach starts with an architectural vision. The vision is then communicated to all architecture stakeholders and developers. After successful communication, feedback is obtained and the architectural vision is adapted. However, according to Ambler, AMDD does not provide a clear time frame for the results of the iterative cycles, nor are there approaches to stakeholder engagement [6].

Rhubart [117] states that enterprise architecture and agile development can coexist. Enterprise architecture (typically top-down driven architecture) and agile development (typically bottom-up driven) are both decision making frameworks and therefore share common ground. Rhubart states, that in order to align top-down and bottom-up architecture, getting the necessary buy-in from architects, developers, and other stakeholders at all levels of the organization is mandatory. Next to the importance of conversation, in particular with the developing teams, the involvement of enterprise architects at the project level is considered to be very crucial. In addition, Rhubart states that the awareness must change with respect to architects. Architects should no longer be perceived as obstacles. The understanding should be promoted that a certain amount of time is needed to deal with architectural decisions for the company in order to increase the overall value of a project [117]. According to Rhubart, the gap between enterprise architecture and agile development can be reduced if both are seen as different perspectives on the same goal [117].

Friedrichsen and Schrewe [39] recommend more agility. They point to typical EAM problems such as the danger of defining formalities or the loss of the overall picture [14]. In order not to limit agility too much, clear goals, risks and limited room for maneuver should be in the foreground. Governance mechanisms such as frameworks and tools should be used to ensure that goals can be achieved more efficiently [39].

Buckl et al. [18] focus on the type of combination between agile and enterprise architecture, i.e. how to work agilely with architecture. In doing so, they mention increasing complexity due to an increasing number of components and their dependencies as well as the resulting challenge for a successful EAM in practice as the cause for the rethinking and adaptation of agile methodologies from software development. More precisely, they name four major challenges. First, the interests of stakeholders must be aligned with the EAM and defined in a common terminology. Second, the EAM must ensure early and consistent delivery of concrete enterprise architecture products. Thirdly, the commitment and participation of all stakeholders must be ensured by the EAM. Finally, the EAM must continuously adapt to the changing environment. To meet these challenges, the researchers describe how agile EAM can be implemented based on SCRUM. To achieve this, events, artifacts and roles are related and adapted to the EAM. For example, the role of the product owner (PO) is changed. According to Buckl et al. [18], the PO must be a person with knowledge of both the EAM and of organizational units and, accordingly, refines and prioritizes the product backlog in close cooperation with the enterprise architecture stakeholders. The PO is the only point that ensures that the needs of all stakeholders are met. After mapping all SCRUM concepts (events, artifacts and roles) to their counterparts in EAM, they come to the conclusion that EAM can be performed in a more agile and lightweight manner [18]. Since this is approach has not been evaluated so far, the research is still in progress.

Bente et al. [14] focus on the application of agile and lean methods in enterprise architecture. They propose six building blocks that can be used to promote agility. The authors offer an approach on how to build an agile project, promote collaboration and participation, and streamline architectural processes [14]. The approach includes fictitious examples without quantitative results as a basis.

Bachmann et al. [9] and Nord et al. [108] pursue a holistic approach. They present four architectural tactics that support the design of system architecture, organizational structure, and product infrastructure for agility. The tactics include various options that can be used depending on the phase of a system. These include vertical and horizontal system decomposition, matrix and extended team structures, architecture and infrastructure runway, and deployment strategies. The authors suggest, for example, that a team should be organized horizontally at the beginning of development (horizontal system decomposition). This is justified by the fact that teams must ensure a stable infrastructure at the beginning. This involves setting up the build and deployment infrastructure, building architectural elements, deciding on tools and preparing a rough architectural sketch (architecture and infrastructure runway). Only when this phase has been completed and the most important interfaces have been defined may some team members begin developing features (matrix organization). As soon as the interfaces are more stable, a vertical feature-oriented development can be started (vertical system decomposition) [9, 108].

Roth et al. [101] describe an iterative process for the EAM function that consists of three main steps. These include EAM, generated artifacts and relevant stakeholders and the relationship between research and practice. In the first step (model, collect, motivate) initial data are collected for the design of the enterprise architecture model [101]. The second step (communicate, explain, involve, support, receive feedback) describes how the added value should be explained by the enterprise architecture team to the stakeholders involved [101]. In the last step (adapt, reflect), the feedback collected in step two will be analyzed by the enterprise architecture TEAM, and based on this, existing EAM functions or processes will be adapted [101].

Hensema [58] validates that agile principles have a positive impact on EA. Further she states that numerous EA challenges can be related to stakeholders. In addition, it was found that the development of the enterprise architecture is strongly oriented to the waterfall principles and therefore could benefit from the agile method of software development [58].

Hanschke et al. [54] shows a new holistic approach to combine EAM and agile methodology. Their findings are based on a survey by a consulting firm, a railway company and an automobile manufacturer. The focus was on Scrum and the necessary cooperation of competent implementation teams with a central enterprise architecture function. She believes that EAM and agile software development can complement each other, on the one hand through the application of agile methods in the creation of the enterprise architecture, and on the other hand through the stronger focus on collaboration between EAs and agile development teams" [54]. This could give the EAM more flexibility and the ability to cooperate, while agile methodologies benefit from a stronger goal orientation [54].

Uludağ et al. [136] conducted an explorative case study. It describes how the introduction of domain-driven design can support a large scale agile development program with three agile teams. He notes that the development speed can be increased by balancing emergent and intended architecture. In addition, the relationship between centralized and decentralized architectural decisions must be found.

Canat et al. [25] share Hanschke's opinion [54] that EAM and agile software development can complement each other. In their paper "Enterprise Architecture and Agile Development" they point out that reducing the distance between developers and architects in order to improve collaboration is. Since enterprise architecture focuses on both business and IT, enterprise architecture models also have to contain accurate information about the software portfolio and the dependencies between software and infrastructure.

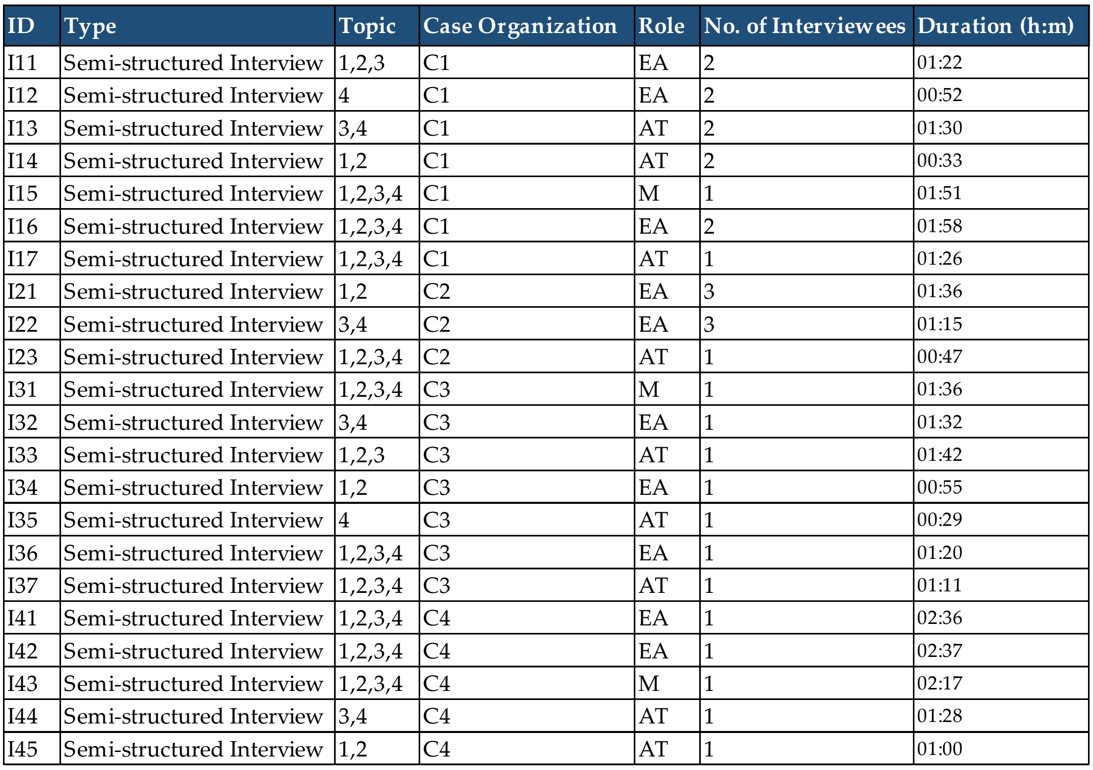
Niemann [107] describes possible solution approaches and a possible development of architecture management on the basis of current deficits of architecture management organizations, derived from a study carried out specifically for this purpose. In the study, architecture management organizations were asked about their current self-assessment. Niemann [107] particularly emphasized 11 factors that influence digitization. He divided them into internal and external factors and acceptance factors. Among the internal influencing factors are, for example, the range of services offered by the EAM to the customer, EAM tools, contents of the enterprise architecture repository, organization of the EAM, communication and feedback as well as the architects' self-image. External factors include the importance of digitization for the business, the mandate and authority of the EAM, and identification and knowledge of customers. The acceptance factors include channels such as process interfaces and committees and the value proposition of the EAM. Niemann [107] states that the organizations surveyed have insufficient knowledge about their customers and methods for collecting and modeling business motivation. Furthermore, there is a lack of support for digitization projects as well as a lack of an adequate feedback culture. Based on the deficits listed above, Niemann defines solution approaches in the areas of qualification, customer interface, operational architecture management and feedback culture [107]. The author uses the abbreviation VUCA, which is a combination of the terms Vision, Understanding, Clarity and Agility [73]. A vision can provide orientation in a highly volatile environment, understanding can help to deal with uncertainty, structural clarity helps to deal with complexity and agility helps us to deal with changing conditions and values. Niemann comes to the conclusion that the future enterprise architecture will focus on effectiveness and added value and will act in a solution-oriented manner [107]. In addition, he describes the enabling of teams, the setting of guidelines and the orientation towards customer feedback as essential. Furthermore, the new enterprise architecture provides both structure and methodology for digital transformation, guarantees feasibility of solutions and actively identifies risks and potentials [107]. Finally, the future enterprise architecture should be firmly anchored in the company and actively involved in teams. In addition, personal responsibility and self-organization as well as short feedback cycles should be established.

4. Case Study

This chapter includes the results of the case study starting with background information on the agile transformation of each case organization in Section 4.1. In general, the case study is divided in four parts, described in Sections 4.2, 4.3, 4.4, and 4.5. For each of the four parts, the questions provided to the interviewees can be viewed in Appendix A.1.

4.1. Case Description

The results of this master's thesis are based on a multiple embedded case study in cooperation with four leading German companies from the automotive, information technology, insurance and retail sector in 2018. Twenty-one persons from the three stakeholder groups AT, EA and M were interviewed (see Table 4.1). A total of 21 interviews were conducted on the four topic blocks architecture principles, architecture boards, role and value contribution of EAM. Throughout this scientific work, the companies are named "case organizations" and abbreviated as C1, C2, C3, and C4 in the following tables.



ID	Type	Topic	Case Organization	Role	No. of Interviewees	Duration (h:m)
I11	Semi-structured Interview	1,2,3	C1	EA	2	01:22
I12	Semi-structured Interview	4	C1	EA	2	00:52
I13	Semi-structured Interview	3,4	C1	AT	2	01:30
I14	Semi-structured Interview	1,2	C1	AT	2	00:33
I15	Semi-structured Interview	1,2,3,4	C1	M	1	01:51
I16	Semi-structured Interview	1,2,3,4	C1	EA	2	01:58
I17	Semi-structured Interview	1,2,3,4	C1	AT	1	01:26
I21	Semi-structured Interview	1,2	C2	EA	3	01:36
I22	Semi-structured Interview	3,4	C2	EA	3	01:15
I23	Semi-structured Interview	1,2,3,4	C2	AT	1	00:47
I31	Semi-structured Interview	1,2,3,4	C3	M	1	01:36
I32	Semi-structured Interview	3,4	C3	EA	1	01:32
I33	Semi-structured Interview	1,2,3	C3	AT	1	01:42
I34	Semi-structured Interview	1,2	C3	EA	1	00:55
I35	Semi-structured Interview	4	C3	AT	1	00:29
I36	Semi-structured Interview	1,2,3,4	C3	EA	1	01:20
I37	Semi-structured Interview	1,2,3,4	C3	AT	1	01:11
I41	Semi-structured Interview	1,2,3,4	C4	EA	1	02:36
I42	Semi-structured Interview	1,2,3,4	C4	EA	1	02:37
I43	Semi-structured Interview	1,2,3,4	C4	M	1	02:17
I44	Semi-structured Interview	3,4	C4	AT	1	01:28
I45	Semi-structured Interview	1,2	C4	AT	1	01:00

Table 4.1.: Details of interviews and observations

4.1.1. Background Information about Case Organizations

Case organization 1:
The reasons for the transformation towards an agile EAM are the stuck storage formation, the rejection of concepts and the too low implementation speed of models and solutions in case organization 1. This results in the following objectives, which are aimed at with the transformation: Increase of the implementation speed and establishment of an end to end responsibility. The transformation initiated by IT at the beginning of 2017 initially began with a six-month pilot project. Since mid-2017, the entire organization has been oriented towards agile methods. At the beginning of 2018, a strong business & IT alignment was strived for. The transformation receives support from the top management and the executive board (EB) and is strongly driven top-down. To support and establish agility, extensive company-wide communication measures and agile methods / frameworks based on various standards have been introduced. Among others, SAFe, Scrum and LeSS are used for this purpose. There is no company-wide rollout of a standard. Today, both a new role model and a redefinition of IT processes are among the successes. In addition, IT budgeting is completely product-oriented. However, the problem that IT is perceived more as a service provider with a strong cost orientation still exists today. This problem will be addressed with increased commitment on the part of management levels. The assignment of all roles in all product domains and a transition to company-wide and consistent application of agile methods is among the steps still to be implemented during the transformation. In addition, existing processes must be improved. In principle, a slowdown of the transformation is perceived. Due to the change, there is a risk of a reduction of the implementation speed in the products. The question about the extent to which the EAM currently enables the scaling of agile practices was rated 4 on a scale of 1 ″no enablement at all″ to 10 ″very strong enablement″.

Case organization 2:
In case organization 2, the reasons for the transformation are changes in the market conditions as well as state-of-the-art development processes in order to win new talents for the enterprise. This results in the following goal: Short reaction times, even with major changes in the product portfolio. The change/transformation initiated by a central transformation team at the beginning of 2015 initially began with the introduction of agile methods. This was followed in 2016 by the switch to agile corporate and coordination processes. In order to support and establish agility, extensive company-wide communication measures and agile methods / frameworks were introduced which are based on various standards. In the beginning, scaling was strongly supported by SAFe. Gradually, more elements from LeSS were introduced. An attempt is now being made to find a suitable solution for the company. The current status includes both a conversion of portfolio planning from annual planning to 3-month planning as well as a separation of management roles into process, topic and personnel management. However, the challenge still exists today to convince adjacent areas such as product management or data center operation of the agile transition. This would also lead to a willingness

to change in adjacent fields. On the one hand, the challenge must be met by using the EAM to develop meaningful portfolio decision alternatives both in the as-is and the to-be. On the other hand, it must be met through cooperation in setting up continuous project initiation processes and the structured linking of project plans to a strategic level. The establishment of initiated changes in adjacent areas, but also in personnel development, are among the steps still to be implemented during the transformation. In addition, the scope of the EAM must be increased. Coverage of all projects instead of individual projects is desired. In principle, the absence of the desired effects of agility is considered to be a possible risk. The question about the extent to which the EAM currently enables the scaling of agile practices was rated 8 on a scale of 1 ″no enablement at all″ to 10 ″very strong enablement″.

Case organization 3:
In case organization 3, the reasons for the transformation towards an agile EAM are the lack of flexibility and speed of adaptation to changing conditions, lack of business and IT alignment and high costs. This results in the following objectives, which are aimed at with the transformation: Faster time-to-market in IT product development, increased flexibility and adaptation speed, higher customer satisfaction, an improvement of the collaboration model between business and IT as well as a cost reduction in the long term. The transformation initiated by IT in October 2017 initially began with the design of an ideal product domain map to dissolve the existing silo organization. Since then, six different work streams for the transformation have taken place. They were driven top-down by CEO, COO, and CTO. The focus was on the renewal of the architecture and transformation of the organization in the direction of product organization. In order to support and establish agility, extensive company-wide communication measures and agile methods / frameworks based on various standards were introduced. Among others, SAFe, Scrum, and LeSS are used for this purpose. Today, about twenty enabled product teams working according to the new architecture principles and shifting the hosting model from on-premise to cloud are among the successes. The latter already makes it possible to deploy and operate the first applications and services on the cloud. However, even today adapting the minimum number of employees and ensuring the necessary communication is still a challenge. This challenge must be met with the help of coaching and communication workstreams. The next steps in the transformation include coaching and supporting existing product teams, developing and communicating How-To Guides to relieve the EA, and implementing the API-first architecture. Basically, missing skills, low speed, and competing projects are perceived as possible stumbling blocks. The question about the extent to which the EAM currently enables the scaling of agile practices was rated 6 on a scale of 1 ″no enablement at all″ to 10 ″very strong enablement″.

Case organization 4:
In case organization 4, the reasons for the transformation to an agile EAM are the changes in the business model, the impact of digitization and the lack of software de-

velopment flexibility. This results in the following objectives, which are the goal of the transformation: Improvement of the time-to-market, improvement of effectiveness and efficiency and an increase of customer satisfaction as well as the increase of flexibility in software development. The transformation initiated by IT in 2016 began with first attempts at the project level. At first, a large project was involved. After success, it was extended with another framework to another large project. At the same time, a main department was reorganized to agile structures. In order to support and establish agility, extensive company-wide communication measures and agile methods / frameworks based on various standards were introduced. Among others, SAFe, Scrum, eScrum, Kanban, and Nexus are used. There is no company-wide roll-out of a standard. Today, so-called ″dos and don′ts″ such as consistent compliance with agile principles are among the achievements. However, some challenges still exist today. These include change management at all hierarchical levels, the parallelism of agile and non-agile working methods and organizational forms as well as the consistent implementation of frameworks and agile methodologies. These challenges must be met with the help of training courses, reorganizations and role definitions. The further steps to be implemented in the course of the transformation include professionalization in the affected areas as well as the conversion of further organizational units. In principle, training, reorganization and role definitions are perceived as possible slowdowns in transformation. The question about the extent to which the EAM currently enables the scaling of agile practices was rated 8 on a scale of 1 ″no enablement at all″ to 10 ″very strong enablement″.

4.2. Architecture Principles

`How is your EAM organization structured? Which architectural roles do you have and how are they assigned to the ATs?`

Case organization 1:
According to the interviewees, the EAM organization of case organization 1 is hierarchically structured. Starting with the main departments, the structure extends to the departments and the sub departments (see Figure 4.1). In addition, there are five EAM development circles in total, which consist of one management circle and four subordinate circles (see Figure 4.1). Further circles exist below the four subordinate circles. A circle can either be a center of excellence (CoE) or a community of practice (CoP). Each circle consists of various representatives of subprojects and one enterprise architect governance (EAG). In Figure 4.1, the green person represents an EAG and the red person is head of the EAM. Generally, no strict responsibilities are assigned in order to create a group as homogeneous as possible. Decisions are primarily made according to the subsidiarity principle.

> *"As much as possible is delegated downwards. 80-90% takes place on the lower levels, the rest are decisions that have overlapping effects and are therefore discussed in the main circle (MC)"*
>
> *-I15, Head of EAM, M; C1*

However, if the issues are very critical or cross-departmental, the main circle (MC) is always involved. According to the interviewees, there are currently three different architect roles, specifically EAs, enterprise architects governance (EAG) and solution architects (SA). EAs are assigned to topic clusters and thus fulfill the role of an overlapping architect. EAG fulfill the role of an overlapping architect as well but are mainly responsible for the EAM methodology. SAs, also known as IT architects, are part of the component teams.

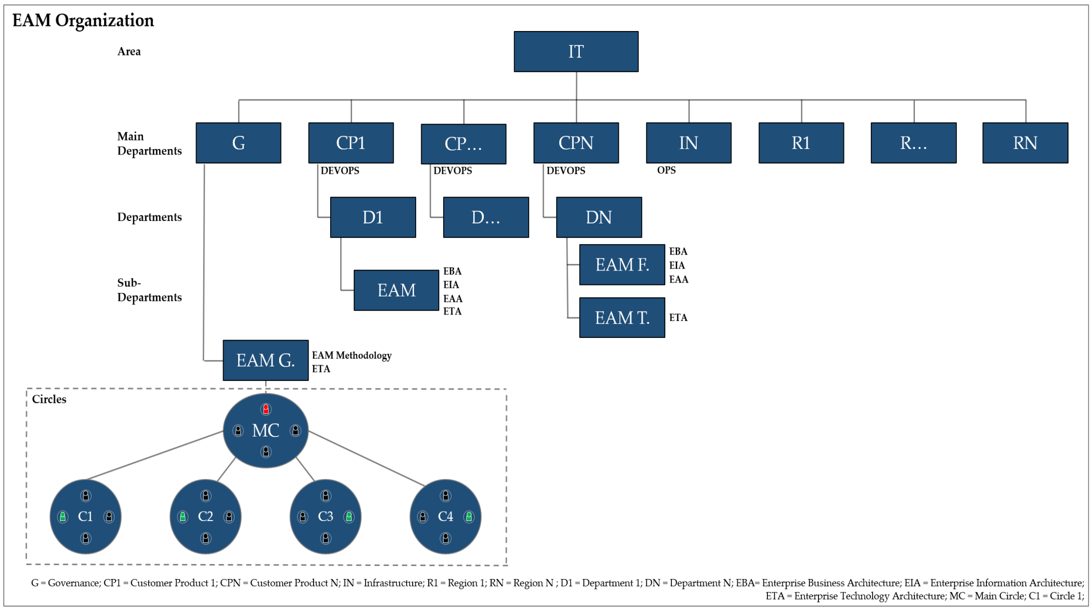
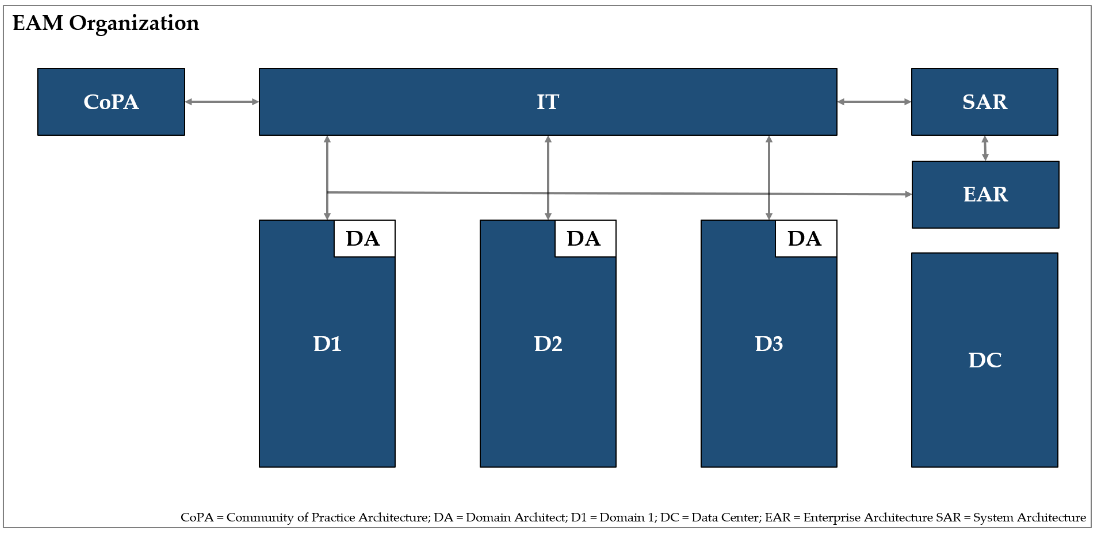

EAM Organization
Area
IT
Main Departments
G
CP1
CP...
CPN
IN
R1
R...
RN
DEVOPS
DEVOPS
DEVOPS
OPS
Departments
D1
D...
DN
Sub-Departments
EAM
EBA
EIA
EAA
ETA
EAM F.
EBA
EIA
EAA
EAM T.
ETA
EAM G.
EAM Methodology
ETA
Circles
MC
C1
C2
C3
C4
G = Governance; CP1 = Customer Product 1; CPN = Customer Product N; IN = Infrastructure; R1 = Region 1; RN = Region N ; D1 = Department 1; DN = Department N; EBA= Enterprise Business Architecture; EIA = Enterprise Information Architecture;
ETA = Enterprise Technology Architecture; MC = Main Circle; C1 = Circle 1;

Figure 4.1.: Company 1 - Structure of EAM Organization

Case organization 2:
According to the interviewees, the EAM organization of case organization 2 is hierarchically structured (see Figure 4.2). The EAM organization is roughly divided into three main domains (D1, D2, D3). Currently three different roles of an architect, specifically EA, Domain Architect (DA) and SA, are defined. Each domain has a DA. This is the main contact person for the EAM and is the interface between EAs and SAs. The latter are part of the teams and work closely together with the developers and DAs. In addition, the areas enterprise architecture (EAR) and system architecture (SAR) exist. EAR architecture entails the whole organization and has a focus on strategy whereas system architecture entails one or more systems of the organization and focuses rather on the

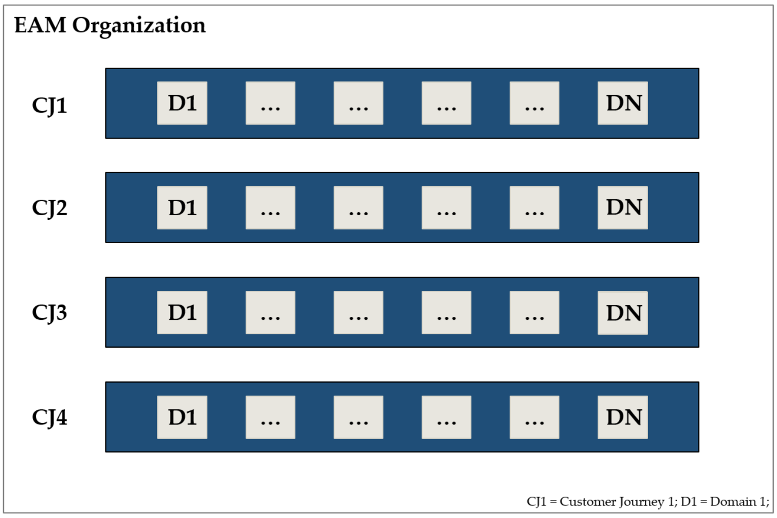
operation. According to the interviewees, there is a close exchange between SAR, EAR, IT and the individual domains. The IT fulfills the function of a coordination interface between the individual areas and is in continuous exchange with the community of practice architecture (CoPA).



CoPA = Community of Practice Architecture; DA = Domain Architect; D1 = Domain 1; DC = Data Center; EAR = Enterprise Architecture SAR = System Architecture

Figure 4.2.: Company 2 - Structure of EAM Organization

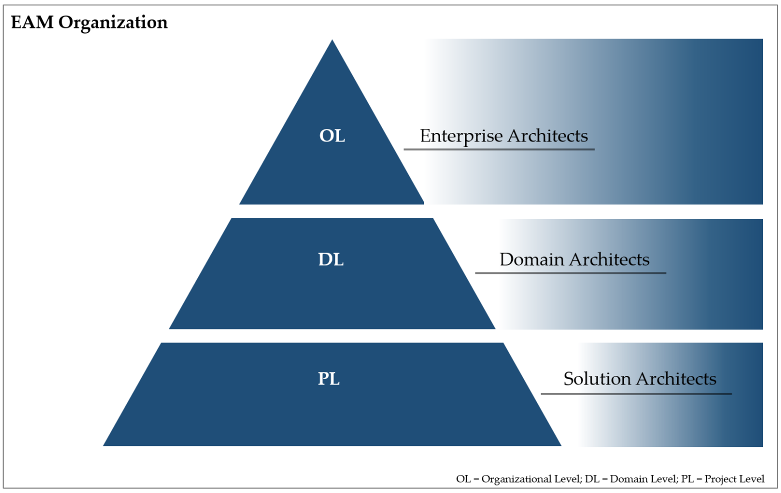
Case organization 3:
According to the interviewees, the EAM organization of case organization 3 is divided into four customer journeys (CJ1, CJ2, CJ3, CJ4) (see Figure 4.3). A customer journey contains several domains. One domain contains 1-n products. A product is only assigned to one domain. A domain is also referred to as a product cluster. Currently there are three different architect roles, specifically EA, DA, and SA. There is one EA per CJ, which means that one EA is responsible for several domains. Within a domain (or product cluster) one DA exists, but not every domain has a DA. The same applies to the SAs within the ATs. It should also be noted that the SA is not subordinate to the DA.



CJ1 = Customer Journey 1; D1 = Domain 1;

Figure 4.3.: Company 3 - Structure of EAM Organization

Case organization 4:
The EAM - Organization of case organization 4 is organized by divisions (see Figure 4.4). The goal is to build up the individual areas according to domains in order to break up the silos. According to the interviewees, there are currently three different architect roles, specifically EA, DA and SA.



OL = Organizational Level; DL = Domain Level; PL = Project Level

Figure 4.4.: Company 4 - Structure of EAM Organization

As shown in Figure 4.4, each role is broadly assigned to an enterprise level.

In addition, there are five architectural levels: strategy, organization, IT application, information and infrastructure / technology. According to the interviewees every architect should have knowledge of all five architecture levels on the one hand and should be an expert in his special field at the same time. The role of the DA is filled by cross-divisional sales and cross-sectional architects.

Which drivers are significantly responsible for the creation of architectural principles (Examples: regulatory requirements, digitization, technological innovations, changing business models, software quality)?

In the following, the driving forces responsible for creating architectural principles are listed in Table 4.2. They are sorted in descending order by relevance.

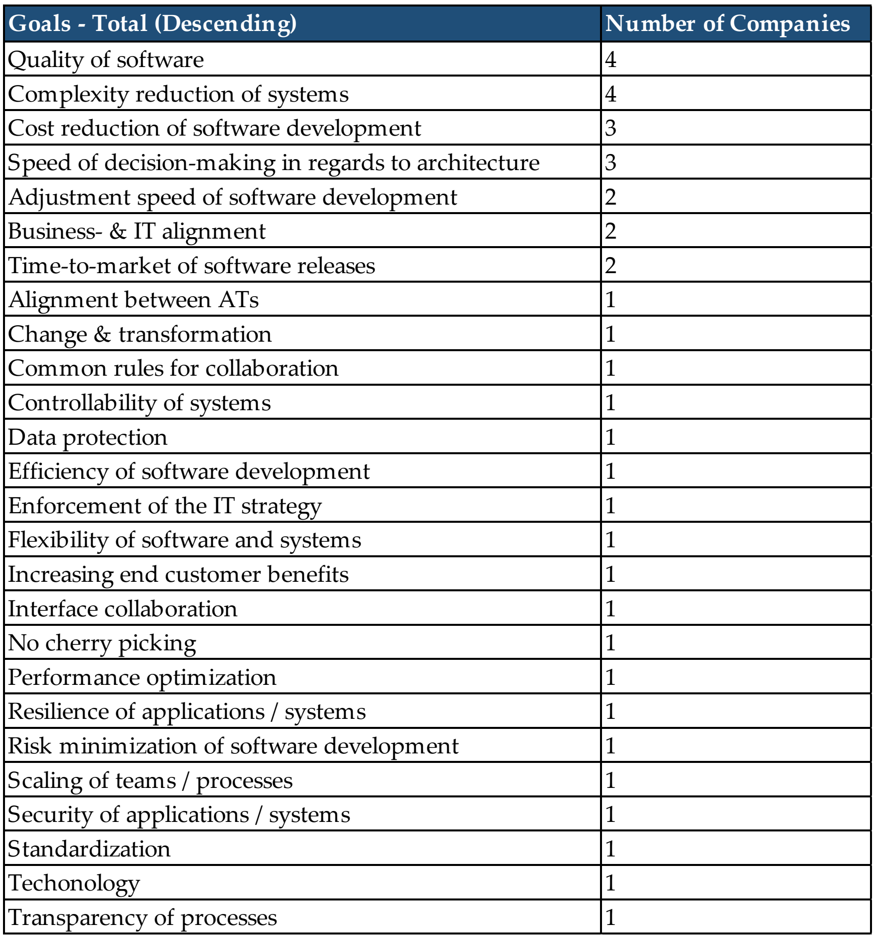
Drivers - Total (Descending)	Number of Companies	AT	EA	M
Changing business models	4	3	0	1
Compliance and regulatory specification	4	2	4	1
Technology advancements	3	1	1	1
Digital transformation	3	1	2	0
Drive for innovation	3	3	2	1
Competitiveness	2	1	0	0
Market change	2	2	0	0
Scalability, flexibility, efficiency of software development	2	0	1	1
Security of software	2	0	1	1
Change in customer behaviour	1	2	4	2
Complexity through globalization	1	0	1	0
Common understanding -> common language	1	2	2	0
IT strategy	1	0	2	0
Software quality	1	1	1	0

Table 4.2.: Drivers across all case organizations

As shown in Table 4.2, 100% of the case organizations name "Changing business models" and "Compliance and regulatory specifications" as important driving forces. 75% name "Technological advancements", "Digital transformation" and "Drive for innovation" as driving forces for creating architecture principles.

What are your goals in defining architectural principles?

Table 4.3 shows the goals pursued with the definition of architectural principles.



Goals - Total (Descending)	Number of Companies
Quality of software	4
Complexity reduction of systems	4
Cost reduction of software development	3
Speed of decision-making in regards to architecture	3
Adjustment speed of software development	2
Business- & IT alignment	2
Time-to-market of software releases	2
Alignment between ATs	1
Change & transformation	1
Common rules for collaboration	1
Controllability of systems	1
Data protection	1
Efficiency of software development	1
Enforcement of the IT strategy	1
Flexibility of software and systems	1
Increasing end customer benefits	1
Interface collaboration	1
No cherry picking	1
Performance optimization	1
Resilience of applications / systems	1
Risk minimization of software development	1
Scaling of teams / processes	1
Security of applications / systems	1
Standardization	1
Techonology	1
Transparency of processes	1

Table 4.3.: Goals across all case organizations

As shown in Table 4.3, 100% of case organizations name "Quality of software" and "Complexity reduction of systems" as the primary goal of defining architectural principles. 75% of the case organizations name "Cost reduction of software development" and "Speed of decision-making in regards to architecture" as goals for creating architecture principles.

Which architectural principles do you use in an agile environment? Which architectural principles do you use from the attached Excel list (Please specify them in the Excel list)?

For the selection of architectural principles, comprehensive literature research was carried out at the beginning of this master's thesis. Following a structured literature review, 127 architectural principles were identified [48, 64, 76, 130, 140, 144]. Subsequently, these were filtered according to the following criteria in collaboration with a scientist from the Chair of Software Engineering for Business Information Systems (sebis). Criteria for the inclusion or exclusion of architectural principles were:

Inclusion:

1. Support of agile software architecture and development
2. Impact on:
   a) architecture design **or**
   b) organization of teams **or**
   c) software development **or**
   d) support of agile values and principles

Exclusion:

1. Must not contradict agile values and principles (and name them concretely) **or**
2. Slightly relevant to architecture **or** development **or**
3. Specification too abstract

Based on this approach, 30 of 127 architectural principles were considered suitable for the agile environment and used as a basis in the interview question.

Figure 4.5 shows the implementation of architecture principles across all case organizations sorted by relevance in descending order. The color represents the implementation stage of architecture principles. The number inside of the line shows how many case organizations share the same state. The implementation stage is sectioned into fully implemented (color: green), partly implemented (color: blue), planned (color: grey), and not planned (color: orange).

For example, the architecture principle "Strictly separate build and run stages" is fully implemented by two case organizations, one case organization plans to implement it and one has not planned it yet.

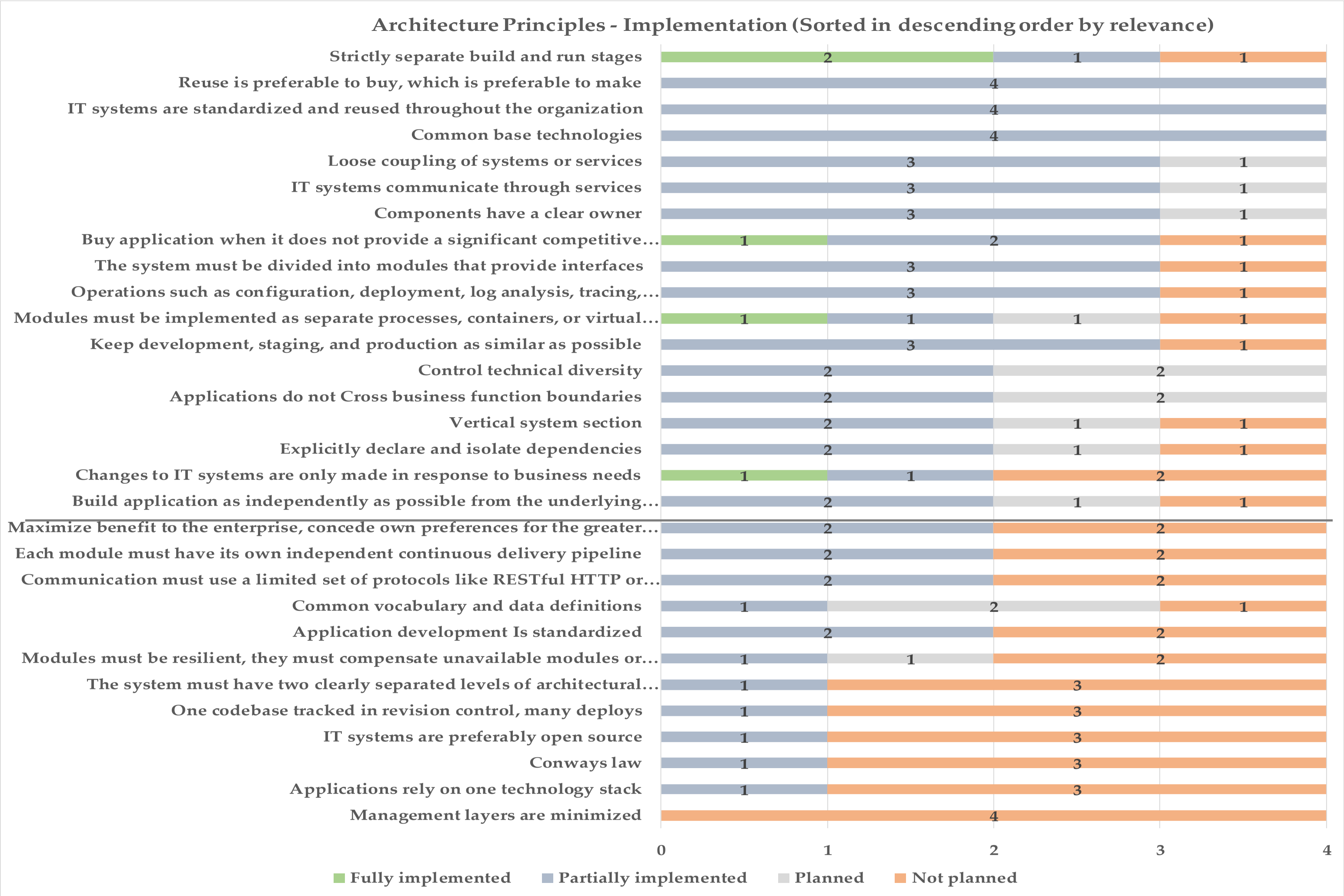
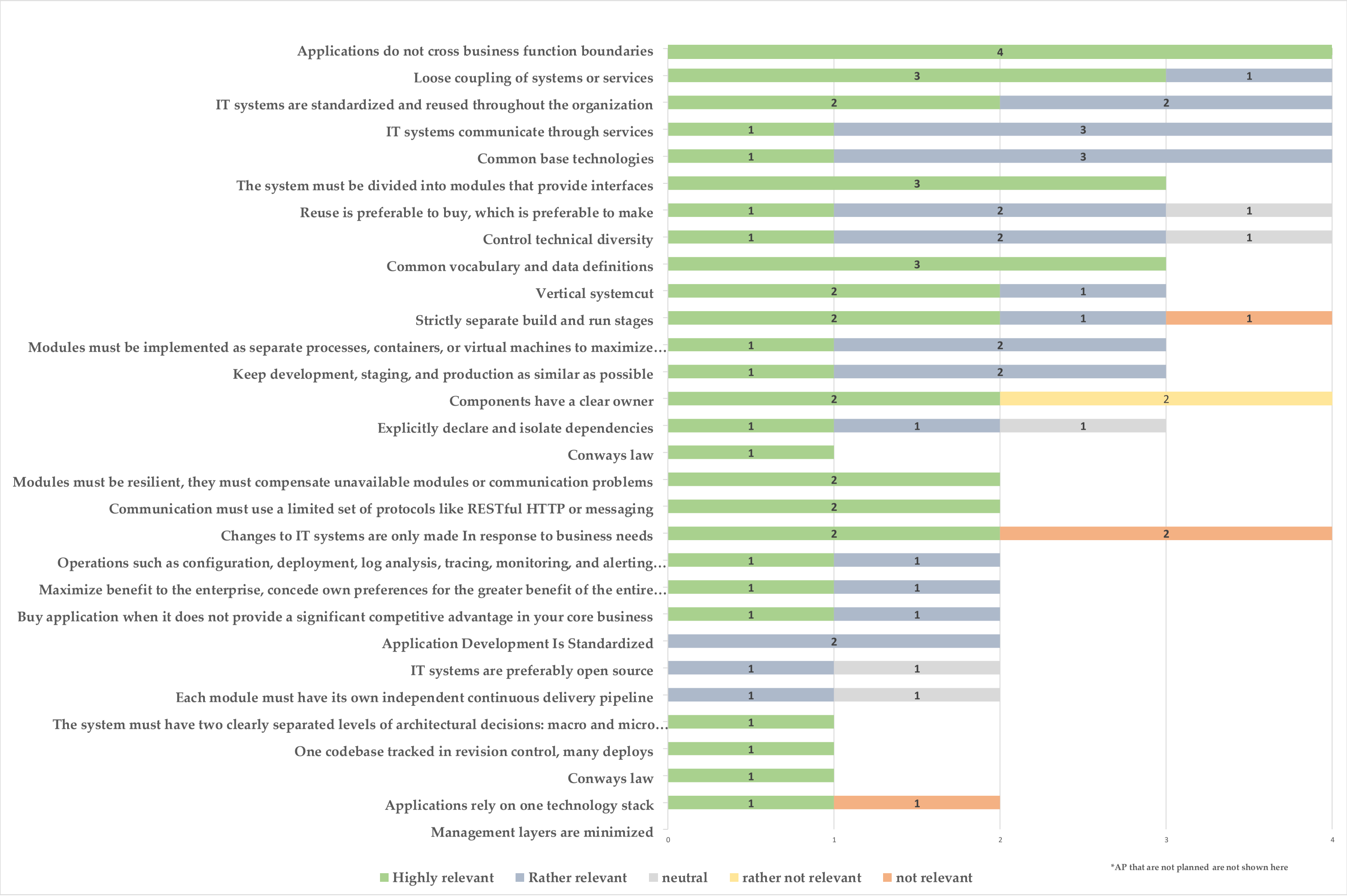


Figure 4.5.: Architecture Principles - Implementation across all case organizations

As shown in Figure 4.5, there are 12 architecture principles that can be observed repeatedly in three case organizations. This is indicated by the grey line in the middle of the diagram.

Figure 4.6 shows the importance of architecture principles across all case study organizations sorted by relevance in descending order. The color represents how relevant an architecture principle is. The number inside of the line shows how many case organizations share that state. The relevance is sectioned into highly relevant (color: green), rather relevant (color: blue), neutral (color: grey), rather not relevant (color: yellow), and not relevant (color: orange).
For example: "Loose coupling of systems or services" is highly relevant across all case organizations.



*AP that are not planned are not shown here

Figure 4.6.: Architecture Principles - Importance across all case organizations

Figure 4.7 shows the area of validity across all 30 architecture principles. The color represents the area of validity of an architecture principle. The area of validity is sectioned into the following levels: IT-Portfolio (color: green), IT-Organization (color: blue), organization (color: grey), program (color: yellow), and team (color: orange).
For example: Overall, 72% of the architecture principles are defined on the IT portfolio level.

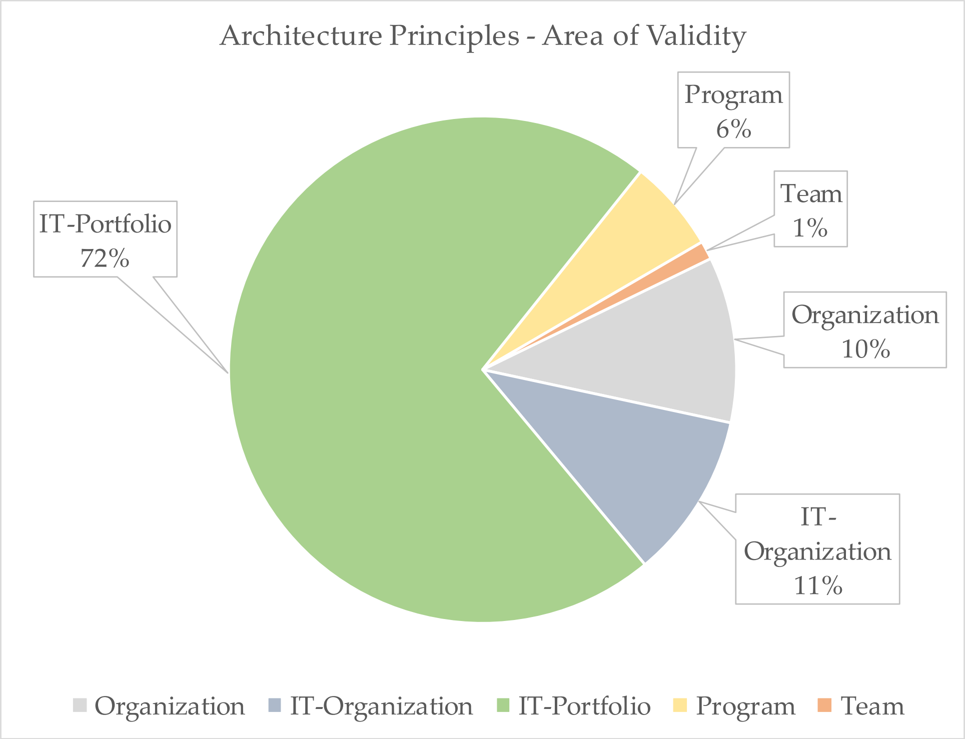
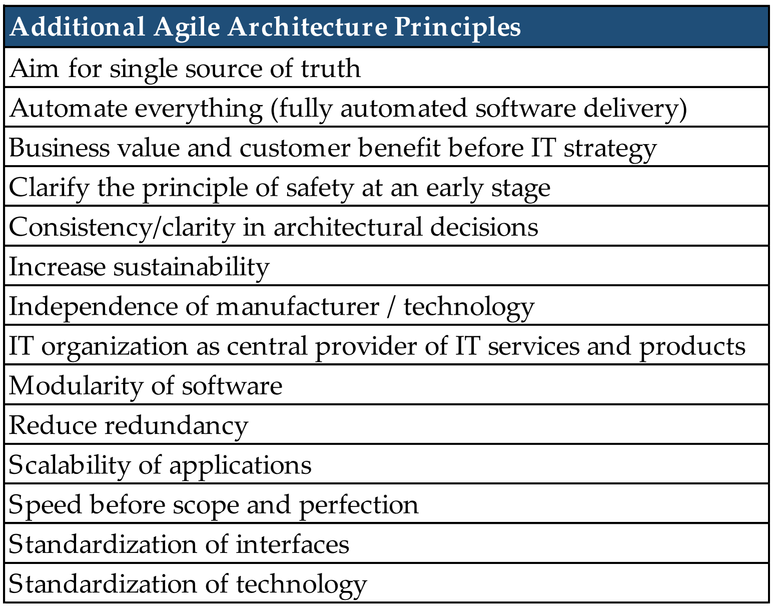


Figure 4.7.: Architecture Principles - Area of validity across all case organizations

As shown in Figure 4.7, 72% of architecture principles are defined on the IT portfolio level and therefor are relevant for the whole IT portfolio.Only 11% are defined at IT organization level. 10% on organizational level and only 6% on the program level. Rest is defined on team level.

Table 4.4 shows additional agile architecture principles mentioned by the interviewees.



Additional Agile Architecture Principles
Aim for single source of truth
Automate everything (fully automated software delivery)
Business value and customer benefit before IT strategy
Clarify the principle of safety at an early stage
Consistency/clarity in architectural decisions
Increase sustainability
Independence of manufacturer / technology
IT organization as central provider of IT services and products
Modularity of software
Reduce redundancy
Scalability of applications
Speed before scope and perfection
Standardization of interfaces
Standardization of technology

Table 4.4.: Additional architecture principles summarized from all case organizations

Are there architectural principles which are unsuitable for the agile environment or which are not applied? If so, why?

Case organization 1:

- **Stakeholder group AT:**
    - **Architecture principles that encourage monolithic approaches because agility is restricted, releases are restricted, and deployment is not possible** (I17, Group Leader IT, AT, C1)
    - **Architecture principles that do not lead to proprietary interfaces but generic web services to achieve a high level of reusability** (I17, Group Leader IT, AT, C1)
    - **Architecture principles that are too specific** (I14, Scrum Master, Requirement Engineer, AT, C1; I15, Head of EAM, M; C1; I45, Developer, AT, C4)
- **Stakeholder group EA:**
    - **Architecture principles that lead to separation of development and operations** (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1)

- **Stakeholder group M:**
  - **Architecture principles that are too specific and therefore offer less freedom** (I15, Head of EAM, M, C1)

Case organization 2:

- **Stakeholder group EA:**
  - **Architecture principles that encourage big design up-front** (I21, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)

Case organization 3:

- **Stakeholder group AT:**
  - **Architecture principles that are highly centralized** (I33, Scrum Master / Senior Agile Coach, AT, C3)
  - **Architecture principles that are top-down centered** (I33, Scrum Master / Senior Agile Coach, AT, C3)
- **Stakeholder group EA:**
  - **Architecture principles that focus primarily on cost savings and efficiency** (I34, Enterprise Architect 2, EA, C3)
  - **Architecture principles that build silos** (I34, Enterprise Architect 2, EA, C3)
  - **Architecture principles that encourage using standard software for standard processes** (I34, Enterprise Architect 2, EA, C3)
  - **Architecture principles that lead to standard processes across multiple products** (I34, Enterprise Architect 2, EA, C3)

Case organization 4:

- **Stakeholder group AT:**
  - **Architecture principles that lead to a transient object plexus** (I45, Developer, AT, C4)
  - **Architecture principles that keep individual objects transient during a process in the course of an automation process** (I45, Developer, AT, C4)

- **Stakeholder group EA:**
  - **Architecture principles that lead to functional redundancy** (I42, Enterprise Architect, EA, C4)
  - **Architecture principles that lead to technical redundancy** (I42, Enterprise Architect, EA, C4)
- **Stakeholder group M:**
  - **Deterministic and irreversible architecture principles** (I43, Head of Main Department, M, C4)

`Who is responsible for the creation and specification of the architectural principles? + Is there a guideline for the creation of architectural principles? + (If yes): With which specifications? + (If yes): Who created this guide and is responsible for its maintenance?`

The EAM is responsible for the creation and specification of architectural principles in 75% of the case organizations. 25% of the case organizations state that CTO office and system architects are responsible. In addition, all case organizations have a guideline for creating architectural principles. The EAM is responsible for both creating and maintaining the architectural principles.

The guideline contains the following specifications:

- Name
- Description
- Implications

Additionally, case organization 3 requires the following specifications:

- Area of validity
- Benefit
- Specification examples

Case organization 4 lists the definition of a principle in general and the dependencies to other principles in addition to the above.

`When do new architectural principles become valid?`

In 75% of the case organizations, the architectural principles are valid after approval in the community. The exception is case organization 2. In case organization 2, the validation is made by the CTO office in cooperation with the system architects.

75% of case organizations decide democratically whether an architecture principles becomes valid or not. However, case organization 3 uses a different approach. Decisions are not based on a democratic approach but are made using the highest-paid person's opinion principle (HiPPO principle). This usually is the chairperson.

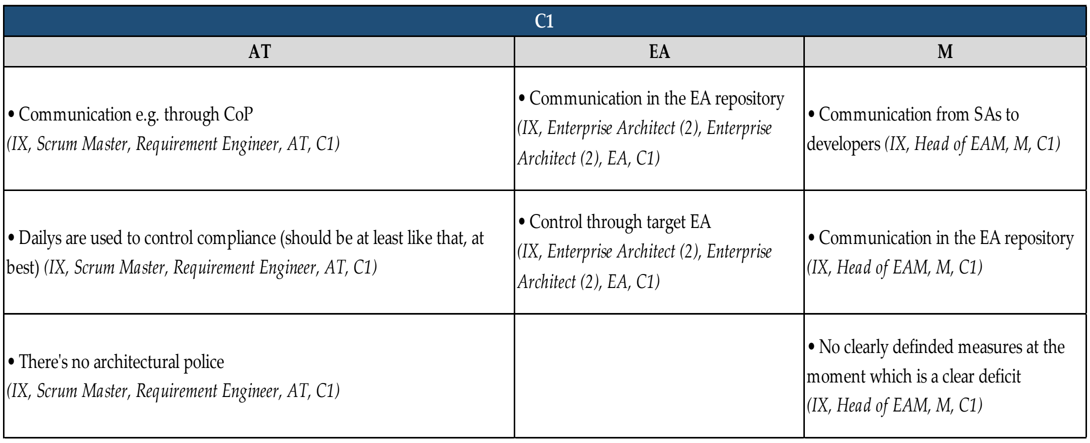
The time of validity varies according to importance, principle, priority, importance and dependency in all case organizations. Legacy systems are only adapted to critical issues such as new data protection or security principles.

50% of case organizations document their decisions directly in Confluence. The remaining case organizations have no specific processes.

What measures exist for the implementation of the newly created architectural principles?

In the following section, the measures for the implementation of the newly created architectural principles per case organization are listed.

Case organization 1:
Table 4.5 shows the measures taken by case organization 1 per stakeholder group.

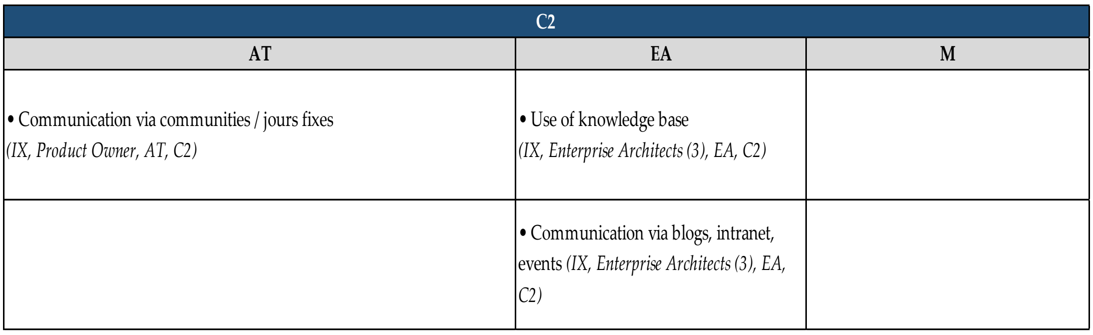


C1		
AT	EA	M
• Communication e.g. through CoP *(IX, Scrum Master, Requirement Engineer, AT, C1)*	• Communication in the EA repository *(IX, Enterprise Architect (2), Enterprise Architect (2), EA, C1)*	• Communication from SAs to developers *(IX, Head of EAM, M, C1)*
• Dailys are used to control compliance (should be at least like that, at best) *(IX, Scrum Master, Requirement Engineer, AT, C1)*	• Control through target EA *(IX, Enterprise Architect (2), Enterprise Architect (2), EA, C1)*	• Communication in the EA repository *(IX, Head of EAM, M, C1)*
• There's no architectural police *(IX, Scrum Master, Requirement Engineer, AT, C1)*		• No clearly definded measures at the moment which is a clear deficit *(IX, Head of EAM, M, C1)*

Table 4.5.: Measures for the implementation - case organization 1

All stakeholder groups declare increased communication as a measure of implementation.

Case organization 2:
Table 4.6 shows the measures taken by case organization 2 per stakeholder group.

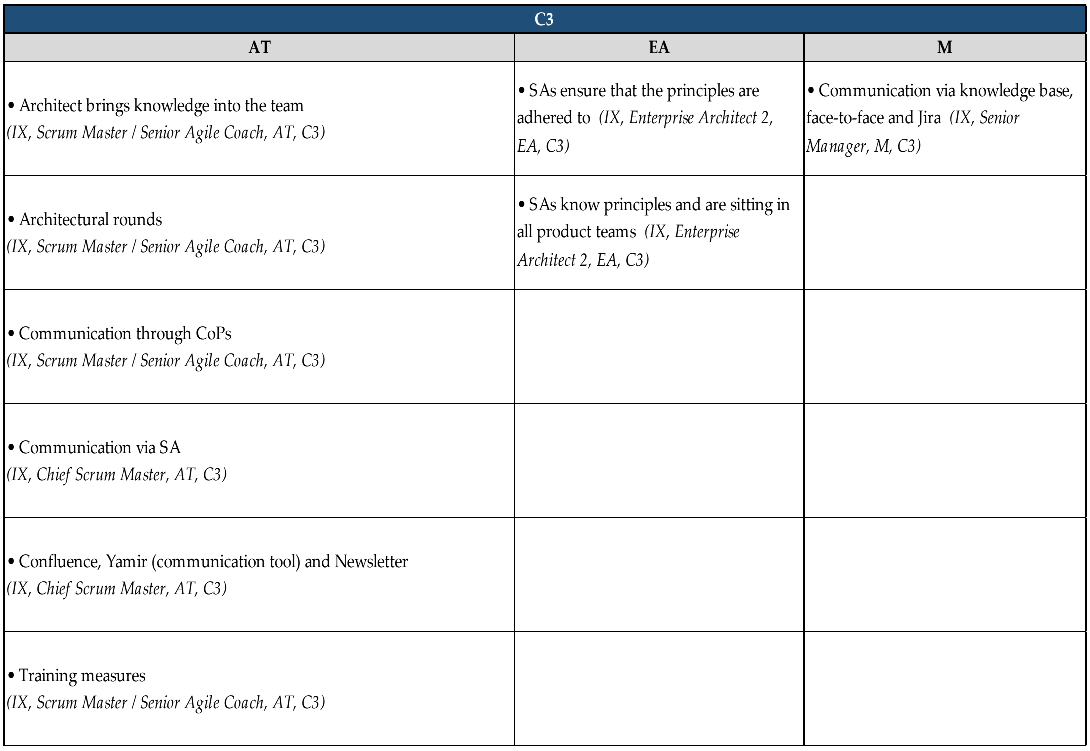


C2		
AT	EA	M
• Communication via communities / jours fixes *(IX, Product Owner, AT, C2)*	• Use of knowledge base *(IX, Enterprise Architects (3), EA, C2)*	
	• Communication via blogs, intranet, events *(IX, Enterprise Architects (3), EA, C2)*	

Table 4.6.: Measures for the implementation - case organization 2

All stakeholder groups declare increased communication as a measure of implementation.

Case organization 3:
Table 4.7 shows the measures taken by case organization 3 per stakeholder group.

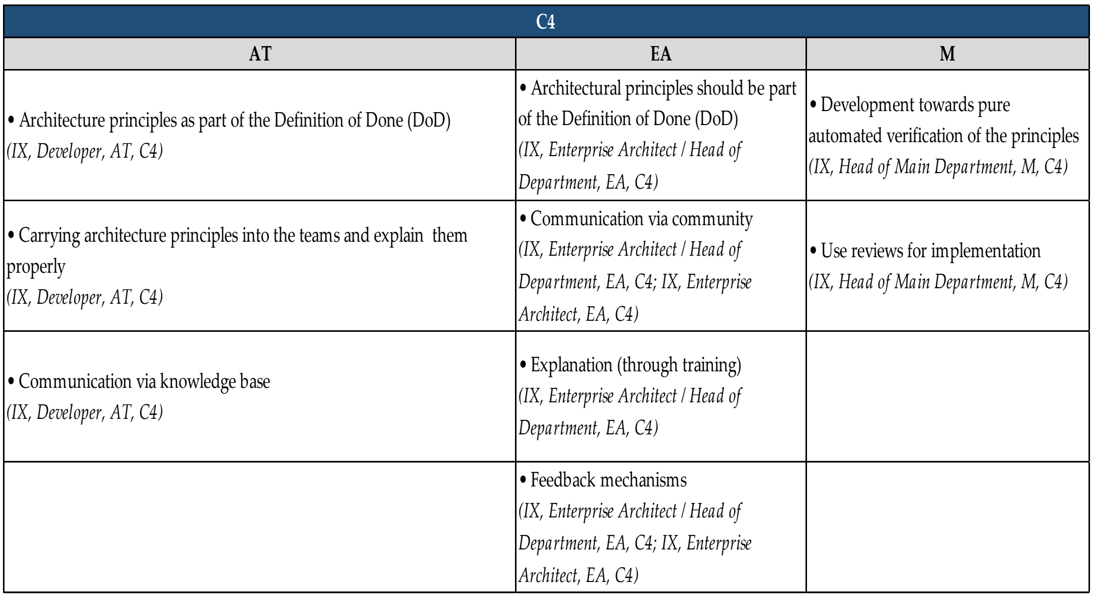


C3		
AT	EA	M
• Architect brings knowledge into the team *(IX, Scrum Master / Senior Agile Coach, AT, C3)*	• SAs ensure that the principles are adhered to *(IX, Enterprise Architect 2, EA, C3)*	• Communication via knowledge base, face-to-face and Jira *(IX, Senior Manager, M, C3)*
• Architectural rounds *(IX, Scrum Master / Senior Agile Coach, AT, C3)*	• SAs know principles and are sitting in all product teams *(IX, Enterprise Architect 2, EA, C3)*	
• Communication through CoPs *(IX, Scrum Master / Senior Agile Coach, AT, C3)*		
• Communication via SA *(IX, Chief Scrum Master, AT, C3)*		
• Confluence, Yamir (communication tool) and Newsletter *(IX, Chief Scrum Master, AT, C3)*		
• Training measures *(IX, Scrum Master / Senior Agile Coach, AT, C3)*		

Table 4.7.: Measures for the implementation - case organization 3

All stakeholder groups declare increased communication as a measure of implementation. According to the stakeholder the groups AT and EA, SAs transfer the knowledge of new architecture principles into the teams.

Case organization 4:
Table 4.8 shows the measures taken by case organization 4 per stakeholder group.



C4		
AT	EA	M
• Architecture principles as part of the Definition of Done (DoD) *(IX, Developer, AT, C4)*	• Architectural principles should be part of the Definition of Done (DoD) *(IX, Enterprise Architect / Head of Department, EA, C4)*	• Development towards pure automated verification of the principles *(IX, Head of Main Department, M, C4)*
• Carrying architecture principles into the teams and explain them properly *(IX, Developer, AT, C4)*	• Communication via community *(IX, Enterprise Architect / Head of Department, EA, C4; IX, Enterprise Architect, EA, C4)*	• Use reviews for implementation *(IX, Head of Main Department, M, C4)*
• Communication via knowledge base *(IX, Developer, AT, C4)*	• Explanation (through training) *(IX, Enterprise Architect / Head of Department, EA, C4)*	
	• Feedback mechanisms *(IX, Enterprise Architect / Head of Department, EA, C4; IX, Enterprise Architect, EA, C4)*	

Table 4.8.: Measures for the implementation - case organization 4

All stakeholder groups declare increased communication as a measure of implementation. According to the stakeholder groups AT and EA, architecture principles should be part of the DoD and explained properly.

In which way is compliance with the architectural principles checked and ensured?

Case organization 1:
In case organization 1, the verification of compliance with architectural principles is based on trust and is currently, according to the case organization, clearly in deficit. Attempts are being made to counter this by frequent contact and continuous communication. Occasionally, a review takes place during the official conferences (review). There is no formal structured process for compliance. Attention is only paid to compliance at the beginning of the release process. However, there is no explicit oversight after development and implementation.

Case organization 2:
Case organization 2 checks compliance with architectural principles manually and only

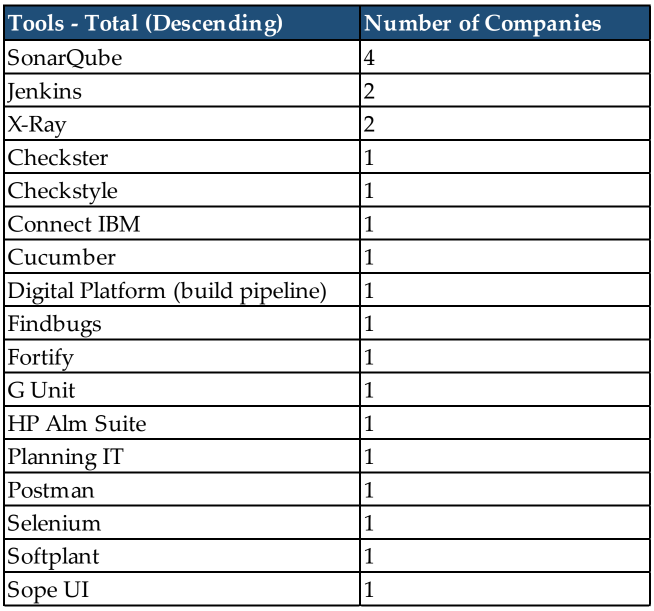
sporadically. There is no predefined process. The Jour Fixe, an explicit compulsory event for architects, permits an approval but no real control.

Case organization 3:
In case organization 3, compliance with the architectural principles is the responsibility of the team architects (here SA). With the help of code, peer reviews and the use of SonarQube, the architects manually check the code quality.

Case organization 4:
Case organization 4 has no explicit control mechanisms. Only manual checks are carried out from time to time.

Which tools are used to test compliance with architectural principles?

Table 4.9 lists the tools used to check compliance with architectural principles. Tools are sorted by relevance in descending order.



Tools - Total (Descending)	Number of Companies
SonarQube	4
Jenkins	2
X-Ray	2
Checkster	1
Checkstyle	1
Connect IBM	1
Cucumber	1
Digital Platform (build pipeline)	1
Findbugs	1
Fortify	1
G Unit	1
HP Alm Suite	1
Planning IT	1
Postman	1
Selenium	1
Softplant	1
Sope UI	1

Table 4.9.: Tools used to test compliance summarized from all case organizations

Interestingly, although the tools used vary greatly, all case organizations use SonarQube for adhering to architectural principles.

What problems do you identify when introducing or implementing architectural principles?

Table 4.10 lists the problems encountered when introducing or implementing architectural principles. They are sorted by relevance in descending order.

Problems - Total (Descending)	Number of Companies
Compliance (philosophy)	3
Lack of knowledge and understanding	3
Lack of verification and control	3
Poor communication	3
Abstraction level of architectural principles	1
Added value of the principles is missing	1
Clear definition of roles	1
Conflicting goals	1
Cultural change	1
Dealing with mortgages	1
Decision vacuum	1
Gap between EA and products	1
Interaction of the cross-sectional role and the operative implementation inside the team	1
Lack of commitment	1
Lack of intrinsic motivation	1
Learning effect	1
Low enablement	1
Project pressure vs perfect solution	1
Short-term vs. medium-term solution	1
Transfer to application development	1
Validity of principles	1

Table 4.10.: Problems when introducing or implementing architecture principles summarized from all case organizations

75% of the case organizations report poor communication, lack of compliance, lack of knowledge and understanding and lack of verification and control as a problem with the introduction or implementation of architectural principles.

4.3. Architecture Boards

Name typical examples of architectural decisions in an agile environment:

In the following, typical architectural decisions in an agile environment are summarized from all case organizations.

- **Block or continue software development when architectural decisions are violated or when it contradicts the architectural principle.**(I34, Enterprise Architect

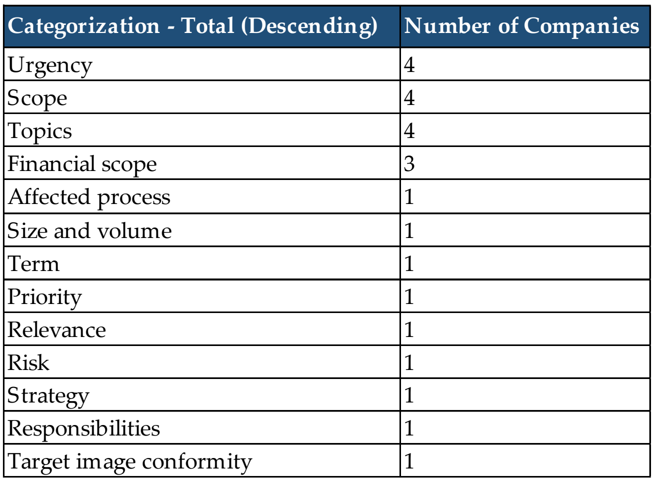
2, EA, C3)

- **Choice of repository** (I45, Developer, AT, C4)
- **Cloud first** (I17, Group Leader IT, AT, C1, I15, Head of EAM, M, C1; I23, Product Owner, AT, C2; I36, Enterprise Architect 1, EA, C3)
- **Contribution of architectural principles** (I45, Developer, AT, C4)
- **Correct professional assignment** (I21, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)
- **Creation and definition of solution space** (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, EA, C1; I15, Head of EAM, M, C1; I42, Enterprise Architect, EA, C4)
- **Decide on which libraries to use, whereas especially third-party libraries can cause problems** (I33, Scrum Master / Senior Agile Coach, AT, C3)
- **Decoupling options** (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, EA, C1)
- **Definition of process flow** (I37, Chief Scrum Master, AT, C3)
- **Frameworks and tools for software development** (I17, Group Leader IT, AT, C1; I14, Scrum Master, Requirement Engineer, AT, C1; I37, Chief Scrum Master, AT, C3; I45, Developer, AT, C4)
- **How to cut and disassemble monoliths** (I43, Head of Main Department, M, C4)
- **Interface design and landscape** (I14, Scrum Master, Requirement Engineer, AT, C1; I37, Chief Scrum Master, AT, C3)
- **Is middleware development permitted or not**(I34, Enterprise Architect 2, EA, C3)
- **Libraries must be managed centrally** (I33, Scrum Master / Senior Agile Coach, AT, C3)
- **Loose coupling of systems** (I43, Head of Main Department, M, C4)
- **Make or buy** (I41, Enterprise Architect / Head of Department, EA, C4; I43, Head of Main Department, M, C4)
- **Make product cut suitable to generate benefits** (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, EA, C1)
- **Minimized dependencies** (I36, Enterprise Architect 1, EA, C3)
- **Modularity regarding process, information, application (Micro-architecture)** (I41, Enterprise Architect / Head of Department, EA, C4, I43, Head of Main Department, M, C4)
- **Necessity of big data technologies (Elastic / MongoDB)** (I17, Group Leader IT, AT, C1)
- **Objective flexibility and autonomy** (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, EA, C1)
- **Prevent a best breed approach** (I31, Senior Manager, M, C3)

- **Single responsibility principle** (I36, Enterprise Architect 1, EA, C3)
- **Taking out technical mortgages** (I23, Product Owner, AT, C2)
- **Technology decisions** (I34, Enterprise Architect 2, EA, C3)
- **Use of auto-generated code regarding security risks and testability** (I14, Scrum Master, Requirement Engineer, AT, C1)
- **Use of APIs** (I31, Senior Manager, M, C3; I36, Enterprise Architect 1, EA, C3; I41, Enterprise Architect / Head of Department, EA, C4)
- **Use of capability assessments** (I34, Enterprise Architect 2, EA, C3)
- **Use of container in software development (yes/no)** (I17, Group Leader IT, AT, C1)
- **Use of encapsulation** (I41, Enterprise Architect / Head of Department, EA, C4)
- **Use of infrastructure components regarding adaptability in the future** (I14, Scrum Master, Requirement Engineer, AT, C1; I37, Chief Scrum Master, AT, C3; I34, Enterprise Architect 2, EA, C3; I45, Developer, AT, C4)
- **Use of public clouds** (I34, Enterprise Architect 2, EA, C3)
- **Use of service repository** (I17, Group Leader IT, AT, C1)
- **Virtualized infrastructure vs. dedicated infrastructure** (I17, Group Leader IT, AT, C1)
- **Which services should support which business capabilities** (I21, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)

How do you categorize architectural decisions?

Table 4.11 lists the categories used for architectural decisions. These are sorted by relevance in descending order.



Categorization - Total (Descending)	Number of Companies
Urgency	4
Scope	4
Topics	4
Financial scope	3
Affected process	1
Size and volume	1
Term	1
Priority	1
Relevance	1
Risk	1
Strategy	1
Responsibilities	1
Target image conformity	1

Table 4.11.: Categorization of architectural decisions

100% of the case organizations state that ”Urgency”, ”Scope” and ”Topics” are used to categorize architectural decisions. 75% of the organizations cite ”Financial scope” as a category.

How and where are architectural decisions documented?

The following section describes how and where architectural decisions are documented. A distinction is made between case organizations.

Case organization 1:
In case organization 1, architectural decisions are stored in Confluence and in the book of standards (I14, Scrum Master, Requirement Engineer, AT). An enterprise decision repository, which is a repository for enterprise-wide application landscapes, also exists. The latter is part of the product documentation in Confluence, and stores all architecture builds / blueprints (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1). Planning IT is also used (I15, Head of EAM, M, C1; I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1).

Case organization 2:
In case organization 2, architectural decisions are made in the architecture boards and stored informally in SharePoint (I23, Product Owner, AT). In addition, there is an EAM knowledge base (I21, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2) that also contains informal decision protocols (I21, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2).

Case organization 3:
In case organization 3, architectural decisions are stored in Word documents, PDFs, PowerPoint, tickets and mails (I37, Chief Scrum Master; I33 Senior Agile Coach / Scrum Master AT, C3). There is neither a decision backlog nor a wiki in which all decisions are stored combined. The majority of decisions are distributed and not centralized and can be viewed via Confluence, Jira, Planning IT and SS (Incident Management) (I37, Chief Scrum Master; I33 Senior Agile Coach / Scrum Master, AT; I31, Senior Manager, M; C3).

Case organization 4:
In case organization 4, comprehensive architectural decisions are documented in Confluence and made in the architecture community (I45, Developer, AT; I41, Enterprise Architect / Head of Department, EA; I43, Head of Main Department, M; C4). In the past, a central architecture decision book existed. This, however, no longer exists (I41, Enterprise Architect / Head of Department; I42, Enterprise Architect, EA, C4). Today, the approach is inconsistent and decentralized. Handling the documentation is up to the teams, each team handles it differently. The decisions made are often non-comprehensive and made in collaboration with the architecture community (I41, Enterprise Architect / Head of Department; I42, Enterprise Architect, EA, C4). In some cases, a decision in an AT is not documented. This is due to the lack of "desire and time" to deal with the documentation (I41, Enterprise Architect / Head of Department; I42, Enterprise Architect, EA, C4). An escalation mode is also missing. Additionally, there are two different wikis. In the Enterprise Architecture Wiki comprehensive decisions are documented, whereas in the Project Wiki only project-specific decisions are documented (I41, Enterprise Architect / Head of Department; I42, Enterprise Architect, EA, C4).

Overall, 100% of the case organizations use Confluence to document architectural decisions. 75% do not have a specific form and process for the documentation of architectural decisions.

At what level is an architectural decision made by which role?

In the following section, the architectural decisions made by a certain role on the individual organizational level are listed for each case organization.

Case organization 1: Table 4.12 shows the architectural decisions on each organizational level of case organization 1.

Level	Role	Type of Decision
Organization	• CTO	• Business architecture • Strategy
IT Organization & Portolfio	• EA (Governance) • Portolio manager	• Architecture principles • Architecture tools and processes • Budgeting • Functional and technical solution space • Functional product cut • IT strategy • Product portfolio priorization • Target enterprise architecture
Program & Team	• Developer • EA • Product owner • Program manager • SA	• Documentation of non compliance with architectural standards • Narrowed technical solution space • Product features • Software and solution architecture

Table 4.12.: Architectural decisions made by a certain role on the individual organizational level

Case organization 2:
Table 4.13 shows the architectural decisions on each organizational level of case organization 2.

Level	Role	Type of Decision
Organization	• Executive Board	• Strategy
IT Organization	• CTO	• IT Strategy
Portfolio	• Enterprise Architect • System Architect	• Functional Product Cut • Strategic Roadmap • User Journeys
Program	• Domain Architect	• Alignment of Domain to Enterprise Architecture • Technology Decisions
Team	• Developer • Domain Architect • Solution Architect	• Technology Decisions

Table 4.13.: Architectural decisions made by a certain role on the individual organizational level

Case organization 3:
Table 4.14 shows the architectural decisions on each organizational level of case organization 3.

Level	Role	Type of Decision
Organization	• Chief Technology Officer (CTO) • Lead countries (ESP, DE)	• Business architecture • Strategy
IT Organization	• Chapter lead • EA • Head of enterprise architecture	• Architecture principles • Architecture framework • Governance and compliance • IT strategy
Portfolio	• Chapter lead • Chief product owner • DA • EA	• Architecture blueprints • Budgeting • Domain principles • Dependencies between products • Governance and compliance • High level technology decisions • Product decisions in the domain • Product portfolio priorization • Release planning
Program	• Chief Product Owner • DA • Program Manager	• Dependencies between subproducts • Compliance with architecture principles • Release planning
Team	• Product Owner • Senior Developer • SA	• Compliance with architecture principles • Software architecture

Table 4.14.: Architectural decisions made by a certain role on the individual organizational level

Case organization 4:
Table 4.15 shows the architectural decisions on each organizational level of case organization 4.

Level	Role	Type of Decision
Organization	• CTO	
IT Organization	• Enterprise Architect • IT Management	
Portfolio	• Enterprise Architect • Portfolio Steering Committee • Program Manager	• Cross-domain Architectural Decisions • Functional Features • Product Portfolio Priorization • Technical Depts
Program	• Domain Architect • Product Owner • Solution Architect	• Product Features • Solution Architecture
Team	• Developer • Solution Architect	• Software Architecture

Table 4.15.: Architectural decisions made by a certain role on the individual organizational level

What forms of architectural boards do you have in your company? Please specify the existing forms in the attached Excel list.

In the following, a brief definition of the different types of circles used in the case organizations is given, to provide a better understanding. For more detail, refer to Tables 4.17 and 4.18.

Architecture Board:
The architecture board is a circle of people responsible for development decisions and overseeing the implementation of the strategy. The circle should be representative of all the key stakeholders in the architecture and will typically comprise a group of executives responsible for the review and maintenance of the strategy architecture and all of its sub-architectures. In detail, its goal is to create consistency between the sub architecture, identify reusable components, enforce architecture compliance, discuss changing business needs and create transparency with explanation of goals and reasons for decisions.

Center of Excellence:
A Center of Excellence (CoE) is a circle of experts with specialized skills and expertise knowledge which is often purposefully created by an organization with funded members. The circle provides leadership, addresses skill or knowledge deficits and purposely propagates knowledge within an organization.

Community of Practice:
A Community of Practice (CoP) is a circle of people who share a specific craft and / or profession. The focus of the CoP is on developing expertise, skills, and proficiency. The goal of the CoP is to learn from each other, discuss and solve problems together and increase their own knowledge and propagate it within the organization, master the discipline and learn. In contrast to CoEs, a CoP is on a voluntary basis and is usually driven by the common interest of the participants in a topic and not by the companies themselves. CoPs can be the result of a CoE, once its official funding is cut by the organization.

Executive Board:
An Executive Board (EB) is a circle of managers and executives. The EB is responsible for strategic planning, prioritization in the portfolio and strategic portfolio decisions. Furthermore, it communicates business and organizational constraints and is required for the strategic alignment of business and IT.

Product Portfolio Board:
The Product Portfolio Board (PPB) is a circle of people responsible for the complete IT product portfolio of the company, consisting of portfolio structure, building principles and approval of contents. Who participates depends on the product portfolio. Usually, the core teams of a technology stack and a responsible manager take part. The PPB defines release versions and dates for the product catalog in coordination with resources. The goal is to create acceptance of the product portfolio and define uniform technology specifications for all development teams.

The following section describes how the different types of circles are used in the case organizations. Table 4.16 shows the type of circle that exists in each organization.

Type of Circle	C1	C2	C3	C4
Architecture board	+	+	+	+
Executive board	+	+	+	+
Community of practice	-	+	+	+
Product portfolio board	+	+	-	+
Center of excellence	+	-	+	-

Table 4.16.: Type of circles used in case organizations

Table 4.17 describes the different types of circles. First, a short description is given. This is followed by the goals of the circle, the frequency of the meetings, the persons heading the meeting (chairmen) and the participants.

Table 4.18 describes the decision-making process, influence or area of validity, outcome of decision, binding nature of the decision made, documentation of the decision and the suitability for the agile environment.

Name	Description	Goals	Frequency	Lead	Participants
Architecture board	• Communication of (new) architectural principles, guidelines and decisions to the chapter leads • Gradually replaced by new (E)A decision process (CoE EAM) • Interface, portfolio, program • Supreme body for development decisions	• Approval of comprehensive buildings (across several process clusters) • Approval of non-overlapping buildings (within a process cluster) • Coordination and equal understanding between central architects and domains • Creating transparency, explaining the goals of the principles and reasons for decisions, but above all, informing the developers about these	• Every 1-4 weeks	• Chapter lead architecture • EA • System architect	• Chapter lead architecture • Chief developer • CTO • Domain architects • Enterprise architects • EA group leaders from the process cluster • IT main department heads • IT department heads • Purchasing department heads • System architects
Center of excellence for EAM	• Circle of experts for the control of far-reaching development activities	• Defining architecture constraints and documentation standards • Defining coding standards and indicators for software maturity •Identification of relevant stakeholders and of necessary development tasks within a development activity, if necessary, definition of acceptance criteria and approvers of individual development tasks	• Every 1-2 weeks	• 1 Head of department	• Several department heads • Several EA group heads
Center of excellence for EAM - Process cluster	• Expert group for the control of development activities within a process cluster	• Identification of relevant stakeholders and of necessary development tasks within a development activity, if necessary definition of acceptance criteria and approvers of individual development tasks	• Every 1-4 weeks	• 1 IT department manager	• Several IT department heads • EA group leaders from the process cluster
Center of excellence for Product portfolio mgmt	• Expert group for the substantive discussion of changes to the product portfolio, subcommittee for Product Portfolio Board	• Build-up appropriate tools for management of product portfolio • Build-up IT discipline "Product Portfolio Management" incl. its roles and processes • Definition of what a product is as well as its structure and criteria • Map out product portfolio of IT organization	• Every week	• Product owner of product catalog	• 1 participant (e-architect) per IT department
Community of Practice for Architecture	• Coordination of business architecture and assignment of capabilities to domains • Development of target architecture • Free format with single events	• Exchange between architects • Transparency	• Every week	• chapter lead Architecture • System architecture	• All those interested in architecture
Community of Practice for Software Development	• Development and decision on API Guidelines • Development and decision on Backend Dev. Guidelines • Development and decision on Frontend Dev. Guidelines • Free format with given tasks	• Definition of uniform API guidelines • Definition of uniform Dev Guidelines • Further development of developers	• Every week	• chapter lead Backend Development • chapter lead Data Governance • chapter lead Frontend Development • Chief Technology Office	• Architects • Developers • Data Governance • Security
Executive Board	• Management	• Definition of the strategy and prioritization in the portfolio • Relevant excerpt: strategic portfolio decisions	• Every month	• Rolling between management	• Management
Product Portfolio Board	• It defines release versions and dates for the Product Catalog in coordination with resources / finance steering • Technology decisions at portfolio level • The Product Portfolio Board (PPB) is responsible for the complete IT product portfolio of enterprise: portfolio structure, building principles and approval of contents	• Acceptance of the product portfolio • Uniform technology specifications for all development teams	• Every 2 weeks	• CTO	• Core teams to cover the technology stack and optional participants (incl. system architecture and enterprise architecture) • 1 Participant (head of department) for each main process Business + IT

Table 4.17.: Architecture circles across case organizations (1/2)

Name	Decision-making process	Influence or area of validity	Outcome of the decision	Binding nature of the decision made	Documentation of the decision	Suitability for the agile environment
Architecture board	• Decision by head of department • Information committee, if necessary escalation into other committees • No decision in the meeting; principles, decisions etc. are made in advance in the communities • Unanimous decision of the 2 heads of department	• Organization-wide • Within a process cluster	Approved development	• Binding • Very strong recommendation, decision by management	• Confluence • Panel control system (incl. protocol) • Protocol	• 50% of the companies say low • 50% state that it is suitable since decentralized integration by business area architects
Center of excellence for EAM	• By mutual agreement	• IT organization	• Fixed development activities	• Binding	• Confluence	• Good
Center of excellence for EAM - Process cluster	• By mutual agreement	• Within a process cluster	• Fixed development activities	• Binding	• Confluence	• Good
Center of excellence for Product portfolio mgmt	• By mutual agreement	• Total Porftolio	• Recommendation for product structure	• Referral	• Confluence	• Good
Community of Practice for Architecture	• Democratically	• Not relevant	• Not relevant	• Binding	• Confluence	• Very well suited, learning together
Community of Practice for Software Development	• Democratically	• Organization-wide		• Binding	• Confluence	• Very well suited, learning together
Executive Board	• By mutual agreement • Topics and decision documents are presented by topic managers	• Binding for entire organization	• Decision on and prioritization of strategic projects	• Binding	• Protocol	• Required for strategic alignment
Product Portfolio Board	• By mutual agreement	• Mandatory for complete development; deviations must be approved separately by CTO • Whole organization	• Approved product structure • Document with all valid specifications	• Binding	• Confluence • Document Technology Guideline	• On the one hand, technology specifications are not suitable, as they can restrict the autonomy of the teams; on the other hand, prerequisites, e.g. in order to be able to relocate topics between teams (uniform technology stack and architectural pattern) • Medium

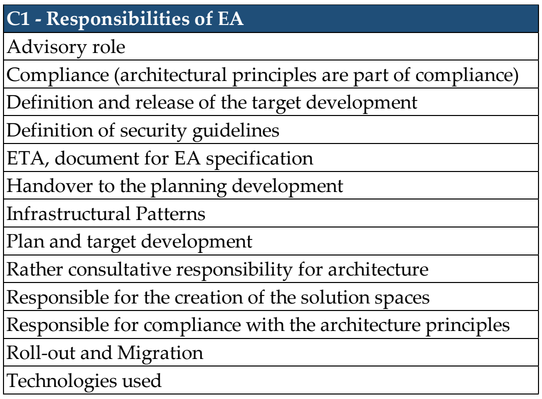
Table 4.18.: Architecture circles across case organizations (2/2)

4.4. Role of Enterprise Architecture Management

What responsibilities do Enterprise Architects have in your company?

In the following, the responsibilities of the EAs are listed for each case organization.

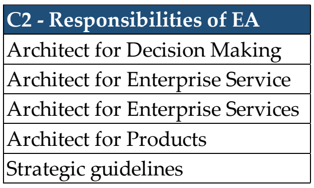
Case organization 1:
Table 4.19 lists the responsibilities of EAs in case organization 1.



C1 - Responsibilities of EA
Advisory role
Compliance (architectural principles are part of compliance)
Definition and release of the target development
Definition of security guidelines
ETA, document for EA specification
Handover to the planning development
Infrastructural Patterns
Plan and target development
Rather consultative responsibility for architecture
Responsible for the creation of the solution spaces
Responsible for compliance with the architecture principles
Roll-out and Migration
Technologies used

Table 4.19.: Responsibilities of EAs - Case organization 1

Case organization 2:
Table 4.20 lists the responsibilities of EAs in case organization 2.



C2 - Responsibilities of EA
Architect for Decision Making
Architect for Enterprise Service
Architect for Enterprise Services
Architect for Products
Strategic guidelines

Table 4.20.: Responsibilities of EAs - Case organization 2

Case organization 3:
Table 4.21 lists the responsibilities of EAs in case organization 3.

C3 - Responsibilities of EA
Advising executives
Application and information system architecture
Architect for communication and collaboration
Architect for continuously applying principles
Architect for decision making
Business architecture
Coaching and consulting of developers
Capability planning
Communicate and monitor principles and paradigms
Decides which domain is current from a system point of view
Future architecture blueprint will be created
Guidelines (book of standards)
Make frameworks available
Placement and replacement
Preparation of guidelines
Responsible for compliance with the TOGAF cycle
Strategic planning of IT
Support during operationalization
Training architects
Zoning plan

Table 4.21.: Responsibilities of EAs - Case organization 3

Case organization 4:
Table 4.22 lists the responsibilities of EAs in case organization 4.

C4 - Responsibilities of EA
Architect for communication and collaboration
Architect for continuously applying principles
Architectural principles
Consulting function for guidelines and solutions
Development of the tool for IT knowledge base
Establishment and management of the architecture communities
Further development and skill development (architecture)
Guardians and advocates of the principles
Internal consulting for the IT organization
Problem solving
Provide of information regarding application landscapes / system provision / dependencies
Reference architecture
Specify the guidelines as to how components may communicate
Structure and maintenance of the wiki
Support of methodologies and process modelling
Target images and roadmaps
Technical enablement of ATs
Technology assessment
Tool support
Trainings and skill development
Trend marketing

Table 4.22.: Responsibilities of EAs - Case organization 4

How did the responsibilities of Enterprise Architects change in an agile environment?

Case organization 1:
The responsibilities of Enterprise Architects in the agile environment have not fundamentally changed in case organization 1. EAs have an increased workload due to increased support, the stronger enabling of the AT and being in charge of a large number of components. This also increases the scope of their responsibility (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1). The interviews showed that EAs feel strongly overloaded in many ways. In the past, for example, one EA was responsible for 4 systems, where today one EA is responsible for approximately 20 components (I13, Scrum Master, Requirement Engineer, AT, C1). In addition, the role of the chief architect no longer exists. Due to the excessive workload, bottlenecks occur in some places. This leads to reduced performance of the EA and results in a dissatisfied AT. The dissatisfaction is reflected in statements such as:

> *"The EA has lost importance, so what is the point of a quick solution with a high degree of abstraction?"*
>
> *-I17, Group Leader IT, AT, C1*

Among other things, the latter causes the gap between EAs and ATs to become larger (I13, Scrum Master, Requirement Engineer, AT, C1). The survey also revealed that today's activities will continue to diversify in the future as a result of change. Today, an EA primarily defines a target image and ensures overall coherence. In the future, a collaborative solution development and the business and IT alignment will move more into the focus (I15, Head of EAM, M, C1).

Case organization 2:
In case organization 2, the responsibility of the EA has not changed as the EAM was only established a year ago (I23, Product Owner, AT, C2). The primary assumption is that dependency management will receive greater priority, the level of detail of the models provided will decrease slightly and the overarching context will become increasingly important (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2).

Case organization 3:
According to the interviewees of case organization 3, collaboration between the EA and the AT and the work methodology have primarily changed (I37, Chief Scrum Master, AT, C3). The responsibility has remained the same with the exception of stronger support and the enabling of ATs. EAs are now assigned to a domain, are therefore closer to the products and work more closely with the ATs (I36, Enterprise Architect 1, EA, C3; I32, Enterprise Architect 2, EA, C3). According to some EAs, artifacts provisioned by EAs are no longer as detailed as they were in the past (I36, Enterprise Architect 1, EA,

C3). In the future, when new EAs are hired, more attention will be paid to technical know-how instead of business know-how (I36, Enterprise Architect 1, EA, C3). A future EA should be able to develop software and no longer provide abstract PowerPoint models (I36, Enterprise Architect 1, EA, C3). In addition, the responsibility of an EA will become more specific, including product responsibility (I36, Enterprise Architect 1, EA, C3).

Case organization 4:
In case organization 4, the EA's responsibility in the agile environment has changed and is moving away from a governance control function (I42, Enterprise Architect, EA, C4). An EA is no longer only a solution architect, but primarily a consultant and enabler (I42, Enterprise Architect, EA, C4). In the past, an architect had a veto right. However, due to the continued development of the EA toward consultant and enabler, this veto right no longer exists. The evaluation of projects and architectural conformity is now the responsibility of the DA (I42, Enterprise Architect, EA, C4). In addition, the evaluation and/or recommendation of project orders is taken over by decentralized architects (I41, Enterprise Architect / Head of Department, EA, C4). Today, an EA is located at the program level and is responsible for several teams (I41, Enterprise Architect / Head of Department, EA, C4). The focus is on principle-based action and communication through communities. The term ”guardian and advocate of principles” is used (I43, Head of Main Department, M, C4). In addition, the EA is also responsible for identifying new and suitable technologies and developing new skill sets and action patterns from them. (I43, Head of Main Department, M, C4). Collaboration and methodology have changed significantly. This can primarily be explained by the changed responsibility and structural changes caused by the omission of the waterfall model in software development (I41, Enterprise Architect / Head of Department, EA, C4). The latter leads to shorter conception phases and thus to new agile working methods. Consulting and enabling create a new form of communication and collaboration (I41, Enterprise Architect / Head of Department, EA, C4).

What distinguishes your Enterprise Architects and how do they differ from the classic role of Enterprise Architects in other companies?

In the following, the characteristics of the EAs are listed. Characteristics will be listed per stakeholder group.

Stakeholder group AT:

- **Architectural timetable including target image** (I44, Developer, AT, C4)
- **Basic knowledge of architecture** (I44, Developer, AT, C4)
- **Close and agile collaboration with teams** (I33, Scrum Master / Senior Agile Coach, AT, C3; I32, Enterprise Architect 2, EA, C3)

- **Component knowledge across the enterprise** (I44, Developer, AT, C4)
- **EA has an overarching role** (I13, Scrum Master, Requirement Engineer, AT, C1)
- **EAs act within the guidelines of the project** (I13, Scrum Master, Requirement Engineer, AT, C1)
- **EAs need to have experience with implementation** (I17, Group Leader IT, AT, C1)
- **EAs have more experience and know what is realistic** (I17, Group Leader IT, AT, C1)
- **Finding a solution through close collaboration with ATs** (I33, Scrum Master / Senior Agile Coach, AT, C3)
- **Role of the EA focuses more on technical expertise** (I33, Scrum Master / Senior Agile Coach, AT, C3; I42, Enterprise Architect, EA, C4)
- **Open for feedback** (I37, Chief Scrum Master, AT, C3)

Stakeholder group EA:

- **Better business and IT alignment** (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1,EA, C1)
- **Close and agile collaboration with teams** (I33, Scrum Master / Senior Agile Coach, AT, C3; I32, Enterprise Architect 2, EA, C3)
- **Communication of solution portfolio with pros and cons of a specific technology** (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, EA, C1)
- **Concept and clear picture are provided in the agile environment** (I42, Enterprise Architect, EA, C4)
- **Decoupling of professional and disciplinary leadership, because PO is responsible for professional leadership, especially to advance the product** (I32, Enterprise Architect 2, EA, C3)
- **Distance from detailed actual data acquisition** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)
- **EA team is organized across divisions for clear responsibilities and mapping to products** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)
- **In the case of a greater need in a domain, the procedure is more detailed accordingly (demand driven)** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)
- **Less anchoring in the business** (I41, Enterprise Architect / Head of Department, EA, C4)
- **Less planning and coordinating decision-making role** (I41, Enterprise Architect / Head of Department, EA, C4)

- **Less strongly noticed in the organization (business)** (I41, Enterprise Architect / Head of Department, EA, C4)
- **More technical expertise available** (I33, Scrum Master / Senior Agile Coach, AT, C3; I42, Enterprise Architect, EA, C4)
- **Reduction to the essential topics (here the priorities are strongly based on leverage)**(I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, EA, C1)
- **Solution architects previously active in applications are now EAs** (I36, Enterprise Architect 1, EA, C3)
- **Solution space becomes more and more technical** (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, EA, C1)
- **Staffing of ATs is still appropriate, if the tasks are clearly distributed** (I42, Enterprise Architect, EA, C4)
- **Greater focus on the usefulness of the models (lean approach)** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)

Stakeholder group M:

- **Architects must have good social competence; the competence to convince** (I43, Head of Main Department, M, C4)
- **Decisive step: First, architects build pipelines themselves and thus know what these tools are capable of which makes it easier to convince and increase intrinsic motivation** (I43, Head of Main Department, M, C4)
- **Fewer deep dives in processes** (I15, Head of EAM, M, C1)
- **Need of Governance EA know-how and in general competency in the processes and products** (I15, Head of EAM, M, C1)
- **No pure PowerPoint creator** (I43, Head of Main Department, M, C4)
- **Objective of the EA roll: More width, less depth** (I15, Head of EAM, M, C1)
- **Overall entrepreneurial approach** (I15, Head of EAM, M, C1)
- **Technology affinity** (I43, Head of Main Department, M, C4)

How did the working methodology of the Enterprise Architect change in an agile environment?

In the following, the working methodology of the EAs in an agile environment are listed per stakeholder group.

Stakeholder group AT:

- **Collaborative decision** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2; I33, Scrum Master / Senior Agile Coach, AT, C3)

- **Closer to teams** (I37, Chief Scrum Master, AT, C3; I32, Enterprise Architect 2, EA, C3)
- **Earlier: Decision above the heads of the developers (lack of involvement)** (I33, Scrum Master / Senior Agile Coach, AT, C3)
- **ETA specifies guidelines** (I13, Scrum Master, Requirement Engineer, AT, C1)
- **KANBAN Boards better suited than scrum because of greater focus** (I13, Scrum Master, Requirement Engineer, AT, C1; I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, EA, C1)
- **Less governance through PowerPoint** (I33, Scrum Master / Senior Agile Coach, AT, C3)
- **More preparation and communication** (I33, Scrum Master / Senior Agile Coach, AT, C3)
- **More support and enabling / more collaboration, less top-down** (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, EA, C1; I15, Head of EAM, M, C1; I33, Scrum Master / Senior Agile Coach, AT, C3)
- **Much more work on communities** (I33, Scrum Master / Senior Agile Coach, AT, C3)
- **Prioritization** (I13, Scrum Master, Requirement Engineer, AT, C1)
- **Refinement / no longer for standard projects** (I13, Scrum Master, Requirement Engineer, AT, C1)
- **Stronger collaboration** (I37, Chief Scrum Master, AT, C3)
- **Use of agile tools** (I13, Scrum Master, Requirement Engineer, AT, C1)

Stakeholder group EA:

- **Adaptation to agile procedures** (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, EA, C1; I41, Enterprise Architect / Head of Department, EA, C4)
- **Architects must find their way into agile models** (I41, Enterprise Architect / Head of Department, EA, C4)
- **Closer to teams** (I37, Chief Scrum Master, AT, C3; I32, Enterprise Architect 2, EA, C3)
- **Conceptual work is almost completely omitted** (I36, Enterprise Architect 1, EA, C3)
- **Daily "Scrum Whiteboard Meetings": Progress and problems are discussed using Jira, duration max. 15-20 mins. In the sprint, reviews are made on important topics, 2 team members have an SM certification, both members play an active role in planning (dual role). Currently, still time estimation and no 100% scrum usage** (I42, Enterprise Architect, EA, C4)

- **EAs have created their own agile methods with small role-specific modifications** (I41, Enterprise Architect / Head of Department, EA, C4)
- **Lack the time to assess projects every 3 months** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)
- **Faster feedback cycles** (I32, Enterprise Architect 2, EA, C3)
- **If technologically new territory, then spikes are carried out only when necessary** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)
- **Integration into agile projects is difficult** (I41, Enterprise Architect / Head of Department, EA, C4)
- **KANBAN Boards / better suited than scrum because of greater focus** (I13, Scrum Master, Requirement Engineer, AT, C1; I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, EA, C1)
- **Lower level of artifact detail / No PO for EA artifacts** (I32, Enterprise Architect 2, EA, C3)
- **More support and technical enabling / more collaboration, less top-down** (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, EA, C1; I15, Head of EAM, M, C1; I33, Scrum Master / Senior Agile Coach, AT, C3)
- **Participate in planning** (I36, Enterprise Architect 1, EA, C3)
- **Proximity to PO** (I36, Enterprise Architect 1, EA, C3)
- **Scrumban, planning in 2-week cycles with whiteboards, duration: 3 hours, including a retrospective with a duration of max 20 mins** (I42, Enterprise Architect, EA, C4)
- **Service providers** (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, EA, C1)
- **Shorter decision-making processes** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)
- **Well-founded decision** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)
- **Working in short planning horizons (3 to 6 months max)** (I41, Enterprise Architect / Head of Department, EA, C4)

Stakeholder group M:

- **Development decisions are deliberate** (I15, Head of EAM, M, C1)
- **EAs have created their own agile methods with small role-specific modifications** (I41, Enterprise Architect / Head of Department, EA, C4)
- **Integration into agile projects is difficult**(I41, Enterprise Architect / Head of Department, EA, C4)
- **Minimal required governance, not everything is planned in advance** (I15, Head of EAM, M, C1)

- **More support and enabling / more collaboration, less top-down** (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, EA, C1; I15, Head of EAM, M, C1; I33, Scrum Master / Senior Agile Coach, AT, C3)
- **Shift from direct and control to enabling and support** (I15, Head of EAM, M, C1)
- **Shift to emergent and intentional architecture** (I15, Head of EAM, M, C1)

How did the environment of the Enterprise Architect change in an agile environment?

In the following, changes to the environment of the EAs through agility are listed per stakeholder group.

Stakeholder group AT:

- **A sense of responsibility for the product is difficult to reconcile with the ideas of the EA** (AT vs EA) (I37, Chief Scrum Master, AT, C3)
- **ATs are functioning, now cross-over roles must be taken (especially architects)** (I13, Scrum Master, Requirement Engineer, AT, C1)
- **Interface is becoming more important** (I44, Developer, AT, C4)
- **Much closer to teams and product/solution** (I33, Scrum Master / Senior Agile Coach, AT, C3; I44, Developer, AT, C4)
- **No scaling at portfolio and organizational level, currently only at program and team level** (I13, Scrum Master, Requirement Engineer, AT, C1)
- **On request: Backlog refinement usually includes an architect** (I44, Developer, AT, C4)
- **Proximity to customers** (I44, Developer, AT, C4)
- **Self-determination of the teams** (I37, Chief Scrum Master, AT, C3)
- **Unchanged: Gap between operative working and EAM is too large** (I13, Scrum Master, Requirement Engineer, AT, C1)

Stakeholder group EA:

- **Architects must accompany projects much longer than before, as the main activity took place in the long conception phase** (I41, Enterprise Architect / Head of Department, EA, C4)
- **Better and more frequent communication through EAM between product management, dev teams and portfolio management** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)

- **Clear roles and responsibilities in each product** (I32, Enterprise Architect 2, EA, C3)
- **EA is now product- and process-oriented / traded such as developers** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)
- **Improved coordination** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)
- **Involved in less projects** (I42, Enterprise Architect, EA, C4)
- **Less directly involved in the project through reorganization** (I42, Enterprise Architect, EA, C4)
- **No longer in the classic governance cluster, as they are in the product teams** (I36, Enterprise Architect 1, EA, C3)
- **Open space has increased transparency**(I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, EA, C1)
- **PO has a clear product focus** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)
- **Rather involved in initialization or early phase**(I42, Enterprise Architect, EA, C4)
- **Role as consultant strengthened** (I42, Enterprise Architect, EA, C4)
- **PO as a contact person for EA** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)
- **Spatial networking** (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, EA, C1)
- **Stronger collaboration with other architects through community** (I42, Enterprise Architect, EA, C4)
- **Today: Responsibility for the domain** (I42, Enterprise Architect, EA, C4)

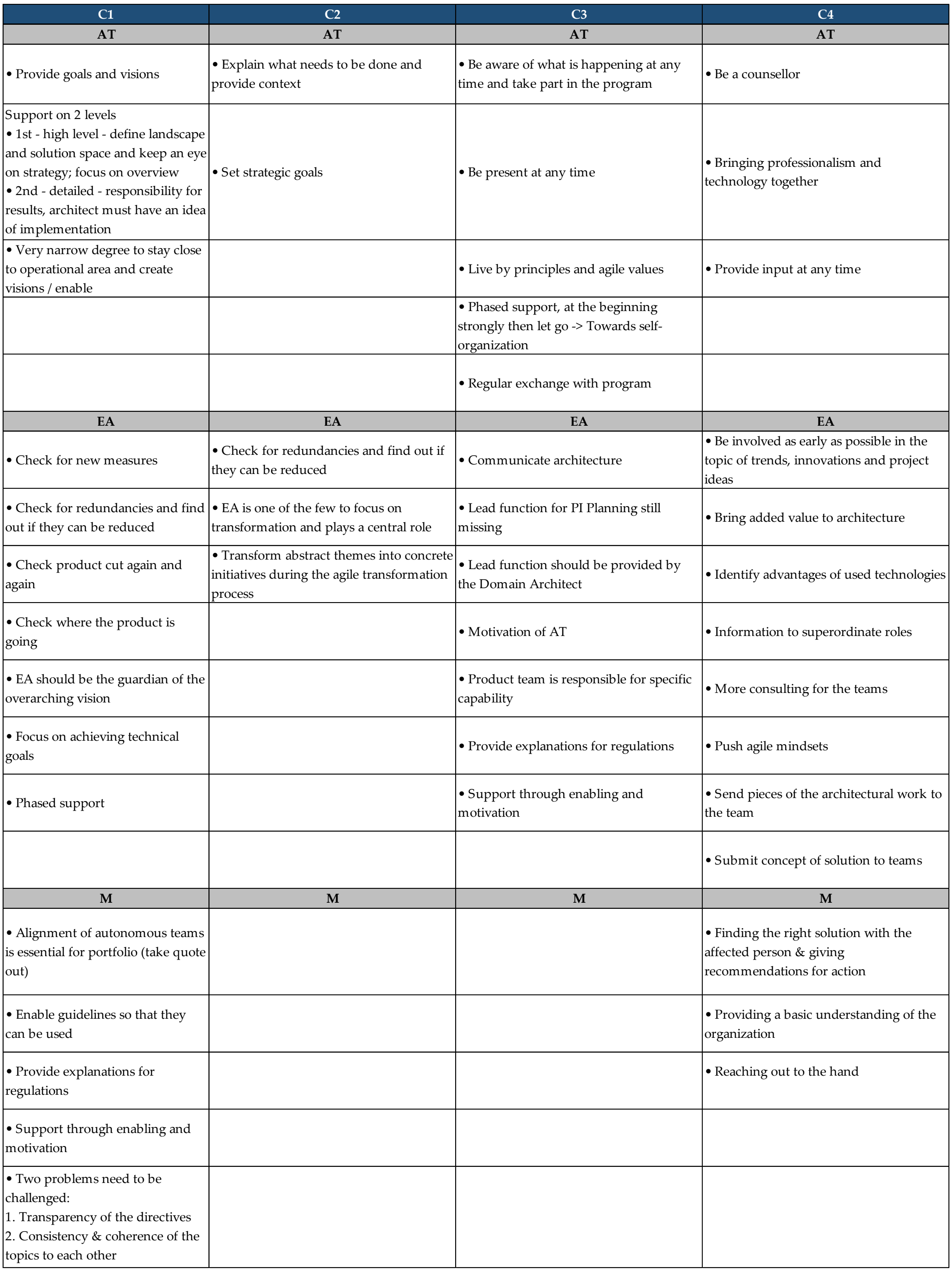
Stakeholder group M:

- **Architects must accompany projects much longer than before, as the main activity took place in the long conception phase** (I41, Enterprise Architect / Head of Department, EA, C4)
- **Architects must accompany projects much longer than before, as the main activity place in the long conception phase** (I41, Enterprise Architect / Head of Department, EA, C4)
- **Clear contact persons** (I31, Senior Manager, M, C3)
- **Deviations permitted if principles are adhered to** (I43, Head of Main Department, M, C4)
- **Domain structure also in business!** (I31, Senior Manager, M, C3)
- **Dynamism in the business environment is strongly technology-driven** (I15, Head of EAM, M, C1)

- **Easier to allocate stakeholders** (I31, Senior Manager, M, C3)
- **From project orientation to product orientation, less directly involved in the project** (I15, Head of EAM, M, C1)
- **Need for greater continuous collaboration within product teams** (I15, Head of EAM, M, C1)
- **Stronger business and IT alignment** (I31, Senior Manager, M, C3)
- **Target: Remove ivory tower, solution: Using the help of pattern creation based on principles** (I43, Head of Main Department, M, C4)

How should Enterprise Architects support agile programs and agile transformation?

Answers to the question, how an EA should support an agile program or transformation are listed in Table 4.23 below.



C1	C2	C3	C4
AT	AT	AT	AT
• Provide goals and visions	• Explain what needs to be done and provide context	• Be aware of what is happening at any time and take part in the program	• Be a counsellor
Support on 2 levels • 1st - high level - define landscape and solution space and keep an eye on strategy; focus on overview • 2nd - detailed - responsibility for results, architect must have an idea of implementation	• Set strategic goals	• Be present at any time	• Bringing professionalism and technology together
• Very narrow degree to stay close to operational area and create visions / enable		• Live by principles and agile values	• Provide input at any time
		• Phased support, at the beginning strongly then let go -> Towards self-organization	
		• Regular exchange with program	
EA	EA	EA	EA
• Check for new measures	• Check for redundancies and find out if they can be reduced	• Communicate architecture	• Be involved as early as possible in the topic of trends, innovations and project ideas
• Check for redundancies and find out if they can be reduced	• EA is one of the few to focus on transformation and plays a central role	• Lead function for PI Planning still missing	• Bring added value to architecture
• Check product cut again and again	• Transform abstract themes into concrete initiatives during the agile transformation process	• Lead function should be provided by the Domain Architect	• Identify advantages of used technologies
• Check where the product is going		• Motivation of AT	• Information to superordinate roles
• EA should be the guardian of the overarching vision		• Product team is responsible for specific capability	• More consulting for the teams
• Focus on achieving technical goals		• Provide explanations for regulations	• Push agile mindsets
• Phased support		• Support through enabling and motivation	• Send pieces of the architectural work to the team
			• Submit concept of solution to teams
M	M	M	M
• Alignment of autonomous teams is essential for portfolio (take quote out)			• Finding the right solution with the affected person & giving recommendations for action
• Enable guidelines so that they can be used			• Providing a basic understanding of the organization
• Provide explanations for regulations			• Reaching out to the hand
• Support through enabling and motivation			
• Two problems need to be challenged: 1. Transparency of the directives 2. Consistency & coherence of the topics to each other			

Table 4.23.: Support of EAs on agile programs and agile transformation

How is the role of the architect exercised in your company?

This case study distinguishes between 4 different roles of the architect (see Figure 4.8).

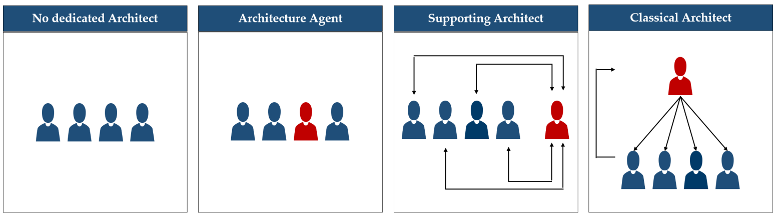


Figure 4.8.: Role of EAs

The Classical Architect embodies the old role of an Enterprise Architect with great decision-making power, system know-how instead of technical know-how, strict specifications, distance to the AT (ivory tower), system responsibility, a good overview of the overall organization and who mainly works with other architects. The Supporting Architect, on the other hand, is characterized in particular by the function of supporting and enabling ATs and working primarily in focus with the AT. The latter requires a high level of technical know-how and social skills and leads to greater proximity to the ATs. In addition, a supporting architect has less decision-making power, escalation possibilities and component rather than system responsibility. The Architectural Agent is the weakest form of EA with less functionality and responsibility than the supporting architect. If no expert with architectural know how is in the team, the whole team works as a team architect. In this case, there is no Dedicated Architect.

Role EA - Total (Descending)	Number of Companies
Supporting Architect	4
Classical Architect	4
Architectural Agent	2
No Named Architect	1

Table 4.24.: Role of EAs in case organizations

The role of the supporting and classical architect can be found in all four companies, see Table 4.24. It should be noted that different roles of an architect may exist in one company. For example, 95.24% chose the role of supporting architect, 38.10% the role of classical architect. This is primarily due to the current transformation process.

What are the characteristics of the role chosen in question 3.2?

The following Table 4.25 shows the characteristics of a supporting architect.

Supporting Architect
Able to code for short demonstration
Able to increase intrinsic motivation
Acts as a change agent
Clear responsibility
Component knowledge
Enable ATs
Guarantor of compliance with architectural principles
Less decision-making power
Methodically trained
More knowledge about tools and technology
Out of the box thinking
Participates in action
Professional competence
Provide context
Provide guideance
Proximity to the team
Social competence
Supporter and consultant
Understands vision and goals

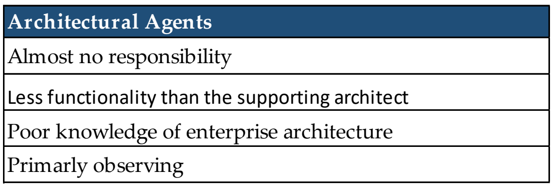
Table 4.25.: Characteristics Supporting Architect

Table 4.26 shows the characteristics of a classical architect.

Classical Architect
Governed
Has less insight (ivory tower)
High enterprise architecture know-how
High enterprise architecture methodology know-how
High process know-how and understanding

Table 4.26.: Characteristics Classical Architect

Table 4.27 shows the characteristics of an architectural agent.

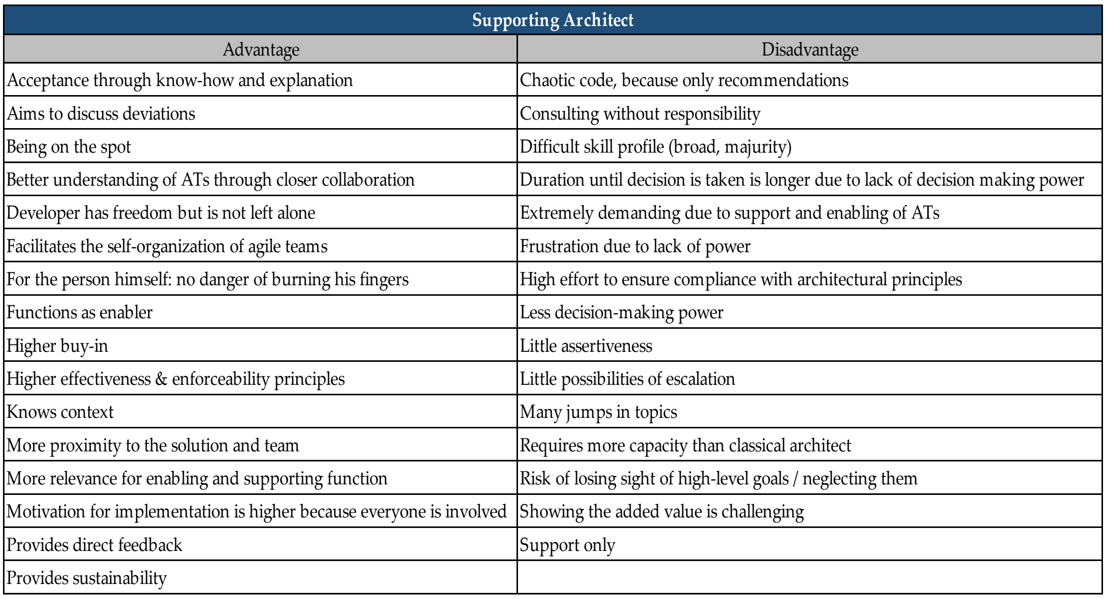


Architectural Agents
Almost no responsibility
Less functionality than the supporting architect
Poor knowledge of enterprise architecture
Primarly observing

Table 4.27.: Characteristics Architectural Agent

What are the advantages and disadvantages of the role mentioned in question 3.2?

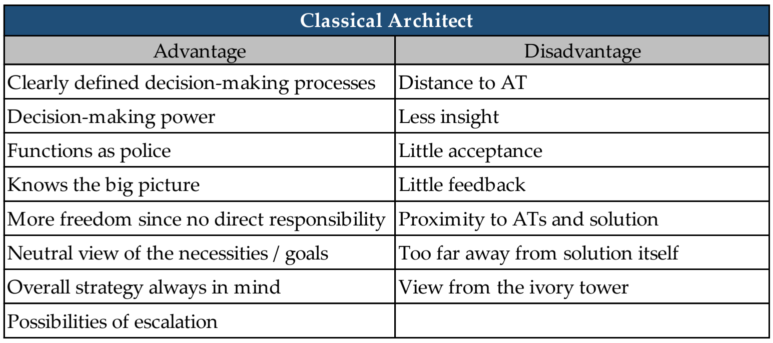
Table 4.28 shows the advantages and disadvantages of the role Supporting Architect.



Supporting Architect	
Advantage	Disadvantage
Acceptance through know-how and explanation	Chaotic code, because only recommendations
Aims to discuss deviations	Consulting without responsibility
Being on the spot	Difficult skill profile (broad, majurity)
Better understanding of ATs through closer collaboration	Duration until decision is taken is longer due to lack of decision making power
Developer has freedom but is not left alone	Extremely demanding due to support and enabling of ATs
Facilitates the self-organization of agile teams	Frustration due to lack of power
For the person himself: no danger of burning his fingers	High effort to ensure compliance with architectural principles
Functions as enabler	Less decision-making power
Higher buy-in	Little assertiveness
Higher effectiveness & enforceability principles	Little possibilities of escalation
Knows context	Many jumps in topics
More proximity to the solution and team	Requires more capacity than classical architect
More relevance for enabling and supporting function	Risk of losing sight of high-level goals / neglecting them
Motivation for implementation is higher because everyone is involved	Showing the added value is challenging
Provides direct feedback	Support only
Provides sustainability	

Table 4.28.: Advantage and Disadvantage of Supporting Architect

Table 4.29 shows the advantages and disadvantages of the role Classical Architect.



Classical Architect	
Advantage	Disadvantage
Clearly defined decision-making processes	Distance to AT
Decision-making power	Less insight
Functions as police	Little acceptance
Knows the big picture	Little feedback
More freedom since no direct responsibility	Proximity to ATs and solution
Neutral view of the necessities / goals	Too far away from solution itself
Overall strategy always in mind	View from the ivory tower
Possibilities of escalation	

Table 4.29.: Advantage and Disadvantage of Classical Architect

Which of the roles listed in question 3.2 should in your opinion be lived more strongly in the future?

100% of the interviewees are of the opinion that the role of the supporting architect should be more strongly lived in the future, however, only on the assumption that the role model changes somewhat. On the one hand, decision-making and control powers should be extended (I42, Enterprise Architect, EA, C4), methodological skills strengthened (I33, Scrum Master / Senior Agile Coach, AT, C3) and enablement increased (I42, Enterprise Architect, EA, C4). The latter requires in particular more technological know-how. Among other things, an EA should be able to program and thus provide hands-on assistance (I33, Scrum Master / Senior Agile Coach, AT, C3). The artifact provided by an EA should be implementable (I43, Head of Main Department, M, C4). In addition, when filling a position for an EA, social skills should receive greater priority (I43, Head of Main Department, M, C4). An EA must be able to deal well with people and conflict situations. The more self-assured and the better a decision is justified, the higher the intrinsic motivation of the developers (I43, Head of Main Department, M, C4). In addition, there is a desire to promote the community approach more strongly (I43, Head of Main Department, M, C4).

Should Enterprise Architects be part of the ATs?

> *"The one with a responsibility to deliver is part of the team"*
>
> *- I15, Head of EAM and QM, M, C1*

61.91% of respondents believe that an EA should not be part of an AT. This is justified by the overarching role. An EA must keep the overview. This can no longer be guaranteed if he is part of an AT. Furthermore, there is no capacity to assign an EA to every AT (I42, Enterprise Architect, EA, C4). Furthermore, code is not relevant in detail for an EA (I44, Developer, AT, C4). Domain and objective are more important than the exact solution (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect,

EA, C2). An EA lives from the up-front architecture and should pass it on to the ATs at the beginning of a project (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2). He is responsible for the how and where. The detailing is then the task of the AT (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2). Without an overarching role that coordinates the entire spectrum, chaos would arise (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, Enterprise Architect, Enterprise Architect, EA, C1).
9.52% of interviewees believe that an EA should be part of an AT. This is justified by a better insight (I36, Enterprise Architect 1, EA, C3). Among other things, this allows an EA to recognize which problems currently exist more quickly. This means that an EA can start directly at the right place and quickly drive the problem solution forward (I31, Senior Manager, M, C3).

28.57% of interviewees believe that an EA should be part of an AT in phases. At the beginning of a project/product a strong integration would be beneficial (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1, Enterprise Architect, Enterprise Architect, EA, C1). With increasing progress and the development of the necessary know-how, "keyword "enabling", in the team, the EA should gradually withdraw from the AT and only take on a guiding function (I17, Group Leader IT, AT, C1). In addition, the necessity of an EA in an AT was seen in particular on strategic topics (I43, Head of Main Department, M, C4). According to the respondents, there is less need for existing topics.

To what extent should Enterprise Architects be involved in the agile programs?

Case organization 1:
According to 87.5% of interviewees in case organization 1, EAs should be strongly involved in the agile programs in order to have an overview of the big picture and to recognize correlations and also, because the SAs sit in the ATs and have no time to keep the overview (I13, Scrum Master, Requirement Engineer, AT, C1). In addition, the role of the EA at program level is seen as a major lever for the implementation of architectural goals (I17, Group Leader IT, AT, C1).

100% of the EAs suggest a phased integration to avoid capacity problems. At the beginning the EA should be integrated strongly and with increasing progress become less and less integrated until only a guidance function remains.

12.5% of the respondents stated that integration should primarily take place at domain level. A domain contains several products and is referred to here as the strategic level. At the latter level, strategic decisions are made that require the overarching view of an EA (I15, Head of EAM and QM, M, C1). An integration at product level should only take place if necessary (I15, Head of EAM and QM, M, C1). In order to make this possible, the solution space with architecture principles to be adhered to and a selection of possible technologies should help to provide the necessary relief at product level and enable work at domain level (I15, Head of EAM and QM, M, C1).

Case organization 2:
In case organization 2, 100% of the interviewees state, that EAs should be strongly involved in the agile programs. The interviewees argued that despite strong involvement, a lot of responsibility should be passed on to the teams. An EA should be able to "let go" (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2). The definition of roles should also change somewhat in the future. An EA should be more strongly integrated into the processes and focus the interfaces to the PO more strongly (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2).

Case organization 3:
In case organization 3, the response varies from phased to very strong to complete. However, all interviewees agree that the presence of the EA must always be felt. Every member of a development alliance must know the EA. One respondent made the following statement:

> *"If a developer can name the Enterprise Architect, then the integration has reached the right degree"*
>
> *-I33, Scrum Master / Senior Agile Coach, AT, C3*

The interviewees also stated that despite strong integration, a certain distance is also important. This is due to the fact that an EA is "the bigger system" and must not lose sight of the whole.

100% of the EAs proposed to integrate the EA in phases at the beginning of a project, since important architectural decisions are made particularly in the conception phase (I32, Enterprise Architect 2, EA, C3). After that, a selective integration should take place for more complex requirements in order to counteract a capacity problem (I36, Enterprise Architect 1, EA, C3).

100% of the interviewees from stakeholder group M are of the opinion that an EA should be very strongly integrated. This was justified in particular by the transformation currently taking place. Since the company is still in the coining phase, everything is still "quite new" and there is a lack of experience at all ends (I31, Senior Manager, M, C3).

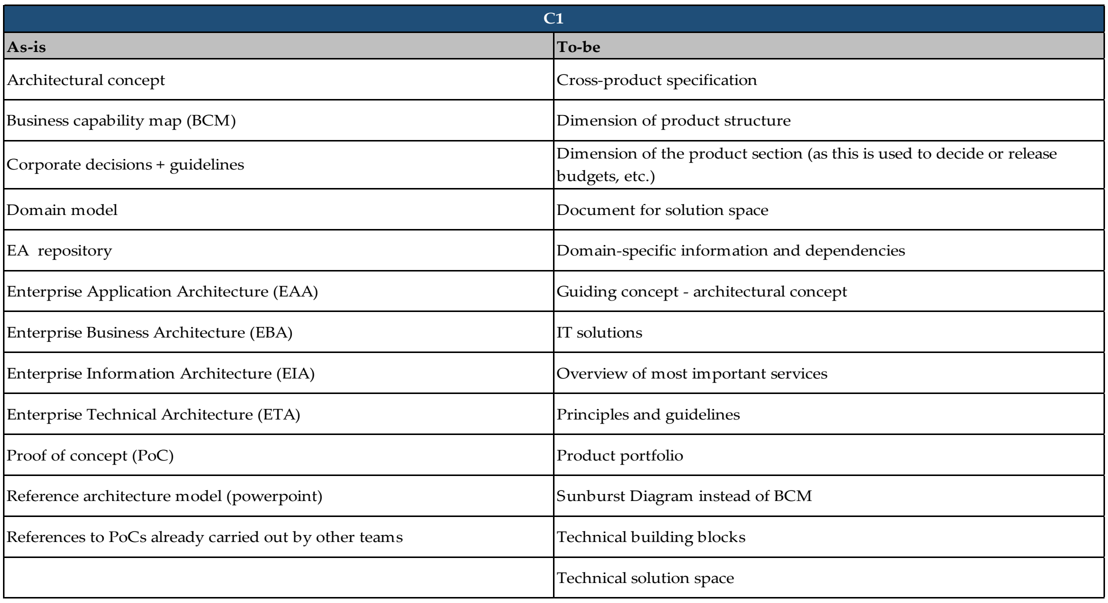
Case organization 4:
In case organization 4, 100% of the interviewees state, that Enterprise Architects should be strongly involved in the agile programs. A distinction is made here according to topic and ability (I43, Head of Main Department, M, C4). For strategic topics, an EA

should be fully integrated into the AT, which means the EA should be part of the AT (I43, Head of Main Department, M, C4). For already existing topics, the EA should only take on an advisory function, without being part of the AT (I43, Head of Main Department, M, C4). In general, the more innovative and new a program, the earlier the EA should be integrated (I42, Enterprise Architect, EA, C4). Here, communication and collaboration play an important role (I44, Developer, AT, C4). The architecture community consists of EA, SA and DA (I42, Enterprise Architect, EA, C4). In the future, this community should become the hub for architectural decisions and thus increase integration (I42, Enterprise Architect, EA, C4). In addition, EAs should be involved more often in planning and objectives and accompany the program with an advisory function (I41, Enterprise Architect / Head of Department, EA, C4).

Which artifacts are provided by the Enterprise Architects to the stakeholders in an agile environment? What artifacts should be provided by Enterprise Architects to stakeholders in the future?

The following section shows artifacts that are provided by the EAs to the stakeholders in an agile environment per case organization.

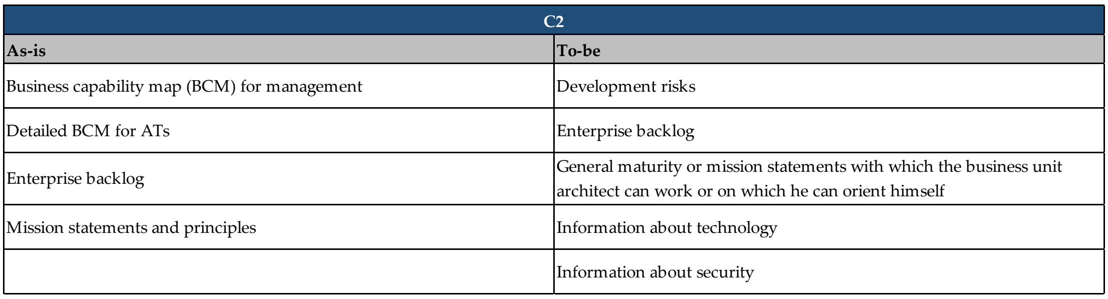
Case organization 1:
Table 4.30 lists artifacts that are and should be provided by EAs to ATs of case organization 1.



C1	
As-is	**To-be**
Architectural concept	Cross-product specification
Business capability map (BCM)	Dimension of product structure
Corporate decisions + guidelines	Dimension of the product section (as this is used to decide or release budgets, etc.)
Domain model	Document for solution space
EA repository	Domain-specific information and dependencies
Enterprise Application Architecture (EAA)	Guiding concept - architectural concept
Enterprise Business Architecture (EBA)	IT solutions
Enterprise Information Architecture (EIA)	Overview of most important services
Enterprise Technical Architecture (ETA)	Principles and guidelines
Proof of concept (PoC)	Product portfolio
Reference architecture model (powerpoint)	Sunburst Diagram instead of BCM
References to PoCs already carried out by other teams	Technical building blocks
	Technical solution space

Table 4.30.: Artifacts (as-is and to-be) of case organization 1

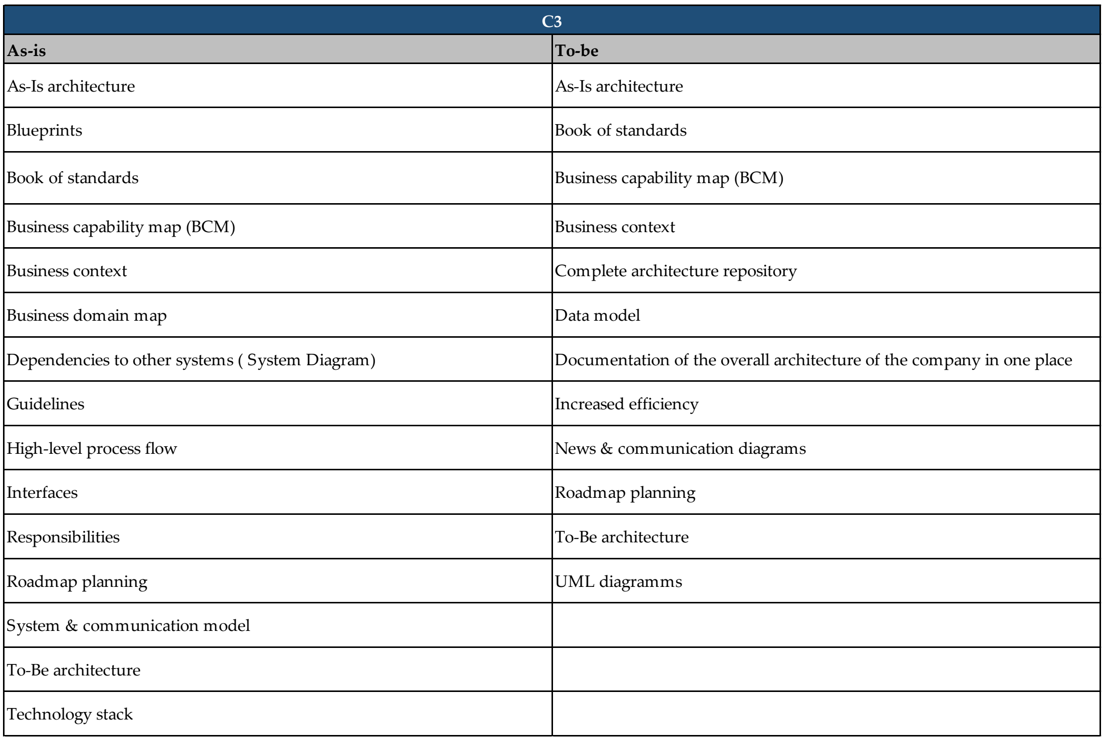
Case organization 2:
Table 4.31 lists artifacts that are and should be provided by EAs to ATs of case organization 2.



C2	
As-is	**To-be**
Business capability map (BCM) for management	Development risks
Detailed BCM for ATs	Enterprise backlog
Enterprise backlog	General maturity or mission statements with which the business unit architect can work or on which he can orient himself
Mission statements and principles	Information about technology
	Information about security

Table 4.31.: Artifacts (as-is and to-be) of case organization 2

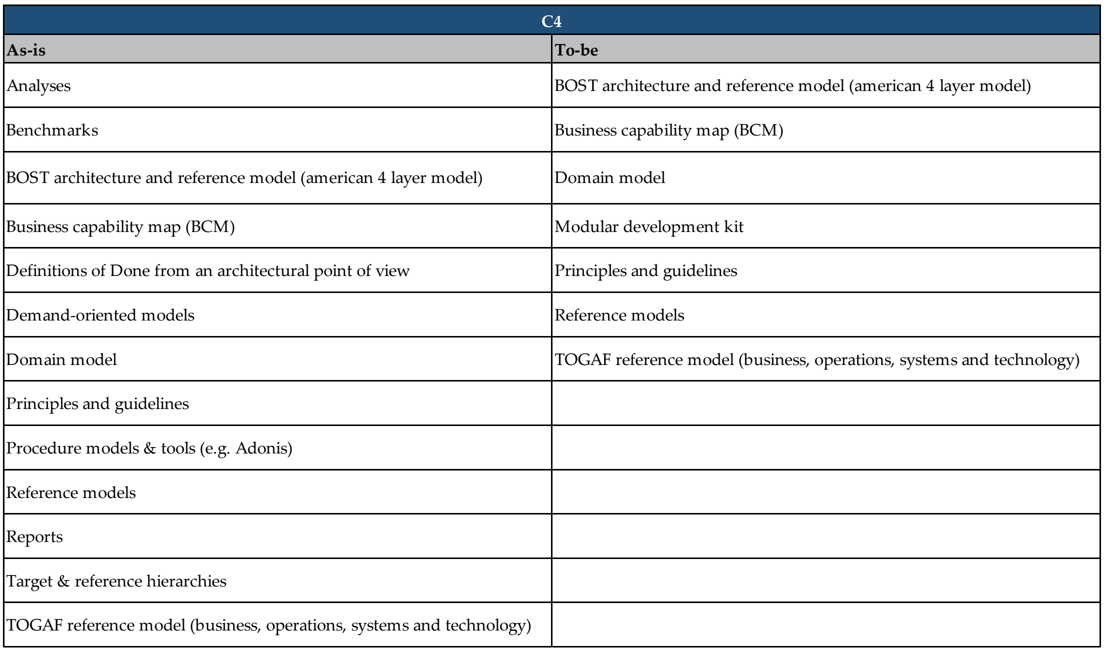
Case organization 3:
Table 4.32 lists artifacts that are and should be provided by EAs to ATs of case organization 3.



C3	
As-is	**To-be**
As-Is architecture	As-Is architecture
Blueprints	Book of standards
Book of standards	Business capability map (BCM)
Business capability map (BCM)	Business context
Business context	Complete architecture repository
Business domain map	Data model
Dependencies to other systems (System Diagram)	Documentation of the overall architecture of the company in one place
Guidelines	Increased efficiency
High-level process flow	News & communication diagrams
Interfaces	Roadmap planning
Responsibilities	To-Be architecture
Roadmap planning	UML diagramms
System & communication model	
To-Be architecture	
Technology stack	

Table 4.32.: Artifacts (as-is and to-be) of case organization 3

Case organization 4:
Table 4.33 lists artifacts that are and should be provided by EAs to ATs of case organization 4.

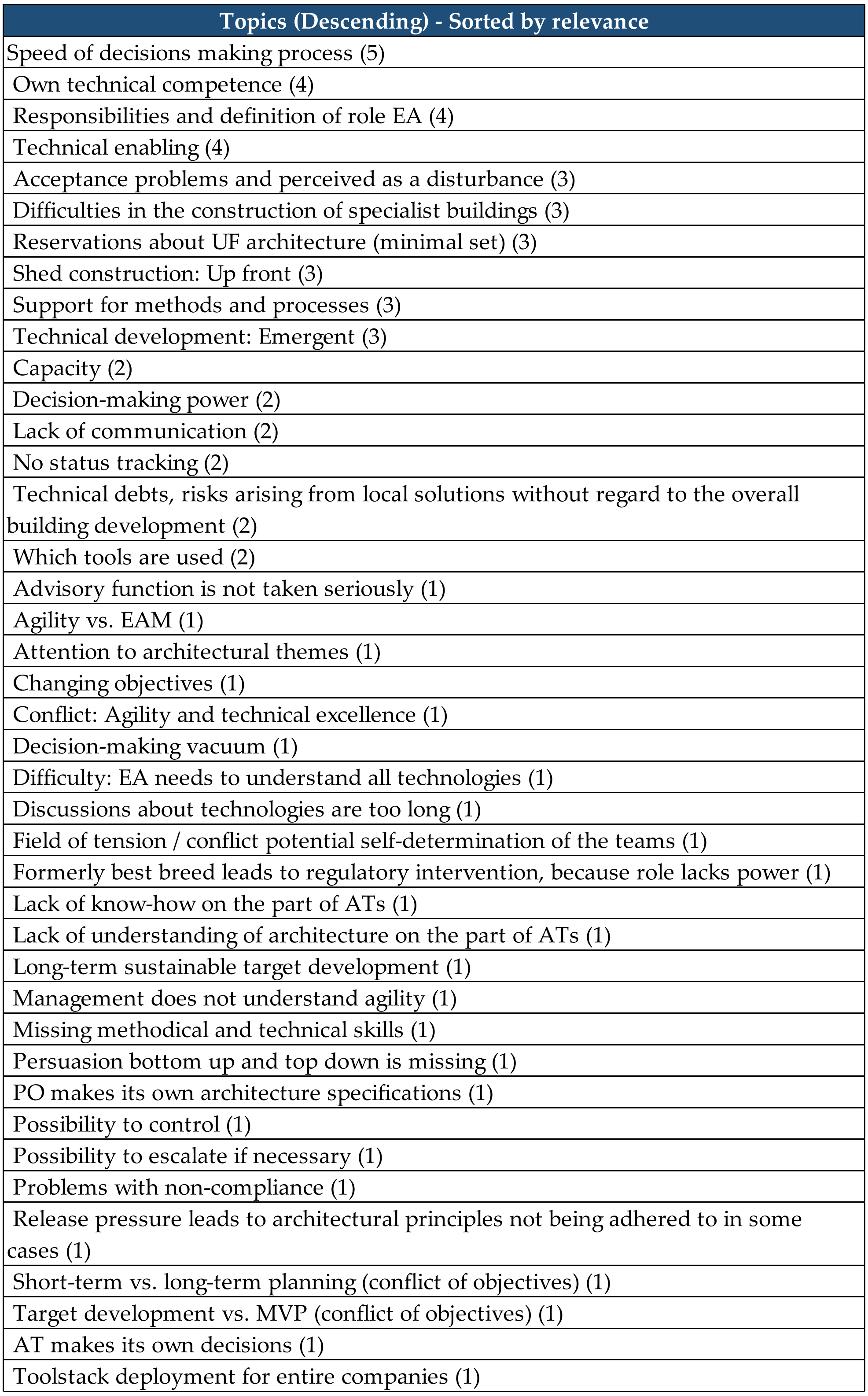


C4	
As-is	**To-be**
Analyses	BOST architecture and reference model (american 4 layer model)
Benchmarks	Business capability map (BCM)
BOST architecture and reference model (american 4 layer model)	Domain model
Business capability map (BCM)	Modular development kit
Definitions of Done from an architectural point of view	Principles and guidelines
Demand-oriented models	Reference models
Domain model	TOGAF reference model (business, operations, systems and technology)
Principles and guidelines	
Procedure models & tools (e.g. Adonis)	
Reference models	
Reports	
Target & reference hierarchies	
TOGAF reference model (business, operations, systems and technology)	

Table 4.33.: Artifacts (as-is and to-be) of case organization 4

What problems do Enterprise Architects encounter in an agile environment?

In the following, a list of the problems experienced by EAs in an agile environment is given (see Table 4.34). The problems are sorted by relevance and listed in descending order. The number in the brackets shows how often the problem has been stated.



Topics (Descending) - Sorted by relevance
Speed of decisions making process (5)
Own technical competence (4)
Responsibilities and definition of role EA (4)
Technical enabling (4)
Acceptance problems and perceived as a disturbance (3)
Difficulties in the construction of specialist buildings (3)
Reservations about UF architecture (minimal set) (3)
Shed construction: Up front (3)
Support for methods and processes (3)
Technical development: Emergent (3)
Capacity (2)
Decision-making power (2)
Lack of communication (2)
No status tracking (2)
Technical debts, risks arising from local solutions without regard to the overall building development (2)
Which tools are used (2)
Advisory function is not taken seriously (1)
Agility vs. EAM (1)
Attention to architectural themes (1)
Changing objectives (1)
Conflict: Agility and technical excellence (1)
Decision-making vacuum (1)
Difficulty: EA needs to understand all technologies (1)
Discussions about technologies are too long (1)
Field of tension / conflict potential self-determination of the teams (1)
Formerly best breed leads to regulatory intervention, because role lacks power (1)
Lack of know-how on the part of ATs (1)
Lack of understanding of architecture on the part of ATs (1)
Long-term sustainable target development (1)
Management does not understand agility (1)
Missing methodical and technical skills (1)
Persuasion bottom up and top down is missing (1)
PO makes its own architecture specifications (1)
Possibility to control (1)
Possibility to escalate if necessary (1)
Problems with non-compliance (1)
Release pressure leads to architectural principles not being adhered to in some cases (1)
Short-term vs. long-term planning (conflict of objectives) (1)
Target development vs. MVP (conflict of objectives) (1)
AT makes its own decisions (1)
Toolstack deployment for entire companies (1)

Table 4.34.: Problems experienced by EAs in an agile environment

How should these problems be addressed in your opinion?

In the following, problems and possible solutions of every case organization are listed in Tables 4.35, 4.36, 4.37, and 4.38.

Problems and Solutions (Descending) - Relevance	C1		C2		C3		C4	
	Relevance	Solution	Relevance	Solution	Relevance	Solution	Relevance	Solution
Balancing short-term vs long-term planning (conflict of objectives)	+	• Following product orientation • Greater involvement of ATs through stronger collaboration	+		+	• Establishing a culture based on principles of grassroots democracy	+	• Greater involvement of ATs through stronger collaboration • Organizing community of practice for architecture (architecture guilds)
Demonstrating the value of EAM			+	• Raising awareness at all organizational levels	+	• Organizing community of practice for architecture (architecture guilds)	+	• Raising awareness at all organizational levels
Dealing with loss of decision-making power					+	• Conducting architectural spikes • Establishing a culture based on principles of grassroots democracy • Organizing community of practice for Architecture (architecture guilds) • Taking highest paid person's opinion	+	• Greater involvement of ATs through stronger collaboration • Organizing community of practice for architecture (architecture guilds)
Dealing with technical implementation and repective technology trends	+	• Conducting architectural spikes • Establishing innovation and planning iterations • Phasing in with technical implementation and high-level architecture					+	• Raising awareness at all organizational levels
Defining clear roles and responsiblities	+	• Applying AKV			+	• Defining a collaboration model • Using RACI		
Defining end to end responsibilities	+	• Following product orientation	+	• Applying Dev ops				
Ensuring that ATs adhere to architecture principles					+	• Adding architecture attributes to the Definition of Done and Definition of Ready • Defining quality gates • Establishing a collaborative approach for architecture principles and guidelines • Greater involvement of ATs through stronger collaboration • Implementing continuous delivery pipeline	+	• Raising awareness at all organizational levels
Fostering technical excellence					+		+	• Greater involvement of ATs through stronger collaboration • Organizing community of practice for architecture (architecture guilds)

Table 4.35.: Problems and solutions per case organizations (1/4)

Problems and Solutions (Descending) - Relevance	C1		C2		C3		C4	
	Relevance	Solution	Relevance	Solution	Relevance	Solution	Relevance	Solution
Managing technical debts	+	• Following product orientation	+					
Providing ATs appropriate tools, methods and processes			+	• Establishing EAM knowledge base	+	• Implementing continuous delivery pipeline		
Avoiding big design upfront			+	• Using agile modelling				
Balancing upfront and emerging architecture			+	• Creating architecture enablers				
Creating a comprehensive overview of the tool landscape	+						+	
Creating a proper upfront plan	+	• Following product orientation						
Creating proper upfront architecture design of the system			+	• Implementing architectural runway				
Dealing with frequency of architectural decisions			+	• Providing processes and tools that foster faster decisions				

Table 4.36.: Problems and solutions per case organizations (2/4)

Problems and Solutions (Descending) - Relevance	C1		C2		C3		C4	
	Relevance	Solution	Relevance	Solution	Relevance	Solution	Relevance	Solution
Dealing with incorrect practices of agile development							+	• Greater involvement of ATs through stronger collaboration • Organizing community of practice for architecture (architecture guilds)
Dealing with turbulent environment	+	• Following product orientation						
Developing an evolvable and sustainable long-term target architecture	+							
Empowering ATs to make decisions					+	• Applying AKV		
Enabling technical competence	+	• Applying AKV						
Enhancing speed of architectural decision-making	+	• Using face-to-face communication						
Establishing common understanding of agile thinking and practices					+	• Prefering open communication		
Establishing lightweight review process for adapting new technologies	+							

Table 4.37.: Problems and solutions per case organizations (3/4)

Problems and Solutions (Descending) - Relevance	C1		C2		C3		C4	
	Relevance	Solution	Relevance	Solution	Relevance	Solution	Relevance	Solution
Facilitating communication between EAs und ATs							+	• Raising awareness at all organizational levels
Facilitating prototypical implementations	+							
Finding the right balance between architecture improvement and business value					+	• Greater involvement of ATs through stronger collaboration • Organizing community of practice for architecture (architecture guilds)		
Missing methodical & technical skills					+	• Organizing workshops and trainings		
Providing ATs approriate architecture artifacts with suitable depth of detail					+	• Don't build an ivory tower		
Reliefing EAs due to system cut into many subsystems	+	• Increasing and optimizing capacity of EAs						
Tracking progress of ATs	+	• Increasing and optimizing capacity of EAs • Involving architects and architecural activities at program level						

Table 4.38.: Problems and solutions per case organizations (4/4)

What recommendations would you give Enterprise Architects in an agile environment?

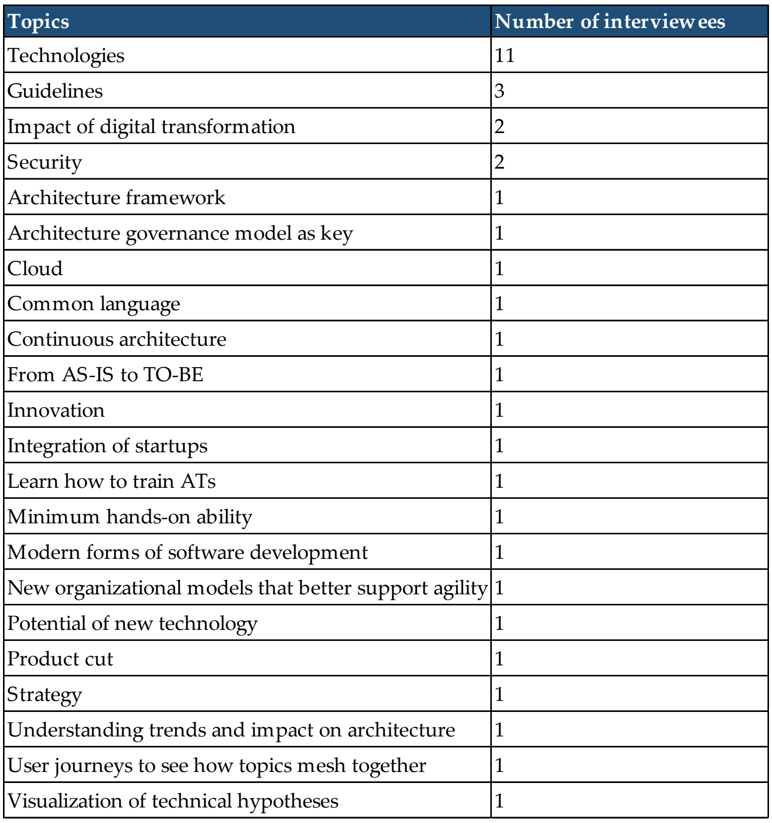
In the following, a list of the recommendations for EAs in an agile environment is given. The recommendations are sorted by relevance and listed in descending order.

- **Enter the teams** (I17, Group Leader IT, AT, C1; I37, Chief Scrum Master, AT, C3; I36, Enterprise Architect 1, EA, C3)
- **Assertiveness is important** (I13, Scrum Master, Requirement Engineer, AT, C1; I33, Scrum Master / Senior Agile Coach, AT, C3)
- **Basic professional and technical understanding is mandatory** (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1)
- **Communication is key for success** (I37, Chief Scrum Master, AT, C3; I41, Enterprise Architect / Head of Department, EA, C4)
- **Have perseverance with support**(I13, Scrum Master, Requirement Engineer, AT, C1; I33, Scrum Master / Senior Agile Coach, AT, C3)
- **Have stamina with support**(I13, Scrum Master, Requirement Engineer, AT, C1; I33, Scrum Master / Senior Agile Coach, AT, C3)
- **Proactively demonstrate the benefits of EA** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2; I44, Developer, AT, C4)
- **Look at the code** (I11, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1)
- **Traceability and transparency are very important to increases acceptance and intrinsic motivation** (I23, Product Owner, AT, C2; I33, Scrum Master / Senior Agile Coach, AT, C3)
- **Addressing problems in the community and winning comrades-in-arms, increasing sensitivity in management** (I42, Enterprise Architect, EA, C4)
- **Attend events often** (I44, Developer, AT, C4)
- **Authenticity and sufficient level of traction** (I15, Head of EAM, M, C1)
- **Ambitiousness** (I33, Scrum Master / Senior Agile Coach, AT, C3)
- **Part of planning's, grooming's** (I36, Enterprise Architect 1, EA, C3)
- **Common language** (I42, Enterprise Architect, EA, C4)
- **Demonstrating the usefulness of the artifacts leads to increased productivity** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)
- **Enable overall entrepreneurial optimum** (I15, Head of EAM, M, C1)
- **Enable the ATs** (I41, Enterprise Architect / Head of Department, EA, C4)
- **Experience is indispensable and makes it easy to assess problems** (I43, Head of Main Department, M, C4)
- **Find degree between operative and vision** (I13, Scrum Master, Requirement Engineer, AT, C1)

- **Move away from the ivory tower** (I36, Enterprise Architect 1, EA, C3)
- **Increase awareness** (I44, Developer, AT, C4)
- **Insist on getting feedback** (I37, Chief Scrum Master, AT, C3)
- **Key: Acceptance at management levels** (I41, Enterprise Architect / Head of Department, EA, C4)
- **No PowerPoint architecture** (I15, Head of EAM, M, C1)
- **Nothing to blandish leads to truth** (I33, Scrum Master / Senior Agile Coach, AT, C3)
- **Pay attention to the choice of words, the type of communication is very important** (I41, Enterprise Architect / Head of Department, EA, C4)
- **Seek cooperation with the AT** (I37, Chief Scrum Master, AT, C3)
- **Understand the things you are recommending** (I43, Head of Main Department, M, C4)
- **Should have worked agile once** (I43, Head of Main Department, M, C4)

On which topics should Enterprise Architects focus in the future?

In the following, the topics that an EA should deal with in the future are listed, see Table 4.39. A multiple selection was possible.



Topics	Number of interviewees
Technologies	11
Guidelines	3
Impact of digital transformation	2
Security	2
Architecture framework	1
Architecture governance model as key	1
Cloud	1
Common language	1
Continuous architecture	1
From AS-IS to TO-BE	1
Innovation	1
Integration of startups	1
Learn how to train ATs	1
Minimum hands-on ability	1
Modern forms of software development	1
New organizational models that better support agility	1
Potential of new technology	1
Product cut	1
Strategy	1
Understanding trends and impact on architecture	1
User journeys to see how topics mesh together	1
Visualization of technical hypotheses	1

Table 4.39.: Future topics an EA should pay attention to

52.38% of interviewees stated that an EA should focus primarily on technologies in the future. 14.29% stated that guidelines should be given greater prominence. 9.5% pointed to the impact of digital transformation and security.

4.5. Value Contribution of Enterprise Architecture Management

What information do Enterprise Architects need from ATs to provide appropriate architectural models?

In the following, the information EAs need from ATs to provide appropriate architecture models is shown. The information is sorted by relevance and displayed in descending order.

- **Infrastructure components, technologies and tools** (I13, Scrum Master, Requirement Engineer, AT, C1; I17, Group Leader IT, AT, C1; I32, Enterprise Architect 2, EA, C3; I44, Developer, AT, C4; I42, Enterprise Architect, EA, C4)

- **To-be architecture** (I12, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1; I23, Product Owner, AT, C2; I35, Scrum Master / Senior Agile Coach, AT, C3; I31, Senior Manager, M, C3)
- **Interfaces and dependencies** (I12, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1; I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2; I44, Developer, AT, C4)
- **Product description and vision** (I15, Head of EAM, M, C1; I35, Scrum Master / Senior Agile Coach, AT, C3; I36, Enterprise Architect 1, EA, C3; I32, Enterprise Architect 2, EA, C3)
- **As-Is architecture** (I35, Scrum Master / Senior Agile Coach, AT, C3; I36, Enterprise Architect 1, EA, C3; I32, Enterprise Architect 2, EA, C3)
- **Information objects** (I12, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1; Enterprise Architects, EA, C1; I36, Enterprise Architect 1, EA, C3; I42, Enterprise Architect, EA, C4;)
- **Reusability of building blocks and functionality** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2; I41, Enterprise Architect / Head of Department, EA, C4)
- **Risks of technical debts** (I23, Product Owner, AT, C2; I36, Enterprise Architect 1, EA, C3)
- **Technical framework conditions of the systems** (I37, Chief Scrum Master, AT, C3; I43, Head of Main Department, M, C4)
- **User stories, use cases, epics** (I36, Enterprise Architect 1, EA, C3; I43, Head of Main Department, M, C4)
- **Acceptance criteria of software** (I36, Enterprise Architect 1, EA, C3)
- **Added value for the case organization** (I23, Product Owner, AT, C2)
- **Applications affected** (I36, Enterprise Architect 1, EA, C3)
- **Architecture framework** (I36, Enterprise Architect 1, EA, C3)
- **Buffer in architecture design to allow changes** (I36, Enterprise Architect 1, EA, C3)
- **Business driver affected** (I36, Enterprise Architect 1, EA, C3)
- **Business objects affected** (I36, Enterprise Architect 1, EA, C3)
- **Business processes affected** (I31, Senior Manager, M, C3)
- **Clarification of weak points** (I35, Scrum Master / Senior Agile Coach, AT, C3)
- **Components affected** (I36, Enterprise Architect 1, EA, C3)
- **Data objects affected** (I36, Enterprise Architect 1, EA, C3)
- **Data volume needed** (I17, Group Leader IT, AT, C1)
- **Dependencies on other programs** (I23, Product Owner, AT, C2)
- **Information flows affected** (I36, Enterprise Architect 1, EA, C3)
- **Nonfunctional requirements** (I36, Enterprise Architect 1, EA, C3)
- **Roles and permissions** (I41, Enterprise Architect / Head of Department, EA, C4)

- **Services needed** (I36, Enterprise Architect 1, EA, C3)
- **Team organization** (I36, Enterprise Architect 1, EA, C3)
- **Value stream** (I36, Enterprise Architect 1, EA, C3)

Interestingly, although the information needed vary greatly, the focus is clearly on technical information followed by context information with product description and visions.

Which architectural models are provided to the ATs by the Enterprise Architects?

Table 4.40 shows the architectural models provided to ATs by
the EAs per case organization.

C1	C2	C3	C4
Application landscape	Business capability map	Activity diagram	Business capability map
Architecture blueprints		Application landscape	Business domain map
Architecture patterns		Architecture blueprints	Class models for overall architecture
Business capability map		Architecture spikes	Data model
Communication diagram		Business capability map	Framework for Process Modeling
Infrastructure components		Business domain map	Guidelines
Integrated architecture framework		Business driver	Layer model for software architecture
Interface diagrams		Business object model	Process models based on BPN 2.0
Landscape / Solution space		Component diagram	Reference architecture
Product structure		Data architecture	Technical architectural model
Programming languages		Function application / component matrix	
Reference architecture		Functional development planning	
Software, technologies and tools		Guidelines	
Sunburst diagramm		High-level process flow	
		Impact business domain map	
		Information flow overview	
		Information system architecture	
		Integration architecture	
		Interface model	
		List of project or component specific non functional requirements	
		Microservice and business function	
		Package diagram	
		Process application / component matrix	
		Process function matrix	
		Process models based on BPN 2.0	
		RACI matrix	
		Reference architecture	
		System and communication architecture	
		Technology landscape	
		Use case overview	

Table 4.40.: Architectural models provided to the ATs by EAs

As shown in Table 4.40, the number of different models varies greatly across all case organizations. Only the BCM is provided to the ATs in all case organizations. 75% of the case organizations provide architectural patterns, reference architectures and specific technology stacks. 50% provide architecture blueprints, a business domain map, data, interface and process models. Just to name the most important ones.

What do ATs expect from Enterprise Architects' architectural models?

Table 4.41 shows the expectations of the ATs from the EAs' architectural models. For example: "Availability" was mentioned throughout all case organizations. More precisely, summarized from all case organizations, two interviewees from the stakeholder group AT, four interviewees from the stakeholder group EA and one interviewee from the stakeholder group M mentioned "Availability".

AT Expectation - Total (Descending)	Number of Companies	AT	EA	M
Availability	4	2	4	1
Binding force	4	5	6	0
Quality	4	3	4	1
Relevance	4	3	10	2
Added value and effectiveness for short-term planning	3	0	1	0
Applicability and implementability	3	0	2	1
Level of detail	3	2	2	1
Recommendations for action, setting the framework	2	2	1	0
Scoping of products	2	1	2	0
Provide context	2	0	2	0
Alignment to domains	1	0	0	1
Common language	1	0	0	1
Reliability of the model	1	0	3	0
The opinion of the team should be reflected in the proposed solution	1	0	1	0
Transparency of the overall architecture	1	1	0	0

Table 4.41.: Expectations of ATs from EAs' architectural models

As shown in Table 4.41, 100% of the case organizations state that availability, binding force, quality and relevance of architecture models are important. 75% of the case organizations named added value and effectiveness for short-term planning, applicability and implementability, and level of detail as important.

Is it difficult to find an Enterprise Architect as a contact person?

Figure 4.9 shows how difficult finding an EA as a contact person is.

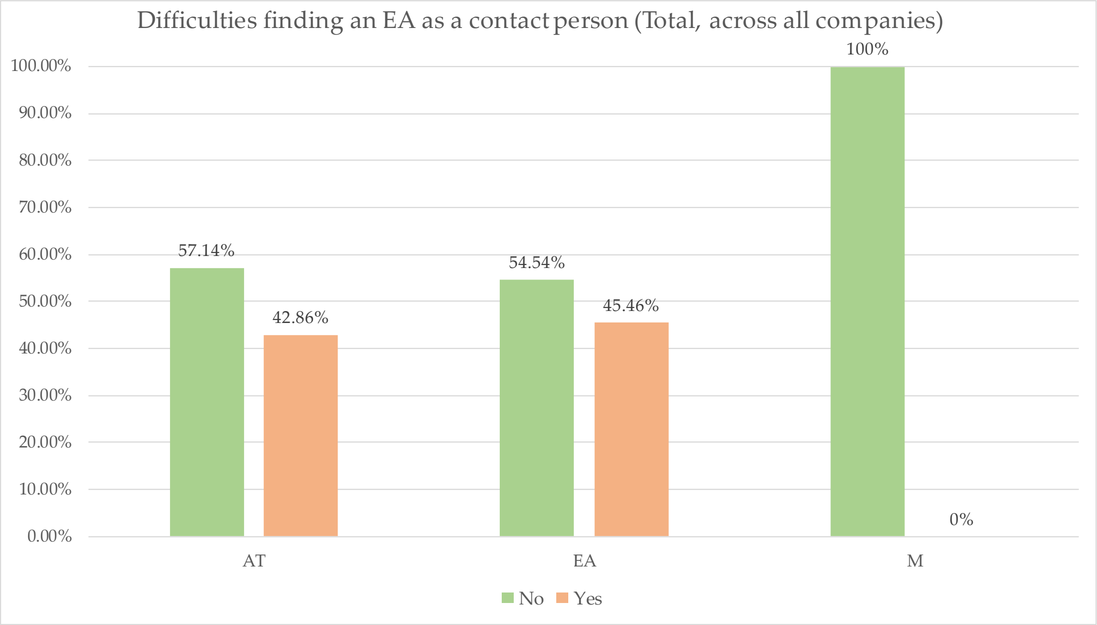


Figure 4.9.: Difficulties finding an EA as a contact person

42.86% of the stakeholder group AT state that finding an EA as a contact is difficult. 57.14% have no problems finding an EA. A similar ratio can be found in the stakeholder group EA. Here, 45.46% consider it difficult to find an EA as a contact person. This is the highest score among all stakeholder groups interviewed. Worth noting is that the results of the two stakeholder groups, AT and EA, are very similar. Managers, on the other hand, are 100% convinced that there are no difficulties in finding an EA as a contact person.

How do Enterprise Architects communicate with ATs?

Case organization 1:
In case organization 1, direct communication between EAs and ATs takes place (see Figure 4.10). All common communication media such as email and telephone are used (I13, Scrum Master, Requirement Engineer, AT, C1). Important topics are documented in Confluence (I16, Enterprise Architect, Enterprise Architect, EA, C1). Depending on the topic and the people involved, certain concerns that cannot be clarified face-to-face are brought into the community of practice architecture (CoPA) (participants: architects) and discussed and resolved together (I17, Group Leader IT, AT, C1). Then, the

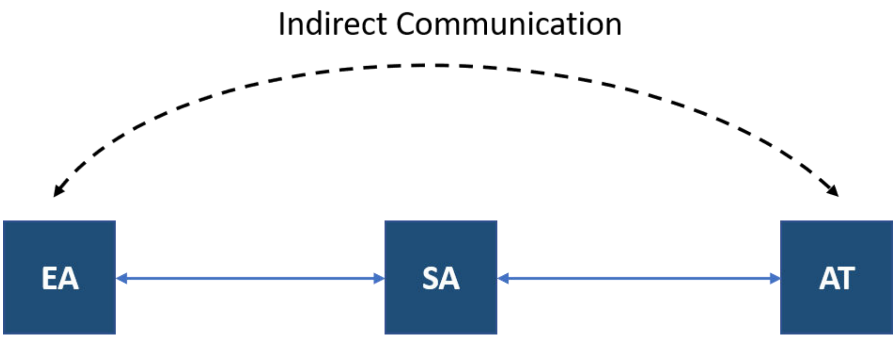


Figure 4.10.: Communication between EAs and ATs in case organization 1

solution architects carry the discussed topics into the AT (I12, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1). All interviewees confirmed that face-to-face communication takes place frequently and increases productivity.

Case organization 2:
In case organization 2, primarily indirect communication takes place between EAs and ATs, i.e. discussed elements are not directly communicated, see Figure 4.11.

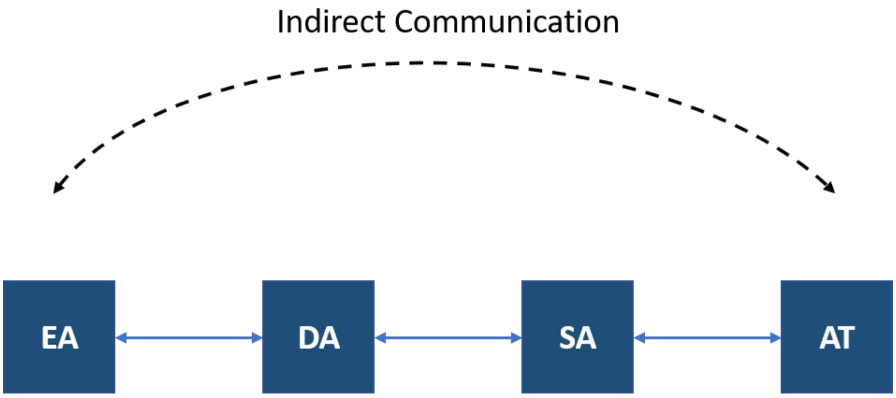


Figure 4.11.: Communication between EAs and ATs in case organization 2

As shown in Figure 4.11, communication between the EA and the AT takes place, for example, with the help of the DA and the SA (I23, Product Owner, AT, C2). This means that there is no direct connection between the EA, the AT and the SA. Once a month, a meeting with domain architects takes place (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2). Furthermore, according to the stakeholder group EA, ad-hoc meetings are possible by telephone or e-mail (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2). Since the EAM was first introduced only a year ago, the case organization is still in the finding phase (I23, Product Owner, AT, C2). This characterizes a clear capacity problem. A stronger involvement or direct communication is therefore not possible at the moment (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2; I23, Product Owner, AT, C2).

Case organization 3:
In case organization 3, both indirect and direct communication takes place between EAs and ATs (see Figure 4.12. The former takes place almost exclusively through solution and domain architects (I35, Scrum Master / Senior Agile Coach, AT, C3). According to the interviewees, the flow of information is strongly driven top-down (I37, Chief Scrum Master, AT, C3). In addition to the possibility of exchanging information face-to-face, a weekly meeting with the CoPA takes place (I32, Enterprise Architect 2, EA, C3; I37, Chief Scrum Master, AT, C3). Only the architects of a product cluster (=domain) are involved in that meeting. Among these are: domain architects, EA, and solution architects (I37, Chief Scrum Master, AT, C3). Participation is binding (I36, Enterprise Architect 1, EA, C3). According to the stakeholder group EA for DA and SA, there is also the possibility to schedule a ”one-to-one” with an EA on a weekly basis (I32, Enterprise Architect 2, EA, C3). Neither the stakeholder group AT nor the stakeholder group M were aware that this possibility exists.

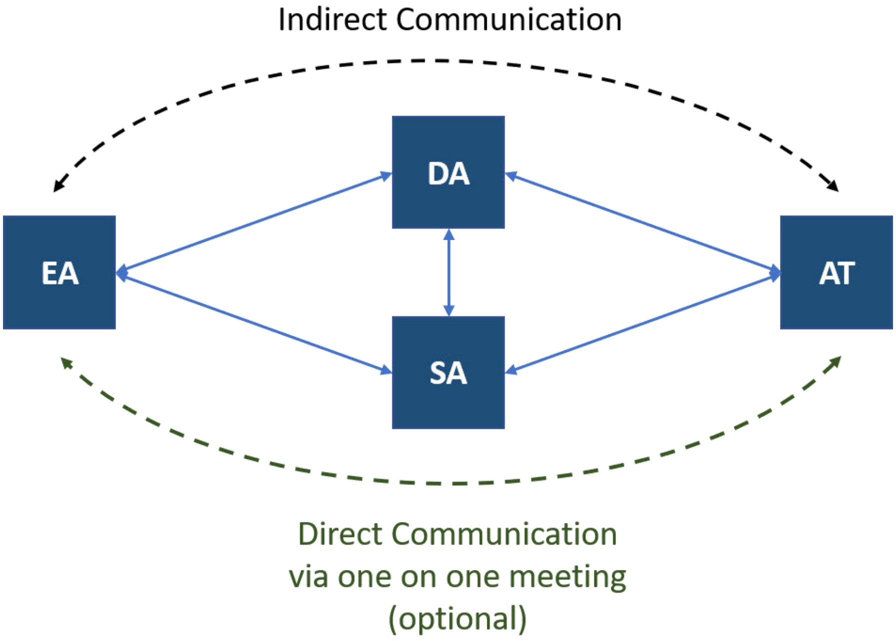


Figure 4.12.: Communication between EAs and ATs in case organization 3

Exceptions are projects in which all project members sit in the same building. Co-location was mentioned as a keyword here (I31, Senior Manager, M, C3). Due to the short distances, a permanent exchange between all roles is possible (I31, Senior Manager, M, C3). According to the interviewees, this is very helpful and primarily ensures increased productivity due to the short distances.

Case organization 4:
In case organization 4, both direct and indirect communication takes place between EAs and ATs (see Figure 4.13). If the EA is part of the AT in certain phases, the former becomes relevant. A great deal of communication through personal meetings such as daily scrums, sprint plannings or sprint retrospectives takes place (I43, Head of Main Department, M, C4). If an EA is not part of the team, an indirect communication channel is primarily used. In this case, the developer communicates his request to the DA through the SM. The DA then communicates the request to the EA (I43, Head of Main Department, M, C4). According to the stakeholder group EA, the indirect communication path is the normal case. Apart from a CoPA every two weeks (I42, Enterprise Architect, EA, C4), no other regular meeting exists that would enable an exchange between EAs and ATs (I44, Developer, AT, C4). In addition, some of the interviewees named the tool ”Connect” as a frequently used communication medium (I42, Enterprise Architect, EA, C4; I44, Developer, AT, C4)

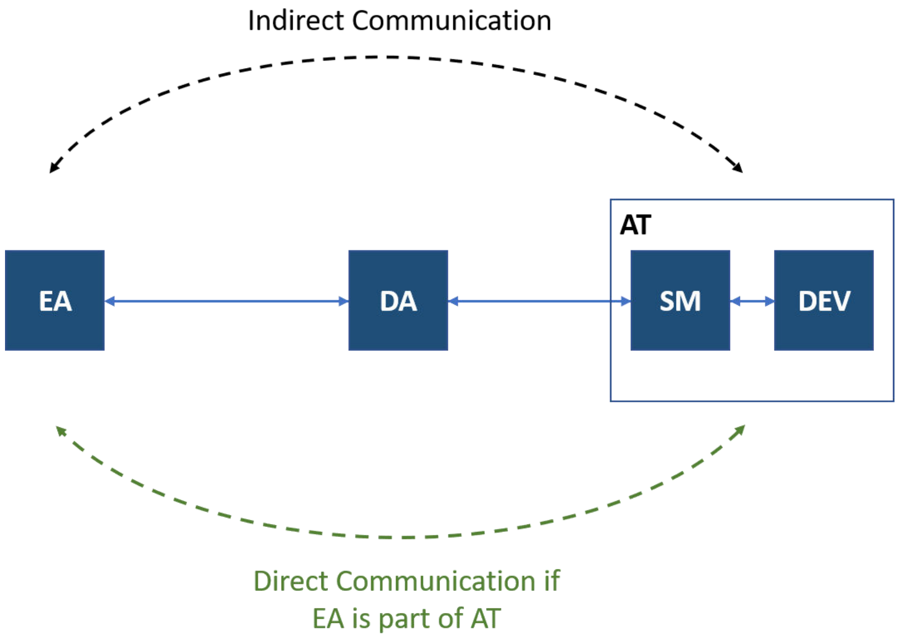


Figure 4.13.: Communication between EAs and ATs in case organization 4

What do ATs expect from the communication with Enterprise Architects?

Stakeholder group AT:
Members of ATs currently describe the communication between EAs and ATs as insufficient. They would prefer more direct communication. Communication through a go-between such as solution architects or domain architects is not considered ideal (I37, Chief Scrum Master, AT, C3). Additionally, they hope that the EAs understand this and propose concrete solutions (I17, Group Leader IT, AT, C1). The following assertion was often made:

> *"We don't need help from the ivory tower with PowerPoint slides but hands-on and concrete solution proposals, because pure theory is useless and only costs time"*
>
> *-I35, Scrum Master / Senior Agile Coach, AT, C3*

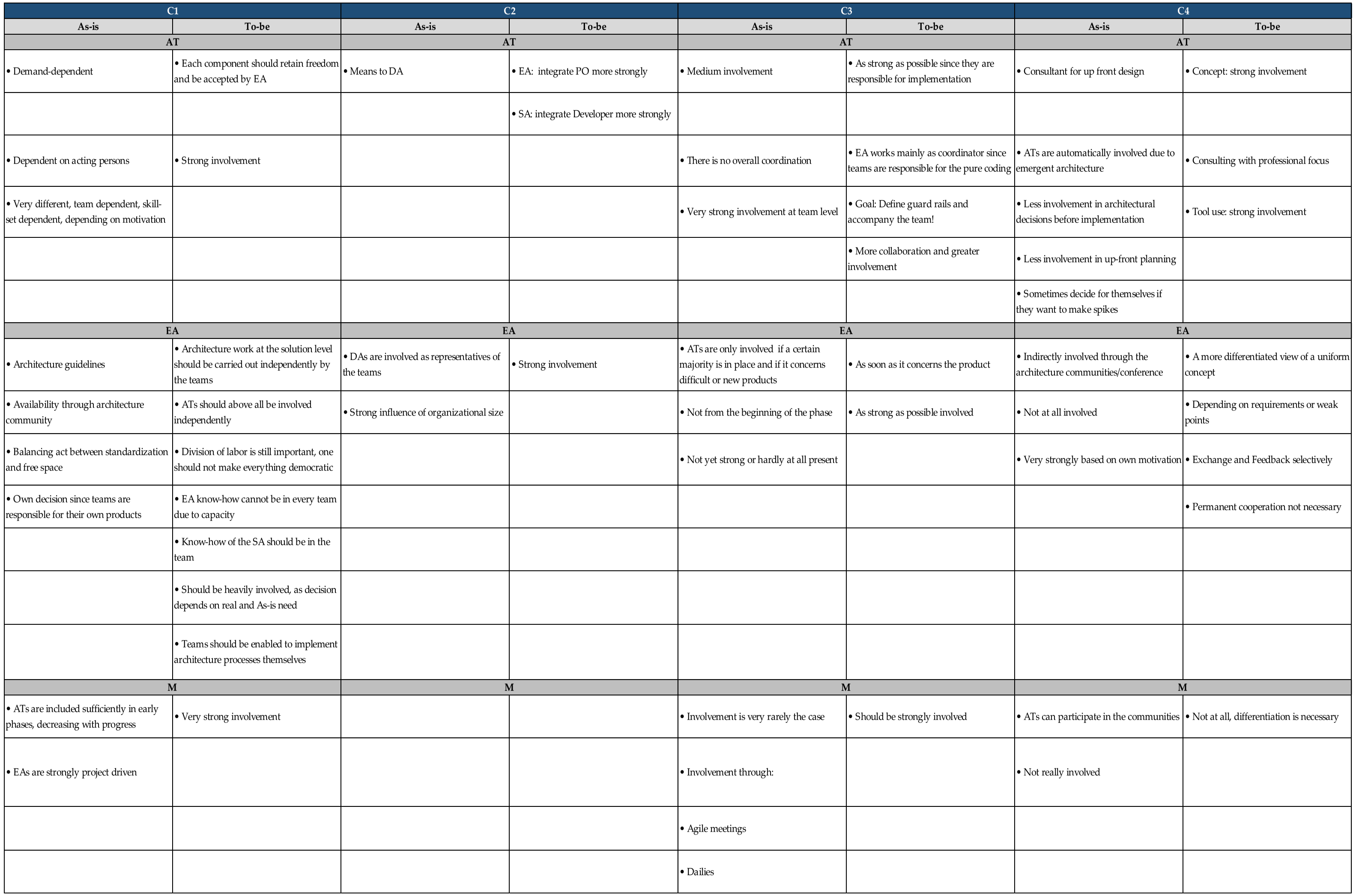
This clearly shows that more technical know-how and associated targeted solution proposals are expected. An EA should know and understand the real needs and requirements of the AT in detail (I17, Group Leader IT, AT, C1). In the best case, the EA first provides a solution space / guideline and, if necessary, also provides support through implementation and decision making (I44, Developer, AT, C4).

Stakeholder group EA:
According to the stakeholder group EA, an AT expects more personal contact and higher availability if needed (I16, Enterprise Architect, Enterprise Architect, EA, C1; I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2). Also, more concrete proposals for solutions should be provided and the presence at planning meetings should come to the foreground (I36, Enterprise Architect 1, EA, C3). The EA should be open to certain technologies and topics (I42, Enterprise Architect, EA, C4), support the team in an advisory capacity and inform about dependencies in the overall context (I41, Enterprise Architect / Head of Department, EA, C4). A target picture would also be helpful (I32, Enterprise Architect 2, EA, C3).

Stakeholder group M:
Stakeholder group M believes that an AT primarily wants fast feedback, relevance and feasibility (I15, Head of EAM, M, C1). Furthermore, ATs would like a concrete justification for a proposed technology or architecture (I31, Senior Manager, M, C3). Proposed solutions should be target group adequate (I15, Head of EAM, M, C1). This means that a developer requires concrete information about technical aspects, whereas a scrum master is satisfied with a higher level of abstraction. In addition, Ms emphasized that a team should be able to recognize when communication is actually relevant (I15, Head of EAM, M, C1). The role of the EA should not be used without it being necessary (I43, Head of Main Department, M, C4).

To what extent are ATs involved in the architecture processes that are relevant to them? To what extent should ATs be involved in the architecture processes that are relevant to them?

Table 4.42 compares the as-is state of the involvement of the ATs into the architecture processes relevant to them with the to-be state.



C1		C2		C3		C4	
As-is	To-be	As-is	To-be	As-is	To-be	As-is	To-be
AT		AT		AT		AT	
• Demand-dependent	• Each component should retain freedom and be accepted by EA	• Means to DA	• EA: integrate PO more strongly	• Medium involvement	• As strong as possible since they are responsible for implementation	• Consultant for up front design	• Concept: strong involvement
			• SA: integrate Developer more strongly				
• Dependent on acting persons	• Strong involvement			• There is no overall coordination	• EA works mainly as coordinator since teams are responsible for the pure coding	• ATs are automatically involved due to emergent architecture	• Consulting with professional focus
• Very different, team dependent, skill-set dependent, depending on motivation				• Very strong involvement at team level	• Goal: Define guard rails and accompany the team!	• Less involvement in architectural decisions before implementation	• Tool use: strong involvement
					• More collaboration and greater involvement	• Less involvement in up-front planning	
						• Sometimes decide for themselves if they want to make spikes	
EA		EA		EA		EA	
• Architecture guidelines	• Architecture work at the solution level should be carried out independently by the teams	• DAs are involved as representatives of the teams	• Strong involvement	• ATs are only involved if a certain majority is in place and if it concerns difficult or new products	• As soon as it concerns the product	• Indirectly involved through the architecture communities/conference	• A more differentiated view of a uniform concept
• Availability through architecture community	• ATs should above all be involved independently	• Strong influence of organizational size		• Not from the beginning of the phase	• As strong as possible involved	• Not at all involved	• Depending on requirements or weak points
• Balancing act between standardization and free space	• Division of labor is still important, one should not make everything democratic			• Not yet strong or hardly at all present		• Very strongly based on own motivation	• Exchange and Feedback selectively
• Own decision since teams are responsible for their own products	• EA know-how cannot be in every team due to capacity						• Permanent cooperation not necessary
	• Know-how of the SA should be in the team						
	• Should be heavily involved, as decision depends on real and As-is need						
	• Teams should be enabled to implement architecture processes themselves						
M		M		M		M	
• ATs are included sufficiently in early phases, decreasing with progress	• Very strong involvement			• Involvement is very rarely the case	• Should be strongly involved	• ATs can participate in the communities	• Not at all, differentiation is necessary
• EAs are strongly project driven				• Involvement through:		• Not really involved	
				• Agile meetings			
				• Dailies			

Table 4.42.: Involvement of ATs into architecture processes (as-is and to-be

The strength of involvement differs immensely. Currently (as-is), there is little or no involvement of ATs in architecture processes. Across all case study organizations a strong involvement was considered to be the desired status. The level of involvement also differs greatly among the interviewees. Some of the interviewees suggest a phased involvement, whereas others advocate a complete involvement.

How do Enterprise Architects support ATs? What kind of support do ATs expect from Enterprise Architects?

Table 4.43 compares the as-is state of the support of the EA for the ATs with the to-be state.

C1		C2		C3		C4	
As-is	To-be	As-is	To-be	As-is	To-be	As-is	To-be
AT		AT		AT		AT	
• Exchange of architecture information between the ATs, since there is hardly any communication between the teams so far	• Designing architectural themes together with the teams	• No direct communication	• Overall picture of the customer journeys	• A very weak consulting service currently exists	• Hands on (on a technical level)	• Consulting service	• Assistance with decisions
• More support	• Gap between ivory tower and real need	• No support	• To-be architecture	• Too high-level	• More details, less high-level	• Framework specifications	• Framework specifications
• Provide landscape (provide rough solution space) and tools	• New ideas and concrete solutions				• More direct support	• To-be architecture	• Providing solutions
	• SA is missing in the component				• Talk through problems and solutions; language of the developers must be clear	• Tools	
EA		EA		EA		EA	
• Ensuring that artefacts are also available as a basis for decision-making	• Concrete solutions	• Dependencies with expertise and technologies		• Architecture context	• Architecture context	• Consulting and support	• An understanding for architecture hints
• Preparatory activities for the teams	• Face-to-face communication			• Architecture principles	• Be able to react to changes quickly	• Independent action on architecture-specific topics	• IT target image
• Reference architecture	• Fast feedback			• Business context	• Business context	• Independent support for agile teams	• Recommendations regarding technology standards
• technology capability map (TCM) and product catalogue	• IF team is able to work quickly, then EA did a good job			• Currently very reactive and situational	• Concrete solutions	• Only what is needed	
	• Specification of a frame				• Reliability of models		
	• Working processes						
M		M		M		M	
• To-be architecture	• Carrying capacity			• Models are marked on whiteboards and explained, at the end there is a solution	• Common understanding	• Architecture principles	• An understanding for architecture hints
• Low enabling	• Practicability			• Permanent availability	• Culture	• Architecture spikes	
• Low support in implemenation	• Robustness and practical maturity				• Face-to-face communication	• Business capability map	
• Support for planning development						• Domain model	
						• Tools	

Table 4.43.: Support of EA for AT (as-is and to-be

As shown in Table 4.43, the way EAs support ATs differs significantly. Whereby especially the stakeholder groups AT and EA state that artifacts and models provided (as-is) are too high-level, the stakeholder group M does not seem to be aware of this. All case organizations state that a better overview, information about the goals, more communication and more detailed models or artifacts are necessary to better support.

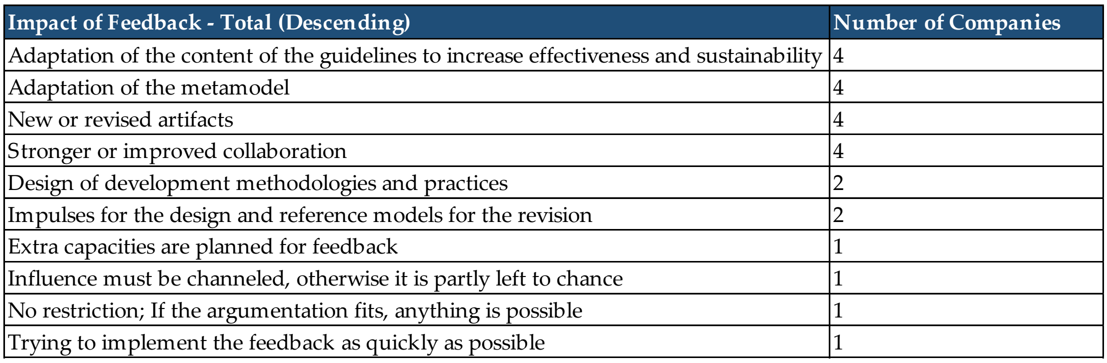
In what form and frequency can the ATs give feedback to Enterprise Architects? In what form should feedback from ATs be given to Enterprise Architects?

Table 4.44 compares the as-is state of the form and frequency of feedback from the EA to the ATs with the to-be state.

C1		C2		C3		C4	
As-is	To-be	As-is	To-be	As-is	To-be	As-is	To-be
AT		AT		AT		AT	
• All communication media such as email and phone	• After CoPA, internal employees meet in a second meeting to distribute tasks	• Not at all, no capacity	• Still in the planning stage, currently no information possible	• All communication media such as email and phone	• All communication media such as email and phone	• All communication media such as email and phone	• All communication media such as email and phone
• CoPA once a week	• Agile Cycle			• In a refinement	• As it is	• Frequency: Adhoc	• EA should be part of the sprint planning
	• All communication media such as email and phone			• Informally at any time	• CoPA	• No explicit appointment for ATs	• Frequency: Every 3 weeks
	• Feedback in phases in which architectural themes are relevant			• Open culture	• Developing common approaches to solutions	• No official procedure	• Frquency should depend on sprint and situation
	• If architecture topic is relevant, EA should be part of the sprint			• Optional architecture meeting every 4 weeks	• Direct dialogue face-to-face		• Satisfied with the informal way
	• Jira and Kanban Board				• Go through subject areas together		
	• LESS framework principle: "Walker"				• Recognize where current problems lie		
					• Suggestion: "One EA block every week to visit 15 teams"		
EA		EA		EA		EA	
• All communication media such as email and phone	• All communication media such as email and phone	• All communication media such as email and phone	• All communication media like such as email and phone	• All communication media such as email and phone	• All architects and people interested in architecture should participate	• All communication media such as email and phone	• All communication media such as email and phone
• Communication mainly ad hoc	• Objective	• Every four weeks meeting with DA	• Personal feedback through dialogue	• CoPA (every 2 months)	• All communication media such as email and phone	• Confluence	• Focus on the communities
• Feedback in review process	• Personal	• Status meetings		• One-to-one meetings	• CoPA	• Connect	
• Informally at any time	• Preferably via one-to-one meeting				• Panel discussions	• CoPA (every 2 weeks)	
• No structured form of feedback					• Strong communication and discussion	• Different from project to project	
• Through confluence mechanisms such as comments							
M		M		M		M	
• Informally at any time	• Added value, benefits and customer satisfaction should be focused			• Always, through all channels	• As it is now	• After the sprint or during the retro	• All communication media such as email and phone
• No structured form of feedback	• EA is also a product			• Most often, it happens face-to-face	• Everybody knows everybody	• All communication media such as email and phone	• As it is now
	• Find out if service is helpful				• Short distances	• CoPA	

Table 4.44.: Form and frequency of feedback from EAs for ATs (as-is and to-be)

Table 4.44 shows, that across all case organizations almost all communication media are used to provide feedback in an informal way. Some interviewees state that a frequency of every 3 to 4 weeks could be satisfying. CoPAs play a large role across all case organizations and are becoming more and more focused. In addition, optional one-on-one meetings should be established to create more direct communication to the architects and improve collaboration. Furthermore, the focus should also be on community work which could save time and capacity for architects and increase the value of feedback. Table 4.45 shows the impact of feedback from ATs on the EAM. Across all case organizations, feedback has a strong / high influence and is considered to be extremely important and highly desired (I13, Scrum Master, Requirement Engineer, AT, C1; I42, Enterprise Architect, Enterprise Architect, EA, C1; I15, Head of EAM, M, C1; I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2; I35, Scrum Master / Senior Agile Coach, AT, C3; I37, Chief Scrum Master, AT, C3; I42, Enterprise Architect, EA, C4; I41, Enterprise Architect / Head of Department, EA, C4).



Impact of Feedback - Total (Descending)	Number of Companies
Adaptation of the content of the guidelines to increase effectiveness and sustainability	4
Adaptation of the metamodel	4
New or revised artifacts	4
Stronger or improved collaboration	4
Design of development methodologies and practices	2
Impulses for the design and reference models for the revision	2
Extra capacities are planned for feedback	1
Influence must be channeled, otherwise it is partly left to chance	1
No restriction; If the argumentation fits, anything is possible	1
Trying to implement the feedback as quickly as possible	1

Table 4.45.: Impact of feedback from ATs on the EAM

As shown in Table 4.45, the results of the feedback from ATs about the EAM show that "Stronger or improved collaboration", "New of revised artifacts", "Adaptation of the metam odel" or "Adaptation of the content of the guidelines to increase effectiveness and sustainability" are necessary in 100% of the case organizations.

Which criteria are currently used to assess the value contribution of the EAM and which metrics are used to measure it? How could the value contribution be measured in the future?

In the following, approaches for measuring the value contribution of EAM for each case organization are listed. Thereby, as-is state is compared with the to-be state.

Case organization 1:
Table 4.46 compares the as-is state of the value contribution of the EAM with the to-be state.

Case organization 2:
Table 4.47 compares the as-is state of the value contribution of the EAM with the to-be state.

Case organization 3:
Table 4.48 compares the as-is state of the value contribution of the EAM with the to-be state.

Case organization 4:
Table 4.49 compares the as-is state of the value contribution of the EAM with the to-be state.

C1					
As-is	To-be	As-is	To-be	As-is	To-be
AT		EA		M	
• Acceptance /Feedback of book of standards which will be released about twice every year	Combination of several factors 1) Customer feedback 2) Feedback from ATs and products 3) Basis of book of standards review	• Count how often an architecture is discarded	• Check if product cut is correct	• Not at all, there are no KPIs at the mom	Combination of factors • Customer satisfaction • Speed of decision making • Optimal vs. global solution
		• Do not build similar functions differently	• Find out how to work and release more quickly		• Use of guidelines / architectural principles
		• Goal: Speed by increasing the number of releases	• Find out if there is a need for action		
		• Whether one recognizes and lives synergies? (implement the same functionality immediately, avoid redundancy)	• Find out what exactly the value contribution is		
			• Metrics for speed and flexibility		
			Combination of factors • Customer satisfaction etc. • Urgency • Business value • Product utility mode		

Table 4.46.: Value contribution (as-is and to-be) - case organization 1

C2	
As-is	To-be
EA	
	• Check feedback for KPIs
	• Net promoter score (NPS) as an assessment
	• Recommendation rate as highest criterion (subjective)
	• With the help of key figures for the portfolio which are still in development

Table 4.47.: Value contribution (as-is and to-be) - case organization 2

C3					
As-is	To-be	As-is	To-be	As-is	To-be
AT		EA		M	
	• Check for errors and search for reasons for errrors	• Conversion to API Design (RESTful API)	• Adherence to the architecture		• Ascertain the differences (before and after)
	• Check for version skipping (e.g. Java 9)	• How many use the old patterns vs. how many use the new patterns?	• Check the API implementation level		• Split project into 5 business releases, if we define MVP it will be decomposed into use cases
	• Control performance and find causes for low performance	• Positive feedback from the teams and the C-Level	• Count how often an architecture is discarded		Combination of factors • Adjustment • Budget • Correctness • Time
	• Count how often an architecture is discarded	• Speed of problem solving	• Number of dependencies		
	• Existence of face-in-face-out cycles		• Reliability of the architecture		
	• Process 1) Metrics for code e.g. how well is the code encapsulated? 2) Define measures 3) See what happens 4) Check whether the counselling service is good or bad		• Time until features are delivered		
			• Total cost of ownership perspective		
			• Track architecture KPIs		

Table 4.48.: Value contribution (as-is and to-be) - case organization 3

C4					
As-is	To-be	As-is	To-be	As-is	To-be
AT		EA		M	
	• Business value of an architectural story	• Effectivness of community work	KPIs: • Availability of information (KPI approach) • Monitor use of consulting services • Feedback from the architecture community • Find out how good the consulting was	• Customer satisfaction	• Find out if size of monoliths can be reduced
		• KPIs		• Employee satisfaction	• Find out if team is getting faster
		• Number of results produced by community		• Feedback	• Measuring compliance with architecture principles
				• KPIs, to-be KPIs	• No more classic assessment of agile teams

Table 4.49.: Value contribution (as-is and to-be) - case organization 4

As shown in Table 4.46,4.47, 4.48 and 4.49 the as-is state differs significantly from the to-be. In almost every case organization there are only first attempts to measure the value contribution. Currently, 50% of the organizations do not measure the value contribution of the EAM (as-is). In addition, 100% of the case study organizations provide first ideas for measuring. 50% argues that measuring the value contribution of the EAM can only be achieved by measuring a combination of factors, such as budget, correctness, customer satisfaction, feedback from ATs and products, number of adjustments, speed of decision making, and time.

To what extent do ATs want to be controlled by the Enterprise Architects?

100% of the interviewees state that the AT does not wish to be controlled. Alternatively, they would prefer to be supported and guided through the processes and decisions where necessary. They see collaboration as being important. A guideline of architecture principles, for example, is required (I15, Head of EAM, M, C1; I35, Scrum Master / Senior Agile Coach, AT, C3). The latter gives enough freedom and supports the decision-making process (I43, Head of Main Department, M, C4).

Which requirements / specifications can help?

Stakeholder group AT:

- **Architecture principles** (I35, Scrum Master / Senior Agile Coach, AT, C3)
- **Collaboration with the mass instead of top-down** (I35, Scrum Master / Senior Agile Coach, AT, C3)
- **Communication diagram** (I44, Developer, AT, C4)
- **Hands-on technical support** (I37, Chief Scrum Master, AT, C3)
- **"Helping leader" as a leadership model** (I35, Scrum Master / Senior Agile Coach, AT, C3)
- **Interfaces** (I44, Developer, AT, C4)
- **Less abstraction** (I37, Chief Scrum Master, AT, C3)
- **Model specifications** (I44, Developer, AT, C4)
- **Professionalism through adhering to guidelines must be guaranteed** (I35, Scrum Master / Senior Agile Coach, AT, C3)
- **Self-conception of the EA, who is responsible for the implementation of the results** (I17, Group Leader IT, AT, C1)
- **Transparency** (I13, Scrum Master, Requirement Engineer, AT, C1; I37, Chief Scrum Master, AT, C3)

Stakeholder group EA:

- **Agile frameworks, however only conditionally** (I42, Enterprise Architect, EA, C4)
- **Architecture boards** (I36, Enterprise Architect 1, EA, C3)
- **Architecture models in detailed form** (I41, Enterprise Architect / Head of Department, EA, C4)
- **Architecture principles** (I12, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1; I36, Enterprise Architect 1, EA, C3; I32, Enterprise Architect 2, EA, C3)
- **Architecture spikes (interaction of components)** (I12, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1)
- **Blueprints (Released software products)** (I12, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1)
- **Definition of roles and responsibilities** (I36, Enterprise Architect 1, EA, C3)
- **Details on why to use a certain technology** (I42, Enterprise Architect, Enterprise Architect, EA, C1)
- **Higher degree of maturity** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)
- **Processes** (I36, Enterprise Architect 1, EA, C3)
- **Requirements regarding the specialism would help with the leadership** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)
- **Selection of the solution space** (I42, Enterprise Architect, Enterprise Architect, EA, C1)
- **Technical specifications regarding interfaces, infrastructure and application** (I41, Enterprise Architect / Head of Department, EA, C4)
- **Transparency** (I12, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1)

Stakeholder group M:

- **Architecture principles** (I43, Head of Main Department, M, C4)
- **Extremely individual and therefore difficult to answer** (I31, Senior Manager, M, C3)
- **Include architectural themes directly in Definition of Done (DoD)** (I43, Head of Main Department, M, C4)
- **Only a few clear and comprehensible framework conditions** (I15, Head of EAM, M, C1)
- **Transparency (I15, Head of EAM, M, C1)**

Which requirements / specifications would not be helpful or restrictive?

Stakeholder group AT:

- **Artifacts and specifications not arising from joint discussions with the teams** (I17, Group Leader IT, AT, C1)
- **Commanding** (I13, Scrum Master, Requirement Engineer, AT, C1)
- **If the EA intervenes too strongly in the activities of the component teams** (I13, Scrum Master, Requirement Engineer, AT, C1)
- **Mandatory requirements** (I13, Scrum Master, Requirement Engineer, AT, C1)
- **Must be softened by the EA** (I35, Scrum Master / Senior Agile Coach, AT, C3)
- **New technologies block, no room for further development given** (I44, Developer, AT, C4)
- **No possibilities for spiking** (I44, Developer, AT, C4)
- **One technology stack** (I44, Developer, AT, C4)
- **Restrict self-organization** (I13, Scrum Master, Requirement Engineer, AT, C1)
- **Rigid big design up front architecture is "deadly"** (I35, Scrum Master / Senior Agile Coach, AT, C3)
- **This is what the EA must achieve and communicate: Joint decision is essential** (I35, Scrum Master / Senior Agile Coach, AT, C3)
- **Too high a level of abstraction** (I37, Chief Scrum Master, AT, C3)
- **Very detailed tool specifications leading to strong restriction** (I17, Group Leader IT, AT, C1; I44, Developer, AT, C4)

Stakeholder group EA:

- **Frame too detailed** (I32, Enterprise Architect 2, EA, C3)
- **Freedom with regard to product portfolio** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)
- **Guide element no longer state-of-the-art because too slow** (I12, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1)
- **If the EA intervenes too strongly in the activities of the component teams** (I32, Enterprise Architect 2, EA, C3)
- **Lack of transparency** (I36, Enterprise Architect 1, EA, C3)
- **One size fits all** (I12, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1)
- **One technology stack** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)
- **Provide tips and recommendations** (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2)

- **Strong restrictions make control more difficult** (I42, Enterprise Architect, EA, C4)
- **Too high a level of abstraction** (I32, Enterprise Architect 2, EA, C3)
- **Too many rules** (I36, Enterprise Architect 1, EA, C3)
- **When benefits for the project cannot be presented and justified** (I12, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1)

Stakeholder group M:

- **Blunt and opaque regulations** (I15, Head of EAM, M, C1)
- **Extremely individual results in difficult to answer** (I31, Senior Manager, M, C3)
- **Pure theory without practical relevance and technical expertise** (I43, Head of Main Department, M, C4)

4.5.1. Expectations of Agile Teams

In the following section, the expectations of agile teams (AT) regarding architectural models, availability of EAs, communication with EAs, involvement in architecture processes, support and feedback of EAs are presented per stakeholder group. Afterwards, the answer to the question, whether or not an AT recommends the role of an EA is is given. Last but not least, the main findings of this section will be shown.

Figure 4.14 shows a spider chart based on the ratings given from the interviewees. The ratings are summarized from all case organizations. The numbers 1-7 are given for a better overview.

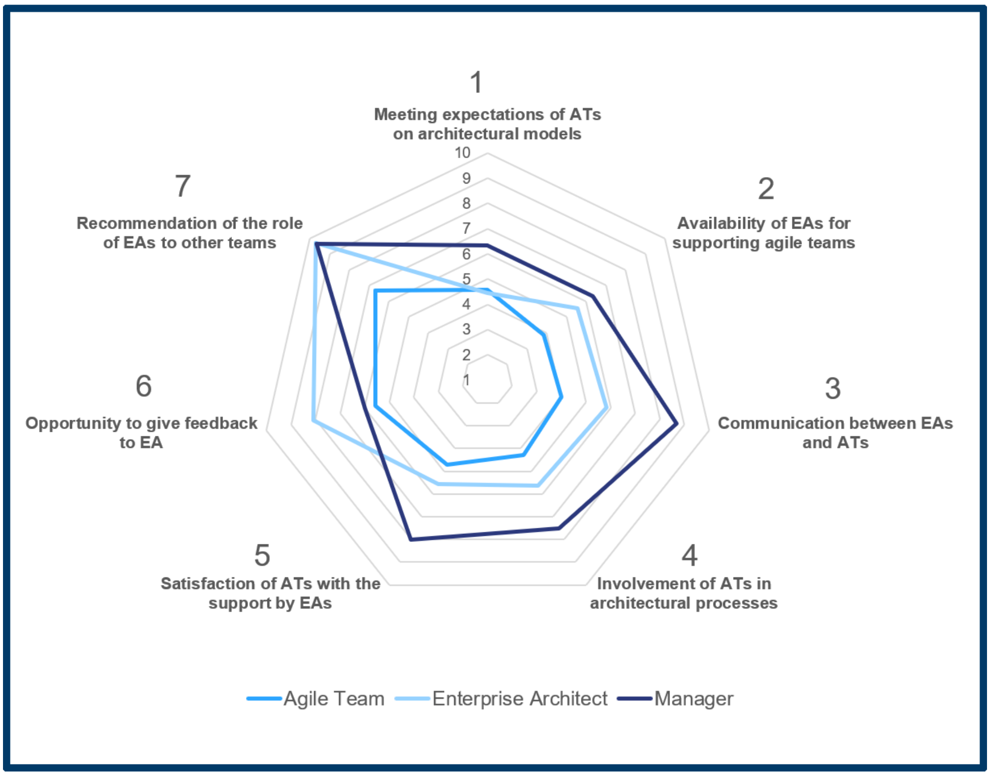


Figure 4.14.: Spider chart across all case organizations

Figure 4.14 shows the difference of expectations among the three stakeholder groups AT, EA and M on a scale from 1 (not fulfilled) to 10 (fully fulfilled).

(1) Reasons: Meeting expectations of ATs on architectural models

Stakeholder group AT:
85.71% of the interviewees from the stakeholder group AT state that the models provided by the EA are too unspecific and therefore hardly provide any assistance. According to one interviewee, the models are ”highly extensible” (I35, Scrum Master / Senior Agile Coach, AT, C3). Furthermore, there is a lack of both punctuality and speed of delivery of the models and support (I17, Group Leader IT, AT, C1; (I37, Chief Scrum Master, AT, C3). Among other things, poor communication (I35, Scrum Master / Senior Agile Coach, AT, C3), a low degree of maturity (I23, Product Owner, AT, C2) and the duration of decision-making are listed as causes (I37, Chief Scrum Master, AT, C3). In addition, there are no specific architecture guidelines (I35, Scrum Master / Senior Agile Coach, AT, C3). Accordingly, expectations are not met (I13, Scrum Master, Requirement

Engineer, AT, C1).

14.29% of the interviewees of the stakeholder group AT are of the opinion that the models do not lack anything.

> *"I enjoy the freedom"*
>
> *-I44, Developer, AT, C4*

Thus, the expectations are fulfilled (I44, Developer, AT, C4).

Stakeholder group EA:
The majority of the stakeholder group EA states that the models provided by the EA do not meet the AT's expectations. The current changes were the primary cause mentioned (I12, Enterprise Architect, Enterprise Architect, EA, C1; I32, Enterprise Architect 2, EA, C3). On the one hand, there is too little time to provide more detailed models, and on the other hand, the target architecture is too oriented towards the up-front design (I12, Enterprise Architect, Enterprise Architect, EA, C1). The target architecture should evolve parallel with the project and be adapted iteratively, if necessary (I12, Enterprise Architect, Enterprise Architect, EA, C1). In addition, the development and maintenance of the knowledge base is not yet complete (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2). Some of the EA interviewees argued that a clear understanding of ATs is missing (I36, Enterprise Architect 1, EA, C3) and that expectations vary widely between POs and developer (I42, Enterprise Architect, EA, C4). The communication of the models to the ATs is poor (I32, Enterprise Architect 2, EA, C3). Here the following statement was made:

> *"Not all product teams know the reference model and also do not know where to find it!"*
>
> *-I32, Enterprise Architect 2, EA, C3*

In addition, there is a lack of guidance on how to use a model (I32, Enterprise Architect 2, EA, C3).

Stakeholder group M:
Stakeholder group M shares the overall opinion that the models provided by the EAs do not yet meet the expectations of the ATs. Many standard solutions are used to reduce complexity (I15, Head of EAM, M, C1). However, this does not only have advantages, but also leads to additional effort (I15, Head of EAM, M, C1) due to non-optimal solutions and unnecessary functionality. The primary reason for this were the current changes (I31, Senior Manager, M, C3; I43, Head of Main Department, M, C4). Time, cost and technical support also play an important role in providing detailed architectural models (I15, Head of EAM, M, C1). According to the interviewees, quality differs depending on the age of the model. This does not automatically imply that old proven models still meet expectations today.

> *"Better models for new themes, worse models for old themes"*
>
> *-I43, Head of Main Department, M, C4*

(3) Reasons: Communication between EAs and ATs

Stakeholder group AT:
85.71% of the interviewees from the stakeholder group AT state that the expectations regarding communication have not yet been fulfilled. This is primarily due to the large workload of the EAs (I13, Scrum Master, Requirement Engineer, AT, C1), which results in a lack of capacity (I23, Product Owner, AT, C2). This affects punctuality and speed of communication (I17, Group Leader IT, AT, C1) and leads to disappointment in the team.

> *"The teams are disappointed, they don't feel properly cared for"*
>
> *-I35, Scrum Master / Senior Agile Coach, AT, C3*

The main reason cited here, in addition to the current changes, is the way in which communication is conducted. For a developer there is hardly any possibility to contact an EA directly. (I37, Chief Scrum Master, AT, C3). Communication always takes place indirectly through an intermediate role (I37, Chief Scrum Master, AT, C3).

14.29% of interviewees from the stakeholder group AT believe that the expectations regarding communication are fulfilled.

> *"I receive the information I need"*
>
> *-I44, Developer, AT, C4*

Required information is provided promptly. The communication channel works and expectations are met (I44, Developer, AT, C4).

Stakeholder group EA:
81.82% of the interviewees from the stakeholder group EA believe that the expectations regarding communication are not fulfilled. The general argument is that more communication is necessary and that there is a great need for optimization (I12, Enterprise Architect, Enterprise Architect, EA, C1). The added value of the work is recognized but could be more appreciated by the ATs (I12, Enterprise Architect, Enterprise Architect, EA, C1). Larger topics can be better planned than smaller topics (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2). This is due to the lack of capacity (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2; I36, Enterprise Architect 1, EA, C3). Additionally, there are various expectations on the part of the ATs (I42, Enterprise Architect, EA, C4). For example, ATs often do not know which activity an EA performs (I36, Enterprise Architect 1, EA, C3). There is a

lack of detailed guidance and adequate assistance (I32, Enterprise Architect 2, EA, C3), because a high-level architectural vision is not helpful (I32, Enterprise Architect 2, EA, C3). In addition, communicating through the go-between does not always work (I42, Enterprise Architect, EA, C4). This leads to dissatisfaction and unwillingness on the part of the EA (I41, Enterprise Architect / Head of Department, EA, C4). A proposed solution here could be controlling stakeholder expectations (I42, Enterprise Architect, EA, C4).

18.18% of interviewees from the stakeholder group EA believe that expectations regarding communication are met. The reasoning for this is that there are no complaints (I16, Enterprise Architect, Enterprise Architect, EA, C1).

Stakeholder group M:
Stakeholder group M is of the opinion that the expectations regarding communication are fulfilled. According to the interviewees, there is enough time for a close exchange between the EA and the AT (I15, Head of EAM, M, C1). The availability of contact persons is always guaranteed (I31, Senior Manager, M, C3). One of the interviewees stated that, if necessary, the EA can also be part of the team and thus no communication problems can arise (I43, Head of Main Department, M, C4). A need for optimization is only seen in cross-project communication and the specification of topics (I15, Head of EAM, M, C1).

(4) Reasons: Involvement of ATs in architectural processes

Stakeholder group AT:
85.71% of the interviewees from the stakeholder group AT state that expectations regarding the integration of ATs into architectural processes relevant for them are not fulfilled. Involvement is primarily insufficient (I23, Product Owner, AT, C2; I37, Chief Scrum Master, AT, C3). Although there are initial approaches to counteract this problem through increased communication, this is not enough.

> *"Much more is needed"*
>
> *-I35, Scrum Master / Senior Agile Coach, AT, C3*

The demand is significantly higher than the available capacity (I13, Scrum Master, Requirement Engineer, AT, C1).

> *"Solution Architects in the ATs are still too busy with troubleshooting and debugging instead of looking for a clean architecture"*
>
> *-I35, Scrum Master / Senior Agile Coach, AT, C3*

In addition, communication is mainly based on the push principle (bottom-up) (I17, Group Leader IT, AT, C1). This must change, as the EA also has a responsibility to deliver artifacts, models and support (I17, Group Leader IT, AT, C1).

14.29% of the interviewees from the stakeholder group AT state that their expectations with regard to involvement in architectural processes are fulfilled.

> *"Nothing is missing"*
>
> *-I44, Developer, AT, C4*

Stakeholder group EA:
72.73% of the stakeholder group EA agrees that expectations regarding integration into architectural processes are not met. This is primarily due to the lack of integration and capacity (I12, Enterprise Architect, Enterprise Architect, EA, C1; I16, Enterprise Architect, Enterprise Architect, EA, C1; I36, Enterprise Architect 1, EA, C3; I41, Enterprise Architect / Head of Department, EA, C4). Also, a lack of both the ability of EAs to be integrated and accepted by ATs as well as the clarity of the EAs responsibility exists (I36, Enterprise Architect 1, EA, C3). According to some interviewees, the ATs do not find it necessary to contact an EA (I42, Enterprise Architect, EA, C4). This is due to the lack of support. Some EAs see a need to optimize the involvement of ATs (I42, Enterprise Architect, EA, C4).

> *"Where necessary, the manner of involvement must be clarified"*
>
> *-I42, Enterprise Architect, EA, C4*

An EA also states that feedback plays an important role, as it allows them to optimize their capacities (I36, Enterprise Architect 1, EA, C3).

27.27% of interviewees from the stakeholder group EA state that expectations regarding involvement into architectural processes are fulfilled by the role of the DA (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2).

Stakeholder group M:
66.67% of interviewees from stakeholder group M state that expectations with regard to involvement in architectural processes are generally met. There are overlapping issues, such as technology decisions, where teams cannot be involved due to time constraints. This is particularly the case when this is not a question of a single scope, but of an overlapping technology decision such as switching to a new operating system or skipping a Java version (I15, Head of EAM, M, C1). One interviewee also stated that the target image is more accurate due to the existing involvement in architecture processes (I31, Senior Manager, M, C3). 23.33% of the stakeholder group M interviewees did not give a reason.

(5) Reasons: Satisfaction of ATs with the support by EAs

Stakeholder group AT:
85.71% of the interviewees from the stakeholder group AT state that their expectations regarding support are not fulfilled. This is primarily explained by the lack of availability, lack of capacity and therefore the large workload of the EA (I13, Scrum Master, Requirement Engineer, AT, C1). The reason for this are the current changes in the case organizations (I23, Product Owner, AT, C2). Some of the interviewees stated that the EAs see no overall responsibility and do not have sufficient technical know-how to offer support successfully (I17, Group Leader IT, AT, C1).

> *"For high-level requirements, support is good, as soon as it's really about technical content, it's over"*
>
> *-I17, Group Leader IT, AT, C1*

Another reason mentioned was indirect communication (I35, Scrum Master / Senior Agile Coach, AT, C3). Furthermore, sometimes no honest feedback is provided by the AT which does not contribute to improving the situation (I37, Chief Scrum Master, AT, C3).

14.29% of the interviewees from the stakeholder group AT stated that their expectations regarding involvement in architectural processes were met. Here the same answer as in 3.6.2 was given.

> *"Nothing is missing"*
>
> *-I44, Developer, AT, C4*

Stakeholder group EA:
54.55% of the interviewees from the stakeholder group EA state that, on the whole, they meet the expectations with regard to involvement into architectural processes. From their point of view, no stronger support in terms of time is possible due to the lack of capacity (I16, Enterprise Architect, Enterprise Architect, EA, C1; I32, Enterprise Architect 2, EA, C3). The need for optimization is also seen in the level of detail of the artifacts (I12, Enterprise Architect, Enterprise Architect, EA, C1).

45.45% of the interviewees from the stakeholder group EA state that they do not meet their expectations with regard to involvement into architectural processes. Lack of skills, lack of acceptance on the part of the ATs and ambiguities in responsibility and accountability are the main reasons for this assessment (I36, Enterprise Architect 1, EA, C3). Furthermore, the lack of capacity is a major problem (I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2). Some interviewees argue that some ATs do not even see the need to be truly involved (I42, Enterprise Architect, EA, C4).

> *"Where necessary, the manner of involvement must be clarified"*
>
> *-I42, Enterprise Architect, EA, C4*

One EA points out that architects are located further up in the hierarchy than the ATs which results in gaps in the implementation (I41, Enterprise Architect / Head of Department, EA, C4).

Stakeholder group M:
66.67% of the interviewees from stakeholder group M state that they almost completely fulfil expectations with regard to involvement into architectural processes. This is primarily based on a high degree of clarification, explanation and positive feedback (I15, Head of EAM, M, C1; I31, Senior Manager, M, C3). Also, the past success speaks for itself.

> *"There is no major project that has gone totally wrong"*
>
> *-I15, Head of EAM, M, C1*

The need for optimization is seen particularly in capacity. The focus is on the most important and most relevant projects with great leverage (I15, Head of EAM, M, C1). As a result, smaller projects are hardly integrated into architectural processes (I15, Head of EAM, M, C1). Also, the speed of the results is not yet 100% satisfactory.

> *"Too slow: Result is good, but could go faster"*
>
> *-I15, Head of EAM, M, C1*

33.33% of the interviewees from the stakeholder group M did not give reasons.

(6) Reasons: Opportunity to give feedback to EA

Stakeholder group AT:
57.14% of the interviewees from the stakeholder group AT state that their expectations with regard to feedback are almost fulfilled.

> *"Everything fits"*
>
> *-I37, Chief Scrum Master, AT, C3*

This is explained by the very positive, well-founded and helpful feedback (I13, Scrum Master, Requirement Engineer, AT, C1). According to the interviewees, there could be even more feedback in the future (I17, Group Leader IT, AT, C1).

42.86% of the interviewees from the stakeholder group AT state that their expectations regarding the feedback are not fulfilled. Due to maturity and capacity, there is a primary lack of support (I23, Product Owner, AT, C2). There is not enough feedback (I35, Scrum Master / Senior Agile Coach, AT, C3).

> *"The feedback is not pursued"*
>
> *-I35, Scrum Master / Senior Agile Coach, AT, C3*

Stakeholder group EA:
81.82% of interviewees from the stakeholder group EA state that they almost completely fulfil their expectations with regard to feedback. Statements such as:

> *"It works, there are no complaint! I don't know how I could improve it"*
>
> *-I16, Enterprise Architect, Enterprise Architect, EA, C1*

or

> *"Fine, no issues"*
>
> *-I22, Enterprise Architect, Enterprise Architect, Enterprise Architect, EA, C2*

confirm this. Sufficient possibilities (I32, Enterprise Architect 2, EA, C3), quick solutions (I32, Enterprise Architect 2, EA, C3; I41, Enterprise Architect / Head of Department, EA, C4), workshops (I32, Enterprise Architect 2, EA, C3; I36, Enterprise Architect 1, EA, C3) and many feedback mechanisms (I42, Enterprise Architect, EA, C4) are evidence that expectations have been met. Some interviewees also argue that the lack of maturity due to the current changes leads to less need for feedback and different expectations (I42, Enterprise Architect, EA, C4). The expectations of the ATs tend towards solutions (I42, Enterprise Architect, EA, C4). In addition, it was added that some teams do not use the role of the EAs, and therefore there is no dissatisfaction with the feedback (I36, Enterprise Architect 1, EA, C3).

18.18% from the stakeholder group EA describe the fulfillment of expectations as negative. This is due to the low number of feedback mechanisms and their structure. Also, the added value is not appreciated by ATs. Some of the interviewees attributed this to the low degree of maturity (I12, Enterprise Architect, Enterprise Architect, EA, C1).

Stakeholder group M:
33.33% of interviewees from stakeholder group M state that their expectations with regard to feedback are met. This was explained by a high level of availability and willingness (I31, Senior Manager, M, C3). 33.33% of interviewees from stakeholder group M state that their expectations regarding feedback are not met.

> *"Many teams see the EAM as a classic ivory tower and therefore give hardly any feedback"*
>
> *-I15, Head of EAM, M, C1*

33.33% of interviewees from stakeholder group M did not give an exact assessment, but gave an explanation. According to the interviewees, the participation in the feedback culture is too low.

> *"Too few people are involved - more active participation instead of just consuming is desired"*
>
> *-I43, Head of Main Department, M, C4*

In addition, it was stated that the feedback process is currently still strongly top-down driven (I43, Head of Main Department, M, C4).

Findings Across all case Organizations

(1) Meeting expectations of ATs on architectural models

Stakeholder group AT:

- Currently, architecture models are too unspecific and therefore not helpful for addressing issues of the ATs
- Quality of architectural models could alternative: should? be improved
- The duration of the delivery of architectural models is tedious
- The minority (14,29%) of ATs enjoys the freedom due to the missing level of detail of the architecture models

Stakeholder group EA::

- EAs already know, that, due to the changes, the architecture models must be adapted
- The fear of "big design up front" (BDUF) comes true
- EAs want to apply agile architecture principles, i.e. iterative development of architecture is necessary
- Communication and support for the architecture models must increase

Stakeholder group M:

- Using a "best breed" concept instead of a "one size fits all" would be highly beneficial for ATs
- Standard solutions lead to a reduction of complexity but also to additional effort and sub-optimal solutions
- Old proven models are not necessarily state of the art

(2) Availability of EAs for supporting ATs

Stakeholder group AT:

- The rating of the time availability of an EA was rated with 3.9 by the stakeholder group AT on a scale from 1 ″not available at all″ to 10 ″always available″.

Stakeholder group EA:

- The rating of the time availability of an EA was rated with 5.5 by the stakeholder group EA on a scale from 1 ″not available at all″ to 10 ″always available″.

Stakeholder group M:

- The rating of the time availability of an EA was rated with 6.3 by the stakeholder group M on a scale from 1 ″not available at all″ to 10 ″always available″.

All stakeholder groups state that a lack of capacity restricts the EA's time availability.

(3) Communication between EAs and ATs

Stakeholder group AT:

- Lack of capacity restricts the EAs time availability
- Lack of capacity leads to indirect and slow communication
- Indirect communication leads to the dissatisfaction of ATs
- Minority (14,29%) of ATs is satisfied with the communication and the information provided

Stakeholder group EA:

- EAs feel that the value of their contribution is recognized by the ATs but not appreciated
- EAs believe that indirect communication is not the best form of communication
- Minority (18,18%) of EAs is satisfied with the communication as there are no complaints
- EAs are partly unaware of their role in the agile environment, as existing assumptions no longer apply
- Lack of capacity makes it difficult to dive deeply into topics

Stakeholder group M:

- Ms believe that there is always enough time for communication between EAs und ATs
- They perceive the communication between ATs and EAs differently than the communication between ATs and EAs

(4) Involvement of ATs in architectural processes

Stakeholder group AT:

- Solution Architects are too busy with troubleshooting
- ATs do not feel sufficiently involved in relevant architectural processes
- Communication is based on the push principle (bottom-up)
- Team architects act reactively and not proactively
- Minority (14,29%) of ATs is satisfied with the involvement

Stakeholder group EA:

- Lack of involvement and capacity leads to low involvement of ATs in architecture processes
- Lack of support leads to the fact that the ATs do not want to communicate with EAs at all
- The way in which ATs are integrated must be defined by clear processes

Stakeholder group M:

- Involvement in overall architectural processes (e.g. technology decisions) is not possible due to time constraints
- Perception of the involvement of ATs in the architecture processes differ significantly to EAs

(5) Satisfaction of ATs with the support by EAs

Stakeholder group AT:

- Lack of capacity in combination with being overworked leads to poor support of AT
- High-level support is good, but technical support is not satisfactory
- Not satisfied with indirect communication
- Minority (14,29%) of ATs is satisfied with the support
- Due to the new environment, EAs do not see how they can make a good contribution to the ATs
- ATs expect technical support, but EAs are not capable of providing it

Stakeholder group EA:

- Due to missing capacity, EAs are not able to support ATs more significantly
- Level of detail of artifacts could be greater
- Lack of architectural skills, lack of acceptance of the EA by the ATs and ambiguities pertaining to competence and responsibility leads to bad support
- Some of the ATs do not see the need to talk to EAs

- Only half of the EAs think that ATs are involved in respective processes
- Hierarchical gaps between EAs and ATs prevent the stronger involvement of ATs in architectural processes

Stakeholder group M:

- Involvement into architectural processes is almost completely fulfilled due to positive feedback
- The need for optimization is seen in the lack of capacity
- Only most important and relevant projects with big leverage are in focus, and thus only the involved ATs are integrated in architectural processes
- ATs are only integrated slowly
- Majority (66,67%) of Ms think that ATs are involved in architectural processes

(6) Opportunity to give feedback to EA

Stakeholder group AT:

- Majority (57,14%) of ATs state: very positive, well-founded and helpful feedback leads to fulfilled expectations on the part of ATs
- Feedback culture is not being pursued

Stakeholder group EA:

- Plenty possibilities, quick proposals for solutions, workshops and many feedback mechanisms are evidence for fulfilling the expectations
- The expectations of the ATs tend to be directed towards a solution
- Majority (81.82%) of EAs think that the expectations regarding feedbacks are fulfilled
- Transformation effects lead to different expectations among stakeholders
- Since several ATs are not supported by EAs, they do not have any expectations on feedback
- Minority (18,18%) of EAs think that appropriate feedback mechanisms are missing
- Minority (18,18%) of EAs also think that the added value of feedback is not well-understood

Stakeholder group M:

- High availability and willingness to support ATs leads to positive feedback
- Many ATs see EAM as an ivory tower and therefore give little feedback
- Participation in the feedback culture is too low
- Feedback is welcome but at the same time only provided reluctantly
- The feedback process is currently still strongly top-down driven
- Feedback cycles are missing

(7) Recommendation rate of the role of EA

Stakeholder group AT:

- The recommendation rate of the role of the EA was rated 6.7 on a scale from 1 ″no recommendation″ to 10 ″strong recommendation″ by the stakeholder group AT

Stakeholder group EA:

- The recommendation rate of the role of the EA was rated 9.5 on a scale from 1 ″no recommendation″ to 10 ″strong recommendation″ by the stakeholder group EA

Stakeholder group M:

- The recommendation rate of the role of the EA was rated 9.7 on a scale from 1 ″no recommendation″ to 10 ″strong recommendation″ by the stakeholder group M

Main Findings across all case organizations

The interesting point here is that the rating of the stakeholder group AT differs significantly from the other two stakeholder groups. The reason for this is that the ATs are dissatisfied with the current situation. The added value of the EAM has not yet reached the ATs, and the gap between the ATs and the EA is still too large.

Across all case organizations:

- Lack of capacity restricts EAs time availability and limits support
- Low degree of maturity due to changes
- Lack of technical know-how of EAs which leads to superficial artifacts
- More detail for architectural artifacts required by ATs
- Expectations differ between stakeholder groups
- Gap between self-perception and external perception differs significantly and leads to different expectations

4.5.2. Recommendation Rate of Enterprise Architects

The net promoter score (NPS) was used to ascertain how likely the role of the EA will be recommended to a team that had previously worked without an EA by each stakeholder group. This decision was made primarily because of the simplicity and practicality of the approach. The NPS is calculated by the difference between promoters and detractors. The answers are measured on a scale from 0 (not likely at all) to 10 (very likely). Promoters are people who gave a score of either 9 or 10. Detractors are those who gave a score of 1 to 6. Passives who selected a 7 or 8 are considered indifferent and are not included in the evaluation. To calculate the NPS, the detractors (in percent of the respondents) are subtracted from the promoters (in percent of the respondents).

$$NPS = promoters - detractors \qquad \textit{(in \% of all respondents)}$$

The result of the NPS is a value between +100 and -100.

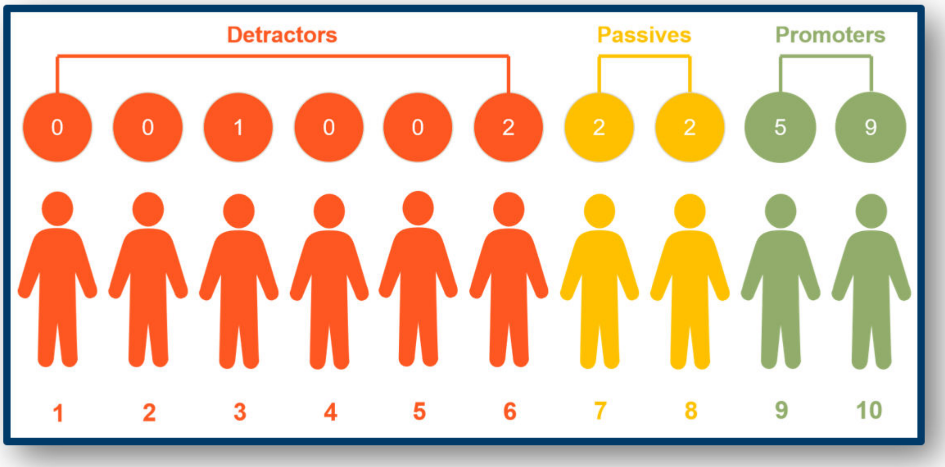


Figure 4.15.: NPS across all case organizations

Figure 4.15 shows the number of detractors, passives and promoters. It can be seen that 4 out of 21 interviewees fall out of the evaluation due to having given a score of 7 or 8. In total there are 3 detractors and 14 promoters.

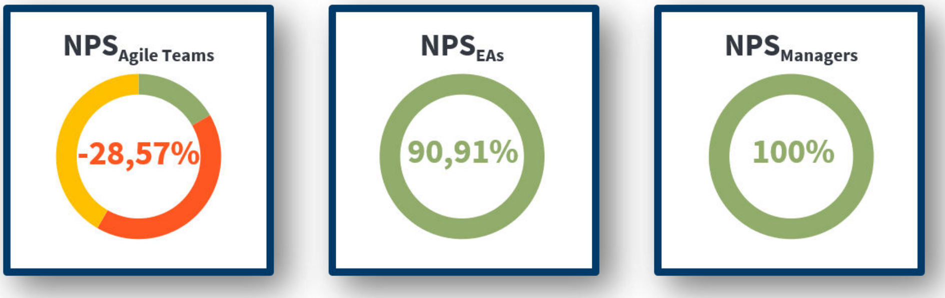


Figure 4.16.: Overview of the different NPS score for the respective stakeholder groups

Figure 4.16 shows the NPS per stakeholder group. Interestingly, the stakeholder group AT, currently with -28.57%, clearly refrains from recommending EAs to others. This supports the findings of the interviews. The stakeholder group EA considers its role to be very relevant (90.91%). In addition, interestingly, stakeholder group M recommends the role of EA with 100%. It seems that there is no transparency in the case organizations between team and organizational level. Stakeholder group M has a different perception of the cooperation between the EA and the AT.

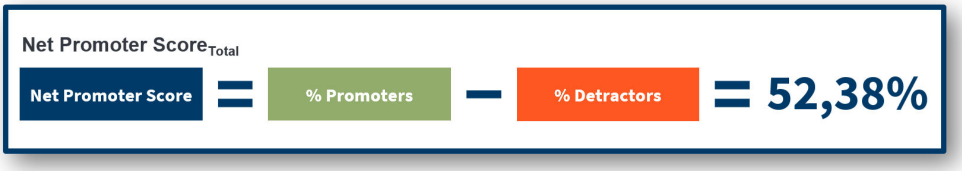


Figure 4.17.: NPS score across all stakeholder groups

Figure 4.17 shows the NPS across all stakeholder groups. The recommendation rate of 52.38% for the EA confirms the relevance of the role, but also shows that not everyone sees the added value.

4.5.3. Enterprise Architecture Management Model

Figure 4.18 shows the to-be EAM Model that has been created based on the findings of this master's thesis. IT describes a to-be EAM model for the agile environment based on the findings of the case study and the agile EAM model [101]. The model is divided into five areas. Starting at the top left, moving clockwise, four different architectural levels - marked in blue - of a company are described. The iterative EAM process, divided into the phases: Plan, Do, Check, and Act (PDCA) [116], is at the center of the model. Each level works closely with the EAM process and the enterprise architecture team. In the following, the interaction between the individual architectural levels and the EAM process as a link is described in more detail.

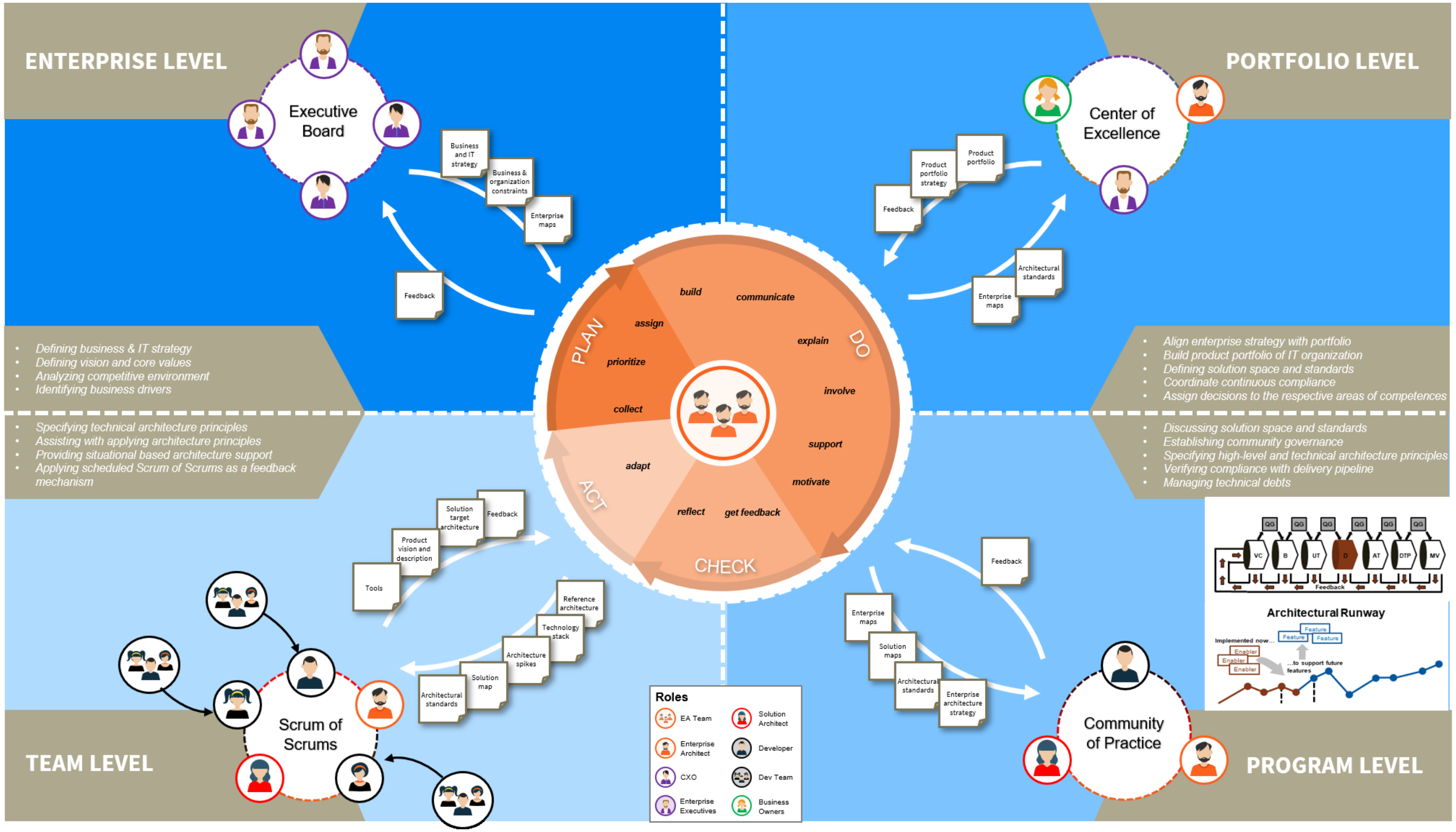

ENTERPRISE LEVEL
Executive Board
Business and IT strategy
Business & organization constraints
Enterprise maps
Feedback
Defining business & IT strategy
Defining vision and core values
Analyzing competitive environment
Identifying business drivers
PORTFOLIO LEVEL
Center of Excellence
Product portfolio
Product portfolio strategy
Feedback
Architectural standards
Enterprise maps
Align enterprise strategy with portfolio
Build product portfolio of IT organization
Defining solution space and standards
Coordinate continuous compliance
Assign decisions to the respective areas of competences
PLAN
assign
prioritize
collect
build
communicate
explain
DO
involve
support
motivate
get feedback
reflect
CHECK
adapt
ACT
Specifying technical architecture principles
Assisting with applying architecture principles
Providing situational based architecture support
Applying scheduled Scrum of Scrums as a feedback mechanism
Discussing solution space and standards
Establishing community governance
Specifying high-level and technical architecture principles
Verifying compliance with delivery pipeline
Managing technical debts
Solution target architecture
Feedback
Product vision and description
Tools
Reference architecture
Technology stack
Architecture spikes
Solution map
Architectural standards
TEAM LEVEL
Scrum of Scrums
Roles
EA Team
Solution Architect
Enterprise Architect
Developer
CXO
Dev Team
Enterprise Executives
Business Owners
Feedback
Enterprise maps
Solution maps
Architectural standards
Enterprise architecture strategy
Community of Practice
PROGRAM LEVEL
QG
VC
B
UT
D
AT
DTP
MV
Feedback
Architectural Runway
Implemented now...
Enabler
Feature
...to support future features

Figure 4.18.: To-be EAM Model

The enterprise level (top left) describes the highest level. The task of the stakeholders is to define the business and IT strategy and the vision and core values. On the other hand, business drivers are identified and the competitive environment analyzed. Decisions and agreements are made in the EB by the enterprise executives (EE) and the C-level. Decisions made here are binding for the entire organization. Artifacts such as "business & IT strategy", "business & organization constraints" and "feedback" flow as input from the enterprise level into the EAM process for further processing and specification. Enterprise maps flow back from the EAM process to the enterprise level.

The portfolio level (top right) describes the second highest level. The task of the department heads is to consider the artifacts provided by the EA team, such as the enterprise map and the architectural standards, when making decisions and to transfer them to the CoE. At this architectural level, the goal is to align the enterprise strategy with the portfolio, create the product portfolio and define the solution space and standards. In addition, the continuous compliance of architectural principles is coordinated on the one hand, and on the other hand decision responsibility, such as specific technical architectural decisions within the solution space, is delegated to the corresponding competence. The CoE is used as the decision-making mechanism. The CoE functions as a group of experts for disseminating specific knowledge, decision making and substantive discussion of changes to the product portfolio. Participants of the CoE are EE, EA and business owner (BO). At this level, product portfolio, product portfolio strategy and feedback flow back into the EAM process for further adaptation and processing. The EAM process results in the artifacts "enterprise map" and "architectural standards" relevant to the portfolio.

The program level describes the third architectural level. Here, a program represents a group of several ATs. At this level, standards and solution spaces are discussed, community governance is established, high-level and technical architecture principles are specified collaboratively, and compliance with them is monitored using a delivery pipeline. In addition, technical debts are managed. The architectural runway also ensures that important architectural features are implemented in advance. The focus is on the so-called CoPA. With the help of the CoPA, problems are discussed and the tasks arising at this level are solved. EAs, developers and SAs sit in the community and discuss important topics and issues, taking into account the artifacts provided by the EA team such as the enterprise and solution map, the enterprise architecture strategy and the architectural standards. The program level returns important feedback on the artifacts provided to the EAM process. Based on the feedback, this process adapts the provided artifacts and thus prevents technical debts.

The team level describes the fourth enterprise level. The task of the ATs, EAs, and SAs is to define the technical architecture principles and to specify the high-level principles defined at program level. On the other hand, the individual roles should support each other in the application of the architectural principles. The EAs will provide assistance depending on the situation or on purely architecture-related questions and problems. In addition, the so-called scrum of scrums (SoS) is used at team level as a reactive feedback mechanism. Participants of the SoS are primarily developers, EAs, and SAs. The goal of the SoS is to find solutions for problems and questions. In order to enable the most detailed support and successful working methods possible, the artifacts ″solution target architecture″, ″product vision and description″, ″tools″ and ″feedback″ from the ATs flow as input into the EAM process. Artifacts such as ″reference architecture″, ″architecture standards″, ″solution maps″, ″technology stacks″ and ″architecture spikes″ flow from the EAM process back to the ATs.

The EAM process begins with the PLAN phase. Here, information is collected from the artifacts provided by the individual architectural levels, prioritized according to relevance, quality, binding force or availability, and assigned to the person responsible. Based on these artifacts, new artifacts or models are created in phase 2 - referred to as DO. These are then communicated and explained to the individual stakeholders. Here, the EA team has to ensure that all necessary stakeholders are involved, that questions are clarified immediately and that inconclusive stakeholders are convinced of the new artifacts or models. Phase 3, ″Check″, describes the feedback mechanism. After Phase 2, the aim is to obtain and reflect feedback, for example on the artifacts or models made available. In phase 4, necessary adjustments to the artifacts or models will be made based on the insights gained in phase 3. The iterative EAM process then starts over again.

5. Discussion

This chapter starts of with the key findings of this master's thesis in Section 5.1. Afterwards, a set of recommendations of action based on the key findings are presented in Section 5.2. Last but not least, the limitations of this thesis are discussed in Section 5.3.

5.1. Key Findings

In the following, the key findings of this master's thesis are presented:

- **The majority of architecture principles are defined on the IT portfolio level.** The fact that the architecture principles are defined at IT portfolio level shows that the architecture specifications are primarily used to establish common cross-team architectural standards.

- **There is no explicit control mechanism for the verification of compliance with architecture principles.** Due to the lack of capacity of EAs, currently established control mechanisms for the verification of architecture principles are based on trust and performed either sporadically or not even available.

- **There are difficulties when introducing and implementing architecture principles.** Although EAs are using CoPAs and Confluence for communicating and documenting architecture principles, ATs have trouble accessing them. One reason for this is the indirect communication between EAs and ATs, because most of the communication is performed between EA and SA and not directly with ATs. Furthermore, the documentation of architecture principles is poor, which leads to difficulties understanding it.

- **Creation and definition of solution space is an important architectural decision in an agile environment.** The solution space is a document that specifies which architectural principles, standards and guidelines must be adhered to, which architectural goals must be met, which code libraries or patterns exist or are already available, and which dependencies exist to other applications at the beginning of a project or product development. The advantages of defining a solution space for ATs are the ability to show alternatives and the specification of certain architecture guidelines without restricting their freedom too much and thereby hindering agility. The definition and use of a solution space solves the classical problem of the ATs stumbling around without a guideline by giving them a general framework to work with.

- **EAs' responsibility has shifted from directing and controlling towards supporting and enabling of ATs in an agile environment.** The traditional role of the EAs restricts the agility of the ATs by prescribing strict processes. Therefore, the old role model alone is no longer suitable. As the teams work agilely, more and more responsibility lies within the teams themselves which results in less decision-making authority on the part of the EAs. This also applies to architectural decisions. In addition, due to the changed role model towards the supporting function instead of the pure default, an EA has fewer escalation possibilities.

- **The required characteristics and capabilities of EAs have changed in an agile environment.** The EAs competence to convince various stakeholders, the ability to provide delivery pipelines, and to have a technical understanding of different technologies makes it easier to increase the intrinsic motivation of ATs to adhere to architectural standards.

- **The working methodology of EAs has changed in agile environment.** Due to the adaption to agile procedures, decisions are made collaboratively. EAs are focusing on the provision of architectural spikes to reduce the risk of technical problems and the technical understanding of the used frameworks and tools. In an agile environment, EAs must accompany ATs much longer than before, as the main activity of EAs formerly was primarily in the beginning of the project in the long conception phase. This has changed, since the conception takes place iteratively in an agile environment and an AT is supported by an EA from start to finish.

- **The role of the supporting architect is necessary in an agile environment.** Because of the adaption to agile procedures and the associated shift from direct and control to support and enablement of ATs, the importance of the role of the supporting architect increases due to more proximity to the solutions and ATs and therefore a better understanding of ATs. In addition, the solution architect provides direct feedback and achieves acceptance through know-how and explanation.

- **New requirements for the role of EA cause several issues in the execution of tasks and responsibilities.** Due to the change of roles and responsibilities of EAs in an agile environment, EAs face new issues. Because of their restricted capacity, the EAs are not able to satisfy the needs of the ATs and sometimes are becoming a bottleneck. Specifically, they are not able to guide all products / projects of the ATs and are not able to provide fast feedback or support to the ATs which leads to dissatisfaction of ATs.

- **The communication between EAs and ATs is mainly indirect.** The communication between ATs and EAs is handled by a go-between such as solution architects or domain architects, mainly because of capacity problems, which is not considered to be ideal because information can be lost and misunderstandings can arise.

- **The as-is and to-be involvement of ATs in architecture processes differs immensely.** Currently, there is little or no involvement of ATs in the architecture processes. A strong involvement is considered to be the desired status among all stakeholder groups in order to give ATs the ability to participate in the decision-making process and thus to facilitate self-determination and -organization.

- **There is no formal mechanism for ATs to provide feedback to EAs.** Due to the ongoing transformation processes, formal feedback mechanisms have not been established yet. This decreases the quality of artifacts provided by the EA and leads to dissatisfaction of ATs.

- **Measurement of the value contribution of the EAM is difficult.** The EAM is difficult to monitor because of its complexity and interrelations to different areas.

- **Value contribution of EA across all stakeholder groups differs significantly.** The added value of the EA has not yet arrived at team level. This may be due to the fact that the EA does not meet the expectations of the ATs and acceptance problems arise as a result.

- **Gap between self-perception and external perception differs significantly among stakeholder groups leading to different expectations.** In general, EAs are assessing themselves better on their value contribution than perceived by ATs.

- **Value contribution of an EA across all stakeholder groups confirms the relevance of the role, but also shows that not everyone sees the added value (52.38%).** The NPS regarding the value contribution of an EA differs significantly. Stakeholder group AT clearly refrains from recommending the EA to others (-28.57%). The stakeholder group EA considers its role to be very relevant (90.91%). Stakeholder group M recommends the role of the EA with 100%. So far, ATs have little working experience with EAs and because of that, they do not understood their value yet. The Ms, who work in close collaboration with the EAs, have a better understanding of the value contribution of the EAs.

5.2. Recommendations Of Action

The following section provides a set of recommendations for action based on the key findings (see Section 5.1).

- **The majority of architecture principles are defined on the IT portfolio level.** Architecture principles should be defined on IT portfolio level to establish common cross-team architectural standards.

- **There is no explicit control mechanism for the verification of compliance with architecture principles.** In order to ensure that ATs adhere to architecture standards, explicit control mechanisms must be established. Otherwise it cannot be guaranteed that ATs comply with architecture principles and standards. This problem can be solved, for instance, with the help of automated and standardized code reviews inside of a delivery pipeline.

- **There are difficulties when introducing and implementing architecture principles.** In order to introduce and implement architecture principles in ATs properly, a direct communication between EAs and ATs is essential. Furthermore, EAs need to have enough time, know-how, and suitable social skills to support and enable ATs.

- **Creation and definition of solution space is an important architectural decision in an agile environment.** The definition of a solution space in advance ensures compliance with relevant architectural standards, guidelines, and objectives and helps not to restrict the freedom of the ATs and thus of the agility. On the other hand, the solution space allows ATs certain freedom by allowing emergent architecture development.

- **EAs' responsibility has shifted from directing and controlling towards supporting and enabling of ATs in an agile environment.** The role of the supporting EA will become more and more important in the future. For this role to be lived, as it should be, sufficient capacity of EAs must be available. Supporting and enabling of ATs over a longer period of time takes time and effort. Especially since the intensity increases with the new responsibility.

- **The required characteristics and capabilities of EAs have changed in an agile environment.** The required characteristics of the role EA changed in an agile environment due to the shift from direct and control to support and enable. For instance, on the one hand, an EA needs to have deep technical know-how in order to be able to code for short demonstration. On the other hand, the EA needs to be methodically trained and must have helpful social skills, such as sensitivity and personal skills, to be able to work with and motivate different stakeholders. Therefore, companies either need to empower the own EAs through trainings or search for different skills when hiring, e.g. social skills and technical know-how.

- **The working methodology of EAs has changed in agile environment.** Due to agile procedures, an increased interaction between EAs and ATs is needed, which requires specific social skills such as sensitivity and personal skills. Additionally, EAs are applying lean and agile practices in order to support ATs. For instance, architecture spikes as recommended by XP or Scrum for prioritizing topics or Kanban for optimizing workflows should be used.

- **The role of supporting architect is necessary in an agile environment.** The role of supporting architect needs to be established because of the different requirements, needed in an agile environment, such as the ability to code for short demonstration and to increase the intrinsic motivation of ATs to adhere to architecture principles.

- **New requirements for the role of EA cause several issues in the execution of tasks and responsibilities.** Due to the new responsibilities, the workload of EAs has increased. Companies need to be aware of this and adapt accordingly. To ensure a proper support and enablement, the availability of EAs needs to be guaranteed by providing the right organizational structure and a suitable lightweight EAM process.

- **The communication between EAs and ATs is mainly indirect.** Indirect communication leads to loss of information. That is why direct communication between AT and EA should be mainly used. Therefore, companies should provide appropriate coordination arenas.

- **The as-is and to-be involvement of ATs in architecture processes differs immensely.** ATs should be involved strongly in the architecture processes in order to improve the quality of architecture through constant feedback.

- **There is no formal mechanism for ATs to provide feedback to EAs.** Formal feedback mechanisms should be established. For example, SoS (with representatives from the teams) act as reactive feedback mechanisms and optional one-on-one meetings to support more direct communication between ATs and EAs and thus improve collaboration. Furthermore, CoPAs should be implemented. The latter can save time and capacity for EAs and increase the value of feedback.

- **Measurement of the value contribution of the EAM is difficult.** Value contribution of the EAM should be measured by combining specific KPIs, such as budget, correctness, customer satisfaction, feedback from ATs and products, number of adjustments and speed of decision making.

- **Value contribution of EA and the gap between self-perception and external perception differs significantly across all stakeholder groups.** Reflection and stronger communication should be established to raise awareness of the value contribution of EAs and close the gap between self-perception and external perception. With the implementation of the PDCA described in the to-be EAM model in Section 4.5.3, this problem can be solved.

5.3. Limitations

This master's thesis has a few limitations which must be mentioned here. The potential threats to the validity of this master's thesis will be discussed based on Runeson and Höst's [122] assessment criteria. The threats to validity can be divided into four types: construct validity, internal and external validity and reliability. Since this master's thesis does not seek to establish any causal relationships, threats to internal validity are not discussed.

Construct validity: The validity reflects to what extent the operational measures that are studied really reflect what the researcher has in mind, and what is being studied according to the research questions [122]. To minimize this threat, three countermeasures were taken. First, multiple sources were used for data collection. Those included semi-structured interviews with different stakeholder groups, unstructured interviews with enterprise architects and agile coaches during three workshops and studying internal documentations. Second, workshop protocols and interviews were coded by the author of this master's thesis and reviewed by a second researcher. Third, key informants of the case organizations have reviewed a draft of this master's thesis.

External validity: The threat to external validity relates to what extent the findings can be generalized [122]. This aspect of validity is addressed in this multiple-case study by using literal replication. In addition, this master's thesis focuses on analytical generalization [122] by providing a profound description of the cases. In particular, this multiple-case study provides empirical insights that allow for a deep understanding of the importance of the EAM in an agile environment. The results shown should be seen as valuable insights for other organizations facing challenges similar to those of the case organizations.

Reliability: The threat to reliability relates to what extent the data and the analysis depend on the researcher and whether repeating the multiple-case study would produce the same results [122]. Three countermeasures have been taken to counter this threat. First, a case study database was created containing notes and documents of the case studies, such as audio recordings, interview guidelines, and workshop protocols. Second, a case study protocol with detailed procedures for data collection and analysis was specified. Third, the multiple-case study was designed in such a way that most of respondents and multiple interviewers allowed for data and observer triangulation.

6. Conclusion and Outlook

This chapter summarizes the master's thesis in Section 6.1 and provides a brief outlook of possible future investigations in Section 6.2.

6.1. Summary

The EAM has established itself as an important management function for large IT organizations by aligning both business and IT with strategy and goals and providing appropriate governance mechanisms. Over the last years, frequent changes in market conditions, technical and regulatory changes and the necessity to reduce costs force companies to perform / carry out complex business transformations at more and more frequent intervals. The result is that large IT organizations must be able to act flexibly and in a timely manner. The IT acts as an enabler for transformation and represents an important success factor for the delivery of customer value on time. However, in order to be able to achieve this or provide sufficient support, the IT must continuously align itself with the business and also ensure that business and IT are aligned. In order to realize the agile requirements coming from the business, the IT must also adapt to the agility. Thereby, in addition to software development, the EAM in particular must adapt to the agile working methodology. However, this proves to be difficult because the two approaches are fundamentally contradictory.

By conducting a multiple embedded case study, this master's thesis provided a collection of specific recommendations for action to improve the collaboration between enterprise architects and agile teams and thereby support large-scale agile development. Based on four research questions (see Section Objectives) this master's thesis examined the field of tension between agile and traditional EAM and analyzed the status quo of agile principles, circles, the role of the EA team and the value contribution of the EAM. In total, 64 semi-structured interviews with 21 persons from the three stakeholder groups: AT, enterprise architecture, and management were conducted.

The results indicate that the lack of capacity of enterprise architects hampers the support of agile teams and is the reason why currently established control mechanisms for the verification of architecture principles are either based on trust, performed or either sporadically or not performed. In addition, large IT organizations seem to lack appropriate scaling options at portfolio and organizational levels. Furthermore, bad communication and the lack of technical know-how and feedback mechanisms of EAs leads to superficial artifacts provided to the ATs, which causes the latter's satisfaction rate to drop. In addition, the required characteristics of the role of the EA changed in an agile environment due to the shift from direct and control to support and enable, which forces large IT organizations to either empower their own architects through trainings or search for specific characteristics when hiring EAs. Nevertheless, the new requirements for the role of EA cause several issues in the execution of tasks and responsibilities, since their workload increases. In addition, the findings show that the EAs' value contribution for supporting ATs has not yet arrived at the team level, leading to significant acceptance issues. Finally, the gap between self-perception and external perception differs significantly and leads to different expectations in general. In order to solve these issues, this master's thesis provided specific recommendations for action and a to-be EAM model. Due to the changing requirements in an agile environment, the focus should shift from classic EAs to the role of supporting EAs. The latter requires a higher capacity to handle the increased workload by supporting and enabling ATs over longer periods of time. To solve the capacity problem, large IT organizations must adapt accordingly and increase the number of EAs. In addition, support and enablement require new skills such as deep technical know-how and social skills to overcome the acceptance problem in particular. Therefore, an organization-wide change is essential. To solve the capacity problem, large IT organizations must adapt accordingly and increase the number of EAs. Another recommended course of action is to define a solution space in advance, in order to ensure compliance with the architecture of relevant standards and guidelines, which helps not to restrict the freedom of ATs and thus agility. Also, to ensure that ATs comply with the architecture standards, explicit control mechanisms such as automated and standardized code reviews must be established within a delivery pipeline. A further measure of assistance can be the introduction of feedback cycles, which increase communication between the various stakeholders and thus often increase awareness of the added value of EAs. With the implementation of the PDCA described in the to-be EAM model in Section 4.5.3, this master's thesis provides an initial approach to decrease the problems mentioned above.

6.2. Outlook

Since the case organizations are currently undergoing a transformation process over a longer period of time, we plan to conduct long-term studies on the collaboration between EAs and ATs. First and foremost, we plan to study how our case organizations will solve contemporary challenges in the future, such as the communication gaps between EAs and ATs, the restrictive capacities of EAs, and the interplay between top-down and bottom-up governance. We also want to investigate how the traditional role model of EAs will unfold in the case organizations in the future. Moreover, we aim to explore what value EAs can deliver to ATs and how the value contribution can be measured. Finally, we will strengthen our current findings by a complementary quantitative study which aims, among others things, to identify recurring challenges and best practices of EAs and ATs.

With respect to future research, we encourage other researchers to investigate appropriate coordination mechanisms that can help to close the communication gaps between EAs and ATs. We also suggest that the research community study the collaboration between EAs and ATs from a sociological point of view, e.g. by applying multi-team systems theory from sociology [97]. In future studies, researchers should also evaluate the applicability of various governance styles, such as top-down prescriptive governance, community governance or self-governance [96], in an agile environment and develop approaches in order to implement the best choice.

A. Appendix

A.1. Semi-structured Interviews

A.1.1. General Information

1. **Section: Overview of the transformation**
   a) What are the reasons for the transformation?
   b) What are the goals of the transformation?
   c) When did the transformation begin?
   d) How did the transformation take place?
   e) What success have you achieved so far?
2. **Section: Agile methods and practices**
   a) Which agile and large scale agile methods / frameworks are used in your organization?
3. **Section: Challenges and solutions**
   a) What are the greatest challenges of the transformation?
   b) How did you try to solve the challenges?
4. **Section: Plans for the future**
   a) What are the next steps?
   b) What are possible stumbling blocks?
   c) On a scale from 1 ″no enablement at all″ to 10 ″very strong enablement″, how strong does the EAM enable the scaling of agile practices?

A.1.2. Architecture Principles

1. **Section: Background questions**
    a) Which role description applies to you?
    b) For how many years have you been working in the field of agile software development or EAM?
    c) How is your EAM organization structured? Which architectural roles do you have and how are they assigned to the agile teams?

2. **Section: Drivers and Goals**
    a) Which driving forces are significantly responsible for creating architectural principles?
    b) What are your goals in defining architectural principles?

3. **Section: Specification and Classification**
    a) Which architectural principles do you use in an agile environment? Which architectural principles from the attached Excel list do you use? (Please specify them in the Excel list)
    b) Are there architectural principles that are unsuitable for the agile environment or which are not applied? If so, why?
    c) Who is responsible for the creation and specification of the architectural principles?
    d) Is there a guideline for creating architectural principles?
    e) (If yes): With which specifications?
    f) (If yes): Who created this guideline and is responsible for its maintenance?

4. **Section: Application and Compliance**
    a) When do new architectural principles become valid?
    b) What measures exist for the implementation of the newly created architectural principles?
    c) In which way is compliance with the architectural principles checked and ensured?
    d) Which tools are used to test compliance with architectural principles?
    e) What problems do you identify when introducing or implementing architectural principles?

5. **Section: Discussion**
    a) Generally speaking, what do you expect from the cooperation with TUM?

b) If necessary, can we contact you again in the context of the case study? If yes, please enter your name and email address. Naturally, we will not make this data available to third parties.

c) Are there any comments or open questions?

A.1.3. Architecture Boards

1. **Section: Background Information**

   a) Which role description applies to you?

   b) For how many years have you been working in the field of agile software development or EAM?

2. **Section: Architectural Decisions**

   a) Name typical examples of architectural decisions in an agile environment.

   b) How do you categorize architectural decisions?

   c) How and where are architectural decisions documented?

   d) At what level and by which role is an architectural decision made?

3. **Section: Architecture Boards**

   a) What forms of architectural boards do you have in your company? Please specify the existing forms in the attached Excel list.

4. **Section: Discussion**

   a) Generally speaking, what do you expect from the cooperation with TUM?

   b) If necessary, can we contact you again in the context of the case study? If yes, please enter your name and email address. Naturally, we will not make this data available to third parties.

   c) Are there any comments or open questions?

A.1.4. Role of Enterprise Architecture Team

1. **Section: Background Information**

   a) Which role description applies to you?

   b) For how many years have you been working in the field of agile software development or EAM?

2. **Section: Responsibility**

   a) What responsibilities do Enterprise Architects have in your company?

   b) How did the responsibilities of Enterprise Architects change in an agile environment?

c) What distinguishes your Enterprise Architects and how do they differ from the classic role of Enterprise Architects in other companies?

d) How did the working methodology of the Enterprise Architect change in an agile environment?

e) How did the environment of the Enterprise Architect change in an agile environment?

3. **Section: Collaboration**

a) How should Enterprise Architects support agile programs and agile transformation?

b) How is the role of the architect practiced in your company?

c) What are the characteristics of the role selected in question 3.2?

d) What are the advantages and disadvantages of the role mentioned in question 3.2?

e) In your opinion, which of the roles listed in question 3.2 should be lived more strongly in the future?

f) Should Enterprise Architects be part of the agile teams?

g) To what extent should Enterprise Architects be involved in the agile programs?

4. **Section: Artifact**

a) What artifacts are provided by the Enterprise Architects to the stakeholders in an agile environment?

b) What artifacts should be provided by Enterprise Architects to stakeholders in the future?

5. **Section: Problems and Outlook**

a) What problems do Enterprise Architects encounter in an agile environment?

b) In your opinion, how should these problems be addressed?

c) What recommendations would you give Enterprise Architects in an agile environment?

d) On which topics should Enterprise Architects focus in the future?

6. **Section: Discussion**

a) Generally speaking, what do you expect from the cooperation with TUM?

b) If necessary, can we contact you again in the context of the case study? If yes, please enter your name and email address. Naturally, we will not make this data available to third parties.

c) Are there any comments or open questions?

A.1.5. Value Contribution of Enterprise Architecture Management

1. **Section: Background Information**
    a) Which role description applies to you?
    b) For how many years have you been working in the field of agile software development or EAM?
2. **Section: Enabling**
    a) What information do Enterprise Architects need from agile teams to provide appropriate architectural models?
    b) Which architectural models are provided to the agile teams by the Enterprise Architects?
    c) What do agile teams expect from Enterprise Architects' architectural models?
    d) Are the expectations regarding the architectural models fulfilled? (On a scale from 1 ″not fulfilled″ to 10 ″more than fulfilled″)
    e) Please give reasons for your answer:
3. **Section: Cooperation**
    a) Is it difficult to find an Enterprise Architect as a contact person?
    b) How do you rate the availability of an Enterprise Architect? (On a scale from 1 ″not available at all″ to 10 ″always available″)
    c) How do Enterprise Architects communicate with agile teams?
    d) What do agile teams expect from the communication with Enterprise Architects?
    e) Are expectations in terms of communication fulfilled? (On a scale from 1 ″not fulfilled″ to 10 ″more than fulfilled″)
    f) Please give reasons for your answer:
    g) To what extent are agile teams involved in the architecture processes that are relevant to them?
    h) To what extent should agile teams be involved in the architecture processes that are relevant to them?
    i) Are the expectations in terms of the integration into architectural processes fulfilled? (On a scale from 1 ″not fulfilled″ to 10 ″more than fulfilled″)
    j) Please give reasons for your answer:
    k) How do Enterprise Architects support agile teams?
    l) What kind of support do agile teams expect from Enterprise Architects?

m) Are the expectations in terms of support fulfilled? (On a scale from 1 ″not fulfilled″ to 10 ″more than fulfilled″)

n) Please give reasons for your answer:

o) In what form and frequency can the agile teams give feedback to Enterprise Architects?

p) In what form should feedback from agile teams be given to Enterprise Architects?

q) Are the expectations in terms of feedback fulfilled? (On a scale from 1 ″not fulfilled″ to 10 ″more than fulfilled″)

r) Please give reasons for your answer:

s) On a scale from 1 ″not fulfilled″ to 10 ″more than fulfilled″, how likely is it that you would recommend the EA? (To an agile team that has developed without the support of an Enterprise Architect)

4. **Section: Review**

   a) How does feedback from agile teams affect the EAM?

   b) Which criteria are currently used to assess the value contribution of the EAM and which metrics are used to measure it?

   c) How could the value contribution be measured in the future?

5. **Section: Governance**

   a) To what extent do agile teams want to be controlled by the Enterprise Architects?

   b) Which requirements can help?

   c) Which specifications would not be helpful or restrictive?

6. **Section: Discussion**

   a) Generally speaking, what do you expect from the cooperation with TUM?

   b) If necessary, can we contact you again in the context of the case study? If yes, please enter your name and email address. Naturally, we will not make this data available to third parties.

   c) Are there any comments or open questions?

Bibliography

[1] Mohd Abul Ala Abid. Information technology infrastructure library (itil). *Information Technology*, 1(1), 2012.

[2] Frederik Ahlemann, Eric Stettiner, Marcus Messerschmidt, and Christine Legner. *Strategic enterprise architecture management: challenges, best practices, and future developments*. Springer Science & Business Media, 2012.

[3] Stephan Aier, Christian Fischer, and Robert Winter. Construction and evaluation of a meta-model for enterprise architecture design principles. 2011.

[4] Stephan Aier, Stephan Kurpjuweit, Jan Saat, and Robert Winter. Enterprise architecture design as an engineering discipline. *AIS Transactions on Enterprise Systems*, 1(1):36–43, 2009.

[5] Stephan Aier, Christian Riege, and Robert Winter. Unternehmensarchitektur–literaturüberblick und stand der praxis. *Wirtschaftsinformatik*, 50(4):292–304, 2008.

[6] Scott W Ambler. Agile enterprise architecture. *Agile Enterprise Architecture: Beyond, Enterprise DataModeling*, 2009. accessed: 2018-11-30.

[7] Scott W. Ambler. Agile architecture: Strategies for scaling agile software development, 2010. accessed: 29/08/2018.

[8] Ross J Anderson. *Security engineering: a guide to building dependable distributed systems*. John Wiley & Sons, 2010.

[9] Felix Bachmann, Robert L Nord, and Ipek Ozakaya. Architectural tactics to support rapid and agile stability. Technical report, Carnegie-Mellon Univ Pittsburgh PA Software Engineering Inst, 2012.

[10] Udayan Banerjee. Agile development and enterprise architecture practice - can they coexist, 2011. accessed: 30/08/2018.

[11] John Beachboard, Kregg Aytes, and Jack Probst. It governance and it management: Is there a difference that makes a difference? In *Proceedings of Informing Science & IT Education Conference (InSITE)*. Citeseer, 2010.

[12] Kent Beck, Mike Beedle, Arie Van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, et al. Manifesto for agile software development. 2001.

[13] Peter Beijer and Theo de Klerk. *IT Architecture - Essential Practice for IT Business Solutions*. Lulu. com, 2010.

[14] Stefan Bente, Uwe Bombosch, and Shailendra Langade. *Collaborative enterprise architecture: enriching EA with lean, agile, and enterprise 2.0 practices*. Newnes, 2012.

[15] Peter Bernus, Laszlo Nemes, and Günter Schmidt. *Handbook on enterprise architecture*. Springer Science & Business Media, 2012.

[16] Sabine Buckl. Developing organization-specific enterprise architecture management functions using a method base. 2011.

[17] Sabine Buckl, Alexander M Ernst, Josef Lankes, Florian Matthes, and Christian M Schweda. Enterprise architecture management patterns–exemplifying the approach. In *Enterprise Distributed Object Computing Conference, 2008. EDOC'08. 12th International IEEE*, pages 393–402. IEEE, 2008.

[18] Sabine Buckl, Florian Matthes, Ivan Monahov, Sascha Roth, Christopher Schulz, and Christian M Schweda. Towards an agile design of the enterprise architecture management function. In *Enterprise Distributed Object Computing Conference Workshops (EDOCW), 2011 15th IEEE International*, pages 322–329. IEEE, 2011.

[19] Sabine Buckl, Florian Matthes, Christian Neubert, and Christian M Schweda. A wiki-based approach to enterprise architecture documentation and analysis. In *ECIS*, pages 1476–1487, 2009.

[20] Sabine Buckl, Florian Matthes, and Christian M Schweda. A viable system perspective on enterprise architecture management. In *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on*, pages 1483–1488. IEEE, 2009.

[21] Sabine Buckl, Florian Matthes, and Christian M Schweda. Socio-technic dependency and rationale models for the enterprise architecture management function. In *International Conference on Advanced Information Systems Engineering*, pages 528–540. Springer, 2011.

[22] Sabine Buckl and Christian M Schweda. On the state-of-the-art in enterprise architecture management literature. 2011.

[23] Sabine Buckl, Christian M Schweda, and Florian Matthes. A design theory nexus for situational enterprise architecture management. In *Enterprise Distributed Object Computing Conference Workshops (EDOCW), 2010 14th IEEE International*, pages 3–8. IEEE, 2010.

[24] Axel Buecker, Paul Ashley, Martin Borrett, Ming Lu, Sridhar Muppidi, Neil Readshaw, et al. *Understanding SOA security design and implementation*. IBM Redbooks, 2008.

[25] Mert Canat, Núria Pol Català, Alexander Jourkovski, Svetlomir Petrov, Martin Wellme, and Robert Lagerström. Enterprise architecture and agile development: Friends or foes? In *2018 IEEE 22nd International Enterprise Distributed Object Computing Workshop (EDOCW)*, pages 176–183. IEEE, 2018.

[26] Yolande E Chan and Blaize Horner Reich. It alignment: what have we learned? *Journal of Information technology*, 22(4):297–315, 2007.

[27] Tjan Hien Cheng, Slinger Jansen, and Marc Remmers. Controlling and monitoring agile software development in three dutch product software companies. In *Proceedings of the 2009 ICSE Workshop on Software Development Governance, SDG 2009*, 2009.

[28] Alistair Cockburn. *Agile software development: the cooperative game*. Pearson Education, 2006.

[29] IEEE Computer Society. Software Engineering Standards Committee and IEEE-SA Standards Board. Ieee recommended practice for software requirements specifications. Institute of Electrical and Electronics Engineers, 1998.

[30] Daniela S Cruzes and Tore Dyba. Recommended steps for thematic synthesis in software engineering. In *Empirical Software Engineering and Measurement (ESEM), 2011 International Symposium on*, pages 275–284. IEEE, 2011.

[31] Thomas H Davenport, H Hammer, and Tauno J Metsisto. How executives can shape their company's information systems. *Harvard Business Review*, 67(2):130–134, 1989.

[32] Steven De Haes and Wim Van Grembergen. It governance and its mechanisms. *Information Systems Control Journal*, 1:27–33, 2004.

[33] Jan LG Dietz. *Architecture: building strategy into design*. Academic Service The Hague, 2008.

[34] Alexander M Ernst. *A pattern-based approach to enterprise architecture management*. PhD thesis, Technische Universität München, 2010.

[35] L Fehskens. What the "architecture" in "enterprise architecture" ought to mean. In *Open Group conference, Boston. The Open Group, Reading Google Scholar*, 2010.

[36] Christian Fischer, Robert Winter, and Stephan Aier. What is an enterprise architecture principle? In *Computer and Information Science 2010*, pages 193–205. Springer, 2010.

[37] Organisation for Economic Co-operation and Development. *G20/OECD Principles of Corporate Governance*. OECD, 2015.

[38] Sallyann Freudenberg and Helen Sharp. The top 10 burning research questions from practitioners. *Ieee Software*, 27(5):8–9, 2010.

[39] U. Friedrichsen and I. Schrewe. *Leichtgewichtige Unternehmensarchitekturen - Wie Agilität bei der Einführung eines EA Management helfen kann*. In OBJEKTspektrum. EAM/2010, 2010.

[40] k. Fuhrer. "teil 2: Standardblickwinkel und praxissichten planung und steuerung." in: Enterprise architecture deliverables: Welche ergebnisse liefert enterprise architecture? 2011.

[41] k. Fuhrer. "teil 3: Standardblickwinkel und praxissichten geschäft und it-entwicklung." in: Enterprise architecture deliverables: Welche ergebnisse liefert enterprise architecture? 2011.

[42] k. Fuhrer. "teil 4: Standardblickwinkel und praxissichten infrastruktur und technologie, resümee und ausblick." in: Enterprise architecture deliverables: Welche ergebnisse liefert enterprise architecture? 2011.

[43] Gabler. Corporate governance. `https://wirtschaftslexikon.gabler.de/definition/corporate-governance-28617/version-252243`, 2018. accessed: 2018-08-21.

[44] Markus Gaulke. *Praxiswissen COBIT: Grundlagen und praktische Anwendung in der Unternehmens-IT*. dpunkt. verlag, 2014.

[45] Stuart Gillan and Laura T Starks. A survey of shareholder activism: Motivation and empirical evidence. pages 10–34, 1998.

[46] Stuart L Gillan. Recent developments in corporate governance: An overview, 2006.

[47] Boris Gloger. Scrum. *Informatik-Spektrum*, 33(2):195–200, 2010.

[48] Danny Greefhorst and Erik Proper. *Architecture Principles The Cornerstones of Enterprise Architecture*. 2011.

[49] Danny Greefhorst and H.A. Proper. A practical approach to the formulation and use of architecture principles. In *2011 IEEE 15th International Enterprise Distributed Object Computing Conference Workshops*, pages 330–339. IEEE, 8 2011.

[50] Michael Hafner and Ruth Breu. *Security engineering for service-oriented architectures*. Springer Science & Business Media, 2008.

[51] Mohammad Kazem Haki and Christine Legner. Enterprise architecture principles in research and practice: Insights from an exploratory analysis. In *ECIS*, page 204, 2013.

[52] Inge Hanschke. *Enterprise Architecture Management - simple and effective*. Carl Hanser Publishing House, 2011.

[53] Inge Hanschke. Eam–einfach und effektiv. *Wirtschaftsinformatik & Management*, 4(4):72–76, 2012.

[54] Sebastian Hanschke, Jan Ernsting, and Herbert Kuchen. Integrating agile software development and enterprise architecture management. In *System Sciences (HICSS), 2015 48th Hawaii International Conference on*, pages 4099–4108. IEEE, 2015.

[55] Matheus Hauder, Sascha Roth, Christopher Schulz, and Florian Matthes. Current tool support for metrics in enterprise architecture management. In *Proceedings of the DASMA Software Metrik Kongress*, 2013.

[56] Matheus Hauder, Sascha Roth, Christopher Schulz, and Florian Matthes. Agile enterprise architecture management: an analysis on the application of agile principles. In *International Symposium on Business Modeling and Software Design BMSD*, 2014.

[57] Matheus Hauder, Christopher Schulz, Sascha Roth, and Florian Matthes. Organizational factors influencing enterprise architecture management challenges. In *21st European Conference on Information Systems (ECIS), Utrecht, Netherland*, 2013.

[58] MA Hensema. Applying agile in enterprise architecture. Master's thesis, University of Twente, 2015.

[59] R Hoda, N Salleh, and J Grundy. The rise and evolution of agile software development. *IEEE Software*, page 1, 2018.

[60] Jan Hoogervorst. Enterprise architecture: Enabling integration, agility and change. *International Journal of Cooperative Information Systems*, 13(03):213–233, 2004.

[61] Jan AP Hoogervorst. *Enterprise governance and enterprise engineering*. Springer Science & Business Media, 2009.

[62] Bettina Horlach, Tilo Böhmann, Ingrid Schirmer, and Paul Drews. It governance in scaling agile frameworks. In *Multikonferenz Wirtschaftsinformatik (MKWI) 2018*, pages 1789–1800, Lüneburg, Germany, 2018.

[63] IEEE. Iso/iec/ieee systems and software engineering – architecture description, ingenierie des systemes et des logiciels - description de l'architecture. *ISO/IEC/IEEE 42010:2011(E) (Revision of ISO/IEC 42010:2007 and IEEE Std 1471-2000)*, pages 1–46, Dec 2011.

[64] INNOQ. Independent systems architecture - principles, 2018. accessed: 2018-11-24.

[65] IT Governance Insitute. It-governance für geschäftsführer und vorstände–zweite ausgabe 2003.

[66] ISACA. *COBIT 5: A business framework for the governance and management of enterprise IT*. Isaca, 2012.

[67] ISO, IEC, IEEE, International Standard, ISO/IEC/IEEE, International Organization Of Standardization, ISO/IEC/IEEE 42010, ISO/IEC/IEEE, IEEE, and International Organization Of Standardization. Iso/iec/ieee 42010:2011 - systems and software engineering – architecture description. *ISOIECIEEE 420102011E Revision of ISOIEC 420102007 and IEEE Std 14712000*, 2011.

[68] I ITGI. Board briefing on it governance. *Information Technology Governance Institute. Disponível em http://www. itgi. org*, 2003.

[69] I ITGI. Board briefing on it governance. *Information Technology Governance Institute. Disponível em http://www. itgi. org*, 2008.

[70] Michael C Jensen. The modern industrial revolution, exit, and the failure of internal control systems. *the Journal of Finance*, 48(3):831–880, 1993.

[71] Alexandre J.H.de O.Luna, Philippe Kruchten, Marcello L.G. do E.Pedrosa, Humberto R.de Almeida Neto, and Hermano P.de Moura Moura. State of the Art of Agile Governance: A Systematic Review. *International Journal of Computer Science and Information Technology*, 2014.

[72] Wolfgang Johannsen and Matthias Goeken. *Referenzmodelle für IT-Governance: Methodische Unterstützung der Unternehmens-IT mit COBIT, ITIL & Co*. dpunkt. verlag, 2011.

[73] Robert Johansen. *Leaders make the future: Ten new leadership skills for an uncertain world*. Berrett-Koehler Publishers, 2012.

[74] Nicolai M Josuttis. *SOA in practice: the art of distributed system design*. " O'Reilly Media, Inc.", 2007.

[75] Leon Kappelman. *The SIM guide to enterprise architecture*. CRC Press, 2009.

[76] P Aleatrati Khosroshahi, M Hauder, AW Schneider, and F Matthes. Enterprise architecture management pattern catalog v2. *Techncail Report, Technical University of Munich. Germany*, 2015.

[77] Thomas Kude, Miroslav Lazic, Armin Heinzl, and Alexander Neff. Achieving it-based synergies through regulation-oriented and consensus-oriented it governance capabilities. *Information Systems Journal*, 28:765–795, 2018.

[78] S. Kurpjuweit and R. Winter. Concern-oriented business architecture engineering. In *Proceedings of the 2009 ACM symposium on Applied Computing*, pages 265–272. ACM, 2009.

[79] S.n Kurpjuweit and R. Winter. Viewpoint-based meta model engineering. In *EMISA*, volume 2007, page 143, 2007.

[80] Matthias Lange and Jan Mendling. An experts' perspective on enterprise architecture goals, framework adoption and benefit assessment. In *Enterprise Distributed Object Computing Conference Workshops (EDOCW), 2011 15th IEEE International*, pages 304–313. IEEE, 2011.

[81] Marc Lankhorst. *Enterprise architecture at work: Modelling, communication and analysis*. Springer, 2009.

[82] Marc Lankhorst and P. Johnson. *Proceedings of the Second Workshop on Trends in Enterprise Architecture Research (TEAR 2007). St. Gallen. 2007*. 2007.

[83] AurÃ©lie Leclercq-Vandelannoitte and Bertin Emmanuel. From sovereign it governance to liberal it governmentality? a foucauldian analogy. *European Journal of Information Systems*, 27(3):326–346, 2018.

[84] Timothy C Lethbridge, Susan Elliott Sim, and Janice Singer. Studying software engineers: Data collection techniques for software field studies. *Empirical software engineering*, 10(3):311–341, 2005.

[85] Asa Lindstrom. On the syntax and semantics of architectural principles. In *System Sciences, 2006. HICSS'06. Proceedings of the 39th Annual Hawaii International Conference on*, volume 8, pages 178b–178b. IEEE, 2006.

[86] Mikael Lindvall, Dirk Muthig, Aldo Dagnino, Christina Wallin, Michael Stupperich, David Kiefer, John May, and T Kahkonen. Agile software development in large organizations. *Computer*, 37:26– 34, 01 2005.

[87] Jan Löhe and Christine Legner. Overcoming implementation challenges in enterprise architecture management: a design theory for architecture-driven it management (adrima). *Information Systems and e-Business Management*, 12(1):101–137, 2014.

[88] Carsten Lucke, Marco Bürger, Thomas Diefenbach, Jan Freter, and Ulrike Lechner. Categories of enterprise architecting issues-an empirical investigation based on expert interviews. *DC Mattfeld & S. Robra-Bissantz(Eds.), Multikonferenz Wirtschaftsinformatik*, pages 999–1010, 2012.

[89] Carsten Lucke, Sascha Krell, and Ulrike Lechner. Critical issues in enterprise architecting–a literature review. 2010.

[90] Jerry Luftman and Tal Ben-Zvi. Key issues for it executives 2011: Cautious optimism in uncertain economic times. *MIS Quarterly Executive*, 10(4), 2011.

[91] A. J. H. de O. Luna, Philippe Kruchten, E. L. Riccio, and H. P. de Moura. Foundations for an agile governance manifesto: a bridge for business agility. *13th International Conference on Management of Technology and Information Systems*, 2016.

[92] Alexandre J H De Oliveria Luna, Cleyverson P Costa, and CÃ©sar A D C Nascimento. Agile governance in information and communication technologies : Shifting paradigms. *Journal of Information Systems a Technology Management*, 2010.

[93] P Makosz and U Gelinas. Control objectives for information and related technology cobit. *IS AUDIT AND CONTROL JOURNAL*, pages 12–13, 1996.

[94] Ruth Malan and Dana Bredemeyer. Less is more with minimalist architecture. *IT professional*, 4(5):48–47, 2002.

[95] Thomas Mannmeusel. Management von unternehmensarchitekturen in der praxis: Organisatorische herausforderungen in mittelständischen unternehmen. In *Analyse und Gestaltung leistungsfähiger IS-Architekturen*, pages 35–57. Springer, 2012.

[96] Eric Marks. Governing enterprise agile development without slowing it down: Achieving friction-free scaled agile governance via event- driven governance. Technical report, AgilePath Corporation, 2014.

[97] Michelle A. Marks, John E. Mathieu, and Stephen J. Zaccaro. Multi-team systems. *International Handbook of Work and Organizational Psychology*, 2:289–313, 2001.

[98] Ben Martin. Agile enterprise architecture, 2013. accessed: 03/09/2018.

[99] Marina Martynova and Luc Renneboog. Evidence on the international evolution and convergence of corporate governance regulations. *Journal of Corporate Finance*, 17(5):1531–1557, 2011.

[100] Florian Matthes, Ivan Monahov, Alexander W Schneider, and Christopher Schulz. Towards a unified and configurable structure for ea management kpis. In *Trends in Enterprise Architecture Research and Practice-Driven Research on Enterprise Transformation*, pages 284–299. Springer, 2012.

[101] Florian Matthes and Sascha Roth. *Enterprise architecture visualization tool survey 2014*. Technical University of Munich, 2014.

[102] Robert R Moeller. *Executive's guide to IT governance: improving systems processes with service management, COBIT, and ITIL*. John Wiley & Sons, 2013.

[103] Mariana Mykhashchuk, Sabine Buckl, Thomas Dierl, and Christian M Schweda. Charting the landscape of enterprise architecture management. In *Wirtschaftsinformatik*, page 83, 2011.

[104] E Newcomer and G Lomow. Understanding soa with web services (independent technology guides) 2004.

[105] Dave Nicholette. Enterprise architecture and agile. musings of a software development manager, 2007. accessed: 30/08/2018.

[106] Klaus D Niemann. Enterprise architecture management and its role in it governance and it investment planning. In *Information Resources Management: Concepts, Methodologies, Tools and Applications*, pages 996–1026. IGI Global, 2010.

[107] Klaus D. Niemann. Unternehmensarchitektur und digitalisierung. *HMD Praxis der Wirtschaftsinformatik*, 55(5):907–927, Oct 2018.

[108] Robert L Nord, Ipek Ozkaya, and Philippe Kruchten. Agile in distress: architecture to the rescue. In *International Conference on Agile Software Development*, pages 43–57. Springer, 2014.

[109] Architecture Working Group of the Software Engineering Committee et al. Recommended practice for architectural description of software intensive systems. *IEEE Standards Department*, 2000.

[110] Taiichi Ohno. *Toyota production system: beyond large-scale production*. crc Press, 1988.

[111] Martin Op't Land and Erik Proper. Impact of principles on enterprise engineering. In *ECIS*, pages 1965–1976, 2007.

[112] Martin Op't Land, Erik Proper, Maarten Waage, Jeroen Cloo, and Claudia Steghuis. *Enterprise architecture: creating value by informed governance*. Springer Science & Business Media, 2008.

[113] Carla Marques Pereira and Pedro Sousa. Enterprise architecture: business and it alignment. In *Proceedings of the 2005 ACM symposium on Applied computing*, pages 1344–1345. ACM, 2005.

[114] Erik Proper, Marc Lankhorst, Marten Schönherr, Joseph Barjis, and Sietse Overbeek. *Trends in Enterprise Architecture Research: 5th Workshop, TEAR 2010, Delft, The Netherlands, November 12, 2010, Proceedings*, volume 70. Springer Science & Business Media, 2010.

[115] A. Qumer and B. Henderson-Sellers. A framework to support the evaluation, adoption and improvement of agile methods in practice. *Journal of Systems and Software*, 2008.

[116] Richard A Reid, Elsa L Koljonen, and J Bruce Buell. The deming cycle provides a framework for managing environmentally responsible process improvements. *Quality Engineering*, 12(2):199–209, 1999.

[117] Bob Rhubart. Agile enterprise architecture, 2010.

[118] Gary L Richardson, Brad M Jackson, and Gary W Dickson. A principles-based enterprise architecture: Lessons from texaco and star enterprise. *MIS quarterly*, pages 385–403, 1990.

[119] Jeanne W Ross, Peter Weill, and David Robertson. *Enterprise architecture as strategy: Creating a foundation for business execution.* Harvard Business Press, 2006.

[120] Sascha Roth, Florian Matthes, and Marin Zec. *Enterprise architecture visualization tool survey 2014.* Technical Report, Technische Universität MÃ¼nchen, 2014.

[121] Per Runeson and Martin Höst. Guidelines for conducting and reporting case study research in software engineering. *Empirical software engineering*, 14(2):131, 2009.

[122] Per Runeson and Martin Höst. Guidelines for conducting and reporting case study research in software engineering. *Empirical software engineering*, 14(2):131, 2009.

[123] Jaap Schekkerman. *Enterprise architecture good practices guide: how to manage the enterprise architecture practice.* Trafford Pub., 2008.

[124] Marten Schöenherr. Towards a common terminology in the discipline of enterprise architecture. In *International Conference on Service-Oriented Computing*, pages 400–413. Springer, 2008.

[125] K Schwaber and M Beedle. Agile software development with scrum prentice hall ptr upper saddle river. *NJ, USA*, 2001.

[126] Ganesan Senthilvel, Ovais Mehboob Ahmed Khan, and Habib Ahmed Qureshi. *Enterprise Application Architecture with. NET Core.* Packt Publishing Ltd, 2017.

[127] Robert W Shirey. Security architecture for internet protocols. a guide for protocol designs and standards. *Internet Draft*, 1994.

[128] Andrei Shleifer and Robert W Vishny. A survey of corporate governance. *The journal of finance*, 52(2):737–783, 1997.

[129] Robert E Stake. *The art of case study research.* Sage, 1995.

[130] G. Steinacker. Architekturprinzipien, 2013. accessed: 2018-11-24.

[131] Dirk Stelzer. Enterprise architecture principles: literature review and research directions. In *Service-oriented computing. ICSOC/ServiceWave 2009 workshops*, pages 12–21. Springer, 2010.

[132] Don Tapscott and Art Caston. *Paradigm Shift: The New Promise of Information Technology.* ERIC, 1993.

[133] RI Bob Tricker and Robert Ian Tricker. *Corporate governance: Principles, policies, and practices.* Oxford University Press, USA, 2015.

[134] Ömer Uludağ. Scaling agile practices workshops. `https://wwwmatthes.in.tum.de/pages/1lihu1sjq8jpk/Scaling-Agile-Practices-Workshops`. Accessed: 2018-07-01.

[135] Ömer Uludağ, Matheus Hauder, Martin Kleehaus, Christina Schimpfle, and Florian Matthes. Supporting large-scale agile development with domain-driven design. In *International Conference on Agile Software Development*, pages 232–247. Springer, 2018.

[136] Ömer Uludağ, Martin Kleehaus, Xian Xu, and Florian Matthes. Investigating the role of architects in scaling agile frameworks. In *Proceedings - 2017 IEEE 21st International Enterprise Distributed Object Computing Conference, EDOC 2017*, 2017.

[137] Patrick van Bommel, Pieter M Buitenhuis, Stijn J.B.A. Proper, and Erik H.A. Hoppenbrouwers. Architecture principles - a regulative perspective on enterprise architecture. In *Enterprise modelling and information systems architectures : concepts and applications ; proceedings of the 2nd International Workshop on Enterprise Modelling and Information Systems Architectures, St. Goar, Germany, October 8 - 9*, 2007.

[138] Wim Van Grembergen, Steven De Haes, and Erik Guldentops. Structures, processes and relational mechanisms for it governance. In *Strategies for information technology governance*, pages 1–36. Igi Global, 2004.

[139] Jack Van't Wout, Maarten Waage, Herman Hartman, Max Stahlecker, and Aaldert Hofman. *The integrated architecture framework explained: why, what, how*. Springer Science & Business Media, 2010.

[140] TOGAF Version. 9.1, an open group standard. *Published in the US by The Open Group*, 2011.

[141] Roel Wagter, Martin Van Den Berg, Joost Luijpers, and Marlies Van Steenbergen. *Dynamic enterprise architecture: how to make it work*. John Wiley & Sons, 2005.

[142] Alain Wegmann. The systemic enterprise architecture methodology (seam). business and it alignment for competitiveness. Technical report, 2002.

[143] Peter Weill and Jeanne W Ross. *IT governance: How top performers manage IT decision rights for superior results*. Harvard Business Press, 2004.

[144] A. Wiggins. Architekturprinzipien, 2017. accessed: 2018-11-24.

[145] Robert Winter. Architectural thinking. *Wirtschaftsinformatik*, 56(6):395–398, 2014.

[146] Robert Winter and Stephan Aier. How are enterprise architecture design principles used? In *Enterprise Distributed Object Computing Conference Workshops (EDOCW), 2011 15th IEEE International*, pages 314–321. IEEE, 2011.

[147] Robert Winter and Ronny Fischer. Essential layers, artifacts, and dependencies of enterprise architecture. In *Enterprise Distributed Object Computing Conference Workshops, 2006. EDOCW'06. 10th IEEE International*, pages 30–30. IEEE, 2006.

[148] Robert Winter and Joachim Schelp. Enterprise architecture governance: the need for a business-to-it approach. In *Proceedings of the 2008 ACM symposium on Applied computing*, pages 548–552. ACM, 2008.

[149] André Wittenburg. *Softwarekartographie: Modelle und methoden zur systematischen visualisierung von anwendungslandschaften*. PhD thesis, Technische Universität München, 2007.

[150] Robert K. Yin. *Case Study Research: Design and Methods (Applied Social Research Methods)*. Sage Publications, fourth edition. edition, 2008.

[151] John A Zachman. A framework for information systems architecture. *IBM systems journal*, 26(3):276–292, 1987.