Using an extended structure to document new Enterprise Architecture Management patterns

ÖMER ULUDAG, Technische Universität München (TUM)

By guiding the continuous development and transformation of an enterprise, the enterprise architecture (EA) management function is expected to provide companies business value. Hence, the importance of the EA management function has been growing in organizations. Since the design of an EA management function is not an easy task, various frameworks and EA management tools are awaited to deliver guidance for performing the EA management function. Unfortunately, the provided approaches are either too abstract to enable realization support or far too generic neglecting enterprise-specific EA related concerns. As a consequence, the pattern-based approach to EAM has been developed. Since the initial publication of the EA management pattern catalog (EAMPC) in 2007, the EA management body-of-knowledge has grown significantly. Consequently, an extension of the EA management pattern language in terms of new EA management patterns, EA management pattern related elements, and new relationship types is required. This provides researchers and practitioners an access point for yet not accessible knowledge within the context of the EA management function.

1. INTRODUCTION

Nowadays' chief executive officers regard the effective management and utilization of information through information technology (IT) as a key success factor to business effectiveness and as an indispensable driver to achieving competitive advantage [The Open Group 2013b]. For providing the aforementioned benefits, an establishment of alignment between business and IT is required, which represents a major challenge for IT managers [Reich and Benbasat 2000]. The enterprise architecture (EA) management function addresses this need by bridging the gap between business and IT [Winter and Fischer 2007]. The most widely used definition of EA is based on the ISO/IEC 42010 definition of architecture provided by The Open Group Architecture Framework (TOGAF): "*Fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution.*" The EA management function deals with the establishment and continuous evolution of EA [Aier et al. 2011]. Since the topic of the EA management function has entered mainstream interest, different frameworks have been created, like TOGAF [The Open Group 2013b], Zachman framework [Zachman 1987] or Federal Enterprise Architecture Framework [Chief Information Officers Council 1999, Buckl et al. 2007]. These frameworks promise to provide guidance for the establishment and execution of the EA management function, but cannot provide guidance without reducing specificity of their approach. In order to overcome the problem with the generality of these frameworks, the pattern-based approach to EA management can be used to complement them and make them less generic [Buckl et al. 2009b]. Patterns are general, reusable solutions to a common problem, which are dependent on their context. This idea of patterns was firstly introduced in the field of architecture by [Alexander et al. 1977]. Later this idea was adopted in the area of software engineering and software architecture [Gamma et al. 1994, Buschmann et al. 1996]. Based on the previous described properties of patterns, EA management patterns describe solutions, which are observed practices for recurring problems in the context of the EA management that can be adapted to a specific enterprise context [Ernst 2008]. In order to retrieve patterns from practice on a scientific basis, the pattern-based design research (PDR) was used, which led to the EA management pattern catalog (EAMPC) that has been published in 2008 [Buckl et al. 2008b, Buckl et al. 2013]. Due to this initial publication of the EAMPC, the EA management body-of-knowledge has grown significantly. This development calls for an evolution of the existing EA management pattern language. The extension contains the identification of the previous mentioned new elements, and the further development of the structure and content of the EA management pattern language. Finally, this extension should be published in the second version of the EAMPC, in order to make new EA related knowledge accessible to researchers and practitioners.

This paper follows the PDR method and describes the extension of the EA management pattern language in terms of new EA management patterns, EA management pattern related elements and new relationship types. For that reason, Section 2 gives an overview of the PDR approach. Additionally, the EA management pattern approach is presented and an appeal for the further extension is stated. In a last step, the extension of the EA management pattern language is described. Initially, Section 3 describes required steps in the research approach. The outcome of the research is presented in Section 4. Exemplified documentations for new EA management patterns and EA management pattern related elements are demonstrated in Section 5. Finally, Section 6 concludes the paper and sketches future directions of research.

2. RESEARCH FUNDAMENTALS

Since the EA management function forms an interesting research subject of the information systems (IS) discipline, practitioners are also highly interested in this field [Buckl et al. 2009b]. However, researchers in the area of IS applying the design science paradigm are confronted with the challenge to make scientific contributions which also target to solve current and anticipated problem in practice. This problem is often referred to as the *rigor* and *relevance* challenge of design science research. In order to ensure the relevance of the research outcome, the PDR is proposed by [Buckl et al. 2013]. This paper follows the PDR method in order to ensure relevance research outcome for practice. Thus, the PDR method is briefly described in the following.

2.1 Pattern-Based Design Research

IS researchers, who practice design science, target the creation of a novel artifact, for instance a solution to a relevant problem. However, researchers have to take into account two major criteria while creating a solution: *rigor* and *relevance*. Rigor can be achieved by applying sound methodologies, whereas relevance can be achieved by addressing the needs of involving practitioners in the research. Anyway, the diversity of management concerns and differing organizational contexts in the field of the EA management function have hampered the development of a single and embracing management approach. Moreover, the proposed solution by the research projects in close cooperation with the industry partner has not only to satisfy the organization's pace, but also has to provide a general valid solution for the IS community. In order to face up the aforementioned challenges, the PDR approach is used which also accounts the aspect of rigor in practice-driven research projects [Buckl et al. 2013]. The PDR approach makes use of pattern, design theories and a design theory nexus. In the following, only highly relevant parts of the PDR approach for this paper is presented. For detailed information see [Buckl et al. 2013].

The PDR method consists of four main activities. The first main activity *observe and conceptualize* represents the problem diagnosis, whereas the second *activity pattern-based theory building and nexus instantiation* enables the abstraction of observed solutions. *Solution design and application*, which is the third activity, incorporates the creation of an IS artifact. The last activity *evaluation and learning* closes a direct feedback loop from practice to academia.

The first activity *observe and conceptualize* has a major importance for this work, because it suggests observing good practices from industry, which can be documented in a typical pattern structure. For the documentation, it is required to describe at least the following concepts: *problem* to be addressed, *solution* that has proven to work good, *context* in which the solution can be applied to, and *forces* that frame the solution space. Accordingly, in the research phase of this paper good practices were observed and several of them were documented. A part of them are presented in Section 5 and additional ones can be found in the Appendix. This observation and collection of good practices builds the base for the *pattern-based theory building and nexus instantiation* activity, wherein good practices or pattern candidates are evaluated for their fulfillment of the rule of three. The rule of three was established in [Coplien 1996] and proposes the assumption that a documented pattern must provide at least three known usage references in practice to ensure reusability of provided solution. However, accordingly to the PDR method, the second version of the EAMPC will only contain best practices. This means that only pattern candidates with at least three known appearances in practice will be documented in the EAMPC.

2.2 Pattern-based Approach

The pattern-based approach to EA management has been developed to address typical problems of EA management approach like too abstract guidelines, which lack appropriate guidance to be used in practice, or monolithic approaches pursuing an all or nothing approach, neglecting enterprise-specific EA related concerns. An initial set of patterns has been collected from literature and practice, which has been evaluated in an extensive survey. This resulted in the creation of the first EAMPC containing 120 EA management patterns [Buckl et al. 2009b]. The objective of the EAMPC is to complement existing EA management framework like TOGAF or Zachman framework. The EAMPC provides a holistic and generic view on the problem of the EA management function by providing additional detail and guidance. This is essential in order to establish the EA management function systematically in a step-wise fashion within a given enterprise.
The EAMPC identifies the dependencies between:

- individual management *concerns* (Which concern is relevant for which stakeholder?),
- management *methodologies* (Which activities are required to address a concern?),

– supporting *viewpoints* (Which viewpoints help stakeholders to collaboratively perform these activities?), and
– *information models* (Which information is required in order to generate a particular viewpoint?).

Methodologies, viewpoints and information models are presented as EA management patterns, which describe possible solutions for recurring problems in the context of the EA management function that can be adapted to a specific enterprise context [Buckl et al. 2008a]. A *problem* in the EA management function typically consists of a *goal* representing an abstract objective, i.e. provide transparency, and a *concern*, i.e. area of interest in the enterprise, e.g. business support, application systems [Buckl et al. 2011]. The EA management pattern approach is a concern driven approach and distinguishes between three different pattern types for reasons of usability and separation between information and its representation [Ernst 2010, Buckl et al. 2008a].

The conceptual UML class diagram in Figure 1 gives an overview of the inner structure of the utilized concept, the dependencies between the three EA management pattern types and their relationships to problems, which are delineated in the following.

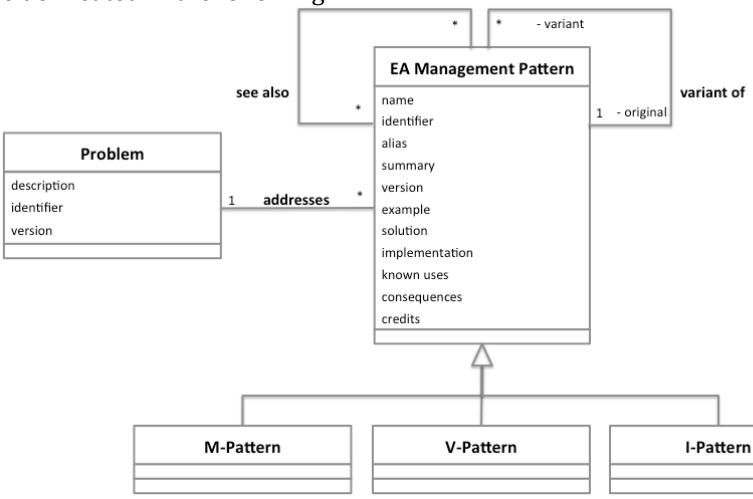

Fig. 1. UML class diagram describing the relationships between problems, M- Patterns, V-Patterns and I-Patterns (based on [Ernst 2008]).

2.2.1 *Form of EA management patterns.* Ernst [2010] proposes a template for documenting EA management patterns. For the reasons of readability and usability of the EA management patterns, the EAMPC uses the same template for all EA management patterns. The distinct sections contain the inner structure of an EA management pattern and its relationship to a problem and other EA management patterns (see attributes and relationships of Figure 1). However, the initial version of the EAMPC does not cover all the sections proposed by [Ernst 2008]. For the reason that this paper builds on the utilized template of the EAMPC, only the used sections by the EAMPC are described in the following table. The other sections are not further considered due to understandability.

Table 1 Template for documenting EA management patterns (based on [Buckl et al. 2008a, Ernst 2008])

Overview Section	
Id	An unique alphanumerical identifier
Name	A short and expressive name for the EA management pattern
Alias	Names this EA management pattern is also known as (optional)
Summary	A short summary of the EA management pattern
Version	Version number of the EA management pattern
Solution Section	
	Detailed description of the EA management pattern
Consequence Section	
	Consequences resulting from the usage of the EA management pattern (optional)

A methodology pattern (M-Pattern) also includes a *Problem Section*. This section lists the concerns, which are addressed by the respective M-Pattern. Empty consequence sections are omitted for the reason of brevity [Buckl et al. 2008a]. As 1 indicates that there are three specializations for an EA management pattern:

Methodology pattern, viewpoint pattern, and information model pattern. Descriptions of those three types of EA management patterns are given in the following section.

2.2.2 *EA management pattern types.* The first EA management pattern type is the M-Pattern. An M-Pattern is defined as a documentation of a proven practice solution to a recurring problem for a specific context in form of a process for the EA management function. It also describes roles, the steps to be taken in the process, inputs and outputs of the process, as well as known variants, and consequences related to its usage. During its execution, the documented process can make the use of one or more viewpoint and information model.

Viewpoint patterns (V-Patterns) provide a modeling technique that is used by one or more M-Patterns. Thus, it proposes a way to present data stored according to one or more information model patterns (I-Patterns). A V-Pattern is defined as a documentation of a proven practice solution to a recurring problem for a specific context in form of viewpoints for the creation of views. It also describes techniques for the creation and usage of the view, as well as known variants, and consequences related to its usage. The documented viewpoint is a representation or input method for information. This information can be stored to one or more I-Patterns. A viewpoint can be graphical, tabular or text based.

The last EA management pattern type is the I-Pattern. I-Patterns supply an underlying model for data, which are visualized in one or more V-Pattern or used in one or more M-Patterns. An I-Pattern is defined as a documentation of a proven practice solution to a recurring problem for a specific context in form of an information model fragment for the creation of an information model. The I-Pattern also includes required definitions and descriptions of the used information objects. Furthermore, it documents techniques for information model fragment implementation and usage, as well as known variants, and consequences related to its usage [Ernst 2010].

2.2.3 *Relationship types between of EA management patterns*. Figure 1 indicates that there exist two relationship types between EA management patterns. The first one, namely *see also*, references the EA management patterns to other EA management patterns solving similar problems, and to EA management patterns that help to refine the pattern under consideration. The second one, namely *variant of*, provides a brief description of variants and specializations of an EA management pattern [Ernst 2010].

2.2.4 *Motivation for the extension of the existing concept*. Due to the fact that the EA management body-of-knowledge has grown significantly and the documentation of EA management patterns is not a one-time approach, it seems necessary to develop a new concept for the new EAMPC. This section contains important information that advocates the aforementioned extensions.

Firstly, the actual structure of the EA management pattern language contains the *see also* and *variant of* relationship types between patterns. However, Ernst [2010] shows further existing relationship types, which are not documented yet, like the *sequence of elaboration* relationship type. Therefore, Ernst [2010] predicates that the definition of pattern sequences for supporting the pattern user in applying the patterns could be a future area of research.

Secondly, to narrow down the number of potential concerns of an organization, maturity models can be used for determining the degree of maturity of its EA management function. The reason for this is that diverse developed EA management functions have different concerns. After selecting concerns in maturity levels, EA management patterns can then be identified for addressing them.

Thirdly, the observed good practices in the research phase of this paper and the new concept of the EAMPC show that an M-Pattern is not necessarily required to address a concern, because a V-Pattern can also address a concern directly. Nevertheless, the current concept does not include this relaxed point and the documentation template for V-Patterns has to be adjusted.

Fourthly, the current pattern-based approach to EA management does not consider stakeholders explicitly. Ernst [2010] also highlights this point and mentions that the documentation of stakeholders would offer another way for selecting EAM patterns and would also improve the value of the patterns themselves as the relevant stakeholders are an important and valuable information.

Lastly, design related tasks of the EA management function also include architecture principles that are a key concept in the definition of EA [Zadeh et al. 2012]. Besides M-Patterns, architecture principles also address some concerns and provide guidance, which are also not documented yet. All aforementioned points call for an extension of the current structure for documenting EA management patterns, which is presented in the subsequent sections.

2.3 Extension of the existing structure for documenting EA management patterns

The extension of the current structure for documenting EA management patterns includes the following aspects:

- extension of the existing conceptual model (see Figure 1) and actual documentation template (see Table 1)
- documentation of new EA management pattern related elements: *influence factors* and *stakeholders*,
- extension of current structure of EA management patterns and pattern language elements: creation of *Hierarchical Reference* and *References to other Standards* sections, and *type* denotation of V-Patterns, and
- documentation of new *compounds*, *complements* and *sequences* relationship types.

The figure below demonstrates the extension of the existing conceptual model presented in Figure 1.

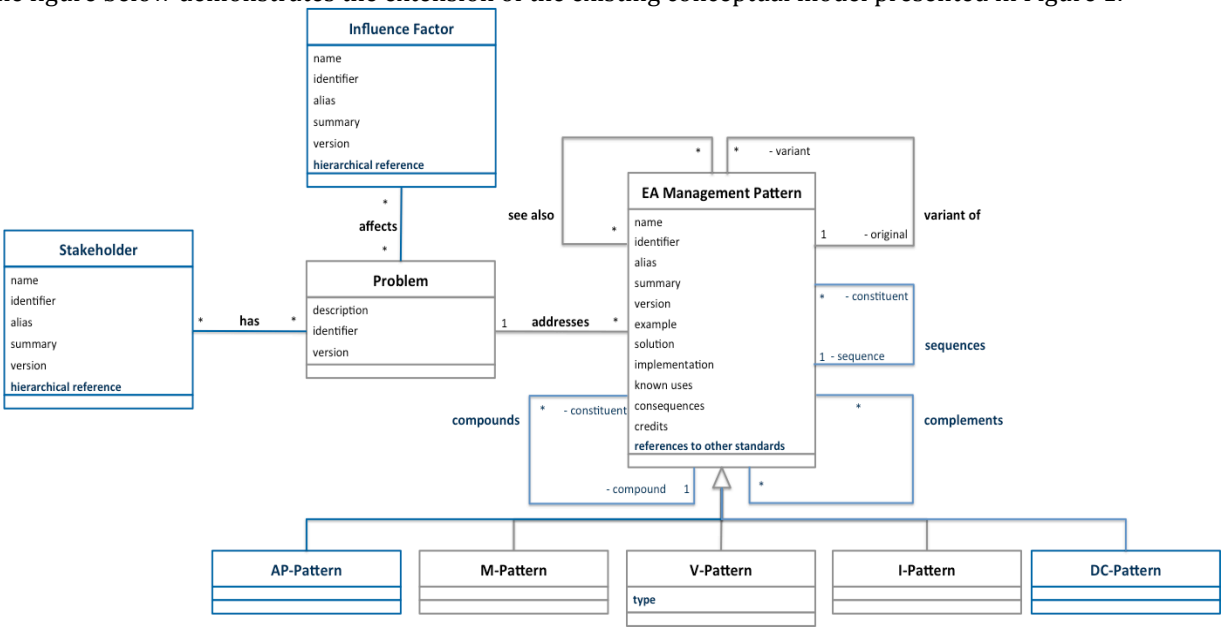

Fig. 2. UML class diagram describing the extended structure of the initial conceptual model of EA management patterns.

First of all, blue highlighted parts of the above UML class diagram indicate new features of the extension. As Figure 2 demonstrates, two new classes: architecture principle pattern (*AP-Pattern*) and data collection pattern (*DC-Pattern*) are added, which are special types of EA management patterns that inherit the same attributes as the generalization class *EA Management Pattern*. Further two added classes are *Influence Factor* and *Stakeholder*. An influence factor may affect one or more problems. Furthermore, a stakeholder may have one or more specific concerns, whereas a specific concern may be related to one or more stakeholders. Momentously, the initial conceptual model of EA management patterns does not realize a classification between EA management patterns and other types of elements. This is due to the fact that the native EA management pattern language only contains concerns as such an element. Now, these types of elements are extended by two more classes: *Influence Factor* and *Stakeholder*, wherefore a classification seems to be necessary. Consequently, this paper classifies between EA management patterns: AP-, M-, V-, I- and DC-Patterns and EA management pattern related elements: influence factors, stakeholders and concerns.

Further on, the UML class *EA Management Pattern* includes the *References to other Standards* attribute. It is contained in EA management patterns, which refers to similar concepts of other EA management standards. Besides this, EA management pattern related elements contain a *Hierarchical Reference Section*, which lists all other EA management pattern related elements and EA management patterns that are related with it. Henceforth, a type attribute extends V-Patterns, which states their specific visualization types.

Lastly, the class *EA Management Pattern* is extended by three additional relationship types: *compounds*, *sequences* and *complements*. The subsequent sections present more detailed information of the above-mentioned extensions.

An essential goal of this extension is to provide a coherent documentation of EA management patterns and EA management pattern related elements. As a result, two types of documentation templates are presented in the following.

Table 2 Extended template for documenting EA management patterns

Overview Section	
Id	An unique alphanumerical identifier
Name	A short and expressive name for the EA management pattern
Alias	Names this EA management pattern is also known as (optional)
Summary	A short summary of the EA management pattern
Version	Version number of the EA management pattern
Problem Section (only for M-, AP- and V-Patterns)	
	The concerns a pattern addresses
Solution Section	
	Detailed description of the EA management pattern
Consequence Section	
	Consequences resulting from the usage of the EA management pattern (optional)
Complements Section	
	A list of competitive or complementing EA management patterns (optional)
Compounds Section	
	Description of constituent EA management patterns for the compound EA management pattern, respectively a reference to the compound EA management pattern for the constituent EA management pattern (optional)
Sequences Section	
	A list of constituent EA management patterns for the sequence EA management pattern (optional)
References to other Standards Section	
	A list of similar concepts of other EA management standards to the corresponding pattern (optional)

By comparing Table 1 with Table 2, a *Problem Section* is now also available for AP- and V-Patterns. Reasons are on the hand that an AP-Pattern can alike addresses a specific concern, and on the other hand that with the new EAMPC concept, V-Pattern can also address a concern without using an M-Pattern. The *Overview Section* of a V-Pattern also contains the *Type* attribute, which describes its visualization type. *Complements*, compounds and *sequences*, which are new introduced relationship types, are provided as optional sections.

An exception is that the *Problem Section* of a V-Pattern also lists related stakeholders compared with M- and AP-Patterns, because a stakeholder may use a V-Pattern without having a specific concern. For the other two aforementioned EA management patterns, a concern has to be addressed without referencing to a stakeholder directly. Another difference between these two tables is the documentation of the *References to other Standards*, which is available for all EA management patterns.

Table 3 Template for documenting EA management pattern related elements

Overview Section	
Id	An unique alphanumerical identifier
Name	A short and expressive name for the EA management pattern related element
EAM Topic	Corresponding EA management topic of the concern
Alias	Names this EA management pattern related element is also known as (optional)
Summary	A short summary of the EA management pattern related element
Version	Version number of the EA management pattern related element
Hierarchical Reference Section	
	A list of related EA management patterns and EA management pattern related elements (optional)

The initial documentation of a concern in the EAMPC only contains an identifier, a formulation of a question and a list of M-Pattern addressing it. But, the documentation of influence factors and stakeholders, and the goal to make the documentation mostly consistent, also a documentation template for EA management pattern related elements is created. Table 3 shows a general documentation structure for influence factors, stakeholders and concerns. Attention should be paid for the fact that not all attributes are relevant for the distinct EA management pattern related elements. For instance, a concern contains an identifier, a corresponding EA management topic, a summary, a version and a section for hierarchical references. Compared to the initial documentation, a concern contains a short summary and not a formulation of a question. Furthermore, by providing the corresponding EA management topic of a concern, the EAMPC user is quickly able to classify the concern in the context of EA management function. However, name and alias are not relevant for the documentation of concerns. The remaining two EA management pattern related elements include all parts of the documentation template except the part for EA management topic, since it is only relevant for concerns. In the subsequent sections, the briefly described concepts are at first further detailed and then their benefits for their documentation are provided.

2.3.1 New EA management pattern related elements. As Figure 2 indicates, two new EA management pattern related elements should be documented in the EAMPC, namely influence factors and stakeholders, which are described in the following.

2.3.1.1. *Influence factors.* Today, enterprises have to survive in a continually changing environment that demands continuous transformation [Buckl et al. 2010c, Buckl et al. 2012]. A commonly accepted tool that provides guidance for such enterprise transformation is the EA management function. Such a management function has to be embedded within the organizational context [Buckl et al. 2010b]. By applying the theory of systems, an organization can be seen as system, which is surrounded by an environment. The system theory claims that systems receive inputs from their environment. Referred to an organization, it implies that an organization is influenced by its environment [Bossel 1994]. Burger [2013] adopts this thought in the context of application landscapes, whereas the application landscape is directly surrounded by the organizational context and that in turn is surrounded by its suprasystem the competition or market. Furthermore, Burger [2013] distinguishes between environmental influencing factors and organizational influencing factors. Environmental influencing factors are e.g. laws, globalization, or competitors, while organizational influencing factors represents e.g. new business model, employee's different skills and personalities, or outsourcing. Factors of the latter kind affect the organizational influencing factors, which in turn has an effect on the application landscape. Since the management of application landscape is a focal point of EA management function, this adoption can be also applied to the EA management pattern approach [Buckl et al. 2009a]. For the reason of clarity, the aforementioned factors are in the following merged to influence factors.

Buckl et al. [2010d] provides a pattern structure that contains a context description, which is concerned with causes and environmental factors that may have lead to a specific problem that a pattern solves. Ernst [2010] also applies this structure in the context of the EA management function. As mentioned in Section 2.2, a problem in EA management typically consists of a goal and a concern. Due to the fact that influence factors may lead to highly diverse problems, they consequently lead to concerns. Accordingly, knowing which influence factors may lead to which concerns may be valuable and important information. Additionally, the pre-study of this research (see Section 3) observed fourteen distinct influence factors in practice that affect existing and newly found concerns. Unfortunately, this knowledge does yet not exist in the EAMPC. Thus, the documentation of influence factors seems to be necessary.

A special type of influence factors represents maturity levels. Various domains use the concept of maturity levels to organize topics with the domains in order to help organizations to improve their approaches on industry and government best practices [Ernst 2010]. Also in the field of EA and its management, maturity models have been developed, like the National Association of State Chief Information Officers Enterprise Architecture Maturity Model or the Extended Enterprise Architecture Maturity Model. Due to limit, these models are not further detailed, instead a four step approach proposed by [Ernst 2010] for selecting EA management patterns based on maturity models is presented:

- **Select maturity model**, which should be applied or which already is in usage in the organization.
- **Determine maturity level** the company has reached in the EA management function to reduce the number of potential concerns.

– **Select concerns in maturity level** and thereby select EA management patterns addressing the existing concerns. If a concern can be addressed by multiple EA management patterns, then the EA management pattern, which best fits the company's context, has to be identified.
– **Select additional EA management patterns** is equal to the same step described in preceding approach.

In order to apply the maturity model-based approach for selecting EA management patterns, the maturity levels of the respective model can be documented as influence factors. As a result, an influence factor can reference to concerns that typically occur for the specific maturity level of the EA management function.

2.3.1.2. *EA stakeholders.* Typical application scenarios of the EA management function are inter alia strategic IT planning, business-IT alignment, process optimization, and architecture conformity of projects [Aier et al. 2008]. For that reason, a considerable challenge for the EA management function is to foster the communication between the involved stakeholders, e.g. the project director, the standards manager, and the enterprise architect in the case of an architecture review process. Consequently, two major tasks of the EA management function are the provision of an appropriate form of information for the respective stakeholder and the commitment of the key stakeholders [Buckl et al. 2010e, Niemi 2007]. As a consequence, the key stakeholders and their requirements for the EA management function need to be identified [Niemi 2007, Armour et al. 1999]. Since practitioners consider TOGAF as an interesting framework within the context of the EA management function, this paper utilizes the definition of a stakeholder based on TOGAF [The Open Group 2013b, Buckl et al. 2009b]. As a result, a stakeholder is defined as:

"An individual, team, or organization (or classes thereof) with interests in, or concerns relative to, the outcome of the architecture. Different stakeholders with different roles will have different concerns"

Besides this definition, TOGAF also provides a development method for enterprise architecture, the so-called architecture development method (ADM), which describes the ten different phases of EA development. After the preparation and initialization of the EA management approach (*Preliminary* phase of the TOGAF ADM cycle), the scope of the EA management function endeavor is defined within the *Architecture Vision* phase. A core objective of this phase is the identification of relevant stakeholders and their concerns, which is supported by guidelines of ADM's stakeholder management.

A common phenomenon in the EA management function is that academia and practice use diverse denotations for the same stakeholder role. For instance, TOGAF and Niemi [2007] utilize the term *Project Manager*, whereas Raadt et al. [2008] use the term *Business project manager* for denoting the specific stakeholder role. This is also the case for EA management function practicing companies. The extensive pre-study of this research (see Section 3) observed 137 different denotations for stakeholder roles, which are consolidated to forty-eight different stakeholder candidates due to the fact that multiple denotations express the same stakeholder role. This phenomenon is also observed by [Niemi 2007]. Niemi [2007] states that the view of stakeholders in the EA context is conspicuously inconsistent, even with the great number of stakeholders identified by the literature.

The stakeholder map provided by ADM's stakeholder management and the conceptual model of architectural description indicate that beyond to concerns, viewpoints might be also stakeholder specific [The Open Group 2013b, ISO/IEC/IEEE 2011]. Therefore, considering stakeholders for selecting V-Patterns or concerns is important and valuable information, which is yet not accessible in the EAMPC. For that reason, and for the reason of the inconsistent view of stakeholders in academia and practice, the documentation of stakeholders seems to be indispensable. The documentation of stakeholders overcomes the aforementioned problems by providing a well-understood definition, by referencing stakeholders to relevant concerns and V-Patterns and by referencing other EA management standards like TOGAF for similar concepts and terminologies.

2.3.2 Extension of structure of EA management patterns and EA management pattern related elements. Figure 2 indicates that the current structure of the documentation template of EA management patterns and EA management pattern related elements is extended. It includes for EA management patterns the *References to other Standards Section* and for the EA management pattern related elements the *Hierarchical Reference Section*. The first mentioned section might include references to other EA management frameworks. The latter one is contained in influence factors, stakeholders and concerns in order to determine and refer to related EA management patterns for these EA management elements. Lastly, V-Patterns are extended by a *type* attribute

that states their specific visualization types: EA visualization, metric or report. This extension also indicates that EA metrics and reports are documented now.

2.3.2.1. *Metric and report patterns.* The existing version of the EAMPC and its wiki implementation actually contains 74 V-Patterns. Presently, a V-Pattern in the EAMPC provides an exemplary view for the viewpoint, possibly together with some textual explanations, and a corresponding legend [Buckl et al. 2008a]. The exemplary view of the viewpoint does not need to be graphical, it can be also tabular or text based. However, besides of EA visualizations, there also exist other types of visualizations, namely metrics and reports, which are not included in the present EAMPC yet.

In recent times, metrics are gaining attention in industry to encourage the analysis of an EA and its evolution. These metrics aim to measure and assess organization-wide progress of transformations. They are based on business processes, applications, infrastructure information, and their interrelations. The amount of this information and the efficient computation of EA metrics require the support of adequate tools [Hauder et al. 2013]. Several vendors exist in practice, which provide appropriate tools for inter alia the visualization and preparation of EA metrics. The data collection phase of this research identified several of them, for instance planningIT, ADOit or MEGA Modeling Suite. Matthes et al. [2012] provide a collection of 52 usable EA metrics. Additionally, each metric is described by a structured template, which ensures consistency among the documented metrics. The structure template consists of different information categories: *description*, *calculation description*, *organization-specific instantiation*, *goals*, *layers*, *sources* and *information model*. The documentation template presented by Table 2 can be used to model an EA metric as a V-Pattern. The following figure shows a mapping template for conveying the aforementioned information with into the documentation template of EA management patterns. It therefore facilitates the integration of already documented EA metrics by the EAM KPI Catalog into the EAMPC, and the documentation of newly found EA metrics.

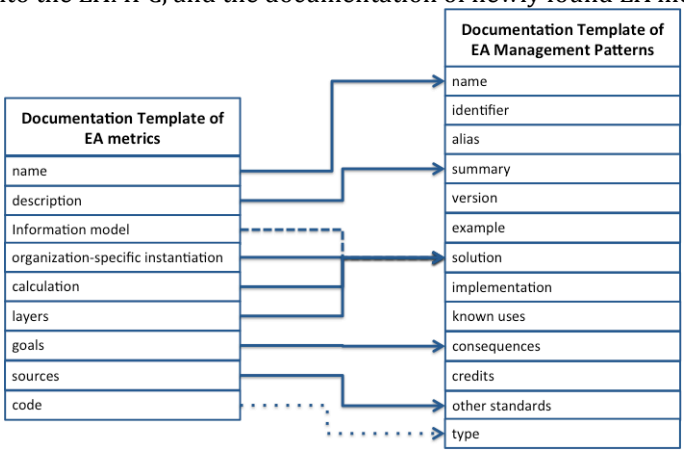

Fig. 3. Template for mapping EA metrics according [Buckl et al. 2008a] with documentation template of EA management patterns

As Figure 3 indicates, all information of the documentation template of EA metrics can also be modeled in the documentation template of EA management patterns. The *name* and the *description* of an EA metric can be directly represented by the *name* and *description* of a V-Pattern. However, the *information model* of the EA metric is not directly described by the V-Pattern itself, but by the *Solution Section* of the corresponding I-Pattern. O*rganization-specific instantiation* and *calculation,* and *layers* can be directly mapped onto the *Solution Section* of the V-Pattern. Nevertheless, the *Solution Section* of the V-Pattern is not organization-specific. Therefore, the concrete attributes of the organization-specific instantiation shall be exemplified or generalized. The *Consequence Section* of the V-Pattern can represent the *goals* of the EA metric, which provide information about the promised benefits by selecting it. *Sources* that refer to other standards can be mapped onto the *References to other Standards Section* of the documentation template of EA management patterns. Since the documentation template already contains a unique identifier, the *code* of the EA metric cannot be directly mapped on to the V-Pattern. Instead, the *type* attribute, which is exclusive for V-Patterns, may contain besides the term *metric* also the EA metric identifier in brackets. For instance, the metric *Application continuity plan availability* with the code *EAM-KPI-0001* is mapped to the V-Pattern with the name *Application continuity plan availability*, the unique identifier *V-112* (the number is chosen arbitrarily), and the type *metric (EAM-KPI-0001)*. The former logic for setting identifiers should not be changed.

The above-mentioned EA management tools also provide report functionalities. Reports are textual documentations, which support the users when performing different EA management function tasks and represent the last type of visualizations [Matthes et al. 2008]. These reports have a logical structure that can be tabular arranged and therefore visualized as the previous presented visualization types. For instance, a tabular report provides enough information in order to build its information model and respectively its corresponding I-Pattern. Additionally, the table of a report can be exemplified by the *Solution Section* of a V-Pattern.

For the reason that V-Patterns should support users to perform EA management function tasks, EA metrics and reports should also be documented in the EAMPC as V-Pattern and complement the existing EA visualizations.

2.3.2.2. References to other Standards Section. The EAMPC supports different usage scenarios, of which one is of major importance for this paper and is detailed in the following. A present usage scenario for the EAM pattern approach is to inspire and assess an existing EA management approach. For this reason, the EAMPC is used as a reference book, which contains suggestions for improving the approach currently used in an organization [Buckl et al. 2009b]. In this case, the major goal is to complement the distinct phases of ADM by providing best practices for methodologies, viewpoints and information model fragments. However, actually, the EAMPC does not contain explicit references of EA management patterns to other EA management standards. By including the *References to other Standards Section*, the usage scenario for inspiring and assessing existing EA management approach can be enhanced. This section is available for EA management patterns that are referenced to similar concepts of other EA management standards. For instance, organizations, which are already using certain architecture principles from TOGAF, are able to refer to similar AP-Patterns provided by the EAMPC. This provides the enhancement of transparency between the EAMPC and other standards, because users may directly compare and classify the used EA management patterns with other related concepts. Additionally, it enhances the EAMPC's functionality as a reference book, because EAMPC users may deepen their knowledge on a selected EA management pattern by guiding to related information of similar concepts.

2.3.2.3. Hierarchical Reference Section. Actually, a concern, which is considered in the EAMPC, consists of an identifier, a formulation of a question and a list of M-Pattern addressing it. However, this list has to be extended by AP-Patterns and V-Patterns, because the newly documented AP-Patterns are also addressing concerns and the EAMPC concept 2015 allows V-Patterns to address concerns directly without implicitly using M-Patterns. Due to the fact that influence factors and stakeholders exhibit the same structure as concerns, similar reference lists are required for them. For the aforementioned reasons, the documentation of references will gain more complexity. In order to handle this complexity and arrange references neatly for improving the usability of the EAMPC, the *Hierarchical Reference Section* is introduced and described.

First of all, this section is only available for influence factors, stakeholders and concerns, because the proposed structure for EA management patterns already allows EA management patterns to refer to linked patterns. Secondly, the *Hierarchical Reference Section* of an EA management pattern related element lists all other EA management pattern related elements and EA management patterns that are related with it. The following table provides an overview of reference lists, which are present in hierarchical reference sections.

Table 4 Reference lists contained in hierarchical reference sections

EA management pattern related element	**Existing reference lists in the *Hierarchical Reference Section***
Influence factor	– Affects concern(s)
Stakeholder	– Has concern(s) – Uses V-Pattern(s)
Concern	– Affected by Influence factor(s) – Related Stakeholder(s) – Addressed by M-Pattern(s) – Addressed by AP-Pattern(s) – Addressed by V-Pattern(s)

Moreover, Table 4 shows all existing relationships of influence factors, stakeholders and concerns with other EA management pattern related elements and EA management patterns, which are drawn as association edges in Figure 2. The *Hierarchical Reference Section* of an influence factor contains all concerns that are affected by the influence factor. Accordingly, the *Hierarchical Reference Section* of a stakeholder contains all

concerns that the stakeholder has and all V-Patterns that the stakeholder is using. Lastly, the *Hierarchical reference Section* of a concern includes all IFs that affect the concern, all stakeholders that are related with the concern, and all M-, AP- and V-Patterns that addresses the concern.

This setup has the benefits that the EAMPC user is easily top down or bottom navigated by the *Hierarchical Reference Section* and that the usability of the EAMPC is improved. This consistent structured section is implemented by all EA management pattern related elements.

2.3.3 *Extension of EA management pattern types*. The new EAMPC concept 2015 does not only include the extension of the existing structure of EA management patterns and EA management pattern related elements, but also the introduction of new patterns and elements that are described in the following.

2.3.3.1. *Architecture principle patterns.* The definition of EA in Section 1 indicates that principles represent an essential element of an EA. An EA principle constrains and guides the EA's design and may in turn provide justification for decision-making throughout an EA [Buckl et al. 2010a]. The characteristics of EA principles are for one thing that they define underlying general rules and guidelines for the use and deployment of IT resources and assets across the enterprise, and for another thing that they reflect a level of consensus among the various elements of the enterprise. Another important characteristic is that each EA principle should be clearly related back to the business objectives and key architecture drivers [The Open Group 2013b]. TOGAF recommends that EA principles should be defined on a standard way and additionally provides a template for this use case. This template consists of the four sections: *Name*, *Statement*, *Rationale* and *Implications*, which closely resemble in their content with specific sections of the template for documenting EA management patterns from Section 2.2.1.

The *Name Section* of the TOGAF template, which contains the essence of the rule, matches with the *Name* Section of Table 2. The *Statement Section* of the TOGAF template succinctly and unambiguously communicates the rule, which can be transferred to the *Summary Section* of the EA management pattern documentation template. The *Rationale* and *Implication Sections* highlight both the business benefits of adhering to the EA principle and the requirements for the business and IT for carrying out to the principle – in terms of resources, costs and activities. These two sections can be subsumed to the *Consequence Section* of the EA management pattern documentation template. Resulting from this comparison, EA principles exhibit broadly similar characteristics as EA management patterns in terms of documentation.

Although EA principles play an important role in practice, the design activity issues of the EA are often neglected [Fischer et al. 2010]. Also, the existing literature remains vague about what can considered as suitable EA principle for guiding EA design and evolution [Haki and Legner 2013]. This problem is also stated by Zadeh et al. [2012] that accentuate the fact that EA principles have received less attention than other EA concepts such as models and views, despite the fact that they perceived importantly. The extensive survey conducted by [Winter and Aier 2011] comes to a similar observation. The survey concludes that other EA management approaches and tools have successfully expanded their focus from the IT architecture to the business side, but that this widened scope cannot be observed for EA principles. It reasons this by arguing that the current EAM practice regarding EA principles is too static and is too less customer oriented, because stakeholders are not involved and business architecture principles are missing.

In order to overcome the aforementioned problems, a stakeholder oriented and embracive documentation of EA principles seems to be necessary. Especially because of EA principles are perceived as important means for EA design and essential factor for the success of the EA management function. For the reason that the documentation of EA principles is similar to the existing EA management patterns, henceforth, the concept of AP-Patterns is introduced.

2.3.3.2. *Data collection patterns.* During the process of analyzing the current state and planning towards a desired target state of the architecture, organization's struggle with documenting the EA and lack concrete guidance during this process [Roth et al. 2013]. However, Roth et al. [2013] conducted a survey with 140 organizations in order to obtain an overview of practices and challenges organizations face when documenting their EA. The survey results identified two key challenges while documenting EA, namely the huge efforts of EA data collection and the quality of the resulting model. In order to overcome the major problem of the absence of defined processes or best practices for documenting an EA, patterns, which are used in practice to collect EA information, can be identified and synthesized. As a conclusion, the documentation of data collection of EA information as EA management patterns seems to be useful. As a result, DC-Patterns, which document

recurring solutions to the data collection problem, are introduced. They should answer the following questions for ensuring an adequate support for organizations:

- How often is the data collected or updated, e.g. in real-time, on daily basis, or event driven?
- Who is responsible for the data collection?
- Which source, e.g. person, role or IT system, is used for the data collection?
- How is the data collection process integrated with the EA repository, e.g. with the help of a manual interface or automated interface or without integration?

Another point that has to be considered is how DC-Patterns are related to existing EA management patterns. Since a DC-Patterns contains required information, e.g. data source or responsible person, for building an I-Pattern, it is directly associated with an I-Pattern. For creating a DC-Pattern, there exist two alternatives. The first alternative proposes to build a DC-Pattern for exactly one I-Pattern, whereas the second one suggests creating a DC-Pattern for exactly one class of an I-Pattern. The selection of the proper alternative is based on the evaluation results of the pre-study of this research, which is presented in Section 4.

2.3.4 *Extension of relationship types between EA management patterns.* In the area of pattern-based approaches, there exist additional types of relationships between patterns than presented in Section 2.2.3. Buschmann et al. [2007] discusses further relationship types that are presented in the following.

2.3.4.1. *Pattern Complements.* The *pattern complements* relationship type is based on the idea that more than one solution can be used to address a problem. This solution can be either complementary with respect to competition or complementary with respect to structural correctness. The first aspect assumes that one pattern may complement another one, because it provides an alternative solution to the same or a similar problem, and therefore is complementary in terms of the design decision that can be taken. The second one implies that one pattern may complement another one, because it completes a design, which acts as a natural pairing to the other in a given design.

2.3.4.2. *Pattern Compounds.* Many patterns can be grouped together at one apparent level of detail. Yet they can be examined closely, other patterns contained within them can be seen. Based on this understanding, a *pattern compound* is defined as a named, commonly recurring, cohesive combination of other patterns. A pattern compound is a mutually supportive community of patterns that define a larger design fragment than what might be considered a pattern element. The constituent patterns of the compound pattern commonly appear together to resolve a particular problem in a given context.

Buschmann et al. [2007] proposes three different approaches for documenting pattern compounds. The third suggested approach seems to be the most appropriate one, because it offers more option to readers. Nevertheless, it requires documenting both the pattern compound and its constituent patterns. This approach treats the documentation of a compound as a collection of patterns and then as a specific role combination of the collection. This documentation type also brings the compound together in a brief description that builds on its constituents. A focal point for documenting this relationship type is that the compound pattern does not directly refer to one or more I-Patterns. Instead, the constituent EA management patterns do refer to the corresponding I-Patterns.

2.3.4.3. *Pattern Sequences*. The application of one particular pattern after another is also called as *pattern sequences*. A pattern sequence represents a process made up of constituent patterns that are arranged in an order that achieve a particular architecture, architectural feature, or change of environment with desired properties. A pattern sequence is thus a successive progression of design decisions. Each point in the pattern sequence responds to the feedback and context created by the previous pattern. To be useful in practice, detailed description about the context at the start and the goal and desired properties of the pattern sequence is required.

2.3.4.4. *Motivation for adopting additional relationship types between EA management patterns*. The extended documentation of EA management patterns with the relationship types above enable an additional opportunity for structuralizing EA management patterns, and to reveal new interactions between them which are until today concealed. This is due to the fact that the current relationships: variant of and see also do not describe these types of interactions between EA management patterns.

3. RESEARCH APPROACH

The previous described and motivated concepts for extending the actual EA management pattern language have to be examined for their documentation applicability in the new EAMPC by providing the following research approach in accordance with the PDR method. The following research approach consists of five distinct phases. The following figure provides an overview of the research approach.

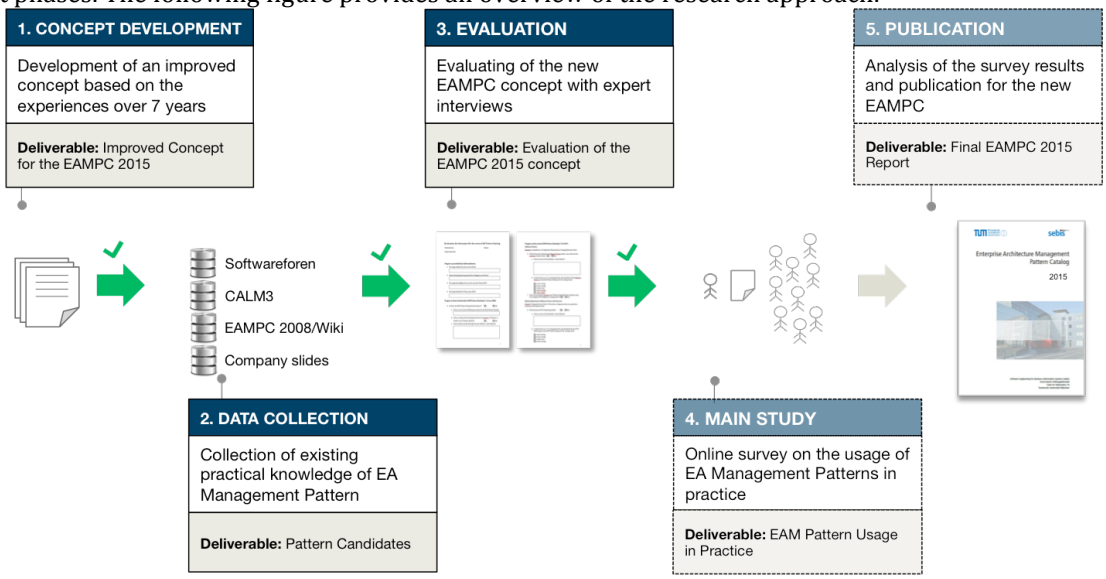

Fig. 4. Research approach for the development of the second version of the EAMPC

Since the initial publication of the EAMPC in 2007, the EA management body-of-knowledge has grown significantly and a lot of experiences in this area and EA management pattern language have been collected. Based on these experiences, the initial concept of the EA management pattern language has been revised. As Figure 4 indicates, the revision has been conducted in the *concept development* phase of this research. As a result, a new improved concept for the second version of the EAMPC has been developed. This improved concept includes inter alia the extension of the present EA management pattern language presented in Section 2.3, like the documentation of stakeholders, influence factors, and AP- and DC-Patterns. The concept development was performed between January and February 2015. This paper builds on the concept development phase and includes the second and third phases of the research approach.

The *data collection* phase was performed from February to March 2015. During this phase, four distinct data sources have been analyzed. Initially, existing EA management patterns of the EAMPC 2008 and wiki were merged together. In total, this resulted in the observation of 59 concerns, 25 M-Patterns, 74 V-Patterns and 54 I-Patterns. Secondly, 1208 slides from the 50 different companies, like Debeka, Gothaer Systems, BMW, BITMARCK, Evonik and T-Systems, ranging from insurance to pharmaceutical, automobile to corporate consulting, were analyzed. The analyzed slide sets were inter alia presented in various EA management conferences, like the EAMKON in 2009 in Stuttgart or Softwareforen in 2012 in Leipzig, and in the CALM3 - Complexity of Application Landscapes – Measures, Models, Management Workshop 3. Furthermore, the presentation dates of the analyzed slide sets ranged from October 2009 to September 2014, wherefore the actuality of the analyzed EA management related issues could be ensured. The content of the analyzed slides contains mainly insights of the organization's performed EA management function. For instance, which tools are used in order to support the user while performing a special task or what are major challenges while planning the future state of the application landscape. The analysis aimed to collect good practices. In this case, it aimed the observation of existing and new proposed EA management patterns and EA management pattern related elements. A particular point is that the observed good practices or pattern candidates were enriched with the existing EA management patterns, M-, V- and I-Patterns and concerns (until that time, AP- and DC-Patterns, influence factors and stakeholders have not been documented yet). Lastly, a survey with 14 respondent companies, which also participated in Softwareforen in 2014 in Leipzig, was conducted in 2014. In this survey, the current status of the EA management function practice was analyzed. Additionally, the analysis of the survey also resulted in the observation of EA management patterns and EA management pattern related

elements. These good practices were also subsumed to the aforementioned collection of EA management patterns. Following the PDR method, the newly observed good practices are not yet proven. Therefore, they have to be examined for their fulfillment of the rule of three in the following *pattern-based theory building and nexus instantiation* activity of PDR method. On that account, the observed good practices were separated from the existing EA management patterns and concerns in the data collection. After the enrichment and observation of good practices, the data collection was consolidated. The consolidation included the analysis whether the observed good practice is redundant or usable, or not. For instance, this is the case when an observed pattern candidate was already documented as an EAM management pattern or two observed pattern candidates can be combined together. All in all, 14 influence factor candidates, 48 stakeholder candidates, 40 concern candidates, 61 M-Pattern candidates, 19 AP-Pattern candidates, 91 V-Pattern candidates, 19 I-Pattern candidates and 30 DC-Pattern candidates were observed. An important point to mention is that the found I-Pattern candidates explicitly or implicitly described the information model for the corresponding V-Pattern candidate. However, after screening observed V-Patterns from V-Pattern candidates, the corresponding I-Pattern can be generated. Moreover, the observed 30 DC-Pattern candidates represent observed tools, like EA management repositories, configuration management databases, or responsible roles for the data administration, like application owner or process manager. Nevertheless, after the documentation of these DC-Pattern candidates the initial proposed concept and the structure for the documentation of the DC-Patterns (see Section 2.3.3.2.) are finalized. Consequently, the number of found I-, and DC-Pattern candidates is not very expressive.

The *evaluation* phase was performed from March to April 2015. Therein, a questionnaire was designed in order to evaluate the proposed extensions of Section 2.3 and developed EAMPC concept 2015. Afterwards, some companies, which have already participated in the survey in 2014, and further organizations, which partook in Softwareforen 2015 in Leipzig, were interviewed. Totally, eight companies replied to the survey. The results of the survey are presented in Section 4.

In the *main study* phase from May to June 2015, collected pattern candidates will be examined for their occurring in other companies within an extensive survey. If they are observed at least at three companies, they will be considered as observed EA management patterns. Furthermore, data for old pattern will be collected and summarized. After the analysis of the extensive survey, the second version of the EAMPC will be published in September 2015.

4. RESEARCH EVALUATION

This section presents the results of the conducted pre-study for evaluating the developed concepts for the second version of the EAMPC.

4.1 Questionnaire and data collection

Data was collected by means of a questionnaire that comprised four sets of questions:

(1) General context of the respondent and company (length of employment, role of respondent in company, experience with EA management function and experience of the company with EA management function)
(2) Four questions regarding the first version of the EAMPC (usage frequency, helpfulness of current structure of the EAMPC, etc.)
(3) 37 questions regarding the new extensions and the EAMPC concept 2015 (importance and helpfulness of documentation of a new concept, like stakeholder, influence factor or DC-Pattern, etc.)
(4) One final question with regard to further concepts, which should be implemented in the second version of the EAMPC (not reported in this paper)

The latter mentioned questions represent the main and most important part of the survey. The asked questions can be found in the Appendix.

4.2 Descriptive survey results

Almost all respondents own the role of an Enterprise Architect (six of eight respondents). One respondent has the role of a field manager for IT architecture and standards while the other one is an advisor for methods and tools. The majority of survey participants are well experienced in the field of EA management function (see Figure 5). 4 respondents reported long EA management function experiences (5 years or more), 3 three to five years and 1 three years or less.

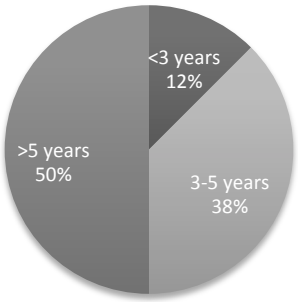

Fig. 5. Distribution of EA management function experience of respondents participating in the survey

The majority of organizations participating in the survey are to some degree experienced in the field of EA management function (see Figure 6). 4 respondents reported EA management function experience between three to five years, 2 companies have EA management function experience more than 5 years and 1 three years or less.

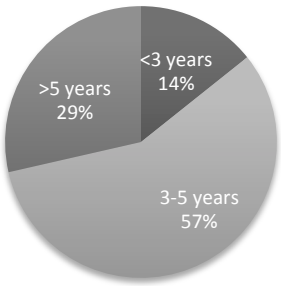

Fig. 6. Distribution of EA management function experience of organizations participating in the survey

4.3 Evaluation results

In the main part of the survey, participants answered some questions to the present EAMPC. All respondents already know the existence of the EAMPC. However, the usage frequency of the EAMPC is varying. 5 respondents used the EAMPC ad hoc or occasionally. 2 respondents used the EAMPC as a one-time approach, whereas 1 respondent did not utilize the EAMPC so far. 6 of 8 respondents found the former structure of the EAMPC auxiliary, while 1 respondent did not find the EAMPC helpful.
As stated before, the main objective of this evaluation survey is to evaluate the new EAMPC concept 2015 and the extended structure to document new EA management patterns and EA management pattern related elements. The following table displays the findings of the evaluation.

Table 5 Research findings for the evaluation of the new EAMPC concept

Concrete new EAMPC Concept	Evaluations
Influence factors	– Q1: **8 of 8** respondents perceive the documentation of influence factors as **helpful** – Q3: The respondents view data provided by the EAMPC for influence factors **in some degree** as **important** (avg. Likert scale value: **4.1**) – Q4: **5 of 8** respondents **do not see the necessity** to categorize concerns or EA management patterns regarding the typical EA management function maturity level.
Relationship between influence factors & concerns	– Q7: **7 of 8** respondents find the documentation of the relationship between influence factors & concerns as **helpful** – Q9: The respondents view data provided by the EAMPC for the relationship between influence

	factors & concerns as **important** (avg. Likert scale value: **4.3**)
Stakeholders	– Q1: **8 of 8** respondents perceive the documentation of stakeholders as **helpful** – Q3: The respondents view data provided by the EAMPC for stakeholders factors as **important** (avg. Likert scale value: **4.4**)
Relationship between stakeholders & concerns	– Q7: **8 of 8** respondents find the documentation of the relationship between stakeholders & concerns as **helpful** – Q9: The respondents view data provided by the EAMPC for the relationship between stakeholders & concerns as **important** (avg. Likert scale value: **4.3**)
Relationship between stakeholders & V-Patterns	– Q7: **6 of 8** respondents find the documentation of the relationship between stakeholders & V-Patterns as **helpful** – Q9: The respondents view data provided by the EAMPC for the relationship between stakeholders & V-Patterns **in some degree** as **important** (avg. Likert scale value: **4.0**)
AP-Patterns	– Q1: **8 of 8** respondents perceive the documentation of AP-Patterns as **helpful** – Q3: The respondents view data provided by the EAMPC for AP-Patterns as **important** (avg. Likert scale value: **4.3**)
Relationship between concerns and AP-Patterns	– Q7: **6 of 8** respondents find the documentation of the relationship between concerns & AP-Patterns as **helpful** – Q9: The respondents view data provided by the EAMPC for the relationship between concerns & AP-Patterns **in some degree** as **important** (avg. Likert scale value: **4.0**)
DC-Patterns	– Q1: **7 of 8** respondents perceive the documentation of DC-Patterns as **helpful** – Q3: The respondents view data provided by the EAMPC for DC-Patterns **in some degree more or less as important** (avg. Likert scale value: **3.8**)
Relationship between I- & DC-Patterns	– Q5: **5 of 8** respondents **prefer** the selection of **alternative 2**: one DC-Pattern per one class of an I-Pattern – Q7: **6 of 8** respondents find the documentation of the relationship between I- & DC-Patterns as **helpful** – Q9: The respondents view data provided by the EAMPC for the relationship between I- & DC-Patterns as **neither very important nor unimportant** (avg. Likert scale value: **3.1**)
V-Pattern extension	– Q1: **6 of 8** respondents perceive the extension of V-Patterns as **helpful** – Q3: The respondents view data provided by the EAMPC for the extended V-Patterns **in some degree more or less as important** (avg. Likert scale value: **3.7**)
Relationship between concerns and V-Patterns	– Q7: **4 of 8** respondents find the documentation of the relationship between concerns & V-Patterns as **helpful** – Q9: The respondents view data provided by the EAMPC for the relationship between concerns & V-Patterns as **neither very important nor unimportant** (avg. Likert scale value: **3.0**)

By summarizing the key findings of Table 5, all new EA management patterns and EA management pattern related elements are perceived as helpful (see Q1). Also, the extension of V-Patterns is appreciated by practice. This is also reflected by responses to the question Q3. The total average of the Likert scale value amounts to 4.1, which states that the data for new proposed EA management patterns and EA management pattern related elements provided by the EAMPC are noticed in some degree as important. This indicates that these patterns and elements can be of use for the respondents. The majority of the respondents also find the documentation of the new relationships as helpful (see Q7). The relationship between concerns and V-Patterns is perceived as less important than other relationships. The reason for their argumentation is that the EAMPC user has to know in which way a viewpoint is created. Therefore, they see M-Patterns as mandatory to address a specific concern. Turning back to the general findings of the proposed relationships, the Likert scale value amounts to 3.8 (see avg. value of all Q9). Somewhat, this value underlies the first presented Likert scale value, but still shows the importance of data provided by the EAMPC for relationships between new proposed EA management patterns and EA management pattern related elements. The reason why the second value performs poorly compared to the first one, lies also in the relationship between concerns and V-Patterns (see Q9). The reason is the same, as before, the methodology for the creation of a solution has to be replicable.

A more detailed analysis of the particular sections shows that the documentation of influence factors seems to be very reasonable. All respondents perceive the documentation of influence factors as helpful, because it provides a complementary view on the specific organization with external factors. Furthermore, the link between cause and effect can be clarified. Later, effects for the EA management function can be deduced. The majority of the respondents do not see respondents the necessity to categorize concerns or EA management

patterns regarding the typical EA management function maturity level. Consequently, data for the relationship between influence factors and concerns provided by the EAMPC is perceived as important for the respondents. In their opinion, this data helps them to check whether all dependencies between the company and its environment are recorded or not. Furthermore, it fosters the management of new applications and changing application landscapes.

Secondly, the documentation of stakeholders and their relationships between concerns and V-Patterns is also perceived as helpful. Furthermore, data provided by the EAMPC for stakeholders and their relationships are of importance for the interviewees. The reasons are of great significance. On the one hand, the documentation of stakeholder can be used as a basis for discussion and communication. Thereby, the EAMPC user is able to determine the addressed audience and to identify their needs. Furthermore, this indicates that the documentation of stakeholders is seen as decision and preparation guidance. On the other hand, the documentation of the relationship between stakeholders and concerns enables the EAMPC user to assess the potential concerns of a stakeholder. Additionally, it serves a clearer understanding of a stakeholder and its pain points. As mentioned before, the relationship between stakeholders and V-Patterns appears less important than the relationship between stakeholders and V-Patterns. Nevertheless, some respondents argue that the documentation of this relationship would support the solution making process.

Thirdly, Section 2.3.3.1. clearly illustrated that AP-Patterns are seen as important means for EA design and essential factor for the success of EA management function for practice. The survey reaches the same result. For all respondents, the documentation of AP-Patterns is helpful and the data for AP-Patterns provided by EAMPC important. The respondents see AP-Patterns as guidance for architecture related problems. A collection of AP-Patterns would provide them reference help for selecting appropriate architecture principles for their organization and completion of their already utilized architecture principles. The relationship between AP-Patterns and concerns is also perceived as helpful and in some degree as important. Respondents would use this relationship type as argumentation help for selecting a certain technology. Furthermore, it enables to classify concerns according to AP-Patterns and strengthens the awareness of this relationship type, because it was beforehand neglected.

Fourthly, 7 out of 8 respondents find the documentation of DC-Patterns helpful. However, the importance of the data for DC-Patterns provided by the EAMPC appears less important than other presented concepts. The reason for this appearance is that the DC-Patterns are for their opinion very specific to an organization. Another reason is that the value of benefit of the documentation of DC-Patterns seems to be questionable. Moreover, the relationship between I- and DC-Patterns exhibit less helpful and less important perception by the questionees. Once more, the relationship of a DC-Pattern with an I-Pattern is also argued as very specific for an organization, wherefore the documentation of this relationship type seems to be not reasonable. Another respondents see this relationship as important and helpful, because the documentation would provide more transparency in terms of responsibilities for data maintenance and source of data. As stated before, two alternatives for constructing a DC-Pattern were proposed. The majority of the respondents prefer to select the second alternative, one DC-Pattern per one class of an I-Pattern. Their argumentation is that this alternative facilitates to find the responsible person for the data maintenance of a class. Additionally, in their opinion, this alternative is more practical as classes of I-Patterns are stored in different locations.

Finally, the extension of V-Patterns by EA metrics and reports seems to be helpful for 6 of 8 respondents. Moreover, the data for extended V-Patterns provided by the EAMPC seems more or less important for the respondents. First of all, they find that reports and metrics belong to the toolbox of an EA management function, because they provide additional appropriate visualization types for their concerns. Further on, the documentation of EA metrics and reports provides them additional important templates and examples. Besides this, one respondent also argues that the documentation of EA reports would ease the identification of appropriate reports for EA management function related problems. However, interviewees are discordant regarding the relationship between concerns with V-Patterns. Half of the respondents find the documentation of this relationship type as helpful, whereas the other half does not. This is also reflected by the Likert scale value of this relationship, which amounts 3.0. As mentioned before, the main reason for this disagreement is that the solution process has to be clear, which indicates that M-Patterns are seen as obligatory for some respondents. The following section exemplifies the extended structure for documenting new EA management patterns and EA management pattern related elements.

5. EXEMPLARY EA MANAGEMENT PATTERN DOCUMENTATION

The data collection phase of the research approach identified several new EA management pattern candidates and EA management patterns related element candidates. Although, they are yet not checked whether they constitute best practices or not, several pattern and pattern related candidates could be observed frequently. This can be seen as an indication that a specific EA management pattern candidate or EA management pattern related element will highly probable observed in the main study at least in three different companies. With this consideration, for each introduced EA management pattern and EA management pattern related element, the documentation of one frequently observed candidate is exemplified. Further documentation examples are listed in the Appendix.

5.1 Exemplified EA management element documentations

The subsequent sections include documentation examples for introduced EA management patterns and EA management pattern related elements.

5.1.1 *Influence Factor – New Business Vision (IF-5).*

5.1.1.1. *Overview Section.*

Table 6 Influence Factor Overview

Influence Factor Overview	
Id	IF-5
Name	New business vision
Alias	Vision change
Summary	The term "Vision" refers to a possible and desirable future state of the organization [Campbell and Yeung 1991]. Changing market situations require companies to redefine their vision. In order to contribute to the vision, the EAM has to be aligned with the business strategy.
Version	2.0

5.1.1.2. *Hierarchical Reference Section.*

The Influence Factor has an influence on the concerns:

- **C-34:** How does the long-term vision, the target of the application landscape, look like?
- **C-104:** Determination of existing relationships between business applications and organizational units and of way of their visualization.
- **C-147:** Ways to merge two different application landscapes.

5.1.2 *Stakeholder - CIO (S-4).*

5.1.2.1. *Overview Section.*

Table 7 Stakeholder Overview

Stakeholder Overview	
Id	S-4
Name	CIO
Alias	CTO, CIO Management Board
Summary	The Chief Information Officer (CIO) is defined as the highest-ranking IS executive who typically exhibits managerial roles requiring effective communication with top management, a broad corporate perspective in managing information resources, influence on organizational strategy, and responsibility for the planning of IT to fit with a company's competitive environment [Grover et al. 1993].
Version	2.0

5.1.2.2. *Hierarchical Reference Section.*

The Stakeholder has the following concern:

- **C-167:** Context-sensitive information need.

This Stakeholder uses the viewpoints:

- **V-61:** Technical Project Portfolio Overview
- **V-114:** Product and Process Individuality Matrix

5.1.3 *AP-Pattern – Loose Coupling of Systems or Services (AP-10).*

5.1.3.1. *Overview Section.*

Table 8 AP-Pattern Overview

AP-Pattern Overview	
Id	P-10
Name	Loose Coupling of Systems or Services
Alias	
Summary	The service-orientation can be reached through the usage of loosely coupled services. The loose coupling of systems reduces dependencies between interface systems.
Version	2.0

5.1.3.2. *Problem Section.*

The Architecture Principle addresses the following concerns:

- **C-91:** The activities modifying the application landscape should be aligned to the needs, which have been specified by the defined strategies. Thereby, financial aspects and necessities dictated by the environment of the organization, e.g. via laws, regulations, etc. should be considered.
- **C-119:** Definition of target application landscape.
- **C-140:** Ways to manage IT systems dependencies better and to increase flexibility in IT systems.

5.1.3.3. *Solution Section.*

The loose coupling of systems contributes to the concerns above with the reduction of system dependencies and application landscape complexity. With this contribution, the management of dependencies between IT systems can be improved and the flexibility of IT systems can be increased (C-140). Additionally, individual services or systems are easy substitutable, and as a consequence, they are more flexible to changing business needs (C-91). Furthermore, target application landscape requirements can be ensured (C-119).

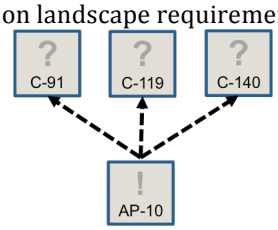

Fig. 7. Related concerns of AP-Pattern *Loose Coupling of Systems or Services*

5.1.3.4. *References to other Standards Section.*

See also:

- This AP-Pattern AP-10 constitutes a base for the AP-Pattern AP-17 and for the Principle 6: Service Orientation in Architecture Principles in TOGAF [The Open Group 2013b].

5.1.4 *DC-Pattern – Capability (DC-6).*

5.1.4.1. *Overview Section.*

Table 9 DC-Pattern Overview

DC-Pattern Overview	
Id	DC-6
Name	Capability
Alias	

Summary	This DC-Pattern contains the class Capability for the I-Pattern Domain Model (I-100).
Version	2.0

5.1.4.2. *Solution Section.*

The data maintenance for the class *Capability* is done manually. The enterprise architect provides the data source for required information for this DC-Pattern. The data is updated ad hoc, or when attributes of the class Capability are changed or new ones are added.

The following stakeholder is responsible for data maintenance:

– **S-45:** Enterprise Architect

Fig. 8. Related I-Pattern of DC-Pattern *Capability*

Capabilities are used by the following I-Pattern:

– **I-100:** Domain Model

6. REFLECTION AND OUTLOOK

At the beginning of this paper, the PDR method and the pattern-based approach to EA management were described. Later, the pursued target of this approach was presented, that inter alia aims to complement existing EA management frameworks. Afterwards, the EAMPC and the EA management pattern language elements were described. A motivation section revealed the necessity to extend the current structure for documenting EA management patterns and to introduce new pattern types and concepts like AP-Patterns, stakeholders and influence factors. On that account, an existing conceptual model, describing the relationships between EA management patterns and EA management pattern related elements, was envisioned. Based on this model, each introduced classes, associations and attributes were characterized and then motivated for their documentation benefit. In Section 3, the research approach for the development of the second version of the EAMPC was described. Within this research approach, the essential role of this paper was highlighted. Subsequently, the results of the conducted survey for the evaluation of new introduced EAMPC concepts were delineated. By summarizing the key finding of the survey, all introduced EA management patterns and EA management pattern related elements were perceived as helpful. All respondents viewed data for these patterns and elements provided by the EAMPC as important. Some new introduced relationships like the relationship between concerns and V-Patterns were not equally viewed as important as other introduced relationships.

The utilization of new introduced EA management patterns and EA management pattern related elements for complementing existing EA management frameworks constitutes an interesting area for further research. The proposed documentation elements and structures have to be evaluated in the main study of this research approach and as the circumstances require adjustment. Due to the fact that in the context of EA management function many stakeholders exist, a classification of them seems to be necessary in order to prevent inconsistencies. For instance, TOGAF tries to address this problem by classifying EA management function stakeholders in five broad categories. Although, only the minority of the respondents saw the necessity to categorize concerns or EA management patterns regarding the typical EA management function maturity level, the application of a maturity model-based approach seems to be valuable, because it provides a directed selection of EA management patterns for addressing concerns. For that reason, a future area of research might be the identification and adoption of an appropriate maturity model that can then be provided by the EAMPC. Therein, concerns might be categorized according to the specific maturity levels. Finally, another area for future research might be developing a concept for assessing the cost-benefit ratio for using a specific EA management pattern.

APPENDIX

A.1 Asked questions in the pre-study survey for evaluating proposed concepts

Table 10 Question structure for evaluating EAMPC 2015 concept

Concrete new EAMPC Concept	Questions for evaluating a new specific EAMPC Concept
Influence factors, stakeholders, AP-Patterns, DC-Patterns, and V-Pattern extension	– Q1: Would the documentation of the new concept, e.g. influence factors, help you, if it would be linked with e.g. concerns? – Q2: Why it would be useful/not useful for you? – Q3: On a 5-level a Likert scale ranging from "not important" (1) through "very important" (5), how important would be the data, e.g. influence factors provided by the EAMPC?
Influence factors	– Q4: Do you see the necessity to categorize concerns or EA management patterns regarding the typical EA management function maturity level?

DC-Patterns	– For the creation of DC-Patterns, there exist the following two alternatives:
	  – Alternative 1: 1 DC-Pattern pro I-Pattern
	  – Alternative 2: 1 DC-Pattern per one class of an I-Pattern
	– Q5: Which alternative do you would prefer?
	– Q6: What is your reasoning for your selection?

Relationships (RSs) between influence factors & concerns, RSs between stakeholders & concerns, RSs between stakeholders & V-Patterns, RSs between concerns & AP-Patterns, RSs between I- & DC-Patterns, and RSs between concerns & V-Patterns	– Q7: Would the relationship between, e.g. relationship influence factors and concerns help you? – Q8: Why it would be useful/not useful for you? – Q9: On a 5-level a Likert scale ranging from "not important" (1) through "very important" (5), how important would be the data, e.g. relationship between influence factors and concerns provided by the EAMPC?

One important point to mention is that the generally stated new concept on the right side of Table 5, e.g. in the first question, is substituted by the concrete concept provided on the left hand side of Table 5.

A.2 Exemplified EA management pattern related element documentations

A.2.1 *Influence Factor – Competition (IF-4).*

A.2.1.1. *Overview Section.*

Table 11 Influence Factor Overview

Influence Factor Overview	
Id	IF-4
Name	Competition
Alias	
Summary	The term "Competition" is defined by a process of responding to a new force and is also described as a method of reaching a new equilibrium - it furthermore connotes the rivalry of two or more persons [Stigler 1957].
Version	2.0

A.2.1.2. *Hierarchical Reference Section.*

The Influence Factor affects the following concern:

– **C-122:** Evaluation of alignment between application landscape and business model.

A.2.2 *Stakeholder – IT Management (S-3).*

A.2.2.1. *Overview Section.*

Table 12 Stakeholder Overview

Stakeholder Overview	
Id	S-3
Name	IT Management
Alias	IT Coordinator, IT Manager, IT Executive, Corporate IT
Summary	The IT Management's efforts involve planning, organizing, controlling and directing the introduction of IT within a company [Boynton and Zmud 1987].
Version	2.0

A.2.2.2. *Hierarchical Reference Section.*

The Stakeholder has the following concerns:

- **C-5:** Which activities or projects have to be started, in order to increase conformance to standards? What has to be done in order to modify the current business applications to increase their conformance to standards and reduce heterogeneity?
- **C-29:** At the beginning of a planning period the available IT budget has to be assigned to project proposals. Project proposals that will be approved have to be selected, others have to be rejected or delayed.
- **C-124:** Reducing and measurement of application landscape complexity.
- **C-142:** Determination of which business applications are necessary to support business capabilities. Assignment of Applications to functional domains. Identification and description of hierarchical degree of functional domains.
- **C-150:** Requirement for amendment of the application landscape.
- **C-153:** Overview of end of lifecycle applications.
- **C-154:** Verification of application relevance.
- **C-160:** Overview of functionalities, supporting systems and technical demand for action. Creation of inventories of applications with their components and functions.
- **C-161:** Checking required and used functionalities.
- **C-168:** Architectural assessment of architecture request.
- **C-169:** Architectural assessment of change request or activity. Analysis and planning of system changes or adoptions.

This Stakeholder uses the viewpoints:

- **V-17:** Process Support Map
- **V-110:** Domain Model Overview
- **V-181:** Functions and Applications
- **V-184:** End of Lifecycle Applications Report
- **V-213:** Application Landscape Complexity

A.2.3 *Concern – C-124.*

A.2.3.1. *Overview Section.*

Table 12 Concern Overview

Concern Overview	
Id	C-124
EAM Topic	Application Landscape Planning
Summary	Reducing and measurement of application landscape complexity
Version	2.0

A.2.3.2. *Hierarchical Reference Section.*

This concern is related to the following Stakeholders:

- **S-3:** IT-Management
- **S-45:** Enterprise Architect

The concern is addressed by the M-Patterns:

- **M-67:** Centralization of Application Logic
- **M-68:** Early Determination of Requirements in Target Architecture
- **M-69:** Management of Application Landscape Complexity
- **M-70:** Ascertainment of Application Landscape Complexity Indicators
- **M-71:** Elimination of Functional Redundancies
- **M-72:** Counting Applications
- **M-73:** Optimization of Complexity Drivers
- **M-74:** Looking at Total Cost of Ownership on various Levels
- **M-91:** Introduction of an Integration Platform
- **M-126:** Ascertainment of Actual State only in so far as relevant for Attainment of Target State

The concern is addressed by the Architecture Principle:

- **P-9:** Technology Portfolio is based on few Technologies

This concern is addressed by the following viewpoint:

- **V-118:** Complexity Measurement Dashboard

A.2.4 *Concern – C-170.*

A.2.4.1. *Overview Section.*

Table 13 Concern Overview

Concern Overview	
Id	C-170
EAM Topic	
Summary	Identification of types and relationships that are used in the modeling of an enterprise architecture.
Version	2.0

A.2.4.2. *Hierarchical Reference Section.*

The concern is addressed by the M-Pattern:

– **M-116:** Creation of an integrated Enterprise Architecture Meta-Model

This concern is addressed by the following viewpoint:

– **V-111:** Enterprise Architecture Meta-Model Overview

A.3 Exemplified EA management pattern documentations

A.3.1 *Ascertainment of Application Landscape Complexity Indicators (M-70)*

A.3.1.1. *Overview Section.*

Table 14 M-Pattern Overview

M-Pattern Overview	
Id	M-70
Name	Ascertainment of Application Landscape Complexity Indicators
Alias	
Summary	The methodology determines complexity indicators of an application landscape as for instance, counting the number of involved business applications, or of used interfaces.
Version	2.0

A.3.1.2. *Problem Section.*

The M-Pattern addresses the following concern:

– **C-124:** Reducing and measurement of application landscape complexity.

A.3.1.3. *Solution Section.*

The methodology addresses the concern above by ascertaining the complexity indicators of the application landscape. Thereby the complexity of an application landscape can be investigated through taking the following indicators into account:

– Number of involved business applications
– Type and number of used interfaces
– Degree of the used technological standards
– Technology variety
– Interactions of the business applications through interfaces
– Number of released IT products

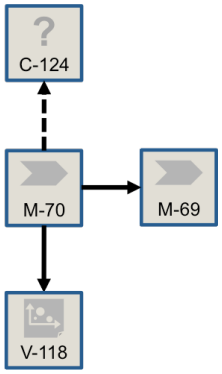

Fig. 9. Related concern, M- and V-Pattern of M-Pattern *Ascertainment of Application Landscape Complexity Indicators*

More importantly, the analysis of functional redundancies and redundancies in data storage are also part of the ascertainment of application landscape complexity which should be considered.

The M-Pattern uses the following viewpoint:

– **V-118:** Complexity Measurement Dashboard

This methodology delivers indicator values for the Complexity Measurement Dashboard, which juxtaposes functional domains in opposition to the analyzed IT complexity indicators in order to determine the complexity of an application landscape.

The results of this M-Pattern is used by the following M-Pattern:

– **M-69:** Management of Application Landscape Complexity

In order to overcome the management of the application landscape complexity (M-69), complexity indicators of the application landscape have to be defined and analyzed. Furthermore, dependencies between various technical aspects also have to be determined.

A.3.2 *Creation of an integrated Enterprise Architecture Meta-Model (M-116)*

A.3.2.1. *Overview Section.*

Table 15 M-Pattern Overview

M-Pattern Overview	
Id	M-116
Name	Creation of an integrated Enterprise Architecture Meta-Model
Alias	
Summary	This M-Pattern describes the way for the creation of a meta-model of the enterprise architecture on high abstraction view.
Version	2.0

A.3.2.3. *Problem Section.*

The M-Pattern addresses the following concerns:

- **C-170:** Identification of types and relationships which are used in the modeling of an enterprise architecture.

A.3.2.4. *Solution Section.*

This M-Pattern suggests the creation of an enterprise architecture meta-model, which contains all relevant enterprise architecture objects and describes the relation of these objects among themselves on abstraction level. For this purpose, initially, the used architecture objects are identified. These are for instance the organizations, business processes, business applications, information flows, business objects, business domains or services. In a next step, the relationships between these objects have to be determined. In the last step, the architecture objects are clustered in to specific domains. The domains can be classified across different aspects:

- Differentiation in diverse architecture levels, such as functional architecture level, infrastructure application level and services level
- Differentiation in technology hierarchy levels, such as business layer, application layer, and infrastructure layer

Fig. 10. Related concern and V-Pattern of M-Pattern *Creation of an integrated Enterprise Architecture Meta-Model*

The purpose of this M-Pattern can be tool-driven or demand-driven. In the first case, an EAM repository models the intended enterprise architecture objects with their relationships. It also provides the possibility to adjust the enterprise architecture objects easily. In the latter case, the information models of the architecture objects are derived by the requirements of the EAM and based on them, an integrated meta-model of the architecture objects can be generated. In this instance, the UML notation can be used in order to create the architecture meta-model. For the representation of the architecture objects, a typical rectangle with a name description can be used. For the specification of the relationships UML associations and multiplicities can be utilized. The edges represent the relationships. To prevent the complexity of the meta-model only the necessary architecture objects and their relationships should be taken into account. Otherwise, the model becomes less abstract and the intention of the meta-model gets lost.

The M-Pattern uses the following viewpoints:

– **V-111:** Enterprise Architecture Meta-Model Overview

The V-Pattern V-111 represents the created architecture meta-model with all of its components and their relationships among themselves.

A.3.2.5. *Consequence Section.*

As a consequence of this M-Pattern all relevant architecture objects and their relationships have to be identified and be set in correlation.

A.3.2.6. *References to other Standards Section.*

See also:

– Chapter 34 Content Metamodel in TOGAF provides information for the creation and management of an enterprise architecture meta-model [The Open Group 2013b].

A.3.3 *Technology Portfolio is based on few Technologies (AP-9)*

A.3.3.1. *Overview Section.*

Table 16 AP-Pattern Overview

AP-Pattern Overview	
Id	AP-9
Name	Technology Portfolio is based on few Technologies
Alias	
Summary	In order to stem the increasing application landscape complexity, the technology portfolio of the organization should only contain few technologies.
Version	2.0

A.3.3.2. *Problem Section.*

The Architecture Principle addresses the following concern:

– **C-124:** Reducing and measurement of application landscape complexity.

A.3.3.3. *Solution Section.*

For the reduction of the application landscape complexity, the amount of used technologies should be reduced on a desirable minimum. Nevertheless, the technology portfolio must be able to satisfy the business needs.

Fig. 11. Related concern of AP-Pattern *Technology Portfolio is based on few Technologies*

A.3.3.4. *Consequence Section*.

As a consequence of this Architecture Principle, information about used technologies has to be collected and a technology portfolio has to be created and analyzed. Afterwards redundant technologies can be examined and then removed from the technology portfolio. An excessive reduction of technologies might lead to sub-optimal solutions. Thus, the right stakeholders need to be involved to create a set of standards that meet the business needs of the organization.

A.3.3.5. *References to other Standards Section*.

See also:

– Principle 20: Control Technical Diversity in Architecture Principles in TOGAF [The Open Group 2013b].

A.3.4 *Enterprise Architecture Meta-Model Overview (V-111)*

A.3.4.1. *Overview Section.*

Table 17 V-Pattern Overview

V-Pattern Overview	
Id	V-111
Type	Visualization
Name	Enterprise Architecture Meta-Model Overview
Alias	
Summary	This V-Pattern visualizes the types and relationships used to document the enterprise architecture on high abstraction level.
Version	2.0

A.3.4.2. *Problem Section.*

The V-Pattern addresses the following concern:

– **C-170:** Identification of types and relationships which are used in the modeling of an enterprise architecture.

A.3.4.3. *Solution Section.*

After the application of the M-Pattern M-116, the following viewpoint is created.

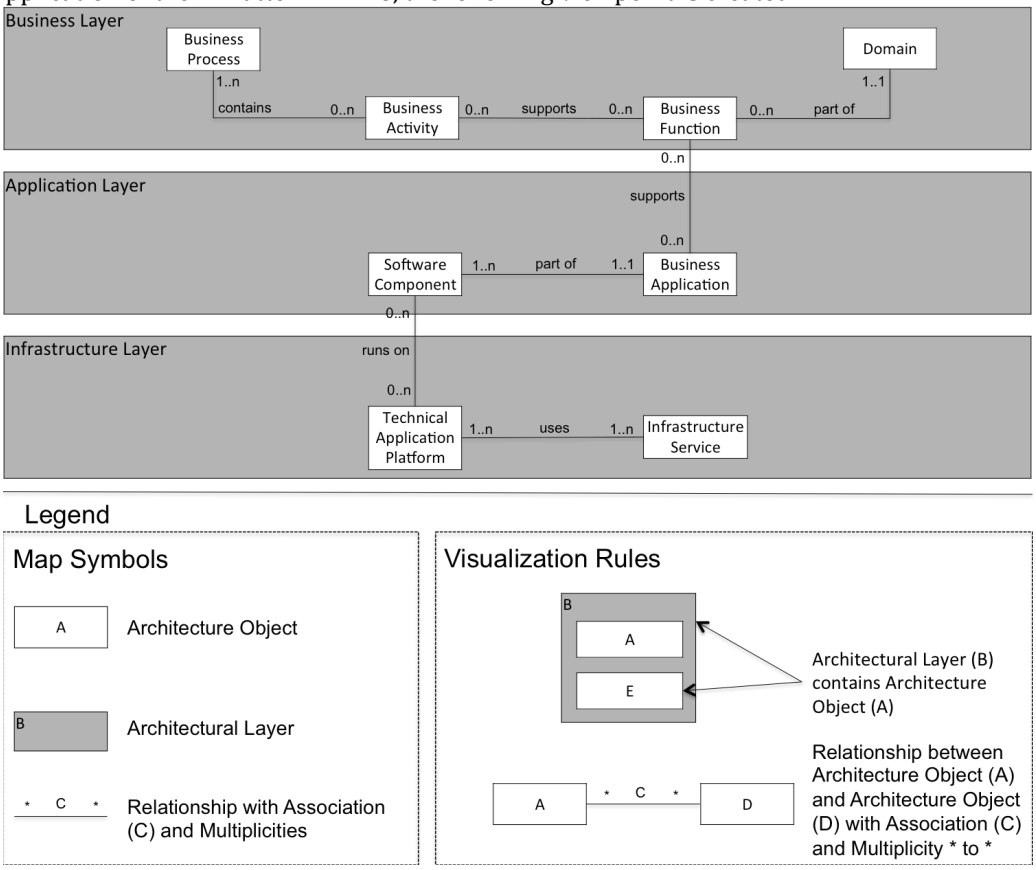

Fig. 12. Exemplary view for V-Pattern Enterprise Architecture Meta-Model Overview

This V-Pattern visualizes the enterprise architecture objects and their relationships among themselves in an integrated way on a high abstraction level. Naturally, further enterprise architecture objects and relationships exist. They can substitute or complement the types and relationships as provided in the previous visualization.

Fig. 13. Related M- and I-Pattern of V-Pattern *Enterprise Architecture Meta-Model Overview*

This V-Pattern is based on the I-Pattern:

– **I-101:** Architecture Objects and Architecture Object Relationships

A.3.4.4. *References to other Standards Section.*

See also:

– This V-Pattern can be compared with the Full Metamodel of TOGAF. In contrast to the V-Pattern illustrated above, TOGAF's meta-model also sets architecture principles, requirements, and roadmaps with the core entities of an enterprise architecture in relation. Furthermore, it extends the core architecture objects and relationships with extension aspects such as motivation extensions, governance extensions or data extensions [The Open Group 2013b].
– ArchiMate does not provide an integrated meta-model of the distinct architectural layers. Nevertheless, the meta-models for the business layer, application layer and technology layer can be found separately in ArchiMate [The Open Group 2013a].

A.3.5 *Complexity Measurement Dashboard (V-118)*

A.3.5.1. *Overview Section.*

Table 18 V-Pattern Overview

V-Pattern Overview	
Id	V-118
Type	Visualization
Name	Complexity Measurement Dashboard
Alias	
Summary	The complexity measurement dashboard juxtaposes functional domains in opposition to the analyzed technological aspects in order to determine the complexity of an application landscape.
Version	2.0

A.3.5.2. *Problem Section.*

The V-Pattern addresses the following concern:

– **C-124:** Reducing and measurement of application landscape complexity.

A.3.5.3. *Solution Section.*

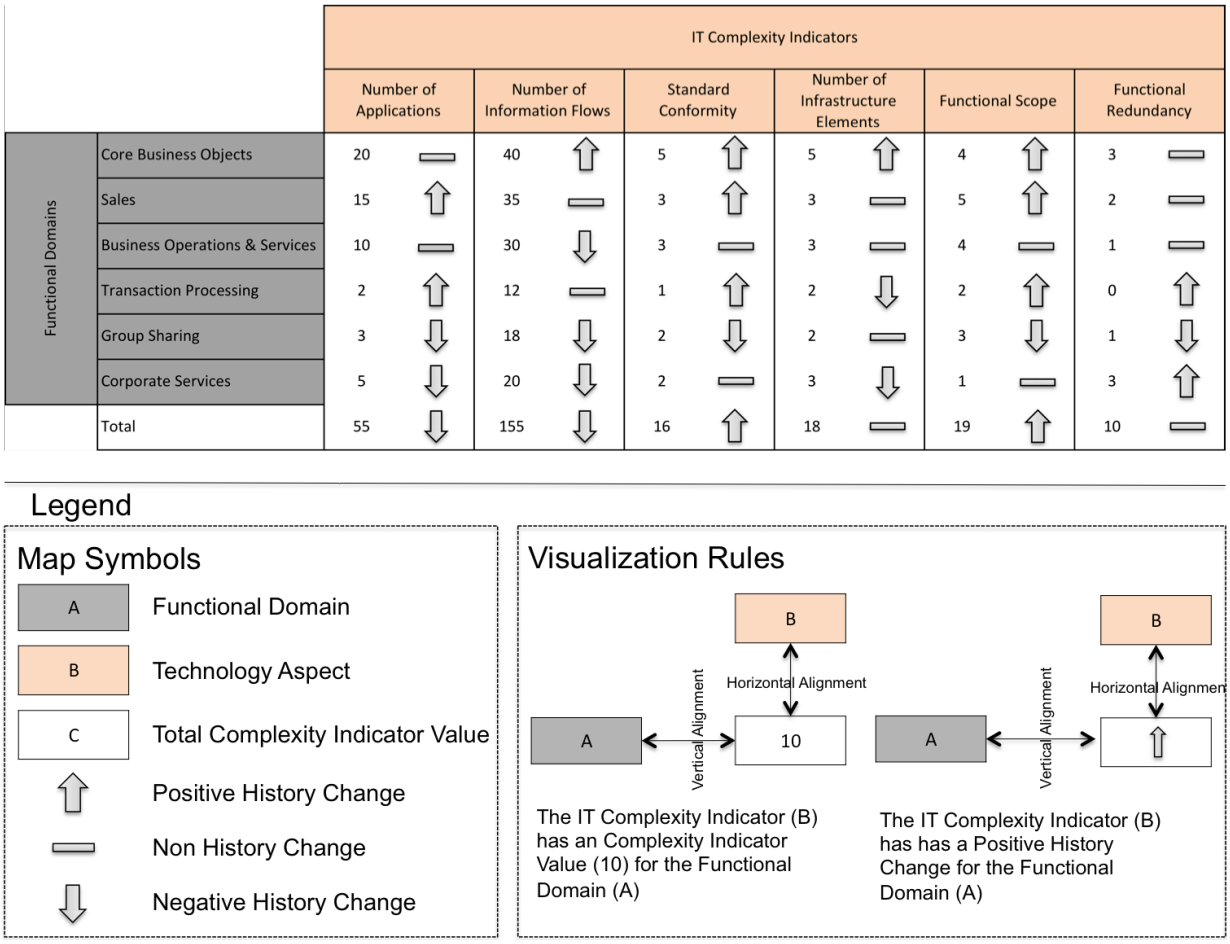

Fig. 14. Exemplary view for V-Pattern Complexity Measurement Dashboard

After the analysis of application landscape complexity indicators (M-70), they are listed as IT complexity indicators which then can be linked with the corresponding functional domains. For each functional domain

and complexity indicator, a value for the complexity is listed. Additionally, a history of the complexity change is also provided. As a result, this V-Pattern visualizes the total complexity of the application landscape.

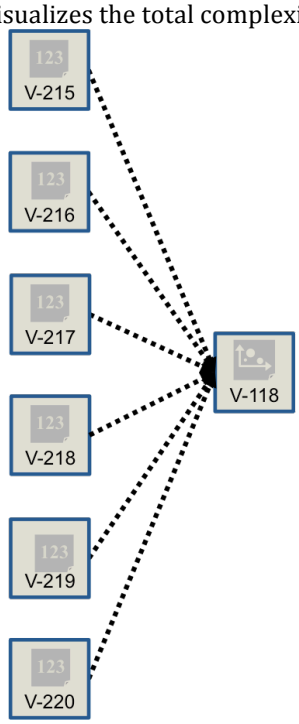

Fig. 15. Constituent V-Patterns for compound V-Pattern *Complexity Measurement Dashboard*

A.3.5.4. *Compounds Section.*

This V-Pattern represents a compound pattern for application landscape complexity aspects. Possible V-Patterns, which can be used as constituents for this compound pattern, are the following:

- **V-215:** Number of Applications
- **V-216:** Number of Information Flows
- **V-217:** Standard Conformity
- **V-218:** Number of Infrastructure Elements
- **V-219:** Functional Scope
- **V-220:** Functional Redundancy

A.3.6 *Number of Infrastructure Elements (V-218)*

A.3.6.1. *Overview Section.*

Table 19 V-Pattern Overview

V-Pattern Overview	
Id	V-218
Type	Metric
Name	Number of Infrastructure Elements
Alias	
Summary	This V-Pattern represents a metric and it returns the number of infrastructure components used to realize a business application.
Version	2.0

A.3.6.2. *Problem Section.*

The V-Pattern addresses the following concern:

– **C-124:** Reducing and measurement of application landscape complexity.

A.3.6.3. *Solution Section.*

This V-Pattern can be used in order to determine the complexity of an application landscape. A higher number of infrastructure elements indicate a higher complexity. In combination with the other complexity metrics: Number of Applications (V-215), Number of Information Flows (V-216), Standard Conformity (V-217), Functional Scope (V-219) and Functional Redundancy (V-220), the complexity of an application landscape can be reduced and measured.

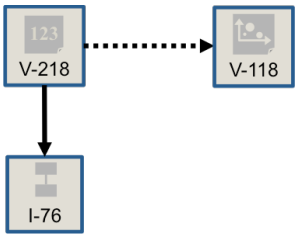

Fig. 16. Related I- and compound V-Pattern of V-Pattern *Number of Infrastructure Elements*

This V-Pattern is based on the I-Pattern:

– **I-76:** Infrastructure Usage

A.3.6.4. *Compounds Section.*

This V-Pattern can be aggregated for the following compound V-Pattern:

– **V-118:** Complexity Measurement Dashboard

A.3.7 *Infrastructure Usage (I-76)*

A.3.7.1. *Overview Section.*

Table 20 I-Pattern Overview

I-Pattern Overview	
Id	I-76
Name	Infrastructure Usage
Alias	
Summary	This I-Pattern contains the information models for infrastructure components and business applications.
Version	2.0

A.3.7.2. *Solution Section.*

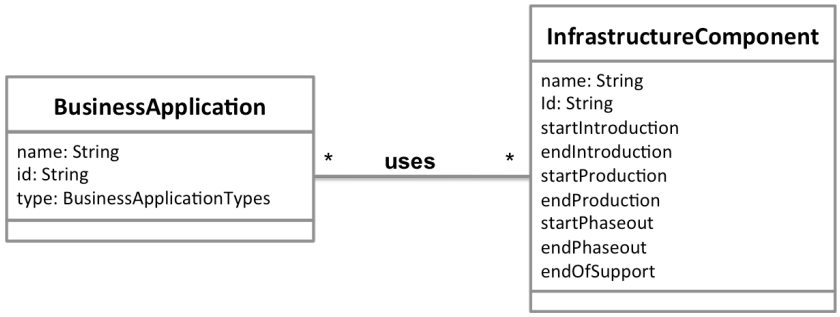

Fig. 17. Information model fragment for I-Pattern *Infrastructure Usage*

- BusinessApplication: A business application is a software system, which is part of an information system of an organization. An information system is according to [Krcmar 2005] understood as a sociotechnical system, which is, besides the software system, made up of the infrastructure the software system is based on, and a social component, namely the employees or stakeholders concerned with it. Thereby, infrastructure and social component are not considered as belonging to the business application, while the characterization "business" restricts the term to applications that support at least one process of the respective organization. Thus, business application denotes here an actual deployment of a software.
- InfrastructureComponent: Infrastructure components are deployed middleware or hardware systems e.g. a database management system.

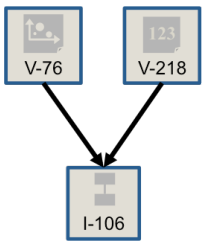

Fig. 18. Related V-Patterns of I-Pattern *Infrastructure Usage*

This I-Pattern provides required information for the V-Patterns:

- **V-76:** Technology Usage
- **V-218:** Number of Infrastructure Elements

A.3.7.3. *References to other Standards Section.*

See also:

- Exact equivalent definitions and descriptions with corresponding information models for the both classes BusinessApplication and InfrastructureComponent do not exist in ArchiMate. Nevertheless, similar descriptions of the classes above are Application Component and System Software in ArchiMate [The Open Group 2013a].

A.3.8 *Architecture Objects and Architecture Object Relationships (I-101)*

A.3.8.1. *Overview Section.*

Table 21 I-Pattern Overview

I-Pattern Overview	
Id	I-101
Name	Architecture Objects and Architecture Object Relationships
Alias	
Summary	This I-Pattern contains the information model for architecture meta-objects and their relationships.
Version	2.0

A.3.8.2. *Solution Section.*

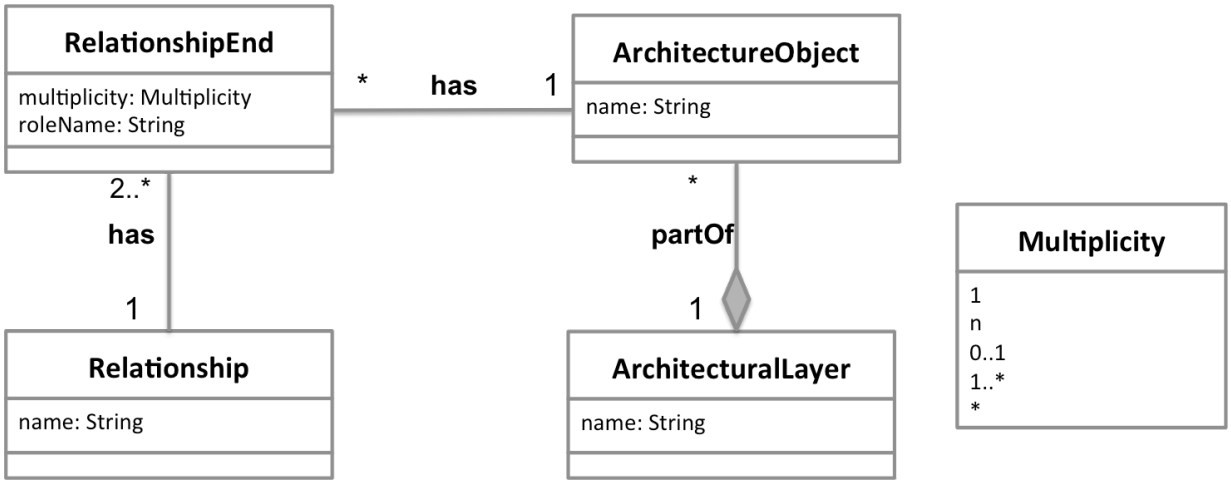

Fig. 19. Information model fragment for I-Pattern *Architecture Objects and Architecture Object Relationships*

- ArchitecturalLayer: Describes a logical grouping into areas relevant to architecture levels or technology architecture level, e.g. application layer, infrastructure layer.
- ArchitectureObject: An ArchitectureObject specifies the architecture meta-object, e.g. business application, business process, business objects or business domains.
- Multiplicity: A multiplicity specifies the number of possible occurrences of an element.
- Relationship: A relationship describes a connection between elements.
- RelationshipEnd: A relationshipEnd specifies the multiplicity of elements that can be connected to it and defines the role name of the connected elements.

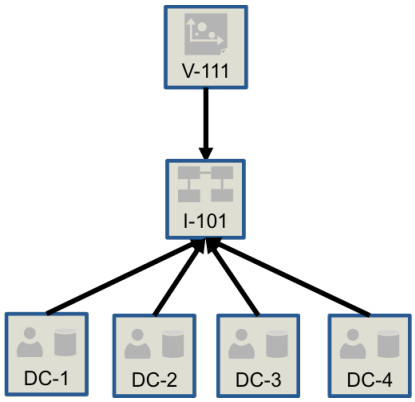

Fig. 20. Related V- and DC-Patterns of I-Pattern *Architecture Objects and Architecture Object Relationships*

This I-Pattern contains the required Information Models for the V-Pattern:

– **V-111:** Enterprise Architecture Meta-Model Overview

The classes of this I-Pattern are the following DC-Patterns:

– **DC-1:** Relationship End
– **DC-2:** Relationship
– **DC-3:** Architecture Object
– **DC-4:** Architectural Layer

A.3.9 *Business Domain (DC-5)*

A.3.9.1. *Overview Section.*

Table 22 DC-Pattern Overview

DC-Pattern Overview	
Id	DC-5
Name	Business Domain
Alias	
Summary	This DC-Pattern contains the class Business Domain for the I-Pattern Domain Model (I-100).
Version	2.0

A.3.9.2. *Solution Section.*

The data maintenance for the class *Business Domain* is done manually by the analysis of business functions and system functions. The data is updated annually.

The following stakeholder is responsible for the data maintenance:

– **S-121:** Business Architecture Management

Fig. 21. Related I-Pattern of DC-Pattern *Business Domain*

Business domains are used by the following I-Pattern:

– **I-100:** Domain Model

REFERENCES

Stephan Aier, Bettina Gleichauf, and Robert Winter. 2011. Understanding Enterprise Architecture Management Design – An Empirical Analysis. In *Proceedings of the 10th International Conference on Wirtschaftsinformatik WI 2011*. Wirtschaftsinformatik, Zurich, 637–644.

Stephan Aier, Christian Riege, and Robert Winter. 2008. Unternehmensarchitektur – Liteaturüberblick und Stand der Praxis. *Wirtschaftsinformatik*. 50, 4 (2008), 292–304.

Christopher Alexander, Sara Ishikawa, and Murray Silverstein. 1977. A Pattern Language. University of Oxford Press, New York, USA.

Frank J. Armour, Stephen H. Kaisler, and Simon Y. Liu. 1999. Building an Enterprise Architecture Step by Step. *IT Professional*. 1, 4 (1999), 31–39.

Hartmut Bossel. 1994. Modeling and simulation. AK Peters Wellesley, MA, USA.

Andrew C. Boynton and Robert W. Zmud. 1987. Information Technology Planning in the 1990's: Directions for Practice and Research. *Journal of Management Information Systems Quarterly*. 11, 1 (1987), 59–71.

Sabine Buckl, Alexander M. Ernst, Josef Lankes, Kathrin Schneider, and Christian M. Schweda. 2007. A Pattern based Approach for constructing Enterprise Architecture Management Information Models. In *Proceedings of the 8th International Conference on Wirtschaftsinformatik*. Wirtschaftsinformatik, Karlsruhe, Germany, 145–162.

Sabine Buckl, Alexander M. Ernst, Josef Lankes, and Florian Matthes. 2008a. Enterprise Architecture Management Pattern Catalog (version 1.0, February 2008). Chair for Informatics 19 (sebis), Technische Universität München, Munich, Germany.

Sabine Buckl, Alexander M. Ernst, Josef Lankes, Florian Matthes, and Christian M. Schweda. 2008b. Enterprise Architecture Management Patterns – Exemplifying the Approach. In *Proceedings of 12th International IEEE Enterprise Distributed Object Computing Conference*. IEEE Computer Society, Munich, Germany, 393–402.

Sabine Buckl, Alexander M. Ernst, Florian Matthes, and Christian M. Schweda. 2009a. An Information Model for Managed Application Landscape Evolution. J*ournal of Enterprise Architecture,* 5, 1 (2009), 12–26.

Sabine Buckl, Alexander M. Ernst, Florian Matthes, René Ramacher, and Christian M. Schweda. 2009b. Using Enterprise Architecture Management Patterns to complement TOGAF. In *Proceedings of Enterprise Distributed Object Computing Conference*. IEEE Computer Society, Los Alamitos, CA, USA, 34–41.

Sabine Buckl, Florian Matthes, Sascha Roth, Christopher Schulz, and Christian M. Schweda 2010a. A Conceptual Framework for Enterprise Architecture Design. In *Proceedings of the 5th International Workshop on Trends in Enterprise Architecture Research*. Springer, Delft, Netherlands, 44–56.

Sabine Buckl, Florian Matthes, and Christian M. Schweda. 2010b. A Design Theory Nexus for Situational Enterprise Architecture Management. In *Proceedings of the 14th International IEEE Enterprise Distributed Object Computing Conference*. IEEE Computer Society, Vitoria, Brazil, 3–8.

Sabine Buckl, Florian Matthes, and Christian M. Schweda. 2010c. A situated Approach to Enterprise Architecture Management. In *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*. IEEE, 587–592.

Sabine Buckl, Thomas Dierl, Florian Matthes, and Christian M. Schweda. 2010d. Building blocks for Enterprise Architecture Management solutions. In *Practice-Driven Research on Enterprise Transformation*, Frank Harmsen, Erik Proper, Frank Schalkwijk, Joseph Barijs, and Sietse Overbeek (Eds.). Springer, Berlin, Heidelberg, Germany, 17–46.

Sabine Buckl, Florian Matthes, and Christian M. Schweda. 2010e. Future Research Topics in Enterprise Architecture Management – A Knowledge Management Perspective. *Journal of Enterprise Architecture*. 6, 3 (2010), 16–27.

Sabine Buckl, Florian Matthes, and Christian M. Schweda. 2011. A Method Base for Enterprise Architecture Management. In *Engineering Methods in the Service-Oriented Context*, Jolita Ralyté, Isabelle Mirbel, and Rebecca Deneckère (Eds.). Springer, Heidelberg, Germany, 34–48.

Sabine Buckl, Florian Matthes, and Christian M. Schweda. 2012. Designing Enterprise Architecture Management Functions – The Interplay of Organizational Contexts and Methods. In *Trends in Enterprise Architecture Research and Practice-Driven Research on Enterprise Transformation: 7th Workshop, TEAR 2012, and 5th Working*, Stephan Aier, Mathias Ekstedt, Florian Matthes, Erik Proper, Jorge L. Sanz (Eds.). Springer, Berlin, Germany, New York, USA, 236–252.

Sabine Buckl, Florian Matthes, Alexander W. Schneider, and Christian M. Schweda. 2013. Pattern-based Design Research – An Iterative Research Method Balancing Rigor and Relevance. In *Design Science at the Intersection of Physical and Virtual Design*, Jan Vom Brocke, Riitta Hekkala, Sudha Ram, and Matti Rossi (Eds.). Springer, Heidelberg, Germany, 73–87.

Josef M. Burger. 2013. *Modeling Application Landscapes as Dynamic Systems*. Master's thesis. Technische Universität München, Munich, Germany.

Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, and Michael Stal. 1996. Pattern Oriented Software Architecture: A System of Patterns. John Wiley & Sons, Inc., New York, NY, USA.

Frank Buschmann, Kevin Henney, and Douglas C. Schmidt. 2007. Pattern Oriented Software Architecture Volume 5: On Patterns and Pattern Languages. John Wiley & Sons, Inc., Chichester, West Sussex, England, United Kingdom.

Chief Information Officers Council. 1999. Federal Enterprise Architecture Framework Version 1.1.

Andrew Campbell and Sally Yeung. 1991. Brief case: Mission, vision and strategic intent. In *Long Range Planning*, 24, 4 (1991), 145–147.

James O. Coplien. 1996. Software Patterns: Management Briefs. Cambridge University Press, Cambridge, MA, USA.

Alexander Ernst. 2008. Enterprise Architecture Management Patterns. In *Proceedings of the 15th Pattern Languages of Programs*. ACM, New York, NY, USA, 1–20.

Alexander Ernst. 2010. *A Pattern-Based Approach to Enterprise Architecture Management*. Ph.D. Dissertation, Technische Universität München, Munich, Germany.

Christian Fischer, Robert Winter, and Stephan Aier. 2010. What Is an Enterprise Architecture Principle? Towards a Consolidated Definition. In *Computer and Information Science*, Roger Lee (Ed.). Springer, Berlin, Heidelberg, Germany, 193–205.

Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. 1995. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley Professional, Reading, MA, USA.

Varun Grover, Seung-Ryul Joeng, William J. Kettinger, and Choong C. Lee. 1993. The Chief Information Officer: A Study of Managerial Roles. *Journal of Management Information Systems*, 10, 2 (1993), 107–130.

Mohammad Kazem Haki and Christine Legner. 2013. Enterprise Architecture Principles In Research And Practice: Insights from An Exploratory Analysis. In *Proceedings of the 21st European Conference on Information Systems (2013)*. ECIS, Utrecht, Netherlands.

Matheus Hauder, Sascha Roth, Christopher Schulz, and Florian Matthes. 2013. Current Tool Support for Metrics in Enterprise Architecture Management. In *Proceedings of DASMA Software Metrik Kongress (Metrikon 2013)*.Metrikon, Kaiserslautern, Germany.

ISO/IEC/IEEE Systems and Software Engineering – Architecture description. 2011. ISO/IEC/IEEE Standard 42010:2011.

Helmut Krcmar. 2005. Informationsmanagement. Springer, Berlin, Germany.

Florian Matthes, Sabine Buckl, Jana Leitel, and Christian M. Schweda. 2008. Enterprise Architecture Management Tool Survey 2008. Chair for Informatics 19 (sebis), Technische Universität München, Munich, Germany.

Florian Matthes, Ivan Monahov, Alexander Schneider, and Christopher Schulz. 2012. EAM KPI Catalog (version 1.0). Chair for Informatics 19 (sebis), Technische Universität München, Munich, Germany.

Eetu Niemi. 2007. Enterprise Architecture Stakeholders - a Holistic View. In *Proceedings of the 13th Americas Conference on Information Systems*. Keystone, CO, USA.

Bas van der Raadt, Sander Schouten, and Hans van Vliet. 2008. Stakeholder Perception of Enterprise Architecture. In *Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg, Germany, 19–34.

Blaize H. Reich and Izak Benbasat. 2000. Factors that influence the social dimension of alignment between business and information technology objectives. *Management Information Systems Quarterly*. 30, 3 (2000), 81–113.

Sascha Roth, Matheus Hauder, Matthias Farwick, Ruth Breu, and Florian Matthes. 2013. Enterprise Architecture Documentation: Current Practices and Future Directions. In *Proceedings of the 11th International Conference on*

Wirtschaftsinformatik. Wirtschaftsinformatik, Leipzig, Germany.
George J. Stigler. 1957. Perfect Competition, Historically Contemplated. *Journal of Political Economy*, 65, 1 (1957), 1–17.
The Open Group. 2013a. ArchiMate. Version 2.1 Specification. Zaltbommel: Van Haren Publishing.
The Open Group. 2013b. TOGAF Version 9.1, First edition, Third impression. Zaltbommel: Van Haren Publishing.
Robert Winter and Stephan Aier. 2011. How are Enterprise Architecture Design Principles Used? In *Proceedings of the 15th IEEE International EDOC Conference Workshops*. IEEE, Helsinki, Finland, 314–321.
Robert Winter and Ronny Fischer. 2007. Essential Layers, Artifacts, and Dependencies of Enterprise Architecture. *Journal of Enterprise Architecture*. 3, 2 (2007), 7–18.
Mohammad E. Zadeh, Gary Millar, and Edward Lewis. Mapping the Enterprise Architecture Principles in TOGAF to the Cybernetic Concepts – An Exploratory Study. In *Proceedings of the 45th Hawaii International Conference on System Sciences*. IEEE, Hawaii, USA, 4270–4276.
John A. Zachman. 1987. A Framework for Information Systems Architecture. *IBM Systems Journal*. 26, 3 (1987), 276–292.