Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 53 (2016) 158 – 165

www.elsevier.com/locate/procedia

The 10th International Conference on Axiomatic Design, ICAD 2016

A systematic approach to enterprise architecture using axiomatic design

Farhangmehr Behrouz[a,*], Mehdi Fathollah[b]

[a] Department of industrial engineering, Islamic Azad university, Karaj branch, Alborz 31485- 313, Iran
[b] PhD, Assistant Professor, Department of industrial engineering, Islamic Azad university, Karaj branch, Alborz 31485- 313, Iran

* Corresponding author. Tel.: +98-912-363-7782; fax: +98-263- 441 -8156. E-mail address: f.mapna@yahoo.com

Abstract

Today, the organizations have complex structures and therefore, Enterprise Architecture (EA) could provide them with solutions to describe, coordinate, and align their business elements in order to achieve the strategic goals and deploy organizational governance. In this regard, various frameworks offered according to enterprise activities field. Multi-layered and pyramidal structure is the common feature of most frameworks, from strategic planning on top of the pyramid to information technology infrastructure at the bottom. So far, several models and methods are developed to specify the architecture requirements of each layer and trace architectural components at different layers (often with different substances), mainly just by descriptive and graphic tools. Translating and converting strategic requirements to processes, data, and technology, providing the organization big picture in detail and handling change management are the main purpose of EA. These cannot be achieved unless the requirements are accurately and systematically determined from the top to the bottom of the pyramid. Also, the architecture of each layer is designed to respond the requirements of the upper layer, while specifying the exterior and outward relationship between heterogeneous architectural components, not only does not cover all the needs, but also could be misleading for the organization. This paper attempts to deploy a methodology based on Axiomatic Design (AD) by using two axioms to systematically analyze the current enterprises capability and map the requirements of each layer of EA as the design domain into other domains.

Peer-review under responsibility of the scientific committee of The 10th International Conference on Axiomatic Design

Keywords: Axiomatic Design; Enterprise Architecture; Information Technology Analysis

1. Introduction

Most of the enterprises are encountering business changing, for instance a development of products and services or economic situations. According to these situations, they have to absolutely improve their business processes in order to be able to survive. In this regards, these enterprises, should adapt themselves to these changes effectively [1], [2]. In view of the increased business and organizational extension and dynamics, integration, agility and the ability to change, are becoming more and more important. Enterprises should thus pay considerable attention to their enterprise architecture [5]. Enterprise Architecture (EA) is the process of translating business vision and strategy into effective enterprise change by creating, communicating, and improving the key principles and models that describe the enterprise's future state and enable its evolution. [Gartner, 2008]

Large and medium-sized organizations regard the alignment of business and IT as the most important motive for working on an EA. Other important reasons for putting EA on the agenda are support for change processes and strengthening the flexibility of the company. [Roeleven, 2010]. Since EA artifacts are not sufficient for make alignment between business and IT within enterprises, enterprises are looking to find a method to address theirs challenges on competiveness by implementing EA artifacts [7]. So far Several EA Implementation Methodologies or EAIMs have been proposed by academics and practitioners in literature [8].

Peer-review under responsibility of the scientific committee of The 10th International Conference on Axiomatic Design
doi:10.1016/j.procir.2016.07.012

Although they are different in implementation practices and development phases, they are common in the concepts, principles of transition from current architecture (As-Is) to desire architecture (To-Be) [9].

In spite of the huge interest in EA it turns out that 66 percent of programs did not fulfill expectations [Roeleven, 2010]. This study intends to analyze the applications of AD theory to the Enterprise Architecture (EA), in order to provide a methodology for dealing with existing challenges. Axiomatic Design (AD) is distinguished from other systematic design methods by having design axioms that guide good design decisions [Suh, 1990]. In “literature review” section key concept of AD, EA and complexity are described. Also, in “research gap and need for action” section, current challenges in EA, and corresponding solutions by AD are presented. “Proposed EAIM” section describes proposed algorithm for EAIM based on AD and last section summarizes the work and suggests future research works.

2. Literature review

2.1. Key Concepts of Axiomatic Design

Axiomatic Design (AD) principles have been expanded and applied to numerous engineering and non-engineering applications and proved to provide structured implementation procedures [Kulak et al., 2010]. AD Theory was proposed by Nam Pyo Suh. The goal of AD is to establish a scientific basis for design and to improve design activities by providing a theoretical foundation based on logical and rational thought processes and tools [Suh, 2001 p.5].

The AD framework divides the design process into 4 domains [Suh, 2001 p.11]: the customer domain, the functional domain, the physical domain and the process domain. In each domain, there is a characteristic vector. Respectively, they are customer attributes (CAs), functional requirements (FRs), design parameters (DPs) and process variables (PVs). As shown in Figure 1, the domain on the left relative to the domain on the right represents "what we want to achieve", whereas the domain on the right represents the design solution of "how we choose to satisfy the needs (i.e., the what)" [Suh, 2001 p10].

The process of matching variables in one domain (e.g., FRs) with other variables in another domain (e.g., DPs) is called mapping: to go from WHAT to HOW [Cochran et al., 2000]. Therefore, when mapping the right domain to the left domain, “zigzagging” decomposition is used. Designers are requested to create a design hierarchy. FRs and DPs, PVs must be decomposed into a hierarchy respectively until a complete detailed design or until the design is completed [Suh, 2001 p21].It is noted that DPs are defined according to FRs in the same level and FRs of the lower level are defined based on the characteristics of DPs in the upper level. This decomposition process continues until the leaf (bottom) level is reached. The domains may have several levels of abstraction that jointly describe the technical system architecture [Marques etal., 2009].

During the mapping processes, the designer is guided by two fundamental axioms to produce a robust design: the Independence Axiom and the Information Axiom [Suh, 2001 p.16].

- Independence Axiom: Maintain the independence of the functional requirements (FRs).
- Information Axiom: Minimize the information content of the design.

In particular, the axioms provide criteria for distinguishing bad designs from good ones [Suh, 1990]. One important point to note is that Axim 2 is only applied when Axiom 1 has been satisfied. In most design tasks, it is necessary to decompose the problem hierarchically. The FRs, DPs, and PVs mapping process can mathematically be described as vectors [Suh, 2001 p18] in the design matrix. A design equation should be written for each transition between domains and at each decomposition level. Detailed information and elaborations on the scientific background of AD are provided by Suh [2001].

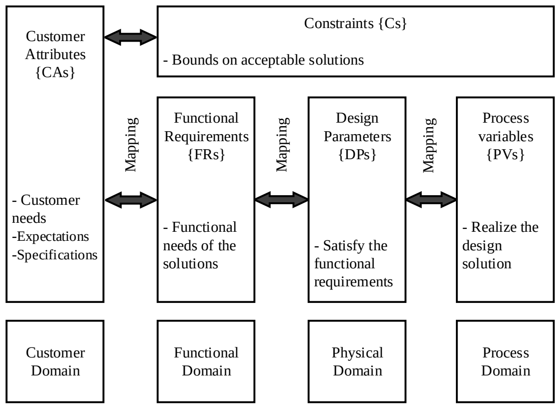

Fig. 1. Axiomatic design domains [Suh, 2001]

2.1.1. The Independence Axiom

Using vector notations for FRs and DPs, the relationship is expressed as the following design equation:

$$FR = ADP$$

Matrix A is called a design matrix. The characteristics of matrix A determine if the Independence Axiom is satisfied. If the design matrix is a diagonal matrix, it is an uncoupled design. Because each DP can satisfy a corresponding FR, the uncoupled design perfectly satisfies the Independence Axiom. When the design matrix is triangular, the design is a decoupled design. A decoupled design satisfies the Independence Axiom if the design sequence is correct. When a design matrix is neither diagonal nor triangular, the design becomes a coupled design. In a coupled design, no sequences of DPs can satisfy the FRs independently. Therefore, an uncoupled or a decoupled design satisfies the Independence Axiom and a coupled design does not. If a design is coupled, an uncoupled or decoupled design must be found through a new choice of DPs. It is noted that constraints (Cs) exist in the design. Constraints are generally defined from design specifications and they must be satisfied [14]. As an index for