SEAM FRAMEWORK

A lightweight approach to a goal-oriented management of enterprise architectures
Executive Summary

Constantly altering market conditions, technological progress, and new legal constraints force today’s enterprises to undergo drastic transformations. In the course of last decade, Enterprise Architecture Management (EAM) has turned out to be a valuable instrument to deal with the challenges arising in the process of transformation. Focusing on the continuous alignment of business and IT, EAM considers an enterprise from a holistic perspective. Accordingly, the discipline accounts for business-related elements like processes and organizational units, technical entities like applications and infrastructure components, as well as their different types of relationship.

To date, more than a dozen frameworks for EAM are available, each of them set out to support enterprise architects in their work. However, most of these oftentimes text-intensive, multi-focused, and partially inconsistent approaches intermingle EA, EAM, and the adjustment of the management itself while requiring significant amount of time until being fully understood and ready to use in practice.

In response to this unpleasant situation, this document introduces the SYRACOM EAM framework SEAM, a lightweight, goal-oriented, and easy-to-grasp approach for EAM. The practice-proven framework has been successfully applied since 2003 within more than 25 industry projects at 15 different organizations from the automotive, aviation, energy, financial, logistics, and telecommunication industry. During the projects, which comprised more than 2,500 person days of work with the framework, SEAM has been iteratively refined, extending and aligning its structure and categories with the actual needs of the enterprise architects. Moreover, additional content increments were added every time a novel method, deliverable, goal, principle or risk was specified. Pursuing business as well as technological professions, our clients and partners both appreciate SEAM as a framework that is comprehensive, consistent, and minimal.
ACKNOWLEDGEMENTS

We express our gratitude to all colleagues of SYRACOM for their viable feedback on our paper. Furthermore, we thank our client the financial group, for giving us the opportunity to put SEAM into action within an industrial setting. Last but not least, we want to thank Dr. Gregor Kaczor, Senior consultant at SYRACOM AG, and Prof. Dr. Gerhard Wanner from HFT Stuttgart for their detailed review, lifting the document’s quality to a higher level.
# Table of Contents

1. Introduction ........................................................................................................................... 6
   1.1 Motivation .................................................................................................................... 6
   1.2 Who we are .................................................................................................................. 7
   1.3 Document structure ...................................................................................................... 7

2. Explaining SEAM – constituents and design ................................................................. 8
   2.1 Conception areas and internal components .............................................................. 8
   2.2 External Components ............................................................................................... 18
   2.3 Solution building block ........................................................................................... 20

3. Initiating SEAM – framework application ................................................................. 21
   3.1 7-step application method ....................................................................................... 21
   3.2 SEAM and governance ............................................................................................ 23

4. Exemplifying SEAM – a case study from the financial industry .................................. 24
   4.1 Organization ............................................................................................................. 24
   4.2 Context of the case study ......................................................................................... 24
   4.3 Course of action ....................................................................................................... 24
   4.4 Results ....................................................................................................................... 26

5. Summary .............................................................................................................................. 27

6. References ........................................................................................................................... 28

7. Nomenclature ...................................................................................................................... 31
1. **INTRODUCTION**

1.1 **MOTIVATION**

Within recent years, much has been written about the management of enterprise architectures, both in academic (cf. [7, 26]) and practitioner circles (cf. [9, 20]). Driven by constantly changing business, technical, and regulatory demands, the discipline’s most important task consists in ensuring the continuous alignment of business and IT of an enterprise [41].

While an enterprise can be defined as any collection of organizations that has a common set of goals, an extended enterprise comprises partners, suppliers, and customers, too [37]. From our understanding, each organization has an implicit Enterprise Architecture (EA) that can be made explicit by means of specific descriptions. These descriptions are expressed by means of deliverables that concentrate on certain aspects of the EA. An EA’s description is limited in scope, level of detail, as well as the point in time/period it is making explicit.

In line with the ISO/IEC 42010:2007 [16], SYRACOM defines an Enterprise Architecture (EA) as “The fundamental organization of a system [the enterprise] embodied in its components [the elements], their relationships to each other, and to the environment, and the principles guiding its design and evolution.”

Elements can be of immaterial (e.g., software) or material (e.g., data racks) nature, refer either to activities (e.g., business processes), resources (e.g., business application), or organizations. Elements can be grouped into different categories they belong to. Often, these categories are either called architectures (e.g., business, application, and infrastructure architecture).

As any other managing discipline, Enterprise Architecture Management (EAM) serves at achieving previously defined goals [19]. Usually, these goals pertain to different elements of the EA including their properties and relationships. Common goals for EAM are, among others, the standardization and consolidation or the cost reductions [19]. To reach a goal, EAM needs to take a holistic stance. It further has to provide means to plan, organize, implement, and monitor the description, definition, as well as transformation of the EA, consequently the enterprise itself. Lastly, EAM also has to account for goals pertaining to the discipline itself, thus adjusting the way EAs are managed.

Plenty of frameworks exist to facilitate the task of managing an EA, i.e., successfully reaching the given goals (cf. [42]). There are widespread representatives like the consortia-developed The Open Group Architecture Framework (TOGAF®) [37] or the proprietary Zachman [43], but also less prominent approaches like the university-driven Building blocks for Enterprise Architecture Management Solutions (BEAMS) [32] or the open-source designed Purdue Enterprise Reference Architecture (PERA) [29]. Having worked with them during past assignments, we remarked that most of these frameworks:

- do not strictly differentiate between an EA and its actual description,
- intermingle EA, EAM, and the adjustment of the EAM itself,
- need a significant portion of time to grasp basic concepts,
- do not explain how the approach can be adapted to the organization,
- are often very voluminous, text heavy, and multi-focused, and
- are partially inexact, incomplete, inconsistent, and/or unclear.

Above detriments prompted us to go back to the drawing board to design SEAM, the EAM framework of SYRACOM. This document presents our goal-oriented, consistent, and easy-to-grasp approach.
1.2 WHO WE ARE

Since 2003 the tool-independent SYRACOM Business Unit EAM successfully advises clients in their architectural endeavors. Among others, previous and current mandates range from business/IT transparency, IT standardization, EA tool introduction, EAM assessment, and optimization projects. All client-specific results benefit from the goal-oriented, consistent, and lightweight EAM framework SEAM (SYRACOM EAM Framework). The practice-proven instrument combines the advantages known from widespread frameworks like TOGAF® 9.1, Zachman 3.0 and IAF 9 while being continuously improved together with researchers from German universities. In addition to the know-how and experience gained within projects and further training, all SYRACOM EAM consultants are TOGAF® level 2 certified. Moreover, the company actively supports the Open Group in improving its EA framework. At regular intervals, SYRACOM organizes the EAM Think Tank inviting clients to discuss best-practices, lessons-learned, and stumbling blocks in EAM. Please refer to the official website of the SYRACOM Business Unit for additional details [36].

http://www.SYRACOM.de/EAM

1.3 DOCUMENT STRUCTURE

The document is structured as follows: Chapter 2 presents the constituents of SYRACOM’s company-internal framework SEAM, a goal-oriented approach to EAM we have been using for the last 10 years within more than 25 industry projects. The application of SEAM is outlined in Chapter 3. Using a case study at a financial group, the document further shows how SYRACOM usually applies SEAM (Chapter 4).
2. Explaining SEAM – constituents and design

This Chapter introduces the elements of SEAM and their relationship to each other. SEAM applies basic and simple-to-remember terms that are also known to people who are not experts in the domain. From our experience, avoiding overly complex jargon allows for a much easier and smooth integration of EAM within an organization while reducing the chance of an ivory tower (cf. [39]).

2.1 Conception areas and internal components

The conception areas Strategy, Method, Organization, Tool, and Deliverable form the principal foundation of SEAM (cf. Figure 1). Each of the five areas comprehends at least one framework-internal component either referring to the EA or its management. While the former helps to describe, define, and transform an EA, the latter enables the (reflexive) description and adjustment of the management thereof. One common example for an EA method component would be “Standardize business applications”, a typical method targeting at the EAM is “Conduct EAM assessment”. Both, EA as well as EAM components, bear an unambiguous and self-explanatory name.

As also depicted in Figure 1, SEAM is further subdivided into three separate layers. While the Structure layer constitutes the conceptual foundation, the two overlying layers include more tangible representatives. The higher a layer is positioned, the more concrete components it possesses. However, the components of lower layers are less specific, thus can be used as abstract prescriptions for different EAM initiatives.
Figure 2 illustrates the Structure layer of SEAM. This document centers on the components found on this layer.

Figure 3 illustrates the relationships between the internal components of SEAM’s Structure layer as described in the subsequent Section of this Chapter. Note, that the model, which is based on ArchiMate [38], does not differentiate between EA and EAM components. For reasons of clarity, all relations accounting for the (reflexive) description, definition, and transformation of EA/EAM components have been omitted. Strictly speaking, the EAM method possesses additional associations to the goal, principle, risk, KPI, deliverable, tool, committee, and role component possessing a *-multiplicity on both ends and a “describes, defines, and transforms” name. Finally, the model does not distinguish between input and output deliverables.
Components can be categorized by means of various criteria (e.g., purpose, point of time when being applied, responsible organizational unit). Categorized components form the Categorization layer of SEAM.

Figure 4 illustrates our basic categorization scheme we make use of during our client engagements. As depicted, deliverables are divided into reports, diagrams, tables/matrices, lists/catalogs, and (meta-)models.

For their application, components have to be realized first before being instantiated afterwards, i.e., adapted to the particular needs of the organization. Realizing signifies that a specific component has to be developed. For instance, an EA deliverable (e.g., process-support map) is crafted or a certain EA method (e.g., gather data about business applications) is designed. The sum of all realizations is placed on the Realization layer of SEAM. In our practical work, we leverage this content, tailoring suited realizations to the need of the clients.

This document specifies the individual components of the Structure layer in addition to their relationships. To foster understanding, we sketch several examples at the end of each explanation and point towards further literature we have considered when working on the topic.

2.1.1 STRATEGY

The conception area Strategy gives reasons why EAM has to be established within an organization. Furthermore, it defines boundary conditions that have to be paid attention to when performing the architectural work. In this sense, the conception area provides means to specify the problem EAM afterwards has to cope with.

EA/EAM goal

EA and EAM goals express a desired (or even fundamentally necessary) clearly defined state for a future point in time. Transitioning to this state brings along positive (and sometimes also negative) consequences for the enterprise, for instance cost reduction or transparency over the application landscape. If possible, these consequences should be rendered quantifiable allowing for the evaluation of the goal. While EA goals apply to the EA, EAM goals attempt to optimize the management thereof. Both goal types
should have a succinct and memorable name to ease communication. Being derived from (not necessarily explicitly formulated) business and/or IT goals, EA/EAM goals can have an operative, tactical, or strategic trait [3]. They can change over time, for instance, due to influence factors, new findings, or altering business/IT goals. Goals have to be methodologically defined in order to render them unambiguous and consistent. Among others, specified goals contain the focus area (a.k.a. area of interest or concern in [32]), a deadline, and different restrictions (e.g., goal only applies to a certain organizational unit or location). EA as well as EAM goals are reached through executing methods that generate deliverables required to address the goal. While a goal’s degree of achievement can (and sometimes must) be measured via Key Performance Indicators (KPIs), its final achievement can be jeopardized through risks.

Generally, EA/EAM goals either demand for describing, defining, or transforming the EA or the management thereof. Current EA elements, EA/EAM components, their properties, and relationships are described. A future state of EA elements and EA/EAM components, however, must be defined. Once the current as well as the future states are specified, transformation goals call for the definition of a roadmap detailing on how to get from the one state to another (cf. [1]).

Goals can have different forms of relations to each other (e.g., complementation, conflict, specification, generalization) [15, 28]. For instance, an organization may decide to pursue a specific subgoal in the form of a pilot project before tackling the goal in a whole. Goals should not be specified on a meta-level by uniquely referring to other goals (e.g., “The goal not to have a goal”). Furthermore, EA goals can exert influence on their EAM counterparts and vice versa (e.g., Reducing costs for EAM vs. Expensive gathering of data about the as-is application landscape). Without any existing EA goal, there should not be an EAM goal at all. Lastly, goals can propagate, e.g., first have an impact only on the business architecture but later on also require modifications in the application architecture.

- **EA goal:** Transparency over the current applications in Berlin until 05/13
- **EAM goal:** Permanent cost savings for gathering EA data

A list of possible EA/EAM goals can be found in [3, 28]. To the present day, research on the structure of such goals is rather sparse.

**EA/EAM Key Performance Indicator**

Key performance indicators (KPIs) are numbers, which, at a certain moment, make an objective and reproducible statement about a planned or actual property state of an object. Thereby, a property can refer to either a certain point in time or a period, have a qualitative/quantitative character, are relative or absolute numbers [18].

EA and EAM KPIs aim at measuring the achievement of at least one corresponding EA/EAM goal (cf. [22] for EA goals and associated KPIs). In general, multiple KPIs can be employed to measure the same goal. As an advantage, indicators validate if the goal has been understood correctly. However, the danger lies in selective optimization while blanking out other important factors not in scope of the measurement.

As for goals, EA as well as EAM KPIs should have succinct and memorable names starting with a noun. Inevitably, the application of a KPI entails additional effort. Time and money is needed to design and maintain the indicator as well as to gather the required data about EA elements, EA/EAM components, their properties, and relationships. Since this work does not directly contribute to the achievement of the underlying goal, the indicator’s value and effort for application should be assessed regularly through a cost/benefit analysis [18]. Lastly, KPIs can be embedded in deliverables [21].
• EA KPI: Total number of multi-tenant business applications
• EAM KPI: Costs and time needed for gathering data about all IT components

More information about KPIs in EAM particularly regarding their attributes can be found in [22]. A set of more than 50 different EA KPIs is contained in [23].

**EA/EAM principle**

EA/EAM principles are unambiguously, enduring, and binding conventions restricting the way to achieve a certain EA/EAM goal. Being motivated by drivers like values, risks, potential rewards, constraints [12], they support at least one business/IT goal [34]. However, this goal is not necessarily linked to the EA/EAM goal whose achievement they constrain. Consequently, EA/EAM principles can be in existence without any stated EA/EAM goal. Furthermore, the supporting principles should be revised if the business/IT goal changes or is decommissioned.

As a rule of thumb: the more precise and clearly a principle is defined, the easier it is to follow. However, a self-evident and easy-to-grasp principle does not automatically imply its adherence. On the contrary, overlong and too sophisticated principles complicate communication and compliance. Another issue is the consideration of the (partly very strong) horizontal and vertical influence range of principles within the enterprise, in particular if defined and enacted by a distant organizational unit not impacted by the convention. If clearly defined and communicated, principles provide a stable and consistent solution space for the selection and execution of EA/EAM methods. The space should be neglected only under exceptional circumstances. In detail, principles represent rules which have to be taken into account when describing, defining, or transforming EA elements, EA/EAM components, their properties, and relations. Influencing the choice and implementation of methods, principles have also an indirect impact on the tools, the organization, and the deliverables being produced.

If more than one EA/EAM principle is applied, their (possibly negative) interdependencies have to be considered [37]. In particular, the principle execution order should not make any differences for the method solution space. To keep the organization focused, the number of principles has to remain manageable. As for goals, meta-principles, i.e., principles uniquely referring to other principles have to be avoided. Since principles may interfere with goals, they should be always formulated in consideration of the latter. In addition, their adherence should be regularly checked.

• EA principle: Prefer standard software to individually developed software
• EAM principle: Balanced architecture boards containing an even amount of business and IT representatives

A good overview on the attributes of EA/EAM principles can be found in [12]. The book also includes a catalogue with 59 examples. Ten information architecture-related principles can be found in the Oracle white paper of [35].

**EA/EAM risk**

EA and EAM risks can be defined as events that negatively compromise the achievement of at least one EA and EAM goal respectively (based on [13]). Translated into project management terms, risks make the fulfillment of a certain goal more costly and time-expensive provided that desired quality remains invariant. An EA/EAM risk is characterized by the product of probability of occurrence and expected loss. The first parameter refers to the frequency of occurrence for a given period of time, the latter to the incurred costs, both time- and money-wise. For the risk, its probability, and expected loss ordinal scales as proposed for instance by TOGAF® can be used to facilitate architectural work [37].
The root cause for a particular EA/EAM risk can be manifold given the cross-functional nature of the topic. Possible sources are, among others, EA/EAM components, internal and external influences, but also other EA/EAM frameworks and not EAM-related disciplines. Often, a risk can be categorized according to the underlying cause. However, sometimes it’s the combination of different causes that lead to the emergence of a risk (e.g., complicated EA tool + untrained and inexperienced staff). The management, i.e., identification, classification, assessment, mitigation, and monitoring of risks, always tie in with extra workload addressed within existing or new EA/EAM methods. However, as for KPIs, this work does not directly contribute to the fulfillment of the goal but rather increases the cost for achieving it. Thus, risk probability and expected loss have to be carefully pondered against the effort required for the additional management tasks. Besides, any addressing of EA/EAM risks has to be aligned with the general risk attitude of the enterprise (risk averse vs. risk affine).

In principal, there are two complementary alternatives for mitigating an EA/EAM risk:

- Lowering the probability through à-priori measures (e.g., training of enterprise architects)
- Reducing the incurred loss through à-posteriori measures (e.g., existence of an older description of the IT architecture).

After mitigation, the term residual-risk can be used [37]. However, in the course of time, the residual-risk can regrow, making additional mitigation measures mandatory. The management of a particular EA/EAM risk may lead to the growth of another one, not necessarily being associated to EAM. In consequence, the risk management activities need to be synchronized, if needed using dedicated deliverables.

- EA risk: An unavailable information provider possibly hampers the timely gathering of EA data
- EAM risk: An insolvent EA tool vendor possibly requires the transition to a different software support

2.1.2 ORGANIZATION

This conception area defines who actually does architectural work in an enterprise. For SEAM an organization plays a dual role: on the one hand it demands for EAM work on the other it supplies it. The area Organization comprises the three components actor, role, and committee (cf. Figure 2).

An actor assumes at least one role, may work in a committee, and is in relation to other actors. Actors and relationship form the organizational structure. A role can be fulfilled by more than one internal or external actor. From our experience, all duties, responsibilities, and rights of a certain role in EAM should be defined upfront to avoid possible conflicts. This, however, first requires the execution of distinct EAM methods helping to identify the different EAM actors and their roles. Afterwards, actors should be classified, e.g., depending on their level of interest and/or their power (cf. [37]).

Committees (e.g., architecture board, standardization committee) consist of at least two actors. These meet up in regular intervals in order to discuss, consult, prepare decisions, and finally, jointly arrive at conclusions with regards to EA elements and EA/EAM components, their properties, and relations. Furthermore, they fulfill a monitoring, controlling, and escalating function applied during the description, definition, and transformation of the EA and EAM. Composition (including member rotation), size, and area of responsibility of the committee are contingent on the task and should reflect the structure of the organization. Further characteristics of this task-centric body are a limited lifetime coupled with flat hierarchies.

Opposed to other EAM aspects, the topic of risks is rather underrepresented in current literature. Please consider [37] for further reading material on risks in the domain of EAM.
Subsequently, we focus on the different roles constituting important components of SEAM’s Categorization layer (cf. Figure 4). We frequently observe these role types in our work and partly read them up in related literature. Deliberately, we avoid using the (in our opinion overloaded) term “stakeholder”, as for instance defined by the International Organization for Standardization, 2007 or excessively used in TOGAF® [37]. Depending on the organizations type, roles can be further differentiated in order to reflect their focus area (e.g., business architect, IT architect). It goes without saying that each role not only has to be motivated but also capable to perform its duties.

**Sponsor**

As a representative of the top-management (e.g., chief executive/information officer), the sponsor role supports the accomplishment of EA and EAM goals, and therefore the execution of dedicated EA/EAM methods. Preferably, an actor with a sponsor role simultaneously takes up the duties of a beneficiary. In this sense, the sponsor becomes aware of the benefits EAM brings along. Equally important is his/her close participation at regular committee & board sessions.

**Beneficiary**

Beneficiaries (e.g., sales representative, security manager) are actors who have a vital interest in describing, defining, or transforming the EA or its management. In the context of EAM, literature often uses the term stakeholder when speaking of this group. Beneficiaries have the task to formulate EA/EAM goals in addition to the principles which cut through the solution space of possible goal achievement methods. They may further demand for the application of KPIs [22] and the management of risks that might threaten the goals. In principle, the higher a beneficiary in the organization the lesser his/her interest in detailed information concerning the EA or its management. On the flipside, the focus area grows the closer an actor with this role is associated to top management. There are different methods to identify beneficiaries, an overview can be found in [4].

**EA information provider**

An EA information provider (e.g., application owner, IT-service manager) disposes of information about at least one EA element, its properties and/or relations for a given point in time [8]. Also called data-provider by literature [13], this role ensures the achievement of EA description and definition goals. Information providers contribute their in-depth architectural knowledge during the execution of EA methods (e.g., interviews, surveys, data gathering using an EA-tool). Thereby, he or she follows the EA principles, addresses EA risks, and (if needed) captures data for calculating the EA KPIs.

**EA transformer**

As the name indicates, EA transformers (e.g., software engineer, processes re-engineer) are in charge of transforming the EA, hence transitioning at least one EA element, its properties, and relations from the current to a target (intermediate) state. In doing so, they ensure the achievement of EA transformation goals. Similar to their counterparts, the information providers, they respect EA principles, address different EA risks, and gather the data needed for the EA KPIs when performing their work.

**Enterprise architect**

Enterprise architects are responsible for reaching EA as well as EAM goals. Acting as a human link between the different actors and their roles, architects have sound interpersonal, communication, and social skills, possess a broad knowledge of business, legal, and IT [3], and are able to manage projects. The distribution of the actors within an organization as well as their different role-dependent interests can become a challenge for enterprise architects [8]. For instance, sometimes EA information providers do not understand the final benefit of their efforts in timely and regularly providing high quality EA data. As a consequence they behave reluctant and unwillingly. Conversely, EA transformers face problems to put their transformation project in a broader enterprise-wide context. By applying the different role-specific languages, the enterprise architect acts as a messenger between those groups helping to balance the different interests. Much easier for this role
TOGAF® accentuates the consideration of stakeholders as actors requesting and benefiting from architectural work [37]. The Open Group approach also contains information on the role of an enterprise architect in the shape of its capability framework.

2.1.3 METHOD

The single-component conception area Method defines how the EA and its management are performed. Acting as the cornerstone of SEAM, EA and EAM methods represent systematic approaches aimed at achieving one or more EA and EAM goals (in part). Having a succinct and memorable name that starts with a verb and terminates with a noun (e.g., Capture sponsors, Describe current IT-architecture) methods consist of several single steps whereas each step can be a method again. The period of executing a certain method is always contingent on the goals it pursues. If all underlying goals have been achieved or were marked as being invalid, the method becomes obsolete. If, in turn, the goal has to be re-achieved, only selected steps of the method have to be repeated (e.g., Create transparency of the current application landscape). Principles have an impact on how goals are achieved, thus which method (steps) can be executed. Altering goals and principles require the revision of a method. As illustrated in Figure 4, methods can be categorized differently (cf. [32]).

The execution of EA/EAM method can be based on newly defined or existing projects and processes. While projects strive for the one-time accomplishment of a goal, embedding methods in processes attempts to achieve goals repeatedly. Despite the latent fear of overloading established processes, enterprises should tend to use this type of carrier for reaching their EA/EAM goals [2]. Irrespective, every new process or project entails the decision, whether this initiative should be conducted centralized or decentralized (e.g., gathering and maintaining information about IT components).

Describing and defining an EA or its management can be done on the basis of processes and projects. By contrast, any subsequent transformation is performed by means of existing or newly created projects.

In practice and literature describing & defining the EA/EAM in addition to transforming the EAM is often subsumed under the umbrella term EAM initiative (also endeavor, undertaking, etc.). Surprisingly, for transforming the EA, i.e., changing the enterprise, the phrases business and/or IT project are commonly applied.

EA as well as EAM method generate at least one deliverable, and (occasionally) require one or more deliverables to be properly executed. The verb in the method’s name should reflect the expected main deliverable, e.g., “visualize” for diagrams, “model” for meta-models, “relate” for tables & matrices, and “list” for lists and catalogues. Both types have to be executed by one or more actors who are responsible for the course of action and the resulting outcome. Additional actors can be accountable for the method, are consulted during method’s execution, and informed about status and deliverables. RACI matrices, as proposed in COBIT [17], help to capture the degree of participation of a certain role with regards to the different method steps prior to its first execution (cf. also [13]).

The application of EA and EAM tools allow for greater me-
Method execution efficiency while reducing the chance for possible errors to occur. In turn, distinct EAM methods serve at describing, defining, and transforming (including introducing) EA and EAM tools. Both method types can be enriched with additional steps ensuring the capturing of KPI data and management of risks. Thereby, the interval of measuring and risk addressing might demand for a more frequent execution of (parts of) the method. Due the usage of identical resources (e.g., EA tools, enterprise architects) EA/EAM methods may interfere with each other. Possible negative relations have to be accounted for by dedicated synchronization mechanisms.

- EA method: Standardize IT components
- EAM method: Assess EA tool support

The topic of methods in EAM is intensively discussed in [6]. For a more practice-related material please consider [13].

2.1.4 DELIVERABLE

Called like its sole component, the conception area Deliverable states the (intermediate) outcome of architectural work, hence the what. Deliverables play a dual role, acting either as input for a downstream method or resulting from a predecessor method’s execution. EA and EAM deliverables are intended to describe (and therewith also to define) EA elements, EA/EAM components, their properties, and relations for a certain point in time. They are used to specify the transformation of an EA and the management thereof, too. A deliverable bears a unique and meaningful name whereas its described content should be unambiguous. This can be achieved with either standardized notations like UML [27] and ArchiMate [38] or a self-explanatory legend. Each type of deliverable has to (partially) contribute to the achievement of one or more goals brought up by the beneficiaries. While some deliverables are directly visible to this role, hence are beneficiary-specific, others remain hidden (e.g., meta-model often unknown to top management which is rather interested in the resulting visualizations). Furthermore, deliverables can be nested, also within artifacts not created by means of an EA/EAM method (e.g., controlling report). Further, each deliverable has a lifecycle that depends on the state of the present or future EA element or EA component. If this state changes so does the deliverable’s lifecycle.

A deliverable may have a varying quality, defined via type-dependent quality factors (e.g., readability, comprehensibility, data accuracy). In accordance to the EA/EAM goal, beneficiaries may call for specific (contractually binding) quality factor levels. When describing a future state of the EA or its management, different variants might exist, which can (as their predecessors) be archived once a new version for this point in time is created (cf. [6]). Inspired by TOGAF®, a possible categorization scheme for deliverables are (meta-)models, lists & catalogues, tables & matrices, diagrams, cockpits & dashboards, and documents [37]. The meta-model plays an essential role since it depicts the focus area specified by the different EA/EAM goals. Being a prerequisite for other deliverables, a meta-model should be correct, complete, consistent, comprehensible, confined, and changeable (cf. 6c goals [25]).

- EA deliverable: IT component/application matrix
- EAM deliverable: EAM assessment report

Reading suggestion: A comprehensive overview on visualizations and meta-models as two important types of EA deliverables can be found in [32]. Spread across the document, TOGAF® also contains several sample EA and EAM deliverables [37].
2.1.5 TOOL

This conception area specifies which means are used to perform the architectural work. EA and EAM tools enable actors in performing at least one EA or EAM method more efficiently while reducing the chance of possible human errors. Thereby, a tool is defined as a software product that has been developed inside or outside the applying enterprise. Being free of charge or under licensing agreement, EA/EAM tools support the goal achievement process through the provision of deliverables. EA and EAM software do not necessarily have to be dedicated software, but rather can be multi-purpose tools also leveraged for non-EAM related tasks (e.g., Microsoft Office). Frequently, a goal can be achieved by using more than one tool. By the same token, a tool often covers several goals. However, sometimes a combination of different tools, specialized, for instance on modeling, data gathering, and/or analysis, is required for accomplishing the goal.

EA tools contain data about EA elements, their properties, and relations. They offer facilities to key in, process, and visualize the information about the EA while storing this data in a database. Furthermore, they provide mechanisms for actor communication, meta-modeling, simulation, and sometimes even non-EAM disciplines (e.g., project portfolio management, IT-service management). EA tools can be described, defined, and transformed (e.g., tool introduction or modification) through EAM methods. Up-to-date information about available EA software can be found, among others, in [31].

In the spirit of SEAM, EA tools facilitate the management of the architecture of an enterprise whereas EAM software helps to capture and optimize the management discipline of an EA. Speaking of software we would also like to point out that we frequently observed organizations virtually being “slaves” of their EA tools. From our standpoint this mindset should be adjusted in the way tools are considered as pure instruments for a more efficient and less error-prone goal fulfillment.

- EA tools: leanIX from leanIX GmbH
- EAM tools (not dedicated): Microsoft Excel
2.2 EXTERNAL COMPONENTS

There are components that are not part of an organization’s EAM, but however, exert influence on the discipline, complement it, or fall under its influence. SEAM accounts for this fact as illustrated in Figure 5 and explained in the following Sections.

Influences of EAM

2.2.1 INFLUENCE ON SEAM

An enterprise is exposed to various influence factors, also called contextual [40] or contingency factors [30] as well as the organizational context [32]. These factors stem from organization-internal and external sources often categorized in business, technology, and legal [6]. Ranging between short and long-term, influence factors can have a varying impact on the management of the EA (e.g., weak, medium, strong). They may either benefit the enterprise or inflict damages to it. In the latter case, they have to be considered as well as treated as EA/EAM risks that hamper the achievement of EA or EAM goals respectively. In line with goals or principles, influence factors interfere, i.e., amplify, complement, or erase each other. Moreover, external
lized, decentralized, federal), as well as the origin (e.g., top-down, bottom-up, pilot) of EAM initiatives [32]. Finally, the adoption of advanced architectural design paradigms and modeling capabilities, deployment and monitoring of EA data and services, and organizational penetration of EAM [30] fall in the category of internal factors, too.

- **Goal**: To achieve a certain business or IT goal, the accomplishment of several EA/EAM goals may become mandatory.
- **Principle**: EA/EAM principles often have to be established due to existing business/IT goals.
- **Risk**: The root cause for EA/EAM risk could be found outside the domain of EAM.
- **Actor**: Actors, usually not working in the domain, might be called in to support the one-time achievement of a certain EA/EAM goal.
- **Method**: Methods from other disciplines are leveraged to pursue the EA/EAM goals.
- **Deliverable**: Non-EA/EAM deliverables might be integrated within deliverables of EAM.
- **Tool**: Software, which is typically not tailored to the management of EAs, can be leveraged to describe, define, and finally transform EA or the management thereof.

Although the causes for external influence factors lie beyond an enterprise's sphere of action, they yet have to be considered.

- **Internal factor**: EAM is unknown to the top-management. Instead, all initiatives are triggered by the IT department.
- **External factor**: Legal regulations, like the Clinger-Cohen Act or the Sarbanes-Oxely Act.

[30, 40] examine organizational factors from an academic point of view. Despite its scientific background, more practice-oriented information can be found in the BEAMS, the EAM framework conceived at TU München, Germany [32].

### 2.2.2 INTEGRATION WITH SEAM

By reason of its solid and well-defined structure SEAM can be easily integrated with other (EA/EAM) frameworks. Below, we briefly sketch out how some of the content of TOGAF® [37] and BEAMS [32] can be mapped to our approach in a straightforward manner.

#### The Open Group Architecture Framework (TOGAF®)

The phases of the TOGAF’s Architecture Development Method (ADM) in addition to the ADM Guidelines & Techniques (cf. [37]) can be directly transformed into EA and EAM methods. Furthermore, the enterprise and architecture principles provided by Open Group’s approach serve as a role model for the SEAM principle component. Enterprise continuum, the Technical reference model and the Integrated Information Infrastructure Reference Model, as well as the architecture repository facilitate the structuring of EA deliverables defined in SEAM. Lastly, the elements of the Architecture Capability Framework (e.g., roles and skills) represent valuable input for enriching the definitions of SEAM’s roles, particular those of an enterprise architect.

#### Building blocks for Enterprise Architecture Management Solutions (BEAMS)

Regarding BEAMS, the 10 goals of the framework in combination with the organizational context can be mapped to SEAM EA goals and internal influence factors components. Furthermore, it is not difficult to translate the method building blocks of the modular approach into EA methods of SEAM. In addition, the configure & adapt part of BEAMS is mapped to EAM methods. Similarly, the viewpoint and information building blocks serve as a valuable foundation for EA deliverables.
2.2.3 IMPACT OF SEAM

Since EAM does not happen in a vacuum but rather has to generate benefit for the organization, other areas rely on its output [13]. Put more precisely, SEAM has an impact on non-EA/EAM disciplines. For instance, with its methods and deliverables, SEAM is well suited to support the discipline of project management, among others, with detailed analyses (e.g., scoping, interfaces), project monitoring (e.g., audits, reviews), or projects synchronization. Given that an IT-service can be considered as an element of the EA, SEAM might prove very useful for the IT-service management, in particular when more emphasize is put on business applications and processes. Further areas, where SEAM assumes a supportive (sometimes even servant) role, are (examples): risk management, strategies & goal management, project portfolio management, and governance. Regarded from a different angle, however, these areas also exert influence on the framework.

[2, 13] both provide an examples how EAM can serve other disciplines within the organization.

2.3 SOLUTION BUILDING BLOCK

A solution building block represents an area-overarching EA or EAM component placed on SEAM’s realization layer. SYRACOM proposes solution building blocks to clients who desire to pursue a goal we have been already achieved during several previous engagements. In this vein, solution building blocks comprise a method we successfully used during past projects, including the required organization, input/output deliverables, meaningful tool support, and recommended method carrier (existing vs. new process or project). In accordance to the internal and external influence factors of the enterprise as well as the current state of EAM within the organization, it also specifies the time and money needed to achieve the goal under consideration. Typical representatives we offer to our customers are, among others:

- EA: EA tool support including the elicitation of requirements, tool assessment, and assistance during deployment and introduction stage.
- EAM: EAM assessment for evaluating the methods, deliverables, strategy, tools, and organization established for the management of EAs.
3. Initiating SEAM – Framework Application

Based on a 7-step EAM method, this Chapter explains how to put SEAM successfully into action. In addition, information on governing the architectural work is provided.

3.1 7-Step Application Method

Whereas the framework’s conception areas, components, and their relations constitute the static part, subsequent 7-step application method reflects the dynamic facet of SEAM. To employ the EAM approach, at least one enterprise architect needs to carry out the framework’s EAM method named “Initiate SEAM” on an ongoing basis. The cyclic method pursues the generic and overarching goal “Improve the organization through EAM”. The last of its seven steps terminates with a set of EA/EAM projects and/or processes being either newly established or hooked up with current activities. The following table summarizes the properties of the method and gives a flavor how this framework realization layer component could be documented.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Initiate SEAM</td>
</tr>
<tr>
<td>Category</td>
<td>EAM method</td>
</tr>
<tr>
<td>Goal(s)</td>
<td>Improve the organization through EAM</td>
</tr>
<tr>
<td>Actor(s)</td>
<td>▪ Sponsor (consulted, informed)</td>
</tr>
<tr>
<td></td>
<td>▪ Beneficiary (consulted, informed)</td>
</tr>
<tr>
<td></td>
<td>▪ EA information provider (informed)</td>
</tr>
<tr>
<td></td>
<td>▪ EA transformer (informed)</td>
</tr>
<tr>
<td></td>
<td>▪ Enterprise architect (responsible, accountable)</td>
</tr>
<tr>
<td>Tool(s)</td>
<td>Microsoft Office</td>
</tr>
<tr>
<td>Input deliverable(s)</td>
<td>If existing, list of already established</td>
</tr>
<tr>
<td></td>
<td>▪ EA/EAM goals, KPIs, principles, and risks</td>
</tr>
<tr>
<td></td>
<td>▪ EA/EAM methods including organization, tools, deliverables</td>
</tr>
<tr>
<td>Step(s)</td>
<td>See below</td>
</tr>
<tr>
<td>Output deliverable(s)</td>
<td>New or updated list on</td>
</tr>
<tr>
<td></td>
<td>▪ EA/EAM goals, KPIs, principles, and risks</td>
</tr>
<tr>
<td></td>
<td>▪ EA/EAM methods including organization, tools, deliverables</td>
</tr>
<tr>
<td>Carrier</td>
<td>Dedicated process (centralized or decentralized)</td>
</tr>
<tr>
<td>Interdependencies</td>
<td>None</td>
</tr>
</tbody>
</table>

Table 1 - Properties of the EAM method „Initiate SEAM“
Generally, it is advisable to start small, thus focusing on a subset of achievable goals with only few stakeholders (cf. [13]). Once EAM has earned its merits, more opportunistic goals are tackled.

**Step 1: Determine and reconfirm sponsors and beneficiaries**
The enterprise architect seeks for beneficiaries and sponsors, thus actors who either state a new EA/EAM goal or support its accomplishment. The architect may apply different question techniques, as for instance suggested in TOGAF® [37, p. 253]. Upon their identification, new sponsors and beneficiaries are classified, e.g., according their organizational power, level of interest, influence on EAM. In addition, actors, whose goals are still pending, are recontacted and reclassified.

**Step 2: Define and revise goals, KPIs, and principles**
Through questioning the beneficiaries in workshops and/or interviews, the enterprise architect determines the current EA/EAM goals. To (re-)define a goal as much exhaustive and precise as possible, he/she may leverage existing deliverables crafted during previous EAM initiatives (e.g., outdated cluster map depicting the as-is application landscape). Sponsor, beneficiary, and the architect conjointly agree whether a new goal’s achievement should be measured using one or more KPIs. If so, the enterprise architect recommends concrete indicators taking also the required organizational effort into account (cf. method described in [24]). Furthermore, he/she (re-)examines the set of EA/EAM principles either already employed or newly brought up by sponsors, beneficiaries, or him/herself. Lastly, these actors discuss and resolve possible conflicts between goals and principles.

**Step 3: Determine and confirm influence factors**
In questionnaire-driven interview sessions, workshops and by analyzing secondary and company-owned resources (e.g., strategy document) the enterprise architect discovers and confirms internal and external influence factors for the EAM. Among others, he/she determines the organization’s degree of familiarization with the discipline and assesses its readiness to embark on an EAM initiative.

**Step 4: Evaluate alternative EA/EAM frameworks**
The enterprise architect checks whether other EA/EAM frameworks have been or are presently employed within the organization. Centering on methods and deliverables, he/she further validates, if these approaches can be partly or even fully leveraged for reaching the EA/EAM goals at hand. This goal-centric approach enables the enterprise architect to decide at an early stage already if SEAM is (still) the most suitable instrument or whether an alternative framework is a better fit (cf. [10]).

**Step 5: Select and readjust methods and tools**
In consideration of the EA/EAM goals and principles, the enterprise architect picks the most appropriate EA/EAM methods. For each new goal, he/she additionally determines the goal-dependent method’s carrier type and validates if existing processes and projects can be made use of. For any existing goal, the architect makes sure that the chosen method is still the optimal one. For both - selection and reexamination - he/she harnesses the experience made when addressing similar goals in the past. The architect further validates to which extent (dedicated EA/EAM) tools might facilitate and speed up the execution of a method. Where tools already support a method, he/she checks whether this is still in the interest of the organization. In case KPIs are employed, he augments each method with activities required to gather the indicator data.

**Step 6: Engage enterprise architects, EA information providers, and EA transformers**
Based on the selected EA methods, the enterprise architect appoints EA information providers for EA description and definition as well as EA transformers for EA transformation. In turn, he/she engages additional enterprise architects EAM methods have to be executed. After the appointed actors have been familiarized with the method, possible tool support, and expected deliverables, they officially acknowledge and commit to the underlying goals’ fulfillment. If required, the enterprise architect establishes additional committee entities supporting consulting, decision, and control tasks. Finally, any existing organizational structures being already in the process of executing EA/EAM
methods, are re-examined and adapted in line with the updated circumstances (e.g., revised goals, new influence factors, enhanced tool support).

**Step 7: Identify and confirm risks**

Based on the influence factors, the previously specified framework components, and alternative EA/EAM frameworks, the enterprise architect identifies all EA/EAM risks that potentially hamper the goals’ achievement. Each risk is classified, assessed (probability of occurrence, expected loss), and eventually communicated to the sponsors and beneficiaries [37, p. 313]. Together, these three actors define whether and how a particular EA/EAM risk should be addressed. Afterwards, the enterprise architect incorporates the required risk mitigation steps in all those methods that attempt to achieve the respective goal. Lastly, he/she reviews all existing risk measures and adjusts them when needed.

Incorporated in new/existing projects and/or processes, the (tool supported) EA/EAM methods are executed through the actors eventually leading to the generation of goal-specific deliverables. It goes without saying that this process is accompanied by careful quality assurance, targeted project management, as well as solid synchronization activities (also with non-EAM related disciplines). The method “Initiate SEAM” is reiterated once a certain amount of time has elapsed or a particular event occurred.

**3.2 SEAM AND GOVERNANCE**

Since governing the management of EAs (and less obvious, the optimization of the management) is a frequently discussed topic, we shortly point out the six distinct mechanisms SEAM comprises to successfully govern EAM:

1. KPIs for a continuous goal measurement
2. Principles to guide the process of goal achievement
3. Addressing of risks that put a successful goal achievement at stake
4. Organizational structure with clearly defined actors and role profiles
5. Methods with an unambiguous and traceable allocation of roles (RACI matrices)
6. Deliverables whose level of quality can be regulated by contracts

From our experience, the actual EA/EAM activities and its governance should be closely linked. Attaching a mechanism directly to one of the framework components helps to keep bureaucracy at a minimum while increasing its organizational acceptance.

Hanschke recommends KPIs, a sound organizational structure, as well as a clear data gathering method as fundamental elements of an EAM governance [13]. Greefhorst and Proper define principles as the primary enablers for effective architecture governance [12]. In addition to provide means for risk mitigation, TOGAF® suggests drawing up architecture contracts during the implementation governance phase of the architecture development method [37]. These contracts act as a joint agreement between development partners and sponsors on deliverables and quality.
4. Exemplifying SEAM – a case study from the financial industry

This Chapter describes an exemplary application of SEAM at a German financial group between 2008 and 2012. Attention is paid particularly to the findings and lessons-learned our client and we made when conjointly applying the framework. Due to confidentiality reasons, fictitious names are used.

4.1 Organization

Headquartered in Germany, the financial group employs more than 10,000 people with an annual profit of more than 100 Mio Euro in 2011. The organization operates subsidiaries in several European countries. Especially during the past decade the group acquired several financial institutions at regular intervals. These targets have been completely absorbed after a period of integration. In terms of the corporate strategy and goals, the company intends to focus on cross-selling and the improvement of its product offering reflected an organizational efficiency improvement program. The Chief Executive Officer (CEO) is well aware of the current uncertainty and volatility on the financial markets. He/she also acknowledges the new legal regulations his group has to follow in the very new future. As a result, the CEO pushes for a higher business and IT project throughput making his group more effective vis-à-vis constant changes.

4.2 Context of the case study

In 2008 financial group made its first move into the domain of EAM, at this time, mainly triggered in a bottom-up manner by the IT department which was facing the rising complexity caused by the acquisitions. Currently, 10 full-time employees are engaged in planning, defining, and transforming the company’s business and IT architectures. Each of these internal experts attended a TOGAF® course and most of them successfully passed the according certification exam. While much emphasize was put on the application and infrastructure architecture between 2008 and 2010, the EAM team shifted its focus to business and information matters during the past two years. Considering the EA tool landscape, the financial group runs alfabet planningIT for application and infrastructure, ARIS business designer for business and information, and CA Clarity for project (portfolio) management tasks. Regarding the relationship to non-EAM disciplines, much support is given to the management of business and IT project portfolios and single projects. Since 2008 SYRACOM has been being able to consult the financial group in its EAM initiative. Starting with the design and automated creation of process-application-support visualizations, SYRACOM has ever since applied SEAM as the main approach at the financial group. The framework in combination with the competences in the field make SYRACOM skillful and experienced architecture coaches, mentors, and project managers for questions and tasks centering on EAM.

4.3 Course of action

In January 2012 SYRACOM reiterated anew through the 7-step SEAM initiation method “Initiate SEAM” aiming at realigning the financial group’s EAM activities. Subsequently, we describe the outcome of each individual step.

Due to the fact, that the client was familiar with SEAM from previous iterations, SYRACOM quickly identified the Chief Operation Officer (COO) and the Chief Financial Officer (CFO) as general sponsors, the business, risk, and security managers as well as the IT-revision as potential beneficiaries for the new EAM efforts (step 1). The common goal was to gain transparency over the company-wide business objects and their need for protection on application level until the first quarter of 2013 (step 2). Based on the objects and an underlying model, contact persons had to be appointed afterwards who would assume responsibility for a specific set of entities. The main driver behind this EA
goal was the optimization of overall business and IT project execution time. According to sponsors and beneficiaries, transparency over the business objects would help to detect and resolve negative project interdependencies. Additionally, legal regulations urged the company to report on the business objects it creates, updates, and deletes on a regular basis (step 3). The client further decided to select the total number of classified business objects as a primary KPI.

As mentioned above, the internal enterprise architects were educated in TOGAF®. However, the EA framework was deemed inappropriate for achieving the goal mainly due to its text-heaviness, consistency errors (e.g., architecture principles are classified as IT principles), and lacking business object model support. Instead, domain-specific business object models like the German Versicherungsanwendungsarchitektur [11] and the Banking Industry Architecture Network [5] were taken into consideration given their focus on object models (step 4).

Since SYRACOM had never consulted a company in this matter, no turn-key SEAM solution building block was at disposal. Instead, SYRACOM devised a completely new method entitled “Design of the business object architecture” (step 5). Its vehicle was a dedicated project which running from February 2012 till February 2013. Throughout the period of implementation, no EA or EAM principle restricted the consultants in their actions. In accordance with the cross-organizational character of the task, several business departments assumed the role of the EA information provider and quality assurance. For classifying the business objects’ need for protection, employees of the Information-Security Management joined SYRACOM’s efforts. Given that the EAM initiative was about transparency, no EA transformers were required. Lastly, several internal business architects supported SYRACOM in the role of enterprise architects (step 6). In terms of risks, the financial group determined the tight timetable in combination with its lack of experience with business object models. Besides the incompleteness of the resulting business object model, the group feared that SEAM would turn out to be too abstract and methodical for being actually applicable in practice (step 7).

As a first step of the method, SYRACOM devised a foundation-model for the business objects. In order to avoid starting from scratch and lose precious time, the consultants leveraged an existing meta-model placed on the realization layer of SEAM and enriched this deliverable

![Diagram of Business Object Meta-Model](FIGURE 6 - Business object meta-model)
with concepts found in the studied business object reference models. The resulting artifact is depicted in Figure 6, again using the ArchiMate modeling language (blue = active entity; yellow = behavior; green = passive entity). One subject (e.g., financial service provider) assumes a role (e.g., buyer, seller) that contributes to a process (e.g., sell real estate) being triggered by an event (e.g., certain amount of time elapsed). Furthermore, a role has a legal relationship (e.g., purchase contract) to an object (e.g., real estate). Both, legal relationship and process are described by an information carrier (e.g., document). With this model at hand, the subject matter experts of the different business departments managed to identify and classify their main business objects. In total, more than 1,700 objects were found in the course of 2012. After a quality assurance step, Information-Security Management was in charge of defining the need for protection. To increase the overall efficiency and reduce the risk for human errors, gathered data was kept in Microsoft Access database being established as an additional organization-specific EA tool. Drawing on this repository, Microsoft Visio was used to create beneficiary-specific visualizations. On the one hand, these deliverables helped to respond to the pressing legal requirements. On the other hand, they significantly contributed to the reduction of project execution time.

4.4 RESULTS

Against the overall expectations of the sponsors and beneficiaries, the goal of gaining transparency over the business objects as well as their need for protection was achieved in time and in budget. In particular, the client appreciated the foundation-model for business objects SEAM provided on its realization layer. According to the customer, the pre-defined model helped to save a significant amount of time and money. As a positive side effect, the project success led to our subsequent commission in 2013. Among others, the tasks include the definition of business object model usage scenarios, enhancement of the approach with project-(in) dependent specializations and instantiations, and integration of organizational responsibilities.

The applied method was captured as a solution building block, newly created deliverables like the basic business object meta-model or the method were generalized and added to the content layer of SEAM. As a major advantage, the industry partner emphasized the clear, consistent, and simple structure of SEAM, helping to tackle his problem in a comprehensible, traceable, and ordered manner.
5. Summary

This document presented SEAM, a goal-oriented EAM framework SYRACOM has been successfully using in an industrial context since 2003. Opposed to alternative approaches, SEAM provides a minimal, consistent, and easy-to-grasp framework foundation consisting of conception areas and core components. Upon this baseline, the categorization and realization layer are placed allowing the adaptation of the artifact to the individual organizational needs. In addition to this static structure, SEAM includes a 7-step initiation method SYRACOM applies when embarking on a clients’ EAM engagement. In contrast to alternative frameworks, SEAM explicitly discerns EAM and the adjustment of EAM. In our experience, this differentiation is important and oftentimes overseen. To demonstrate the framework’s applicability while identifying future points of improvement, the document briefly depicted a case study describing how SEAM was put into practice at a German financial group.

If you have questions or any comments with regards to SEAM, please send a note to SEAM@syracom.de. We highly appreciate your Feedback.
6. References


38. The Open Group (Ed.): ArchiMate 2.0 Specification. Berkshire/UK (2011)


7. **Nomenclature**

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADM</td>
<td>Architecture Development Method</td>
</tr>
<tr>
<td>BEAMS</td>
<td>Building blocks for Enterprise Architecture Management Solutions</td>
</tr>
<tr>
<td>CEO</td>
<td>Chief Executive Officer</td>
</tr>
<tr>
<td>CFO</td>
<td>Chief Financial Officer</td>
</tr>
<tr>
<td>COBIT</td>
<td>Control Objectives for Information and Related Technology</td>
</tr>
<tr>
<td>COO</td>
<td>Chief Operation Officer</td>
</tr>
<tr>
<td>EA</td>
<td>Enterprise Architecture</td>
</tr>
<tr>
<td>EAM</td>
<td>Enterprise Architecture Management</td>
</tr>
<tr>
<td>IAF</td>
<td>Integrated Architecture Framework</td>
</tr>
<tr>
<td>KPI</td>
<td>Key Performance Indicator</td>
</tr>
<tr>
<td>PERA</td>
<td>Purdue Enterprise Reference Architecture</td>
</tr>
<tr>
<td>RACI</td>
<td>Responsible Accountable Consulted Informed</td>
</tr>
<tr>
<td>SEAM</td>
<td>SYRACOM Enterprise Architecture Management</td>
</tr>
<tr>
<td>TOGAF</td>
<td>The Open Group Architecture Framework</td>
</tr>
<tr>
<td>UML</td>
<td>Unified Modeling Language</td>
</tr>
</tbody>
</table>
ABOUT SYRACOM

Founded in 1998, the SYRACOM group is an independent business and IT consultancy which focuses on the design and implementation of efficient and sustainable business processes. Based on their profound industry knowledge, SYRACOM consultants constantly bridge the gap between business and IT. SYRACOM’s strong commitment to follow a holistic approach concerns strategy enactment, organization and process optimization, as well as the implementation of customized business and IT solutions. The company’s international clients are active in diverse industries, e.g., banking and insurance, telecommunication, automotive, as well as transport and logistics. Besides Enterprise Architecture Management (EAM), further fields of competences of SYRACOM are: business engineering, enterprise performance management, project management, as well as application management, development, and integration.

Copyright
Copyright © 2013 SYRACOM AG, Germany. All rights reserved.

No part of this publication may be reproduced, stored, archived, or transmitted in any form by any means (electronical, mechanical, by photocopying, or otherwise) without the prior printed permission of the publisher, SYRACOM. The information contained and opinions expressed herein may be changed without prior notice.

Trademarks
All trademarks or registered trademarks are the property of their respective owners.