Towards a Method Framework for Enterprise Architecture Management – A Literature Analysis from a Viable System Perspective

Sabine Buckl, Florian Matthes and Christian M. Schweda

Chair for Software Engineering of Business Information Systems (sebis),
Technische Universität München,
Boltzmannstr. 3, 85748 Garching, Germany
{buckls,matthes,schweda}@in.tum.de
http://wwwmatthes.in.tum.de

Abstract. The discipline of enterprise architecture (EA) management is albeit a long history still developing. This is becomes obvious, when literature on the EA management function is analyzed. Multiple approaches describe different make-ups for the overall function, while a common sense does yet not exist. In this paper, we analyze EA management functions as proposed in literature from a systemic perspective and derive typical management activities such a function should encompass. Based on these activities, a method framework for EA management is derived, which is assessed in a case study from the financial industry.

1 Motivation and overview

Enterprise architecture (EA) management is a discipline, which has recently gained increased attention from academia and practitioners. Thereby, a few topics which are nowadays regarded to be part of EA management, have a long history in information systems research. This can be exemplified with the topic of business-IT-alignment, which has been discussed e.g. by Henderson and Venkatraman in the late nineties as strategic alignment [1]. While these discussions might have catalyzed the evolution of EA management, the overall discipline is still subject to ongoing development. This in particular applies as different research communities continue to argue on the perspective, from which EA management should be approached. Some researchers emphasize on business aspects, advocating for an understanding of EA management as an economic management discipline (cf. Frank in [2]). In contrast, other groups point to the engineering aspects (cf. Aier et al. in [3]) or take a systemic perspective on the topic (cf. Buckl et al. in [4] and Wegmann in [5]). The approaches nevertheless agree that EA management needs to provide a holistic view on an enterprise, accounting for aspects from all layers, ranging from business to IT aspects.

Regardless of the question of perspective, other indications for the ongoing development of the EA management discipline exist. A prominent example for this is the topic of EA modeling. Although most EA management approaches

emphasize on the importance of modeling the EA, no common metamodel (called information model in accordance with Buckl et al. in [6]) has yet been established. In the last years, many information models were proposed but none of them has yet gained broad acceptance. Some researchers even challenge the hypothesis that such a model exists (cf. Buckl et al. in [7] and Kurpjuweit and Winter in [8]). They expect enterprises to have largely different expectations on the benefits of EA management, and therefore assume that an information model is an enterprise-specific artifact. Similar discussions apply to the overall make-up of the EA management function. Many different activities have been argued to be inseparable parts of EA management (see Section 3). In contrast, approaches presenting constituents of the EA management function or comprehensive processes descriptions are rare in academic literature (for one example cf. Hafner and Winter in [9]). Similarly, few practitioners (cf. Niemann in [10] and Schekkerman in [11]) and standardization bodies (cf. The Open Group in [12]) discuss processes but stay on a fairly abstract level. These processes are usually complemented with a remark that "they have to be adapted to the company's needs"[12], while the details of this adaptation are left to the reader.

We expect the EA management function, similar to the information model, to be enterprise-specific, although – on a more abstract level – every EA management function might be comprised of similar activities. Thereby, we must provide additional clarification in respect to the understanding of the term EA in different research communities. While some researchers refer to the term EA as the management function, aiming at managing the evolution of the EA, others regard the EA as the inevitable fact, which refers to the make-up of the enterprise summarized as "every system has an architecture"[13]. The terminology used in this paper adopts the later wording and clearly distinguishes between the artifact (EA) and the corresponding management function (EA management).

The article presents a first step towards establishing a consolidated method framework for EA management, which can be configured according to the enterprise-specific needs of a company. The framed method is grounded in a systemic perspective on EA management, which is exhibited in Section 2. From this perspective, Section 3 revisits prominent approaches to EA management from literature and collects typical work packages that these approaches propose. In addition, the representations of the EA, the so called EA descriptions, are analyzed. With the activities and the descriptions at hand, Section 4 proposes a method framework for EA management, consisting of four main activities of an EA management function and three different types of EA descriptions. The framework further describes how the activities relate to each other, and specifies which descriptive information about the EA is exchanged between them. In this respect, it can be regarded as abstract method framework for the EA management function, providing the answers to the article's research questions:

- *Which typical activities constitute an EA management function?*
- *Which information objects are created by, exchanged between, and used for these activities?*
- *How do the activities relate in a method framework for EA management?*

Section 5 sketches the results of a case study on the EA management function of a company in the financial industry. We thereby show to which extent the method framework can be validated. The article concludes with Section 6, which summarizes the findings, shows limitations of the research approach chosen, and gives indications of future areas for research.

2 EA management from a systemic perspective

Enterprises form highly complex systems consisting of various different elements interlinked by a large number of interdependencies. These systems are further embedded into a changing environment that they continuously have to adapt to. In particular, market changes and new legal requirements force enterprises to adjust their architectures, e.g. to rework their business processes or to evolve their IT artifacts. Additionally, newly emerging technologies may enable new business opportunities that an enterprise should proactively seek to gain competitive advantage (Ross et al. in [14] and Wagter et al. in [15]). Both the reactive and the proactive change of the enterprise fall into the responsibility of enterprise-level management functions, as application or project portfolio management, but also of the EA management function. In this respect, the different management functions on the one hand and the EA management function on the other hand form an interacting system. Understanding this system of systems is a necessary prerequisite for developing a method framework for EA management.

The viable system model (VSM) (Beer in [16, 17, 18]) provides a framework for describing complex management systems from a systemic perspective. In the following, we discuss the five subsystems of the VSM – operation, coordination, control, planning, and identity – and identify these subsystems with constituents from the EA management system.

- The *enterprise-level management functions* form system one (operation) directly changing the EA via projects. Especially the management functions surrounding the project lifecycle contribute to system one. Exemplary functions are: enterprise-wide demand management, where demands are captured and prioritized; strategies and goals management, where demands and projects are aligned with the enterprise's goals; synchronization management, where project dependencies are monitored (cf. Wittenburg et al. in [19]).
- The *communication function* of EA management forms system two (coordination) by which architecture descriptions are distributed via appropriate communication channels. Thereby, the different enterprise-level management functions (cf. Wittenburg et al. in [19]) are provided a shared understanding of the as-is (current) and the to-be (planned) state of the EA. Based on his shared understanding peer-level coordination between the enterprise-level management functions should be fostered.
- System three (control) forms the *reactive function* of EA management, that establishes higher level control over the coordination function. In particular, the

reactive EA management observes the behavior of the enterprise-level management functions in coordination and assures that no 'oscillatory' effects between these functions develop. This would for instance be the case, if projects would adapt to comply with current architectural standards for business applications, while simultaneously the standards were adapted to incorporate the realities of the new application portfolio.
- Where system three ensures *stability* in the interactions of the enterprise-level management processes, EA management also encompasses a *proactive function* in system four (planning). Latter system is responsible for anticipating changes in the environment of the enterprise and for addressing these changes by altering the status-quo that is maintained by the underlying *homeostatic* control in system three.
- Completing, system five (identity) is responsible for EA management governance, i.e. is concerned with questions of the overall scope and reach of EA management. It further shapes the design of the EA management function itself. Thereby, it ensures a balance between short-term and long-term efforts, and steers the EA management system as a whole.

3 State-of-the-art in EA management literature

This section provides an overview about selected EA management approaches from a viable system perspective as introduced above. Thereby, we focus on activities described as being part of the EA management function and detail on the EA descriptions they expect for input or provide as output. In the description of the approaches, the original terms employed by the authors are used.

One of the most prominent frameworks for EA management is proposed by The Open Group – *The Open Group Architecture Framework (TOGAF)* ([12]). The core contribution of TOGAF in respect to describing the EA management function is the *Architecture Development Method (ADM)*, which delineates how an EA can be developed and maintained. The ADM describes EA management as an iterative and stepwise process consisting of different phases. The initialization of one EA management cycle is performed in the *preliminary phase*, where decisions about the scope and reach of the management endeavor are made (system 5). Thereby, the topic how to link EA management to other enterprise-level management functions is decided upon. The following four phases *architecture vision, business [architecture], information systems [architecture]*, and *technology architecture* are concerned with the development of a target state, the investigation of the current state, and gap analyses comparing these states. From a viable system perspective these phases present the reactive and proactive EA management. The transition planning from the status quo to the desired target architecture is performed in phase *opportunities and solutions* and decided upon in phase *migration planning*. The execution of the transformation is monitored in the *implementation and governance* phase. Finally, the overall performance of the management process is measured and assessed in the phase *architecture*

change management, which therefore deals with aspects of EA management governance. The aforementioned phases are continually adapted to the needs and concerns of the stakeholders in an activity, called *requirements management*. TOGAF complements the description of the activities with elaborations on the input and output artifacts of each phase – namely visualization artifacts, e.g. a *solution concept diagram* or a *business interaction matrix*, as well as textual documentations, e.g. reports or catalogs. The aforementioned EA descriptions together with the stakeholder management, which a dedicated chapter of TOGAF emphasizes on, contribute to the communication task of EA management. A characteristic of the TOGAF framework is that each iteration of the ADM cycle is project-driven, which on the one hand guarantees the sponsorship for the EA management initiative, but on the other hand makes it hard to ensure the continuity of the outcomes. A consequence of this approach is the absence of an activity, which keeps the EA documentation up to date.

Similar to the TOGAF ADM, Schekkerman [11] describes EA management as an iterative and stepwise process. Each iteration starts with the development of the *EA vision* (phase 1), which defines the environment, business drivers, and guiding principles. In addition, the *scope and context* (phase 2) as well as the *goals, objectives, and requirements* (phase 3) of the EA management endeavor are defined. Subsequent phase 4 derives different *opportunities and solutions* from existing documentations of the current state and future architecture plans. Thereby, special attention is paid to support decision making during management via adequate visualizations, models, and reports, which are chosen in this phase. Based on the opportunities identified in the preceding phase, different *evolution scenarios* are developed and evaluated regarding their organizational impact (phase 5). The costs and benefits of the scenarios are analyzed via *business case calculations* (phase 6) to support funding of the EA management endeavor. The results of the preceding phases are used in phase 7 to set up a *scheduled transformation plan*, including capability planning for the EA. Finally, a *governance structure* is implemented (phase 8), which defines the responsibilities as well as roles, groups, and committees needed. The EA descriptions developed in and exchanged between the phases are only briefly alluded to. Further, viewpoints used in the different phases of the EA management process are only discussed with regards to content without providing graphical representation. Furthermore, EA management governance is not presented as being part of the EA management function, although the importance of EA management maturity is discussed and a model to assess the maturity is presented.

Niemann also emphasizes on the iterative and stepwise nature of EA management incorporated in the corresponding management "cycle", that consists of four phases – *document, analyze, plan,* and *act* – and a parallel *check phase* [10]. The document phase is concerned with gathering and maintaining information about the current state of the EA. The architects have to decide on the adequate level of detail of the documentation and define the appropriate EA descriptions to populate the model as part of the communication system of EA management. For the latter case Niemann further proposes different kinds of visualizations

in [10], which can be used to document parts of or provide an overview over the EA. Although the approach elaborates on questions regarding *what* should be documented, it does not detail on the question *how* this information should be gathered and maintained. Based on the documentation an analysis of the current state of the EA is performed in order to identify potentials for improvement and optimization (reactive EA management). Niemann presents different areas for analysis, e.g. dependencies, heterogeneity, complexity, or conformity and provides methods as well as appropriate visualizations to perform the analysis. During the plan phase integrated development plans leveraging identified potentials for improvement and optimization are established. They represent planned states of the EA that are further assessed regarding their impact on e.g. business and IT goals, costs, and risks. The assessment should result in the selection of the optimal development plan in respect to the criteria devised before. This plan is realized in the act phase. Therefore, on the one hand reference architectures and blueprints are developed and implemented. On the other hand the required governance structures and processes are set up, e.g. the role and responsibilities of the enterprise architect are refined. In respect to the viable systems perspective a focal point in the approach lies on the reactive system of EA management. The EA management governance system is presented in the check phase, in which the performance of the previously described phases is measured and controlled. Thereby, key performance indicators (KPIs) are defined to analyze the overall performance of the EA management endeavor.

Hafner and Winter present a *consolidated process model for enterprise application architecture management* in [9]. Although the paper restricts itself to enterprise application management, the approach is discussed here, as the presented process model is designed with the goal of effective and efficient business-IT-alignment, and therefore takes an EA perspective. The process model contains the phases *architecture planning, architecture development, architecture communication,* and *architecture lobbying*. The architecture planning phase is concerned with the documentation of current states of the EA. Thus, also EA principles are identified, derived, and updated, which guide the evolution of the EA. The proactive and reactive aspects of EA management are reflected in the architecture development phase, in which strategic and operational requirements regarding the EA are continuously recorded, consolidated, and prioritized. Subsequently, these requirements are incorporated in planned states of the EA. The phases architecture communication and architecture lobbying explicitly refer to the communication function of EA management. Nevertheless, aspects on how to relate the EA management endeavor to existing enterprise-level management processes are only briefly alluded to. More precisely, Winter and Hafner in [9] resort their approach to identifying target groups for training, information delivery, etc. While the task of analyzing the EA is made explicit as part of the consolidated process model, the assessment and improvement of the EA management approach itself (EA management governance) is not discussed.

Another prominent approach in the field of EA management is the *systemic enterprise architecture methodology (SEAM)* [5]. The methodology defines the

role of EA management as to *federate the efforts of the specialists [from the enterprise-level processes] to ensure successful projects.* This point-of-view interprets EA management as the glue between the different processes, i.e. bringing together information in this multi-disciplinary environment, thereby especially emphasizing on the communication function. The federation of efforts is achieved via enterprise models, which form means of analysis and communication of EA relevant information. These models account for the multi-disciplinarity of the environment, but go beyond specific models for each discipline, e.g. process chains or network topology models. They provide an integrated view on the enterprise. In [20], Le and Wegmann 2005 highlight two additional aspects of EA management: firstly, the reactive aspect, which deals with necessary business and technology changes ex post; secondly, the proactive aspect, which anticipates future changes of that kind and prepares the enterprise to them by increasing agility and flexibility. In contrast SEAM abstains from discussing questions of how to establish and govern the EA management process.

In addition to the aforementioned approaches, which claim to define their own EA management function, various approaches exist that focus on selected topics in the context of EA management. Lankhorst et al., for example, detail on the topics of EA communication, documentation, and analysis in [21]. Therefore, a specialized modeling language is introduced, which fosters the communication between business and IT stakeholders, and can also be used for documenting current, planned, and target states of the EA. As means for decision-support, different kind of analysis techniques, including analtytical and simulation techniques are discussed. Thus, the approach focuses on aspects of reactive and proactive EA management in the sense of a viable system perspective, while the aspect of the communication system is discussed as a side-effect of the proposed modeling. Similar considerations hold for the approach of *multi-perspective enterprise modelling (MEMO)* presented by Frank [2]. The approach focuses on the activity of EA modeling by providing special purpose languages for different parts of the EA, e.g. the *IT modelling language (ITML)* [22] – for modeling IT related aspects – or for different activities performed in the context of EA management, e.g. the *ScoreML* [23] – contributing to the field of analyzing EAs. Although, the EA management function is not in the focus of the approach of Frank, he contributes to the field of reactive and proactive EA management in the terms of our viable system perspective.

4 A method framework for EA management

Based on the above discussions of the EA management function and special purpose approaches for dedicated EA management activities, we devise a method framework for EA management. Central to our framework is the understanding of the three different architectural states – *current, planned,* and *target* – that can be found throughout the approaches discussed in Section 3. Table 1 revisits the state-of-the-art in EA management with a focus on EA descriptions.

	[2]	[5]	[9]	[21]	[10]	[11]	[12]
target state	○	●	◑	◑	○	●	●
planned state	●	●	●	◑	●	●	●
current state	●	●	●	◑	●	●	●

Table 1. EA description in literature

These architectural states are subject to different activities during EA management. Aggregating them to a high level view, we identified activities for

developing and describing an architecture state, which is concerned with describing the enterprise-level management functions (system one) as well as developing planned and target states of the EA in an proactive manner (system four),

communicating and enacting an architecture state, which considers communication function aspects (system two),

analyzing and evaluating and architectural state, which analyses architectural (system three), and

configuring and adapting the EA management function itself, which represents EA management governance (system five).

Table 2 revisits the state-of-the-art from Section 3 and summarizes the results.

	[2]	[5]	[9]	[21]	[10]	[11]	[12]
Develop & describe	◑	●	●	◑	◑	●	●
Communicate & enact	○	◑	●	○	●	○	○
Analyze & evaluate	●	●	●	●	●	○	●
Configure & adapt	◑	●	○	○	◑	◑	◑

Table 2. EA management activities in literature

The method framework for EA management provides the abstract frame consisting of the aforementioned activities and EA descriptions, any EA management endeavor encompasses. This framework is not concern[1]-specific, i.e., it is a generic method that can be used in combination with typical EA management concerns as e.g. discussed in [25]. As detailed below, the activity *configure and adapt* activity is concerned with determining the scope and reach of the EA management function. Thus, the goals of the EA management endeavor are mapped to corresponding concerns, which can be detailed utilizing concern-relationships (cf. [26]). Subsequently, we introduce the activities of the method framework briefly and provide additional details on the architectural descriptions that are created and consumed by activities. The activity *develop and describe* is further

[1] The term *concern* is used here in accordance with its definition in the ISO Std. 42010, which defines a concern as "those [areas of] areas of interest which pertain to the system's development, its operation or any other aspect that are critical or otherwise important to one or more stakeholders" [24].

subdivided to exemplify the development and description of current, planned, and target states of the EA. Similar subdivisions could be introduced for the other activities but are not detailed here for reasons of brevity.

Develop & describe target state – This activity is concerned with creating a target state of the EA based on the business and IT strategies that the enterprise seeks to implement. Different sub-activities are e.g. the creation of a target business architecture and the design of a target application portfolio. In the target business architecture the future product portfolio of the enterprise is reflected and complemented by corresponding business processes. The target application portfolio is designed towards the support for the intended business architecture. In addition, a target infrastructure architecture is set up, describing the basic services as well as the execution environment, which the business applications can rely on. The target state further goes beyond simple architectural descriptions on different EA levels. It establishes *architecture principles* that guide the evolution from the current to the target state. Such principles reflect specific parts of the business or IT strategy that do not directly shape the make-up of the future architectures. To exemplify this, one could think of an outsourcing strategy that would be converted to an architectural principle demanding that support for business processes of low criticality is not provided in-house, if a suitable outsourcing provider is available.

Develop & describe current state – This activity is concerned with creating a description of the current state of the EA, i.e., the as-is architecture. Thereby, all levels of architectures ranging from business and organization level, via the application and information level, to the infrastructure and data level are considered. Further, information on projects, which affect the EA, as well as on business and IT strategies is documented. The same is true for information on current architectural principles and standards. The develop & describe current state activity is thereby greatly influenced by the EA concerns that drive the EA management function. For implementing the activity, different ways can be used, ranging from documentation endeavors on regular basis, to continuous endeavors accompanying the EA relevant projects (cf. Moser et al. in [27]). Irrespective the chosen way, the activity *develop & describe current state* provides and maintains an appropriate description of the current state of the EA.

Develop & describe planned state – With the target and the current state at hand, the activity derives intermediary architectural plans that are realized by projects. These projects are thereby, not solely derived from the two architecture descriptions, but also based on the demands, from enterprise-wide demand management. In this respect, a planned state is not expected to strictly develop towards the target state, but can also pursue a different road of development in response to an urgent business need. The intermediary states are in this way tightly coupled to the planned projects that are necessary for their implementation. More precisely, each planned project of EA relevance contributes some changes to an intermediary state of the EA. By selecting sets of architecturally compatible projects, i.e., projects whose changes do not interfere, different scenarios for the intermediary states can be derived. The descriptions of these

scenario architectures, which also encompass references to the thereby addressed demands, form the output of the activity *develop & describe planned state*.

Communicate & enact – EA management is heavily concerned with making plans as well as defining architectural principles and propagating them to the enterprise-level management functions. This propagation aims at influencing the decision making in the related functions. Therefore, communicating and enacting architectural principles is always connected to contributing to the decision making in the enterprise-level management functions. Enacting takes the architecture plan and principles as input and effects the decision making in the other management functions. Again, as with the other activities, different ways to implement the activity *communicate & enact* exist. These range from the fairly non-interfering way of informing the decision makers to the most powerful method of having the right to stop projects, which are non-conformant to the EA. This activity hence always takes the description of the planned EA and the architectural principles as input, but can create multiple output artifacts that are handed over to the enterprise-level management functions. These artifacts thereby depend on the method of communication and enactment chosen.

Analyze & evaluate – At some points during the management of the EA, different states of the EA, i.e. current, planned, and target state, or architectural plans, i.e. different scenarios of the planned state, exist. The *analyze & evaluate* activity makes these architectures comparable in order to prepare a subsequent decision on the state to pursue. Different properties of the architecture may thereby be of interest, ranging from the compliance with architectural principles to economic properties. Functional properties of the architecture, as e.g. the provided business support, may also be important (cf. Niemann in [10]). Most commonly non-functional properties, e.g. the availability of certain business services (cf. Johnson et al. in [28]) or the flexibility of the overall architecture are used for analyzing different states. In literature, a broad variety of approaches to EA analysis have been proposed, differing widely in respect to the employed level of formalization, ranging from expert-based assessments (cf. Niemann in [10]) to indicator-based computations (cf. Frank et al. in [23] and Iacob and Jonkers in [29]). The approaches also vary concerning their time reference: some approaches are designed to analyze current architectures (cf. Niemann in [10]), while other approaches (cf. De Boer et al. in [30]) provide prediction capabilities that can be used to analyze architectures not yet realized.

Configure & adapt – Before starting an EA management endeavor the goals and objectives of the initiative should be clearly defined. Based on these goals, decisions must be taken during the activity *configure & adapt* regarding the management subject of the EA management function. Relevant stakeholders must be identified and the concerns, which should be addressed, need to be defined. Further, decisions on the scope and reach on the EA management function must be made, ranging from bottom-up approaches, in which only a certain division of the enterprise is considered regarding a certain aspect like standardization, to top down approaches, where the whole enterprise is examined regarding multiple aspects like risk management, compliance, etc. After the

initial establishment of an EA management function, the configure and adapt activity is concerned with measuring the overall performance of the EA management function. Adaptations can be necessary, e.g. if the enterprises mature or the scope and reach of EA management change.

The above described activities form an idealized framework. In reality, the different activities of the EA management function are executed parallel and with distinct frequency and duration. The method fragment does controversely not add prescriptions on the frequency and ordering of the activities and steps to be taken, but provides an abstract and general frame of the main constituents of an EA management function.

5 A case study from the financial industry

In order to validate the method framework for EA management, we conducted a case study in the financial industry. Subsequently, we give a short characterization of the enterprise at hand and then discuss to which extent the method framework can be used to classify the approach taken.

The case study was conducted at an internationally operating bank from Germany. The topic of EA management has a long history in this enterprise since a merger in the year 1996. Prior to the merger both companies independently conducted enterprise-wide data modeling endeavors. After the merger, the enterprise-wide data models were maintained, although a change in the focus as well as the reach took place. In certain parts of the enterprise the focus shifted towards a strongly business process centric approach, while other parts continued with data modeling. In the year 2002 the term EA management makes its first appearance, when a project was launched to increase the business-IT-alignment based on a holistic approach. In this holistic approach architectural information from different parts of the enterprise was consolidated and used to identify fields for action. In order to assess the advances made in this field, a similar project was launched in the year 2005, which refined the utilized EA management process. The take-over by an international banking company at the end of 2005 changed the overall make-up of the company significantly. In particular, the IT departments of the formerly independent enterprises, as well as the IT assets developed, operated, and managed by them, were to undergo extensive changes leading to an increased centralization of structures.

The EA management function currently operated at the banking company encompasses the evolution of the technical as well as the business architecture. Thereby, the technical architecture is organized in the following layers: operative, system, integration, and application layer. The business architecture covers the business process and the business model layer. The goals of the efforts are among others defined as follows: The EA management function

1. supports planning processes,
2. demonstrates benefit of architecture development,
3. identifies and aligns needs for action,

4. develops future scenarios of the EA as well as migration plans, and
5. ensures balance between short-term realization of business functions and long-term improvement of the EA.

These goals were selected as they can be used to illustrate the systemic approach to EA management, which was taken by the banking company. Thereby, the goals 1 to 5 directly map to the respective systems of the viable systems approach as presented in Section 2, while no counterpart of the EA management governance function or the activity *configure & adapt* is alluded to.

The management function established at the banking company consists of the following activities:

(1) **Creation and adjustment of IT strategy:** Based on the enterprise business strategy, the IT strategy is developed, which includes information on core competencies, products, business areas, etc, and is used to design a target state of the EA. Furthermore, an IT security strategy is formulated.
(2) **Development and update of architectural guidelines and standards:** Architecture principles are identified and guidelines as well as standards are developed and updated on this basis. To decide on new guidelines or standards, an architecture board was introduced.
(3) **Identification of needs for action originating from business and IT:** Business and IT demands are collected and analyzed in respect to their strategic or operative importance. The identified needs are further assessed and prioritized according to the architectural principles identified as architecture conformity, costs, risks, benefit, etc.
(4) **Development and update of architecture artifacts:** EA descriptions, like viewpoints, artifacts, guidelines, and standards are developed from three perspectives: the functional, technical, and security perspective. They are updated on a yearly basis either prior to or after the creation of the annual project plan. Therefore, defined EA descriptions like e.g. the technical building block maps are used.
(5) **Check architecture conformity:** The EA conformity in respect to the architectural principles is ensured via quality gates for projects. Thereby, the vertical escalation in the organizational structure depends on the scope of the project.

The EA management function as presented above was subject to various changes in the past, where the performance of the function itself was assessed. Such an assessment took place in the year 2005, where impediments, which hampered the successful management of the EA, were identified. As a consequence of this assessment, decisions on architectural guidelines (cf. Activity (2)) were not longer taken in a central board, if the activities have only local impact. Thereby, an overloading of the architecture board was prevented and the decision process was sped up. Although this assessment is not part of the documented process of EA management in the company, it refers to the EA management governance discussed in Section 2.

In summary, the EA management function established at the banking company can be mapped to the activities of the method framework presented in Section 4 as shown in Table 3.

	(1)	(2)	(3)	(4)	(5)
Develop & describe	●	●		●	
Communicate & enact					●
Analyze & evaluate			●		
Configure & adapt					

Table 3. Mapping the method framework to the banking company

6 Conclusion and outlook

This paper aims at establishing a method framework for EA management. Therefore, existing literature on EA management is analyzed from a viable system perspective. The objective of this analysis is the identification of typical management activities performed in this context. Furthermore, the architecture descriptions exchanged between these activities were of interest to analyze the relationships between the activities. As a result of the paper, it can be stated that a method framework for EA management should contain the following activities: *develop & describe* different states of the EA, *communicate & enact* architectural principles and plans, *analyze & evaluate* different states of the EA, and finally *configure & adapt* the EA management function. The identified activities could further be evaluated via observing a case study at a banking company, in which the EA management function of the company was analyzed. Nevertheless, the case study presented in this article only provides an ex post evaluation. In order to further investigate the applicability and suitability of the proposed activities, an ex ante setting, where the activities identified are used to establish an and enterprise-specific EA management function, is necessary. In addition, further case studies need to be conducted to prove the applicability in different industry sectors and for different company sizes.

The case study discussed in this paper hints to the need for configurability of the EA management function. Via configurable method building blocks for the different activities of the method framework, an effective EA management function can be designed and established. Making the configuration points explicit in the method framework, problems and exceptional situations during EA management can be linked back to these points, where adaptations have to take place as part of the activity *configure & adapt*.

This paper presents a method framework for EA management on a very abstract level. In order to foster the applicability of the approach in practice, more detailed information on the execution of the single activities would be beneficiary. Such best practice realizations could be documented as EA management

patterns (cf. [25]) for which the method framework would not only provide a classification but also would supply information on how to interrelate and integrate single patterns.

References

1. Henderson, J.C., Venkatraman, N.: Strategic alignment: leveraging information technology for transforming organizations. IBM Systems Journal **32**(1) (1993) 472–484
2. Frank, U.: Multi-perspective enterprise modeling (memo) – conceptual framework and modeling languages. In: Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS 2002), Washington, DC, USA (2002) 1258–1267
3. Aier, S., Kurpjuweit, S., Saat, J., Winter, R.: Enterprise Architecture Design as an Engineering Discipline. AIS Transactions on Enterprise Systems **1** (2009) 36–43
4. Buckl, S., Matthes, F., Schweda, C.M.: A viable system perspective on enterprise architecture management. In: 2009 IEEE International Conference on Systems, Man, and Cybernetics. (2009)
5. Wegmann, A.: The Systemic Enterprise Architecture Methodology (SEAM). Technical report, EPFL (2002)
6. Buckl, S., Ernst, A.M., Lankes, J., Matthes, F., Schweda, C., Wittenburg, A.: Generating visualizations of enterprise architectures using model transformation (extended version). Enterprise Modelling and Information Systems Architectures – An International Journal **2**(2) (2007) 3–13
7. Buckl, S., Ernst, A.M., Lankes, J., Schneider, K., Schweda, C.M.: A pattern based approach for constructing enterprise architecture management information models. In: Wirtschaftsinformatik 2007, Karlsruhe, Germany, Universitätsverlag Karlsruhe (2007) 145–162
8. Kurpjuweit, S., Winter, R.: Viewpoint-based meta model engineering. In Reichert, M., Strecker, S., Turowski, K., eds.: Enterprise Modelling and Information Systems Architectures – Concepts and Applications , Proceedings of the 2nd International Workshop on Enterprise Modelling and Information Systems Architectures (EMISA'07), St. Goar, Germany, October 8-9, 2007. LNI, Bonn, Germany, Gesellschaft für Informatik (2007) 143–161
9. Hafner, M., Winter, R.: Vorgehensmodell für das management der unternehmensweiten applikationsarchitektur. In Ferstl, O.K., Sinz, E.J., Eckert, S., Isselhorst, T., eds.: Wirtschaftsinformatik, Heidelberg, Germany, Physica-Verlag (2005) 627–646
10. Niemann, K.D.: From Enterprise Architecture to IT Governance – Elements of Effective IT Management. Vieweg+Teubner, Wiesbaden, Germany (2006)
11. Schekkerman, J.: Enterprise Architecture Good Practices Guide – How to Manage the Enterprise Architecture Practice. Trafford Publishing, Victoria, BC, Canada (2008)
12. The Open Group: TOGAF "Enterprise Edition" Version 9. `http://www.togaf.org` (cited 2010-02-25) (2009)
13. Rechtin, E.: Systems Architecting of Organizations – Why Eagles can't swim. CRC Press LLC, New York, NY, USA (1999)
14. Ross, Jeanne, W., Weill, P., Robertson, David, C.: Enterprise Architecture as Strategy. Harvard Business School Press, Boston, MA, USA (2006)

15. Wagter, R., van den Berg, M., Luijpers, J., van Steenbergen, M.: Dynamic Enterprise Architecture: How to Make IT Work. John Wiley (2005)
16. Beer, S.: The Heart of Enterprise. John Wiley, New York, NY, USA (1979)
17. Beer, S.: Brain of the Firm. 2nd edn. John Wiley, New York, NY, USA (1981)
18. Beer, S.: Diagnosing The System for Organisations. John Wiley, New York, NY, USA (1985)
19. Wittenburg, A., Matthes, F., Fischer, F., Hallermeier, T.: Building an integrated IT governance platform at the BMW Group. International Journal Business Process Integration and Management **2**(4) (2007)
20. Le, L.S., Wegmann, A.: Definition of an object-oriented modeling language for enterprise architecture. System Sciences, 2005. HICSS '05. Proceedings of the 38th Annual Hawaii International Conference on (2005) 179c
21. Lankhorst, M.: Enterprise Architecture at Work: Modelling, Communication and Analysis. Springer, Berlin, Heidelberg, Germany (2005)
22. Kirchner, L.: Eine Methode zur Unterstützung des IT-Managements im Rahmen der Unternehmensmodellierung. PhD thesis, Universität Duisburg-Essen, Berlin, Germany (2008)
23. Frank, U., Heise, D., Kattenstroth, H., Schauer, H.: Designing and utilising business indicator systems within enterprise models – outline of a method. In: Modellierung betrieblicher Informationssysteme (MobIS 2008) – Modellierung zwischen SOA und Compliance Management 27.-28. November 2008, Saarbrücken, Germany (2008)
24. International Organization for Standardization: Iso/iec 42010:2007 systems and software engineering – recommended practice for architectural description of software-intensive systems (2007)
25. Chair for Informatics 19 (sebis), Technische Universität München: Eam pattern catalog wiki. http://eampc-wiki.systemcartography.info (cited 2010-02-25) (2010)
26. Buckl, S., Matthes, F., Schweda, C.M.: Interrelating concerns in ea documentation – towards a conceptual framework of relationships. In: 2nd European Workshop on Patterns for Enterprise Architecture Management (PEAM2010), Paderborn, Germany (2010)
27. Moser, C., Junginger, S., Brückmann, M., Schöne, K.M.: Some process patterns for enterprise architecture management. In: Software Engineering 2009 – Workshopband, Bonn, Germany, Lecture Notes in Informatics (LNI) (2009) 19–30
28. Johnson, P., Nordström, L., Lagerström, R.: Formalizing analysis of enterprise architecture. In: Interoperability for Enterprise Software and Applications Conference, Bordeaux, France, Springer (2006) 10
29. Iacob, M.E., Jonkers, H.: Quantitative analysis of enterprise architectures. In Konstantas, D., Bourrières, J.P., Léonard, M., Boudjlida, N., eds.: Interoperability of Enterprise Software and Applications, Geneva, Switzerland, Springer (2006) 239–252
30. de Boer, F.S., Bonsangue, M.M., Jacob, J., Stam, A., van der Torre, L.: Enterprise architecture analysis with xml. In: Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS 2005). Volume 8., Los Alamitos, CA, USA, IEEE Computer Society Press (2005) 222b