A New Approach for Collaborative Enterprise Architecture Development

Vadim Agievich[1,3], Victor Taratukhin[1,2], Jörg Becker[2], Rinat Gimranov[3]
[1] Higher School of Economy, National Research University, Moscow, Russia
[2] ERCIS – Headquarters, Münster, Germany
[3] Surgutneftegas, OJSC, Surgut, Russia
agievich_va@surgutneftegas.ru

***Abstract*—One of the challenges of Enterprise Architecture (EA) in large companies is organizing processes of baseline EA development and change management under conditions of steady change that are introduced by implemented and changed IT solutions. The research paper describes a new approach of developing and keeping relevant baseline EA description together with IT project teams using Solution Architecture models. The approach is in the process of implementation in one of the leading Russian oil and gas companies Surgutneftegas.**

Keywords-Enterprise architecture; Solution Architecture; architecture framework; BPM; large company; metamodel; IT project;

I. INTRODUCTION

Today's CEOs know that the effective management and exploitation of information through IT is a key factor to business success, and an indispensable means to achieving competitive advantage. Enterprise Architecture addresses this need, by providing a strategic context for the evolution of the IT system in response to the constantly changing needs of the business environment [1]. Enterprise Architecture (as J.Schekkerman defines in [3]) is a complete expression of the enterprise; a master plan which "acts as a collaboration force" between aspects of business planning such as goals, visions, strategies and governance principles; aspects of business operations such as business terms, organization structures, processes and data; aspects of automation such as information systems and databases; and the enabling technological infrastructure such as computers, operating systems and networks [3].

At the moment a number of different Enterprise Architecture frameworks and tools exist but challenges still remain both from framework and organizational perspectives as H.Shah and M.Kourdi describe in [10]. Surgutneftegas' experience shows that creating of a comprehensive and integrated Enterprise Architecture description is a challenging task both in a technical an organizational sense. EA team should collect the information needed for "As-Is" EA modeling and structure it according to the chosen EA framework within an EA repository. But in large enterprises changes in IT environment are never ending. It's a common place situation when several IT projects are running simultaneously conducted by different organizational structures. A problem rises before an EA team: How to keep up-to-date EA repository in such circumstances? Enterprise Architects (describing the enterprise part by part) may miss changes done by an IT project team. All the changes that are made in IT projects should be tracked and every Solution Architecture should be taken in account. Otherwise models will lag behind the real world [11], [12].

The problem depicted above is related to a baseline EA and subsequent EA maintenance. Doing this work by several people of EA team is like washing windows of a skyscraper. When the last floor is finished the first floor is already dirty. In order to solve the problem of obtaining a holistic and relevant reflection of an enterprise an EA team should work in collaboration with IT project teams (that create and implement solutions) and IT solutions exploitation teams (that make changes to the solutions). In order to do this all these people need a collaborative approach based on a collaboration platform. Such an approach does not exist because Enterprise Architecture and Solution Architecture (SA) are often seen as different disciplines and different practices. As M.Ricca related in [9], of the two, EA is sometimes regarded as the rich and decadent relative while SA is the honest, reliable, and hard-working one.

Authors' original idea is to use Solution Architecture models as a source of information for a baseline Enterprise Architecture. In this case collaboration between actors mentioned above may be established on the basis of unified building blocks (object catalogs).

The paper describes the approach for collaborative development of Enterprise Architecture using Solution Architecture models.

II. ENTERPRISE ARCHITECTURE AND SOLUTION ARCHITECTURE

A. Enterprise Architecture

Enterprise Architecture is often referred as a blueprint for how an organization achieves the current and future business objectives using IT. It examines the key business, information, application, and technology strategies and their impact on business functions. It provides the framework for planning and implementing a rich, standards-based, digital information infrastructure with well-integrated services and activities [4]. It's a strategic information asset base, which defines the mission, the information necessary to perform the mission, the technology, and the transformational processes for implementing new technologies in response to the changing

978-1-4673-1773-3/12/$31.00 ©2013 IEEE

mission needs. An Enterprise Architecture includes a baseline ("As-Is") Enterprise Architecture, target ("To-Be") Enterprise Architecture, and a transition plan [3].

This research paper considers the issue of creating an integrated, consistent and relevant baseline Enterprise Architecture in conditions that were indicated before.

B. Enterprise Architecture framework

Enterprise Architecture frameworks define the scope of the Enterprise Architecture and decompose various elements of the architecture onto structured levels and elements [5]. These elements and their relations are described in a metamodel that is the core of most Enterprise Architecture frameworks. According to TOGAF 9.1 [1] a metamodel is a model that describes how and with what the architecture will be described in a structured way.

Fig. 1 presents the metamodel of The Open Group Architecture Framework (TOGAF) that is one of the most well-known and highly-adopted EA frameworks. The role of TOGAF is to provide an open standard for architecture that is applicable in many scenarios and situations. In order to meet this vision, it is necessary to provide a fully-featured Enterprise Architecture metamodel for content and also to provide the ability to avoid carrying out unnecessary activities by supporting tailoring [1].

The Open Group Architecture Framework is used in Surgutneftegas for EA development.

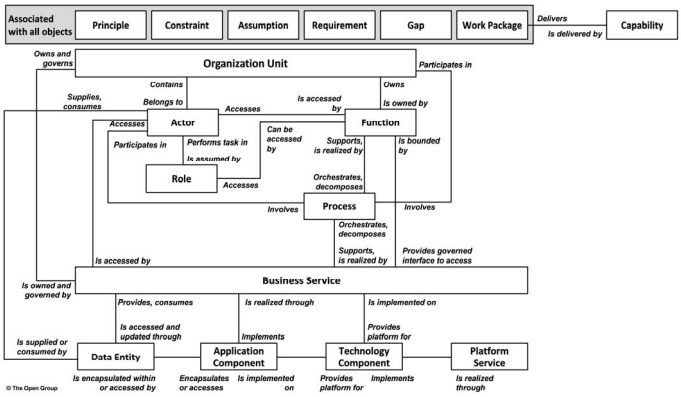

Figure 1. TOGAF core content metamodel [1]

Models in an EA repository should be organized in a way that provides a complete set of objects and relationships between them according to the metamodel definition.

TOGAF also includes an Architecture Development Method (ADM) that describes a method for developing and managing the lifecycle of an EA [1].

C. Solution Architecture

According to TOGAF 9.1 [1] Solution Architecture is a description of a discrete and focused business operation or activity and how IS/IT supports that operation. A Solution Architecture typically applies to a single project or project release, assisting in the translation of requirements into a solution vision, high-level business and/or IT system specifications, and a portfolio of implementation tasks.

From this point of view IT solutions may be treated as a system that comprises business (e.g. processes that are to be automated), information systems (that are to be developed or bought and then implemented) and information about the solution implementation. Engineering such a system according to business needs is one of the main goals of an IT project team. The processes of such activities are regulated by ISO/IEC 15288 [7]. As described in [7] the Architectural Design Process can be used to transform the defined set of technical requirements into an acceptable architectural design solution that fulfills the technical requirements for the system being engineered. The architectural design solution should be documented in a technical data package or database that includes a set of architectural design solution specifications (architecture descriptions) and other configuration descriptions [7].

D. Models for Solution Architecture

Requirements for architecture descriptions are specified in ISO/IEC/IEEE 42010 [8]: Architecture descriptions are used by the parties that create, utilize and manage modern systems to improve communication and co-operation; enabling them to work in an integrated, coherent fashion. Architecture frameworks and architecture description languages are created as assets that codify the conventions and common practices of architecting and the description of architectures within different communities and domains of application. The standard [8] in chapter 4.5 "Architecture frameworks and architecture description languages" also provides examples of architecture frameworks that can be used for architecture descriptions: Zachman's information systems architecture framework, UK Ministry of Defense Architecture Framework, The Open Group's Architecture Framework (TOGAF) and others.

In order to stay compatible with EA models SA models may and should be based on the same architecture framework.

Architectural design processes include logical architecture (or high-level) design and physical architecture design. The former is closer to EA development in the sense of granularity. The standard [7] indicates (but doesn't limit) types of models that solution architects may use for logical design phase. Among them:

- A functional flow block diagram reflecting the decomposition of major functions into their sub-functions.
- A data flow diagram that decomposes functions while explicitly showing the data needed for each function.
- A data structure with corresponding functions and processing flows related to the data and associated with assigned technical requirements.
- The states and modes of the system.

Similar models are used by Enterprise Architects to describe Business, Data and Application architectures. Among them (according to TOGAF):

- Functional Decomposition View.

- Data Dissemination View.
- Data Entity / Business Function Matrix.
- Business Footprint View.
- Process / System Realization View.

Similarity of these models shows that logical high-level architecture models for solutions contain the same data (the same building blocks and their connections) as EA models do. However SA models are more much narrower in scope. They describe only the automated part of an enterprise. If we look again on TOGAF metamodel (Fig. 1), this part is presented by elements like Data Entity, Application Component, Technology Component, Platform Service, and their connections to other components (Data, Application and Technology domains). We can obtain corresponding data for the whole enterprise from models that are developed to describe all existing IT solutions of the enterprise. What is left? Business domain.

Underling the above statements the authors are coming to the conclusion that compatible EA and SA models could be obtained using an Enterprise Architecture framework. Moreover, the data that are necessary for EA models may be obtained from SA models.

E. Connecting SA to EA

Enterprise modeling should focus on bringing together already existing techniques and integrating these at the appropriate level of abstraction [2].

Enterprise Architecture and Solution Architecture have to work together to effectively build an architectural landscape and reap its benefits to the maximum extent with minimum waste. In fact, EA and SA (which includes network, software, security architecture, and so on) are facets of the same business function: architecture [9]. The problem is that these functions act in different scopes and activities. EA comprises the whole enterprise (or a part of it according to the EA scope); SA acts on the level of automating particular processes of the enterprise and describes mostly information systems and data (lower part of the metamodel shown at Fig. 1). EA is broader in description of business both horizontally and vertically. A collection of SA models cannot call itself an "EA". Architects need a holistic organizational structure description and information about business processes that are both automated and not automated. Hence if we have that information in the repository and make SA and EA models using a unified toolset based on unified building blocks (object catalogs) we may present a baseline EA as a projection of initial Business Architecture (that is made on the phase "B. Business Architecture" of the TOGAF ADM) and all Solution Architectures of the company (see Fig. 2).

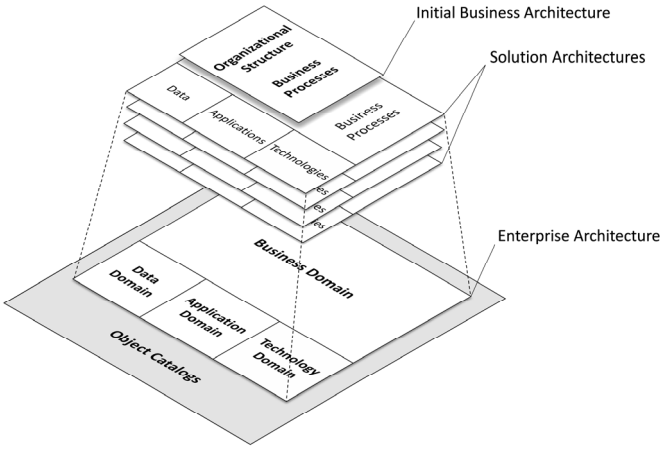

Figure 2. Enterprise Architecture as a projection of Solution Architectures

III. ARCHITECTURE DEVELOPMENT METHOD MODIFICATION

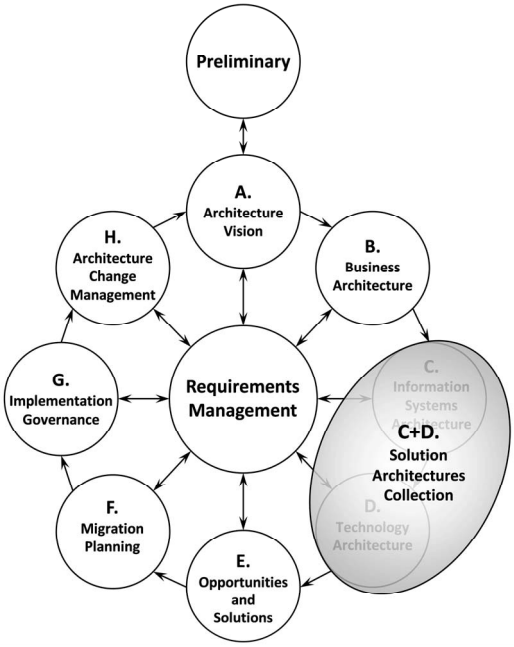

Figure 3. Modified TOGAF Architecture Development Method

Analyzing connections of SA and EA described above authors also came to a conclusion that TOGAF ADM may be tailored for the case of a baseline EA development to use in the collaborative approach. The rational is the following.

As TOGAF [1] describes, the objective of the phase C is to develop Data Architecture and Application Systems Architecture. In order to develop the Data Architecture architects should define the major types and sources of data necessary to support the business. It is important to note that this effort is not concerned with database design. The goal is to define the data entities relevant to the enterprise, not to design logical or physical storage systems [1]. In order to develop the Data Architecture architects should define the major kinds of

application system necessary to process the data and support the business. It is important to note that this effort is not concerned with applications systems design. The goal is to define what kinds of application systems are relevant to the enterprise, and what those applications need to do in order to manage data and to present information to the human and computer actors in the enterprise [1].

The Technology Architecture (phase D) seeks to map application components defined in the Application Architecture phase into a set of technology components, which represent software and hardware components, available from the market or configured within the organization into technology platforms [1].

As it was shown before the conjunction of all existing Solution Architectures of the company delivers the same information that is collected on phases “C. Information Systems Architecture” and “D. Technology Architecture”. Consequently instead of these phases the authors propose the phase “C+D. Solution Architectures Collection” (fig. 3) for. However the corresponding process will be completely different. The work should be organized among those people who maintain corresponding solutions. It may involve some kind of solution redocumentation using a unified toolset and building blocks (object catalogs). In organizational terms the work will be distributed between architects and maintenance teams. Such an approach may reduce time and effort.

The process for phase “H. Architecture Change Management” may also be modified using the same approach as for the C+D phase.

IV. Requirements for Organizing of IT Projects

In order to realize the collaborative approach for EA development and obtain compatible data for EA repository from IT project teams and maintenance teams in a large company, corresponding activities should be strictly specified by official company standards and policies that should prescribe:

- Solution Architecture design as an obligatory phase of any IT project.
- Modeling of Solution Architecture using a unified design toolset (in the “development” area – models are not yet in the Baseline Enterprise Architecture).
- The procedure of Solution Architecture verification and approval.
- Checking (“Quality Gate”) the approved Solution Architecture before the development phase and during the implementation phase.
- Verification of conformance of the Solution Architecture models to the implemented solution.
- Transferring models of the Solution Architecture within the unified design toolset from the “development” area to the “Baseline Enterprise Architecture” area.

Respective processes introduced to IT implementation and change management practices will provide the relevance of baseline Enterprise Architecture.

V. Appropriate Organizations for the Method

In order to leverage the collaborative approach described in this article it is important for an organization to have the ability to dictate unified rules for conducting and documenting IT projects as well as using unified tools for Solution Architecture modeling. Besides that the collaborative approach should be based on the traditional EA approach.

According to Gartner, in a traditional approach the EA team engages the organization to facilitate the EA process, focused on prescriptive content that serves to guide project decision making consistent with the "master plan" embodied in the architecture. Most of the standard industry frameworks and processes support a traditional approach to EA which delivers very specific directives to projects on how to develop solutions that will meet business requirements, while reducing complexity in technology, information and business processes. This approach tends to work well in organizations where decision making is largely centralized and are relatively stable in terms of the pace of change [6].

VI. Requirements for a Toolset

Usage of the collaborative approach described in this paper is impossible without an integrated toolset. Besides all the capabilities that Enterprise Architecture activities demand the toolset should satisfy some requirements.

In order to preserve data consistency, models should be put in a unified database. The SA modeling tool must be integrated with (or be the same as) the EA modeling tool. The SA modeling tool should have “development” area for those models that are not yet approved and corresponding IT solutions are not implemented. Otherwise these incomplete and/or unapproved models will create a chaos in the baseline EA. In order to support approval and “Quality Gates” processes the tool should have the ability of maintaining statuses for models (e.g. “Under construction”, “Approved”, and “Implemented”).

The other obligatory condition that will ensure consistency is usage of the same building blocks for EA and SA. Modelers must use unified object catalogs (e.g. data entities catalog, applications catalog, functional decomposition diagram). When creating an SA model a modeler should take object instances only from corresponding catalogs. If a catalog does not contain an instance needed for a model, the instance should be added by a dedicated person through a request from the modeler. Repository governance and maintenance processes should also be established.

VII. Conclusion

The authors propose a new approach for collaborative development of a baseline Enterprise Architecture using Solution Architecture models.

SA models may serve as a source of information for an EA repository and help to keep the baseline EA relevant if the following conditions are satisfied:

- A unified integrated toolset and unified object catalogs should be used by both Enterprise Architects and Solution Architects.
- The toolset should comply with the special requirements described above.
- All IT solutions should be documented using the toolset.
- IT projects should be guided through the use of "Quality Gates" (e.g. a high-level design review) for ensuring the quality of Solution Architecture descriptions.

The approach described above is in the process of implementation at one of the leading oil and gas companies in Russia "Surgutneftegas". The IT management of Surgutneftegas demands that IT project teams should take part in EA activities and use the EA to guide and support their projects. EA should provide a collaborative platform that will help create integrated and sustainable solutions.

In this research paper only initial approaches of the method were outlined. Many things should be done to put it in practice.

Firstly, a set of models for modeling SA and EA in an integrated fashion should be defined. Secondly, a tool should be found on the market that may be customized according to the requirements. If there is no such tool than a question of software development rises. The authors believe that without the tool the method will not be affective. Finally it is necessary to have policies and regulations for practical application of the method. These are directions for further research.

REFERENCES

[1] TOGAF 9.1, http://pubs.opengroup.org/architecture/togaf9-doc/arch/

[2] Lankhorst, M.M.: Enterprise architecture modelling—the issue of integration, Advanced Engineering Informatics 18, 205–216, 2004

[3] Schekkerman, J.: How to Manage the Enterprise Architecture Practice. Trafford Publishing (2009).

[4] Dahalin, Z.M., Razak, R.A., Ibrahim, H, Yusop N.I., Kasiran, M.K.: An Enterprise Architecture Methodology for Business-IT Alignment - Adopter and Developer Perspectives. CIBIMA 2010, Article ID 222028, 2010

[5] Shah H., Kourdi M.: Enterprise Architecture Frameworks, IEEE IT Professional, Vol 9, No 5, 2007.

[6] Tudor, B., Goasduff, L.: Gartner Predicts 95 Per Cent of Organisations Will Support Multiple Approaches to Enterprise Architecture by 2015, http://www.gartner.com/it/page.jsp?id=1358913

[7] ISO/IEC 15288 Systems and software engineering – System life cycle processes.

[8] ISO/IEC/IEEE 42010:2011 Systems and software engineering – Architecture description.

[9] Ricca, M.: Bridging the Gap Between Enterprise Architecture and Solution Architecture for Maximum Benefit, http://www.oracle.com/technetwork/articles/entarch/ricca-bridging-ea-sa-457138.html

[10] Shah, H., Kourdi, M.: Frameworks for Enterprise Architecture, IT Pro September - October 2007.

[11] Becker, J., Kugeler, M., Rosemann, M. : Process Management: A Guide for the Design of Business Processes, Springer (2003).

[12] Becker, J., Delfmann, P.: Reference Modeling: Efficient Information Systems Design Through Reuse of Information Models, Physica Verlag (2007).