Utilizing Enterprise Architecture for More Effective Requirements Engineering

Ömer Korkut, Devrim Barış Acar, Erhan Keleş, Oral Gürel

STM A.Ş. Consultancy Services Department,
Ankara Teknoloji Geliştirme Bölgesi, Cyberplaza C Blok 2.Kat Ankara, Turkey

Abstract-Requirements Engineering (RE) plays a vital role in successful software-intensive systems development. Generally, requirements have a tight relationship with organizational goals and constraints which can be contributed by an Enterprise Architecture (EA). This paper discusses the link between RE activities and EA paradigm. Our experiences in developing requirements for a series of large-scale software integration projects in both public and private sectors inspired the content of the paper.

I. Introduction

Developing software-intensive systems has remained a challenging activity in spite of remarkable progress in computing domain. Furthermore, rapidly evolving technology and sophisticated business requirements are placing ever-increasing pressure on software development process. Especially large-scale software projects continue to fail at an unacceptable rate. One of the main reasons for these failures is the inability of conveying requirements to intended software product. Therefore, the following statement emphasizing the utmost importance of an effective RE still preserves its topicality. "The hardest part of building a software system is deciding precisely what to build. No other single part of the conceptual work is as difficult as establishing the detailed technical requirements, including all of the interfaces to people, to machines, and to other software systems. No other part of the work so cripples the resulting system if done wrong. No other part is more difficult to rectify later" [1].

RE process aims to determine a baseline for requirements correctly first and then manage it throughout the software development phase. In order not to fail at the first step, it is crucial to detect the right sources to derive requirements. Ideally, both strategic goals and tactical demands are the motivation behind requirements. The Open Group Architecture Framework (TOGAF™) puts goals and requirements in the heart of the architecture development process. Business goals are identified explicitly in EA which is a conceptual blueprint that defines the structure and operation of an organization [2]. For the last two decades, EA has evolved as a practice providing pragmatic artifacts such as requirements, specifications, guiding principles, and conceptual models that describe the next major stage of evolution of an organization, often called the "future state" [3]. With its dynamic nature, EA may become not only a valuable resource to refer during requirements development phase, but also an important reference usable throughout the entire RE process.

In this paper, we present and argue our premise that EA should play a more significant role in RE process. We specifically emphasize that lack of a solid EA in a large organization may cause severe consequences regarding requirements development. The remainder of this paper is organized as follows: Section II provides a general overview of RE process focusing on requirements development. Section III discusses EA paradigm and accentuates its benefits for the employing enterprise. Section IV conveys our findings and recommendations in RE gained in the course of a variety of complex software integration projects realized for not only governmental but also private organizations in different domains such as finance, healthcare, construction, energy and technology. Finally, Section V concludes with a brief summary of the subject matter and our suggestions for future work.

II. Requirements Engineering

RE is the set of activities on identifying and communicating the purpose of a software system and the cases in which it will be used. RE connects real-world needs of stakeholders (users, customers and other participants) to a software based system's capabilities. Basically, RE is made up of two major processes: requirements development and requirements management [4].

A. Requirements Development

Requirements Development phase is constructed from elicitation, analysis, specification and validation steps. At the end of the development phase a series of baseline documents such as Software Requirements Specification (SRS), User Requirements Specification (URS) or Business Requirements Document (BRD) will be generated. Requirements development steps may be executed iteratively if a defect or gap is detected prior to the baseline [5].

- **Requirements Elicitation**

Elicitation step is the process of determining the needs of all stakeholders. Therefore, the first activity of elicitation step is defining the stakeholders. The elicitation step aims to collect requirements by approaching from different directions such as business requirements, customer requirements, user requirements, constraints, security requirements, information requirements, standards etc. The specification of software requirements starts with observing, interviewing people or investigating any documentation owned by the stakeholder that defines their processes. The resultant product of elicitation step is not a well-formed specification document. It is rather a long list of items that defines what the stakeholders plan to perform with the system.

- **Requirements Analysis**

Requirements Analysis is the process of detailing and prioritizing the requirements, organizing the requirements in a hierarchical manner and evaluating the feasibility of the requirements by building proof of concept products or prototypes. One of the main goals of analysis step is verifying that there is no gap in the requirements.

- **Requirements Specification**

Specification step is the period in which all the requirements are documented. The crucial point of specification is that, the generated document will be the starting point of traceability. Therefore, each requirement written in the specification should be atomic, comprehensible, and should not conflict with any other.

- **Requirements Validation**

Validation is the last step of requirements development phase. In this step, documented requirements are reviewed by the stakeholders to confirm that they satisfy customer needs. At the end of the validation step a baseline requirements document is generated.

B. Requirements Management

In the course of project lifecycle, requirements will continue to change, so the requirements document. Requirements Management encompasses activities of managing and tracking changes in requirements. Managing changes involves, creating a change management process (and a control board), performing

impact analysis on changes and generating new versions of requirements document. Tracking process is saving the history of each requirement, tracking the status and establishing the traceability matrix.

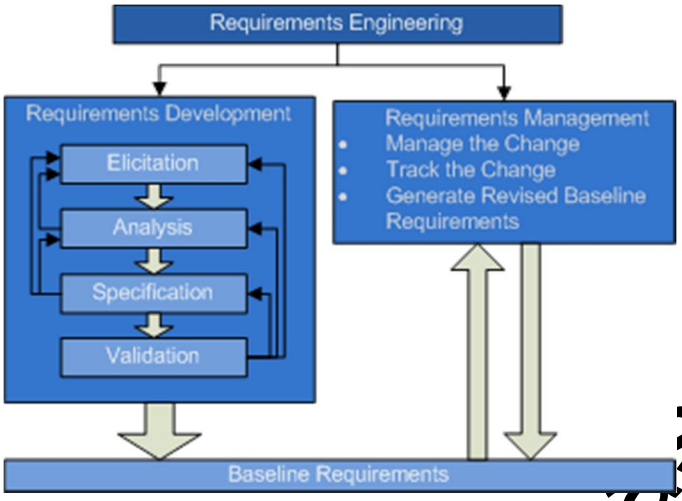

Fig. 1 Phases of Requirements Engineering

III. Enterprise Architecture

A project is initially defined by its scope and vision that describes the high-level business requirements, stakeholders and system boundaries in general. The scope and vision, thus, constrains the set of requirements described by the project. However, for large scale enterprises, we believe that defining the project's scope and vision is not enough for successful RE activities because they don't refer to organizational context and rationale of business requirements which EA provides.

A. Definition

Reference [6] describes EA as a high-level representation of the enterprise, used for managing the relation between business and IT. EA is also defined as "strategic approach which takes a systems perspective, viewing the entire enterprise as a holistic system" [7]. According to [8], EA provides:

- Strategic context for the evolution of IT System by describing the organizational context of the business requirements.
- General guidelines for design.

B. Enterprise Frameworks

In an architectural approach, a framework is needed for the communication of the decisions, requirements, constraints, enablers and feedback among the stakeholders [9]. EA frameworks like TOGAF™ and Zachman [10] provide guidance on how to conduct and structure the artifacts of an EA. In general, these frameworks define layered architectures. For example TOGAF™ presents four architecture domains [8]:

- Business architecture defines the business strategy, governance, organization and key business processes.
- Data architecture describes the structure of an organization's logical and physical data assets and data management resources.
- Applications architecture provides a blueprint for the individual applications to be deployed, their interactions, and their relationships to the core business processes of the organization.

- Technology architecture includes IT infrastructure, middleware, networks, communications, processing, standards, etc.

With respect to RE, the artifacts of these four domains generally help providing the initial set of requirements and a general set of constraints and guidelines for the further requirements development. This encourages requirements reuse as stated in [11] [12]. For example, as demonstrated in [9], the enterprise wide security requirements once defined are used by different systems.

C. Enterprise Architecture Benefits

The benefits of following an EA are not limited to requirements reuse. One of the main advantages of establishing an EA is helping enterprises to align business and IT processes. As McKeen and Smith [13] argues, the above mentioned strategic alignment is possible only when an organization's goals, activities and the information systems that support them remain in harmony.

Several researches have been conducted to assess the further benefits of EA. According to [14], the primary goals of EA with respect to practitioners are:

- To get the holistic view on the IT landscape as well as supported business processes by creating transparency,
- To manage complexity by using the holistic view and then consolidating IT applications or standardizing processes,
- To align business and IT.

This holistic approach provides the enterprise architects the opportunity to understand their as-is organization. Using this information, they can identify the bottlenecks, unnecessary processes or applications and points of further innovation, by which they form the shape of the future to-be architecture. This to-be architecture, described and used as a reference is generally dynamic in nature, and makes use of requirement engineering activities as pictured in Fig.2 [8]. For projects, by the help of requirements engineering activities, the reference architecture is adapted and specialized, and where appropriate these adaptations are generalized in reference architecture for future re-use. This process is named as architecture development cycle.

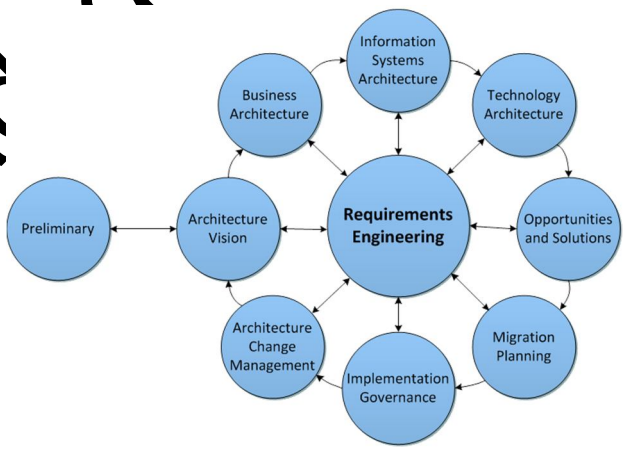

Fig. 2 Architecture Development Cycle

Taking these benefits into consideration, we think that the lack of a solid EA in a large organization results in poor coordination between departments, cost increases, requirement conflicts across projects and other undesired consequences.

IV. Findings and Recommendations

We gained experience from various software projects realized for different organizations of which the majority did not have an EA. In this section, we will share our findings and recommendations regarding to the correlation between effective RE activities and a solid EA.

A. Better Scope Management

High level organizational goals and strategies are either stated or referred in EA. While working with an organization not having an EA on a software project, we went through a serious scope creep problem. The reason for that was all stakeholders were trying to realize their own goals since the organizational ones were not clearly identified and communicated. One of our major lessons learned after this particular project was that an organization should have a solid EA to achieve projects aligned with its goals and strategies.

B. Faster Requirements Elicitation

Considering that EA contains organizational goals, guidelines, principles, policies, capabilities and constraints; its relevant parts can provide the initial set of requirements that can be input to elicitation step [12].

We discovered that some requirements elicited by inspecting EA were mature enough to be directly used in requirements analysis step while some of them provided guidelines but needed to be further detailed. For the latter, we recommend additionally performing traditional elicitation techniques such as devising surveys and questionnaires, organizing workshops with stakeholders, observing in force work processes and building use-cases. Even for these cases, since there was a starting point coming from EA, we experienced that the overall elicitation process was less time consuming.

C. More Organizational Requirements

Utilizing EA also assures that the determined requirements are more organizational and less personalized. In one specific software project we were involved, because of the lack of an EA, we only used traditional techniques for requirements elicitation. During this step, we witnessed that interfering stakeholder interests and frequent stakeholder turnover resulted in numerous repetitions. Therefore, in order to minimize the risk that may be caused by stakeholder turnover and conflicting benefits, we recommend referring to EA in elicitation process, if it is available.

D. Requirements Re-use

Our team had the opportunity to take part in requirements development of two different projects for the same organization. Some of the functional and most of the non-functional requirements dictated by EA in the first project were used with minimal or no modification in the second project. Hence, we deduced that EA driven elicitation approach enables re-use of once determined requirements as a source for other projects in the same organization.

E. Supporting Requirements Analysis

EA provides information about the business, application, data and technology architecture. Having this knowledge in hand eases breaking down the elicited requirements into more detailed and technical ones thus reducing the difficulty and complexity of the analysis step. In one of our projects, we elicited a user requirement for redundancy in a multi-located distributed system. In analysis step, we examined the

organization's well-defined technology and data architecture and in result clearly identified the desired requirements to achieve high availability.

F. Requirements Specification Structuring

In our works we observed that using EA provided architectural layers such as business, data, application and technology facilitate classification of requirements in the specification step. When there is an EA available we utilized architectural layers to categorize and specify analyzed requirements recorded in relevant documents. This methodology improved the structure of the specification documents, thus ensuring stakeholders to have a better grasp of the project.

G. Validation and Stakeholder Management

EA driven requirements development reduces time spent to acquire a formal validation from stakeholders by eliminating the expected gaps and defects in former steps through double checking user requirements with written organizational goals and reference EA. In integration projects for organizations without EA we had difficulties in successfully completing the validation step due to the problem of a large variety of stakeholders coming from different organizational cultures. In these cases, even after exhausting communication sessions with stakeholders in both elicitation and analysis steps, it was almost impossible to reach a consensus among stakeholders in the first iteration of validation step. The main reason of this failure was the conflicting interests and undetermined stakeholder hierarchy.

V. Conclusions

With all the studies and widely accepted processes about RE, the ability to effectively derive, trace and re-use software requirements has still some room for improvement with regard to large-scale complex systems. Enterprise level architectural approach has recently emerged as a candidate concept that can have positive impact on RE processes. In our opinion, this approach should be used to complement current well-defined and widely used RE processes and not to replace any of them.

In this paper, we investigated and listed some of the relationships between EA and RE with reference to our experience in various software projects. Our conclusion is that, an established EA improves the requirements gathering, analysis, specification and validation steps of requirements development.

In an era of fast-changing technology and business requirements, EAs also need to be updated and managed dynamically. As the use of architectural approach in RE becomes more prevalent, we may have the chance of using well-defined and performed RE processes to evolve and improve already established EAs. We suggest that this would be a remarkable topic for future work.

References

1. F. Brooks, "No Silver Bullet: Essence and Accidents of Software Engineering.",[*IEEE Computer Magazine* , 20, 1987], pp. 10-19.
2. Enterprise Architecture, http://searchcio.techtarget.com/definition/enterprise-architecture.
3. The Federation of Enterprise Architecture Professional Organizations, "A Common Perspective on Enterprise Architecture", [*Architecture & Governance Magazine vol. 9 issue 4 (2013)*], pp. 10-16.
4. K. Wiegers and J. Beatty, *Software Requirements*, Third Edition, p.15.
5. I. Summerville, *Software Engineering*, Eighth Edition. Addison-Wesley, p.143.
6. W. Engelsman and R. Wiering, "Goal-Oriented Requirements Engineering and Enterprise Architecture:Two Case Studies and Some Lessons Learned", *REFSQ 2012, Springer-Verlag*, p.306.
7. D. J. Nightingale and D. H. Rhodes, "Enterprise Systems Architecting: Emerging Art and Science within Engineering Systems", p.1.

8. The Open Group, *TOGAF Version 9.1*, p.6, p.48.
9. M. Pulkkinen, A. Naumenko, K. Luostarinen, "Managing information security in a business network of machinery maintenance services business – Enterprise architecture as a coordination tool", [*The Journal of Systems and Software 80*], p.1609.
10. Zachman Framework, http://www.zachman.com/about-the-zachman-framework.
11. V. Voucharas, M. V. Steenbergen, S. Jansen, S.Brinkkemper, "The Contribution of Enterprise Architecture to the Achievement of Organizational Goals: Establishing the Enterprise Architecture Benefits Framework", Technical Report (June 2010), p.42.
12. W. Engelsman, M. E. Iacob, H. M. Franken, "Architecture-Driven Requirements Engineering", p.2.
13. J. D. McKeen, H. A. Smith, *Making IT Happen: Critical Issues in IT Management*, Wiley, 2003.
14. M. Lange, J. Mendling, "An Experts' Perspective on Enterprise Architecture Goals, Framework Adoption and Benefit Assessment", p.5.