

5-15-2012

FROM ENTERPRISE MODELLING TO ARCHITECTURE-DRIVEN IT MANAGEMENT ? A DESIGN THEORY

Jan Loehe

EBS Universität für Wirtschaft und Recht

Christine Legner

Faculty of Business and Economics (HEC), University of Lausanne

Follow this and additional works at: <http://aisel.aisnet.org/ecis2012>

Recommended Citation

Loehe, Jan and Legner, Christine, "FROM ENTERPRISE MODELLING TO ARCHITECTURE-DRIVEN IT MANAGEMENT ? A DESIGN THEORY" (2012). *ECIS 2012 Proceedings*. 232.

<http://aisel.aisnet.org/ecis2012/232>

This material is brought to you by the European Conference on Information Systems (ECIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ECIS 2012 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

FROM ENTERPRISE MODELLING TO ARCHITECTURE-DRIVEN IT MANAGEMENT – A DESIGN THEORY

Löhe, Jan, EBS Business School, Konrad-Adenauer-Ring 15, 65187 Wiesbaden, Germany,
jan.loehe@ebs.edu

Legner, Christine, Faculty of Business and Economics (HEC), University of Lausanne, 1015
Lausanne, Switzerland, christine.legner@unil.ch

Abstract

Enterprise architectures (EA) are considered promising approaches to reduce the complexities of growing information technology (IT) environments while keeping pace with an ever-changing business environment. However, the implementation of enterprise architecture management (EAM) has proven difficult in practice. Many EAM initiatives face severe challenges, as demonstrated by the low usage level of enterprise architecture documentation and enterprise architects' lack of authority regarding enforcing EAM standards and principles. These challenges motivate our research. Based on three field studies, we first analyze EAM implementation issues that arise when EAM is started as a dedicated and isolated initiative. Following a design-oriented paradigm, we then suggest a design theory for architecture-driven IT management (ADRIMA) that may guide organizations to successfully implement EAM. This theory summarizes prescriptive knowledge related to embedding EAM practices, artefacts and roles in the existing IT management processes and organization.

Keywords: Action research, Field study, Enterprise architecture management, IT management, IS design theory.

1 Introduction and motivation

Enterprise architectures (EAs) are considered promising approaches to align the required changes in corporate strategy and business processes with an increasingly complex IT landscape (Riege and Aier, 2009; Schmidt and Buxmann, 2011). The publication of the Zachman Framework (Zachman, 1987) in the 1980s initiated an extensive discourse on EA in research and practice (cf. Schelp and Winter, 2009; Schönherr, 2009): Early work focused on enterprises' fundamental components, their relationships as well as their appropriate representations (Aier et al., 2008b; Frank, 1995; Österle, 1995; Scheer, 1991). While this work centred on "what" in terms of enterprise modelling methods and notations, little attention was paid to the set-up and implementation of EA concepts in organizations (the "how"). In the meantime, enterprises have gathered practical experience with EA concepts: they started documenting their EA on different layers (Aier et al., 2008b; Lankhorst, 2005) and assigned responsibilities for the further EA development to dedicated architecture teams and roles, such as enterprise architects (Strano and Rehmani, 2007).

EAM initiatives are associated with substantial challenges, and often fail (Zink, 2009). Some of the criticism against EAM is that it requires a lot of effort (Morganwalp and Sage, 2004), that the benefits are not directly measurable and are realized with some time lag (Schmidt and Buxmann, 2011). Also, EA's strategic alignment potential is hindered by lacking governance, insufficient development support from business and IT management, as well as by inadequate skills (Seppanen et al., 2009; Winter and Schelp, 2008). This motivates our research, which will first analyze the key challenges when implementing EAM in practice. From three field studies, we conclude that EAM often suffers from being regarded as a separate and parallel initiative, although it needs to be embedded in established management processes and organization. Our research goal is to propose a design theory that may guide organizations to successfully implement EAM. Although we acknowledge EAM's broader enterprise focus, this article concentrates on EAM as an integral part of IT management. This focus is due to our research setting, which – similar to other empirical studies in the field (e.g. Schmidt and Buxmann, 2011) – involves IT-led EAM initiatives. We formulate our results as a design theory (Gregor and Jones, 2007; Walls et al., 1992), which we call the design theory for architecture-driven IT management (ADRIMA).

In the remainder of the paper, we first introduce our research methodology and process. Section 3 then presents observations from three field studies that allow us to analyze typical challenges and derive design requirements to address them. In section 4, we evaluate the current state of research. We then introduce a design theory for architecture-driven IT management. We conclude with a brief summary and a discussion of our contribution as well as the implications for future research.

2 Research methodology and process

The main purpose of our research is to propose a theory of design and action (Gregor, 2006) that will support companies implementing EAM. Our research result is a design theory that synthesizes prescriptive knowledge related to EAM implementation that is actionable, communicable and can be developed mutually (Gregor and Jones, 2007). Compared to natural and behavioural sciences' theories, design theories not only represent systems of statements targeted at describing, explaining and predicting real world phenomena (Bacharach, 1989; Dubin, 1976), but also denote sets of prescriptive statements to guide effective and feasible design (Walls et al., 1992). Accordingly, we aim to systemize and capture prescriptive knowledge to inform research and practice how to best construct artefacts intended to solve current EAM challenges (cf. Baskerville, 2008; Baskerville and Pries-Heje, 2010; Gregor, 2009). In this paper, we will concentrate on constructs, which represent domain-specific vocabulary and conceptualizations and a set of design principles, i.e. principles of form and function, to embed EAM in IT processes and the existing organizational structures.

We developed our design theory through a research process – inspired by the work of Hevner et al. (2004) and Hevner (2007) – covering relevance, rigour and a design cycle (Figure 1).

To identify relevant research questions and current EAM implementation in practice, the first step of our research comprised an interpretive field study (Klein and Myers, 1999). We observed the EAM initiatives of three large German companies for periods of between 10 and 18 months and worked in close cooperation with their EAM teams to develop and implement EAM. Owing to the organizations' size, their regional distribution and their dependence on the extensive use of IT, these companies were good candidates for EAM implementation. The selection of the companies was driven by purposeful sampling, i.e. their need for a holistic approach to EAM, as well as their willingness to cooperate and make multiple information sources available to researchers. In the course of the field study, we gained in-depth insight into the EA documentation and tools, the setup of the EA initiative and the methodologies applied. In addition to direct observation, we conducted semi-structured interviews and workshops to grasp the EAM implementation process and its challenges, thereby following a pluralistic research approach (Mingers, 2001). From the companies' specific EAM experiences, we were able to generalize typical implementation challenges through triangulation. We completed the first phase – the relevance cycle – by deriving design requirements to address these challenges.

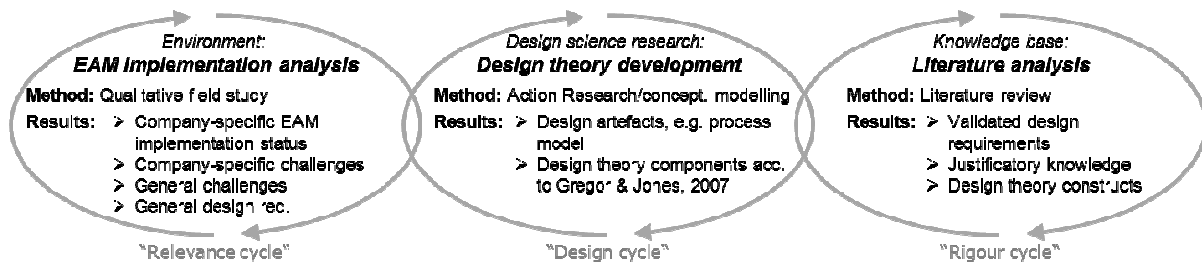


Figure 1. Research process according to Hevner (2007)

In the course of the rigour cycle, we compared our explorative insights to EAM literature. We found that EAM studies mention similar challenges – which acknowledged the validity of our design requirements – and used the literature to identify justificatory knowledge that supports our design theory.

During the design cycle, we developed and evaluated our design theory with the help of two of the three companies. In an iterative action research approach (Baskerville and Wood-Harper, 1998), we helped these companies solve their immediate EAM implementation challenges. We then generalized the findings as design theory by relying on the design and evaluation of purposeful artefacts, since these are regarded as the central basis of design theories (Gregor, 2009). We applied a consensus-oriented conceptual modelling approach (Becker and Niehaves, 2007) to construct and evaluate artefacts, such as IT process models and EA documentation. In periodic, one-to-four-week intervals we collaborated on-site with company representatives, notably enterprise architects. To improve the artefacts' content validity, we corroborated the different design alternatives with academic and practitioner literature, as well as with the archival documentation of our partners' IT units. In addition, executives, project managers and methodology experts with multi-year IT management experience validated the artefacts, for example, by means of process model walk-throughs, in different workshops. Their comments helped to continuously improve the artefacts' design until it was complete, consistent and viable for implementation in the companies. We recorded the comments, design changes and evaluation results by maintaining a research database. Based on these data, pattern matching and a comparison of the findings from multiple respondents, we synthesized the findings in a reference process model and formal design theory. In this paper, we will focus on design principles, because they represent a reconceptualization from a specific artefact to a class of solutions and capture formalized knowledge that can be reused to develop other instantiations. Applying our design theory, by means of expository instantiation, to different EAM implementations allows us to further validate and enhance it (Gregor and Jones, 2007).

3 Field studies on EAM implementation

In this section, we analyze EAM implementation and the associated challenges in three large German companies with a historically grown, complex IT landscape (Table 1). Despite several years of experience with EAM, all three companies were facing significant challenges which necessitated adjustments to their EAM approach. From our observations in the field, we were able to derive the typical challenges of and general design requirements for EAM implementation in organizations.

3.1 EAM implementation challenges in the field studies

Company A, a real estate business, is busy with its third attempt to implement EAM. Its first attempt was aborted in the initial modelling phase. Documenting the complete as-is EA was simply overwhelming in terms of level of effort and detail. A second attempt was started with the goal of capturing the EA on a higher level of abstraction according to the meta-model of the EAM tool planningIT. It resulted in a nearly complete as-is EA documentation, which quickly became obsolete: missing EAM roles meant it was not updated and it was not used in IT projects or for reports. Moreover, a great deal of detailed EA information, for example, in manuals, remained outside the EAM tool or with specific stakeholders, which meant the centrally collected EAM documentation was perceived of little use. Since the first two EAM initiatives occurred separately from the established IT processes in which the EA is regularly planned or changed, most IT and business representatives saw little benefit in these initiatives. In the third attempt, the company pursues a pragmatic approach which concentrates on concrete EA application scenarios in the IT operations' process. This approach defines EA models as well as appropriate analysis and reports to use for IT providers' contract management, inventory and document management, as well as for error handling, migration and maintenance processes.

Company B, an automobile manufacturer, applies EAM to harmonize its group-wide IT landscape. EAM is not driven in one top-down initiative, but by several individual initiatives in various IT organization units. In a first step, a group-wide core business process and organizational model was documented with the ARIS tool. This was complemented by technical infrastructure documentation, the definition of technology standards and their enforcement in project reviews. In addition, company B documented existing business applications and introduced a group-wide IT landscape planning methodology. More recently, the EAM gained further momentum from a SOA initiative in which documentation and modelling methods for domains, reusable IT modules and technical IT services had been developed. Nevertheless, the anticipated EAM benefits, such as improved business-IT alignment and reusable application services, failed to appear. The organization realized there was a gap between its long-term EA planning and the numerous development projects that constantly change the EA. Additionally, the distributed EAM approach led to scattered EA documentation. The various EAM tools provided redundant and low quality information and inconsistent documentation versions. Company-wide, standardized EA documentation had not yet been implemented, as it was seen as an overhead, and the EAM governance focused solely on enforcing technical standards. Motivated by the shortcomings, the organization intends to establish an end-to-end IT demand management process. By documenting and analysing demands based on EA models, the goal was to operationalize EAM, maintain up-to-date EA documentation and establish proactive IT planning.

The EAM in company C, an auditing firm, is driven by its global EAM strategy. A central EAM goal is to support the business strategy implementation, but the activities still focus mostly on IT architecture. The German subsidiary was allowed to decide how EAM had to be implemented and selected a model-oriented procedure. It developed a meta-model and reference models based on Federal Enterprise Architecture Framework (FEAF), covering all relevant areas from IT infrastructure assets to business processes. The initiative focused on assessing the existing application landscape, developing a catalogue of EA principles and standards, as well as establishing governance mechanisms to allow for long-term-oriented IT management. During realization it became apparent that the FEAF framework approach was not sufficient, not only because FEAF's reference models are oriented towards US

authorities, but also because it does not consider implementation in a firm's existing organizational structure. Furthermore, very little EA information was used in IT projects due to deficient EAM skills and a relatively slow central EA modelling process. The EAM team had only limited authority to direct the project managers who mainly drive the EA change. Project managers preferred familiar modelling approaches and usually showed resistance to the EAM team's additional documentation requirements. It became apparent that the company-wide EAM guidelines often lead to the project managers feeling constrained. Moreover, a lack of coordination between the EAM initiative and a parallel ITIL initiative created contradictory perceptions and redundant documentation. The required adjustments generated additional effort, which in turn decreased the EAM initiatives' acceptance. The EAM team is thus concentrating on monitoring and reporting on the application and IT service portfolio, as well as the IT vendor management process.

	Company A: Real estate	Company B: Automotive	Company C: Auditing
Scope	Subsidiary IT	Group IT	Subsidiary's business and IT
EAM organization	One central IT organization unit	Different IT organization units	One central IT organization unit
EAM approach	Modelling-driven, based on EAM tool	Driven by modelling, planning, method and technological standards	Modelling-driven, based on EA framework
EAM methods / EA framework	Method orientation towards EAM tool	Self-developed methods, partly based on EAM tool and TOGAF	Customized, based on FEAF and others, including TOGAF and ITIL
EA meta-model	One, based on EAM tool	Various, based on EAM tools	Customized, based on FEAF
EAM tool and repository	Central repository (Alfabet planningIT)	Several, central repositories (ARIS, Alfabet planningIT, Sparx EA)	Central repository (ARIS with own extension)
EA application scenarios (planned)	IT operations: Contract management; inventory and document management; IT maintenance and migration management	Business process and information management; demand management; application and service portfolio management; IT standards & technology management; IT landscape planning	Business process and information management; application and service portfolio management; IT standards and technology management; IT vendor management; IT landscape planning
EAM implementation challenges	High initial data collection effort; separated from established processes; unused and obsolete documentation; unclear responsibilities and application scenarios	Limited top-down coordination; insufficient horizontal coordination; separated from established processes; technical focus; obsolete and inadequate documentation; low user acceptance and comprehension; missing EAM authority	Limited top-down coordination; insufficient horizontal coordination; FEAF unsuitable for existing structures; separated from established processes; unused documentation; low user acceptance; missing EAM skills; missing EAM authority

Table 1. EAM approaches in three companies

3.2 General EAM implementation challenges and design requirements

All three companies initially focussed on a modelling-driven EAM approach, as proposed by many EA frameworks. They started by documenting their as-is architecture on different EA layers, but faced common challenges with the modelling-driven EAM implementation. The first challenge relates to the *effort regarding the initial documentation* of the EA models and the definition of EA standards by the architecture teams: The companies recognized that complete EA documentation was not feasible, due to the many different stakeholders, the overall organizational complexity and the too large scope. The high initial effort hampered the willingness to further maintain the EA artefacts. In addition, the companies did not consider mechanisms to update the EA documentation, such as after-project changes. Thus, the EA repositories became rapidly outdated and were perceived to be of low quality. A second challenge was that *existing EA artefacts remained unused* in daily work and decision-making: The low utilization of the EA documentation was partly due to its poor quality and obsolescence. Additionally, the EAM initiatives ignored the stakeholders' information needs, and provided EA documentation at the wrong levels of granularity. Indeed, all the EAM initiatives provided basic EA documentation, but without the appropriate EA artefacts' representation, for example in reports, to support the day-to-day

work and decision-making. A third challenge was the *lack of acceptance* in the (IT) organization: The IT employees thought EAM just required additional effort, and had no benefits for their own work. The EAM initiatives' unclear goals and the lacking EAM knowledge hampered their support. A lack of EAM comprehension resulted, also because the EAM was often perceived as having only a technological focus. In addition, the architects had limited access to the IT decision-making committees (e.g., project steering committees, change advisory boards). Consequently, they were unable to promote and enforce EA policies and standards in the existing IT processes, i.e. the planning, building and running of IT. Since the EAM was set up as an independent initiative with a focus on the EA life cycle, it created a management cycle parallel to established IT processes, such as the IT strategy definition, budget and portfolio planning, the IT project delivery and IT service management. The consequences were *coordination problems and rivalry between the EAM initiative and IT processes*, which already shaped decisions related to planning, changing and managing the EA. From these common EAM challenges, we are able to derive four general design requirements (Table 2). These requirements describe, firstly the conditions needed from a practitioner's viewpoint to solve problems, secondly the specification or constraint of the design (cf. IEEE Std 610.12-1990). The requirements emphasize embedding EAM in established IT processes and organization, instead of setting it up as a separate and parallel initiative.

No.	EAM implementation challenges	Design requirements (DR)
1	Great effort relating to the initial collection as well as EA artefacts being outdated and of low quality	Existing IT processes should produce and maintain EA artefacts continuously
2	Low usage of existing EA artefacts in the day-to-day work and decision-making	Existing IT processes should consume and use appropriate EA artefacts
3	Lack of EAM acceptance in the (IT) organization and difficulties to enforce EA policies and standards	Existing IT roles and committees should be aware regarding EAM-specific information and tasks
4	Coordination problems because the EAM initiative sets up processes for managing the EA life cycle parallel to established IT processes	EAM should be embedded into existing organizational structure and processes

Table 2. General EAM implementation challenges and design requirements

4 Current state of research and related work

In this section, we contrast the design requirements of real-world EAM implementations with the existing literature. Table 3 depicts how prior research conceptualizes EAM implementation and assesses the degree to which it satisfies the design requirements. From our extensive literature review and qualitative content analysis (Mayring, 2000), we find that prior work describes EAM implementation in terms of 1) EAM initiatives' building blocks, 2) processes for using and introducing EA, 3) new organizations, roles and governance regimes, as well as 4) EAM application scenarios.

The first cluster ("I" in Table 3) includes established EA frameworks and the literature (e.g. Bricknall et al., 2006) that provide guidelines for EAM initiatives. TOGAF (The Open Group, 2009) suggests an architecture development method (ADM) that starts with an architecture vision. It continues to business, information and infrastructure architectures, opportunities and solutions, migration planning, implementation governance, and finally architecture change management. FEAF advises a top-down EAM implementation for segment architectures. It contains a general EA process (CIO Council, 2001) covering the EA program initiation, the EA development and use with baseline and target architectures, as well as the EA maintenance and control. It focusses on aligning EA with capital planning and investment control, and with system development processes for US authorities. Bricknall et al. (2006) identify critical factors for EAM initiatives: Among others, they refer to top management buy-in, implementation of an EA governance process and EAM's alignment with other enterprise life cycle processes, such as the investment process. Process-driven approaches ("P" in Table 3) provide descriptions of typical EAM processes and activities on a macro (e.g. Keller, 2007) or micro level (e.g. Dern, 2007). With regard to integrating EAM into IT processes, Keller (2007) suggests a high-level comparison matrix between EAM and COBIT. Dern (2007) provides a detailed description of how EAM

should be embedded in IS portfolio planning and software development, without giving directions on how to embed EAM in other IT processes. Other work (e.g. Buckl et al., 2009; Buckl et al., 2010) focuses on individual EAM activities, for example, describing and developing target state EA as well as analysing and evaluating EA, without illustrating how they are embedded in existing processes. Wittenburg et al. (2007) consider EAM the glue between IT management-related functions, such as IT planning and project life cycle management. While this illustration nearly satisfies all design requirements, it does not consider IT roles and committees, or contain prescriptions regarding other processes. Another stream of research emphasizes the application of EAM to address specific stakeholder concerns or application areas (“A” in Table 3). Accordingly, these works represent exemplary EAM usage, for example, EA impact or coverage analyzes to support risk management and sourcing decisions (e.g. Aier et al., 2008a; Bucher et al., 2006). In line with this, EAM patterns (e.g. Buckl et al., 2008) relate EA artefacts and methods to a particular stakeholder’s perspective. These approaches focus on EAM best-practice solutions for specific needs, without providing an implementation method. To address this gap, Buckl et al. (2011) recently suggested a situational EAM design for selecting appropriate EAM patterns. Finally, governance-driven approaches (“G” in Table 3) suggest that EA bodies and roles should be introduced as they are key to EAM implementation (e.g. Boh and Yellin, 2006; Schmidt and Buxmann, 2011). These approaches concentrate on EA policies as well as EA roles and committees, but do not specify their interplay with the existing organization.

To conclude, none of the existing approaches completely satisfied our design requirements. With respect to DR1, only a few contributions (e.g. Bricknall et al., 2006; CIO Council, 2001) describe a continuous maintenance concept for EA documentation, while others (e.g. Aier et al., 2009; Schmidt and Buxmann, 2011) at best mention its significance. With regard to the consumption and usage of appropriate EA artefacts, DR2 is addressed by several approaches. Only Aier et al. (2009) specifically define criteria for adequate EA documentation. DR3 is partly covered in governance-driven work (e.g. Boh and Yellin, 2006) as well as in practitioner-driven contributions and EA frameworks, but without considering the interplay with the existing organization. The contention that EAM should be integrated into established organizational structures is prominent, but mostly remains a generic assertion. Some contributions that satisfy DR2 also build on DR4 by presenting specific IT processes with integrated EAM practices and EA artefacts. However, as they have no general prescriptions that EAM should be embedded in other EA application scenarios, their process model and artefact knowledge can only be used to derive general design principles for a specific class of solutions.

EAM literature		EAM implementation		Mentioned (IT) process examples / main EAM application scenarios	Design req.			
		Governance	Process		1	2	3	4
Bricknall et al. (2006)	C	+ (R/Go/S)	o (E) / o (In)	Investment, software development	+	+	-	o
FEAF (2001)	I	+ (O/R/Go/S)	+ (E) / o (In)	Capital planning and control, software dev.	+	+	+	o
TOGAF (2009)	I	o (R) / + (Go)	+ (E)	Enterprise transformation	o	o	o	-
Buckl et al. (2009), (2010)	P	-	+ (E)	Portfolio, demand, project life cycle mgmt.	o	o	-	-
Dern (2007)	P	+ (O) / + (R)	o (E)* / + (In)	IS Planning, systems/software develop.	o	+	+	+
Keller (2007)	P	+ (O) / o (Go)	+ (E)	IT strategy, appl. portfolio management	o	+	+	-
Wittenburg et al. (2007)	P	-	o (E)**	Portfolio, demand, project life cycle mgmt.	+	+	-	o
Aier et al. (2008a), (2009)	A	o (O) / o (R)	o (E) / o (In)	Bus. proc. optimization, consolidation	o	+	-	-
Bucher et al. (2006)	A	-	o (E)	Bus. continuity planning, risk mgmt.	-	+	-	o
Buckl et al. (2008), (2011)	A	o (O)	+ (E) / o (In)	Portfolio mgmt. (<i>respect. EAM patterns</i>)	o	o	-	-
Boh & Yellin (2006)	G	+ (O/R/S)	o (E)	Technical choices in application design	-	+	+	-
Ross (2003)	G	+ (S/Go/O)	o (E) / o (In)	Business/IT alignment	o	-	-	o
Schmidt & Buxmann (2011)	G	+ (O) / o (R)	+ (E)	System development, IS planning	o	o	+	o

(C)luster: (A)pplication scenario, (G)overnance, (I)nitiative, (P)rocess
EAM implementation: (Go) Standards and principles, (O) Organizational structure/committees, (R) Roles, (In) Introduction process, (E) EAM process, (S) Stakeholder’s buy-in/agreements
Design factor/design requirement is addressed: (-) not, (o) partly/generically/exemplarily, (+) comprehensively
* Focus is on IT portfolio planning and software development process / ** Focus is on IT management processes

Table 3. EAM implementation in literature (excerpt)

5 A design theory for architecture-driven IT management (ADRIMA)

The design theory for architecture-driven IT management (ADRIMA) is a formal representation of our research findings. Although design science has become a popular stream in IS research, there is as yet no commonly accepted way of designing and documenting design theories. For the purpose of this study, we adopt the recommendations by Gregor and Jones (2007), who propose eight documentation components. Owing to space limitations, we will concentrate on the design theory's *constructs* and its *principles of form and function*, represented by five design principles (DP) and supported with *justificatory knowledge* and *testable propositions*. The *purpose and scope* are already defined by the four design requirements, which specify what the theory is for and what its boundaries are. In addition, we briefly illustrate the *expository instantiation* with an IT demand management process example and consider the *principles of implementation* and *artefact mutability* as future research.

5.1 ADRIMA's constructs

The *constructs* represent the theory's entities of interest and their relationships in the sense of the "causa materialis" (Gregor and Jones, 2007). Figure 2 illustrates the main constructs, i.e. the basic vocabulary (March and Smith, 1995) that enables our design theory's communication and description, in the form of a conceptual model.

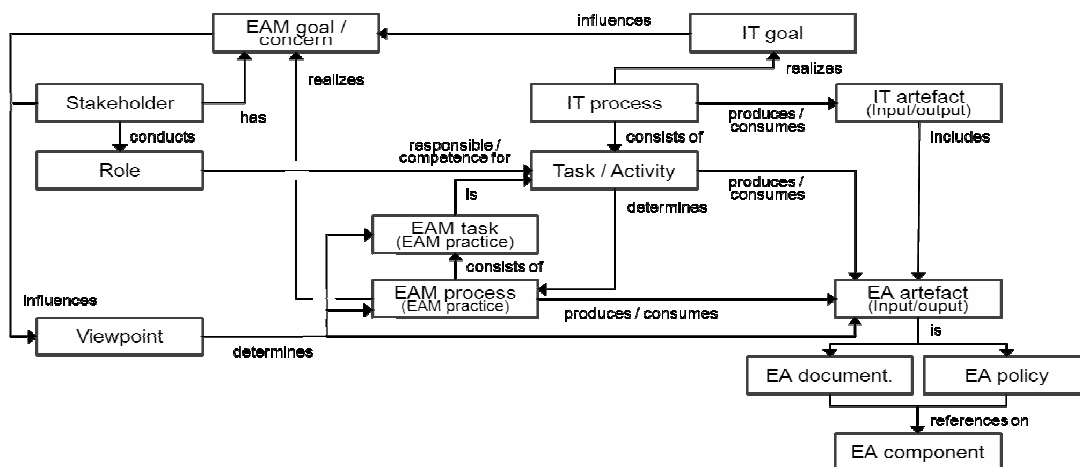


Figure 2. ADRIMA's constructs in a conceptual model

Since our design theory embeds EAM in IT processes, the model depicts how IT management and EAM constructs are interrelated. On the one hand, it represents typical IT processes, tasks / activities, IT artefacts, IT goals, as well as the stakeholders and roles, along with their relationships. On the other hand, it illustrates how these are supported by EAM artefacts and practices. According to van der Raadt and van Vliet (2008), EA artefacts comprise EA documentation and EA policies. EA documentation describes a current (as-is) or future (to-be) EA state for a defined set of concerns and viewpoints based on the fundamental EA components (Aier et al., 2009; Buckl et al., 2008; Maier et al., 2004), while EA policies are used to direct and control the EA. Since the literature uses different terminologies for EA/EAM-related activities, such as EA/EAM task, method, application, function and process, we subsume them under the term EAM practice. Typical EAM practices (cf. Bucher et al., 2006; Buckl et al., 2010; Schmidt and Buxmann, 2011; van der Raadt and van Vliet, 2008) include EAM (1) documentation (e.g., document, update, share an EA artefact); (2) usage (e.g., analyze, evaluate EA artefacts or comply with EA policy); (3) governance (e.g., monitor, direct, enact an EA policy, or monitor, coordinate, further develop EAM practices and EA artefacts, or handle EAM escalation); as well as (4) communication (e.g., communicate, provide feedback on EAM practices and EA artefacts).

5.2 ADRIMA's principles of form and function ("design principles")

DP1. Complement established IT processes with EAM practices and EA artefacts: Established IT processes that govern (direct, monitor,) and manage (plan, develop, operate) substantial EA components and relationships should be complemented by EAM practices and EA artefacts.

The first design principle responds directly to DR4 and the need to generally embed EAM in the IT processes that have a significant impact on the EA. Established IT processes must build the foundation on which to embed EAM practices and artefacts to better fulfil IT goals without reshaping the EAM as another management process cycle (e.g. Buckl et al., 2009; The Open Group, 2009). This implies that EAM practices complement IT tasks. We compare EAM as a management function with knowledge management (KM). Both EAM and KM need to have implicit and explicit knowledge effectively identified, documented and communicated in enterprise-level management processes (cf. Riempp, 2004; Struck et al., 2010). Recent empirical EAM studies confirm this need in respect of EAM implementation in existing structures (e.g. Schmidt and Buxmann, 2011). We suggest following testable propositions: (a) If the EAM practices and artefacts are integrated well into existing IT processes, EAM is more likely to succeed. (b) If the EAM establishes parallel processes, it is more likely to fail.

DP2. Integrate EAM practices horizontally or vertically with specific IT tasks:

- a. EAM usage and documentation practices that fit specific task categories should be horizontally embedded in an established IT process.
- b. EAM communication and governance practices that fit specific task categories should be vertically embedded in an established IT process.

These principles aim to satisfy DR1, DR2 and DR4, and define where and which kind of EAM practices to embed. Moreover, situational factors have been found to determine the application of EAM practices and act as preconditions to determine the application of the most suitable EAM practices (Buckl et al., 2011). From our field study, we have found existing IT tasks as a determining situational factor. During our action research process, we learned that task categories are associated with appropriate EAM practices: IT tasks related to (1) analysis, planning and decision-making should be supplemented by EAM usage or documentation practices, (2) communication and marketing should be supplemented by EAM communication practices, and (3) secure compliance and check quality should be supplemented by EAM governance practices. Correspondingly, an appropriate EAM practice should be embedded in an existing IT process (cf. Riempp, 2004):

- An EAM usage or documentation practice should be horizontally embedded, which means that it directly produces or consumes an EA artefact as a task of the existing IT process.
- An EAM communication or governance practice should be vertically embedded, which means an IT task initiates the production or consumption of an EAM artefact of another EAM process.

For DP2, we suggest subsequent testable propositions: (a) If the applied EAM practices fit the IT tasks, they will increase the IT processes' efficiency and quality; (b) If the EAM usage or documentation is not embedded horizontally in IT processes, they will be less effective.

DP3. Trigger EAM maintenance continuously by means of an EA components' change: Whenever an EA component changes, an IT process's feedback mechanism that continuously updates the EA information base should be triggered to secure the EA information's timeliness and quality.

This principle addresses DR1 and prescribes the conditions and mechanisms to ensure continuous maintenance of EA artefacts. A feedback mechanism should be triggered when an EA component changes to maintain an updated EA documentation. The EA documentation has to be maintained, i.e. created or modified, validated and released, for example, (1) after a new IT planning round defines the to-be architecture; (2) after an IT governance process releases a new or modified EA policy; and (3) when the as-is architecture is changed, i.e. when a EA component or relationship is created or modified. Beside a defined trigger, the feedback mechanism should also contain an interface specification

for the consistent transfer from existing specialized repositories, such as a configuration management database, to the EA repository. This can, for example, be achieved by means of meta-model integration (cf. Fischer et al., 2007). In respect of DP3, we propose subsequent testable propositions: (a) If an IT process contains triggers for continuous maintenance, the quality of the EA documentation will increase; (b) If IT process-specific documentation is consistently transferred into appropriate EA documentation, the EA documentation's up-to-dateness /quality will increase; (c) The effort to ensure triggered, continuous EAM maintenance will be less than that required for various, single EAM initiatives to attain a similar EA information base quality.

DP4. Consider appropriate EA artefacts' granularity: An EA artefact should represent a defined set of stakeholders' viewpoints and satisfy the quality criteria of width, depth and pragmatism.

This principle addresses both DR1 and DR2, and supports the application of EA artefacts in existing IT processes (cf. DP2). As a fully covered EA information base is not feasible, appropriate EA artefacts should focus on major dependencies on a high level of abstraction. An EA artefact forms an EA-specific in-/output that can be part of an IT artefact. Building on Aier et al. (2009), three quality criteria should guide the definition and collection of appropriate EA artefacts:

- The width criterion specifies that an EA information base should only contain EA artefacts necessary to address stakeholders' viewpoints, which are defined by IT tasks and processes.
- The depth criterion specifies that only holistic structures, i.e. those that reflect the entire organization or a group of similar components, are relevant. Detailed information is only relevant if an EA component's change has a significant impact on other components or the details foster its reuse.
- According to the pragmatism criterion, the effort needed for continuous maintenance (cf. DP3) should be less than the benefits relating to the use of the EA artefacts. This implies that EAM usage and maintenance should use appropriate cost and benefit measures.

For DP4, we suggest the subsequent testable propositions: (a) If EA artefacts fit the viewpoints, they will increase IT processes' quality and degree of EAM usage; (b) The EA information base will contain fewer unnecessary EA artefacts if the criteria of width, depth and pragmatism are applied.

DP5. Embed EAM responsibilities and competencies in the established IT organization: IT roles and decision-making committees should be complemented by EAM-specific responsibilities and skills.

This design principle addresses DR3 and complements existing approaches focusing on the role of the architect and architecture boards (e.g. Keller, 2007; Strano and Rehmani, 2007; The Open Group, 2009). However, our field study observations, as well as recent studies (e.g. Schmidt and Buxmann, 2011; Winter and Schelp, 2008), reveal that architecture topics have to be covered and become part of established IT roles and committees, such as project steering or investment committees. Architects have to serve on these committees to support decision-making. The procedures and rules of existing committees must be complemented by EAM feedback, as well as EAM decision-making rights, for example, to handle EA waiver requests or escalate to another committee (cf. van der Raadt and van Vliet, 2008). Moreover, existing roles must be responsible for executing individual EAM tasks, such as updating EA documentation or escalating conflicts with EA standards, and have the appropriate skills to do so. In respect of DP5, we suggest the following testable propositions: (a) If existing IT role profiles are complemented by EAM-specific responsibilities and skills, it is more likely that EA-related responsibilities will be carried out; (b) If architects serve on existing IT committees, the enforcement of EA standards and principles will increase.

5.3 Expository instantiation in the IT demand management process

During our design cycle, we were able to apply our design theory to design a new IT demand management process. In the following, we illustrate this expository instantiation by describing how the design principles were applied: (DP1) The IT demand management process is a planning IT process that shapes the future EA; it should consequently be supplemented by EAM practices and EA artefacts. (DP2) A demand manager assesses new demands (i.e. horizontal integration) on the basis of EA

impact analyzes. An architecture team monitors (i.e. vertical integration) whether the suggested solution is specified by an appropriate EA model, for example, a business function model. (DP3) At go-live (i.e. the trigger), the EA artefacts are updated. (DP4) A to-be business function model is used to document a demand's solution alternative from a business-stakeholder perspective. (DP5) An enterprise architect with a veto right is a member of the demand management committee. He participates in the demand evaluation and prioritization to avoid conflicts with the target EA and EA goals.

6 Discussion and conclusion

This paper provides an explorative analysis of real-world challenges in EAM implementation as well as the design requirements to address them. Its main scientific contribution is a design theory for architecture-driven IT management which comprises five design principles for embedding EAM in the IT organization and processes, which plan, change and manage the EA. By focusing on design principles as well as by emphasizing the EAM embedded in existing IT processes, our research goes beyond previous approaches from science and practice: Existing EA literature and EA frameworks describe EAM as a stand-alone management concept focussing on the EA life cycle. Conversely, existing IT management approaches, such as ITIL or COBIT, do not (yet) or only generically consider EAM. Our research goes beyond recent work that deals with situational EAM introduction (Buckl et al., 2009; Buckl et al., 2010; Buckl et al., 2011), as it integrates EAM with IT management. Hence, our approach is an additional step to consolidate both IT management and EAM perspectives. The formulation of our research findings as design theory goes beyond the existing EAM application scenarios and patterns, since a design theory can be used to derive different instantiations and to identify new EAM patterns. Our research has certain limitations and currently only mirrors an IT-driven EAM perspective. However, we consider EAM a holistic organizational approach and anticipate that the design theory could be extended to embedding EAM in other management structures and processes. Moreover, the design theory has only been instantiated in one organization so far. More instantiations, especially processes other than IT ones, will refine and complement our theory. A further limitation is that the implementation and validation of the process model were carried out by means of action research, which bears the risk of focusing too much on company-specific results that are not generalizable. At the same time, design-oriented action research in an organization is a recommended research approach for the iterative development of design theory results (cf. Siponen et al., 2006; Walls et al., 1992). Our work's implication for practice is that organizations should not introduce EAM in a separate initiative, but should aim at changing established processes to become EAM-aware. As such, organizations can more transparently communicate the value through EAM usage and avoid stakeholder resistance. We suggest that future research should examine the integration of EAM practices and EA artefacts with established organizational practices and artefacts. Future research should also address the further refinement of our design theory in keeping with the work by Gregor and Jones (2007): With regard to artefact mutability, we anticipate differences between the instantiation of processes, whose process description can be formalized in advance, and emergent processes with little formalized procedures. Future research should also address the design theory's instantiation and identify effective principles of implementation. Process modelling is a recommended approach to ensure instantiations and might also result in the development of reference models for architecture-driven IT management processes.

References

- Aier, S., Kurpjuweit, S., Saat, J. and Winter, R. (2009). Enterprise Architecture Design as an Engineering Discipline. *AIS Transactions on Enterprise Systems*, 1 (1), 36-43.
- Aier, S., Riege, C. and Winter, R. (2008a). Classification of Enterprise Architecture - An Exploratory Analysis. *International Journal of Enterprise Modelling and Information Systems Architectures*, 3 (1), 14-23.
- Aier, S., Riege, C. and Winter, R. (2008b). Unternehmensarchitektur – Literaturüberblick und Stand der Praxis. *Wirtschaftsinformatik*, 50 (4), 292-304.
- Bacharach, S.B. (1989). Organizational Theories: Some Criteria for Evaluation. *Academy of Management Review*, 14 (4), 496-515.

- Baskerville, R. (2008). What design science is not. *European Journal of Information Systems*, 17 (5), 441-443.
- Baskerville, R. and Pries-Heje, J. (2010). Explanatory Design Theory. *Business & Information Systems Engineering*, 2 (5), 271-282.
- Baskerville, R. and Wood-Harper, T. (1998). Diversity in information systems action research methods. *European Journal of Information Systems*, 7, 90-107.
- Becker, J. and Niehaves, B. (2007). Epistemological perspectives on IS research: a framework for analysing and systemizing epistemological assumptions. *Information Systems Journal*, 17 (2), 197-214.
- Boh, W.F. and Yellin, D. (2006). Using Enterprise Architecture Standards in Managing Information Technology. *Journal of Management Information Systems*, 23 (3), 163-207.
- Bricknall, R., Darrell, G., Nilsson, H. and Pessi, K. (2006). Enterprise architecture: critical factors affecting modelling and management. In *Proceedings of the Fourteenth European Conference on Information Systems*, pp. 2349-2361, Goteborg.
- Bucher, T., Fischer, R., Kurpjuweit, S. and Winter, R. (2006). Enterprise Architecture Analysis and Application – An Exploratory Study. In *Proceedings of the 10th IEEE International Enterprise Distributed Object Computing Conference Workshops (EDOCW 2006)*, Hong Kong, China.
- Buckl, S., Ernst, A.M., Lankes, J. and Matthes, F. (2008). Enterprise Architecture Management Pattern Catalog (Version 1.0, February 2008). sebis TU München, Munich.
- Buckl, S., Matthes, F. and Schweda, C.M. (2009). A Viable System Perspective on Enterprise Architecture Management. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, 1483 - 1488, San Antonio, TX.
- Buckl, S., Matthes, F. and Schweda, C.M. (2010). Towards a Method Framework for Enterprise Architecture Management. In *Proceedings of the CAiSE 2010 Workshop BUSITAL'10*, pp. 46-60, Hammamet, Tunisia.
- Buckl, S., Matthes, F. and Schweda, C.M. (2011). A Method Base for EA Management. In *Proceedings of the IFIP Working Conference on Method Engineering (ME 2011)*, Paris, France.
- CIO Council. (2001). A Practical Guide to Federal Enterprise Architecture.
- Dern, G. (2007). Management von IT-Architekturen: Informationssysteme im Fokus von Architekturplanung und -entwicklung. Vieweg, Wiesbaden.
- Dubin, R. (1976). Theory Building In Applied Areas. In *Handbook of Industrial and Organizational Psychology*, (M. Dunette ed.), Rand McNally, 17-38.
- Fischer, R., Aier, S. and Winter, R. (2007). A Federated Approach to Enterprise Architecture Model Maintenance. *Enterprise Modelling and Information Systems Architectures*, 2 (2), 14-22.
- Frank, U. (1995). MEMO: Eine werkzeuggestützte Methode zum integrierten Entwurf von Geschäftsprozessen und Informationssystemen. In: *Proceedings of Wirtschaftsinformatik 1995*, 67-82.
- Gregor, S. (2006). The Nature of Theory in Information Systems. *MIS Quarterly*, 30 (3), 611-642.
- Gregor, S. (2009). Building Theory in the Sciences of the Artificial. In *Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology (DESRIST 2009)*, Philadelphia, PA, USA.
- Gregor, S. and Jones, D. (2007). The Anatomy of a Design Theory. *Journal of the Association for Information Systems*, 8 (5), 313-335.
- Hevner, A.R. (2007). A Three Cycle View of Design Science Research. *Scandinavian Journal of Information Systems*, 19 (2), Article 4.
- Hevner, A.R., March, S.T., Park, J. and Ram, S. (2004). Design Science in Information Systems Research. *MIS Quarterly*, 28 (1), 75-105.
- Keller, W. (2007). IT-Unternehmensarchitektur. 1st Edition. dpunkt.verlag, Heidelberg.
- Klein, H.K. and Myers, M.D. (1999). A set of principles for conducting and evaluating interpretive field studies in information systems. *MIS Quarterly*, 23 (1), 67-94.
- Lankhorst, M.M. (2005). Enterprise Architecture at Work. Springer, Berlin.
- Maier, M.W., Emery, D. and Hilliard, R. (2004). ANSI/IEEE 1471. *Systems Engineering*, 7 (3), 257-270.
- March, S.T. and Smith, G.F. (1995). Design and natural science research on information technology. *Decision Support Systems*, 15 (4), 251-266.
- Mayring, P. (2000). Qualitative Content Analysis. *Forum: Qualitative Social Research*, 1 (2).
- Mingers, J. (2001). Combining IS Research Methods: Towards a Pluralist Methodology. *Information Systems Research*, 12 (3), 240-259.
- Morganwalp, J.M. and Sage, A.P. (2004). Enterprise architecture measures of effectiveness. *International Journal of Technology, Policy & Management*, 4 (1), 81-94.
- Österle, H. (1995). Business in the Information Age - Heading for New Processes. Springer, Berlin.
- Riege, C. and Aier, S. (2009). A Contingency Approach to Enterprise Architecture Method Engineering. *Journal of Enterprise Architecture*, 5 (1), 36-48.
- Riempp, G. (2004). Integrierte Wissensmanagement-Systeme - Architektur und praktische Anwendung. Springer, Berlin.
- Ross, J.W. (2003). Creating a Strategic IT Architecture Competency: Learning in Stages. *MIS Quarterly Executive*, 2 (1), 31-43.
- Scheer, A.-W. (1991). Architektur integrierter Informationssysteme - Grundlagen der Unternehmensmodellierung. Springer, Berlin.
- Schelp, J. and Winter, R. (2009). Language communities in enterprise architecture research. In: *Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology*, ACM, New York, NY, USA.
- Schmidt, C. and Buxmann, P. (2011). Outcomes and success factors of enterprise IT architecture management: empirical insight from the international financial services industry. *European Journal of Information Systems*, 18 (1), 168-185.

- Schönherr, M. (2009). Towards a Common Terminology in the Discipline of Enterprise Architecture. In *Service-Oriented Computing – ICSOC 2008 Workshops*, 400–413.
- Seppanen, V., Heikkila, J. and Liimatainen, K. (2009). Key Issues in EA-Implementation: Case Study of Two Finnish Government Agencies. In *Proceedings of the IEEE Conference on Commerce and Enterprise Computing*, 114-120.
- Siponen, M., Baskerville, R. and Heikka, J. (2006). A Design Theory for Secure Information Systems Design Methods. *Journal of the Association for Information Systems*, 7 (11), 725-770.
- Strano, C. and Rehmani, Q. (2007). The role of the enterprise architect. *Information Systems & e-Business Management*, 5 (4), 379-396.
- Struck, V., Buckl, S., Matthes, F. and Schweda, C.M. (2010). Enterprise Architecture Management From a Knowledge Management Perspective - Results from an Empirical Study. In *Proceedings of the Mediterranean Conference on Information Systems (MCIS)*.
- The Open Group. (2009). *TOGAF Version 9 - The Open Group Architecture Framework*. The Open Group, Zaltbommel.
- van der Raadt, B. and van Vliet, H. (2008). Designing the Enterprise Architecture Function. In *Proceedings of the 4th International Conference on the Quality of Software Architectures (QoSA2008)*, 103-118.
- Walls, J.G., Widmeyer, G.R. and El Sawy, O.A. (1992). Building an Information System Design Theory for Vigilant EIS. *Information Systems Research*, 3 (1), 36-59.
- Winter, R. and Schelp, J. (2008). Enterprise architecture governance: the need for a business-to-IT approach. In *Proceedings of the 2008 ACM symposium on Applied computing*, pp. 548-552, ACM, Fortaleza, Ceara, Brazil.
- Wittenburg, A., Matthes, F., Fischer, F. and Hallermeister, T. (2007). Building an integrated IT governance platform at the BMW Group. *International Journal of Business Process Integration and Management*, 2 (4), 327-337.
- Zachman, J.A. (1987). A Framework for Information Systems Architecture. *IBM Systems Journal*, 26 (3), 454-470.
- Zink, G. (2009). How to Restart an Enterprise Architecture Program After Initial Failure. *Journal of Enterprise Architecture*, May 2009, 31-41.