An Approach to Building Effective Enterprise Architectures

Susanne M. Glissmann
IBM Almaden Research
smglissm@us.ibm.com

Jorge Sanz
IBM Almaden Research
jorges@us.ibm.com

Abstract

Cost and time are significant practical concerns when an organization addresses the development of an Enterprise Architecture (EA). In order to overcome these challenges, organizations need to focus on those architecture deliverables that are most relevant to the business problem being addressed with EA, yet without losing cohesive integration over a longer life-cycle.

In this paper, we propose a new approach to develop an Enterprise Architecture. Our technique aims at guiding the creation of a simplified EA with well-integrated business and IT architectures. Furthermore, it re-uses related artifacts that may have been created by different stakeholders in the company over time. Our approach can be used to enhance common Enterprise Architecture standards. In fact, we illustrate how the approach may be used in connection to The Open Group Architecture Framework (TOGAF) method for EA. A key aspect of our proposed technique lies on five key relationship types applied to integrate the different architectural dimensions of EA.

1. Introduction

1.1. Motivation

Due to increasing global competition and growing need for agility, companies today rely more than ever on the support of IT solutions that are well-integrated with the business and can be quickly adjusted to changing requirements [1]. However, companies still struggle to achieve this ideal state. A reason for this is that business and IT stakeholders in a company have different views of the enterprise originated in distinct professional domains involved in the operations of an organization, including the IT function. Thus these views vary not only on the interface between business and IT but also across business units. Consequently, business departments do not align well with one another or with the overall direction of the company. A sales department, for instance, might be operated following the enterprise goal to increase customer satisfaction, whereas the service department might have a strong focus exclusively on cost reduction. Due to the potentially conflicting business goals in this example, it would be challenging to define an IT solution that supports equally well the goals of both business departments involved.

To enhance communication, decision making and traceability of significant investments, both business and IT aspects involved in an organization need a solid documentation spanning the architecture of the business at play, application architecture, data and technology architectures [2]. The suitable combination of these architectures form the company’s Enterprise Architecture (EA).

EA standards such as The Open Group Architecture Framework [3], or seminal work such as Zachman’s Framework [4],[5], provide extensive content models, methods, reference models and principles to develop company-specific architectures. In recent years these EA frameworks have been used and advanced significantly, aiming to provide comprehensive and generic approaches to enterprise architecture. However, despite the abundance of information, companies still experience challenges to follow extensive guidelines provided by the standards bodies and to adjust them to their specific needs. Furthermore, little information is provided by existing approaches to EA on how to re-use existing architecture-related artifacts from different business units or involved departments.

EA initiatives are also often seen as too complex to develop and maintain, or even ineffective to fully meet required business agility or expected business-IT alignment [6]. Furthermore, many EA initiatives do not describe sufficiently well the connections among the different architectures and thus, their value is not fully revealed to all the stakeholders concerned with the business problem at hand [9]. According to the Business Architecture Special Interest Group (BASIG) [10] the integration between the different architectures involved in EA is still in its infancy: Architectures are developed in silos with only minimal connections between them, or ill-defined responsibilities of stakeholders in the business and IT departments.
1.2. Objective

Aiming at the development of enterprise architecture, we propose an approach that improves the building process as well as the final architecture. An “effective enterprise architecture” means a well-integrated and agile instrument to support the identification of business priorities and decision-making in enterprises. On the other hand, an efficient process of building an EA means a shortened development cycle that delivers those core artifacts needed while re-using judiciously architecture information that may already exist in a company.

The proposed approach merges incrementally the business and IT artifacts from a high-level to a detailed level view, focusing on those aspects most relevant for the business. The artifact sources addressed in our approach may differ. For instance, an artifact might be based on different architecture standards, or developed without any particular standard in mind. It might be structured or semi-structured. It might be created with a diverse set of software tools, such as Microsoft Visio, Microsoft PowerPoint, Rational System Architect, Troux or Mega. Furthermore, it might represent the different architectures of a company (i.e., business, application, data or technology architectures) completely or only partially.

1.3. Research Approach & Structure

As our research goal is the construction of a new method, we follow the approach of design science as described in [11]. Design science “seeks to create innovations that define the ideas, practices, technical capabilities, and products through which the analysis, design, implementation, management, and use of information system can be effectively and efficiently accomplished” [11]. It is composed of two activities: creating and evaluating an artifact. Artifacts cover constructs, models, and methods. They can be evaluated through observational, analytical, experimental, testing as well as descriptive evaluation methods [11]. In our approach, we propose a method which builds on the common EA standard TOGAF.

Our paper is structured as follows. In the next Section, we discuss the foundations for our research, i.e., enterprise architecture. In Section 3, we introduce the concept of our approach to building effective enterprise architecture. In Section 4, we describe the limitations and the need for further research of the approach. In the last chapter, we provide a conclusion of our proposed approach.

2. Foundations

In the following, a definition of EA is given and different EA frameworks are briefly reviewed. Then the most common EA dimensions, i.e., business architecture, application architecture, data architecture and technology architecture are illustrated.

2.1. Definition of Enterprise Architecture

The concept of Architecture is used in various disciplines (e.g., construction and information technology) to help managing the complexity of the work done in these disciplines by supporting design, change, communication, and realization of the objects of concern. According to ANSI/IEEE Std 1471-2000, architecture is “the fundamental organization of a system, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution [12]. The analogy between architecture and house construction given by Lankhorst [13] is also enlightening: “Suppose you contract an architect to design your house. You discuss how rooms, staircases, windows, bathrooms, balconies, doors, a roof, etc., will be put together. You agree on a master plan, on the basis of which the architect will produce detailed specifications, to be used by the engineers and builders. How is it that you can communicate so efficiently about that master plan? We think it is because you share a common frame of reference: you both know what a ‘room’ is, a ‘balcony’, a ‘staircase’, etc. You know their function and their relation. A ‘room’, for example, serves as a shelter and is connected to another ‘room’ via a ‘door’. You both use, mentally, an architectural model of a house”.

To develop the architecture of an enterprise, different EA frameworks have been proposed and extensively adopted, for instance The Open Group Architecture Framework [3], Department of Defense Architectural Framework [11] and the Zachman framework [4], [5]. The EA frameworks differ in detail, but have in common that they define conceptual models of the enterprise. Additionally, some EA frameworks may also provide a methodology, a reference framework, and architectural principles. In the context of this paper, the conceptual model and methodology, such as the ones provided by TOGAF, are of high relevance.

The Conceptual Model of an enterprise architecture is also referred to as metamodel, content model, or modeling language [13]. Thereby, when a company models its own EA an instance of the content model standard with the specifics of this particular company is created. For instance, in an organization model of an insurance company, the metamodel element “role” could be instantiated by the real-life elements “clerk” and “insurance manager”.
According to TOGAF, the conceptual model can be documented in catalogs, diagrams, or matrices using metamodel elements (see TOGAF Part IV [3]). Catalogs are lists of elements. Diagrams are renderings of these elements in a graphical format allowing stakeholders to retrieve the required information. Matrices illustrate in a cross table the metamodel elements and their relationships.

The Methodology describes the development process of EA models consisting of the activities to be undertaken, the responsibilities to be defined, and the principles to be considered. In general architecture methodologies contain the following phases: First, one to several initial architecture building phases to document the current and the target state of a company (see for instance [7]); Second an opportunity phase to identify projects based on the analysis of the gap between the current and the target states; Third, an implementation phase to provide guidelines on the project realization; And finally a governance and change management phase describing the maintenance of the architecture [8]. The focus of the proposed approach lies on TOGAF’s initial architecture building phases, i.e., business, information systems, and technology architecture.

2.2. Enterprise Architecture Dimensions

Addressing the needs of the different stakeholders in a company, EA is decomposed into several dimensions. Thereby, slight differences between the naming and the number of dimensions exist between different EA approaches [1]. The EA modeling language ArchiMate, for instance, defines three layers, i.e., the business, the application and the technology layer [13]. Similarly, DoD AF is decomposed into an operational view, a systems and services view, as well as a technical standard view [14], [15], [16]. Zachman differentiates between a contextual, conceptual, logical, physical, and detailed representation view [4]. TOGAF is divided into the four dimensions business, application, data and technology architecture. This latter approach represents the most commonly accepted decomposition of EA and is therefore applied in this paper.

The decomposition of EA into different dimensions concerns the metamodel, as well as the methodology. TOGAF, for instance, offers four architectural building phases, i.e., the business architecture phase, the application architecture phase, the data architecture phase, and the technology architecture phase. For each phase TOGAF recommends thereby, which catalogs, diagrams, or matrices of the following four EA dimensions are to be developed.

In the following paragraphs, the four EA metamodel dimensions from TOGAF are briefly reviewed (see [3]).

Business Architecture (BA). This architecture addresses the concerns of users, planners, and business managers, and focuses on the functional aspects of the system from the perspective of the users of the system [15]. To describe different aspects of the business architecture, a number of BA diagrams are available. According to BASISG the following aspects of the company addressed in general in BA models [10]: motivation, capabilities, organization structure, value chain, metrics, product and services, business processes, business rules, and business decisions. Elements of TOGAF’s BA content model are driver, goal, objective, measure, organization unit, actor, role, location, function business service, contract, service quality, process, event, control and product.

Application Architecture (AA). This architecture is used by system and software engineers responsible for developing and integrating the various application software components of the system [15]. It defines the application landscape of the enterprise. TOGAF identifies a set of diagrams which can be used to describe the application architecture, i.e., application portfolio catalog, interface catalog, system/organization matrix, role/system matrix, system/function matrix, application interaction matrix, application communication diagram, application and user location diagram, as well as the system use case diagram. Elements of the application architecture are the information system service, logical application component and the physical application component.

Data Architecture (DA). This architecture addresses the concerns of database designers and database administrators, as well as system engineers responsible for developing and integrating the various database components of the system [15]. Its goal is to define the data entities relevant to the enterprise. According to TOGAF, in this architecture dimensions the following artifacts are applied: data entity/data component catalog, data entity/business function, system/data matrix, class diagram, and the data dissemination diagram. These artifacts reference TOGAF metamodel elements data entity, logical data component, and physical data component.

Technology Architecture (TA). This architecture defines the physical realization of an architectural solution [15]. It has strong links to implementation and migration planning and is therefore used by software acquirers, operations staff, systems administrators, as well as systems managers. Technology architecture include for instance technology standards, technology portfolio catalog, environments and location diagram, platform decomposition diagram, network computing/hardware view, system/technology matrix, communications engineering view, processing and cost view. According to TOGAF, three metamodel elements are specific for technology architecture, i.e., the platform service, logical technology component, and the physical technology component.
The four EA metamodel dimensions are highly connected through complex relationships between the individual metamodel elements. For example, a data entity (DA) is used by a logical application component (AA), which is used by an actor in a business process to meet a defined set of business objectives (BA). The application component is supported by a platform service (TA). Consequently, in order to maintain a well-integrated enterprise architecture, changes in one metamodel dimension often require modifications in the remaining dimensions.

3. Conceptual Description of the Approach

In the following, the underlying assumptions for the proposed EA development approach are illustrated. Then, we also introduce the different business-IT relationships that form the basis for our approach. We conclude the Section with the incremental phases and general guidelines on how inconsistent parts of the enterprise architectural can be integrated incrementally from a high-level to a detailed-level view. For simplicity, we address the combination of application, data, and technology architecture as IT architecture. It is separated from the business architecture, as the BA has a special function in the EA to guide the development of the remaining three architectures. The proposed approach is research in progress and thus focuses currently strongly on the EA standard TOGAF. As it is explained in Section 4, while the approach hinges on TOGAF, we also plan to adapt our work to other EA standards in the near future.

3.1. Assumptions

Our approach is based on the following assumptions regarding the completeness, consistency, quality, and integration of business and IT architectures in real-life situations.

1. Business Architecture. Companies’ BAs are often inconsistent, incomplete, and not fully integrated. Often the BA models which were created over time by different business units are obsolete and therefore not compliant with the real business situation. The models are based on different architecture standards or no standards at all. Moreover, they are documented in different formats, such as MS Powerpoint, Visio, or professional EA tools.

2. IT Architecture. Similar to companies’ BA models companies’ IT models are often inconsistent, outdated, and only partially connected. Many IT models however might be more structured than BA models as they are developed using the Unified Modeling Language (UML) [17]. For the development of IT models the same tools as for BA development can be used.

3. Business-IT alignment. As a consequence of the inconsistencies of BA and IT models, the relationships between them are often also incomplete and incorrect. Thus, it can be challenging for companies to create a holistic EA view which addresses equally the concerns of the different stakeholders. The poor integration between the business and IT architecture furthermore makes it difficult to understand the impact changes of one architectural dimensions have on the remaining the architectural dimensions.

As a consequence of the made assumptions, EAs often do not fully reach their original purpose to enhance communication between the different stakeholders, and to facilitate the analysis and decision-making.

3.2. Business-IT relationship types

Our approach to building effective enterprise architecture is built on five relationship types which occur between the elements of the business architecture and the elements of the IT architecture (for simplicity in this paper we combine data, application and technology architecture to IT architecture). Figure 1 provides an overview of these relationship types. One to several relationship types can apply for the same elements. In the following, each relationship is introduced and a brief action item addressing the challenges of the particular relationship type is given.

![Figure 1. Business-IT relationship types](image)

3.2.1. Equivalent elements

In general, EA Frameworks assign elements to specific architectural dimensions (compare to TOGAF content model [3]), which address the concerns of different stakeholders. Some elements, despite their origin in a particular dimension, are shared by several architectural dimensions in the
form of catalogs, diagrams and matrices. In these cases, the shared business and IT elements might have the same name and type. For instance, a business process modeled in the BA might reference information systems services to demonstrate how IT resources support the business process. The references to these IT elements in the business architecture are important as they have a high impact on the fulfillment of business objectives, e.g., productivity or quality, as well as on the total process costs.

However, while the elements are shared between the architectural dimensions, the attributes of a mutual element might differ between them. For example, for a business analyst evaluating an application used in a business process, it is important to know its costs, response time and availability. Against this, the IT architect choosing the hardware for this application is interested in the CPU power required by the application. The software designer needs to know the text fields on the application’s user interface (UI). These UI details of the application might be however less relevant for a business executive who only requires a high level understanding.

**Action item:** There are different needs that govern the creation, viewing and editing of the attributes of architecture elements shared across stakeholders. Accomplishing these tasks requires well-defined responsibilities on the attribute level of an element.

### 3.2.2. Disguised shared elements

Whereas the previous relationship type addressed elements of the same type and name, a shared element can have a different name, and / or be of a different type despite the fact that it might describe the same object. For instance, in a business architecture a business process might be documented with an Event-driven Process chain (EPC) diagram [18], whereas in the IT architecture the same business process is simulated with the Business Process Modeling Notation (BPMN) [19]. Furthermore, the name of the same business process might differ. For instance, the business process “customer process” defined in the business architecture, could be called “sales and service process” in the IT architecture.

As a consequence of the inconsistent naming and usage of different element types, it can be difficult to identify elements which describe the same object in different architectural dimensions. Thus, elements might appear several times in different architectures hindering the tracking of changes among the architecture dimensions.

**Action item:** Identify all diagrams and elements of a similar type. For instance, business processes can be modeled in BPMN or EPC diagrams. Map the elements with the same name and the same meaning. Analyze the remaining elements and map them if they describe the same object despite different names. Identify a primary name for the elements to be mapped and store the elements in a data dictionary.

### 3.2.3. False siblings

The mapping of elements with same or similar names or types needs to be undertaken with caution as the meaning of the elements might not always be the same. For instance, an element of type component can refer to an application component in a UML component diagram describing the details of an application [17], or it can refer to a business component in a component business model (CBM), describing a modularized approach to business operations [20].

**Action item:** Evaluate carefully if elements with the same name also have the same meaning. Identify a unique element name for each element type. For instance, name the element type “component” in the UML component diagram “application component”, and the element type “component” in the CBM model “business component”. Then, identify the dependencies between the elements and other elements in the business, as well as in the IT architecture (see 3.2.5).

### 3.2.4. Element simplification across dimensions

Business architecture and IT architecture address the needs of different stakeholders. When re-using information from one architectural dimension in another architectural dimension, it is advisable that only a subset of the elements and selected connections to elements of other architecture dimensions are used. This will avoid overloading stakeholders with information which is of no relevance for them. For example, a data entity defined in the IT architecture can be re-used in the business architecture as it helps determine the data access rights of the different roles in an organization. However, the specific details of each data entity (e.g., table structures) are less relevant to the stakeholders involved with the business architecture.

Furthermore, to further simplify the views for different stakeholders not all elements of the same type and level might be relevant to have in another architectural dimension. For instance, applications in the IT architecture can be divided into different categories, e.g., backend applications, business applications, or database applications (see next page, Figure 2). This categorization can then be used identify for the BA the elements which are relevant for business decision-making. For instance, a business process would only be connected to business applications as these are user-facing applications. Thus, the other application categories can be neglected in the business architecture.
While the connections between business users and applications can be modeled in business architectures as well as in IT architectures, other connections, such as the back-end connections between the application are often only relevant for the IT decision-makers and thus should be defined in the IT architecture. However, changes in the business might require changes in the IT architecture. For instance, the sequence in which the user plans to interact with the applications will require changes regarding the back-end connections of the applications.

Action item: Determine the minimum necessary elements from each of the architectures to be shared or integrated with those from other architectures. Determine the level of detail which should be made visible and available.

3.2.5. Dependencies among elements

As described above, architectures can be connected through shared elements. In addition to this, they also can be connected vertically or horizontally between two different elements of the business and the IT architecture.

Vertical Dependencies. Business and IT architectures can be connected through an element hierarchy (compare to 17). For instance, business metrics, such as “overall customer satisfaction” can be broken down into the measure “customer satisfaction”, which is then assigned to a company’s customer facing processes. Further decomposed into quality and efficiency metrics of a process, the metrics can then be passed on to the IT architectures. For each application supporting the business process IT metrics, such as response time and availability can be determined and connected to the metrics of the business process.

The vertical dependencies can exist within and between the four architectural dimensions.

Horizontal Dependencies. Elements can also be integrated horizontally within and between architectural dimensions. For instance, business components can be supported by business applications defined in an IT architecture. Roles in an organizational structure use application and hardware in their business processes. In the IT architecture, these roles are often combined to user groups. For instance, the managers of a sales department could be combined to a user with advanced data access rights and additional report functionality, which are however not provided to the more operational roles with less responsibility. Furthermore, user and activities can be connected to certain data entities defined in the IT architecture.

Action item: Define the dependencies between the elements and identify the consequences the changes on one element have on the connected elements. For instance, increasing the customer satisfaction metric in the business architecture may require adjustments of the response time and availability of an application supporting a customer-facing process.

3.3. Incremental BA & IT architecture phases

The incremental BA & IT architecture phases of the proposed approach to building effective enterprise architectures aims at extending methods of common EA frameworks, providing guidelines on how to re-use and map EA artifacts that already exist in a company. The approach uses thereby the relationship types identified in the previous Chapter 3.2.

Figure 3 illustrates the four key incremental phases of the approach. The phases are based on the
architecture building phases in TOGAF’s architectural development method, i.e., the business architecture phase (B), the information systems architecture phase (C, consisting of the application and data architecture phase), as well as the technology architecture phase (D). Furthermore, the phases reference the content model elements as defined in the TOGAF Part IV.

TOGAF provides a very detailed content model with a tight integration between the architectural dimensions. However, the ADM method is rather generic [TOGAF 2010] and describes only on a high level how to integrate the different architectural dimensions. Furthermore, the ADM explains majorly how an EA can be developed from scratch. Very little guidance is given on how to re-use existing artifacts. The ADM does allow going back-and-forward between the different phases but does not explain in detail how this should be done systematically. Instead, the ADM describes more how one phase has to be completed first before the next phase can be initiated.

3.3.1. Phase 1 - Business Blueprinting

EA initiatives often fail to meet the expectations of business decision-makers. Thus, phase 1 creates a blueprint of the business by describing aspects of key concern for the business managers and executives. It extends the TOGAF phase B – Business Architecture.

For this, business architects develop a high level capability map (also called CBM map [20], see Figure 4), which categorizes the company into a number of key capabilities. Each capability fulfills a purpose, requires activities to be performed, and uses resources. Furthermore, each capability consumes and offers business services [IBM 2008]. This one page view of the company can be easily discussed with business managers and executives. To simplify the creation of the capability map industry templates can be used and adjusted to meet the specifics of a company.
PowerPoint, Visio or Word documents. Similarly, other elements of the business are collected from the business managers and connected to the capabilities map. This includes for instance, high-level business elements, such as the elements driver, objective, measure, organization unit, actor, role, location, function, business service, contract, service quality, process, event, control and product. All elements are entered in a business catalog.

As the elements are re-used from different sources, the business catalog is at this stage most likely incomplete, inconsistent and not compliant with the real-life. Thus, it is important to consolidate and update the high-level elements defined for the business architecture. The business elements are analyzed according to their relationship type (i.e., equivalent, shared, disguised, or dependent) and modified accordingly (see Chapter 3.2.). Furthermore, for each element clear stakeholders from the different business units are determined.

3.3.2 Phase 2 - IT Blueprinting

To create a common high-level understanding of the IT landscape, in this phase an IT blueprint is developed. This phase of our approach to building effective enterprise architecture extends TOGAF phase C – Information Systems Architecture (consisting of the application and data architecture), and TOGAF phase D – Technology Architecture.

The phase builds on the business capability map developed in the previous phase. For this, first a high-level IT catalog is developed by collecting IT elements from existing sources, such as EA tools, Visio, PowerPoint, or word documents. As the IT landscape is often described in UML diagrams, the IT architecture is in general documented in a more structured way than the business architecture. However, the existing IT diagrams are seldom well connected and up-to-date with the latest code changes or configurations. Thus, to consolidate, the elements in the IT catalog, the relationships between the IT elements are identified and modified as described in Chapter 3.2.. In a next step, the high-level elements are categorized into elements, which are relevant or irrelevant for the business. For instance, elements of type applications are categorized into business applications, back-end, and database applications (compare to Chapter 3.3.4). The elements relevant for the business are then connected to the business capabilities, which they support.

3.3.3 Phase 3 - Detailed Business Architecture

In the previous two phases a high-level view of the business and IT architecture was created and the two architectural dimensions were initially integrated through the capability map. In this phase the parts of the business architecture, which is of most relevancy for the business are developed in more detail. To reduce the EA development effort the other parts of the BA will remain on a high-level description.

To select the parts of the business architecture which require further development, first the enterprise’s key business capabilities, i.e., the capabilities with the highest impact on the enterprise strategies are identified. The capabilities are then further analyzed regarding how well their performance support the strategies. The capabilities with the biggest gap between the current and the target performance are selected to be specified in more detail in the BA.

First, the elements, which were assigned to one capability, are decomposed into more detailed element levels [21]. For instance, the organization unit “sales“ is decomposed into roles and actors. If possible, the more detailed level elements should be connected horizontally to each other. For instance the measure, “friendliness” could be decomposed from the higher level measure element “customer satisfaction” and could be connected to the role element “call center agent”. For each element, all attributes relevant for the different business stakeholders should be defined.

Finally the detailed level elements should be integrated with the IT elements which are connected with same business capability (see Chapter 3.2.)

3.3.3 Phase 4 – Detailed IT Architecture

In this phase, the parts of the IT architecture which are most relevant for the business, i.e., the IT elements which are connected to the key business capabilities and business elements, are further specified.

Similar to the detailed BA architecture phase, first the key IT elements are further decomposed into sub elements and then connected to other detailed IT elements, as well detailed business elements, which are part of the same key capability (see Chapter 3.2.). From an perspective of IT stakeholders other relationships than identified by business stakeholders might be relevant and should therefore be added to the EA content model. Furthermore, IT stakeholders should define the attributes of the IT elements which are relevant for their decision-making on application and infrastructure support. Finally, IT architects need to analyze if from an IT perspective the attributes of the business elements which are shared between or connected with the IT architecture are complete or need to be further extended.

4. Limitations and Further Research

Our approach to building effective enterprise architecture addresses common challenges companies face today when developing a holistic enterprise
architecture. At the current stage, the proposed approach has been used in combination with the EA standard TOGAF. It is planned to further extend the approach for other enterprise architecture standards (e.g., DoDAF, FEA, MoDAF, and Zachman) by taking into consideration the specifics of these standards’ metamodels (i.e., different architectural dimensions, elements and the relationships between the elements), and their methodologies, (i.e., phases and guidelines).

Companies adopt EA standard frameworks to develop their own company-specific enterprise. In order to align the standards to the needs of their own company as well as already existing architecture artifacts, companies often change the EA standard metamodel. For instance, they combine several standard elements to one company-specific element. For simplicity, a company could replace in their own EA the TOGAF elements “logical application component” and ‘physical application component’ with the more abstract element ‘application’. Our approach to building effective enterprise architecture has therefore yet to be validated against real-life situations where EA standards have been adjusted to the company specifics.

EA initiatives are complex as numerous stakeholders with different interests are involved in the EA development process. Changes in the enterprise architecture by one stakeholder often require modifications by other stakeholders. To better manage, the responsibilities of the different stakeholders developing the business and IT architectures, we therefore plan to establish reference responsibilities describing the read, write, and edit responsibilities on the level of elements, as well as element attributes. These responsibilities shall also provide requirements on how changes in one architecture will affect the remaining architectures.

• **BA Element sets fixed constraints on IT architecture.** For instance, business decision-makers identify data which is highly sensitive to the business. The IT architects have to consider this constraint in the development of the IT architecture and therefore cannot outsource applications that access this sensitive data to a public cloud environment.

• **IT Element sets fixed constraints on Business Architecture.** Following the IT strategy to operate only applications of a certain SOFTWARE vendor, an IT decision-maker may refuse the desire of a business manager to choose an application offered by a different software vendor.

• **BA Element provides guidance for IT Architecture.** Business decision-maker identifies a set of hot components, which need to be urgently improved regarding their IT support. IT decision-makers determine the IT investment costs for the given components. As the costs exceed the available IT funding, IT decision-makers prioritize the IT investments focusing only on a sub-set of the components, which were before identified by the business.

• **IT Element provides guidance for Business Architecture.** IT decision-makers inform the business about the availability of a given application. Business decision-makers request the IT architects to improve the service level availability of the application and offer to increase the funding for IT.

In addition to the extensions on the conceptual level, we also plan to improve the tool support of the proposed approach. A sufficient tool support of the approach is necessary, as the approach in combination with EA standards is due to its complexity difficult to manage manually. Instead, existing EA software, such as Mega, System Architect, or Metastorm should be utilized. To illustrate the benefits of the proposed approach when supported by an EA tool, we plan to extend the functionality of the EA tool Rational System Architect.

5. Conclusions

The paper introduced our approach to building effective enterprise architecture. Effectiveness is understood in terms of the business sufficiency and fitness of the technique. The approach can be used in combination with common EA standards. It aims at extending them with guidelines on how to re-use existing architecture artifacts and how to integrate incrementally the different architectural dimensions of an enterprise architecture. The approach defines a set of relationship types between architectures that need to be considered when connecting business and IT elements of different types and names created by numerous stakeholders over time. The paper describes how the approach can be combined with the TOGAF’s EA method ADM. Further conceptual and EA-related tooling enhancements are planned in the near future, including the provisioning of a use-case that will assist practitioners in applying this method successfully. Further comparisons of the present technique to existing conventional methods in terms of efficiency of the overall EA process are being pursued.

6. Acknowledgments

We want to thank the reviewers for some excellent comments that helped us to improve the paper.
7. References


[6] Hoppermann, J., Enterprise Architectures are not proving their value, Forrester, 2009


