Business Building Blocks as Coordination Mechanism for Enterprise Transformations

Claudia Hess, Florian Lautenbacher and Katrin Fehlner
Softplant GmbH
Munich, Germany
{claudia.hess, florian.lautenbacher, katrin.fehlner}@softplant.de

***Abstract*—Enterprise architecture management (EAM) aims at aligning business and IT. Therefore, EAM analyzes the current and defines the target state. In order to reach the target state, EAM plans and monitors transformations affecting business processes, information systems and the underlying IT infrastructure. Thereby, domains are a widely used tool in order to reduce the complexity. The domains structure the enterprise architecture, e.g. according to the business fields a company is working in. Resources and responsibilities are assigned to these domains and transformations are then coordinated within domains. This approach, however, poses the risk that business requirements are solved redundantly although an existing IT system from another domain could have been reused. It even hinders an integration of the IT systems used in the different business fields.**

This paper describes an approach for coordinating transformations based on so-called business building blocks (BBBs). BBBs define the essential business elements independent from their implementation, with a granularity suitable for reuse within and across domains. Our approach determines who is responsible for providing a solution to a certain business requirement by assigning clear responsibilities to the corresponding BBB. This business outcome-driven approach helps focusing on business requirement solutions instead of managing system solutions. Ideally, there is exactly one responsible authority per BBB. In the paper, we provide a description of our BBB identification method and relate the suggested approach into one of the currently most widely adopted EA frameworks, i.e. The Open Group Architecture Framework (TOGAF). We report about the experiences we made when using this method in various EAM projects and recommend best practices.

***Index Terms*—enterprise architecture management, enterprise transformation, business architecture, building blocks**

I. Introduction

Enterprise architecture management (EAM) aims at further developing the IT landscape so that it efficiently supports current and future business needs. EA models serve as information basis for decisions on modifications and enhancements of the IT landscape. They provide a holistic view on an enterprise by aggregating information about the business strategy, business processes, the information systems with their components and interfaces, the data exchanged as well as the underlying infrastructure. Relating these elements provides enterprises with a clear picture of the current business support by IT systems. Driven by the business strategy, a desired target state is formulated from the strategic, business, organizational and technological perspective. Enterprise architecture frameworks provide guidance for analyzing the as-is, defining the target state and managing the purposeful transition from the current to the target architecture.

Required enterprise transformations range from changes in business processes to changes of the information systems and the underlying IT infrastructure. In order to close the gaps between the current and the target state, transformations are planned and initialized by the EAM team. However, besides of these top-down started projects, there are so-called evolutionary (bottom-up) changes, of which the EAM approach has to be aware, too (cf. [1]). Managing enterprise transformations requires thus to coordinate at the same time a huge number of potentially interdependent projects regarding organizational, process- and IT-related aspects. Moreover, these projects are initiated and executed by different people. It is a challenge to align transformations in a way that redundancy or even contradictory results are avoided. An efficient coordination mechanism is thus required which clearly defines scopes and responsibilities for transformations.

In EAM, domains are a widely used tool when planning and coordinating enterprise transformations [10]. The domains structure the enterprise architecture, e.g. according to the business fields a company is working in. Typical examples for such domains are customer service, warehouse management, research & development, financial services, and so forth. Domains represent a stable frame of reference. Processes and IT systems can be grouped into the respective domain. Resources are assigned to the domains and a person or a board can be appointed as responsible for a domain. Transformations are then coordinated within each of these domains. This reduces the complexity considerably as the number of interdependent transformations is significantly smaller. Often, domains are even further detailed in several hierarchies.

However, such domains are static and cannot be easily modified as this would lead to large modifications in resource allocation and of responsibilities and decision making power. It is very important to find the right domains and the appropriate level of granularity in order to avoid developing redundant functionality (cf. [10]). Domains that are structured too fine-granular may hinder an integration of the IT systems used in the different business fields as well as a harmonization of processes between different business units. The optimization is done only within the particular domain. In contrast, a domain that is too general does not reduce the complexity of coordinating business transformations.

Instead of using a static domain model to structure the IT landscape, we suggest using so-called business building blocks (BBBs) as central structuring element and form the required domains dynamically. We claim that this approach supports the coordination of projects, not only within but also across business fields. The contributions of this article are:

- We evaluated EA frameworks with respect to the typical elements of the business architecture which could be used as frame of reference for the coordination of enterprise transformations. Besides of domains, we analyzed especially the usage of business objects in related approaches.
- We developed a methodology how to identify and define business building blocks, the BBBs. The methodology includes how to assign responsibilities to those BBBs and how to ensure the governance afterwards. Moreover, we show how clustering BBBs can be used to build dynamic domains and to benefit from the advantages that domains offer. We relate the suggested approach into one of the currently most widely adopted EA framework (cf. [6]), i.e. The Open Group Architecture Framework (TOGAF) [20]. We therefore implement BBBs as extension to the TOGAF Content Metamodel.
- As this methodology has been applied in several projects in the last years, we report about our experiences and recommend best practices. We show current limitations and directions for future research.

To this end, the paper is structured as follows. Section II gives an overview on related approaches. Section III then introduces business building blocks, the BBBs, and the methodology for creating and using them. The implementation of the BBBs as extension to the TOGAF Content Metamodel is presented in section IV. In section V, we discuss our experiences gained when using BBBs in EAM endeavors as basic coordination mechanism and recommend best practices. Finally, section VI summarizes the approach and shows topics for future research.

II. Related Work

This section surveys similar approaches mostly concerning business objects. It also provides the context information about domain approaches and to which part of EAM architectures BBBs belong.

A. EA Frameworks

EA frameworks provide a good starting point for the introduction and the further development of an EAM approach in an organization. They support developing and organizing EA models and introducing EA concepts into an enterprise [5]. They also provide guidance for efficiently running EAM endeavors. Various standards setting organizations, companies, consultancies, governmental organizations and military agencies published their best practices as EA frameworks. So there are currently more than 50 known frameworks specified by different organizations [14]. Depending on their scope, EA frameworks provide a methodology, a process for developing the architecture and / or suggest a model of architectural artifacts and their relationships. One of the most widely adopted Enterprise Architecture framework in industry today is the TOGAF framework [20]. According to a recent survey [6] 82.2% utilize TOGAF to build their enterprise-specific EA approach on. Other well-known frameworks are, the Zachman Framework [22], DoDAF [9], Quasar 3.0 [11] and the ARIS framework [17].

B. EA Layers

EA frameworks typically comprise several layers. Each layer takes a different point of view. Frameworks slightly differ in the composition and the naming of these layers. TOGAF, for example, distinguishes the following four architecture levels:

- The business architecture describes the aspects of business. It is defined in order to optimize the organization based on business visions and strategies. Artifacts of this layer are thus, for example, actors, processes and business services. Central questions consider the interaction between processes and business services or how the organization is structured.
- The data architecture (part of the information systems architecture) is mainly concerned with information assets that are considered as enterprise information and which should be shared efficiently across the enterprise. It therefore structures the organization's data entities as well as associated data management resources.
- The application architecture (part of the information systems architecture) describes individual application systems, how they are designed and how they interact with each other. Artifacts of the application architecture are thus the application components. In the application architecture those systems that are crucial for the success of an enterprise and which support core competencies are typically described and monitored in more detail than those systems that do not offer competitive advantages.
- The technical architecture describes the hardware, software and network infrastructure that supports applications and their interactions. This layer is introduced in EAM approaches in order to limit, or even to reduce, the number of different infrastructure and technology components that are actually in use and thereby to reduce costs. Artifacts of this layer are therefore technology components and platform services.

C. Domain models

In the following, we will focus on the business architecture as we want to obtain a coordination mechanism on the business level. Grouping an enterprise architecture via domains is typical for the business architecture. Mostly this is done in a top-down approach: the company is divided into domains and those are again splitted into sub-domains. The domains build the basis for classifying and structuring the information systems in the next step. Dern mentions that architecture domains build a focus in the future communication and therefore must not be ambiguous, too complex or with questionable boundaries [8]. In order to find the corresponding domains,

one can either choose a completely manual process or a semi-automatic one. Existing approaches such as Quasar [10], Dern or Keller [12] identify domain candidates in a first step and try to identify the including business components and business services. By identifying business components and services the domains are refined in an iterative way: either they are aggregated again because the identified business services in two domains are closely related or one business domain is divided in several ones. A semi-automatic approach for finding domains would be to use graph clustering approaches [3] on activities and IT systems where dependencies between those are evaluated in order to find appropriate clusters which are then proposed as domain candidates.

Structuring of an architecture via domains poses the advantages of better communication with different stakeholders, e.g. the management or business units, through a repeatable structure that all persons get to know and can easily remember. As soon as the domain model is accepted by the stakeholders, it is easy to show the connections between different domains to identify necessary interfaces.

D. Elements in the business architecture

In the following we will introduce the typical elements that can be found in the business architecture level. As mentioned before, TOGAF's core metamodel [20] includes elements for describing the organization (using organization units, actors and roles) and for functions that deliver business capabilities closely aligned to the organization (processes, business services and functions).

TOGAF does not include an artifact for the elements that are exchanged in and between business processes. One might argue that the exchanged building blocks can be found in the data architecture (the data entities or logical data components that are part of the data modeling extension).

However, TOGAF defines data entities as "An encapsulation of data that is recognized by a business domain expert as a discrete concept. Data entities can be tied to applications, repositories, and services and may be structured according to implementation considerations". Hence, the focus is on the IT-implementation in applications and on the physical data that is stored in repositories and exchanged via (web) services. TOGAF proposes e.g. to use entity relationship models to address database designers and application developers.

Additionally, the concept of logical and physical data components (introduced in the data modeling extension) "allow more sophisticated modeling and the encapsulation of data" [20]. That means, that they are used for grouping data entities which are relevant for a specific area such as governance, security or deployment boundaries around data. For example, some data entities are confidential and may only be accessed by authorized personal, while others are public and can be exchanged without additional security mechanisms.

Quasar 3.0 [11] includes similar elements in the information architecture to determine the information flow. In the business architecture it refers to the elements business goals, business activities, and roles and resources. Output of business architecture activities are business dimensions, services and objects. The outputs of business processes is here as well as in other approaches known as business objects. Already [2] (p.298) described the importance of focusing on business objects for EAM.

E. Views on Business objects

In the following we will detail the different understanding of business objects in several approaches. Quasar 3.0 define a series of patterns and address the building objects that are exchanged in processes in pattern 117: "for each business architecture the major business objects need to be identified as the essential entities a business service is transforming". According to their understanding business objects are the real thing and serve as input or output of business services. Business objects may be identical or infer information objects. However, no further details about how to identify business objects, their definition or how to utilize them in information objects are given in Quasar 3.0.

In the predecessor Quasar Enterprise [10] more information about business objects and their relationships to other elements are given: Basic business services use and consume (i.e. create, transform and eliminate) business objects whereas basic business services are the services on the most detailed level. However, it is not defined when this level is achieved ("Decompose top level business functions recursively using a top-down approach until the *appropriate* level of detail is achieved").

ArchiMate [21] also knows the concept of business objects. They are defined as "a passive element" (i.e.they do not trigger or perform processes) that "has relevance from a business perspective". They are manipulated by behavior such as business processes or functions. The passive entities represent the important *informational* or *conceptual* concepts in which the business thinks about a domain. According to ArchiMate business objects can be accessed by a business process, function, a business interaction, a business event, or a business service. Business objects have relationships with other business objects, such as association, specialization, aggregation or composition. ArchiMate suggests that "the name of a business object should preferably be a noun". [4] states that there is no clear mapping from data entities in TOGAF to elements such as data objects or business objects in ArchiMate.

The Architecture of Integrated Information Systems (ARIS) [18] provides an object-oriented view and groups information objects and implementation classes to business objects in order to support workflow orchestration. The granularity of business objects should consider the principles of high coherence and low coupling. Typically, business objects are build bottom-up by combining IT systems and functions in a business object that includes the business process, all included data and applicable functions. In this approach UML packages are considered good candidates for defining business objects.

The approach of Managed Evolution [15] additionally knows several layers of business objects. The different layers were introduced in order to keep the business object model manageable. They see business objects models as "highly industry-specific models of the business concepts, their properties and their relationships" that "capture the essence of a business in a formal way and forms the basis for consistent IT implementation". They distinguish between enterprise business object models, domain business object models and component business object models. Enterprise business objects are relevant for the whole enterprise, domain business objects are gained by a top-down transformation of the enterprise business objects to domain level and further refined to individual components which are then used as a basis for implementation. Each domain specifies its own key business concepts, attributes, associations and operations in their domain business object model. However, no further description can be found how the enterprise business objects or domain business objects are identified and agreed upon between the stakeholders.

Several other approaches also consider business objects. Although most of them focus on the IT-implementation of business objects (e.g. [13], [7]), they provide insights and tips for modeling business objects. Nicola et al. [16] addresses the aspect that the business object model should be kept verbose and simple so that stakeholder understand it easily. Therefore, the model should be limited to those objects, services, and attributes that are necessary to understand the business rules. The business object model "is a graphical and textual document that businesspeople can verify because it presents familiar objects within their business". Schönthaler et al. [19] describe the interaction between business objects and processes: "Business objects are produced, read, modified, and consumed by activities of the business process (also known as the *CRUD cycle* for Create, Read, Update, Destroy operations)". They can represent documents (contracts, business forms, etc.), database objects, text messages (SMS, emails, etc.) or tangible goods like products or raw materials.

In the following (cf. table I) we compare the introduced approaches that describe business objects. Therefore, we selected the following criteria:

- Material: Does the definition of business object focus on physical objects (e.g. printed documents) or does it also include immaterial objects (such as classes in object-oriented programming)?
- Business-/Service Orientation: Is there a relation to business processes and services?
- Identification process: Does the approach explain how the business objects are identified and defined together with all relevant stakeholders?
- Distinction in business and technical object: Does the approach also include a technical data object in addition to the business object or does the business object include also the IT-technical implementation?
- Distinction between data and behavior: Does the business object include both static and dynamic aspects or are they rather passive elements?
- Perspective: Does the approach focus on the business level or rather the IT-level? This value is our conclusion from the former criteria. In some approaches it is difficult to tell as the focus of the whole approach can be different to the focus of the business objects.

As can be seen in Table I only ARIS and Nicola et al. provide an approach to identify business objects. Alas, both approaches for identifying business objects are IT-centric for business objects as they start with the implementation classes and work in a bottom-up manner (e.g. considering UML packages). Hence, no approach defines a methodology to define business objects focusing on a business perspective or use these business objects for coordination purposes.

III. Business Building Blocks

As we have seen in section II there is to the best of our knowledge no approach for business objects that focuses on the business perspective and includes a detailed identification and coordination mechanism. In the following we introduce such an approach.

A. Definition

A business building block (BBB) is an object that is in tight relationship with the business activities of an enterprise and which is relevant to the value chain. BBBs denote in what an organization is engaged. That is, BBBs represent the essential business elements on which an organization spends much effort, and for which process definitions and IT solutions must be available. However, they are independent of IT-system implementations. BBBs indeed clearly focus on the business perspective and are, in consequence, located in the business architecture. They are described in the terminology of the business units, and not in the terminology of the IT. Thereby, it is ensured that the business understands the topic and hence a shared understanding among business and IT is created.

BBBs can have relationships to each other. They are thus integrated into a BBB model. This model makes dependencies and correlations between BBBs transparent. Moreover, these references indicate where IT components implementing functionality around the BBBs have to interact with one another.

We have chosen the term business building block instead of business object as there is no unique agreed definition for the term business object (see section II). In accordance with TOGAF which defines a building block as "a (potentially reusable) component of business, IT, or architectural capability that can be combined with other building blocks to deliver architectures and solutions" ([20], p.27), we selected the term business building block to avoid confusion.

An important characteristic of a building block is that domain experts acknowledge it as a significant 'thing' and that its scope is well-defined. BBBs comply with this characteristic because they denote a substantial conceptual element closely related to the business activities of an organization. The understanding of a BBB is defined precisely with the respective

Approach	Material	Relation to services	Identification process	Business vs. technical	Data vs. behavior	Perspective
Quasar	both	yes	no	yes	distinction	business
ArchiMate	immaterial	yes	no	yes	distinction	business
ARIS	Immaterial	no	yes	no	both	IT-centric
Managed Evolution	n.a.	n.a.	no	yes	n.a.	business
Nicola et al. [16]	immaterial	no	yes	no	both	IT-centric
Schönthaler et al. [19]	both	yes	no	yes	distinction	business

TABLE I
SUMMARY OF OUR EVALUATION OF APPROACHES CONSIDERING BUSINESS OBJECTS

experts from the business units. Moreover, TOGAF emphasizes that building blocks may interact with other building blocks. BBB models exactly show these references.

Last but not least, TOGAF states that a building block should be loosely coupled to its implementation. This is the case for the BBBs. BBBs are independent from any implementation. They define the key concepts of business without specifying data structures. The BBBs neither define how nor when they are created and used within business processes. They are thus independent from concrete process definitions. Processes can thus be changed without affecting the definition and the scope of the BBBs. Moreover, BBBs are independent from IT implementations. The functionality provided around a BBB is defined by the corresponding business service(s) which are still independent from IT implementations.

B. Approach to Creating BBB models

It is important to establish a method for identifying and defining BBBs to assure a common level of detail. For instance, one may consider each information asset that is produced in a business process as an individual business building block, such as *orders received* and *orders confirmed* and *orders executed*. Someone else, however, may search for commonalities between different process outputs and defines business building blocks with different status, such as a business building block *order* with the status received, confirmed, and executed. Integrating them into a BBB model, the different level of detail may lead to confusion and complicates analysis of the BBB model. A precise definition and common understanding of how BBBs are identified is thus required.

In order to build up a BBB model we suggest a two-step approach (see Figure 1). First, BBB candidates are identified. Second, the BBB candidates are integrated into the BBB model. We recommend to apply the approach in several iterations in order to build up the BBB model of a whole enterprise (or a part of it). Each iteration of the method should have a well-defined scope, for example, a certain process, a project proposal, or an IT system. BBBs are then identified within this well-defined topic, for example the BBBs created and used within the process at issue. In successive iterations, the BBB model is then extended.

Typically, an enterprise architect takes the responsibility for creating a BBB model according to the suggested methodology. In the first step, the enterprise architect calls in the experts from the business units to discuss the candidates. In the second step, he/she often needs to moderate discussions between the different stakeholders.

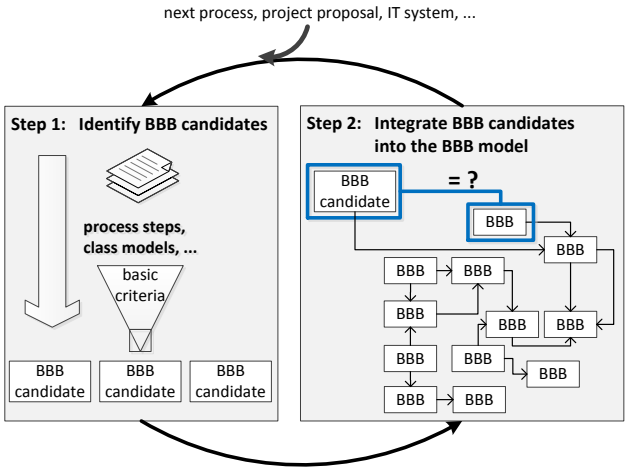

Fig. 1. Business Building Block creation process

1) Identify BBB candidates: As basis for the first step, the enterprise architect starts with collecting relevant existing information sources such as process definitions or class diagrams. The information sources are then analyzed and candidates for BBBs are identified on the basis of the inputs and outputs of process steps, class models, defined information types, or other organization-unit specific elements.

Each BBB candidate has to be examined critically:

- Does the BBB contribute to the purpose of the BBB model? Is it necessary to represent this information in the BBB model or is it, e.g., too detailed or specific of a particular project?
- Does the candidate meet the required basic criteria for being considered as business building block? The following criteria are examples which can be further refined for the purposes of a particular organization:
  - The candidate is indeed an object and not an activity
  - The candidate is relevant in the value chain
- Consider existing best practices for how to design BBBs and follow it where applicable, e.g.
  - Chose the BBB's name with caution. The name should not have a different meaning in another department or in another process context as this may lead to confusion, i.e. avoid homonyms.
  - Model identifying numbers as attribute of the BBB and not as separate BBBs. Avoid modeling particular aspects of a BBB as a separate BBB. This keeps the BBB model as lean as possible.

Finally, the BBB needs to be described accurately. A precise definition of each BBB candidate is a pre-requisite for the next step. This is often a difficult task. At first glance some objects may seem identical. However, when discussing a definition, it becomes evident that people have a quite different understanding.

As a result of this step, a clear understanding of the topic at issue is provided.

2) Integrate BBB candidates into the BBB model: The BBB candidates identified in the previous step are now added to a BBB model one after the other as long as the existing model does not already contain a corresponding BBB. For each candidate which is added to the model it is checked where it fits into the existing model:

- Analyze for duplicates and similar objects in the existing business building block model. There might be similar BBBs although having a different name, i.e., the BBBs refer to the same business aspects with different names. In order to check for duplicates, the definitions of the BBBs are very useful. Comparing the definitions, we might see that our candidate and the BBB from the BBB model are indeed duplicates. In order to resolve such duplicates, it might be necessary to bring all relevant stakeholders together and to agree upon a definition with all of them in order to ensure a common understanding. Conversely, the definitions might reveal that there are differences between the two BBBs at issue. These differences have to be clarified and the BBB definitions adjusted.
- Add the candidate as a new BBB to the BBB model.
- Relate the new BBB to existing BBBs in the BBB model. These references between BBBs are especially important as they allow for further analysis, e.g., with respect to required IT services.

Having completed this step for all candidates, we have a BBB model without redundancies which provides a clear understanding of the domain. It fosters a common understanding between all stakeholders and documents it with the help of the BBBs and their references. Defining BBBs accurately creates a shared understanding among the stakeholders and mitigates the risk of misunderstandings.

C. Assigning Responsibilities for BBBs

The BBBs are used as basis for coordinating enterprise transformations. Responsibilities are assigned to BBBs, stating thereby who is responsible for what. For example, someone might be responsible for providing the required IT functionality around a BBB. Someone else might be responsible for defining the processes around a BBB. Depending on the requirements of the enterprise, different types of responsibilities can be defined, i.e., who is responsible for a specific item (such as process definition, requirement gathering, etc.). In the above example, a type of responsibility might be the IT support for a BBB.

We clearly distinguish the cases in which someone only requests a BBB. For example, someone needs to use the IT functionality around a BBB. This allows for clearly separating the issues addressed by different stakeholders, projects or systems.

For each BBB, it is documented, who is responsible for the BBB (with respect to a certain aspect) and who requests this BBB (with respect to a certain aspect). Documenting all responsibilities and requests provides a type of contract and ensures planning reliability for all involved parties.

The party responsible for a BBB must not necessarily be a physical person. It is important to be able to conclude which persons you have to talk to if there is more than one claim for the responsibility of a BBB. For example, we might also choose projects to be responsible. We recommend to decide on the allowed types in order to ensure consistency. Which types are appropriate, however, heavily depends on the organization way of working.

In the following, we explain what it does mean to be responsible for a BBB and how to proceed if several parties claim the responsibility for a BBB.

1) The Responsibility for a BBB: While there might be many stakeholders who are interested in a BBB, the objective is to agree on a single responsible authority who accounts for the BBB (concerning a certain aspect). All requests concerning this BBB are addressed to this authority. This avoids duplicate work as the party responsible can coordinate the different requirements. As master for the BBB, it is entitled to apply for resources and to decide on the BBB.

However, having the responsibility for a BBB, also comes up with some duties. The responsible authority has to ensure the re-usability of a BBB, for example by providing services for the usage of the BBB to other authorities that request this BBB. Note that the party responsible for a BBB may delegate certain tasks but it is ultimately answerable.

2) Assignment of Responsibilities: Every BBB should only have exactly one responsible authority. In practice, however, existing process and IT landscapes often lead to more than one claim for the responsibility of a BBB. When no unambiguous responsibility can be assigned, we recommend to initiate an escalation process aiming at an agreement meeting in order to determine which authority will be granted the responsibility for the BBB in question.

In the first step, the persons with conflicting responsibility claims for a certain BBB are identified. The enterprise architects are responsible for gathering the view of all stakeholders independent in advance. Based on this information, they elaborate alternatives and evaluate them.

Second, the enterprise architect invites all involved stakeholders to the agreement meeting. The collected advantages and disadvantages for each alternative are to be presented and discussed in the agreement meeting. Finally, a decision about the future responsibility has to be made. Thus, the agreement meeting results in a single responsible authority for a BBB.

Once the single responsible authority is determined, all other authorities will have to express their requests regarding the BBB instead of claiming to be responsible. The responsible authority, in turn, will have to provide corresponding services to the requesting authorities.

D. Clustering of BBBs

Structuring the architecture via domains allows for a repeatable structure that it is easy to remember. Domains, however, represent a static structure. According to our experiences, it is unlikely that the different concerns of all stakeholders can be addressed by the same structure. Moreover, domains can only be changed with large efforts. Therefore, we suggest a dynamic structure, namely a clustering mechanism, that is adaptable to different user groups. Thereby, the BBBs are categorized in different dimensions, the so-called clusters. This is similar to assigning specific flags to objects with similar characteristics. For example, we detail which organization units are working with a BBB. We can also tag a BBB with the information whether it is used in all subsidiary companies or only in the headquarters. Cluster assignments can be used for filtering relevant BBBs, for adding meta-information to a BBB model and as replacement for domain models.

A formerly used domain model can be represented by cluster. By defining a cluster for each domain of a domain model, the BBBs can be arranged in the same structures as according to the domain model. The benefit of cluster assignments, in contrast to static domains, is that they are easily adaptable to business changes and can be tailored to the particular needs of different stakeholders.

Clustering will be especially useful if it is supported by a tool. Having the BBBs tagged with the clusters, it is possible to generate dynamically clusters for specific purposes. For example, a cluster could contain BBBs that are relevant for a specific organizational unit, e.g. accounting. Once the affected BBBs are assigned to an accounting cluster, filter criteria can restrict visualizations of the BBB model to BBBs relevant for accounting. Such customized views of the BBB model facilitate focused discussions and the coordination between stakeholders. The bigger a BBB model is the more likely it is necessary to generate customized views of the BBB model for certain stakeholders.

IV. The Business Building Block Extension to the TOGAF Content Metamodel

A. Extensions to the TOGAF Content Metamodel

EA frameworks typically provide an architectural model which allows for representing an organization's enterprise architecture. These metamodels define the central artifacts of the enterprise architecture and the relationships between them. Thereby, they determine which types of analysis can be done on the enterprise architecture. The TOGAF Content Metamodel is such an architectural model. It consists of the Core Content Metamodel and several extensions.

The core content metamodel encompasses a minimum set of artifacts for an EAM endeavor. It is closely linked to the phases of TOGAF's Architecture Development Method (ADM) as it includes the information that architects collect and analyze in the respective phases. It partitions the artifacts in five units, first architecture principles, vision and requirements, then the three architecture layers defined by TOGAF (i.e. the business, the information systems and the technology architecture) and last the architecture realization. In order to represent the application architecture and its relationship to the technology architecture, for example, the core content metamodel specifies the entities application and technology component and the relationship that an application component is implemented on a technology component. On this basis, analyses such as "show me all applications that are implemented on deprecated platforms" can be made.

Additionally to the core content metamodel, TOGAF defines several so-called extensions for specific purposes. For example, there are extensions for defining services, process modeling and data modeling. They are optional and are used depending on the scope and purpose of an organization's EAM approach. In the following, we provide an extension for business building blocks which can be used in the same manner as the already existing metamodel extensions. The extension is described in the following according to the schema used by TOGAF for the description of extensions.

B. Purpose

The building block extension is intended to allow additional structured modeling of the central business entities and the responsibilities for these business entities. It is intended to increase the transparency on the business level by clearly stating who is responsible for what. Re-usability is enabled by passing on new requirements regarding a BBB to the authority responsible for this BBB.

Business building blocks are strongly related to other artifacts from the business architecture, especially to business processes and business services. The BBB extension directly links BBBs and business services. As business services are linked according to the TOGAF content metamodel to processes (e.g. business service supports process), BBBs are indirectly related to the processes. Thereby, the extension allows a more sophisticated management and governance of changes on business services, IT services, and business processes.

The scope of the extension is as follows:

- Creation of business building blocks that represent the central business entities independently from their implementations in order to provide the basis for discussing business demands and responsibilities on the business level (instead of a technical level).
- Creation of authorities in order to clearly state responsibilities, such as providing an interface to grant access to the data representation of a BBB, and to document requests for a BBB, such as the requirement for such an interface (for details see section III-C).
- Creation of BBB clusters that group BBBs for governance and communication purposes and to ensure transparency (for details see section III-D).
- Creation of business building block diagrams with all business building blocks and their relationships. The business building block diagrams may be restricted to the BBBs that are assigned to a particular cluster in order to show the entities relevant for a certain stakeholder.

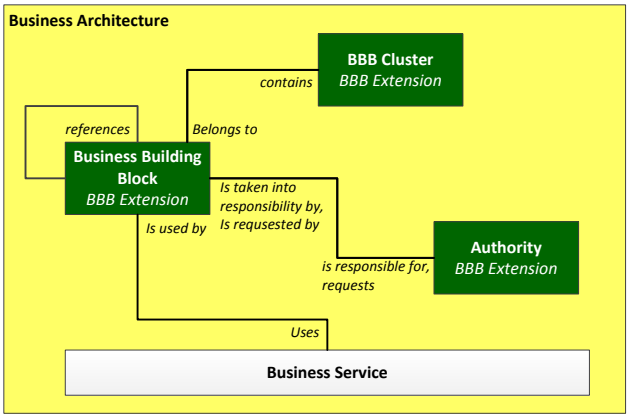

Fig. 2. Business Building Block Extension: Changes to Metamodel

- Creation of responsibility diagrams of the architecture to show who is responsible for a BBB, for example, who is responsible for implementing an IT service that provides functionality around the BBB.

The extension should be used in the following situations:

- Where the architecture should pay special attention to feature re-usability across business units and avoid duplicate work by clearly separating the issues addressed by different stakeholders, projects or systems.
- Where responsibilities should be discussed first from a business perspective. Focusing on BBBs as central elements avoids discussions which mix up the technical details of the implementing IT systems and responsibilities.

The benefits of using this extension are as follows:

- Transparency on the essential business elements on which an organization spends much effort, and for which process definitions and IT solutions must be available.
- Show potential for optimization and re-use across business units and provide based on overlapping claims for responsibilities
- Document assigned responsibilities and thereby provide planning reliability for all involved parties.

C. Required Changes to the Metamodel

Changes to the metamodel entities and relationships are shown in figure 2.

Changes to the metamodel entities and relationships are as follows:

- Business building block is added as a metamodel entity encapsulating the central objects of an enterprise's business activities. Hierarchies of BBBs as well as dependencies between business building blocks are explicitly represented. References between BBBs should be directed and named.
- Authority is added as a metamodel entity. This element can be used to represent both responsible authorities as well as requesting authorities. This is straightforward, as an authority might be responsible for a certain BBB while requesting several other BBBs.
- BBB Cluster is added as a metamodel entity grouping BBBs. It allows for creating dynamic domains.
- If the BBB extension is used, business services will have a relationship with BBBs. Via the business services, BBBs are thus linked to the respective IT implementations as well as to processes. Using this reference provides the basis for further analysis.

Changes to the metamodel attributes are as follows:

- Attributes are added to the new metamodel entity of business building block.

Additional diagrams to be created are as follows:

- business building block diagram
- responsibility diagram

V. Experiences and Best Practices

A. General experiences

In the last years we applied the methodology described in the last sections in several EAM projects. The approach of defining BBBs and their authorities has been proven to be useful in the following cases:

- *New IT-systems* that should be developed (new authorities). For those systems it is necessary to integrate them into the existing IT-landscape and reuse already existing building blocks. The methodology is suited for strategic (top-down identified) projects as well as for operational (bottom-up identified) projects where new IT-systems are proposed.
- Evaluate the dependencies and overlaps of *existing IT-systems* in an organization in order to create a target architecture. A frequent scenario at our customers is that two running IT-systems are supposed to do the same. The enterprise architects are then asked to verify whether this is the case. Instead of looking into the implementation, we check this on the business level. BBBs are identified according to the business purpose of both systems. This analysis might reveal that both systems claim responsibility for the same BBBs and that they consequently (partially) fulfill same functions.
- *Support mergers* between different organizations where existing IT-systems in both organizations exist and should be consolidated. With our methodology this can be started on the business perspective and the consequences can be evaluated when one IT-system is exchanged by another. Our approach identifies existing functionality that can be reused on the business level and clearly assigns responsibilities.

In the following we elaborate on the experiences that we've made within these projects in each of the steps of our approach.

TABLE II
METAMODEL OBJECTS IN THE BBB EXTENSION

Metamodel Object	Description
Business Building Block	A Business Building Block denotes an object that is in tight relationship with the business activities of an enterprise. Business Building Block can have relationships to each other and are used by business processes. They are independent of IT System implementations; for example an order might be a business building block.
Authority	An authority is either responsible for a business building block or requests a business building block.
BBB Cluster	A BBB Cluster provides a classification of business building blocks; for example, organization units might be used as cluster for BBBs.

TABLE III
METAMODEL OBJECTS IN THE EXTENSION AND THEIR ATTRIBUTES

Metamodel Object	Metamodel Attribute	Description
Business Building Block	BBB Attribute	For example, we have a BBB *order* with attributes like an *order number*.

TABLE IV
METAMODEL RELATIONSHIPS

Source object	target object	name
Business Building Block	Business Building Block	references
Business Building Block	Authority	is requested by / is taken into responsibility by
Business Building Block	Business Service	is used by
Business Building Block	BBB Cluster	belongs to
Authority	Business Building Block	requests / is responsible for
BBB Cluster	Business Building Block	contains
Business Service	Business Building Block	uses

B. Best practices

1) Identify BBB candidates: In order to find the right level of detail for a BBB it is helpful to start with the more general term and definition and only refine this when the information sources clearly show that these are two different BBBs and an agreement between both responsible authorities has been found. When identifying BBBs with too much detail, no commonalities can be found between BBB candidates and existing BBBs. Hence, the BBB model will grow too fast and not add value.

With several enterprise architects involved in different projects, it is necessary to align and synchronize these architects at least once a week in order not to come up with different BBB candidates describing the same thing that are discussed in different business fields.

2) Integrate BBB candidates into the BBB model: When integrating BBB candidates into the BBB model, avoid redundant references between BBBs: it is for example recommended to use a single directed link between two BBBs instead of two links in both directions. Each link should be readable from the source BBB to the target BBB, e.g. *order* must be existent for *order confirmation*.

If two BBBs are already linked via a third BBB, try to avoid adding a direct reference between these two BBBs as long as it does not add additional information. This helps to keep the BBB model readable.

3) Assigning Responsibilities for BBBs: When there are quite different viewpoints, it is likely that no decision will be made in an agreement meeting. The project leaders should hence meet in advance to discuss the different viewpoints and agree at least on all advantages and disadvantages that the alternatives would have. In the final agreement meeting, the participants are then asked to make the decision considering these (dis-)advantages.

In each agreement meeting not only the project leaders, but also all responsible managers should be present. Without management attention decisions made in an agreement meeting will not last, but will soon be forgotten by project leads.

In order to get sustainable decisions, all discussions and the final decision should be recorded in writing. These minutes should be distributed to all attendees to have a possibility to demand the right implementation afterwards.

Sometimes the participants are not willing to come up with a common decision. It is helpful to demonstrate the negative effects (such as inconsistent data, redundant implementation, higher costs for development and maintenance) if a decision is not made. This encourages the managers to come to an agreement.

4) Clustering of BBBs: Start with only a few clusters in the beginning. If too many clusters are available that need to be considered for each BBB candidate, this leads to additional work. When defining some cluster groups first, these can easily be extended later. Typical cluster groups are organization units, area of the reference process, type of customers, and so forth.

New clusters can then be added to the cluster group easily, e.g. when new business fields are identified. A typical example is where the possibilities of the internet provide new products such as online shopping, mobile payment, etc. With static domains that often include parts of the functionality of such a new business field, a complete restructuring of the domains would be necessary.

VI. Conclusion and Future Research

In this paper we have described an approach to coordinate enterprise transformations by using so-called business building blocks. We implemented the BBBs as extension to the TOGAF Content metamodel and detailed several best practices that we gained in customer projects in the last years.

A business building block (BBB) denotes an object that is in tight relationship with the business activities of an enterprise. It is a central result in the enterprise's value chain. Responsibilities are assigned to these BBBs. We suggested an approach to identify such BBBs and to determine the party responsible for these BBBs. Responsible authorities are answerable for a BBB and coordinate all requests concerning the BBB. Having agreed upon such an ambiguous responsibility helps avoiding duplicate work. Knowing all requests, the party responsible can bundle similar requirements, and last but not least can prioritize them. Both the party responsible and all stakeholders with requests for a BBB have a better planning reliability.

Since BBBs are elements of the business architecture and are defined in the terminology of the business units, necessary transformations concerning a BBB can be efficiently discussed with business units and executive management. Their abstraction from IT implementation prevents discussing topics with implementation details which is often difficult to understand for business units. On the other hand it allows to plan and agree on requirements of the business rather than on IT requirements. However, as BBBs are (via business services) connected to their IT implementation, they provide a good starting point to understand the IT landscape and in particular the business support of the IT systems. They provide hints for improving the business-IT alignment. Allowing for a cross-domain analysis, BBBs show potential for re-use and thereby reduce the complexity of the whole IT landscape.

We showed that using BBBs rather than classical domain model approaches leads to the following advantages:

- BBBs are more fine-grained than domains.
- Hence, BBB candidates and their definitions can be discussed with each single stakeholder, not with all contact persons from the whole domain.
- In the ideal case only one authority (e.g. project or IT system) exists that is responsible for the design of the BBB, others interact with this IT system via interfaces.
- When several IT systems in one domain exist, it is unclear where the differences between those systems are. When assigning them to BBBs, the differences can easily be seen by all participants.

Future topics for investigation and challenges for further research include intelligent layout mechanisms for showing all BBBs and their references in an easy comprehensible way. Moreover, enterprise architects would benefit from machine support for finding similarities between existing BBBs and new BBB candidates.

A further topic is to state the support of the business strategy by BBBs more precisely. Only topics that support the strategy or strategically relevant business capabilities should be considered in an enterprise transformation. That is, bottom-up identified BBBs (e.g. from process models) need to be evaluated whether they actually support the strategy or not. The presented approach provides a stable basis for these extensions.

References

[1] S. Aier, S. Buckl, B. Gleichauf, F. Matthes, C. M. Schweda, and R. Winter. Towards a more integrated ea planning: Linking transformation planning with evolutionary change. In *The 4th International Workshop on Enterprise Modelling and Information Systems Architectures: Concepts and Applications (EMISA 2011)*, pages 23–36, 2011.

[2] S. Aier, C. Riege, and R. Winter. Unternehmensarchitektur Literaturüberblick und Stand der Praxis. *Wirtschaftsinformatik*, 50(4):292–304, 2008.

[3] S. Aier and M. Schönherr. Modellbasierter Entwurf strukturanaloger Architekturen auf Basis der Partitionierung von Graphen. In *Wirtschaftsinformatik (2)*, pages 199–216, 2007.

[4] G. Berrisford and M. Lankhorst. Using ArchiMate with TOGAF. *Via Nova Architectura*, 2009.

[5] B. Burton and P. Alega. Hype cycle for enterprise architecture. Technical report, Gartner Inc., 2010.

[6] B. H. Cameron and E. McMillan. Analyzing the currrent trends in enterprise architecture frameworks. *Journal of Enterprise Architecture*, pages 60 – 71, February 2013.

[7] B. Daum. *Modeling Business Objects with XML Schema*. Morgan Kaufmann, 2003.

[8] G. Dern. *Management von IT-Architekturen*. CIO - IT-Strategie für Manager. Vieweg and Teubner, 2009.

[9] DoD. Department of defense architecture framework, 2010.

[10] G. Engels, A. Hess, B. Humm, O. Juwig, M. Lohman, J.-P. Richter, M. Voß, and J. Willkomm. *Quasar Enterprise*. Dpunkt.verlag Gmbh, Heidelberg, 1 edition, 2008.

[11] G. Engels, M. Kremer, T. Nötzold, T. Wolf, K. Prott, J. Hohwiller, A. Hofmann, A. Seidl, D. Schlegel, and O. Nandico. *Quasar 3.0 - A situational approach to Software Engineering*. Capgemini, 2012.

[12] W. Keller. *IT-Unternehmensarchitektur*. dpunkt.verlag, Heidelberg, 1 edition, 2007.

[13] R. Lhotka. *Expert C# 2008 Business Objects*. Apress, 2004.

[14] D. Matthes. *Enterprise architecture frameworks kompendium: Über 50 Rahmenwerke für das IT-Management*. Springer, 2011.

[15] S. Murer, B. Bonati, and F. J. Furrer. *Managed Evolution: A Strategy for Very Large Information Systems*. Springer Berlin Heidelberg, 1st edition. edition, Oct. 2010.

[16] J. Nicola, M. Mayfield, and M. Abney. *Streamlined Object Modeling: Patterns, Rules, and Implementation*. Prentice Hall, 2002.

[17] A. W. Scheer. *ARIS – Business Process Frameworks*. Springer, 3 edition, 1999.

[18] A.-W. Scheer. *ARIS - Business Process Modeling*. Springer, 3 edition, March 2000.

[19] F. Schönthaler, G. Vossen, A. Oberweis, and T. Karle. Concepts and modeling languages. In *Business Processes for Business Communities*, pages 21–60. Springer Berlin Heidelberg, 2012.

[20] The Open Group. *TOGAF Version 9 - A Manual (TOGAF Series)*. Van Haren Publishing, 9 edition, 2009.

[21] The Open Group. *ArchiMate 2.0 Specification*. Van Haren Publishing, 2012.

[22] J. Zachman. A framework for information systems architecture. *IBM SYSTEMS JOURNAL*, 26(3), 1987.