The Quest for Underpinning Theory of Enterprise Architecture
General Systems Theory

Nestori Syynimaa
Faculty of Information Technology, University of Jyväskylä, Jyväskylä, Finland
Gerenios Ltd, Tampere, Finland

Keywords: Enterprise Architecture, Theory, General Systems Theory.

Abstract: Enterprise architecture originates from the 1980's. It emerged among ICT practitioners to solve complex problems related to information systems. Currently EA is also utilised to solve business problems, although the focus is still in ICT and its alignment with business. EA can be defined as a description of the current and future states of the enterprise, and as a change between these states to meet stakeholder's goals. Despite its popularity and 30 years of age, the literature review conducted on top information and management science journals revealed that EA is still lacking the sound theoretical foundation. In this conceptual paper, we propose General Systems Theory (GST) for underpinning theory of EA. GST allows us to see enterprises as systems of systems consisting of, for instance, social organisations, humans, information systems and computers. This explains why EA can be used to describe the enterprise and its components, and how to control them to execute the managed change. Implications to science and practice, and some directions for future research are also provided.

1 INTRODUCTION

Enterprise architecture (EA) has a long history dating back to 1980's, although there are some debate on who first introduced the concept (Kotusev, 2016). For larger audience, John Zachman is often called the father of EA, mainly due to his seminal publication entitled *A framework for information systems architecture* (Zachman, 1987) followed by *Enterprise Architecture: The issue of the Century* (Zachman, 1997) ten years later. Originally the Zachman framework was built to solve issues related to increasing complexity of information systems. Today EA is recognised as an essential reference point for both business and technology decisions (Kien *et al.*, 2015).

Enterprise architecture originates from practicum. It is a creation of software and systems engineers, and as such, is not built on scientifically testable foundations (Lapalme *et al.*, 2015). One of the thought-leaders of EA stated that it is not even possible to have a single overarching theory of EA (Graves, 2012; Graves, 2015). By theory we refer to the "statements providing a lens for viewing or explaining the world" (Gregor, 2006).

This conceptual paper aims for strengthening the scientific foundation of enteprise architecture. We seek to provide a type IV theory in Gregor's taxonomy, i.e., a theory of explanation and prediction. To accomplish this, we draw on General Systems Theory (GST) and demonstrate how it underpins enterprise architecture.

The paper is structured as follows. First, in this section, we introduce the paper. In the second section we provide definition of EA and review the state of the current research on EA theory. In the third section, we introduce General Systems Theory. In the fourth section, we discuss about enterprises as systems. In the fifth section, we discuss on GST as a scientific foundation of EA. Finally we provide concluding remarks and directions for future research.

2 ENTERPRISE ARCHITECTURE

To build a sound scientific foundation for enterprise architecture, we start by defining the concept and review the current literature.

Syynimaa, N.
The Quest for Underpinning Theory of Enterprise Architecture - General Systems Theory .
DOI: 10.5220/0006314904000408
In *Proceedings of the 19th International Conference on Enterprise Information Systems (ICEIS 2017) - Volume 3*, pages 400-408
ISBN: 978-989-758-249-3

2.1 Definition of Enterprise Architecture

Currently there are no single accepted definition for enterprise architecture. Thus, we need to draw on various existing definitions from the literature.

Gartner defines EA as "a discipline for proactively and holistically leading enterprise responses to disruptive forces by identifying and analyzing the execution of change toward desired business vision and outcomes" (Gartner, 2013). As such, EA can be seen as an activity aiming for survival of an enterprise.

The Open Group defines architecture as "a formal description of a system, or a detailed plan of the system at component level, to guide its implementation" and "the structure of components, their inter-relationships, and the principles and guidelines governing their design and evolution over time" (The Open Group, 2009, p. 9). These definitions implies that EA is a structure and a description of an enterprise.

Zachman defines architecture as "that set of design artifacts, or descriptive representations, that are relevant for describing an object such that it can be produced to requirements (quality) as well as maintainded over the period of its useful life (change)" (Zachman, 1997). This definition is in line with the previous one; it sees EA as a structure and its description.

MIT Center for Information Systems Research (CISR) defines EA as "the organizing logic for business process and IT capabilities reflecting the integration and standardization requirements of the firm's operating model" (MIT CISR, 2016). CISR also defines EA as description of an enterprise. However, EA is limited only to business processes and IT capabilities. As such, this definition limits EA to certain domains.

The Federation of Enterprise Architecture Professional Organizations (FEAPO) defines EA as "a well-defined practice for conducting enterprise analysis, design, planning, and implementation, using a holistic approach at all times, for the successful development and execution of strategy" (FEAPO, 2013, p. 11). Similar to Gartner's definition, FEAPO sees EA as an activity aiming for strategic advantages.

ISO/IEC/IEEE 42010 *Systems and software engineering – Architecture description* standard defines architecture as "fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution" (ISO/IEC/IEEE, 2011, p. 2). This definition also sees EA as a structure and its description.

As the former definitions demonstrates, enterprise architecture is a vague concept. However, there are two concepts which are shared by the definitions. These are a formal description of the current and future states of an enterprise, and a managed change between these states to meet stakeholders' goals (Syynimaa, 2015).

Enterprise architecture descriptions are divided to layers or domains. Typically there are four layers, namely business, information (or data), information systems, and technology (The Open Group, 2009; van't Wout *et al.*, 2010).

2.2 Current Research on Theory of Enterprise Architecture

The major contribution for any scientific discipline is likely to be found in the leading journals (Webster and Watson, 2002). Therefore, we first searched top eight journals of Information Systems (IS) field as rated by Association of Information Systems (AIS, 2011) for the term "enterprise architecture". From the top IS journals between 2000 and 2016, we found 77 articles. Further analysis of the articles revealed that only 24 were actual enterprise architecture articles. It is notable that the leading IS journal, MIS Quarterly, had no EA articles at all. However, although the MIS Quarterly Executive (MISQE) is not one of the leading IS journals, we decided to include it into our review. As a result, we found 10 EA articles from MISQE. This encouraged us to expand the search to include the top Management Science (MS) journals as rated by Chartered Association of Business Schools (ABS, 2010). Disappointingly, we did not found any enterprise architecture articles from MS journals.

None of the found articles were related to the theory of enterprise architecture. Therefore the search was expanded to include the Journal of Enterprise Architecture (JEA), which is the only journal dedicated to enterpise architecture. We reviewed all JEA issues between 2005 and 2016 to find EA theory articles. As a result, we found eight articles. Most of the articles were about different variations of system theories, such as Viable System Model (VSM). For instance, Zadeh et al. (2012) studied VSM as a theoretical basis for EA principles, whereas Lapalme and de Guerre (2012) and Jensen-Waud (2011) studied EA as a socio-technical system. Sidorova and Kappelman (2011) used Actor-Network Theory (ANT) to study how EA could be better utilised for IT-business alignment.

According to leading enterprise architecture scholars, its scientific foundation needs to be strenghtened (Lapalme *et al.*, 2015). The reviewed articles did not provide such a foundation. Moreover, the number of enterprise architecture articles in top IS journals indicates that EA as a scientific discipline is still immature. Finally, the results from top MS journals indicates that enterprise architecture is still seen purely as an ICT-issue.

3 GENERAL SYSTEMS THEORY

Next phase on our journey to build a sound scientific foundation for enterprise architecture is to introduce our theoretical perspective: General Systems Theory (GST). GST was originally introduced by von Bertalanffy (1951). Its purpose is to be “a body of systematic theoretical constructs which will discuss the general relationships of the empirical world” (Boulding, 1956). In other words, it is a way of thinking about, or an approach to study, the empirical world (Von Bertalanffy, 1968).

Key concept of GST is a *system*. It can be defined as a “set of things working together as parts of a mechanism or an interconnecting network; a complex whole” (Oxford Dictionaries, 2010). Another scientific discipline closely related to systems is *cybernetics*. It can be defined as a scientific study of controlling and communication in animal and machine (Wiener, 1948).

3.1 Closed and Open Systems

Systems can be categorised by their openness. In this sense, there are two types of systems: *closed systems* and *open systems*. Closed system is a system which is not exchanging any material or information with its environment (see Figure 1). Environment refers to a “context determining the setting and circumstances of all influences upon a system” (ISO/IEC/IEEE, 2011, p. 2).

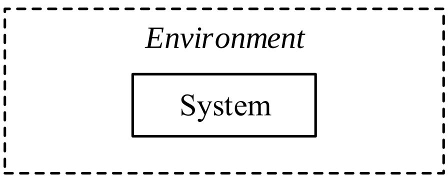

Figure 1: Closed system.

As an example, a (well insulated) coffee cup with its lid on is a closed system: we cannot see inside it, and it does not emit heat nor light.

The open systems are systems which are interacting with their environment, e.g., exchanging material and information. Continuing the previous coffee cup example, if we remove the lid from the coffee cup, it becomes an open system: we can see inside it and we can sense whether its hot. We can also stir the content and even drink the content. By stirring we are providing input to the system and by drinking we are consuming its output.

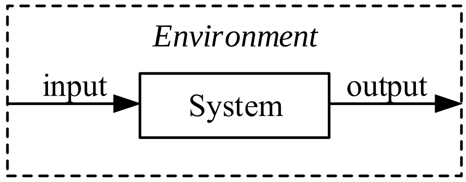

Figure 2: Open system.

3.2 Systems and Feedback Loops

Some open systems have a feedback loop (Ashby, 1957) which makes the system controllable. The system with a feedback loop is illustrated in Figure 3. Such a system is feeding at least some of its output back to system as an input.

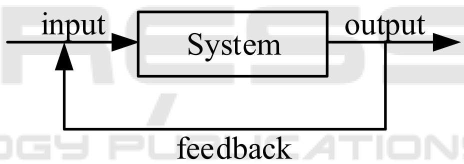

Figure 3: Feedback loop.

There are two types of feedback loops, *positive feedback* and *negative feedback* (Ashby, 1957). System with a positive feedback either increases or decreases indefinitively (Figure 4). An example of a positive feedback system is a population with a fixed positive birthrate. A number of births per year depends on the size of the population, which, in turn, increases the population. And when the population grows, the number of births per year also increases, leading to an exponential growth and finally into destruction of the system. If the fixed birthrate would be negative, the population would be destroyed due to extinction.

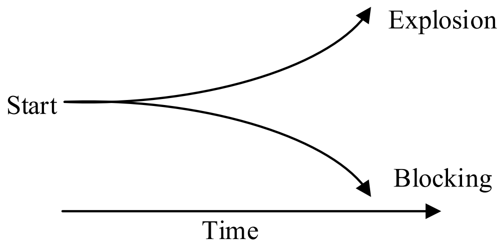

Figure 4: Positive feedback.

Negative feedback, as illustrated in Figure 5, requires a sensor and a controller which makes the system adaptive, i.e., controllable. The controller controls the system by adjusting the input, and the output is monitored with the sensor. The controller has a known reference value where the output is compared to. The variation from the reference value leads to negative correction, adjusting the system towards a desired goal as illustrated in Figure 6.

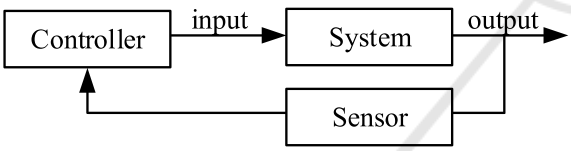

Figure 5: Feedback control.

A classic example of negative feedback system is a radiator with a thermostat. If the starting temperature of the room is higher than the temperature selected by the thermostat, the waterflow to radiator is closed. When the temperature decreases below the desired temperature, the waterflow to radiator is opened again. Eventually this leads to the desired room temperature. This kind of state of self regulatory system, which is keeping its state almost constant, is called homeostasis (Cannon, 1932).

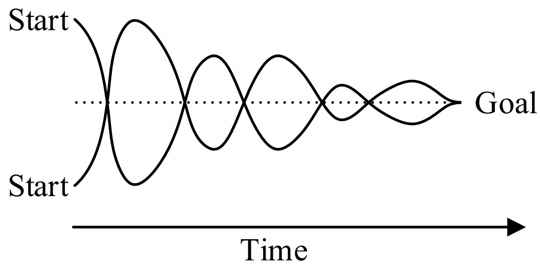

Figure 6: Negative feedback.

Let's discuss about controlling the system in more detail. Ashby introduced his famous Law of Requisite Variety in 1957: "Control can be obtained only if the variety of the controller is at least as great as the variety of the situation to be controlled" (Ashby, 1957). In common terms the law can be stated as follows. In order to the system remain stable, the number of states of the control mechanism of the system must be greater or equal to the number of the states of the system itself. In other words, you can keep the system stable by controlling it only if you have at least as many controlling options than there are possible factors affecting the state of the system. We can demonstrate this using the previous radiator example. If we study the radiator as a system and a thermostat as a controller, we find that there are only two states: water is flowing in radiator or it is not. This is also what the thermostat can control, so we can state that the system is stable according to Ashby's law. However, if we widen our scope and study the room where the radiator is located as a system, the situation is quite opposite. We have a radiator with a thermostat which can heat the room. The maximum theoretical temperature of the room is therefore the maximum temperature of the radiator. We can choose the desired temperature of the room by adjusting the radiator's thermostat. However, what we cannot adjust is the outside temperature. So, if we let the desired room temperature to be +20 C, the maximum temperature of the radiator +60 C, and the outside temperature -80 C, the water of the radiator eventually freezes and the system breaks. The given example here is simplified and does not take into account for instance the insulation of the room etc., but gives a general view what Ashby's law means.

3.3 Hierarchy of Systems

General Systems Theory allows us to see the empirical world as a system of systems. This system of systems is an "arrangement of theoretical systems and constructs in a hierarchy of complexity" (Boulding, 1956, p. 202). This hierarchy can be seen in Table 1. Next we will walk through these different levels.

Table 1: Boulding's Hierarchy of Systems.

Level	Name
9	Transcendental Systems
8	Social Organisations
7	Human Beings
6	Animals
5	Plants
4	Cells
3	Thermostats
2	Clockworks
1	Frameworks

First level, called *frameworks*, is that of the static structure, such as the pattern of atoms in a molecule. The second level, called *clockworks*, is that of simple dynamic system, such as a solar system, a molecule, or a machine. The third level, called *thermostat*, is that of the control mechanism or cybernetic system. The fourth level, called *cells*, is that of open systems or self maintaining structures, such as cells. This is also the level where life differentiates itself from not life. The fifth level, called *plant*, is that of genetic-societal. In this level, for instance, there are division of labor among cells. However, there are no highly specialised sense organs or information receptors. The sixth level is called *animal*. This level is characterised by increased mobility, teleological behaviour and self-awareness. There are higly specialised sense organs, such as eyes and ears. The seventh level is called *human beings*, where an individual human is considered as a system. This level differentiates from the animal level for instance by the self consciousness. The eighth level is called *social organisation*. This level regards social organisations as a system of humans, where humans are not persons but roles. That is, one person can be part of many different social organisations. The ninth level is called *transcendental*. This level includes ultimates and absolutes, and the inescapable unkowables that have a systematic structure and relationships, such as, religions. (Boulding, 1956).

4 ENTERPRISE AS A SYSTEM

Now that we have introduced the enterprise architecture and General Systems Theory, we can discuss enterprises as systems.

4.1 Definition of Enterprise

The simplest definition for an enterprise is "a business or a company" (Oxford Dictionaries, 2010). A bit wider definition is "any collection of organisations that has a common set of goals" (The Open Group, 2009, p. 5). This includes also enterprises which are not businesses *per se*, such as, organisational departments or public sector agencies. Moreover, also extended enterprises, i.e., partners, suppliers, and customers can be included in the enterprise (The Open Group, 2009). In this sense, enterprise is a system defined by its boundaries. An example of different hierarchical levels of enterprise are illustrated in Figure 7. For instance, we can see enterprise as a society, an industry sector, an organisation, or a department of the organisation.

Some scholars see enterprises as socio-technical systems consisting of people and technological artefacts (Emery, 1972; Trist, 1981; Lapalme and deGuerre, 2013). To our mind this view is limited as it rules out, for instance, enterprises which are pure social organisations. Instead, following the Boulding's hiearchy of systems, we define enterprise as a social organisation defined by its boundaries and consisting of social organisations and human beings. Enterprise *may* also include clockworks and thermostats, such as, tools, machines, computers, and information systems.

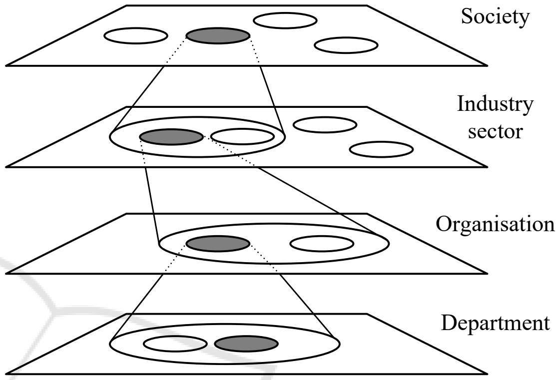

Figure 7: Hierarchical Levels of Enterprise.

4.2 Controlling the Enterprise

Essential to any enterprise is that it must be managed and led (Deming, 2000). Managing the enterprise refers to management of its internal activities. However, enterprises themselves can also evolve, which can take place naturally (by chance) or deliberately by design (Lee, 2010; Proper, 2013). Designing the enterprise refers to any activities related to changing the enterprise for some reason. This requires that the enterprise can be managed and led.

As we regard enterprises as systems, they can be managed by controlling them as any system. However, if we do not know how the enterprise operates, the controlling is difficult – if not impossible. This is called the *Problem of Black Box* (Ashby, 1957). If we only know the input and output interfaces of the enterprise, all we can do is to deduce how the enterprise works by experiment, i.e., by trying different values for inputs and by monitoring the resulting outputs. With enough data, one can form testable hypotheses and learn how the enterprise works. However, enterprise may have states to which it can not be returned. These states are called *inaccessible states* (Ashby, 1957). For instance, if the enterprise is a business, it may run out of money and end up to bankrupt. After that state, the enterprise is not able to continue its operations, regardless of the given inputs.

Another management challenge is very large and complex enterprises, such as, societies. These kind of enterprises can only be treated statistically (Ashby, 1957).

The essential question in controlling the enterprise is what and how to control? This question is out-of-scope of this paper and remains to be answered by future research.

5 ENTERPRISE ARCHITECTURE AND GENERAL SYSTEMS THEORY

Now that we have introduced and discussed different concepts of the article, we can put it all together. The premise of GST is that the empirical world is essentially a system of systems. There are different categories of these systems and each category have their own scientific discipline(s). For instance, physics study frameworks and their properties, psychologists humans, and sociologists social organisations. Most of these scientific disciplines share three concepts (Boulding, 1956): individual, population and growth. Naturally, what we mean by individual depends on the system's category. For instance, physics may see an atom as an individual and a molecule as a population, whereas chemists may see a molecule as an individual.

In previous section we defined enterprise as a social organisation (system) which consists of human beings (systems) and social organisations (systems). This means that enterprise is also a system of systems. Let's return to our definition of enterprise architecture. We defined EA as (1) a formal description of the current and future states of an enterprise, and (2) a managed change between these states to meet stakeholders' goals. Next we will discuss these two definitions using the GST.

5.1 Enterprise Architecture Descriptions

The first part of our definition is a noun referring to architecture descriptions. Architecture description can be defined as a "work product used to express an architecture" (ISO/IEC/IEEE, 2011, p. 2). As GST sees empirical world as a system of systems, it implies that every system, such as an enterprise, can be described. Moreover, if we can describe a system, we can also describe systems it consists of, i.e., its components. As stated earlier, enterprise architecture descriptions are typically produced for four layers; business (B), information (I), information systems (S), and technology (T). These descriptions are expressing the architecture of the enterprise from different viewpoints. The B layer expresses a social organisation: what are roles and processes of the enterprise. The S and T layers expresses the various clockworks and thermostats of the enterprise, namely the information systems and the technology they are built on. However, the I layer does not express any system as such. The I layer typically consists of grammars and dictionaries, i.e., definitions and meanings of things. Therefore, it can be regarded as collection of properties or rules of the enterprise rather than a description of a system.

Architecture descriptions are produced for two reasons. You either want to understand the enterprise or you want to change it somehow. Descriptions of the current state of the enterprise are used to increase our understanding about the enterprise. In other words, how the enterprise works, what are its components, and how these components are linked to each other. Descriptions of the future state(s) of the enterprise are plans of how what the enterprise should be in the future. There can be multiple different scenarios, i.e., descriptions of different future states, each expressing one possible solution to achieve stakeholders' goals.

5.2 Executing the Managed Change

The second part of our definition of enterprise architecture is a verb, referring to changing the enterprise from the current state to the future state. Because GST allows us to see the enterprise a system, it also allows us to control the enterprise. An ability to control the enterprise is a requirement for changing the enterprise.

To successfully change the enterprise, one must know both the current and the future state of the enterprise. The descriptions of the current state should include details on control mechanisms of the enterprise and its components. For instance, if we use a company as an example of the enterprise, the descriptions should include its management structure and processes, organisational structure, staff's skills and knowledge, and organisation culture. The breadth and depth of the needed descriptions depends on the focus and magnitude of the change. The descriptions of the future state should clearly express the desired state of the enterprise. Again, the breadth and depth of the needed descriptions depends on the future state.

When the current and future states of the enterprise are described, one can start to execute the managed state. The challenge is that enterprises are systems of systems. Thus, in order to change the enterprise, we may need also to change its components (systems). As our definition of enterprise suggests, enterprise may consists of many different kinds of systems. These may include social organisations, such as organisational departments, humans, and clockworks, such as machines and computers. Each different system have individual control mechanism and, therefore, require different controlling approach. For instance, changing an organisational culture requires different approach than changing a computer software.

6 CONCLUSIONS

The purpose of this paper was to strengthen the scientific foundation of enterprise architecture. We defined enterprise architecture (EA) as (1) a formal description of the current and future states of an enterprise, and (2) a managed change between these states to meet stakeholders' goals.

In the quest for the underpinning theory of EA, we first reviewed the top IS and MS journals and demonstrated a research gap: EA is lacking a sound theoretical foundation. Therefore, the paper is focused to General Systems Theory (GST), a promising candidate for underpinning theory of EA.

GST allows us to see enterprises as systems of systems. To be more specific, in Boulding's hierarchy, enterprises are social organisations consisting of social organisations and humans. As such, one can produce descriptions of enterprise and its components. As we see enterprises are systems, we also demonstrated that they can be controlled as any other system. However, as enterprise consists of different kind of systems, controlling the enterprise may require individual controlling of each system it consists of.

We were looking for a type IV theory to predict and explain the enterprise architecture. We argue that GST provides explanation and a sound scientific foundation for producing architecture descriptions and executing managed change. However, as the title of Boulding's (1956) paper states, GST is the skeleton of science. In the context of enterprise architecture, this means that GST gives us only a way to interpret enterprises as systems from various scientific point of views. For instance, it does not give us a single theory to control enterprises, but a way to decompose it to other systems and to control them individually. However, we do argue that GST also provides prediction. For instance, if one is executing managed change of a company, a social organisation, one can utilise management and organisation science. Therefore, we argue that GST can be used as scientific foundation of enterprise architecture.

6.1 Implications to Science

Enterprise architecture is a multidisciplinary concept. We demonstrated that enterprises are systems of systems consisting of social organisations and human beings. They may also include other systems, such as machines, computers, and information systems. This implies that EA is not limited to ICT related matters, but covers the enterprise as a whole. Therefore, EA research should be expanded outside of the IS field to include business, management, and organisation sciences. Having said that, we also need to remember that ICT and information systems are essential components of our society, and their importance is growing in the future.

6.2 Implications to Practise

A recent discussion on the role of enterprise architecture implies that currently EA is seen merely as a practice of producing documentation, leaving business development with a little of attention (Bloomberg, 2014). According to Zachman (2015), this is similar to taking X-ray pictures: they are purely snap-shots from a certain time. Instead, as Zachman suggests, enterprise architects should be more like doctors. They should analyse those X-ray pictures, make diagnosis and prescribe solutions. From this point of view, our results may have significant consequences to EA practitioners. As the scope of EA expands outside the ICT to cover the business, practitioners should also expand their skills accordingly.

6.3 Directions for Future Research

As our literature review revealed, EA is still seen purely as an IT issue. Utilising GST as the underpinning theory of EA gives a good starting point to focus the future research on EA.

First, one could research which system categories are typical in different enterprises. For instance, does all enterprises include clockworks or thermostats?

Second, one could research which controlling approaches does the current literature offer for different system categories. For instance, which organisational theories would explain why some organisations are easiern to change than the others? Or which psychological or cognitive theories would explain how to change people?

REFERENCES

ABS. (2010). Academic Journal Quality Guide. Version 4. URL: http://www.associationofbusinessschools.org/sites/default/files/Combined%20Journal%20Guide.pdf [Sep 21st 2014].

AIS. (2011). *Senior Scholars' Basket of Journals*. URL: https://aisnet.org/?SeniorScholarBasket [Nov 24th 2016].

Ashby, W. R. (1957). *An introduction to cybernetics.* London: Chapman & Hall.

Bloomberg, J. 2014). Is Enterprise Architecture Completely Broken? Available from: http://www.forbes.com/sites/jasonbloomberg/2014/07/11/is-enterprise-architecture-completely-broken/].

Boulding, K. E. (1956). General Systems Theory-The Skeleton of Science. *Management Science, 2*(3), 197-208.

Cannon, W. B. (1932). *The Wisdom of the Body.* New York: W W Norton & Company.

Deming, W. E. (2000). *The New Economics: for industry, government, education.* Cambridge, Massachusetts: MIT Press.

Emery, F. E. (1972). Characteristics of Socio-Technical Systems. *In:* Davis, L. E. & Taylor, J. C. (eds.) *Design of Jobs.* Penguin Books.

FEAPO (2013). The FEAPO enterprise architecture perspective initiative: A common perspective on enterprise architecture. *Architecture and Governance Magazine, 9*(4), 11-17.

Gartner. (2013). *IT Glossary: Enterprise Architecture (EA).* URL: http://www.gartner.com/it-glossary/enterprise-architecture-ea/ [Feb 21st 2013].

Graves, T. (2012). *The theory of enterprise-architecture*. URL: http://weblog.tetradian.com/2012/09/16/the-theory-of-enterprise-architecture/ [Sep 29th 2015].

Graves, T. (2015). *Theory and metatheory in enterprise-architecture*. URL: http://weblog.tetradian.com/2015/01/19/theory-and-metatheory-in-entarch/ [Sep 29th 2015].

Gregor, S. (2006). The Nature of Theory in Information Systems. *MIS Quarterly, 30*(3), 611-642.

ISO/IEC/IEEE (2011). Systems and software engineering — Architecture description.

Jensen-Waud, A. (2011). Processes of Sense-Making and Systems Thinking in Government Enterprise Architecture Planning. *Journal of Enterprise Architecture, 7*(3), 8-18.

Kien, S. S., Rosemann, M. & Yetton, P. (2015). Special Issue Editorial: The Critical Contribution of Enterprise Architecture to the Performance of Large Private and Public Organizations. *MIS Quarterly Executive, 4*(14), iii-iv.

Kotusev, S. (2016). The History of Enterprise Architecture: An Evidence-Based Review. *Journal of Enterprise Architecture, 12*(1), 29-37.

Lapalme, J. & de Guerre, D. W. (2012). Can a Re-Discovery of Open Socio-Technical Systems Strengthen EA? *Journal of Enterprise Architecture, 8*(1).

Lapalme, J. & deGuerre, D. (2013). An Open Socio-Technical Systems Approach to Enterprise Architecture. *In:* Gøtze, J. & Jensen-Waud, A. (eds.) *Systems Volume 3. Beyond Alignment. Applying Systems Thinking in Architecting Enterprise.* London: College Publications.

Lapalme, J., Gerber, A., Van der Merwe, A., Zachman, J., Vries, M. D. & Hinkelmann, K. (2015). Exploring the future of enterprise architecture: A Zachman perspective. *Computers in Industry*.

Lee, A. S. (2010). Retrospect and prospect: information systems research in the last and next 25 years. *Journal of Information Technology, 25*, 336-348.

MIT CISR. (2016). *Enterprise Architecture. Introduction.* URL: http://cisr.mit.edu/research/research-overview/classic-topics/enterprise-architecture/.

Oxford Dictionaries. (2010). *Oxford Dictionaries. April 2010*. URL: http://oxforddictionaries.com/ [Nov 18th 2014].

Proper, H. A. (2013). Enterprise Architecture. Informed steering of enterprises in motion. *ICEIS 2013.* Angers, France.

Sidorova, A. & Kappelman, L. A. (2011). Better Business-IT Alignment Through Enterprise Architecture: An Actor-Network Theory Perspective. *Journal of Enterprise Architecture, 7*(1), 39-47.

Syynimaa, N. (2015). Modeling the Dynamics of Enterprise Architecture Adoption Process. *In:* Hammoudi, S., Maciaszek, L., Teniente, E., Camp, O. & Cordeiro, J., eds. ICEIS 2015, LNBIP 241, 2015. Springer International Publishing Switzerland, 577-594.

The Open Group (2009). *TOGAF Version 9.* Zaltbommel, Netherlands: Van Haren Publishing.

Trist, E. (1981). The evolution of socio-technical systems. *Occasional paper, 2*.

Van't Wout, J., Waage, M., Hartman, H., Stahlecker, M. & Hofman, A. (2010). *The Integrated Architecture Framework Explained: Why, What, How.* Berlin: Springer.

Von Bertalanffy, L. (1951). General System Theory: A New Approach to Unity of Science. *Human Biology, 23*, 303-361.

Von Bertalanffy, L. (1968). *General System Theory: Foundations, Development, Applications.* New York: George Braziller.

Webster, J. & Watson, R. T. (2002). Analyzing the Past to Prepare for the Future: Writing a Literature Review. *MIS Quarterly, 26*(2), 13-23.

Wiener, N. (1948). *Cybernetics: Control and communication in the animal and the machine.* New York: Wiley.

Zachman, J. (2015). *Interview of John Zachman by Sunil*

Dutt Jha: What is the Role of the Enterprise Architect? URL: https://www.youtube.com/embed/dcpFEurhxbw.

Zachman, J. A. (1987). A framework for information systems architecture. *IBM Systems Journal, 26*(3).

Zachman, J. A. (1997). Enterprise architecture: The issue of the century. *Database Programming and Design, 10*(3), 44-53.

Zadeh, M. E., Millar, G. & Lewis, E. (2012). Reinterpreting the TOGAF® Enterprise Architecture Principles Using a Cybernetic Lens. *Journal of Enterprise Architecture, 8*(2), 9-17.