Designing the Enterprise Architecture Function

Bas van der Raadt[1] and Hans van Vliet[2]

[1] Capgemini, Global Financial Services / Architecture & Governance Improvement,
Papendorpseweg 100, 3528 BJ Utrecht, The Netherlands
bas.vander.raadt@capgemini.com
[2] VU University, Department of Computer Science
De Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands
hans@cs.vu.nl

Abstract. Enterprise Architecture (EA) is becoming an increasingly mature field of work, but many large organizations still struggle with implementing an integral and truly effective EA function. The literature provides a fragmented picture of the EA function, describing the various separate elements that make up the total package of activities, resources, skills, and competences of the EA delivery function. In our view, the EA function reaches beyond EA delivery and also includes the stakeholders, structures and processes involved with EA decision making and EA conformance. A holistic and integral view on the EA function is essential in order to properly assess an EA function on its performance, and to allow identifying the key points of improvement. In this article, we give such a description of the EA function, which provides the reference model in EA function performance assessments as part of our Normalized Architecture Organization Maturity Index (NAOMI) approach.

Keywords: Enterprise Architecture, Management, Organizational, Function, Reference Model, Governance, Conformance.

1 Introduction

The dream of every CEO is to have one standardized, integrated, flexible and manageable landscape of aligned business and IT processes, systems and procedures. Having complete control over all projects implementing changes in that landscape so that they deliver solutions that perfectly fit the corporate and IT change strategies, makes this dream complete. The reality for many large organizations is quite the opposite. Many large organizations struggle to keep their operational and change costs in control. Key reasons are the inflexibility and enormous complexity of their business and IT structures, processes, systems, and procedures, often distributed across lines of business (LoB) and business divisions (BD) spread out over various regions, countries or even continents [1], [2]. Over the last decade, Enterprise Architecture (EA) has been one of many instruments used by organizations in their attempt to get grip on the current operational environment and the implementation of changes. EA provides standardization, and sets a clear direction for the future to guide changes. Compared with architecture in the physical world, EA provides the

S. Becker, F. Plasil, and R. Reussner (Eds.): QoSA 2008, LNCS 5281, pp. 103–118, 2008.

mechanism for city planning where software architecture is the architecture of one building. EA thus gives boundaries within which a software architect has to operate. Effectively applying EA leads to a reduction in operational maintenance costs due to increased discipline and control, as well as increased responsiveness because EA leads to reduced project duration. Additionally, EA improves risk management as it leads to reduced complexity, and it increases management satisfaction, because it provides an enterprise-wide view on organizational changes [3]. Finally, EA enhances strategic business outcomes because it helps increasing the effectiveness of business processes, applications, data and infrastructure through standardization [4].

Although seen as a vital management instrument by many large organizations [2], EA has generally not reached the desired result. EA has been practiced for at least ten years now, but it still suffers from relative immaturity, as we have experienced in various assessments of EA functions at client organizations (e.g., see [5] and Section 3). They have difficulties establishing an EA function that is fully integrated into the existing corporate or IT governance, as well as stimulating effective collaboration between architects and other stakeholders. Such a fragmented and badly integrated EA function typically fails to fulfill the expectations of all EA stakeholders which leads to the goals set with EA – if explicitly set at all – not being achieved.

In EA research, much effort has been put into various separate elements that make up the total package of activities, resources, skills, and competences that a mature EA delivery function should have in place – e.g., proper tools [6], frameworks [7]; [8] and architects [9]. In order to measure an organization's EA capability maturity, various EA function performance assessments (e.g., [5]; [10]; [11]) have been developed. These primarily focus on assessing whether the elements that determine the maturity of the EA delivery function (e.g. an architecture department or team) are in place. However, based on our experience, this is a limited view. The activities of the EA function should reach beyond merely delivering EA products and should also include other organizational roles, bodies, and activities responsible for EA decision making (e.g. an architecture council) and EA conformance (e.g. project managers and designers). An organization will only be effective with Enterprise Architecture when there is effective formal and informal interplay between the members of the EA delivery function and the stakeholders responsible for EA decision making and EA conformance. Moreover, the entire EA function must be properly integrated into the overall organizational and governance structures in order to be effective.

Currently, the literature lacks a complete reference model of an EA function. Existing EA capability maturity assessment approaches (e.g., [10]; [11]) have incorporated a reference model into their maturity model, but this model is often limited to the EA delivery function. Other practitioner's literature (e.g., [7]; [11]) provides a fragmented view of elements of the EA function. In this paper we provide a clear definition and integral description of the EA function, established into our EA function reference model. This model describes the norm we compare client organizations to, while assessing their EA function's performance. Both the EA function reference model and assessment model are part of our Normalized Architecture Organization Maturity Index (NAOMI) approach [5].

Section 2 of this paper contains our reference model of the EA function. Section 3 contains a case study that shows how one company has implemented its EA function.

In Section 4 we discuss the lessons learned regarding our EA function reference model based on this case study. Finally, in Section 5 we draw our conclusions and discuss future research we will conduct on the topic of the Enterprise Architecture function.

2 Reference Model

Based on scientific and practitioner's literature, and various case studies at a Global Financial Services Companies (e.g. [5]), we have created an integral description of the EA function. We define the EA function as: *The organizational functions, roles and bodies involved with creating, maintaining, ratifying, enforcing, and observing Enterprise Architecture decision-making – established in the enterprise architecture and EA policy – interacting through formal (governance) and informal (collaboration) processes at enterprise, domain, project, and operational levels.*

Based on this definition, we describe our EA function reference model with Section 2.1 describing its *structure*, Section 2.2 its *products*, Section 2.4 its process model, and Section 2.5 the *bodies and roles* involved. Section 2.3 provides a detailed description of *EA delivery* as part of the entire EA function.

2.1 Structure of the EA Function

Figure 1 shows the three main responsibilities of the EA function: (1) EA decision making, (2) EA delivery, and (3) EA conformance (see Fig. 1). *EA decision making* at strategic and tactical level is responsible for approving new EA products or changes in existing EA products, and for handling escalations regarding EA conformance. This is typically performed by one or more governance bodies (e.g. an EA council). Having such governance bodies in place – with proper representation from various stakeholder groups (see Section 2.5.1) – results in better perceived importance, involvement and support of both management and other stakeholders, and it improves effectiveness of the EA function [1]. EA governance bodies vary in the degree to which they have an advisory or formal decision making authority [12].

EA delivery is responsible for providing advice to guide EA decision making at strategic and tactical level. Additionally, EA delivery creates and maintains EA products, validates change results to see whether they conform to the EA, as well as provides support in applying EA products (see Section 2.3).

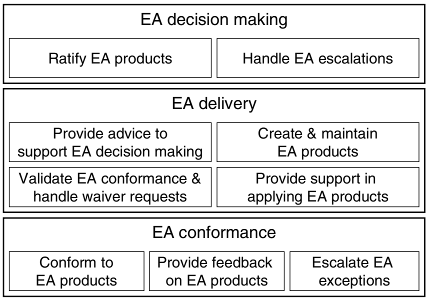
EA decision making
Ratify EA products
Handle EA escalations
EA delivery
Provide advice to support EA decision making
Create & maintain EA products
Validate EA conformance & handle waiver requests
Provide support in applying EA products
EA conformance
Conform to EA products
Provide feedback on EA products
Escalate EA exceptions

Fig. 1. Responsibilities of the three main Enterprise Architecture functions

Finally, *EA conformance* is responsible for implementing organizational changes through solutions as described in the target architectures, complying with the EA policy, and providing feedback on the applicability of the EA products to the EA delivery function. EA conformance is typically the responsibility of members of the organization who are affected by the EA products [13] while running change projects (e.g. project managers) or implementing operational changes (e.g. operational maintenance) at tactical and operational level.

2.2 Products of the EA Function

There are generally two types of EA products: (1) architectures and (2) EA policies [14]. An *architecture* document provides an abstraction of *what* a complex environment looks like, and acts as a means of communication and decision making regarding that environment [2]. Three types of architecture documents exist: (1) target state (to-be, soll) architecture that provides an abstraction of the desired situation, (2) current state (as-is, ist) architecture that describes the current operational environment, and (3) roadmap that describes a realization path from the current state to the target situation. These types of architecture documents aim at one or more of four aspect areas: (1) business architecture, (2) information architecture, (3) information systems, and (4) technical infrastructure [8]. The first two dimensions represent the business aspects of an organization; the latter two represent the IT aspects. In our view, Enterprise Architecture comprises both the business and IT aspects of an organization, and the alignment between them [2].

EA policy prescribes *how* projects should implement organizational changes across various LoBs and BDs through unified principles and practices. EA policies may be specified in three possible forms: (1) standards, (2) rules, or (3) guidelines. Both a standard and a rule must be adhered to; a guideline may be deviated from, provided a waiver has been granted. Enforcing EA policy enables organizations to influence the change activities of subunits without dictating exactly how they handle all of their operational activities [1]. Keeping up-to-date with industry standards allows organizations to change in a predictable way as a response to external developments [4], such as market changes, technological innovations and regulatory changes.

2.3 EA Delivery Function

The EA delivery function is often organized as a separate department [15] or team [1], typically as an organizational staff function. Depending on the size of the organization, the EA function may also consist of one or more individually operating architects. The EA department or team is sometimes led by a chief architect [1]. The origin of the EA function may differ, resulting in a difference in focus on either business aspects or IT aspects [2]. Regardless of the focus, there are generally four types of responsibilities of EA delivery:

1) Provide advice to support EA decision making regarding the target architecture by:

- Helping in building a vision and strategy for the future, based on its relation with its external environment regarding social, environmental and market developments, technological innovations, regulatory changes, etc.

- Describing decision alternatives regarding the target situation [16], and performing an impact analysis on predefined evaluation criteria and indicators (e.g. financial, regulatory) to determine the consequences of those alternatives in order for management to select the most desirable one [17]

2) Create and Maintain EA products that describe the:

- Current state architecture, which provides insight in the as-is situation of the operational environment, together with its bottlenecks and accompanying risks
- Concrete target state architecture, based on the vision and strategy, describing the chosen decision alternative in detail, which is assessed on its ability to cope with possible internal and external changes using various future scenarios
- Roadmap from the current state to the target situation, in which the mutual relation and impact of the elements in the architecture is described, and the sequence of implementation steps is given
- EA policies based on up-to-date knowledge of industry standards and developments within the organization, and determine their potential impact [4]

3) Validate EA conformance by:

- Reviewing programs or projects on their compliance with the applicable:
  - Target architectures at enterprise and domain levels, to ensure that individual program and project results contribute to achieving the general business goals and the target situation described in those target architectures
  - EA policies, to ensure that change activities of programs or projects contribute to achieving the standardization and integration goals set with EA
  - Current state architectures, ensuring the operational readiness of the program and project results before deployment, thus safeguarding the continuity of the operational processes and systems
- Handling waiver requests, assessing the implications of allowing programs and projects that file the requests to deviate from a specific guideline

4) Provide support in applying EA products towards programs and projects (e.g. through training and coaching) in:

- Creating program and project target architectures based on the EA products at domain and enterprise levels
- Conforming to the EA products in running programs and projects

2.4 EA Process Model

Pulkkinen [18] describes an EA process model for the management of architectural decisions in enterprise architecture planning that has three abstraction layers: (1) enterprise level, (2) domain level, and (3) systems level. Decisions made at higher management levels are made explicit in EA products that flow downwards to lower levels, introducing more detail. The architectures and EA policies at a higher level set the boundaries for decision making and implementation at lower levels. From our practical experience with implementing EA functions, this has proven to be an appropriate model. However, based on our practical experience and an exploratory study on the stakeholder's perception of EA performance [19], we altered and extended the EA process model.

The EA process model makes a distinction between *permanent* (e.g., business process chains, BDs, or LoBs) and *non-permanent* (e.g., large programs) domains. However, it is also vital to make a distinction between specific and generic business domains because of their conflicting operating models as a result of different optimization principles. A *specific* domain typically entails a customer facing LoB, which provides a specific product or service, servicing a specific market or client segment, or operating within a defined geographical region. It therefore optimizes its operating model in order to fine tune its services to the needs of its customers [20]. On the other hand, a *generic* business domain (e.g., a shared service center) typically offers generic or infrastructural services to various LoBs and BDs within an organization – thus acting as a cost center – optimizing its structures, processes, systems and procedures so it can minimize its operational costs [21]. In order to best deal with the horizontal integration of specific and generic domains, EA decision making may be centralized, decentralized or implemented in a federal model, depending on the organizational characteristics [12].

We changed the name of 'systems level' into '*project level*'. This leaves open what type of solutions projects deliver. The term 'systems level' suggests that EA decision making and implementation always results into an IT solution [18]. However, within the business and information architecture aspect areas [8], projects may deliver case handling processes that require human involvement and physical information flows (through paper forms); it is not always possible to fully automate business processes into Straight Trough Processing (STP) [22].

Also, we added an *operational level* to the process model, because of the conflict in decision making regarding organizational changes at project level, and organizational stability and continuity at operational level [23]. Decision making about exploiting a continuous and repeating operational environment aims at refinement, through predictable small impact changes, to maximize its continuity and stability. Decision making at project level often is different in nature, because it concerns realizing less predictable high impact changes in the operational situation, potentially compromising the continuity and stability at operational level.

Enterprise-wide decision making – as is the case with EA – should encompass feedback from group and individual levels to ensure continuous improvement [24]. However, in practice such a feedback process is hardly performed. EA decision makers (e.g. senior management) feel that a one yearly decision making cycle is adequate in managing changes [25]. The EA process model incorporates a learning cycle with a downstream flow of decisions (*feed-forward*), and an upstream flow to feed the successes and constraints of implementing those decisions at lower levels back to higher levels (*feedback*) [18]. We elaborate on these concepts using the organizational learning framework of Crossan et al. [24]. We translated the four underpinning key premises of their framework to the situation of EA to enhance the EA process model (see Table 1).

In parallel with the organizational learning theory, the Enterprise architecture practice experiences a tension between *exploration* of new possibilities and *exploitation* of old certainties [26]. EA exploration takes place during decision making at enterprise and domain levels, and results in new architectures and EA

Table 1. The Organizational Learning premises taken from Crossan et al. [24] specified to the Enterprise Architecture construct space.

Premise	Organizational Learning (OL)	Enterprise Architecture (EA)
1	OL involves a tension between assimilating new learning (exploration) and using what has been learned (exploitation).	EA involves a tension between creating new EA products through exploration and exploiting the existing EA products that describe the operational structures, systems and processes, and prescribe the current standards and procedures.
2	OL is multi-level: individual, group, and organization.	EA is multi-level: enterprise, domain, project, and operational.
3	The three levels of OL are linked by social and psychological processes: intuiting, interpreting, integrating, and institutionalizing.	The four levels of EA are linked by formal (governance) and informal (collaboration) processes.
4	Cognition affects action, and vice versa.	Theory (architectures and standards) affects practice (change projects and operational structures, processes and systems), and vice versa.

policies being created and approved. Following, these EA products, describing how changes should be implemented, are fed forward to project and operational level, where these are to be interpreted and followed. The EA delivery function plays a vital role, as mediator between EA decision making and EA conformance, in getting this shared understanding and common behavior. It requires a more pro-active attitude than merely writing down the central decisions and publishing them so that they are available to lower levels. Having an integrated and effective roll out and acceptance plan is vital for the EA delivery function to realize this organizational change [27].

Feedback is vital in respecting the constraints and problems that arise at project or operational level with applying the EA products prescribed. These may not have been anticipated during EA decision making at domain or enterprise level. *Informal feedback* during the collaboration between architects and EA stakeholders at lower levels allows continuous improvement of EA products through refinement of EA decisions at higher levels. This ensures their practical applicability and prevents them from being exclusively used by architects [2]. *Formal feedback* through escalation of EA conformance exceptions and waiver requests also provides vital information for improving the EA products; the number of escalations and waiver requests regarding a specific EA products acts as a quality indicator of that product.

Also, feedback allows incorporating what has been learned at lower levels, through exploration, experimentation and innovation, into EA products prescribed at higher levels. *Best practices* (e.g., a proof-of-concept of a new, innovative technology) at project or operational level are identified and evaluated on their generic applicability [18]. This leads to a *proposal* for changes in existing, or the creation of new, EA products. When ratified, an EA product receives the formal status ‘approved’, and will go through the validity statuses: future, actual, confined and obsolete, before receiving the formal status ‘retired’, introducing an *EA product life cycle*.

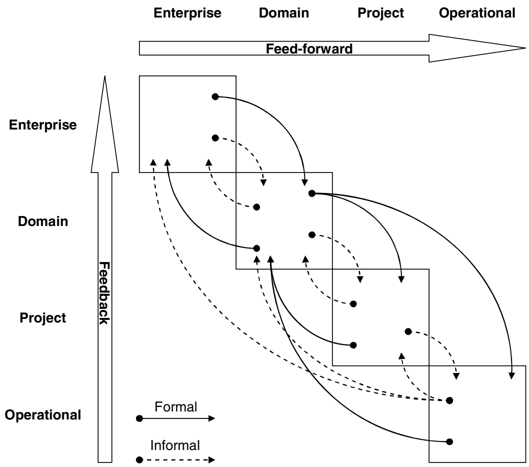

Fig. 2. EA Process Model including a learning cycle of feed-forward and feedback across enterprise, domain, project and operational levels

Figure 2 shows the EA learning cycle constructed of formal and informal EA processes at various organization levels. Table 2 describes the in and output regarding the feed-forward and feedback of these formal and informal processes.

2.5 Bodies and Roles within the EA Function

Within the process model of the EA function described in Section 2.4, various bodies and roles interact while pursuing different objectives and goals.

2.5.1 Bodies and Roles within EA Decision Making

The EA governance bodies within the EA function are responsible for decision making about EA products, giving them a formal status. Also, it handles escalations of non-conformity. An effective EA governance body at any organizational level should: (1) be composed of the various roles that represent the potentially conflicting interests that occur at that organizational level, (2) perform transparent decision making based on objective criteria, and (3) have the proper mandate to enforce the decisions at that organizational level.

The *EA council* at enterprise level acts as a steering committee [1] in order to achieve horizontal integration for coordinating of EA decision making [12]. It is comprised of representatives of the domains within the organization, the chief architect, and a chairman. Both the chairman – a key EA sponsor – and the chief architect – responsible for the quality and effectiveness of the overall Enterprise Architecture – should act in the interest of the enterprise-wide structures, processes, systems and procedures to achieve the corporate strategy. The domain owners are concerned with optimizing their specific domains to achieve their domain specific strategies. When issues cannot be resolved within the EA council, they are escalated towards senior management sponsoring the EA council for final decision making.

Table 2. EA functions and activities performed at enterprise, domain, project and operational levels; at each level input and output is fed forward or back, creating the EA learning cycle

Organizational level	**Functions & activities**	**Formal processes**	**Informal processes**
Enterprise	EA decision making EA delivery	Feed-forward: • (Out) Validate domain level EA conformance	Feed-forward: • (Out) Provide support in applying EA products at domain level
		Feedback: • (In) Handle domain level EA escalations and waiver requests	Feedback: • (In) Use feedback to maintain enterprise level EA products • (In) Use operational expert knowledge and data in EA decision making
Domain	EA decision making EA delivery EA conformance	Feed-forward: • (In) Conform to enterprise level EA products • (Out) Validate project and operational level EA conformance	Feed-forward: • (In) Utilize support in applying EA products • (Out) Provide support in applying EA products at project and operational level
		Feedback: • (In) Handle project and operational level EA escalations and waiver requests • (Out) Escalate domain level EA exceptions, and file waiver requests towards enterprise level	Feedback: • (In) Use feedback to maintain domain level EA products • (In) Use operational expert knowledge and data in EA decision making • (Out) Provide feedback on existing or potentially new EA products
Project	EA conformance	Feed-forward: • (In) Conform to domain level EA products	Feed-forward: • (In) Utilize support in applying EA products • (Out) Provide support in deploying the project result
		Feedback: • (Out) Escalate project level EA exceptions, and file waiver requests	Feedback: • (In) Use operational expert knowledge to run project • (Out) Provide feedback on existing or potentially new EA products
Operational	EA conformance	Feed-forward: • (In) Conform to domain level EA products	Feed-forward: • (In) Utilize support in deploying the project result
		Feedback: • (Out) Escalate operational level EA exceptions, and file waiver requests	Feedback: • (Out) Provide operational expert knowledge and data

At domain level, there may be a formal authority or informal advisory EA governance body (e.g., *domain architecture council*), which is responsible for EA decision making within that domain. Membership is similar to the EA council, only the roles stay within the domain. The domain architecture council handles the ratification of domain specific EA products and handles the escalations regarding non-conformity with those EA products. Only when disputes regarding non-conformity cannot be resolved within the domain architecture council, or the impact of decisions made by the domain architecture council reaches beyond that domain, is that issue escalated towards the EA council at enterprise level. This reduces the workload for the EA council to only the hard-to-resolve, domain overarching issues.

At project level there is typically no formal EA governance body. The *project steering committee* may act as an informal EA governance body. Issues of non-conformity that cannot be resolved and may lead to a project deviating from the enforced EA products will be escalated towards the domain architecture council.

2.5.2 Roles within EA Delivery

At enterprise level, the EA delivery function usually consists of a central EA team [1] or staff department [11], comprised of an EA manager, the chief enterprise architect, and various enterprise architect roles. Each *enterprise architect* is responsible for a specific EA aspect area (i.e. business, information, information system, or technical infrastructure [8]), performing the primary activities of the EA delivery function (see Section 2.1) at enterprise level. The *chief enterprise architect* [1] typically acts as the functional lead of the EA delivery function, overseeing all aspect areas of the enterprise architecture. He or she acts as trusted advisor to the CxO, and is responsible for the quality and effectiveness of the overall Enterprise Architecture. The *EA manager* runs the EA delivery function, performing budget and resource management, planning and coordination, and other operational management tasks.

Organizational domains (e.g., LoBs) typically employ their own specific architects at domain level, who are experts in a specific business or IT area. The *domain architect* acts as trusted advisors to the domain owner (e.g. head of the LoB). Depending on the size and structure of the domain level EA delivery function, autonomously operating architects, a team of architect-like roles, or a formal architecture department may be present. This domain level EA delivery function acts as a *sub-team* [1] of the central EA delivery function at enterprise level.

2.5.3 Roles within the EA Conformance

The members of a project team are responsible for managing and running change projects. These projects should deliver solutions that transform specific parts of the organization’s operational environment into the desired situation described in the target architecture(s) at enterprise and domain levels. Additionally, they should comply with EA policy while running the project. A special role is *project architect*, who acts as an advisor guarding the quality of the project. He or she provides advice in the start-up phase of a project to discuss the important implementation decisions, and is responsible for the delivery of a project design which complies with the enforced EA products. Also, the project architect should provide feedback on the practical applicability of EA products towards the domain level EA delivery function.

A project architect is not member of the EA delivery function. This role has project result responsibility, and can therefore not perform project validations independently.

At operational level, the EA delivery function performs a *gatekeeper* role, performing post implementation reviews of: (1) solutions projects deliver and (2), changes made in the operational environment. In performing these reviews, changes are assessed on operational readiness and EA conformance before being deployed.

3 EA Function at a Large International Company

We conducted a case study at a large international company, henceforth called company A, assessing its EA function against our EA function reference model. We held fully structured interviews with various roles – i.e., domain owners, EA council members, program and project managers, operational managers, architecture managers, architects, designers, subject matter experts – addressing the assessment topics which are part of our NAOMI approach [5]. Also, the assessors studied an extensive set of strategic, project, operational, and communication documents, in order to check the findings from the interviews. With these findings we created an image of the EA function, and compared these to the reference model described in Section 2.

We have done a second case study of the EA function within a comparable organization (company B) using the same approach. This case study had comparable results. For confidentiality reasons, no details hereof can be given. In the case description in this section, we indicate which findings we confirmed with the case study conducted at company B, and which findings were different.

The case study we conducted involved the assessment of an EA function within the operations and IT division of a large international company, with technical infrastructure as the primary focus area. This back-office division consists of various verticals providing operational and IS services to the various LoBs within the front-office, as well as a technology department providing infrastructural services to the verticals. The EA function, as part of the technology department, is responsible for creating enterprise wide infrastructure policies and validating solution designs on their conformance. The EA delivery function consists of a team of architects, each an expert regarding a specific infrastructural domain (e.g., storage, mainframe, internet, etc.), responsible for creating EA policy and performing conformance validations related to that domain. When a solution touches several infrastructural domains, it had to be validated by each domain architect responsible for those domains. The chief infrastructure architect and the infrastructure domain architects held a monthly meeting to approve new infrastructure policies. This was not a formal EA council with representatives from the verticals responsible for EA decision making regarding the infrastructural policies. The infrastructure policies did have impact for those verticals. This monthly meeting resulted in few policies getting a formal status. There was no standard procedure, for policies that received a formal status, to store and publish them in one central repository.

Company B did have a formal EA council with proper representation from the Business Divisions (BD) within the company. The EA council, however, was also unable to assess and approve EA policy proposals created by the EA delivery function, and provide them with a formal status.

Our assessment of the EA function in company A showed that there was no enterprise infrastructure architecture written down that describes the relations and coherence between the infrastructure domains. This resulted in inconsistent and incoherent EA policies across the infrastructure domains. The domain architects provided conflicting advice to the project managers and designers, because they collaborated insufficient with each other, and did not have a enterprise infrastructure architecture to guide them. This made creating a coherent solution design that complied with the EA policies complex for the designers, which frustrated them. Many designers also had little experience with creating solution designs according to the template provided by the EA function. Many solution designs sent to the EA function for validation were therefore of low quality, and were either found inadmissible or were rejected.

Company B did have an enterprise architecture that described the relations and coherence between domains. However, this enterprise architecture wasn't detailed enough to provide a concrete reference for the domain architects. This resulted in similar problems regarding conflicting architectures, policies, and advice by the EA delivery function we found at company A.

The conflicts of opinions and insufficient collaboration between domain architects at company A caused the validation outcome of solution design to be unpredictable; the result depended on which architect performed the validation. All involved domain architects had to accept the solution design in order for the project to receive a building permit. Projects sometimes had to wait months in order for their design to be accepted, because the domain architects could not agree on the outcome. The feedback projects got on the rationale why a solution was rejected, and the explanation on what to improve in their design in order to pass the validation successfully was often insufficient.

In order to deviate from a policy, or request permission to continue implementing the solution when the design was rejected by the EA function, project managers at company A could request a waiver. Decision making about granting projects a waiver was not transparent; they were granted based on undefined criteria, and inadequately communicated to the stakeholders. Domain architects were not always informed about a granted waiver. During the next solution validation they rejected the solution of a project that were granted a waiver. This resulted in projects being stopped even though a waiver was granted, to the frustration of various EA stakeholder groups.

Company B had a similar procedure for projects to request permission to deviate from a EA policy. The EA council that handled these requests was not fully effective.

There were too many EA policies at company A. They were unstructured, and the formal status of many of them was often unknown; there was no life cycle and change management for the policies. The EA policies the domain architects created were often not tested before they were implemented. Because there was no feedback loop from project level upwards, the domain architects were not aware of the practical applicability of the EA policies. There was no central administration of escalations and waiver requests to allow identifying malfunctioning EA policies to be changed. This all resulted in many projects deviating from the EA policies because they were impossible to work with.

The EA policies at company B were also not tested before they were implemented, and there was no feedback loop from projects upwards. Company B did have a central administration of escalation and waiver requests, but these were not used to identify malfunctioning EA policies for improvement.

4 Lessons Learned

Section 3 describes only a fraction of the findings we collected during the EA function assessment we conducted at the large international company A. However, this case shows that the EA maturity level in this company requires quite some improvement. It also shows it is insufficient to only take EA delivery into account to be truly effective with EA; both EA decision making and EA conformance have to be considered as well. In this section we elaborate on the key lessons we have learned.

1) Governance and collaboration must go hand in hand
The case study at company A shows that, if there are no formal and informal structures and processes, it is hard for EA stakeholders to trust each other and to work together. For example, an informal process of EA delivery performing an intake to pro-actively explain projects that are starting up how to create a solution design that satisfies the desired quality criteria, and conforms to the policies may help considerably. This will result in project managers and designers to better understand the purpose and working of solution validations, and deliver high quality designs. However, formal processes are also required. For example, having a transparent policy approval procedure, and a standard procedure for publishing the policies in a central, well-structured repository. This would make it more clear for the EA stakeholders, who are to conform with the policies, which EA policies apply to them. Therefore, it is vital to have both formal and informal structures and processes in place [28]. Formal processes ensure proper connection and coordination of EA decision making and conformity. Informal processes stimulate collaboration. Only combining both allows an effective implementation of EA governance in complex and dynamic environments [12].

2) Don't omit steps in the process model; keep the learning cycle in tact
A feedback loop is essential in getting EA products to be accepted and adhered to at project level. For example, the case study at company A shows that EA policies were not tested, and were not always applicable in practice. By ensuring a feedback loop from projects to the domain architects will solve this issue. This feedback loop may be implemented in the formal processes (i.e., make changes to policies based on escalations and waivers), or informal processes (e.g., by having regular meetings between architects who create the policies and designers who use them).

Having architectures at enterprise and domain level which are connected is vital in getting horizontal integration across domains. For example, the case study at company A shows that there was no enterprise infrastructure architecture available for the domain architects in order to integrate the various infrastructural domains. This illustrates that if one or more steps in the EA process model is omitted, the EA learning cycle becomes incomplete, with negative results.

3) Keep decision making and conformance reviews transparent and consistent
In order for the EA stakeholders to accept EA decision making and EA conformance validation results, it is vital to be transparent and consistent [12]. For example, the case study at company A shows that an unpredictable and unexplained validation result leads to frustration with the project manager and designer. This frustration will decrease with a transparent and consistent validation process, providing that proper feedback is given to guide the validation outcome. Regarding EA decision making, again transparency and consistency is essential. For example, the case study at company A shows that EA stakeholders will become frustrated with impractical and conflicting EA policies, and opaque EA decision making. In this case, transparent decision making regarding policies by representatives of the verticals impacted by those policies will increase the acceptance with the EA stakeholders within those verticals.

4) Governance bodies must represent all EA stakeholder groups with conflicting interests
An organization typically consists of various stakeholder groups at different organizational levels that have conflicts of interest, resulting in power struggles and political disputes [29]. For example, in the large international company we performed our case study there was a conflict of interest between solution delivery centers within the verticals delivering IT solutions, and the data center that deploys those IT solutions. The solution delivery center is concerned with providing a solution that best fits the business requirements; the data center wants to ensure the stability and continuity of the data center. The composition of an EA governance body is vital in properly addressing these conflicts of interest in decision making in order for EA governance to be effective [12].

5 Conclusions

Up till now, the literature provided a fragmented description of the EA function. In this article, we provide an integral description of the EA function in order to set the norm in performing EA function performance assessments. The case study we discuss in detail in this article shows that the maturity of EA functions is typically quite low, resulting in low performance of those EA functions. A second case study – due to reasons of confidentiality we do not describe this case study in detail in this article – confirmed this. In order to properly identify the essential points of improvement and compose an effective improvement plan, one needs a holistic perspective on the EA function. Comparing a specific EA practice with our integral EA function reference model, using an assessment model describing the standard topics of investigation, allows for EA practices to be compared with each other. Our NAOMI approach provides both an EA function reference, and assessment model. In this article we describe our EA function reference model we use to design and implement EA functions within organizations based on the assessment outcome.

In order to better understand what determines the performance of the EA function, we are conducting an empirical study to validate our EA performance framework, which addresses the three main topics of EA efficiency, EA effectiveness, and EA stakeholder satisfaction. We conducted an exploratory study on EA stakeholder perception of EA function performance [19], and we are currently constructing a stakeholder satisfaction assessment approach based on that exploratory study in order to extend our NAOMI approach.

References

1. Boh, W., Yellin, D.: Using Enterprise Architecture Standards in Managing Information Technology. Journal of Management Information Systems, 23(3), 163–207 (2007)
2. Van der Raadt, B., Soetendal, J., Perdeck, M., Van Vliet, H.: Polyphony in Architecture. In: Proceedings 26th International Conference on Software Engineering (ICSE 2004), pp. 533–542. IEEE Computer Society, Los Alamitos (2004)
3. Ross, J.W., Weill, P., Robertson, D.C.: Enterprise Architecture as Strategy – Creating a Foundation for Business Execution. Harvard Business School Press, Boston (2006)
4. Bird, G.B.: The business benefit of standards. StandardView 6(2), 76–80 (1998)
5. Van der Raadt, B., Slot, R., Van Vliet, H.: Experience Report: Assessing a Global Financial Services Company on its Enterprise Architecture Effectiveness Using NAOMI. In: Proceedings of the 40th Annual Hawaii international Conference on System Sciences (HICSS 2007), p. 218b. IEEE Computer Society, Washington (2007)
6. Peyret, H., Leganza, G., Hoekendijk, C., King, O., McCormack, M., Carini, A.: Enterprise Architecture Tools. In: The Forrester Wave^{TM}, Q2, April 25 (2007)
7. Open Group.: TOGAF (The Open Group Architecture Framework) version 8.1.1 ('The Book') (2007), `http://www.opengroup.org/bookstore/catalog/g063v.htm`
8. Mulholland, A., Macaulay, A.L.: Architecture and the Integrated Architecture Framework. Capgemini (2006), `http://www.capgemini.com/services/soa/ent_architecture/iaf/`
9. Clerc, V., Lago, P., Van Vliet, H.: The Architect's Mindset. In: Overhage, S., Szyperski, C.A., Reussner, R., Stafford, J.A. (eds.) QoSA 2007. LNCS, vol. 4880. pp. 231–249. Springer, Heidelberg (2008)
10. META Group, Inc.: Architecture Capability Assessment. META Practice, Vol. 4(7), META Group, Inc. (2000)
11. Van den Berg, M., Van Steenbergen, M.: Building an Enterprise Architecture Practice: Tools, Tips, Best Practices, Ready-to-use Insights. Springer, Heidelberg (2006)
12. Peterson, R.: Crafting Information Technology Governance. Information Systems Management 21(4), 7–22 (2004)
13. Smolander, K., Päivärinta, T.: Describing and Communicating Software Architecture in Practice: Observations on Stakeholders and Rationale. In: Pidduck, A.B., Mylopoulos, J., Woo, C.C., Ozsu, M.T. (eds.) CAiSE 2002. LNCS, vol. 2348, pp. 117–133. Springer, Heidelberg (2002)
14. Ross, J.W., Beath, C., Goodhue, D.L.: Develop long-term competitiveness Through IT assets. Sloan Management Review 38(1), 31–45 (Fall, 1996)
15. Van der Raadt, B., Hoorn, J.F., Van Vliet, H.: Alignment and Maturity Are Siblings in Architecture Assessment. In: Pastor, Ó., Falcão e Cunha, J. (eds.) CAiSE 2005. LNCS. vol. 3520, pp. 357–371. Springer, Heidelberg (2005)
16. Simonsson, M., Lindström, Å., Johnson, P., Nordström, L., Grundbäck, J., Wijnbladh, O.: Scenario-Based Evaluation of Enterprise Architecture - A Top-Down Approach for CIO Decision-Making. In: Proceedings of the International Conference on Enterprise Information Systems (ICEIS 2005) (May 2005)

17. Kazman, R., Klein, M., Clements, P.: ATAM: Method for Architecture Evaluation. Technical Report, CMU/SEI-2000-TR-004 (2000), http://www.sei.cmu.edu/pub/documents/00.reports/pdf/00tr004.pdf
18. Pulkkinen, M.: Systemic Management of Architectural Decisions in Enterprise Architecture Planning. Four Dimensions and Three Abstraction Levels. In: Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS 2006), p. 179a (2006)
19. Van der Raadt, B., Schouten, S., Van Vliet, H.: Stakeholder Perception of EA Function Performance, In: Second European Conference On Software Architecture (ECSA 2008), (Submitted to, April 2008)
20. Moore, G.A.: Strategy and your stronger hand. Harvard Business Review 83(12), 62–71 (2005)
21. Janssen, M., Joha, A.: Issues in relationship management for obtaining the benefits of a shared service center. In: Janssen, M., Sol, H.G., Wagenaar, R.W. (eds.)Proceedings of the 6th international Conference on Electronic Commerce, ICEC 2004, Delft, The Netherlands, vol. 60, pp. 219–228. ACM, New York (2004)
22. Van der Aalst, W., Hofstede, A., Weske, M.: Business process management: A survey. In: van der Aalst, W.M.P., ter Hofstede, A.H.M., Weske, M. (eds.) BPM 2003. LNCS, vol. 2678, pp. 1–12. Springer, Heidelberg (2003)
23. Leana, C.R., Barry, B.: Stability and Change as Simultaneous Experiences in Organizational Life. The Academy of Management Review 25(4), 753–759 (2000)
24. Crossan, M., Lane, H., White, R.: An organizational learning framework: From intuition to institution. Academy of Management Review 24, 522–537 (1999)
25. Baker, B.: The role of feedback in assessing information systems planning effectiveness. The Journal of Strategic Information Systems 4(1), 61–80 (1995)
26. March, J.G.: Exploration and Exploitation in Organizational Learning. Organization Science 2(1), 71–87 (1991)
27. Kotter, J.P.: Leading Change. Harvard Business School Press, Boston (1996)
28. Henderson, J.C.: Plugging into strategic partnerships: The critical IS connection. Sloan Management Review 31(3), 7–18 (1990)
29. Eisenhardt, K.M., Bourgeois III, L.J.: Politics of Strategic Decision Making in High-Velocity Environments: Toward a Midrange Theory. The Academy of Management Journal 31(4), 737–770 (1988)