Truth Lumor

TOWARDS THE DESIGN OF AN AGILE ENTERPRISE ARCHITECTURE MANAGEMENT METHOD

UNIVERSITY OF JYVÄSKYLÄ
DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION SYSTEMS
2016

ABSTRACT

Lumor, Truth
Towards the Design of an Agile Enterprise Architecture Management Method
Jyväskylä: University of Jyväskylä, 2016, 158p.
Information Systems, Master's Thesis
Pulkkinen, Mirja and Hirvonen, P. Ari

Enterprises engage in dynamic environments and do need capabilities that will enable them to (re)configure and integrate existing capabilities or create new capabilities in order to remain viable and competitive in such environments. Enterprise Architecture (EA) by definition has the propensity to contribute to the agility of the enterprise in turbulent environments provided it is managed and guided by a methodic approach in a way that strengthens the capabilities with which the enterprise senses and responds to changes in its environment. However, the agility of the Enterprise Architecture Management (EAM) function and its relation with the agility of the enterprise as a whole has been sparingly studies in the EA literature. This thesis contributes to the EA literature by designing an agile EAM method that supports the agility of the EAM function and the agility of the enterprise as a whole. A number of steps were taken to this end. First, the factors that influence the agility of the enterprise, namely; agility drivers, agility providers and agility capabilities, were identified. Second, literature on enterprise architecture, lean and agile principles and values, and a set of proposed theories were reviewed to form the theoretical foundation for the method design effort. Third, the lean and agile principles and values were adapted for EAM work. Fourth, the adapted lean and agile principles and values were consolidated into eight essential elements (EEs) that an agile EAM should possess in order to support the agility of the EAM function and the agility of the enterprise. Finally, an agile EAM method (Agile EAMM) was constructed to embody the EEs. The proposed Agile EAMM acts as a dynamic capability that provides the sensing capabilities, the leaning capabilities and the coordinating capabilities needed to effectively carry out enterprise architecting as an integrating capability in transforming as-is architecture (existing operational capabilities) into to-be architecture (reconfigured operational capabilities). The compatibility of the Agile EAMM with existing methods was demonstrated using TOGAF's ADM, and its efficacy was evaluated by means of an illustrative scenario. Conceptually, the Agile EAMM has the propensity to support the agility of the EAM function and the agility of the enterprise as a whole. Aside the Agile EAMM, the thesis contributes lean and agile principles and values applicable to EAM, the eight EEs, and areas for further research to the EA literature.

Keywords: Enterprise Architecture Management, Agile Methods, Enterprise Agility

Lumor, Truth
Ketterä yrityksen kokonaisarkkitehtuurin johtamisen metodi ja sen suunnittelu
Jyväskylä: Jyväskylän yliopisto, 2016, 158p.
Tietojärjestelmien, Pro gradu-tutkielma
Pulkkinen, Mirja and Hirvonen, P. Ari

Dynaamisissa ympäristöissä toimiessaan yritykset tarvitsevat kyvykkyyksiä, jotka mahdollistavat sekä nykyisten että uusien voimavarojen (uudelleen)määrittelyn ja integroinnin. Tällaiset kyvykkyydet mahdollistavat kilpailukykyisenä säilymisen. Yrityksen kokonaisarkkitehtuuri (EA) pyrkii edistämään yrityksen ketteryyttä vaihtelevissa ympäristöissä. Se voi vahvistaa kyvykkyyksiä, joiden avulla yritys tunnistaa ja vastaa ympäristössään tapahtuviin muutoksiin, mikäli sitä käytetään metodisena lähestymistapana johtamisessa. Kuitenkin Yrityksen kokonaisarkkitehtuurin johtamisen (EAM) toimintoa ja sen suhdetta yrityksen kokonaisvaltaiseen ketteryyteen on tutkittu vasta vähän. Tässä tutkielmassa suunnitellaan ketterä yrityksen kokonaisarkkitehtuurin johtamisen metodi, joka tukee yrityksen kokonaisarkkitehtuurin johtamisen toimintoa ja yrityksen ketteryyttä kokonaisuudessaan. Mallin suunnittelua edelsi useita vaiheita. Ensimmäisenä pyrittiin tunnistamaan yrityksen ketteryyteen vaikuttavia tekijöitä, eli sitä edistäviä ja mahdollistavia tekijöitä sekä ketteryyskyvykkyyksiä. Tämän jälkeen tarkasteltiin yritysarkkitehtuuria, kevyen ja ketterän kehittämisen taustalla olevia periaatteita ja arvoja sekä ehdotettuja teorioita käsittelevää kirjallisuutta. Tätä aineistoa käytettiin teoreettisena tukena metodin suunnittelun taustalla. Tämän jälkeen kevyet ja ketterät periaatteet ja arvot muokattiin sopiviksi yrityksen kokonaisarkkitehtuurin johtamiseen. Sitten nämä muokatut periaatteet ja arvot koottiin yhteen kahdeksaksi keskeiseksi elementiksi (EE), joita ketterän yrityksen kokonaisarkkitehtuurin johtamisen metodin tulisi sisältää tukeakseen ketterää yrityksen kokonaisarkkitehtuurin johtamisen toimintoa sekä koko yrityksen ketteryyttä. Viimeisenä ketterä yrityksen kokonaisarkkitehtuurin johtamisen metodi (Agile EAMM) muodostettiin ilmentämään aiemmin tunnistettuja kahdeksaa keskeistä elementtiä. Ehdotettu ketterä yrityksen kokonaisarkkitehtuurin johtamisen metodi toimii dynaamisena kyvykkyytenä, joka tarjoaa tunnistavia kyvykkyyksiä, yksinkertaistavia kyvykkyyksiä ja koordinoivia kyvykkyyksiä. Näitä tarvitaan muuttaessa yrityksen kokonaisarkkitehtuuria nykyisestä arkkitehtuurista paranneltuun arkkitehtuuriin, eli olemassa olevista toiminnallisista kyvykkyyksistä uudelleenjäsenneltyihin toiminnallisiin kyvykkyyksiin. Tämän muodostetun yrityksen kokonaisarkkitehtuurin johtamisen metodin yhteensopivuus olemassa olevien metodien kanssa varmistettiin TOGAFin ADM:n avulla, ja sen toimivuutta arvioitiin esimerkkiskenaarion avulla. Käsitteellisesti, kehitetty malli tukee yrityksen kokonaisarkkitehtuurin johtamisen toiminnon sekä koko yrityksen ketteryyttä. Tämän lisäksi tutkielma esittelee kevyitä ja ketteriä arvoja ja periaatteita, jotka ovat sovellettavissa yrityksen kokonaisarkkitehtuurin johtamisessa, kahdeksan keskeistä elementtiä, sekä ehdotuksia jatkotutkimusaiheille.

Avainsanat: Yrityksen arkkitehtuurin hallinta, Ketterä metodi, Ketterä yritys

ACKNOWLEDGEMENT

I am sincerely grateful to my supervisors: Dr. Mirja Pulkkinen and Dr. Ari Hirvonen, for entrusting such a timely industry project to me, and for providing guidance and insightful comments throughout the research process. Many thanks go to the teaching and non-teaching staff of the Department of Computer Science and Information Systems especially, the Head of SIM (Professor Tuure Tuunanen), the International Coordinator (Ms Niina Ormshaw), and the Study Advisor (Dr. Oleksiy Mazhelis), for their immense support during my studies. To my colleagues: Aseem, Dicle, Hojat, Jari, and Kamila, I say a big thank you for giving me a "family away from home" experience. Last but not the least; I thank my wife, daughter, son, and mum, for permitting me to spend such a long time away from home in pursuit of a master's degree. Asantewaa, Dzodzi, Elikem, and Aku, I absolutely owe you a lot.

FIGURES

TABLES

TABLE OF CONTENTS

1 CHAPTER 1 INTRODUCTION

1.1 Background

As enterprises are exposed to continuously changing environments fueled by factors including; changing market requirements, regulations, advancements in technologies, and competitor actions; they are forced to develop capabilities that enable them to sense and respond to these changes in ways that promote their sustainability and competitiveness over time. One of these capabilities is enterprise agility. In this thesis, enterprise agility is defined as;

> "*the ability of the enterprise to quickly (i) sense relevant changes that affect its competitive bases (e.g. speed, flexibility, innovation proactivity, quality and profitability), and (ii) appropriately respond to such changes through the building or reconfiguration of requisite capabilities (including IT-enabled resources, practices, processes, and relationships) to remain innovative and competitive in a fast changing business environment*"

Within the enterprise architecture community, enterprise architecture (EA) is believed to possess the ability to foster business strategy, to align ICT capabilities to such strategies, and to drive organizational impact, for example, in the form of enterprise agility (Espinosa, Boh, & DeLone, 2011). However, enterprises do face challenges in realizing these potentials (Buckl et al., 2011; Kaisler, Armour, & Valivullah, 2005; Lucke, Krell, & Lechner, 2010). For instance, D. Simon, Fischbach, & Schoder (2013) claim that the inability of some architecture teams to harness the full potential of EA at enterprise level occurs when IT and business alignment is relegated to operational level.

Indeed, EA has the propensity to promote enterprise agility provided it is managed and guided by a methodic approach in a way that strengthens the capabilities of the enterprise that enables it to sense and respond to changes in the business environment. However, 'agility' is not generally seen as an attribute of EA management methodologies. For instance, most prior research efforts at developing the building blocks or components of Enterprise Architecture Management (EAM) either do not consider "agility" as an attribute of the EAM itself (e.g. Ahlemann, Stettiner, Messerschmidt, & Legner, 2012; Aier, Gleichauf, & Winter, 2011; Buckl, Dierl, Matthes, & Schweda, 2010), or do not relate the components that promote the agility of the EAM function to the factors that promote the agility of the enterprise as a whole (e.g. Buckl et al., 2011; Buckl, Matthes, & Schweda, 2009). Also, there have been prior research efforts at developing agile approaches for EAM (Armour & Kaisler, 2001; Buckl et al., 2011; Pulkkinen & Hirvonen, 2005). However, they are restricted either to specific aspects of EA for example Enterprise IT Architecture (e.g. Armour & Kaisler, 2001) or the EAM function (e.g. Buckl et al., 2011; Pulkkinen & Hirvonen, 2005) without explicit focus on the agility of the enterprise as a whole.

This thesis aims at filling this gap in literature by developing an enterprise architecture management (EAM) method that enables the agility of the EAM function, and supports the agility of the enterprise as a whole. It was therefore imperative to investigate the factors that influence the agility of the enterprise as a whole. Based on a review of literature from manufacturing, where the concept of enterprise agility emanated, it was discovered that enterprise agility is influenced by three factors including *"Agility Drivers"*, *"Agility Capabilities"* and *"Agility Providers"* (see Vázquez-Bustelo, Avella, & Fernández, 2007; Yusuf, Sarhadi, & Gunasekaran, 1999; Zhengwen Zhang & Sharifi, 2007). The agility drivers are changes in the internal and external environment (e.g. changing customer demands) that forces the enterprise to change or to want to be agile (Sherehiy, Karwowski, & Layer, 2007); Agile capabilities (e.g. responsiveness and competency) provide the enterprise the strength to respond to changes taking place, including opportunities in the environment (Sharifi & Zhang, 1999; Z Zhang & Sharifi, 2000); and agility providers (e.g. relationships with partners, market sensitivity, and enterprise integration) are the means by which the agility capabilities can be achieved (Sharifi & Zhang, 1999; Tseng & Lin, 2011; Z Zhang & Sharifi, 2000). Therefore, for an EAM method to support the agility of the enterprise, it needs to provide mechanisms for sensing agility drivers, and strengthening agility providers to achieve agility capabilities with which the enterprise appropriately responds to the agility drivers. Such an EAM method is referred to in this thesis as an agile EAM method.

However, to support the agility of the EAM function and the agility of the enterprise as a whole, the agile EAM method needs to possess certain building blocks or components that enable it to do so. These building blocks or components are called in this thesis as the Essential Elements (EEs). These EEs were developed by: first, reviewing literature on enterprise agility, enterprise architecture, and a set of justification theories to form a theoretical foundation; second, using the theoretical foundation to adapt agile principles, and lean principles for EAM; and third, consolidating the adapted agile and lean principles into the Eight Essential Elements of Agile EAM Method. This approach was used in order to avoid the shortcomings associated with directly applying the agile principles (Abrahamsson, Conboy, & Wang, 2009; Boehm, 2002; Conboy, 2009) especially to fields outside systems development, and to provide the *"theoretical glue"*(Conboy, 2009; Whetten, 1989) that supports the applicability of these principles to EAM for enterprise wide impact, specifically enterprise agility. The essential elements were verified for *"relevance"* (Hevner, March, Park, & Ram, 2004; Peffers, Tuunanen, Rothenberger, & Chatterjee, 2007) through semi-structured interviews with 8 interviewees consisting of practitioners and researchers.

Based on these essential elements, the thesis proposed that the Agile EAM Method should act as a dynamic capability, and further developed mechanisms (e.g. for architecture decision making, change management, and governance) and processes (e.g. the EAM process) that enable it to support enterprise agility. In-line with the four types of dynamic capabilities proposed by Pavlou & El Sawy (2011), the Agile EAM Method provides the sensing capabilities, learning capabilities and coordinating capabilities needed to effectively carry out enterprise architecting as an integrating capability in transforming as-is architecture (existing operational capabilities) into to-be architecture (reconfigured operational capabilities). The structure of the Agile EAM Method consists of four enterprise levels (Enterprise, Operations, Business Process, and Business Activity) that correspond to EA components levels (Enterprise Architectures, Enterprise Capabilities, Enterprise Solutions, and Enterprise Features) and the EAM levels at which these EA components are produced (EA Strategy, EA Portfolio, EA Programme, and EA Projects). Within the method, each EA intervention is a targeted intervention (Pulkkinen & Hirvonen, 2005) triggered by a change driver (agility driver) and an architecture strategy towards either creating a new EA component, or revamping an existing one. EA component at a particular enterprise level is seen as an "IT-enabled resource" formed out of synergistic relationships amongst all the EA domain components at that level in a way that makes it have emergent capabilities that are more desirable than the individual capabilities of the EA domain components put together (Nevo & Wade, 2010). The thesis demonstrated the compatibility of the method with the Architecture Development Method (ADM) in TOGAF(2009) and evaluated the efficacy of the method using an illustrative scenario analysis base on a hypothetical case presented in Sessions (2007).

The main contributions of this thesis are the Agile EAM method, and the eight EEs of an agile EAM method. The thesis also contributes four agile EAM values, five agile EAM principles, five lean EAM principles, and seven lean EAM wastes, to the EA literature. Also, the thesis contributes a list of Critical Success Factors for EAM to the EA literature, and suggests areas for further research.

1.2 Motivation for the Thesis

The ability of the enterprise to keep up with changes in its environments; both internal and external, is necessary for its survival over the long-term. EAM has the ability to promote the agility of the enterprise. However, the extant literature on EAM methods either lack agility attributes, or do not explicitly relate to the factors that support the agility of the enterprise. The motivation for this thesis is to fill this gap in literature by designing an EAM method that provides mechanisms to support the agility of the EAM function and the agility of the enterprise as a whole.

1.3 Research Question

The main aim of the thesis is to develop an Agile Enterprise Architecture Management Method that enables the agility of the EAM function, and also supports the agility of the enterprise as a whole. In this regard, the thesis sought to answer three major questions:

- Firstly, what are the factors that influence the agility of the enterprise?
- Secondly, what essential elements or components should an enterprise architecture management method possess in order to support the agility of the EAM function and the agility of the enterprise as a whole?
- And lastly, how could these essential elements be conveyed in the form of a method?

1.4 Expected Outcome

It is envisaged that the thesis will produce one main design artifact (the Agile EAM Method) and intermediary artifacts (e.g. the Essential Elements of an agile EAM method). The Agile EAM Method is expected to support the agility of the EAM function and the agility of the enterprise as a whole. In practical terms, the Agile EAM Method should support the EAM function with mechanisms and processes that enable the EAM function to address the factors that influence the agility of the enterprise. Therefore, the main design artifact is expected to contribute to two main areas; EAM agility, and enterprise agility.

1.5 Research Method

This thesis seeks to develop an information systems design artifact in the form of a method (Gregor & Hevner, 2013; S. T. March & Smith, 1995; von Alan, March, Park, & Ram, 2004) that will support the agility of the Enterprise Architecture Management (EAM) function and the agility of the enterprise as a whole. In this regard, the study employs a research method that follows the guidelines for design science research methodology proposed by Peffers et al.(2007). The communication of the research process and findings in this thesis follows the guidelines provided by Gregor & Hevner(2013) based on Peffers et al.(2007).

The conceptual research model employed in this thesis is formed from the combination of "the Critical Success Chain" by Peffers et al. (2003) and "how project critical success factors (CSF) affect organizational benefit of enterprise systems (OBES)" by Zhong Liu & Seddon(2009). Peffers et al. (2003) proposed the critical success chain (CSC) as a means to link IS attributes to critical success factors (CSF) and CSF to organizational goals; whilst Zhong Liu & Seddon (2009) proposed that the effects of CSFs on enterprise benefit from enterprise system (ES) use are mediated by benefit drivers (outcome of the ES implementation processes). The two conceptual models were joined and adapted for EAM. The conceptual model for this thesis establishes a link from the Essential Elements (likened to IS/ES attributes) and CSF of EAM, and from CSFs through EAM mechanisms to enterprise benefits specifically, enterprise agility.

1.6 Findings

In accordance to the research question one, the thesis identified three factors that influence the agility of the enterprise; agility drivers, agility providers, and agility capabilities. To answer research question two, the thesis proposed eight EEs of an agile EAM method based on the review of literature on enterprise agility, EA, Lean and Agile principles and values, and other related theories (theoretical glue). Further, these EEs were conveyed in a form of a method called the agile EAMM, providing an answer to research question three. By means of an illustrative scenario, the thesis evaluated the efficacy of the agile EAMM to support the agility of the EAM function and that of the enterprise as a whole.

1.7 Limitations

The thesis has three main limitations. First, the interviewees engaged in the interview process employed in this thesis were mostly from the public sector in Finland, and therefore, the evaluation of the EEs lacked perspectives from the private sector. Second, the efficacy of the proposed agile EAMM still remains conceptual since the agile EAMM was not evaluated using a case study in a real enterprise. Third, the agile EAMM assumes that the adopting enterprise will support the collaborative nature of decision making that underpins the design of the mechanisms in the method. The adoption and adaption of the method will therefore be more beneficial in enterprises that have or are willing to have a collaborative approach to inter EA domain decision making processes.

1.8 Structure of the Rest of the Thesis

The rest of the thesis is outlined as follows: Chapter 2 presents literature review on enterprise agility to identify the factors that influence the agility of the enterprise (research question 1). It also presents a review of literature on enterprise architecture with focus on key issues including the relation between enterprise architecture, enterprise architecting and enterprise architecture management; organizational benefits of enterprise architecture, and challenges of EAM. Extant literature on EAM methods and EA are reviewed to provide a preliminary list of the EEs of EAM.

Chapter 3 is the second chapter on literature review and establishes the theoretical foundation needed to adapt the lean and agile principles and values for EAM. The chapter starts off by presenting the Lean and Agile principles and values from extant literature and suggesting some theories that could be employed to adapt the Lean and Agile principles and values to EAM. The suggested theories are then presented with concentration on how they relate to enterprise agility and its antecedents. Lastly, the Critical Success Factors (CSFs) of EAM that were identified through systematic literature review are also presented.

Chapter 4 communicates the research processes that are employed towards answering the research questions posed in chapter one. These include the research processes used for conducting the literature review, constructing and evaluating the design artifacts (the Essential Elements of an Agile EAM Method, and the Agile EAM Method), and representing and communicating the research process (writing the thesis). Further, the chapter provides justifications for the selection of the various methods; for instance, Design Science Research (DSR) Method, qualitative interview method, and illustrative Scenario as a DSR artifact evaluation method; employed during the research process. The Conceptual Model for the thesis, that explicates the relationship amongst all the various constructs developed in the thesis, is also presented.

Chapter 5 represents the first iteration in the DSRM process model adopted in this thesis. Hence, in this chapter, the theoretical foundations from Chapter 2 and Chapter 3 are put together to construct the Essential Elements of an Agile EAM Method (research question 2). Further the evaluation of the suggested EEs of an Agile EAM is presented in this chapter.

Chapter 6 presents the second iteration in the method design process. It is dedicated to the construction, demonstration and evaluation of the Agile EAM method in fulfillment of research question 3: "how could these essential elements be conveyed in the form of a method?" Guidelines for instantiating the Agile EAM method are provided in this chapter. Furthermore, the compatibility of the Agile EAM method with a widely used EA development standard (the TOGAF's ADM) is demonstrated in this chapter. Lastly, the chapter evaluates the efficacy of the method in supporting the agility of the EAM function and the agility of the enterprise as a whole.

Finally, Chapter 7 discusses the research process and findings, presents the limitations of the research, makes recommendations for further research and concludes the research.

2 CHAPTER 2 ENTERPRISE AGILITY AND ENTERPRISE ARCHITECTURE

This chapter presents the two major concepts of this thesis; enterprise agility and enterprise architecture. The aim of the enterprise agility subsection is to enable the author of the thesis acquire adequate knowledge of the area of enterprise agility, and lay the theoretical background towards answering research question 1. Consequently, the subsection on enterprise agility concentrates on finding the definition for enterprise agility from different perspectives including manufacturing, management, new product development, and information systems. Following this, the subsection on enterprise architecture seeks to establish sound understanding of the EA literature with particular focus on EA, Enterprise Architecting, and Enterprise Architecture Management and the relationship amongst these three; and other salient issues in EA especially EA governance, EA principles, EA planning, and EA frameworks, levels and domains. Prior works, especially within the academia, that are dedicated to developing an agile EAM method, an agile EA, or components (elements) for EAM are reviewed. This is done to develop a list of preliminary essential elements of an agile EAM method.

2.1 Enterprise agility

This sub-section is dedicated to answering the first research question: "what factors influence the agility of the enterprise?". The sub-section starts off by defining enterprise agility and investigating the types of enterprise agility. Then, relevant literatures that spell out the factors that influence the agility of the enterprise are reviewed. Finally, the sub-section looks at the relation between information systems and enterprise agility.

2.1.1 Definition of Enterprise Agility

There seems to be a convergence in literature on the source of "agility" as applied to the enterprise. Many authors (e.g. Vázquez-Bustelo et al., 2007; Yusuf et al., 1999; Zhengwen Zhang & Sharifi, 2007) asset that the concept of "agility" was first used in the 21st Century Manufacturing Enterprise Strategy report by Nagel and Dove (1991) to introduce the Agile Manufacturing paradigm. According to Zhang & Sharifi (2007, p. 352), Nagel and Dove (1991) defined agility as "a comprehensive response to the business challenges of profiting from rapidly changing, continually fragmenting, global markets for high quality, high performance, customer configured goods, and services".

Unlike its source, there seem to be less convergence in literature on the definition of agility. One explanation could be that agility is multifaceted and is therefore defined according to the context within which it is defined; for example the industry (Overby, Bharadwaj, & Sambamurthy, 2005). Definitions are crafted from various perspectives including manufacturing strategy (Zhengwen Zhang & Sharifi, 2007), production model (Vázquez-Bustelo et al., 2007), quick and effective reaction to environmental changes (H. Cho, Jung, & Kim, 1996), and the enterprise as a whole (Yusuf et al., 1999).

For instance, Quinn et al. define agile manufacturing as the "ability to accomplish rapid changeover from the assembly of one product to the assembly of a different product" (1996, p. 858). Also Zhang & Sharifi (2007, p. 352) define it as "a manufacturing strategy that aims to provide manufacturing enterprises with competitive capabilities to prosper from dynamic and continuous changes in the business environment, reactively or proactively." Others defined agility in terms of production model that integrate resources including technology, human resource and organization (e.g. Vázquez-Bustelo et al., 2007); and capability to survive and prosper in environment of continuous and unpredictable change (H. Cho et al., 1996). However, Yusuf et al. proposed a definition that provides a generic concept. They define agility as

> "the successful exploration of competitive bases (speed, flexibility, innovation proactivity, quality and profitability) through the integration of reconfigurable resources and best practices in a knowledge-rich environment to provide customer-driven products and services in a fast changing market environment" (1999, p. 37).

Within Information Systems (IS) discipline, the concept of "agility" was popularized by the "Agile Manifesto"(Alliance, 2001) that seek to promote the so-called "light-weight" IS development methods (Abrahamsson et al., 2009). As it is in manufacturing, "agility" in IS discipline is also multifaceted and researchers adopt different interpretation of agility (Abrahamsson et al., 2009). In addition to reasons in the manufacturing literature, one reason for this could be that the Agile Manifesto itself seems to be unclear on what constitutes agility. Another reason prevalent in literature is that, IS development environments are different and what constitutes agility might differ from one context to another (Lyytinen & Rose, 2006; Abrahamsson et al., 2009). For instance, Lee & Xia, (2010) defined software team agility as "the software team's capability to efficiently and effectively respond to and incorporate user requirement changes during the project life cycle". At the enterprise level, Overby et al.(2005) define enterprise agility as the ability of the firm to sense environmental change and respond appropriately. Also, based on D'Aveni (1994) and Goldman et al. (1995), Sambamurthy et al. (2003a) defined agility as the "ability to detect opportunities for innovation and seize those competitive market opportunities by assembling the requisite assets, knowledge, and relationships with speed and surprise" (2003a, p. 245).

The aforementioned definitions are context restricted, therefore, for this thesis, the author defines enterprise agility as "*the ability of the enterprise to quickly (i) sense relevant changes that affect its competitive bases (e.g. speed, flexibility, innovation proactivity, quality and profitability), and (ii) appropriately respond to such changes through the (iii) building or reconfiguration of requisite capabilities (including IT-enabled resources, practices, processes, and relationships) to remain innovative and competitive in a fast changing business environment*". The author of this thesis adopts the above definition because of the following reasons;

Firstly, the definition recognizes that different enterprises might have different competitive bases (Yusuf et al., 1999) for example speed, flexibility, integration and low complexity, innovation proactivity, responsiveness, quality, profitability and mobilization of core competencies (Yusuf et al., 1999; Sherehiy et al., 2007) with which the enterprise derives its agility.

Secondly, the definition recognizes the effect of sensing capabilities and responding capabilities on the agility of the enterprise (Nazir & Pinsonneault, 2012; Overby et al., 2005; Seo & La Paz, 2008), and that for a firm to achieve agility, it must possess and maintain a good balance between these two capabilities. For instance Overby et al., (2005) developed a framework for enterprise agility that suggests that firms with high sensing and high responding capabilities are agile. This assertion was later supported by an exploratory research conducted by Nazir & Pinsonneault, (2012). Furthermore, Seo & La Paz, (2008) assert that, if a firm is unable to integrate sensing, processing, and responding systems, the firm will have difficulties achieving agility. As implied in the definition, "capabilities" is used to mean high-level routines (S. G. Winter, 2003); formed partly on individual skills, tacit knowledge, social relations (Pandza, Horsburgh, Gorton, & Polajnar, 2003) and a combination of resources; that enable the organization to perform specific tasks repeatedly (Helfat & Winter, 2011).

Finally, the definition stresses the importance of the firm to screen changes that occur in the business environment for relevance (Seo & La Paz, 2008), and evaluate the appropriateness of response (Overby et al., 2005; Sharifi & Zhang, 1999; Tseng & Lin, 2011) based on the firm's competitive bases (Yusuf et al., 1999) and specific circumstances (Sharifi & Zhang, 1999). According to Overby et al., (2005), an appropriate response is supportive of a firm's goal. They argue that appropriate responses might include embarking on new venture (complex), adjusting existing venture (simple), or taking no action. An agile firm might react proactively (first mover) or reactively (fast follower) provided the response supports the firm's goal. (Overby et al., 2005.)

2.1.2 **Types of enterprise agility**

According to Lu & Ramamurthy, (2011) there are two types of enterprise agility: market capitalizing agility and operational adjustment agility. Market capitalizing agility enables the firm to quickly respond to and capitalize on changes through continuously monitoring and quickly improving product/service to address customers' need. Operational adjustment agility is the ability of the firm's internal processes to physically and rapidly cope with market or demand changes. Market capitalizing agility promotes entrepreneurial mindset about strategic direction, decision making, and adjustment in uncertain environment whilst operational adjustment agility focuses on speedy execution or implementation.

Also Sambamurthy et al. (2003a) suggest three types of enterprise agility namely Customer agility, Operational agility, and Partnering agility. Customer agility co-opts customers in the exploration and exploitation of opportunities for innovative and competitive actions. The customers can be co-opted as sources of innovation, co-creators of innovation and idea testers or promoters. Partnering agility refers to the firm's ability to leverage assets, knowledge, and competencies of other firms (including suppliers, distributors, contract manufacturers, and logistics providers) in exploration and exploitation of innovation opportunities. Operational agility focuses of achieving speed, accuracy, and cost economy in the exploitation of innovation opportunities. They argue, that customer agility, partnering agility, and operational agility collectively reflects the agility of the firm.

In essence, enterprise agility can be classified into two broad types; external agility and internal agility. External agility refers to the ability of the firm to sense and appropriately respond to strategic issues for example changes in the market (such as customer requirements and competitor actions), changes in technology, and changes in business relations and ecosystem structures. Internal agility refers to the ability of the firm to reconfigure its internal operations (or processes), management structures, and resources to support and sustain its strategy. Internal agility is therefore inside facing; it supports and sustains external agility which is outside facing. Thus in line with Overby et al., (2005), enterprise agility applies to both strategic and operational issues.

2.1.3 **Agility Drivers, Agility Capabilities and Agility Providers**

Considerable amount of research has been done to understand the drivers of agility, the capabilities that enterprises need to poses in order to be agile, and the providers of such agility(Sharifi & Zhang, 1999; Tseng & Lin, 2011; Yusuf et al., 1999; Zhengwen Zhang & Sharifi, 2007). Agility drivers, capabilities and providers are summarized in Table 1 below, and briefly discussed in subsections below.

I. Agility Drivers

The primary driver of enterprise agility is change (Yusuf et al., 1999; Tseng & Lin, 2011). Agility drivers are the changes that occur in the business environment to which the enterprise must respond or that forces the enterprise to want to be agile (Sherehiy et al., 2007). These changes are specific to different enterprises, and their relevance is dependent on the specific enterprise, its state or situations (Sharifi & Zhang, 1999). Agility drivers include, change in customer's requirements, change in competition criteria, change in markets, change in technological innovations, and change in social factors(Sharifi & Zhang, 1999; Tseng & Lin, 2011; Zhengwen Zhang & Sharifi, 2007).

II. Enterprise Agility Capabilities

Agile capabilities provide the enterprise the strength to respond to changes taking place, including opportunities in the environment (Sharifi & Zhang, 1999; Zhengwen Zhang & Sharifi, 2000). These capabilities consist of four elements; responsiveness, competency, flexibility/adaptability, and quickness or speed (Sharifi & Zhang, 1999; Tseng & Lin, 2011; Zhengwen Zhang & Sharifi, 2000). Responsiveness is the ability to identify change and respond quickly, reactively or proactively and to recover from the organizational change caused by the response; competency is a measure of how efficiently and effectively an enterprise realizes its goals and objectives; flexibility or adaptiveness is the ability of the enterprise to use the same facility to implement different processes and achieve different goals; and quickness or speed is the ability to complete an activity in the shortest possible time. Together, these four attributes shape the agility capabilities of the enterprise.

III. Enterprise Agility Providers

Agility providers are the means by which the agility capabilities can be achieved. These providers could come from the organization, people, technology and innovation; and integrated together using information systems and technology (Sharifi & Zhang, 1999; Tseng & Lin, 2011; Zhengwen Zhang & Sharifi, 2000). As illustrated in Table 1 below, agility provider include building relationships with partners, customers and competitors; technology; customer and market sensitivity; integration; and information systems.

Table 1: Agility Drivers, Agility Capabilities and Agility Providers

Source	Agility Driver	Agility Capabilities	Agility Provider
Sharifi and Zhang (1999)	• Change in Market • Changes in Competition criteria • Changes in customer requirement • Changes in Technology • Changes in social factors	• Responsiveness • Competency • Flexibility • Quickness	
Yusuf et al (1999)	• Automation and price/cost consideration • Widening customer choice and expectation • Competing priorities • Integration and proactivity • Achieving manufacturing requirements in synergy	• Core competence management • Virtual enterprise • Capability for re-configuration • Knowledge-driven enterprise	• Integration • Competence • Team building • Technology • Quality • Change • Partnership • Market • Education • Welfare
Zhang & Sharifi (2007)	• Change in market place • Change in competition basis • Change in customer requirements • Change in technology • Change in social factors • Internal drivers	• Proactiveness • Responsiveness • Competency • Flexibility • Quickness • Customer focus • Partnership	• Relationship with supplier/customer/competitors • Technology • Integration • Organization • People • Innovation • Information Systems
Tseng & Lin (2011)	• Customer's requirements • Competition criteria • Markets • Technological innovations • Social factors	• Responsiveness • Competency • Flexibility • Robustness	• Collaborative relationships • Process integration • Customer/marketing sensitivity

2.1.4 **Information Systems and Enterprise Agility**

The relationship between Information Systems and Enterprise Agility has been researched extensively in the Information Systems community. The relationship has been investigated from various perspective, for example: information systems alignment and enterprise agility (e.g. Tallon & Pinsonneault, 2011); IT capability and organizational agility (Lu & Ramamurthy, 2011; V Sambamurthy, Wei, Lim, & Lee, 2007), and IT and digital options and enterprise agility (Overby et al., 2005).

According to Lu & Ramamurthy (2011), information system is widely regarded as an enabler of enterprise agility. These information technologies could be in the form of processes, knowledge, and communication technologies (Vallabh Sambamurthy et al., 2003a). From an empirical study involving executives from 128 organizations, Lu & Ramamurthy (2011) discovered a significant positive relationship between IT capabilities and organizational agility; both operational adjustment agility and market capitalization agility. Based on Sambamurthy and Zmud (1997), Lu & Ramamurthy defined IT capabilities as "a firm's ability to acquire, deploy, combine, and reconfigure IT resources in support and enhancement of business strategies and work processes" (2011, p. 932). Similarly Sambamurthy et al.,(2007) proposed the role of complementary relationships between IT capabilities and operational capabilities in creating two types of organizational agility: entrepreneurial agility, and adaptive agility. Also, Nevo & Wade (2010) argue that the key factor in enabling strategic execution is the emergent capability that ensues from the synergistic combination of IT assets and organizational resources to form IT-enabled resources, and not the individual capabilities of IT assets and organizational resources taken in isolation.

Although Henderson & Venkatraman (1993) argue that lack of alignment between business and IT strategies of organizations contributes to the inability of the organization to realize value from IT investments, there seem to be split views on the relationship between IT (especially IT alignment) and enterprise agility in the IS literature (e.g. see Lu & Ramamurthy, 2011; Tallon & Pinsonneault, 2011). A recent research by Tallon & Pinsonneault (2011) to investigate the competing view on the relationship between IT and enterprise agility actually revealed a positively significant relationship between alignment and agility, and between alignment and performance that is fully mediated by agility. However, they also found that, in volatile environments IT infrastructure flexibility moderates the link between alignment and agility. Byrd & Turner define IT infrastructure flexibility as the;

> "ability to easily and readily diffuse or support a wide variety of hardware, software, communications technologies, data, core applications, skills and competencies, commitments, and values within the technical physical base and the human component of the existing IT infrastructure" (2000, p. 172).

Furthermore, according to Nazir & Pinsonneault (2012), an enterprise could have two forms of electronic integration; Internal Electronic Integration (IEI) and External Electronic Integration (EEI). IEI links, and coordinates the links between units within the firm to make them more adaptive to one another whilst EEI fosters connection with business partners by improving environmental scanning through probing, exploring and appropriating new knowledge. In that regard, high EEI is likely to influence high sensing capability through knowledge exploration, and high IEI is likely to influence high response capability through know exploitation, therefore, Nazir & Pinsonneault (2012) concluded, in line with Overby et al.(2005)'s framework, that for an enterprise to be agile, it must poses high EEI and high IEI. Knowledge exploration relates to acquiring new knowledge from the environment, whilst knowledge exploitation relates to using and sharing existing knowledge with the firm.

2.2 Enterprise Architecture

In this sub-section, the concentration is turned to Enterprise Architecture (EA) and related issues. There major concepts; enterprise architecture, enterprise architecting, and enterprise architecture management are presented, and the relationships amongst the three is established. Then other important issues relating to the three major concepts are presented. These include organizational benefits of EA, challenges of EAM, EA frameworks, EA domains and levels, and EA principles, planning, and governance. Literature on extant methods for agile EAM, EAM, agile EA, and components of EAM are reviewed to establish preliminary lists of the Essential Elements of an Agile EAM.

2.2.1 Definition of Enterprise Architecture

Architecture is defined as "the fundamental organization of a system, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution" (ISO/IEC 42010:2007). In TOGAF, architecture is defined from two perspectives. Firstly, architecture is "a formal description of a system, or a detailed plan of the system at component level to guide its implementation" and secondly, architecture is "the structure of components, their inter-relationships, and the principles and guidelines governing their design and evolution over time". (2009, p. 9.) Enterprise architecture is defined by Rood as "a conceptual framework that describes how an enterprise is constructed by defining its primary components and the relationships among these components" (1994, p. 106). EA defines a holistic view of the enterprise instead of taking application by application view (Kaisler et al., 2005). Sessions (2007) describes the EA as the architecture of the whole enterprise consisting of business process, technologies, and information systems.

According to TOGAF, apart from the holistic nature of EA described above, the enterprise architecture can also be represented with specific focus, for example on capabilities (capability architecture) or on enterprise segments (segment architecture). Furthermore, the architecture could also be described to represent different instances in time for instance; "as-is" or baseline architecture that describes the current state of the architecture; and the "to-be" or ultimate end state architecture that describe how the enterprise architecture should be at a particular instance in the future. In between these two extremes is the "transition architecture" with its own target state within a shorter instance in the future. (2009, p. 62).

2.2.2 **Enterprise Architecting**

Enterprise Architecture efforts consist of three value dimensions; "architect, process, and product"(Boster, Liu, & Thomas, 2000). The "architect" is the person – and by extension, the group of people – who engages in the enterprise architecture endeavor; the "process" depicts series of technical and business related activities that the "architect" employs to produce the "product"; and the "product" is the end result and includes the architecture itself together with associated products including architectural drawings and models. The "process" is called enterprise architecting. This is in line with the assertion by Jonkers et al. that "architecture is a process as well as a product" (2006, p. 64). Boster, Liu, & Thomas assert that architecting is "an ongoing, iterative, technical and political process" (2000, p. 46). The ISO/IEC 42010 defines architecting as a "set of interrelated activities of conceiving, defining, describing, documenting, maintaining, improving, and certifying proper implementation of an architecture throughout a system's lifecycle" (2007).

In this thesis, the author replaces "systems" in the ISO/IEC 42010 (2007) definition with "enterprise", and merges it with the assertions by Boster, Liu, & Thomas (2000) to read: *"Enterprise Architecting is a set of ongoing, iterative and interrelated technical and business activities of conceiving, defining, describing, documenting, maintaining, and certifying proper implementation of an architecture throughout an enterprise's lifecycle"*. This definition of enterprise architecting implicitly encompasses all components that make up the enterprise.

Nevertheless, there are other definitions that are domain focused. For instance, Kaisler et al. define Enterprise Architecting as "the set of processes, tools, and structures necessary to implement an enterprise-wide coherent and consistent IT architecture for supporting the enterprise's business operations" (2005, p. 1). The definition adopted by this thesis also takes note of the fact that enterprise architecting is a continuous process. For example, according to Luche et al.(2012), Op't Land et al.(2009) defined enterprise architecting as:

> "..a continuous process involving the creation, modification, enforcement, application, and dissemination of different results. This process should be in sync with developments in the

> environment of the enterprise as well as developments internal to the enterprise, including both its strategy and its operational processes" (2012, p. 2).

In essence, enterprise architecting is a continuous process that transforms the "as-is" architecture to the "to-be" architecture within the context of the enterprise.

2.2.3 **Enterprise Architecture Management**

Enterprise Architecture Management (EAM) can be conceived as set of management activities that are geared towards the establishment and continuous development of an Enterprise Architecture in a way that enables it to control business change from architecture perspective (Aier, Gleichauf, et al., 2011). Ahlemann, Stettiner, Messerschmidt, & Legner (2012) define EAM as

> "a management practice that establishes, maintains and uses a coherent set of guidelines, architecture principles and governance regimes that provide direction and practical help in the design and development of an enterprise's architecture to achieve its vision and strategy" (2012, p. 3).

The EAM function has taken on a strategic role and presents a collection of processes, methods, tools and responsibilities needed to continually align business and information systems in support of enterprise strategies and visions (Ahlemann et al., 2012, p. 8; Simon, Fischbach, & Schoder, 2014). In that regard, EAM is no longer viewed as an IT department job but a strategic function (Ahlemann et al., 2012, p. 15) that approaches enterprise related changes in a holistic and consistent way taking into account environmental influences including markets, regulations, or industry standards (Buckl et al., 2009). It supports enterprise transformation in response to the increasingly dynamic enterprise environment (Ahlemann et al., 2012, p. 3; Buckl et al., 2010), and sustains the ensuing corporate changes (Ahlemann et al., 2012, p. 57).

In essence, Enterprise Architecture Management can be viewed as the managerial endeavor that provides the necessary organizational context to enable enterprise architecting efforts to continuously transform the "as-is" architecture towards a "to-be" state in response to the increasingly dynamic enterprise environment. The environmental stimuli are those agility drivers to which it is necessary for the enterprise to respond to appropriately. EAM provide the organizational context comprising of processes, tools, structures, resources, relationships, and critical success factors that promote the success of the architecting endeavor; and the stimuli that triggers the architecting process. The relationship explicated here is illustrated in the Figure 1 below.

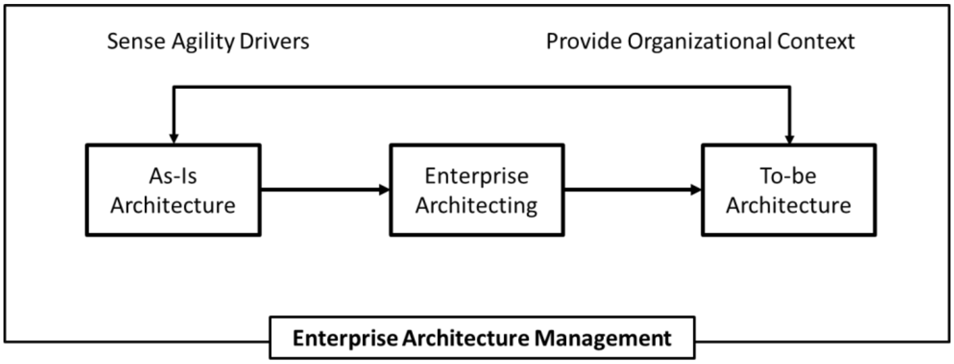


Figure 1: EA, Enterprise Architecting, and EAM

2.2.4 **Organizational Benefits of Enterprise Architecture**

Different authors from the academia (e.g. Richardson, Jackson, & Dickson, 1990; Pulkkinen, 2006) and industry (e.g. TOGAF, 2009) have made claims about the enterprise or organizational benefits of enterprise architecture. This section piggybacks on previous publications that compiled these claims on the benefit of EA to the enterprise (e.g. Niemi, 2008) and/or investigated how EA influences enterprise benefits (e.g. Tamm, Seddon, Shanks, & Reynolds, 2011). Table 2 summarizes the literature sources and the enterprise benefits of EA. Notable amongst these benefits are business-IT alignment (Wan, Luo, Johansson, & Chen, 2013; Tamm et al., 2011; Bradley, Pratt, Byrd, Outlay, & Wynn Jr, 2012; Niemi, 2008); improved communication and collaboration (Niemi, 2008; Tamm et al., 2011); reduced costs (Niemi, 2008; Foorthuis et al., 2010; Espinosa et al., 2011; Tamm et al., 2011); improved change and risk management (Niemi, 2008; Foorthuis et al., 2010; Tamm et al., 2011); and increased interoperability and integration (Niemi, 2008; Espinosa et al., 2011; Foorthuis et al., 2010; Tamm et al., 2011; Wan et al., 2013). Espinosa et al.(2011) assert that enterprise architecture benefits can manifest as EA technical benefits (e.g. reduced redundancy); and EA business process benefits (e.g. increased automation) and EA organizational benefit (e.g. increased productivity and revenue growth). Foorthuis et al.(2010) claim that EA facilitates the achievement of key business goals and the management of organizational complexity. Eighteen (18) out of thirty-three (33) sources analyzed by Niemi (2008) support the claim that enterprise architecture help provide a holistic view of the enterprise. Similarly, Wan et al.(2013) claim that EA promotes a common and integrated understanding of the enterprise, and improves decision making. Indeed, they classify "better decision making" as a strongly desirable and a strongly realizable benefit of EA. EA is also believed to improve agility at the technical level (Espinosa et al., 2011), the business process level (Espinosa et al., 2011), the strategic level (Niemi, 2008), and the enterprise level (Foorthuis et al., 2010; Espinosa et al., 2011; Bradley et al., 2012).

However, an Enterprise Architecture Benefit Model (EABM) proposed by Tamm et al. (2011) suggests that, EA has no direct effect on organization benefit, and that it is through the impact of EA on "benefit enablers" that EA leads to organizational benefit.

The benefit enablers include organizational alignment, information availability, resource portfolio optimization, and resource complementarity. Similarly, an empirical research by Bradley et al. (2012) shows no direct effect of EA maturity on enterprise agility. According to Bradley et al. (2012), the effect of EA maturity on enterprise agility is mediated by IT alignment and/or operational IT effectiveness.

From the above and as illustrated in Table 2, there seem to be convergence in the EA literature on the benefits of EA to the organization. However, most of the benefits that are claimed in the EA literature lack theoretical backing or empirical verification (Tamm et al., 2011). For example, out of the 50 research articles that were analyzed by Tamm et al. (2011) for the organizational benefits of EA, only six (6) (e.g. Richardson et al., 1990; Pulkkinen, 2006) had empirical support for their claims, and only four (e.g. Lindström, 2006) provided references to support their claims (see Appendix 1 Table A1 of Tamm et al., 2011).

Table 2: Organizational Benefits of EA

Source	Benefits
Tamm, Seddon, Shanks, & Reynolds (2011)	Increased responsiveness and guidance to change; improved decision-making; improved communication & collaboration; reduced (IT) costs; business-IT alignment; improved business processes; improved IT systems; re-use of resources; improve integration; reduce risk; regulatory compliance; provides stability; improved interoperability between ISs; improved utilization of IT; effective utilization of IT resources; responsiveness to change; improves information sharing; assists with organizational governance; improves ROI from IT spending; less wasted time/money on projects which do not support business goals; improves IS security; reduces IT complexity; reduces organizational stovepipes; faster development and implementation of new IS; standardizes organizational performance measures; higher business and process flexibility; improve customer satisfaction; enabling business and process change; faster, simpler and cheaper procurement; alignment enabler; change enabler; and reduced time-to-market
Foorthuis et al. (2010), and Foorthuis et al. (2015)	EA enables management to achieve key business goals; enables management of organizational complexity; facilitates the integration, standardization and duplication of processes and systems; enables the enterprise to deal with its environment effectively; yields insights into the current situation; yields insight into the future situation; reduces project costs and project duration; reduces project risk and improves project success; and enables projects to manage complexity.
Espinosa, Boh, & DeLone (2011)	EA technical benefits (reduce redundancy, improved integration, reduced cost, greater agility, increased reuse, and standardization); EA business process benefits (increased automation, increased integration, reduced redundancy, increased modularity, and greater agility); and organizational impact (increased productivity, better organizational agility, improved process timeliness, cost reduction, and revenue growth).
Bradley, Pratt, Byrd,	IT alignment, operational IT effectiveness; enterprise agility (mediated by

Outlay, & Wynn Jr (2012)	IT alignment and operational IT effectiveness)
Wan, Luo, Johansson, & Chen (2013)	Improved business-IT alignment; common and integrated understanding of the enterprise; better decision making; reduced complexity; improved business structure; improved integration and interoperability; resource optimization; and financial & economic results (monetary benefits, effectiveness-related benefits, and timeliness-related benefits)
Niemi (2008)	Reduced costs; provide a holistic view of the enterprise; improved business-IT alignment; improved change management; improved risk management; increased interoperability and integration; shortened cycle times; improved communication; improved management of IT investment; increased reusability; improved alignment to business strategy; increased standardization; reduced complexity; improved strategic agility; improved innovation; improved alignment with partners; increased quality; improved customer orientation; improved decision making; increased efficiency; improved staff management; improved asset management; improved business processes; evolutionary EA development & governance; increased economy of scale; increased stability; and increased market value.

2.2.5 **Challenges of Enterprise Architecture Management**

Challenges in Enterprise Architecting have had appreciable consideration in the EA literature (Bente, Bombosch, & Langade, 2012; Buckl et al., 2011; Chuang & van Loggerenberg, 2010; Kaisler et al., 2005; Lucke et al., 2010). There are technical and non-technical problems relating to Enterprise Architecture Management (EAM), however the latter are predominant (Chuang & van Loggerenberg, 2010; Kaisler et al., 2005; Lucke et al., 2010). Technical challenges in EAM include difficulty in appropriately communicating architectural materials (Lucke et al., 2010) to multiple stakeholder in a way that every stakeholder can understand (Kaisler et al., 2005); lack of adequate modeling tools and EA frameworks (Chuang & van Loggerenberg, 2010; Kaisler et al., 2005; Lucke et al., 2010); difficulties in integration especially with legacy systems (Chuang & van Loggerenberg, 2010); coping with complexities and challenges relating to the assessment of technical architecture maturity, systems scalability, and architecture integrity (Kaisler et al., 2005).

Non-technical challenges in enterprise architecting can be summarized under issues relating to "understanding and management of the enterprise architecture" (Lucke et al., 2010, p. 9). Lucke et al., (2010) identified two broad categories of problems relating to understanding and managing enterprise architecture: they are management problems and semantic problems. The former consists of challenges relating to management commitment, EA governance, stakeholders and coordination; and the latter consists of challenges relating to communication, understanding requirements, and shared understanding. Other researchers (for example, Kaisler et al., 2005; Chuang & van Loggerenberg, 2010; Buckl et al., 2011) share similar views. For instance, Kaisler et al., (2005) mentioned challenge relating to managing stakeholder perspectives, and Chuang & van Loggerenberg, (2010) list communication, obtaining buy-in from EA stakeholders, ownership, perceptions of the enterprise architect, and organizational politics as challenges to EAM. Likewise, Buckl et al.,(2011) communicate challenges relating to alignment of stakeholder interests, and commitment and involvement of all parties as challenges of EAM. Lack of management commitment deprives the architecting process of resources needed to perform, and consequently results in low acceptance of EA within the organization (Buckl et al., 2011). Another challenge that is closely associated to the lack of management commitment is the inability to development meaningful value proposition for enterprise architecture (Kaisler et al., 2005; Lucke et al., 2010; Buckl et al., 2011).

Challenges related to EA governance stem from the lack of structure in EA projects caused by ill-defined roles, responsibilities, processes and procedures (Buckl et al., 2011). Stakeholders and coordination presents another area of challenge to EAM in that architectural projects usually involve multiple stakeholders, with divers and sometimes conflicting views and perspective (Kaisler et al., 2005; Lucke et al., 2010; Buckl et al., 2011). These stakeholders can be dispersed across multiple organizational units separated by geography and/or time (Lucke et al., 2010), and do have different "languages" they understand. For instance, the language barriers pose communication challenges in EAM. The group that develops the EA might be different from the group that uses it (Lucke et al., 2010; Buckl et al., 2011). An empirical study conducted by Chuang & van Loggerenberg, (2010) further reveals that, beyond communication gaps amongst different stakeholder groups, there exist communication challenges within stakeholder groups. The other two semantic problems relating to understanding requirements and shared understanding are off-shoots of challenges relating to communication.

2.2.6 **Enterprise Architecture Frameworks**

According to Sessions (2007), an Enterprise Architecture Framework (EAF) is "a skeletal structure that defines suggested architectural artifacts, describes how those artifacts are related to each other, and provides generic definitions for what those artifacts might look like". EAF can be used to develop different architectures, and should provide a method for describing the end state of the enterprise in terms of a set of building blocks (architecture components) and the relationships between such components (TOGAF, 2009, p. 7). TOGAF further recommends that an EAF should provide a set of tools, a common vocabulary, a list of recommended standards, and compliant products that can be used in implementing the components (2009, p. 7). In essence, EAFs should consist of a description of deliverables that should be produced and a method by which the deliverables should be produced, however some EAFs fall short of one of these two requirements; usually the later (TOGAF, 2009, p. 18). For instance, Sessions (2007) does not consider the Zackman Framework as an EAF because though the Zackman Framework provides vivid description of architectural deliverables, it does not describe a method for developing such deliverables. Also, TOGAF provides detail description of a method

– the Architecture Development Method ADM – but does not adequately describe the architectural deliverables that should be produced (Sessions, 2007).

Apart from the two EAFs mentioned above as examples, there are other EAFs with different scope of coverage. Whilst some are domain specific, others cover the whole enterprise. They include, 4+1 View, Federal Enterprise Architecture Framework (FEAF), Treasury Enterprise Architecture Framework (TEAF), Department of Defense Architecture Framework (DoDAF), Architecture of Integrated Information Systems (ARIS), Model Driven Architecture (MDA), and LTG Architecture Framework (LTGAF).

The aim here is not to present all these EAFs but to point to sources where they are presented and compared. Table 3 below provides a summary of the sources, basis of analysis or comparison, and the frameworks that were analyzed.

Table 3: Comparison of EA Frameworks

Source	**Basis for analysis**	**Frameworks analyzed**
Tang, Han, & Chen, (2004)	Roles in systems and software engineering (Inputs, outcomes, and goals)	Zackman, 4+1 View, FEAF, RM-ODP TOGAF, and DoDAF
Urbaczewski & Mrdalj, (2006)	Context of use Views, level of abstraction, and coverage of the SDLC	Zackman, DoDAF, FEAF, and TOGAF
Leist & Zellner, (2006)	Requisite requirements for EAF usefulness (Specification document, Meta model, role, techniques, and procedure model)	ARIS, DoDAF, FEAF, TEAF, TOGAF, Zackman, and MDA
Sessions (2007)	Elements of EAFs (Taxonomy completeness, process completeness, reference-model guidance, practice guidance, maturity model, business focus, governance guidance, partitioning catalog, vendor neutrality, information availability, time to value)	Zackman, TOGAF, FEA and Gartner
Lim, Lee, & Park, (2009)	EA quality attributes grouped into Views, Perspectives, Scope, and Time	Zackman, TEAF, DoDAF, FEAF, and LTGAF

2.2.7 **Enterprise Architecture Domains and Levels**

Enterprise architecture comprises of different aspects or subsets, that are referred to by TOGAF as “architecture domains” (2009, p. 10). However, there seems to be no consensus on neither what these architecture domains are nor on the terminology used to refer to them. For example, concerning the later, Tang et al. (2004) uses the term “viewpoints”; Hirvonen & Pulkkinen (2004), Pulkkinen & Hirvonen (2005), and Pulkkinen (2006) use the term ”architecture dimensions”; Winter & Fischer(2006) use the term ”Architecture layer”; whilst other authors including Jonkers et al.(2003), Hoogervorst (2004), Iyer & Gottlieb(2004), and TOGAF (2009) use the term ”architecture domain”. Not only are these terminologies different, but also different authors use them to refer to different concepts with regards to enterprise architecture. For instance, whilst Tang et al. (2004) uses "viewpoints" to refer to the different aspects of enterprise architecture, TOGAF (2009) uses "viewpoints" to refer to a model (2009, p. 416) that describes the perspective from which stakeholders from the different aspects of the enterprise architecture (architecture domains) view the architecture (2009, p. 40,413). Also, whereas TOGAF (2009) treats data and application architectures as EA domains, Winter & Fischer(2006) treat them as views of software architecture. In this thesis however, the term "Architecture domain" or "EA domain" is used to refer to the different aspects of the enterprise architecture.

As illustrated in Table 4 below, different authors have proposed different set of EA domains. The divergence in views amongst authors is however not different from what prevails in practice. For instance, all four case companies in Winter & Fischer(2006) had different set of EA domains. From EA literature, it is unclear what factors underpin the differences in the EA domains proposed by researchers, or the differences in the EA domains that exist in enterprises. However, it is not entirely unsafe to explain the later using contingency and fitness theories. Each enterprise may choose to use different structure, and organize the enterprise components in a way that enables the enterprise to establish fit between its environment and strategy, and between its strategy and internal operations thus attaining external and internal fit. Therefore, it is likely that the differences in the EA domains adopted across enterprises are informed by contingencies including the industry type, environmental factors, strategy, organizational structure, maturity, size, and most importantly, the managerial perception of the relationship between the different components of the enterprise (e.g. how business and IS components relate). Also, the different view that authors hold can be attributed to their backgrounds, base literature, and probably the EA domains that have been adopted in their case companies. For instance, Winter & Fischer(2006) might have referred to security and information architectures as examples of cross-layer views because in one of their case companies (case company D), security architecture and data architecture were treated as cross-cutting issues in business, application and technical/IT architectures.

In view of the above, the author of this thesis uses a generic representation of EA domains (EA Domain1, EA Domain2 … EA Domain n) to allow for easy adaption and instantiation of the ensuing agile EA management method within enterprises. However, for ease of representation and illustrations, the EA domains proposed by Hirvonen & Pulkkinen (2004) are used at some points within the thesis.

The notion of enterprise architecture levels presents another area of gross disparity amongst researchers. Jonkers et al.,(2003) proposes business, application and technology levels (layers) of EA; Pulkkinen (2006) states enterprise, domain and system levels of EA; and TOGAF (2009) proses enterprise, segment, and capability levels of EA.

Table 4: EA Domains and Levels

Source	**EA Domains**
Iyer & Gottlieb,(2004)	Process, information/knowledge, infrastructure, and organization

Jonkers et al.,(2006) and Lankhorst (2004)	Process, product, information, application, and technical
Hoogervorst,(2004)	Business, organizational, information, and technology
Jonkers et al.,(2003)	Process, product, organization, information, data, application, and technical infrastructure
Hirvonen & Pulkkinen, (2004) and then later by Pulkkinen & Hirvonen, (2005) and Pulkkinen,(2006)	Business, information, system (application), technology
Winter & Fischer,(2006)	Business, process, integration, software, and infrastructure
TOGAF (2009, p. 61)	Business, data, application, technology
Tang et al. (2004)	Business, information, software, and technical

2.2.8 **Enterprise Architecture Principles**

Enterprise Architecture (EA) principles are the basic philosophies that an enterprise uses to guide the development, evaluation and evolution of its architecture (Richardson et al., 1990). Indeed, EA principles are fundamental to the design, development and evolution of enterprise architecture, as explicitly stated in a widely adopted definition of EA; "…and the principles guiding its design and evolution"(see ISO/IEC 42010:2007, 2007; TOGAF, 2009). There seem to be a consensus in EA literature on the importance of EA principles, for example to the success of EA management and governance, and attainment of Business and IT goals (e.g. Aier, Fischer, & Winter, 2011; C. Fischer, Winter, & Aier, 2010; Richardson et al., 1990; Stelzer, 2010; TOGAF, 2009; R. Winter & Aier, 2011). However, the definition, and the mechanisms for the development, the enactment and the enforcement of EA principles have been sparingly studied. In recent years, however, some authors have shown interest in studying EA principles as presented below.

EA principles can be categorized into two main forms following the definition of EA; representation principles and design principles (Stelzer, 2010). According to Stelzer,(2010) representation principles are employed during the description and modeling of EA, and the evaluation of EA representation. Design principles, on the other hand, guide the construction and evaluation of the enterprise architecture. Representation principles include understandability, consistency, and completeness; whilst design principles include separation of concerns, modularity, or loose coupling. Similarly, according to TOGAF, architecture principles can be categorized into two types; EA principles that govern the EA process, providing guidance for development, maintenance, and EA use; and EA principles that govern EA implementation, providing tenets and guidance for designing and developing information systems (2009, p. 265).

Regarding the qualities of EA principles, TOGAF recommends that EA principles should be understandable, robust, complete, consistent and stable (2009, pp. 267–268). Lindström (2006) argues that EA principles are essentially enterprise wide requirements and posits the application of qualities of requirements to EA principles. Accordingly, Lindström (2006) classifies qualities of EA principles into semantic and syntax qualities. The former includes stability, verifiability, modifiability, correctness and completeness, whilst the later includes consistency, verifiability, unambiguity, and modifiability. Furthermore, there are suggestions in EA literature on the composition or format of EA principles. Richardson et al.(1990) and TOGAF (2009, p. 266) recommend that each EA principle consist of a name, a principle statement, a rational for the principle and the implication of the principle. Lindström (2006) recommends that apart from the name, the statement, the motivation (rational), and the implication, each principle should consist of a "measure" that states how the fulfillment of the principle will be measured; and a "comment" if necessary.

There is however, gross disparity in literature as to where EA principles fit in the enterprise decision making process. For instance, TOGAF states that EA principles are a "subset of IT principles"(TOGAF, 2009, p. 265). TOGAF holds this view probably because of two reasons. Firstly, it recommends a governance mechanism that follows the following hierarchy: corporate governance, technology governance, IT governance and architecture governance; subsuming architecture governance under IT governance (2009, p. 671). Secondly, TOGAF over concentrates architecture principles on capturing "…fundamental truths about how the enterprise will use and deploy IT resources and assets"(2009, p. 268). Indeed TOGAF recommended "Business Principles" as part of its twenty sample principles (2009, pp. 269–273). However, it is arguable that the recommended business principles are more related to information management than to core business resources including business processes and capabilities. Based on Broadbent & Kitzis (2005), Lindström (2006) proposes that business strategy informs business principles; business principles inform architectural principles; architectural principles inform IT governance; and IT governance informs IT strategy.

EA principle development process is another area that is gaining discourse in literature. For instance, Nabukenya, Bommel, & Proper (2006) present a systematic policy (EA principles) development process based on collaborative engineering literature and verified using case studies; Op't Land & Proper (2007) present an EA principle development process by Op't Land(2005); and Fischer et al. (2010) present an example from industry – the EA principle development process of European Transportation Company (ECT). The EA development process is of importance because it ensures that all relevant principles are defined and do represent the top issues of the enterprise (Lindström, 2006). To enable adequate participation of relevant stakeholders in the development of EA principles, the process will have to be anchored at the level that warrants the participation of such stakeholders. TOGAF notes that EA principles are developed by a lead architect in conjunction with the CIO, an architecture board and other business stakeholders (TOGAF, 2009, p. 267).

2.2.9 **Enterprise Architecture Governance**

Generally, architecture governance provides a set of mechanism through which architecture is enacted in the enterprise (Aziz, Obitz, Modi, & Sarkar, 2005). Specifically, EA governance provides mechanisms by which EA and other architectures are managed at the enterprise level (TOGAF, 2009, p. 671). These mechanisms could include control processes, rules, standards, roles and organizational structure (Kamogawa & Okada, 2005) to ensure the successful definition, development, integration, and management of architecture content, and the measurement of the effectiveness of the EA discipline within the enterprise (Aziz et al., 2005). EA governance is expected to act as "glue" between the involved EA domains to ensure that ensuing mismatches are handled effectively (Kluge, Dietzsch, & Rosemann, 2006). According to TOGAF, EA governance concerns managing change processes (design governance) (2009, p. 25); defining operational performance levels, managing system performance against defined performance levels, and implementing systems that ensures effective operations of systems (operational governance) (2009, p. 716); and providing architectural oversight over architectural implementation (implementation governance) (2009, p. 493). In this regard, EA governance is implemented at various levels of the enterprise (Ylimäki, 2008a), and is necessary for long term maintenance of the enterprise architecture (Kamogawa & Okada, 2005). TOGAF recommends that EA strategic plans should be closely linked to the architecture governance process to curb circumvention of design decisions for tactical convenience (2009, p. 700).

Furthermore, Seppänen (2014) argues that the success of EA initiatives depends on the governance model of the enterprise, and the guidance that the governance model provides to the transformation of the enterprise. For instance, Aziz et al. (2005) asserts that a well-functioning EA governance will enable IT to contribute to the creation of an agile enterprise. Similarly, Winter & Schelp (2008) state that, for an enterprise to realize EA process outputs in a consistent and timely manner, there is the need for a clear and effective governance mechanism. However, Winter & Schelp (2008) noted that unlike related fields (e.g. corporate governance, and IT governance), EA governance structures are still in their infancy.

TOGAF (2009) presents an architecture governance framework that conceptually consists of five constructs namely; processes, content, context, repository, and process flow control. The process flow control links the other three constructs together (See Figure 50-1 on TOGAF, 2009, p. 676). TOGAF separates the process, content and context (e.g. organizational form, and drivers) to allow new governance material to be introduced without unduly impacting the processes. TOGAF's architecture governance structure is made up of four levels namely; global governance board, local governance board, design authorities, and working parties (2009, p. 678). Aziz et al.(2005) present an EA governance framework that has four EA domains (Business Architecture, Information Architecture, Technical Architecture and Application Architecture) at its core and seven dimensions of EA governance (leadership, organization, process, enabling tools, management, policies and principles, and investment) at the periphery. They believe, each of these dimensions are crucial and together have the propensity to result in effective EA governance.

2.2.10 **Enterprise Architecture Planning**

Enterprise architecture Planning is another well researched area in the EA literature. According to Saat, Aier, & Gleichauf (2009), Spewak and Hill (1993) defined enterprise architecture planning as "the process of defining architectures for the use of information in support of the business and the plan for implementing those architectures". Indeed, Pulkkinen (2006) argues that, careful planning at the managerial level is necessary in finding the right strategies for and the implementation of any technologies including ICTs. In addition to the managerial level, EA planning occurs at different levels of the enterprise (Hirvonen & Pulkkinen, 2004; Saat et al., 2009) and across different EA domains (Hirvonen & Pulkkinen, 2004; Pulkkinen, 2006; TOGAF, 2009). EA is made up of different elements and relationships amongst the elements such that a change to one element might have multiple effects on other elements and their relationships (Saat et al., 2009; Jugel & Schweda, 2014) and thus making EA planning a complex task for EA stakeholders (Jugel & Schweda, 2014). In this regard, it is not likely to completely plan one of EA domains before planning another (Pulkkinen, 2006). Therefore, EA planning must pay particular attention to change impact on different EA elements (Jugel & Schweda, 2014)

In the EA literature, authors (e.g. Buchanan & Soley, 2002; Hirvonen & Pulkkinen, 2004; Pulkkinen, 2006; Saat et al., 2009; TOGAF, 2009; Jugel & Schweda, 2014) have proposed guidelines, procedures, and methods for EA planning. Buchanan & Soley (2002) present an EA planning model; the META Group's EA Process Model, that starts off architecture planning by combing the environment for trends (including threats and opportunities) to determine whether there should be an architectural change or not. If there should be architecture changes, a business visioning is done, and the enterprise business architecture is defined or refined. The enterprise business architecture then drives the information and technology architectures, and then the enterprise application portfolio. In this model, EA planning is considered "top-down" and hence the other architectures (e.g. technology and information) are considered "downstream" to the business architecture. The planning model is however iterative and progressively elaborate. Hirvonen & Pulkkinen (2004) proposed the "EA Management Grid" to bridge the gap between EA management consulting and software development. The grid consists of four (4) EA domains and three (3) EA levels of abstraction which corresponds to the decision making levels in an enterprise. The grid provides a practice oriented framework to manage requirements and constraints, and support EA projects and quality management. Based on the EA Management Grid, Pulkkinen (2006) conceptualized a method for decisions management in EA planning. Pulkkinen's method suggests that all the four (4) EA domains should be planned at all three (3) levels of abstraction, starting from the strategic level (2006). The method encourages iterations amongst EA domains at each level of abstraction, and iterations within the same domain across different levels of abstraction. The former enables evolving plans in one domain to serve as input to another domain, and the later make it possible for downstream success lessons and standards to become enterprise wide standards.

However, most EA planning methods implicitly assume the state of stability (Saat et al., 2009) and discrete architecture development, and thus do not adequately address the complexity of dynamics (Saat et al., 2009). Saat et al. (2009) argue that, EA transformation is irreversible or not repeatable especially because the environmental factors that prevail during each transformation project might differ from those that prevailed during prior transformation projects. They therefore proposed an evolutionary approach to EA planning (also see Bente et al., 2012, p. 160). Based on industry cases, Saat et al. (2009) identified three challenges that relating to EA planning, namely: volatility of architectural layers, architecture elements and their interrelationships; harmonization of projects and programs; and prediction and management of impacts of changes on different granularity levels. Furthermore, based on Chaos Theory, they proposed six (6) requirements for an EA planning method in order to handle the complexity of dynamics. The six requirements are: using the as-is (current state) as the foundation for planning; ensuring the relevance of planning; separation of points in time to manage shifting to-be models; affected architecture; volatility and life cycles; and different levels of planning.

2.2.11 Components (Building Blocks) of Enterprise Architecture Management

Towards answering the second research question, the author performed a search of extant literature that either explicitly or implicitly prescribed some building blocks/components of EA or EAM. The search was done by using the key words "Components of Enterprise Architecture Management" and "Elements of Enterprise Architecture Management" in Google Scholar. The results were screened for relevance and ten (10) publications were identified. Table 5 presents the ten (10) publications in chronological order based on the year of publication.

Firstly, Armour, Kaisler, & Liu (1999a, 1999b) and Armour & Kaisler (2001) through their three publications present a methodology for Enterprise IT Architecture management. The methodology promotes taking a big-picture view at the enterprise to capture the business vision and develop a common understanding of the EITA from multiple perspectives. According to the methodology, architecture should be developed incrementally and iteratively following a five-step process which includes; initiate the process, characterize the baseline architecture, develop/update target architecture, plan architecture transition, and plan architecture implementation. The methodology describes how the transition and the implementation of the architecture should be done adaptively to accommodate shifting targets. However, the EITA takes an IT-centric view towards EAM and views the big-picture as a system-of-systems, and lacks connection to how the EAM can support the agility of the enterprise as a whole.

Secondly in Pulkkinen & Hirvonen, (2005), the authors developed a process model for discrete enterprise architecture development efforts undertaken as incremental steps, and suited for EA projects that have time and scope limitations. The process model is anchored in an EA gird of four dimensions (business, information, systems and technology), and three organizational decision making levels: strategic, business operations and technical (e.g. regarding systems). The process model promotes collaborative decision making involving business and IT at all levels; cyclical (iterative) and incremental approach towards planning and development; and targeted interventions at developing discrete architecture components using the grid as a guide. The authors proposed that development projects should consist of three phases: the initiation, the working, and the ending phases. Though the method was tested on different projects with different scales and success criteria, the method lacks some essential mechanisms for instance, mechanisms to handle changing requirements and success criteria within one enterprise. It also lacks connection to how enterprise-wide agility will be supported.

Thirdly, based on their experience with industry, Wagter et al. (2005) proposed a model, the Dynamic Architecture Model, that is aimed at supporting agile architecture. The authors define agile architecture as the architecture that is purposefully designed to facilitate the speed of change in content (architecture products) and process (architecture development and maintenance processes): they named it "Dynamic Architecture" (DYA). The DYA model hinges on three core processes that include strategic dialogue, development with architecture, and architectural services(2005, p. 4). The strategic dialogue processes ensure that the enterprise level objectives are determined to inform subsequent project or development proposals; development with architecture ensures that IT solutions are development and implemented based on architectural provisions, for example principles and guidelines; and architecture services support the other two processes with principles, guidelines and models. Further, Wagter et al. (2005) proposed ten principles that underpins the DYA (2005, pp. 53–56) as illustrated in Table 5. Though the DYA model is focused on making the architecture (IT architecture) and the architecture development process (architecting) agile, it provides insight to the agility of the EAM function.

Fourthly, Rouhani et al. (2008) proposed a process model which they called the "Agile Enterprise Architecture Framework (AEAF)". The AEAF consists of seven constructs and eleven interrelationships amongst the constructs. The aim of the process model is to make the enterprise architecture and the architecting process agile. The AEAF provides constructs aimed at attaining agility, for instance constructs aimed at the monitoring and continuous optimization of the architecture and architecting processes; the anticipation of technical and functional architectural requirements in the near and distant future; and the adaptation of competencies and communication of architectural information to create collective intelligence. The process model however lacks clarity on how to apply the constructs, and connection to agility of EAM and the enterprise as a whole.

Fifty, Buckl, Matthes, & Schweda (2009) took a cybernetic point of view at EAM and presented EAM as a Viable System Model (VSM). They further elaborated on how the various components (systems) within the VSM could represent a component of EAM. Consequently, they prescribed five essential components of EAM namely; enterprise-level management processes, communication, reactive EA management, proactive EA management, and EA management governance. The enterprise – level management processes consists of systems that change the EA via projects which are in line with enterprise strategies, schedules, and priorities; the communication construct provides a common basis and foster communication amongst different architectural stakeholders for example through visualization; reactive EA management monitors primary activities and coordinate amongst them in order to deal with immediate concerns and maintain a holistic view of the primary activities; proactive EA management deals with further concerns that emerge from the changes in the enterprise environment to support strategic decision making; and the EA management governance which is embedded in the EAM function establishes balance between present and future architecture efforts, and steers the enterprise as a whole. The authors further prescribed six processes with which the EAM interfaces with other management processes within the enterprise.

A year later, Sabine Buckl and her colleagues proposed another set of building blocks for the EAM from a combination of concepts including the design theory nexus, pattern-based approach to EA management, and situational method engineering (Buckl et al., 2010). They proposed method framework called the Building Blocks of EAM Solutions (BEAMS) consists of loosely coupled activities including "develop and Describe", "communicate and Enact", "analyze and evaluate", and "configure & adapt" for EA management. Central to these components are "EA principles & target state of the EA", "Planned state of the EA", and "Current state of the EA". It is however, unclear how the building blocks support the agility of the EAM function and the agility of the enterprise as a whole.

Seventhly, in Buckl et al (2011), the authors identify challenges with enterprise architecture management and couch them into four (4) attributes that an EAM endeavor must have. These attributes include: the EA management endeavor has to be aligned with the stakeholders' interests expressed in a shared terminology; an EA management endeavor has to ensure an early and periodical delivery of concrete EA products; an EA management endeavor has to ensure commitment and involvement of all parties; and an EA management endeavor has to continuously adapt to a volatile environment with changing criteria for goal fulfillment. By applying agile software development (specifically, SRCUM) principles directly to enterprise architecture management, the authors mapped scrum roles, activities and deliverables to those of enterprise architecture management. However, issues relating to for example enterprise knowledge/awareness creation, and support for enterprise level agility remain to be addressed. The method might also inherit the difficulties associated in scaling team level agile methods to the enterprise level.

Eighthly, Aier, Gleichauf, & Winter (2011) employed Design Science Research (DSR) to analyze the problems underlying EAM and proposed eight factors that influence the design of EAM. These factors include IT Operation Support, Enterprise Focus and management Support, EAM Governance, IT Strategy and IT Governance Support, Information Supply, Integrative Role, Design Impact and Business Strategy Support. They further verified these factors through a survey that involved 119 respondents. The factors proposed here seem to be IT-centric and it is unclear how they promote the agility of the EAM function and the agility of the enterprise as a whole.

Ninthly, Bente, Bombosch, & Langade (2012, pp. 27–28) in their book on Collaborative Enterprise Architecture presented three main guidelines of collaborative enterprise architecture and corresponding six building blocks derived from agile, lean, and enterprise 2.0 practices. The main guidelines include streamlining architecture processes to establish a lean set of processes and rules; adopting evolutionary problem solving approaches by involving all stakeholders; and fostering open participation for example participation in knowledge, participation in decision, and participation in transformation. The authors presented four dimensions of EA challenges; namely perspective, governance, strategy, and transformation, and argued that challenges in EA occur when extreme views to these dimensions: too high or too low, too rigid or too weak, too far reaching or too myopic, and too fast or too slow respectively, are taken. They proposed the six building blocks as remedy to EA challenges.

Lastly, Ahlemann, Stettiner, Messerschmidt, & Legner (2012) in their book "Strategic Enterprise Architecture Management: Challenges, best practices and future developments" proposed an EA management "navigator" that consists of seven building blocks of successful EAM (2012, pp. 38–55). These include "the EAM agenda for the Chief Executive Officer", "EAM governance and organization", "Embedding EAM into strategic planning", "Embedding EAM into the project life cycle", "Embedding EAM into operations and monitoring", "EA frameworks, modeling and tools", and "People, adoption and introduction of EAM".

Clearly, the agility of the EAM and its relation to the agility of the enterprise has sparingly been treated in the EA literature. However, the literature reviewed provides a preliminary list of essential elements, and lays a solid foundation on which the thesis builds to construct the essential elements of an EAM method.

Table 5: Components of EAM in Extant Literature

Author	Focus	Summary of Components of Enterprise Architecture Management
Armour, Kaisler, & Liu (1999a, 1999b), Armour & Kaisler (2001)	Agile EA (EITA)	• Capture the business vision • Provide common understanding from multiple perspectives • Develop the architecture incrementally and iteratively (initiate the process, characterize the baseline architecture, develop/update target architecture, plan architecture transition, plan architecture implementation) • Transition and implement adaptively
Pulkkinen & Hirvonen (2005)	Agile EA Dev.	• Collaboration and a joint decision making process with business and IT managers at all levels • Administrative, infological and technology decisions are considered at every level, with a narrowing scope and lower level of abstraction when going down the levels • Cyclical (iterative) and incremental approach are taken towards planning and development • Targeted interventions using the EAM Grid • Development projects should have 3 phases (the initiation, the working, and the ending phases)
Wagter, Van	Agile	• Processes (Strategic Dialogue, Development with architecture, Ar-

Den Berg, Luijpers, & Van Steenbergen (2005, pp. 53–56)	EA	chitectural Services)
		• Principles:
		  o Architecture is strategic if IT is strategic.
		  o Architecture must facilitate speed of change.
		  o Communication between business and IT management is crucial.
		  o Business objectives govern the development of architecture.
		  o The level of architecture will be continually raised if architecture is aligned to important business changes.
		  o Architecture must be developed "just enough, just in time."
		  o Working under architecture is supported by a theoretical and working model.
		  o Transparent relationships must be defined
		  o Several development strategies are distinguished
		  o Architectural principles and processes must be an integral part of the organization.

Rouhani, Shirazi, Nezhad, & Kharazmi (2008)	Agile EA	• Process monitoring and continuous optimization • Business projection, resource and supporting technologies • Technical and functional anticipation • Adaptation of competencies and collaboration types • Information system and technological system • Processes operation • Design a rudimentary model

Buckl, Matthes, & Schweda (2009)	EAM	• Enterprise –level management processes
		• Communication
		• Reactive EA management
		• Proactive EA management
		• EA management governance
		• EAM process –
		  o Delivery of planned & as-is landscapes
		  o Document changes in landscape
		  o Coordinate with other initiatives and to-be landscape
		  o Transfer to planned landscape
		  o Incrementally detail landscape changes
		  o Transfer planned to as-is

Buckl, Dierl, Matthes, & Schweda (2010)	EAM	• Analyze & Evaluate • Develop & Describe • Communicate & Enact • Configure & Adapt • EA principles & target stake of the EA • Planned state of the EA • Current state of the EA
Buckl et al (2011)	Agile EAM	• The EA management endeavor has to be aligned with the stakeholders' interests expressed in a shared terminology. • An EA management endeavor has to ensure an early and periodical delivery of concrete EA products. • An EA management endeavor has to ensure commitment and involvement of all parties. • An EA management endeavor has to continuously adapt to a vola-

		tile environment with changing criteria for goal fulfillment.
Aier, Gleichauf, & Winter (2011)	EAM	• IT Operations support • Enterprise focus and management support • EAM governance • IT strategy and IT governance support • Information supply • Integrative role • Design impact • Business strategy support

Bente, Bombosch, & Langade (2012, pp. 27–28)	Agile EA	• Establish a lean set of processes and rules : get rid of waste by streamlining architecture processes
		• Adopt evolutionary problem solving : Involve all stakeholders by interlocking architecture scrums
		• Foster and moderate open participation:
		  o Practice iterative architecture through EA Kanban (welcome change and iterative design)
		  o Participation in knowledge (sharing & combining knowledge, integrated information repository)
		  o Participation in decision (collaborative decision making)
		  o Participation in Transformation

Ahlemann, Stettiner, Messerschmidt, & Legner (2012, pp. 38–55)	EAM	• Top management awareness and support (a CxO agenda). • EAM governance and organization. • Embedding EAM into strategic planning. • Embedding EAM into the project life cycle. • Embedding EAM into operations and monitoring. • EA frameworks, modeling and tools. • People, adoption and EAM introduction

2.3 **Summary**

This chapter presented the two major concepts of this thesis; enterprise agility and enterprise architecture. The enterprise agility subsection enabled the author of the thesis to acquire adequate knowledge of the area of enterprise agility, and established the theoretical background towards answering research question 1. The definition for enterprise agility from different perspectives including manufacturing, management, new product development, and information systems were reviewed. Based on extant definition, a more suitable definition of enterprise agility for this thesis was proposed.

The subsection on enterprise architecture established a sound understanding of the EA literature with particular focus on EA, Enterprise Architecting, and Enterprise Architecture Management and the relationship amongst these three; and other salient issues in EA especially EA governance, EA principles, EA planning, and EA frameworks, levels and domains. Also, extant literature that are dedicated to developing an agile EAM method, an agile EA, or components (elements) for EAM were reviewed to develop a list of preliminary EEs of an agile EAM method.

3 CHAPTER 3 THEORETICAL FOUNDATION FOR ADAPTING LEAN AND AGILE PRINCIPLES AND VALUES

This chapter is the second chapter on literature review. It starts off by presenting the Lean and Agile principles and values from extant literature and suggesting some theories that could be employed to adapt the Lean and Agile principles and values to Enterprise Architecture Management (EAM). The suggested theories, referred to here as "probable theoretical glue", are then presented with concentration on how they relate to enterprise agility and its antecedents. Lastly, the Critical Success Factors (CSFs) of EAM that were identified through systematic literature review (as explained in section 4.5 under research methodology) are briefly presented.

3.1 Probable Theoretical Glue

EA literature has been found to be predominantly led by practitioner actions, for instance practitioner publications, and so tend to lack theoretical underpinning for instance, concerning why certain claims are made (Tamm et al., 2011). This might be an inherited characteristic from the general IS design community where either very abstract representations (e.g. mathematical models) are used to represent the business/technology environment, or very informal descriptive IS models (e.g. an EA framework) are used to represent such environments (Hevner et al., 2004). The former is criticized for lack of relevance to the business environment, and the latter is criticized for lack of underlying theory base (Hevner et al., 2004), hampering its generalizability. Indeed, the design artifact is regarded by many as the whole point in DSR and therefore little emphasis is placed on what it means to make contribution to generalized knowledge (Gregor & Hevner, 2013). In this sub-section, the thesis proposes some theories, or what Gregor & Hevner (2013) refer to as the justification theory or kernel theory, with which the Lean and Agile values and principles would be adapted for EAM, taking cognizance of the literature reviewed in chapter 2. This approach helps to curb the shortcomings associated with directly applying the agile principles (Abrahamsson et al., 2009; Boehm, 2002; Conboy, 2009) especially to fields outside systems development, and to provide the *"theoretical glue"*(Conboy, 2009; Whetten, 1989) that supports the applicability of these principles to EAM for enterprise wide impact; specifically enterprise agility. The set of theoretical glue selected here are called "probable theoretical glue" to signify that different authors could select different theories depending on their understanding of the problem area and/or the perspective they wish to take.

3.2 Lean and Agile Principles and Values

In this sub-section, focus is shifted to Lean and Agile values and principles as in extant literature. Also, probable theories that could guide the application of the Lean and Agile values and principles to EAM will be suggested. Further, the justification for the combination of Lean and Agile values and principles will be provided. The sub-section ends with a presentation on the complimentary relationships between Lean and Agile in continuous development.

3.2.1 Agile Principles and Values

Although some authors (e.g. Abbas, Gravell, & Wills, 2008) do argue that the concept of agility in information systems development did exist prior the Agile Manifesto, there seem to be consensus that the Agile Manifesto popularized the concept of agility in software development (Abbas et al., 2008; Abrahamsson et al., 2009). The Agile Manifesto proposes four (4) values and twelve principles. Table 6 lists the agile values and principles together with their corresponding probable theoretical glue. Agile methods (e.g. SCRUM and XP) are embodiments of the agile values and principles. These methods have been widely adopted especially in information systems development projects with small teams of up to an average of seven (7) members, nonetheless, in exceptional situations these methods are successful with projects of up to 250 team members (Cockburn & Highsmith, 2001). Also, the merits of these agile values, principles and methods are widely known within the IS community. However, there exist criticisms about the scalability of the agile values, principles and methods not only to the enterprise level (e.g. Kähkönen, 2004), but also to large scale information system development projects (e.g. Boehm, 2002). As remedies, many authors including academicians (e.g. Boehm, 2002; Kähkönen, 2004) and practitioners (e.g. Leffingwell et al., 2011) have suggested means to scale up agile methods. For example, Kähkönen (2004) suggest that other structures (e.g. community of practice) should be used in conjunction with the formal team structures in agile methods; Boehm (2002) suggests that hybrid approaches that combine "traditional" and agile methods should be used to overcome limitations that result from using either traditional or agile methods alone; and Leffingwell et al., (2011) suggest that "lean method" should be combined with agile methods and managed within a framework of projects and portfolios.

Table 6: Agile Values and Principles

Agile Values	Probable Theoretical Glue
Individuals and interactions over processes and tools	Participation/ Collaboration
Working software over comprehensive documentation	Organizational fitness
Customer collaboration over contract negotiation	Participation / Collaboration
Responding to change over following a plan	Dynamic Capability
Agile Principles	**Probable Theoretical Glue**
Satisfy the customer through early and continuous delivery of valuable software	Organizational fitness
Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage	Contingency-based requirement management
Deliver working software frequently, from a couple of weeks to a couple of months, with a preference for the shorter timescale	Organizational fitness
Business people and developers work together daily throughout the project	Participation/ Collaboration
Build projects around motivated individuals, give them the environment and support they need and trust them to get the job done.	Team building (Agility Provider)
The most efficient and effective method of conveying information with and within a development team is face-to-face conversation	Participation/ Collaboration
Working software is the primary measure of progress	Organizational fitness
Agile processes promote sustainable development. The sponsors, developers and users should be able to maintain a constant pace indefinitely	Dynamic capability / Collaboration
Continuous attention to technical excellence and good design enhances agility	Participatory Design
Simplicity, the art of maximizing the amount of work not done is essential	Continuous improvement
The best architectures, requirements and designs emerge from self-organizing teams	Emerging design (Participatory Design)
At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly	Organizational fitness & Dynamic Capability

3.2.2 **Lean Principles and Wastes**

The origin of Lean thinking can be traced to the shop-floors of Japanese manufacturers, especially Toyota Motor Corporation, however, according to Hines, Holweg, & Rich

(2004) the term "lean thinking" was popularized by the publication "*The Machine That Changed the World : The Story of Lean Production*" in the 1990s. From its birth ground in the manufacturing (especially the automobile) industry, the lean concept has spread into other industry sectors (Hines et al., 2004), for example construction, military, logistics and service industries (Poppendieck, 2011).

Lean thinking is based on a set of principles and values. Hines et al., (2004), based on Womack and Jones(1996), listed the lean principles to include; identifying customer value, managing the value stream, developing the capability to flow production, using "pull" mechanisms to support flow of materials, and pursuing perfection through elimination of waste in the production system. The lean wastes include over production, waiting, transportation, inappropriate processing, inventory, unnecessary motions and defects (Bhasin & Burcher, 2006; Hines et al., 2004). The lean principles and lean wastes and their respective probable theoretical glue are represented in Table 7.

Table 7: Lean Principles and Lean Wastes

Lean Principles	**Probable Theoretical Glue**
Identifying customer value	Participation/ Collaboration
Managing the value stream	Organizational fitness & Dynamic capability
Developing the capability to flow production	Organizational fitness & Dynamic capability
Using "pull" mechanisms to support flow of material	Dynamic Capability/ Participation/ Collaboration
Pursuing perfection through elimination of waste in production system	Organizational fitness & Dynamic capability
Lean Wastes	**Probable Theoretical Glue**
Over production	Emerging design (participatory design)
In-process inventory	Contingency-based requirement management
Extra processing steps	Organizational fitness
Motion	Dynamic capability
Defects	Organizational fitness
Waiting	Organizational fitness & Dynamic capability
Transportation	Collaboration/Participation

3.2.3 **Why Lean and Agile?**

According to Hines et al., (2004) as lean principles are applied in other sectors, the focus on eliminating waste on the factory floor reaches its limitations and newer approaches that contingently enhance perceived value to customers are born. Value in these new approaches is created by adding products or service features that customers perceive to be useful, and/or by removing wasteful activities (Hines et al., 2004). One of the most stated approaches is the lean-agility (Hines et al., 2004) or the leangility approach which is a careful combination of lean and agile paradigms (Naylor, Naim, & Berry, 1999). From a strategic point of view, Hines et al., (2004) concluded that other concepts that enhance customer value, example agility concepts, can be integrated with lean without compromising the core objectives of lean. Indeed, Van Hoek (2000) indicates that the combination of lean and agile or *leangility* might be feasible as lean capabilities support and might be prerequisite to agile performance especially in operational terms. Similarly, by concentrating on lean and agile teamwork within organizations, Putnik, Browaeys, & Fisser (2012) found that lean and agile are not only inseparable but also interdependent. Empirically, a study of the Finnish software industry revealed that though the adoption of lean concepts in general and specifically, the combination of lean and agile concepts are more recent than the adoption of agile principles in software development, approximately 22% (n=88) of the respondents (n = 408) reported that their companies use a combination of lean and agile principles in software development.

Indeed, lean thinking (including values and principles) has been applied in Information Systems (software development) and Enterprise Architecture literature. For instance, Poppendieck & Poppendieck (2003, pp. 18–22) and Poppendieck (2011) adapted lean principles and wastes for software development and Bente et al. did for Enterprise Architecture (2012, pp. 175–193).

3.2.4 **Lean and Agile for Continuous Improvement**

At the heart of Lean and Agile concepts presented above is the concept of continuous improvement. Agile concept stresses iterative and incremental delivery of products; with each of the iterations adding more value and meeting customer needs. Similarly, lean concept stresses the elimination of waste and continuous improvement of the product through value addition, as well as continuous improvement of the processes by which the products are created through the elimination of non-value adding processes, and improvement of the value-adding processes towards perfection.

Continuous improvement can be defined as a company-wide process of enabling of focused incremental innovation (Bessant, Caffyn, Gilbert, Harding, & Webb, 1994). In this regard, it can be seen as an outcome, and the process that produces the outcome (Bessant, Caffyn, & Gallagher, 2001). CI is believed to enable organization-wide process improvement, granular improvement through step-by-step innovation, mobilization of people in an organization around improvement activities, and creation of a learning and growing environment (see Wu & Chen, 2006). Furthermore, Bessant & Francis (1999) argued that CI can be seen as a dynamic capability, with particular focus on CI as a process; since it is the process nature of CI that enables it to involve large portion of the organization in innovation and learning processes, and also define how the organization confronts issues relating to innovation, learning and renewal. Similarly, Anand, Ward, Tatikonda, & Schilling,(2009) noted that process improvement involves organizational learning to make changes in operating routines and could therefore constitute dynamic capability when it is rooted within a comprehensive organizational context. Clearly, for an organization to harness these merits from incremental changes, it needs to sustain the CI initiatives over the long term towards the particular objective it seeks to achieve (Bessant et al., 1994)

However, implementation of CI initiatives is not without problems. Even when short-term gains begin to accrue, the sustainability of CI initiatives in the long-term can be problematic (Bessant et al., 1994; Wu & Chen, 2006). According to Bessant et al. (1994) the successful implementation of CI initiatives will depend on the creation of enabling organizational contexts. They proposed six organizational factors including the need for clear strategic framework, strategic management (including careful planning and monitoring), underlying supportive culture, enabling infrastructure (e.g. organic organizational structure, local autonomy and empowerment, and communication), managing as a process, and supporting tools (1994, pp. 19–22)

CI initiatives are based on concepts such as lean, Six Sigma, and "Plan – Do – Check – Act" Deming cycle (Caffyn, 1999; Anand et al., 2009; Sokovic, Pavletic, & Pipan, 2010). Caffyn(1999) advocate that enterprises implementing CI initiatives should do so as a long-term process following an iterative cycle based on the "Plan – Do – Check – Act" Deming cycle or on an organizational development cycle. Furthermore, the "Plan – Do – Check – Act" Deming cycle is recognized as part of the continuous improvement aspect of quality management (Hervani, Helms, & Sarkis, 2005).

The "Plan – Do – Check – Act" Deming cycle consists of four main steps towards iterative problems solving, and can be used to effect both incremental as well as major improvements. It is usually represented as shown in **Error! Reference source not ound.**. "PLAN" relates to planning activities or tests aimed at improvement; "DO" relates to activities aimed at carrying out the change or test preferably on a small scale, "CHECK/STUDY" relates to activities aimed at checking or studying the results from "DO" to see if the change has been carried out as desired or something went wrong, and identifying the lessons that were learnt; and "ACT" refers to activities aimed at enacting the change or test preferably on a small scale (Moen & Norman, 2006).

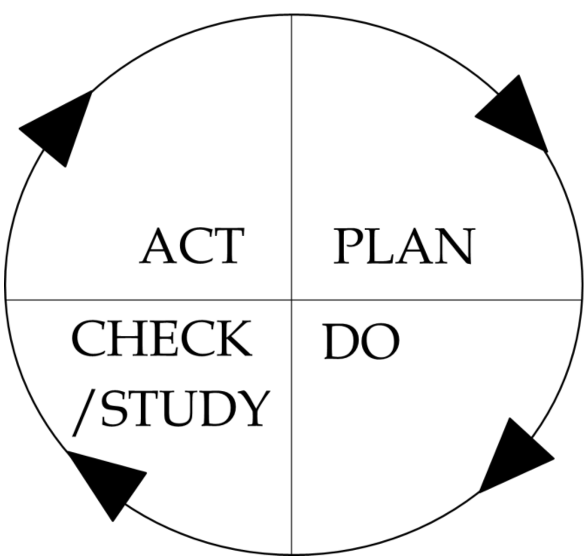


Figure 2: The Deming Cycle

3.3 Review of the Probable Theoretical Glue

In this sub-section the proposed "probable theoretical glue" is presented. The "probable theoretical glue" includes theories on collaborative decision making, participatory design, organizational fitness, contingency based requirement management, and dynamic capability. The import here is to trace how each of the "probable theoretical glue" impacts on enterprise agility or its antecedents for example, the agility providers and agility capabilities that were discussed in chapter two. This is done primarily through the review of previous studies and empirical research on each of the theories that help to trace the path, rather than covering the breadth and depth of each of the theories. In other words, the literature review on each of the probable theoretical glue seeks to find out whether prior research has established a relationship between the probable theoretical glue and enterprise agility or its antecedents. For example, literature review on collaborative decision making seeks to find out whether collaboration amongst different stakeholders has been found in extant literature to have any relation with enterprise agility, and to understand the possible relationship.

3.3.1 Collaborative Decision Making

Collaborative decision making or planning approaches are used to encourage the participation of diverse stakeholders in making decisions, especially within the contexts where social and political fragmentation, shared power, conflicting values, and differences in knowledge are on the increase (Innes & Booher, 1999). Also, where there is the need to accomplish significant or innovative gains (e.g. decisions, relationships, or learning), there is the need to create flexible linkages among stakeholders through collaborative decision making (Innes & Booher, 1999). In this regard, collaborative decision making takes cognizance of diverse characteristics, including "diverse views and multiple knowledges and understandings"(Innes, 2004, p. 9), of participating stakeholders; and promotes consensus based on stakeholder interests and not on arguments about predetermined positions (Innes & Booher, 1999).

Advantages of collaborative decision making are well established in literature. For instance, Innes,(2004) asserts that collaborative decision making results in shared understanding and relationship building among stakeholders, and produces package of interlinked actions and proposals. Furthermore, Innes & Booher,(1999) list agreements and plans, experimentation, innovative ideas, building shared learning, and ability to handle change as outcomes of collaborative decision making.

According to Innes,(2004), there are eight major conditions that a collaborative decision making process should observe. These are;

- Inclusion of a full range of stakeholders;
- A task that is meaningful to the participants and that has promise of having a timely impact;
- Participants who set their own ground rules for behavior, agenda setting, making decisions and many other topics;
- A process that begins with mutual understanding of interests and avoids positional bargaining;
- A dialogue where all are heard and respected and equally able to participate;
- A self-organizing process unconstrained by conveners in its time or content and which permits the status quo and all assumptions to be questioned;
- Information that is accessible and fully shared among participants;
- An understanding that 'consensus' is only reached when all interests have been explored and every effort has been made to satisfy these concerns.

Collaborative decision making has been explored in diverse areas including medical (e.g. Politi & Street, 2011), air traffic management (e.g. Ball, Chen, Hoffman, & Vossen, 2001), IT governance and strategy (e.g. Peterson, 2004), and enterprise architecture (e.g. Bente et al., 2012).

In IT governance, Peterson claims that IT governance should focus on Horizontal Integration Capabilities (HCI); which he defines as "the ability to coordinate and integrate formal and informal IT decision-making authority across business and IT communities" (2004, p. 14). According to him, HIC can be classified into structure capability (connection), process capability (coordination), and rational capability (collaboration). Rational capability (collaboration) integration builds trust and the willingness of business and IT stakeholders to work together resulting in business-IT partnership, and shared leaning and understanding of business/IT objectives (Peterson, 2004). Also, Teo & King(1999) framed four-level typology of Business Planning (BP) – Information Systems Planning (ISP) integration based on prior literature. In type1, BP is separate from ISP with administrative integration; in type2, BP informs ISP; in type3, BP feeds ISP and ISP feeds BP; and in type4, BP and ISP are conducted as an integrated process. Teo & King(1999) argue that unlike in type1 and type2 where the IS function plays a restrictive role, in type3 and type4 the IS function is proactive in supporting and influencing business strategy. Indeed, the reciprocal interest between business and IT, as implicated in type3 and type4, is a necessary precondition for collaborative decision making (Innes, 2004).

Furthermore, through as systematic literature review, Chen et al. (2010) discovered three concepts of IS strategy: first, IS strategy as the use of IS to support business strategy; second, IS strategy as the master plan of the IS function; and third, IS strategy as the shared view of the IS role within the organization. The third concept of IS strategy as found by Chen et al. (2010) aligns with type3 and type4 of BP-ISP integration in Teo & King(1999). According to Chen et al. (2010), conceiving IS strategy as the shared view of the IS role within the organization elevates IS strategy to be part of corporate strategy, makes IS strategy enterprise-centric, and enables IS strategy to guide future IS-related business decisions, and push or support business initiatives. This conception also promotes shared understanding across the organization and enables dynamic alignment (coevolution) of business and IS strategies (Chen et al., 2010).

Similarly, in the enterprise architecture literature, (Lapalme, 2012) presents the three schools of thought (or belief of enterprise architects) on enterprise architecture namely; Enterprise IT Architecting (EITA), Enterprise Integrating (EI), and Enterprise Ecological Adaptation (EEA). The EITA school of thought takes a reductionist look at EAM and develops IT strategies to support business strategies, and therefore treats EA as "glue" between business and IT. In this school of thought, business strategy is assumed to be complete and always leads IT strategy; and all the different organizational dimensions (EA domains) are designed independently. The EI school of thought applies systems thinking, jointly architects all the different EA domains and encourages collaboration across the organization to attain organizational coherency. This school of thought also assumes business strategy is complete and employs EA as a link between strategy and execution. The EEA school of thought supports organizational innovation and adaptation by supporting coevolution between the organization and its environment on the one hand, and "intra-organizational coherency" on the other. This school of thought promotes organizational learning, shared awareness and understanding, and "sensemaking"; and fosters collaboration across the organization to jointly design all EA dimensions. The three schools of thought presented by (Lapalme, 2012) largely align with the work of Teo & King(1999), and Chen et al. (2010). Also, the EI and EEA schools of thought, taken together, support the concept of organizational fit (e.g. Miles, 1984), and types of enterprise agility (e.g. Lu & Ramamurthy, 2011; Vallabh Sambamurthy, Bharadwaj, & Grover, 2003b). Table 8 shows a summary of these different types of business – IS relationships in the literature discussed above.

Table 8: Literature sources on business – IS relationships

Pertserson (2004)	**Teo & King (1999)**	**Chen et al. (2010)**	**Lapalme (2012)**
Structure capability (connection)	BP – ISP with Admin integration	IS as master plan of IS function	Enterprise IT architecting
Process Capability (Coordination)	BP informs ISP	IS Strategy to support Business Strategy	Enterprise Integrating
Rational Capability (Collaboration)	BP feeds ISP and ISP feeds BP BP&ISP as integrated process	IS strategy as shared view of the IS role within the org	Enterprise Ecological adaption

3.3.2 **Participatory Design**

Fundamentally, Participatory Design (PD) processes reflect design as a social process that allows those affected by a design to participate in the design process, and thus "extends the sphere of the design activity beyond the designer "(Luck, 2003). Consequently, the form of the ensuing artifact from PD processes reflect the collective effort of the designer and the users (Redström, 2008). Carroll & Rosson (2007) noted that PD integrates two radical propositions about design; a moral proposition that states that users have the right to participate in the design process, and a pragmatic proposition that states that including users' input will increase the chances of a successful design outcome (e.g. quality of design artifact). Similarly, Bjögvinsson, Ehn, & Hillgren (2012) state that PD is motivated by social and rational idea of democracy that enables proper and legitimate participation of users, and the quest to benefit from participants' knowledge, especially tacit knowledge, in the design process. Indeed, according to Spinuzzi (2005), PD hinges on constructivism theory that resists the notion that knowledge can be completely formalized and classified.

As users participate in PD processes and experiment with design techniques such as prototyping and modeling that make otherwise abstract ideas clearer, they have the opportunity to influence the design of the form of an artifact ("thing-design") or the intended use of the artifact ("use-design") or both iteratively (Redström, 2008), and to anticipate "use before actual use" (Redström, 2008; Ehn, 2008). Complementary to these two views is the notion of "design after design" or "meta-design" that encourages designing in a manner that provides space for further design and appropriation of the designed artifact during actual use or after the initial design project (Redström, 2008; Bjögvinsson et al., 2012). This allows the participation of not only users and designers, but also inanimate things (e.g. models, and other artifacts) in the design process (Ehn, 2008). One criticism against "traditional" PD and at the same time a reason underlying "design after design" is that actual use will most likely differ from anticipated use no matter how participative the design process was.

Also, generally, participation has been identified to have positive effect on use (e.g. Baroudi, Olson, & Ives, 1986; Hartwick & Barki, 1994), and use on the organizational benefits derived from design artifacts (e.g. Delone & McLean, 2003; Seddon, 1997; Lange, Mendling, & Recker, 2015). In information systems, Barki & Hartwick (1989) distinguished between user participation and user involvement. According to them, user participation refers to the behavior or actions that the target user or its representative performs in the systems development process, whilst user involvement refers to the subjective psychological state of mind that defines the extent of relevance that the user attaches to the system or the system development process depending on the user's focus. They further proposed user participation as an antecedent of user involvement. In a later publication, Hartwick & Barki, (1994) found that early in a system development process, a user's intention to use a system may be influenced by subjective norm (intention to use because others expect them to) however, when the system is operational the user's attitude (or the feeling of relevance) towards the system is the crucial determinant of intention to use. They concluded that amongst the activities that a user undertakes during the systems development process, meaningful participation (what McKeen et al. (1994) refer to as "influence" or "effective communication" and Baroudi et al. (1986) refer to as "substantive involvement") has the greatest effect on user involvement, attitude and use.

Similarly, evidence from Baroudi et al. (1986) suggests that user involvement (referring to "user participation" in the terminologies of Barki & Hartwick, (1989)) in the system develop process enhances system usage and impacts the user's satisfaction with the system via system usage. Similarly, though Seddon (1997) found a significant relationship between user involvement and system usefulness (or perceived usefulness), they did find a non-significant relationship between user involvement and user satisfaction, and suggested that user satisfaction is a perceptual benefit that results from the actual use of a system. However, from the analysis of 151 independent systems development projects in eight different organizations McKeen et al.(1994) found a direct relationship between user participation and user satisfaction. Furthermore, Delone & McLean (2003), proposed a relation between system use and net benefits (including organizational benefits) and between user satisfaction and net benefits.

The degree and the determinants of the degree of participation have also been studied in literature. For instance, McKeen et al.(1994) studied the effects of four contingencies; task complexity, system complexity, user influence, and user-developer communication; on the relationship between user participation and user satisfaction. They found that, in cases of high task complexity and/or system complexity the relationship between user participation and satisfaction is stronger than in projects of low task complexity and/or system complexity. In other words, in cases of high task complexity and/or system complexity higher degree of participation is needed than it is in cases of low task complexity and/or system complexity. The other two contingencies were found to have positive relation with user satisfaction regardless of the degree of participation. Also, Vink et al.(2008) found that different stakeholder groups participate at different levels along the development and change processes. For example, top and middle management participate to set goals that are consistent with enterprise strategies, designers facilitate idea generation and prototyping, and middle management participates in improvement selection and implementation. Therefore, for participation to yield optimum results there is the need to engage the relevant stakeholders at the appropriate levels and to the degree required by the complexity of the architecture endeavor at hand, and to enable them to influence the design process and outcome.

3.3.3 **Organizational Fitness**

According to Beer et al. (2005) the concept of "fit" has been widely studied in the business literature. Within the organizational context, "fit" is seen as the state of alignment between the organization and its environment, and the arrangement of internal resources to sustain the alignment (Miles, 1984). In that regard, fit is "a process as well as a state"(Miles, 1984, p. 11). Organizational fit suggests that there is the need for multiple alignments for an organization to perform effectively. For instance, the organization's strategy must be aligned to its environment, its capabilities with its strategy, and its design and culture with its capabilities, and its leadership behavior with its design (Beer et al., 2005). Similarly, Miles (1984) noticed that the proposed remedies for organizational renewal lies in the ability of the organization to achieve fit amongst its strategy, structure, and management processes. Organizational fit, therefore, can be categorized into external or internal fit (Miles, 1984; Beer et al., 2005). External fit relates to the alignment between the organization's strategy and its environment, and internal fit is the consistency amongst the organization's processes, structure and culture, and their alignment with the organization's strategy (Miles, 1984).

However, as organizations and their environments are exposed to frequent changes, a static look at organizational fit becomes unsuitable, and thus organizational fit needs to reflect "organizational dynamics" (Nissen, 2014). Nissen (2014) coined the terms "equilibrating" and "fluxing" to describe the two types of organizational fit that were prevalent in prior literature. Equilibrating organizational fit pursues and maintains periodic fit (stability) through sequences of static organizational (re)designs. In practical terms, the organization is designed to fit its current contingencies and redesigned only when misfit ensues between the current organization design and its contingencies. However, according to Miles (1984) there exists a blur line between fit and misfit. Therefore, organizational redesigns to achieve fit (re-equilibration) are more probable to occur when severe, or at least "enough", misfit ensues (Nissen, 2014). Fluxing orientation pursues continuous changes in organizational design in response to changing contingencies. It is the embodiment of concepts including organizational dynamics, emergent patterns, dynamic capabilities, modular reconfiguration, and organization inertia (Nissen, 2014).

Due to the multiplicity of components within the environment (internal and external) of an organization, research efforts regarding organizational fit typically focus on fit between specific pairs of components (Kanellis, Lycett, & Paul, 1999). Within the IS discipline, the focus is typically on fit between specific organizational dimension and IS (Hong & Kim, 2002; Iivari, 1992; Kanellis et al., 1999; Markus & Robey, 1983). Davenport & Short (1990) long established the recursive relationship between IT capabilities and process redesign, and also provided guidelines for process redesign and organizational structure. Also, Yang & Seddon (2004) posit that the degree of fit between an implemented Enterprise Systems (ES) and organizational needs over time affect the perceived benefits from ES use over time. More recently, Strong & Volkoff (2010) developed a theoretical conceptualization of organization – ES (org-ES) fit which suggests six misfit domains (organizational culture, control, roles, functionality, data, and usability) for ES. These findings supports the view of Iivari (1992) that "the concept of "fit" expresses an idea that the object of design (an organizational structure, or an information system) must match its context in order to be effective".

3.3.4 **Contingency Based Requirement Management**

Requirements Engineering (RE) is believed to be a mature field in Software Engineering that has been adopted into other fields (Jarke & Pohl, 1994), including enterprise architecture. In relation to enterprise architecture, TOGAF defines requirement as "a quantitative statement of business need that must be met by a particular architecture or work package"(2009, p. 35). Similar to requirements in Software Engineering, EA requirements could be functional or non-functional (TOGAF, 2009, p. 103).

In enterprise architecture, managing these requirements poses very difficult problem because: firstly, EA has diverse stakeholders who have different interests, understand different languages, and are at different levels of the enterprise; secondly, EA consists of different architectures classified by EA domains, time frame, and different levels of abstraction and detail; and thirdly, EA concerns the whole enterprise embodied in its components, their interactions, and evolution within a dynamic, complex and competitive enterprise environment. In effect, requirement management in EA needs to pay particular attention to potentially conflicting perspectives from different stakeholder (TOGAF, 2009, p. 7); and changing contingencies including criteria for goal fulfillment (e.g. timeliness, cost efficiency, and percentage of changes to be accommodated), and enterprise competitive bases (e.g. speed, cost, flexibility, need for partnership, and operational efficiency). According to Jarke & Pohl(1994), in order to facilitate goal-directed change in evolving organizations, requirements solicitation and representations (e.g. modeling) must be done throughout the lifetime of the system; here enterprise architecture. In deed Engelsman et al.(2011) assert that requirement modeling is an important part of enterprise architecture management and that "Goals have to be refined into requirements before their realization can be assigned to some artifact, such as a business service, business process, application service or application component" (2011, p. 10). Also, TOGAF places requirement management at the center of its Architecture Development Method (ADM) and emphasizes that it should be carried out throughout, and drive the ADM (2009, p. 204).

With the different and potentially conflicting interests or requirements that EA stakeholders have, comes the difficulty that relates to the ability to interpret and properly evaluate these requirements. For instance, Kaiya, Shinbara, Kawano, & Saeki (2005) refer to the differences in stakeholder perspectives concerning requirements as "*requirement discordance*", and suggest that there are two types of requirement discordance. First, the discordance that results from the different interpretations that EA stakeholders might have for a requirement (*discordance in interpretation*); and second, the discordance that results from differences in the values or level of importance that EA stakeholders ascribe to a requirement (*discordance in evaluation*). They proposed that discordance in interpretation could be alleviated through collaborative processes that foster shared interpretation amongst the different stakeholders, whilst the discordance in evaluation could be alleviated through compromises and establishment of win-win situations.

However, one assumption that might have underpinned the work of Kaiya, et al. (2005) is that the stakeholder who originally states a requirement fully understands the requirement and states it clearly enough for other stakeholders to understand; and that the discordance in interpretation is not actually caused by probable ambiguity of the stated requirement. Also, it is likely that discordance in evaluation could be influenced by time and contingencies. For example, a highly valued requirement today might be valued less tomorrow even by the same stakeholder, in the face of changing contingencies including changes in customer demands, project feedbacks, early-wins, and project constraints. Therefore, beyond the detection of discordance it is imperative for requirement management methods, especially in fast changing environments, to consider how the clarity, complexity, volatility, and conflicts in requirements could be managed. For instance, again from software development perspective Mathiassen, Saarinen, Tuunanen, & Rossi(2007) identified three problem areas in relation to requirements. First, *requirements identity* that relates to communication, physical, conceptual and cultural gaps between developers and would-be users; and that could result in high-risk when the requirements are unknown or indistinguishable. Second, *requirement volatility* that relates to the degree of stability of the requirement; and that could result in high-risk when the requirements change rapidly as stakeholders learn more about the system during development, or in response to sources of volatility example market impacts, company impacts and hidden impacts. And third, *requirement complexity* that relates to how easy it is to understand the requirements; and could result in high-risk when the requirements are difficult to understand, specify, and communicate.

Mathiassen et al. (2007) further suggested four techniques namely; requirement discovery technique, requirement prioritization technique, requirements experimentation technique, and requirement specification technique; with which risks associated with requirement identity, volatility and complexity could be managed. Requirement discovery techniques (e.g. use case-based, and focus group interview) are customer-focused and based on identification and prediction of customer needs, explicate requirements but take less rigid approaches to documentation, appreciate emerging requirements, and rely on personal contact between developers and potential users. Requirements prioritization techniques (e.g. Card sorting, EasyWinWin, and Delphi methods) are resource-centric and based on analysis of, and choice amongst identified requirements vis-à-vis project constraints including cost, technologies, skills and time. Requirements experimentation techniques (e.g. prototyping) are software-centric, employ designs of the artifact as a key means for communication with users, and facilitate learning and joint problem solving amongst developers and users. Requirements specification techniques (e.g. entity relationship modeling) are documentation-centric, and hinge on explicit and agreed upon basis for continued development. According to Mathiassen et al. (2007), discovery techniques should be employed to address requirement identify risks, prioritization and experimentation techniques should be employed to address requirement volatility risks, and prioritization and specification techniques should be employed to address requirement complexity risks. However since requirements might have risk profile consisting of a combination of different degrees of identity, complexity, and volatility risks; Mathiassen et al. (2007) recommended that a combination of requirement development techniques should be used together with compensating actions to curb adverse effect on risks other than the ones targeted by a particular technique. Conflicting requirements from stakeholders can be resolved through collaborative decision making processes (e.g. consensus building) that is described in section 3.3.1 above.

3.3.5 **Dynamic Capabilities**

According to Winter (2003, p. 991), an organizational capability is "a high-level routine (or collection of routines) that, together with its implementation input flows, confers upon an organization's management a set of decision options for producing significant output of a particular type" . It comprises of "individual skills, tacit forms of knowledge, and social relations that are embedded in a firm's routines, managerial processes, forms of communication and culture" (Pandza, Horsburgh, Gorton, & Polajnar, 2003, p. 1011). Winter (2003, p. 991) defines routines as "behavior that is learned, highly patterned, repetitious, or quasi-repetitious, founded in part in tacit knowledge—and the specificity of objectives". Capability can also imply the capacity to carry out an activity in a reliable manner and as such, a capability should have intended purpose, be used to carry out an activity, and enable reliable and repeated performance (Helfat & Winter, 2011).

Firms do possess capabilities that reflect their ability to perform basic functional activities (Collis, 1994), enable them to solve problems and develop new products (Zahra, Sapienza, & Davidsson, 2006), and are dedicated to how the firm "earns a living now" (S. G. Winter, 2003). These capabilities are called "first category" capabilities(Collis, 1994), substantive capabilities (Zahra et al., 2006), zero-order, ordinary or operational capabilities (S. G. Winter, 2003). They are exercised in equilibrium or stationary state of the firm (S. G. Winter, 2003) and reflect the resource base of the firm (Ambrosini, Bowman, & Collier, 2009). Such capabilities include plant layouts, marketing campaigns (Collis, 1994), and ways to develop new products (Zahra et al., 2006).

Unlike operational capabilities, dynamic capabilities are associated with the ability to adapt to changes and survive in turbulent or dynamic environments. Dynamic capabilities are the ability to change, reconfigure (Zahra et al., 2006), extend, modify, or create operational/substantive capabilities (Drnevich & Kriauciunas, 2011). They dynamically improve the activities of the firm (Collis, 1994) e.g. the production process, and cause changes in the product, the scale or the customers (S. G. Winter, 2003). Dynamic capabilities refer to the firm's ability to build, integrate and reconfigure internal and external competencies to address rapidly changing environments (Teece et al., 1997, p. 516). These dynamic capabilities consist of knowledge creation and absorption, knowledge integration, and knowledge reconfiguration (Verona & Ravasi, 2003). Similarly, Zollo & Winter (2002) argued that dynamic capabilities are shaped by experience accumulation, knowledge articulation and knowledge codification. Also, Barnett, Presley, Johnson, & Liles, (1994) claimed that the move towards agility depends on the ability of the enterprise to respond to unanticipated changes by leveraging on its knowledge. These knowledge are accumulated overtime, and stored in organization's procedures, norms, rules, and forms (J. G. March, 1991). Information technologies have played significant role in the development and sharing of organizational knowledge. For instance, Kane & Alavi (2007) discovered that IT-enabled learning mechanisms, including knowledge repositories of best practices, enable capabilities that have distinct effect on exploration and exploitation learning dynamics in organizations. Enterprise architecture can contribute to organizational knowledge through disciplined production and management of enterprise definition that cumulates into a body of enterprise knowledge (Zachman, 1996). In line with earlier arguments, enterprise knowledge increases enterprise sophistication and resilience to high rates of enterprise change over time (Zachman, 1996). For instance, in Verona & Ravasi (2003), the authors noted that knowledge integration and related business processes were central to Oticon's capacity to innovate continuously.

According to Pavlou & El Sawy (2011), dynamic capabilities can be grouped into four categories namely sensing capability, learning capability, integrating capability and coordinating capability. Sensing capability is the ability to spot, interpret, and pursue opportunities in the environment for instance, the ability to generate, disseminate and respond to market intelligence. They define learning capability as the ability to revamp existing operational capabilities with new knowledge through acquisition, assimilation, transformation and exploitation of knowledge. Integrating capability, according to Pavlou & El Sawy (2011), is the ability to embed new knowledge into the new operational capabilities by creating shared understanding and collective sense-making. This category of dynamic capability consists of basic routines including contribution of individual knowledge to group, representation of individual and group knowledge, and interrelation of diverse knowledge inputs to the collective system. Lastly, they define coordinating capability as the ability to orchestrate and deploy tasks, resources, and activities in the new operational capabilities by assigning resources and appointing right persons to right task, and identifying synergies among tasks activities and resources. The relationships amongst these types of dynamic capabilities are illustrated in Figure 3 below.

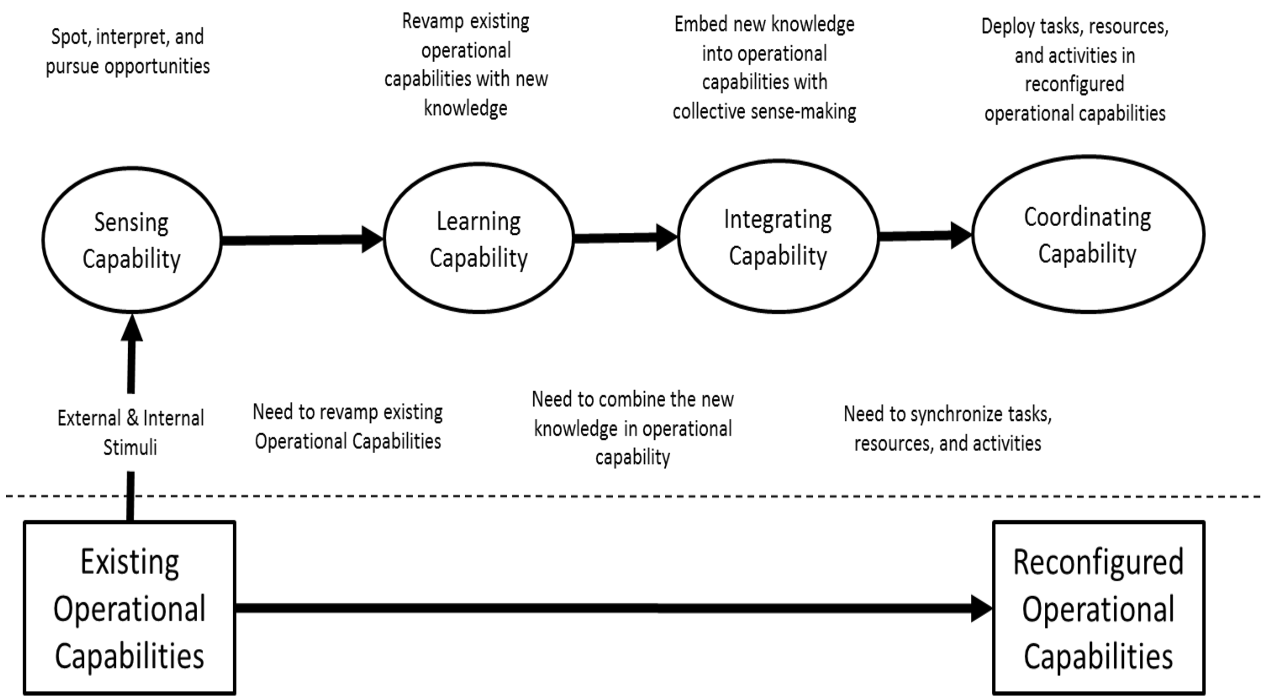


Figure 3: "A framework for representing the proposed measurable model of dynamic capabilities" Source: Pavlou and El Sawy (2011)

3.4 Critical Success factors of Enterprise Architecture Management

Very important to the conceptual model for this thesis is the construct "Critical Success Factors of EAM". The completeness of this construct ensures that the "Essential Elements of an Agile EAM Method" construct that will later be constructed in this thesis, adequately incorporates known CSFs of EAM. Therefore, a systematic literature review (as explained in section 4.5 under research methodology) was employed to identify the CSFs of EAM. Twenty CSFs of EAM were identified from 10 publications. These CSFs are briefly explained below;

- **Interlacing planning and execution**
  Architecture planning determines what, where, when and how resources will be employed to achieve strategic and tacit goals (Carl & Colombi, 2007). Architecture execution on the other hand is concerned with carrying out activities, and making the investments that lead to the planned goals. Carl & Colombi (2007) assert that though planning governs execution, in contemporary businesses environment, planning and execution steps are blurring such that one cannot be ac-

complished without the other. In this regard, planning cannot be done completely before execution; they need to be interlaced.

- **Scoping and purpose**
  The scope and purpose of an architecture endeavor establishes the extent of coverage of the architecture endeavor, and states explicit goals and reasons for the architecture endeavor. The "scope and purpose" answers the question "what should be built and why?"(Keil, Cule, Lyytinen, & Schmidt, 1998), and provides a clear sense of demonstrable value and direction for the architecture endeavor (Ylimäki, 2008a). Defining scope and purpose at the beginning of an architecture endeavor will assist in identifying risks associated with the architecture project (Keil et al., 1998) and increase the chances of success of the EA endeavor (C.-S. Cho & Gibson Jr, 2001). However, defining clear scope and purpose prior to the commencement of architecture project is by no means a trivial task. Scope and purpose are likely to get clearer and clearer to all stakeholders as the project progresses (Keil et al., 1998).

- **Clearly defined vision and goals/strategies**
  Architecture vision establishes the high-level and long-term strategies of the architecture endeavor and communicates the capabilities to be developed and how these capabilities will benefit the enterprise and stakeholders (TOGAF, 2009, p. 83). Architecture visioning is done at the strategic level (Pulkkinen, 2006) and it is indicative of top management commitment and support (TOGAF, 2009, p. 82; Zhong Liu & Seddon, 2009). The architecture vision also establishes the alignment between business and IT strategies (Zhong Liu & Seddon, 2009), and provides the overall EA management function a direction (Schmidt & Buxmann, 2011).

- **Commitment and sponsorship (especially from top management)**
  Commitment and sponsorship has been identified by 9 of the 10 sources reviewed as a factor that is critical to the success of enterprise architecture management endeavors. Commitment is needed from all stakeholder groups especially from top management and should be demonstrated through participating in and resourcing architecture endeavors (Lange et al., 2015; Zhong Liu & Seddon, 2009), and using ensuing EA products (Lange et al., 2015).

- **Development methodology and tools**
  In enterprise architecture management, well-structured and documented methodologies and tools; example frameworks, development methods and processes, principles, and decision mechanisms; are needed to plan, design and create architectural products and artifacts (Ylimäki, 2008a). The availability and enterprise awareness of these tools and methodologies, and especially the architect's in-depth knowledge of these tools and methods (Van der Raadt, Bonnet, Schou-

ten, & Van Vliet, 2010), have been identified as a critical success factor of enterprise architecture management endeavors (Lange et al., 2015; Ylimäki, 2008a; Zhong Liu & Seddon, 2009).

- **Quality of EA products (e.g. models and artifacts)**
  Enterprise Architecture products can be seen as the outcome of the enterprise architecture management function; for example, as-is architecture, to-be architecture, capability models, and roadmaps (Lange et al., 2015). The quality of such EA products is the degree to which the products meet stakeholder needs. According to Närman, Johnson, & Nordström (2007), EA product (e.g. systems) quality attributes include maintenability, security, reliability, efficiency, interoperability, suitability, accuracy and usability. EA product quality has been empirically proven to directly influence EA organizational and project benefit (Lange et al., 2015), and also contribute to agility (Van der Raadt et al., 2010).

- **Effective project & programme management**
  Project management involves the application of knowledge, skills, tools and techniques to set of activities to delivered required project outputs (Zhong Liu & Seddon, 2009). Programme, on the other hand, involves coordination of a group of projects in a manner that brings benefits that would otherwise not be possible if the constituent projects were managed independently (Ferns, 1991). Programme management therefore provides coordinated support, planning, prioritization and monitoring of projects to meet changing enterprise needs (Ferns, 1991), and thereby bridges project delivery and enterprise strategy (Lycett, Rassau, & Danson, 2004). Also, from an empirical research, Schmidt & Buxmann (2011) found that EA programming makes the largest contribution to the overall success of EA management. Furthermore, Ylimäki (2008a) asserted that EA endeavors are usually project based and therefore require effective project and programme management methods to be successful.

- **Skilled team, training and education**
  Skilled team, training and education have been identified by 8 out of 10 of the reviewed papers to be a critical success factor for enterprise architecture management. For instance, Seppänen (2014) concludes that lack of skills and knowledge regarding enterprise architecture has the most (negative) effect on an organization's capability to successfully carry out enterprise architecture management. Other authors have identified the project manager's kills(e.g. Zhong Liu & Seddon, 2009), architect's skills (e.g. Van der Raadt et al., 2010), and skills of the entire architecture team (e.g. Ylimäki, 2008a) as critical to the success of enterprise architecture management.

- **Communication and common language**
  Communication in enterprise architecture refers to the process of informing stakeholders about EA related issues (Schmidt & Buxmann, 2011) for example, EA decisions, designs, and processes. However, considering the multitude of different stakeholder groups with different perspectives in EA management (Kaisler et al., 2005), effective communication can only ensue when there is a common frame of reference that establishes common terminologies that all stakeholders can understand (Carl & Colombi, 2007; Ylimäki, 2008a). Carl & Colombi (2007) explained that in Department of Defense (DoD), terminologies, acronyms and abbreviations are defined at the beginning of EA management endeavors to establish common language across the enterprise environment.

- **Business driven approach**
  The essence of EA is to link all components of the enterprise (e.g. business and Information systems) to drive enterprise strategies and goals (Ylimäki, 2008a), and to define the rules that govern the evolution of the components (TOGAF, 2009). It is therefore imperative for EA management efforts to take into consideration the extent to which business strategies and requirements are covered (Ylimäki, 2008a).

- **Assessment & evaluation**
  Architecture products and artifacts, and the processes that are used to create the products and artifacts need to be assessed and evaluated as part of EA management endeavor. The assessment and evaluation of architecture products, artifacts and processes enables the enterprise and the architecture team to determine the fit between the products, artifacts and processes on one hand, and enterprise strategies, architecture goals and purpose on the other. However, Ylimäki (2008a) noted that performing assessment and evaluation is a non-trivial task probably because the effect of architecture products and artifacts (e.g. architecture decisions) might manifest years after they are produced. This assertion is supported by an empirical research in which some interviewees expected the effects of EA to be realized within no less than 5 to 10 years (see Schmidt & Buxmann, 2011).

- **IT investment and acquisition strategy**
  According to Ylimäki (2008a), the "IT investment and acquisition strategy" CSF refers to the extent to which EA informs IT strategy. It reflects the relationship between architecture development and governance processes on one hand, and the IT investment and acquisition processes and decisions on the other (Ylimäki, 2008b). Indeed, Kamogawa & Okada(2005) noted that EA governance should, amongst other things, consist of an IT investment management framework. Also, the Architecture Capability Maturity Model (ACMM) of the US Department of Commerce's (DoC) enlists IT investment and acquisition as part of the nine elements of enterprise architecture (TOGAF, 2009, p. 686).

- **Organizational culture**
  Organizational culture has been identified as an important CSF for EA management (e.g Kamogawa & Okada, 2005; Ylimäki, 2008a), and has been found to play significant role in grounding EA within the enterprise (Aier, 2014; Lange et al., 2015). For instance, Aier (2014) noted that organizational culture moderates all EA Principle (EAP) mechanisms; EAP grounding, EAP management, EAP guidance, and EAP Application. Similarly, Lange et al.(2015) found that "EA management organizational anchoring" mediates the effect of success factors including "EA management infrastructure quality" and "EA service delivery quality" on "intention to use EA management" and "EA management organization and project benefits". Ylimäki (2008a) suggests that EA endeavors should not be restricted to departments or silos, but they should be discussed across the enterprise.

- **EA governance**
  EA governance provides mechanisms by which EA and other architectures are managed at the enterprise level (TOGAF, 2009, p. 671). These mechanisms could include control processes, rules, standards, roles and organizational structure (Kamogawa & Okada, 2005). In this regard, EA governance is implemented at various levels of the enterprise (Ylimäki, 2008a), and is necessary for long term maintenance of enterprise architecture (Kamogawa & Okada, 2005). For instance, TOGAF recommends that EA strategic plans should be closely linked to the architecture governance process to curb circumvention of design decisions for tactical convenience (2009, p. 700). Furthermore, Seppänen (2014) argues that the success of EA initiatives depends on the governance model of the enterprise, and the guidance that the governance model provides to the transformation of the enterprise.

- **Change management**
  Change management is believed to be a structured approach that enables the transition from the current state to a desired future state (Finney & Corbett, 2007; Yang & Seddon, 2004), and therefore is inevitable in enterprise architecture management. In fact, Aladwani(2001) asserts that improvement strategies commonly involve change. These changes may occur whilst architecture teams are designing, developing and implementing systems and application (Kaisler et al., 2005), or when there are strategic shifts (e.g. mergers and acquisitions, and new product/service innovation) in the enterprise causing changes in desired capabilities and architecture requirements. In both scenarios, requirements rarely remain constant; mandating the employment of appropriate change management processes and mechanisms (Carl & Colombi, 2007). Yang & Seddon (2004) assert that initial implementation and subsequent upgrade projects require considerable change management efforts to overcome organizational inertia.

- **Stakeholder participation and involvement**
  Stakeholder participation has been defined as the extent to which EA stakeholders *are involved* (or better, do actively take part) in EA management decision making (Schmidt & Buxmann, 2011). Participation leads to understanding of the domains that the architecture represents; enables stakeholders to engage in discussions about the evolving architecture and the thinking that lies behind the EA plans (Carl & Colombi, 2007); and creates awareness of the latest developments in EA (Kamogawa & Okada, 2005). Stakeholder participation is antecedent to stakeholder involvement (Barki & Hartwick, 1989). According to Barki & Hartwick(1989), stakeholder involvement is a subjective psychological state and that a user (or a stakeholder for that matter) is involved when he or she considers a system to be important and personally relevant. For example, Lange et al.(2015) found that when stakeholders continue to participate in EA management, their perception of organizational and project benefits are elevated. Similarly, Schmidt & Buxmann (2011) found that stakeholder participation underlines the factors that promote the long-term acceptance and legitimation of the EA management function.

- **Clearly stated EA principles**
  EA principles are the basic philosophies that guide the development, and provide rational for constant examination and re-examination of the architecture and technology plan of an enterprise (Richardson et al., 1990). They are often based on business principles, and provide the foundation for architectural governance (TOGAF, 2009, p. 78) and architectural decision making (Richardson et al., 1990). In TOGAF, architecture principles are identified and established at the preliminary stage, and confirmed and elaborated in later stages (2009, p. 76,85). Schmidt & Buxmann(2011) observed that, usually most enterprises establish architecture principles, standards and reusable reference architectures to guide change projects. Richardson et al.(1990) also found that explicitly stating architecture principles at Star Enterprise resulted in an integrated and flexible architecture. On the contrary, an empirical research by Lange et al.(2015) shows that architecture principles have no effect – neither positive nor negative – on EA success. They however recommended further research into this relationship.

- **Enterprise awareness (cognition) of EA**
  Enterprise awareness (cognition) of EA has been identified to be critical to the success of EA management(e.g. see Kamogawa & Okada, 2005; Lange et al., 2015). It is the extent to which members of the enterprise are aware, understand and identify with the EA processes, products, principles, and mechanisms within the enterprise. EA awareness is believed to be a very influential factor that brings about the values of EA (Kamogawa & Okada, 2005). For instance, Lange et al. (2015) found that, without enterprise awareness of EA management, neither

"EAM infrastructure" nor "EAM service delivery" will yield immediate organizational or project benefits, or lead to continuous engagement with EAM. Furthermore, creating stakeholder awareness of EA, especially for senior management, is of high priority (Kamogawa & Okada, 2005) and is needed to establish an appropriate level of "EAM organizational anchoring" and "actual use" to drive EA success (Lange et al., 2015).

- **Integrated EA repository**
  EA repository is an EA management tool that is used to store descriptions (e.g. catalogs, matrices, and diagram) of EA outputs and artifacts at different levels of abstraction within an enterprise (TOGAF, 2009, p. 12,14). Implementing an integrated EA repository keeps modeling efforts low by using data from existing architecture products and artifacts (R. Fischer, Aier, & Winter, 2007). For example, Winter & Fischer (2006) reported that, one of their case companies relied strongly on using integrated EA repository that contained information on existing models, and thereby limited modeling efforts to only aspects that were not covered in existing models. Also, Fischer et al.(2007) proposed a federated approach, in which an EA repository is used to store a copy of model data from specialized architectures, to reduce the challenges related to keeping models up-to-date. Also Richardson et al.(1990) reported that inventory (repository) that holds a collection of as-is architecture is one of the important components of EA. Indeed, ideally, EA artifacts should be managed within an integrated EA repository (Schmidt & Buxmann, 2011). However, Schmidt & Buxmann (2011) noted that only few organizations base architectural descriptions on a common meta-model and store them in an integrated repository.

- **Adequate documentation (e.g. of ″as-is″ and ″to-be″ architecture)**
  According to Schmidt & Buxmann (2011), EA documentation refers to " the process of capturing and describing the existing EA by means of architectural description" (2011, p. 147). The documentation of the existing (as-is) architecture provides the bases for planning the target (to-be) architecture, and developing the transition plans that moves the enterprise from the current state to the desired state. For instance, during a pre-study Schmidt & Buxmann (2011) discovered that many enterprises based the planning of their to-be architecture on their as-is architecture. According to them, adequate EA documentation aids rational EA management, improves decision-making during planning and implementation, fosters reuse, and prevents redundancies. However, the high amount of artifacts in an EA (Roth, Hauder, Farwick, Breu, & Matthes, 2013), the complexity of the enterprise, and the high rate of change that the enterprise and its components undergo (Farwick, Breu, Hauder, Roth, & Matthes, 2013), make EA documentation difficult.

3.5 **Chapter Summary**

In conclusion, the Lean and Agile principles and values from extant literature were presented in this chapter. For each of the Lean and Agile principles and values, theoretical glues were suggested through which it could be appropriated for EAM. The theories include collaborative decision making, participatory design, organizational fitness, contingency based requirement management and dynamic capabilities. These theories which are referred to as "probable theoretical glue", were presented with concentration on how they relate to enterprise agility and its antecedents for instance the agility drivers and the agility capabilities identified in the previous chapter. Also, a justification based on literature was presented, for the quest to combine Agile and Lean. It was noted that, though the combination of Lean and Agile is not so mature, it provides an opportunity for the Lean and Agile concepts to complement each other. Lastly, the Critical Success Factors (CSFs) of EAM that were identified through systematic literature review (as explained in section 4.5 under research methodology) were briefly presented.

Fortified with knowledge from chapter two and chapter three, the next step within the DSRM process model is to design and develop the artifact; here the method. Chapter five is dedicated to the first iteration of the design effort, that is, the design and development, demonstration and evaluation of the Essential Elements of an Agile EAM Method. Chapter six is dedicated to the second iteration of the design effort; that is, the design and development, demonstration, and evaluation of the Agile EAM Method. However, the next chapter, Chapter four presents the research methodology; following the publication schema for a design science research study (Gregor & Hevner, 2013).

4 CHAPTER 4 RESEARCH METHODOLOGY

The research methodology chapter presents research processes that are employed towards answering the research questions posed in chapter one. These include the research processes used for conducting the literature review, constructing and evaluating the design artifacts (the Essential Elements of an Agile EAM Method, and the Agile EAM Method), and representing and communicating the research process (writing the thesis). Further, the chapter provides justifications for the selection of the various methods; for instance, Design Science Research (DSR) Method, qualitative interview method, and illustrative Scenario as a DSR artifact evaluation method; employed during the research process. The Conceptual Model for the thesis, that explicates the relationship amongst all the various constructs developed in the thesis, is also presented.

4.1 Literature Review

Chapters two and three are dedicated to the review of literature on the two concepts of the thesis, and the establishment of the theoretical foundation for design work later in the research process. Chapter two presents the two major concepts of this thesis; enterprise agility and enterprise architecture. Research papers on enterprise agility are selected from manufacturing, management, new product development, and information systems journal, for review. The aim is to find answers to research question 1. Further, research papers on enterprise architecture are reviewed with particular focus on EA, Enterprise Architecting, and Enterprise Architecture Management and the relationship amongst these three; and other salient issues in EA especially EA governance, EA principles, EA planning, and EA frameworks, levels and domains. Prior work, especially within the academia, that are dedicated to developing an agile EAM method, an agile EA, or components (elements) for EAM are reviewed to develop a list of preliminary essential elements of an agile EAM method.

Chapter three presents Lean and Agile principles and values from extant literature and suggests some theories that could be employed to adapt the Lean and Agile principles and values to Enterprise Architecture Management (EAM). These theories form the "theoretical glue" that underpins the design constructs in order to make them relevant in the organizational context. The chapter also presents the Critical Success Factors (CSFs) of EAM.

4.2 Selection of research method

This thesis aims at producing an information systems design artifact in the form of a method (Gregor & Hevner, 2013; S. T. March & Smith, 1995; von Alan et al., 2004) that intends to provide guidance on how to perform enterprise architecture agilely. In this regard, the thesis employs a research method that follows the guidelines for design science research methodology in information systems research proposed by Peffers et al (2007). The research process includes problem identification within the target environment; specification of motivation and objective for research; review of literature; design of artifact (an agile EA management method); demonstration of the agile EA management method by means of an illustrative scenario analysis; evaluation of result; and finally communication of the research process and results in the form of a master's thesis.

The design science research approach has been criticized by some authors, e.g. Sein et al.(2011). According to Sein et al.(2011), design science research favors technological rigor over organizational relevance and fails to recognize that artifacts evolve from interaction with the organizational context even when the initial design is guided by the researcher's intent. However, although Hevner et al. (2004) note that overemphasis on rigor can lessen relevance; they argue that it is possible for design science research to achieve rigor and relevance. Also, livari (2007) suggests that the focus of action theory is more on improvement problems and adopting technology rather than on construction problems and building technology.

Furthermore, Iivari & Venable (2009) assert that, the widely used definition of action research espoused by Rapoport (1970) assumes the involvement of a concrete client and therefore makes the design process contextual and focused on addressing a specific client's concern. The design effort of this thesis does not have a specific client in mind. Iivari & Venable (2009) further argues that action research and design science research should be kept separate because of ethical problems and pragmatic reasons, especially when design science research is aimed at constructing and testing innovative artifacts. In the case of this thesis the artifact is an agile EAM method. They however recommended that action research could be used for naturalistic evaluation of a solution technology invention example, creation of a method. Also the research adopted the methodology proposed by Peffers et al (2007) because it is compatible with the ontological perspectives that form the basis of the design research presentation style that this master's thesis adopts (see Gregor & Hevner, 2013).

4.3 Conceptual Model for the Research

The conceptual model for this research draws on two prior research conceptual models: "the Critical Success Chain" by Peffers et al. (2003) and "how project critical success factors (CSF) affect organizational benefit of enterprise systems (OBES)" by Zhong Liu & Seddon(2009). Peffers et al. (2003) proposed the critical success chain (CSC) as a means to link IS attributes to critical success factors (CSF) and CSF to organizational goals. The CSF is based on Personal Construct Theory (PCT) and is aimed at increasing the participation of potential users in IS planning. By means of two case studies, the authors modeled CSCs by establishing explicit relationships among IS attributes, CSF, and specific firm goals. They also incorporated implicit importance (see Varva, 1997, p. 386) relationships in their models. The CSC model is represented in Figure 4 (fig1a).

However, Robey et al.(2002) criticized the relationship between CSF and organizational benefits. According to them, most studies do not provide theoretical justifications that explain why the investigated project or business outcomes occur. Zhong Liu & Seddon(2009) attempted a response to this criticism by developing a conceptual model; "how project CSF affect OBES", that proposes that the effects of CSFs on enterprise benefit from enterprise system (ES) use are mediated by benefit drivers (outcome of the ES implementation processes). Liu & Seddon(2009) assumed that the findings of Seddon et al.(2007), that established the relationship between the benefit drivers and enterprise benefits were right, and proceeded to investigate the relationship between the CSFs and the benefit providers. The benefit providers include functional fit, overcoming organizational inertia, and delivering working system. However, Liu & Seddon(2009) suggested further studies into identifying other benefit drivers that mediate CSFs and enterprise benefits from ES use. The "how project CSF affect OBES" model is represented in Figure 4 (Fig1b). By combining these two conceptual models, it is reasonably safe to establish a link between IS/ES attributes and CSFs, and then from CSFs through benefit drivers to enterprise benefits as represented in Figure 4 (Fig1c).

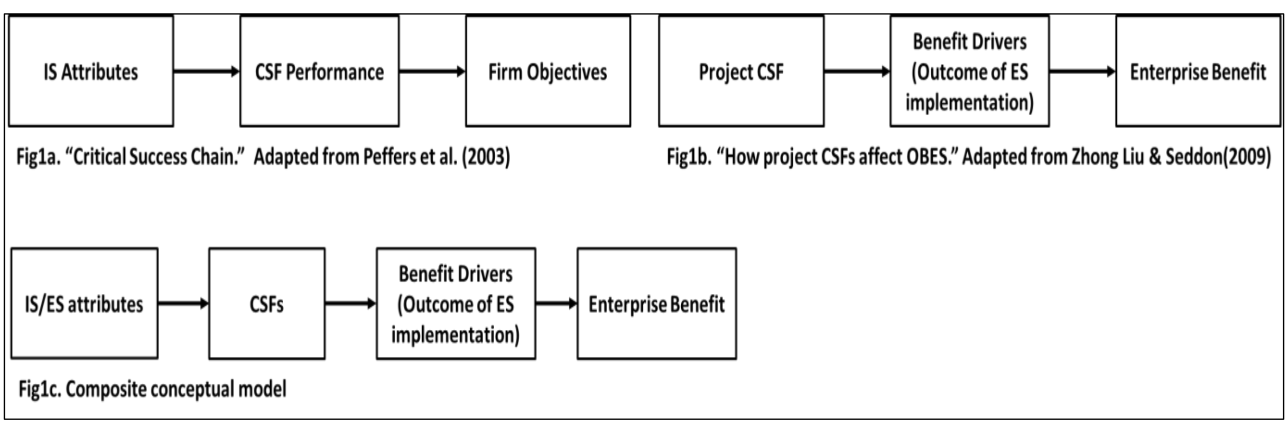


Figure 4: Constructing the conceptual model

The conceptual model for this thesis adapts the "composite conceptual model" in Figure 4 (Fig1c) to enterprise architecture management methods. The resultant conceptual model is represented in Figure 5 below. The first three constructs: foundation theory, adapted agile principles and values for EAM, and adapted Lean Principles and wastes for EAM, present the theoretical lenses throw which the Essential Elements of Agile EAM method construct was proposed. The Essential Elements of Agile EAM method, Critical Success Factors of EAM, EAM Mechanisms and Processes, and Enterprise Agility constructs map respectively onto the IS/ES attributes, CSFs, Benefit drivers, and Enterprise benefit constructs of the composite conceptual model in Figure 4 (Fig1c). The individual constructs are briefly explained below.

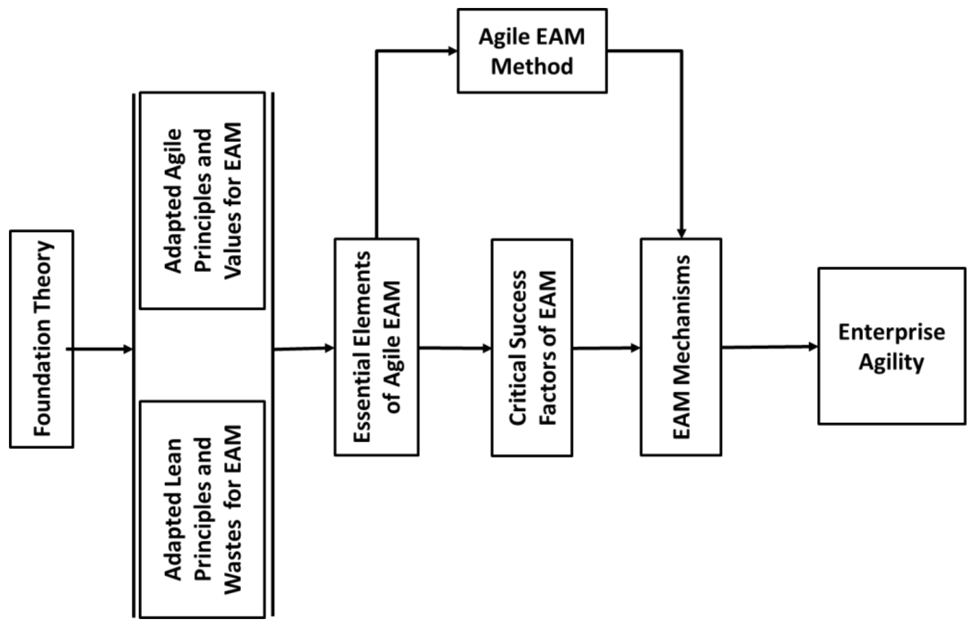


Figure 5: Conceptual Model for the Research

I. Foundational Theory: This construct represents the theoretical lenses that the author employed to relate agile principles and values, and lean principles and wastes to agile enterprise architecture management method. The construct consists of literature reviewed in chapters 2 and 3.

II. Adapted agile principles and values: The adapted agile principles and values construct represents the EAM equivalent of the agile principles and values in the agile manifesto.

III. Adapted lean principles and wastes: This construct represents the EAM equivalent of lean principles and wastes

IV. Essential Elements of an Agile EAM Method: These are the basic attributes or essential elements that should guide the construction and appropriation of agile EAM methods to support the agility of the EAM function and enterprise agility. These attributes should lead to the fulfillment of the CSFs needed for EAM to succeed.

V. CSF of EAM: The CSF of EAM construct provides a list of known critical success factors of EAM that underpins the success of the EAM function.
VI. Agile EAM Method: The agile EAM method is the main design artifact for this thesis. It is informed by the Essential Elements of an Agile EAM Method construct, and provides mechanisms and procedures that mediate the CSFs of EAM and the Enterprise Agility constructs.
VII. EAM mechanisms: This construct consists of mechanisms and procedures that are based on the Essential Elements of EAM constructs and provides the means to attain the "benefit enablers" that mediate between CSFs of EAM construct and the enterprise agility (enterprise benefits) construct.
VIII. Enterprise Agility (Enterprise Benefit): This construct represents the main target effect of the design artifact (the Agile EAM Method) on the enterprise.

4.4 The DSRM Process Model and the Structure of this Thesis

Peffers et al (2007) describe a process model for carrying out, presenting and evaluating Design Science research in Information Systems, called the DSRM Process Model. The process model consists of six constructs representing activities that should be carried out during a Design Science research. The activities include problem identification and motivation; define the objectives for a solution; design and development; demonstration, evaluation; and communication. During the problem identification and motivation activity, the researcher acquires knowledge of the state of the problem and value of the solution, and clearly defines the research problem and justifies the value of the solution. In activity two, the researcher defines the objective of the solution taking cognizance of the available solutions and their efficacy at addressing the research problem. In the third activity, the researcher acquires knowledge of the relevant theories that come to play in determining the functionalities and the architecture of the artifact, and then proceeds to design and develop the actual artifact. During activity four, the researcher uses the artifact to solve a problem through an appropriate method, for instance case study and during activity five the researcher observes and measure the extent to which the artifact was able to support a solution to the problem. And finally, in activity six, the researcher communicates the research process and its outcomes in a manner that is appropriate for the audiences. The DSRM process model is iterative and allows four major entry point including; problem-centered initiation, objective-centered solution, design and development centered initiation, and client/context initiated.

Further, based on Peffers et al (2007), Gregor & Hevner (2013) suggested a schema for presenting and communicating DSR outputs to the design community. They suggest that, a typical communication of the DSR output should include introduction, literature review, method, artifact description, evaluation, discussion and conclusions. The introduction section (chapter) should define the problem, the motivation and significance of the problem, the research questions, overview of methods and the theoretical and practical significance of the solution. Review of prior work including prior design and knowledge relating to the problem to be addressed should be covered in the chapter on literature review. Then, the research approach that was employed during the DSR should be explained in the chapter on research methods. In order to make a new contribution to the knowledge base, the design artifact should be described in a concise manner with appropriate level of abstraction. This description should be the content of the chapter on artifact description. Thereafter, the evaluation of the artifact should be communicated in the chapter on evaluation. The discussions chapter should be dedicated to the interpretation of the results in relation to the objectives of the DSR work and the communication of any limitations related to the research process and results. Finally, conclusions chapter should restate the important findings of the work and why they are important.

This thesis took a problem-centered initiation to the DSRM process model (Peffers et al., 2007) and followed through the process model as illustrated in Figure 6. Also the communication schema suggested by Gregor & Hevner (2013) was used to structure the communication of the DSR (this thesis) with modifications made to accommodate the iterations in the DSR process. Consequently, chapter 1, the introduction chapter, is dedicated to "problem identification and motivation", and "define objectives" activities of the DSR process model. Furthermore, chapter 2 and chapter 3 are dedicated to review of relevant literature related to the objectives of the thesis including prior work, theories, empirical research, and practitioner reports and guide books. Chapter 2 and chapter 3 provide the theory on which the "design & development" activity in the DRM process model is based. The research intends to produce intermediate artifacts including the adopted Agile and Lean principles and values, and the eight EEs of an Agile EAMM. These intermediate design artifacts are proposed in chapter 5. In this regard, chapter 5 aligns to the "design and development" activity of the DSR process model.

Also, in chapter 5, the eight EEs of an Agile EAMM are evaluated by means of a semi-structured interview comprising of practitioners and researchers (Sub-section 5.4). Chapter 5 therefore further corresponds to the "evaluation" activity in the DRM process model and presents the first iteration in the DSRM process. Chapter 6 is dedicated to the construction of the Agile EAMM, representing the second iteration in the DSR process model as chapter 6 also corresponds to the "design and development" activity in the DSR process model. Further, chapter 6 demonstrates and evaluates the Agile EAMM by comparing its compatibility with existing solutions (artifacts), and by evaluating its efficacy through an illustrative scenario analysis (Peffers, Rothenberger, Tuunanen, & Vaezi, 2012). Therefore, chapter 6 corresponds to the "demonstrate" and "evaluation" activities of the DSR process model. Chapter 7 discusses the results from the thesis, draws conclusions, and makes recommendations for future research. All the chapters taken together form the "communication" activity of the DSR process model as represented in Figure 6

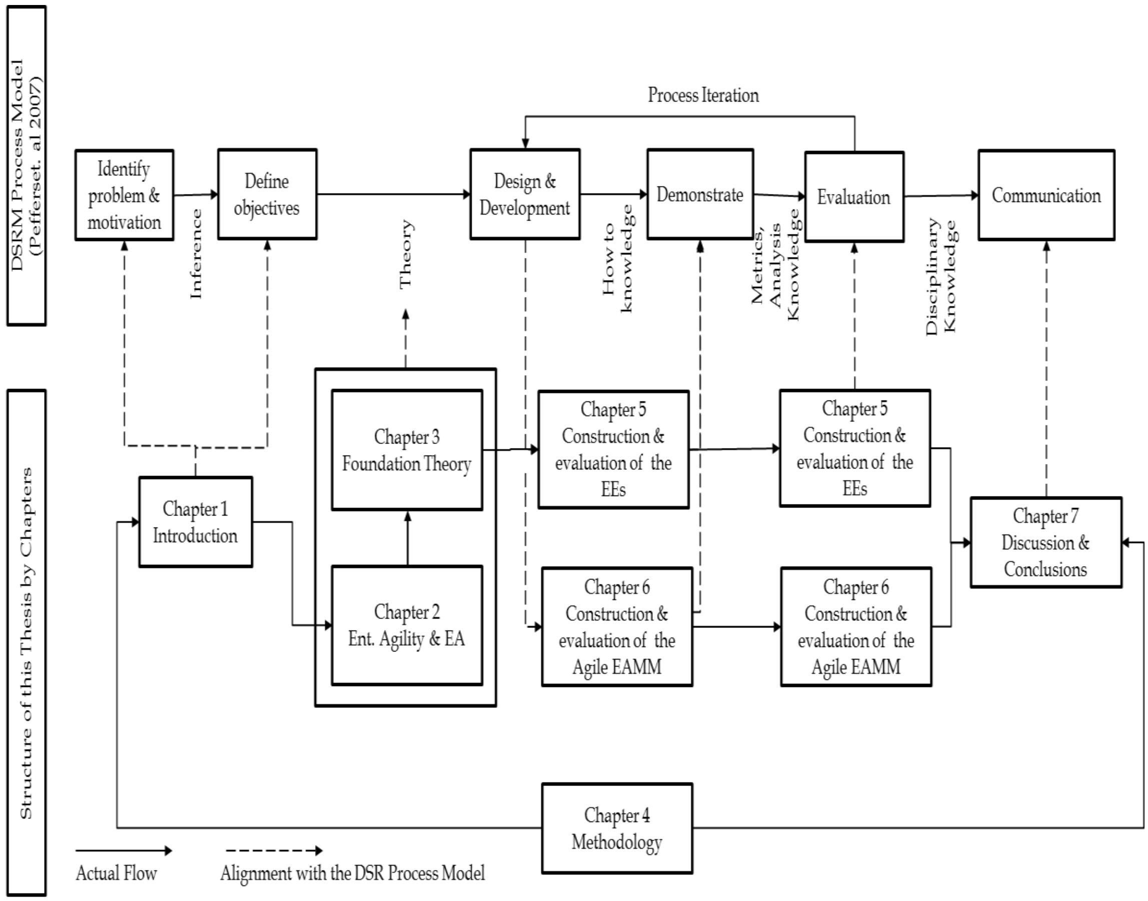


Figure 6: Alignment of the thesis chapters with the DSR Process Model in Peffers et. al (2007)

4.5 Critical Success Factors for Enterprise Architecture Management

The CSFs were identified by means of a systematic literature review. The process for the systematic literature review followed the step recommended by Okoli & Schabram (2010). The review was carried out between 16th May and 30th May, 2015. A summary of the steps are illustrated in the Table 9 below. Twenty (20) CSFs of EAM were identified from 10 publications. The CSFs of EAM are listed in Table 10 below.

Table 9: Summary of systematic research process to identify the CSFs of EAM

Steps	**Activities/**	**Outcome**
Purpose of the literature review	To answer the question: What are the critical success factors of Enterprise Architecture Management?	20 CSFs of EAM
Protocol and training	N/A	N/A

Searching for the literature	The search was conducted using the following search phrases: **1. IEEEXplore =** a) "Critical Success Factors" AND ("Enterprise Architecting" OR "Enterprise Architecture Management"); returned (11 results) b) Critical Success Factors of Enterprise Architecture (18 results)	29 results (with repetitions)
	2. Google Scholar = a) "Critical Success Factors" AND ("Enterprise Architecting" OR "Enterprise Architecture Management"); returned (270 results) b) "Critical Success Factors" AND "Enterprise Architecture": returned (252 results)	523 results (with repetitions)
	3. **Exploratory research –** e.g. by following citations from selected papers (returned I unique result)	1 unique result
Practical screen	Practical screening criteria employed include: i. Peer reviewed journal articles and conference papers[1] ii. Paper and journal article are written in English iii. Paper and journal articles are focused on Critical Success Factors or related topics in EAM	IEEEXplorer: 4 Results Google Scholar : 34 Results
Quality appraisal	Quality appraisal criteria employed include: i. Journal articles, books/book chapters, conference papers that could not be acquired freely within the duration of the search ii. Journal articles, books/book chapters, conference papers that do not contribute towards answering the research question iii. Journal articles, books/book chapters, conference papers that are not written in English iv. Introduction to conferences and editions of journals v. Repeated/duplicated papers	i. **IEEEXplorer:** 1 unique result ii. **Google Scholar**: 9 results (1 repetition with IEEEXplorer results) iii. **Exploratory Research**: 1 unique result iv. **Total** reviewed research papers = 10
Data extraction	Each of the 7 research papers were thoroughly read to identify the Critical Success Factors of EAM	20 Critical Success Factors were identified
Synthesis of studies	All the seven research are mapped against the 20 CSF as illustrated in table….	Comprehensive list of CSF for EAM (see Table 10 and section 3.4)
Writing the review		Process reported in section 4.5 and discussed in details in section 3.4

[1] With the exception of (Carl & Colombi, 2007)

Table 10: Critical Success Factors of Enterprise Architecture Management

	Intertwine planning and execution	Scopying and purpose	Clearly defined vision and goals	Commitment and sponsorship (esp. from top management)	Development Methodology and tools	Quality EA Product (e.g. models and arti-facts)	Effective project & programme Mgt.	Skilled team, training & education	Communication and common language	Business driven approach	Assessment & evaluation	IT investment & acquisition strategy	Organizational culture	EA governance	Change management	Stakeholder participation & Involvement	Clearly stated EA principles	Enterprise awareness (cognition) of EA	Integrated EA Repository	Adequate documentation (e.g. of "as-is" and "to-be" architecture	Research Method	
																					Literature Review	Empirical
(Ylimäki, 2008a)		X	X	X	X	X	X	X	X	X	X	X	X	X			X				X	X
(Yang & Sed-don, 2004)		X	X	X			X	X	X	X					X	X					X	X
(Carl & Colom-bi, 2007)	X	X	X	X					X						X	X					X	
(Kamogawa & Okada, 2005)				X	X			X	X			X	X	X		X		X			X	X
(Lange et al., 2015)		X	X	X	X	X		X	X				X	X		X	X*	X			X	X
(Nikpay, Selamat, Rou-hani, & Nik-fard, 2013)		X		X	X	X	X	X	X	X	X		X	X			X	X			X	
(Schmidt & Buxmann, 2011)		X	X	X	X		X		X	X				X		X			X	X	X	X
(Seppänen,		X	X	X				X		X			X	X							X	X

	Intertwine planning and execution	Scopying and purpose	Clearly defined vision and goals	Commitment and sponsorship (esp. from top management)	Development Methodology and tools	Quality EA Product (e.g. models and arti-facts)	Effective project & programme Mgt.	Skilled team, training & education	Communication and common language	Business driven approach	Assessment & evaluation	IT investment & acquisition strategy	Organizational culture	EA governance	Change management	Stakeholder participation & Involvement	Clearly stated EA principles	Enterprise awareness (cognition) of EA	Integrated EA Repository	Adequate documentation (e.g. of "as-is" and "to-be" architecture	Research Method	
																					Literature Review	Empirical
2014) new																						
(Van der Raadt et al., 2010) new**			X				X	X	X												X	X
(Aier & Schelp, 2010) new		X	X	X			X	X		X			X	X				X			X	X
Total	1	8	8	9		3	7	8	8	6	2	2	6	7	2		4	4	1	1	10	8

* This CSF discarded from factors analyzed by Lange, Mendling, & Recker (2015) but the author of this thesis retains it for reexamination

4.6 Evaluation of the Essential Elements of an Agile EAM Method

This sub-section describes how the suggested EEs of an Agile EAM method are evaluated. The suggested EEs are evaluated for completeness and for relevance. The evaluation for completeness is done by relating the EEs to extant literature, the CSFs of EAM, and the agility providers and agility capabilities of an enterprise. Also, through an interview process, the suggested EEs are evaluated for relevance, especially to practice. The two evaluation processes; evaluation of for completeness and evaluation for relevance, are discussed below.

4.6.1 Evaluation for Completeness

The proposed Essential Elements (EEs) of an Agile EAMM are evaluated against those that were found in extant literature to ensure there is a consensus (e.g. as done in Peffers et al., 2007) rather than a diverging effort in the quest to establish the EEs of an Agile EAMM, and to broaden its scope of use in the construction and appropriation of EAMMs. The evaluation is therefore done against; one, related works in EA literature; two, the CSFs of EAM; and three, the agility capabilities and agility providers of the enterprise. The evaluation for completeness does not however indicate the exhaustiveness of the EEs but rather, it provides an indication of the extent of harmony the EEs have with extant literature whilst contributing to establishing the link between EAM and enterprise agility. The evaluation for completeness is presented in 5.4.1

4.6.2 Evaluation for Relevance

In addition to the evaluation for completeness, the EEs of an agile EAM method are also evaluated for relevance (see Peffers et al., 2007; Hevner et al., 2004) by means of interviews involving key-informants (Bagozzi, Yi, & Phillips, 1991). Though Myers & Newman, (2007) might regard the use of key-informants in qualitative interview as an introduction of *"elite bias"*, the method encourages only people with requisite knowledge by virtue of their professional engagement or research endeavors to respond to the interview questions and share their experience during the process. Besides, the key-informant method has been used in Information Systems research (e.g. Nevo & Wade, 2011; Pavlou & El Sawy, 2011).

Qualitative interview is recognized as a means of gathering experiential data (Schultze & Avital, 2011) and has been used extensively in information systems research (Schultze & Avital, 2011; Sessions, 2007). In qualitative interviews, participants are engaged directly in a conversation that enables them to generate contextual, nuanced and authentic accounts of their experiences (Schultze & Avital, 2011). In this regard, the interview process that will be used in this thesis employs the "romantic approach" where the "researcher takes more than s/he gives", and the interview process is seen as conversational and an opportunity to construct meaning rather than to elicit facts (Schultze & Avital, 2011, p. 4).

Furthermore, since the interview aims at assessing the extent of relevance of proposed artifacts from literature review, there is the need to design a set of questions to which interviewees will respond. This set of questions is referred to as an *"interview guide"* (Bryman, 2012, p. 472) or an *"explicit framework"* (Schultze & Avital, 2011, p. 6) and lists the issues to be addressed during the interview. According to Schultze & Avital (2011), the use of the *explicit framework* is:

> "… in order to help the interviewee access their lived experience and reflect on it, effective interviewing methods should provide frameworks that structure the conversation in such a way that guides the interviewee through this introspective journey while honoring his/her freedom of thought and expression… As such, they [the structured frameworks] assist in the generation of rich data as the interviewees are guided in accessing multiple layers of experience" (2011, p. 5).

Also, Bryman noted that when a research is directed towards investigating fairly clear focus, the interview method is likely to be semi-structured one, guided by an *interview guide* to allow more specific issues to be addressed (2012, p. 472). This thesis therefore employs a semi-structured interview method will be guided by a set of questions aimed at assessing the extent of relevance of the EEs of an Agile EAM method.

4.7 **Design of the Agile EAM Method**

The construction of the Agile EAM method represents the second iteration in the DSRM process model. The Agile EAM method will be designed based on the acquitted knowledge from foundational theories, the EEs, and knowledge gained from the evaluation of the EEs. Therefore, the process embed the acquired knowledge in the mechanisms, processes and structures of the Agile EAM method in fulfillment of research question three: "how could these essential elements be conveyed in the form of a method?"

4.8 **Evaluation of the Agile EAM Method**

Design evaluation is widely accepted as an essential component of DSR (Hevner et al., 2004; Peffers et al., 2012, 2007). For instance, Hevner et al (2004) assert that the utility, quality and efficacy of the design artifact must be rigorously demonstrated by means of well executed evaluation methods. These evaluation methods could take one of different forms; for example case study, scenario analysis, prototyping, action research and technical experimentation (Hevner et al., 2004; Peffers et al., 2012), or a combination of different forms (Peffers et al., 2012).

Results from Peffers et al.(2012) provide a guideline on the evaluation methods to be used for specific DSR outputs. According to Peffers et al.(2012), technical experimentation, case studies and illustrative scenario analysis are largely used in the evaluation of methods as DSR outputs. They defined methods as "actionable instructions that are conceptual (not algorithmic)" (2012, p. 402). Whilst technical experimentation is the most used evaluation method amongst the top-three, its focus is on algorithm implementation and therefore not suitable for the evaluation of the method being designed in this thesis. Peffers et al.(2012) defined Case Study as the "Application of an artifact to a real-world situation, evaluating its effect on the real-world situation", and Illustrative Scenario as the "Application of an artifact to a synthetic or real-world situation aimed at illustrating suitability or utility of the artifact" (2012, p. 402). Based on these definitions, case study and illustrative scenario analysis are therefore suitable for evaluating the method being developed in this thesis. However, with regards to the scope of this thesis, in terms of schedule and volume, case study is less appropriate. The thesis therefore employs an illustrative scenario to evaluate the Agile EAM method. A synthetic enterprise situation described in Sessions (2007) was selected for the evaluation of the Agile EAM method mainly because, the two primary issues presented in the scenario were business – IT misalignment, and IT hampering business agility. The illustrative scenario analysis is presented in 6.6.2.

4.9 **Chapter Summary**

In this chapter, the methods that will be employed during the research and the justifications for such methods were presented. The conceptual model for the construction of the Agile EAM Method that explicates the relationship amongst all the various constructs developed in the thesis was also presented.

5 Chapter 5 Essential Elements of an Agile Enterprise Architecture Management Method

In Chapter 2, the thesis sought to answer research question 1 by investigating the factors that influence the agility of the enterprise namely; agility drivers, agility providers, and agility capabilities, and how IT contributes to the agility of the enterprise. The focus was then turned to finding answers to research question 2: "what essential elements or components should an enterprise architecture management method poses in order to support the agility of the EAM function and the agility of the enterprise as a whole?" A review of extant literature on EA and EAM was done to: identify the salient issues (e.g. Challenges of EA, EA planning and EA governance); understand the three major concepts (i.e. Enterprise Architecture, Enterprise Architecting, and Enterprise Architecture Management) and how they relate; and finally to collate essential elements or components that have been suggested for these three major concepts. Then in Chapter 3, Agile and Lean principles and values were reviewed. In order to make these principles and values applicable to EAM and the agility of the enterprise as a whole, some theories were suggested based on the focus of each principle and value. These suggested theories are to form the "probable theoretical glue" that relate the Agile and Lean principles and values to the essential elements of an agile EAM method, and the agility of the enterprise as a whole.

In this chapter, the theoretical foundations from Chapter 2 and Chapter 3 are put together to construct the Essential Elements of an Agile EAM Method. That is done by using the "probable theoretical glue" to adapt agile principles and lean principles for EAM; and consolidating the adapted agile and lean principles, and essential elements/components identified in extant literature into the Eight Essential Elements of Agile EAM Method. Further the suggested EEs of an Agile EAM are evaluated for completeness and for relevance. The evaluation for completeness is done by relating the EEs to extant literature, the CSFs of EAM, and the agility providers and agility capabilities of an enterprise. The evaluation for completeness is to make sure the suggested EEs relate back to literature as much as possible. Also, through an interview process, the suggested EEs are evaluated for relevance, especially to practice. As explained in Chapter 4

(Research Methodology), a semi-structured method is employed, guided by a structured set of questions. The chapter ends with presentation and analysis of the interview results.

5.1 Adapted Agile Principles and Values

The agility concept has gained consideration within the Enterprise Architecture community. For instance, Buckl et al.,(2011) suggest that to make the EAM function agile, SCRUM roles, activities and deliverables should be mapped to those of enterprise architecture management, and similarly Bente et al. (2012) suggest the use of interlocked SCRUM for EAM. Other authors (e.g. Armour & Kaisler, 2001; Pulkkinen & Hirvonen, 2005) do recognize the need for agility in EA however, they do not explicitly recommend the use of the agile values, principles or methods as-is from the agile manifesto. In this thesis, the agile values and principles as stated in the agile manifesto are adapted for Agile EAM through the lenses of the "probable theoretical glue" presented early on. The adapted agile values and principles for EAM are represented in Table 11 and Table 12 respectively.

Table 11: Adapted Agile Values for Agile EAM

Agile ISD Values	Foundation Theory	Agile EAM Values
Individuals and interactions over processes and tools	FT1: Collaborative decision making FT2: Participatory design	AV1: Individuals and interactions over frameworks, processes, and tools
Working software over comprehensive documentation	FT3: Fitness Theory FT4: Contingency based requirement management	AV2: Working architectural products and artifacts over excessive documentation
Customer collaboration over contract negotiation	FT1: Collaborative decision making FT2: Participatory design FT3: Organizational Fitness	AV3: Participative and evolving architectural design over signed architectural designs, requirements, and plans
Responding to change over following a plan	FT3: Fitness Theory FT4: Contingency based EA requirement management FT5: Dynamic Capability	AV4: Responding to change over strictly following a plan

Table 12: Adapted Agile Principles for Agile EAM

Agile Principles	Foundation Theories	Agile Principles adapted for EAM
Our highest priority is to satisfy the customer through early and continuous delivery of valuable software	FT3: Organizational Fitness FT4: Contingency based requirement management	**AP1:** Satisfy EA stakeholders through early, continuous and incremental delivery of working EA products and artifacts with respect to competitive basis (e.g. strategic timing, effectiveness, efficiency, and flexibility)
Working software is the primary measure of progress		
Deliver working software fre-		

quently, from a couple of weeks to a couple of months, with a preference for the shorter timescale		
Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage	FT3: Organizational Fitness FT4: Contingency based requirement management FT5: Dynamic capability	**AP2:** Embrace changing requirements to support appropriate response to the agility drivers of the enterprise
Business people and developers work together daily throughout the project The most efficient and effective method of conveying information with and within a development team is face-to-face conversation The best architectures, requirements and designs emerge from self-organizing teams	FT1: Collaborative decision making FT2: Participative design	**AP3:** Bring representatives across all EA domains (or Enterprise Domains) at each level to participate in a collaborative decision making, and governance processes throughout the EAM endeavor.
Build projects around motivated individuals, give them the environment and support they need and trust them to get the job done Agile processes promote sustainable development. The sponsors, developers and users should be able to maintain a constant pace indefinitely	FT1: Collaborative decision making FT2: Participatory design FT5: Dynamic capabilities	**AP4:** Motivate individuals and teams, provide them with the necessary support (e.g. tools, resources, commitment, etc.) they need, and trust them to perform the EAM function at a sustained pace.
Continuous attention to technical excellence and good design enhances agility. Simplicity the art of maximizing the amount of work not done is essential At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly	FT2: Participatory design FT3: Organizational Fitness FT5: Dynamic capabilities	**AP5:** Start with a “good enough” design, and through acquisition and integration of knowledge, continuously improve the design, architectural artifacts and the processes by which the architecture artifacts are produced

5.2 Adapted Lean Principles and Wastes

The principles of lean thinking presented in Hines et al., (2004), and the seven lean wastes presented in Bente et al. (2012, p. 176) are adapted for Enterprise Architecture

Management through the lenses of the "probable theoretical glue" presented early on in this thesis.

5.2.1 Adapted Lean Principles for Agile EAM

Here, the Principles of Lean Thinking from manufacturing (Hines et al., 2004) are adapted for Enterprise Architecture Management (EAM). The names of all the principles but "let the customer pull the process" remain the same. The products of EAM (for instance models of enterprise capabilities) are consumed by enterprise stakeholders who might be within the enterprise (for example top management) or external to the enterprise (for example those that exist within virtual enterprises or value networks). Besides, end consumer or market needs are implicated within the strategies of the enterprise, and these strategies drive the enterprise transformation efforts. Therefore the "Let the customer pull the process" changes to "Let strategy and enterprise stakeholder needs pull EAM efforts". The Lean Principles for EAM are briefly described in the Table 13 below.

Table 13: Adapted Lean Principles for Agile EAM

Principles of Lean Thinking (Hines et al., 2004)	**Foundation Theories**	**Adapted Lean Principles for Agile EAM**
Specify Value: define value precisely from the perspective of the end customer in terms of a specific product with specific capabilities offered at a specific price and time	FT1: Collaborative decision making FT2: Participatory design FT3: Fitness Theory FT4: Contingency based requirement management	**Specify Value:** Specify value from the collective perspective of enterprise stakeholders in terms of specific architectural products and artifacts to be produced and strategic intent of such products and artifacts
Identify the value stream: identify the entire value stream for each product or product family and eliminate waste	FT3: Organizational Fitness FT5: Dynamic capabilities	**Identify the value stream:** Identify and map out all processes, resources, and capabilities that are needed across all architectural domains to produce required architectural products and artifacts
Make the value flow: make the remaining value creating steps flow.	FT1: Collaborative decision making FT2: Participatory design FT3: Organizational Fitness FT5: Dynamic capabilities	**Eliminate waste and create flow:** Review architecting and architecture management processes, and architectural products and artifact for waste and eliminate them
Let the customer pull the process: design and provide what the customer wants only when the customer wants it.	FT1: Collaborative decision making FT2: Participatory design FT3: Organizational Fitness FT5: Dynamic capabilities	**Let strategy and enterprise stakeholder needs pull EAM efforts:** EAM should respond to strategy (enterprise & EA domain strategies) and satisfy stakeholder needs from enterprise (or holistic) perspective.
Pursue perfection: strive	FT3: Organizational Fitness	**Pursue perfection:** Start with a work-

for perfection by continually removing successive layers of waste as they are uncovered.	FT5: Dynamic capabilities	ing architecture and/or architecture process and work towards perfection by eliminating waste and tailoring as-is architecture towards (shifting) to-be architecture.

5.2.2 **Adapted Lean Wastes for Agile EAM**

The seven lean wastes presented in Bente et al. (2012, p. 176) are adapted for agile EAM as shown in Table 14. The differences in the lean wastes presented in Bente et al. (2012) and in this thesis are because of two reasons. Firstly, Bente et al. (2012) are more concerned with architecting and secondly, their view of EA is more IT centric or what Lapalme (2012) calls an EITA school of thought. In this thesis, the concentration is on enterprise architecture management from a holistic (or an enterprise-wide) perspective. One of the wastes; "over-architecting" remains the same in name and content. Two of the wastes – Redundant Processes, and Delays – remain the same in names but the contents are changed to reflect EAM rather than architecting. "Partially done work" is changed to "In-Process Inventory of Architecture Products" to include not only the architecture works that have been abandoned mid-way but also the piles of architecture products that do not meet immediate business needs or needs in the near future (what Armour et al. (1999a, p. 40) call "a great architecture for the wrong business"); and architecture artifacts that are no longer needed but have not been retired from the enterprise architecture or taken out of the active architecture repository. "Task switching" which essentially means different teams working on the same item in turns has been changed to "Extra Disruption". By Extra disruptions, the author of this thesis means momentary stoppages in work caused by handing off work from one team to another for further processing (handoffs), switching between tasks (task switching), poor workflow, EA and EAM immaturity, unavailability of information, and low enterprise awareness of EAM processes.

On manufacturing floors from where the concept of "Defect" originated, most of the products are physical and are less likely to be remodeled or revamped to meet new and changing requirements, at least not easily. For instance, a non-defective engine part (e.g. a piston) in a low horsepower model of a car does not become defective when the horsepower requirement increases in a new model; it is just considered unsuitable and a new part (i.e. a new piston) is manufactured to suit the new requirements. In effect, mechanical components are frozen designs and a non-defective component is not necessarily made defective by changing contingencies. However, in EA the products are mostly intangible and the design is a continuous process that allows design artifacts to be revamped to meet changing contingencies. Misfit is used here instead of "defect" because misfit could occur at the time of design/development (i.e. "defect") and/or during use when contingencies cause an artifact that was once fit to become misfit (i.e.

out of fit). Besides, misfit is more in line with organizational and management science theories (see 3.3.3), and applicable in the EA context than defect is.

"Handoffs" is adapted from "transportation" (Poppendieck, 2011; Bente et al., 2012). Transportation means existence of intermediate steps that materials go through before getting to the assembly line where they are needed. For instance, materials from a vendor are delivered to a receiving point, then shipped to a warehouse, and then transported to the assembly line (Kilpatrick, 2003) as shown in Figure 7 below. "Handoffs" has to do with waste that is created when different teams work on the same product in turns instead of working on it together (Poppendieck & Poppendieck, 2003). In this thesis, Handoffs is considered as part of waste created by extra disruptions. The term Non-Targeted Intervention is used to reflect the concept of "transportation" as explained in manufacturing. Non-targeted Interventions include EA and EAM efforts that are not targeted at specific enterprise problems, and EA efforts that are aimed at achieving long haul transition from as-is architecture to to-be architecture. To reduce this waste, specific architecture efforts must be targeted at specific architecture strategies or needs within the context of the whole enterprise architecture (also see Pulkkinen & Hirvonen, 2005). In this way, targeted short-term interventions are employment as means of reaching long-term architecture goals which are in most cases shifting in response to environmental dynamics (e.g. see Armour & Kaisler, 2001). Figure 8 below represents a manufacturing system with reduced transportation waste by means of "point-of-use-storage"(Kilpatrick, 2003), as well as an EAM effort that reduces Non-targeted Intervention waste by means of targeted EAM interventions.

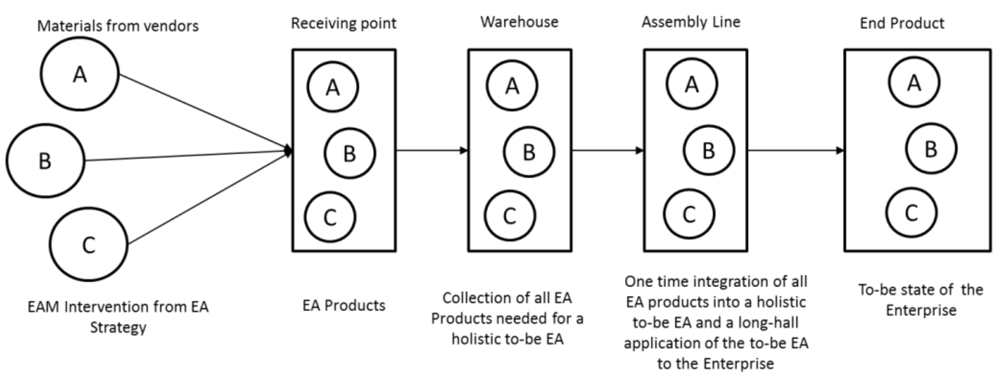


Figure 7: Transportation Waste in Manufacturing and in EAM

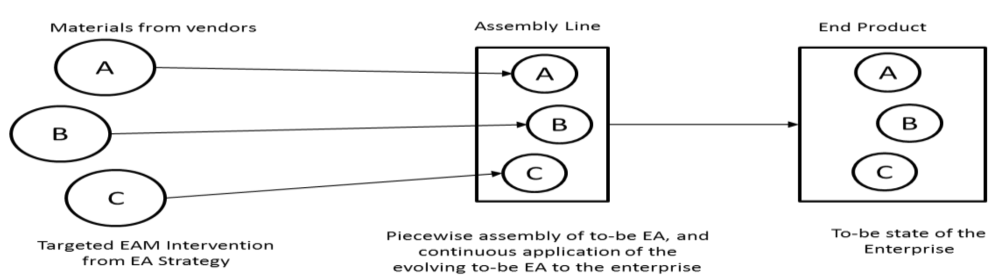


Figure 8: Reduced Transportation Waste in Manufacturing and in EAM

Table 14: The Seven Lean Wastes Adapted for Agile EAM

Seven Lean Wastes in Manufacturing (Hines et al., 2004)	**Waste in Software Development** Bente et al. (2012, p. 176) based on Poppendieck and Poppendieck (2007 & 2003)	**Seven Wastes in EA** Bente et al. (2012, p. 176)	**Foundation Theories**	**Seven Lean Waste adapted for EAM**
Over production	Extra features	Over-architecting	FT1: Collaborative decision making FT2: Participatory design	**Over-architecting:** Producing extra and/or too detailed EA products and artifacts
In-process inventory	Partially done work	Partially done work	FT1: Collaborative decision making FT2: Participatory design FT4: Contingency based EA requirement management	**In-process Inventory of Architectural Products:** Pile of partially done work; architectural products that do not meet immediate/near future strategic needs or which are targeted at too far distant future needs; and architecture products which are no more of use.
Extra Processing Steps	Extra processes, relearning	Redundant processes	FT3: Organizational Fitness FT5: dynamic capabilities	**Redundant Processes:** Non-value adding EA Management (EAM) steps and processes
Motion	Task switching	Task switching	FT1: Collaborative decision making FT2: Participatory design FT6: Dynamic capabilities	**Extra Disruptions:** Disruptions in work caused by *task switching, handoffs,* poor workflow, EA and EAM immaturity, unavailability of information, and low enterprise awareness of EAM processes.
Defects	Defects	Defects	FT2: Participatory design FT3: Organizational Fitness	**Misfit:** EAM processes and (components of) architectural products and artifacts that do not meet or are out of fit with the strategic needs of the enterprise.
Waiting	Delays	Delays	FT1: Collaborative decision making FT2: Participatory design FT3: Organizational Fitness	**Delays:** Deviation from strategic timing (delays) caused by non-fit in EAM process, architectural products and artifacts, and time used to find necessary information or trace architectural product and artifact (or time

Seven Lean Wastes in Manufacturing (Hines et al., 2004)	**Waste in Software Development** Bente et al. (2012, p. 176) based on Poppendieck and Poppendieck (2007 & 2003)	**Seven Wastes in EA** Bente et al. (2012, p. 176)	**Foundation Theories**	**Seven Lean Waste adapted for EAM**
			FT6: Dynamic capabilities	lost during *extra disruptions*)
Transportation	Handoffs	Handoffs	FT1: Collaborative decision making FT2: Participatory design FT4: Contingency based requirement management FT7: EA Literature	**Non-targeted Interventions:** EA and EAM efforts that are not targeted at specific enterprise problems, and efforts to achieve long haul transition from *as-is* architecture to *to-be* architecture

5.3 Proposed Essential Elements of an Agile EAM Method

The Essential Elements of an Agile EAM Method map onto the IS/ES attributes construct in the composite conceptual model presented in Figure 5 (Fig1c). They are the basic attributes that should guide the development and appropriation of an agile EAM method to support both the agility of the EAM function, and the agility of the enterprise. As explicated in the conceptual research model for this thesis **(see** Figure 5 in section 4.3), the EEs are suggested from the adapted agile and lean principles and values, and are supposed to provide support for the Critical Success Factors (CSFs) of EAM. Furthermore, the EEs of an Agile EAM method provide some attributes such as the mechanisms (e.g. decision making and design mechanisms), the characteristics (e.g. acting as dynamic capability), and the versatility (e.g. support for different representation tools, and project management methodologies) that the ensuing EAM method should possess. These mechanisms and processes will mediate the Critical Success Factors (CSFs) of EAM method and Enterprise benefits (especially Enterprise Agility). Table 15 below lists the EEs of an Agile EAM method and the adapted agile and lean principles of an agile EAM method based on which they were suggested.

Table 15: Proposed Essential Elements for an Agile EAM

Agile and Lean Principles and Values Adapted for EAM	Essential Elements of Agile EAM
AV4/AP2/LP1 /LP4	**EE1:** An Agile EAM Method should provide a mechanism to develop and continuously refine EA vision, goals, purpose, principles, and scope to accommodate changes in enterprise competitive bases, and turbulence in the external and internal environment
AV2/AV4/AP1/AP5 /LP2/LP3/LP4/LP5	**EE2:** An Agile EAM Method should maintain appropriate balance between long-term and short term orientation and satisfy EA stakeholders through early, continuous and incremental delivery of effective and efficient EA products towards achieving a (shifting) long term goal/strategies
AV1/AV3/AP3/AP4	**EE3:** An Agile EAM Method should provide an appropriate decision making and governance mechanisms (e.g. collaborative decision making and consensus building) that result in participation and commitment of all stakeholders, and decisions that benefits the enterprise as a whole rather than individual domains
AV1/AP2/AP4/LP3 /LP4/LP5	**EE4:** An Agile EAM Method should promote the development of skilled teams and adaption of methods and tools, and provide mechanisms for defining EA requirements and criteria for project goal fulfillment
AV3/AP5	**EE5:** An Agile EAM Method should support appropriate representation of architectural artifacts and communicate architectural documentations across the enterprise in a common language
AP5/LP3/LP5	**EE6:** An Agile EAM Method should provide mechanism to devel-

	op and maintain an integrated architecture repository for knowledge management and enterprise awareness initiatives
AV1/AV4/LP4/LP5	**EE7:** An Agile EAM Method should act as a dynamic capability (e.g. as sensing, learning, integration, and coordination capabilities), and support enhancement of agility capabilities and development of agility providers of the enterprise
ALL APs, AVs, & LPs	**EE8:** An Agile EAM Method should promote the anchoring/integration of its mechanisms and tools into the culture of the organization

EE1: An enterprise does have a vision that provides the description of where the enterprise wants to be in the long-term in relation to its structure and form, industry, partners, competitors, and clientele. The enterprise is driven towards this vision by its strategies which are formulated to maintain fit between the enterprise and its environment, and to channel its structure, operations and culture towards achieving the organizational fit that is required. EAM is believed to be the managerial endeavor that guides the design and development of an enterprise's architecture to achieve its vision and strategy. In this regard, EAM vision and strategies should be aligned with those of the enterprise, and development efforts that are usually conveyed through projects and programmes should be guided by goals, purpose and scope that are consistent with the EAM visions and strategies. Also, due to the coordinating role of EAM, its principles should be formulated from an enterprise-centric perspective to enable EAM to guide the co-evolution of the various components, e.g. business, information, information systems and technologies; that make up the enterprise. However, as enterprises engage in dynamic environments, there is the need to continuously change the enterprise strategies and organizational design in response to changing contingencies. Consequently, the EAM function needs to produce "dynamic architecture", and guide the pursuance of organizational (re)design and the organizational dynamics that are needed to implement shifts in enterprise strategies. The EAM function therefore needs a method, the Agile EAM method that provides it with a mechanism that enables it to define and refine EAM visions, strategies, goals, purposes, principles and scope to accommodate changes in the enterprise level vision and strategies.

EE2: The EAM function should concentrate on steering the enterprise towards its strategic goals and should therefore maintain a holistic enterprise-centric view. However, in view of the complexity of the enterprise, resulting from its numerous components, the interactions amongst them, and the need to change rapidly in response to environmental dynamics, the EAM function could only attain such long-term vision and strategies through short-term orientation. In other words, the EAM function needs to keep a balance between long-term orientation and short term orientation. Hence, as the EAM function strives to steer the organization towards achieving its vision and strategies, it must do so iteratively and with each of the iterations adding value in the form of increments, that satisfy the concerns of stakeholders, to the enterprise's architecture. Indeed, when stakeholders frequently attain early goals – or "early wins" – in meaningful work, they tend to be creatively productive in the long run (Amabile & Kramer, 2011). This is especially so when they experience small but consistent progress towards solving a major problem in the long-term. At some points in the (re)design of the enterprise to meet pressing strategic demands (e.g. the need to respond to a temporary agility driver), there might be the need to produce some architecture artifacts outside the mainstream architectural plan, or processes. In this thesis, the ensuing artifact is referred to as an ad-hoc architectural product. The EAM function needs to be equipped with mechanisms that enable it to manage the development, evolution and retirement of these ad-hoc architectural products. The Agile EAM method should therefore provide the EAM function with the ability to maintain appropriate balance between long-term and short term orientation and satisfy EA stakeholders through early, continuous and incremental delivery of effective and efficient EA products towards achieving a (shifting) long term goal/strategies.

EE3: EAM endeavors involve diverse stakeholders with likely conflicting views, power, and differences in knowledge and understanding. In this regard, the EAM function as a strategic endeavor should foster collaboration amongst these stakeholders. Collaboration among stakeholders from the various EA domains is necessary to ensure that the various EA domain efforts are geared towards a common goal in the collective interest of the enterprise rather than local optimization at the detriment of enterprise-wide goals. Collaboration will also establish shared understanding amongst the stakeholders and enable them to reach a consensus on enterprise-wide architectural endeavor and its implication for the various domains. In order words, at each level of the enterprise, the EAM function should establish a collective architectural goal from which the various EA domain endeavors are crafted. Also, when stakeholders participate effectively in the architectural process they do not only embed their tacit knowledge in the ensuing architectural artifact, but they also ascribe an extent of relevance to the architectural process and the ensuing architectural artifacts. This stakeholder participation and involvement is likely to result in a high intention to use and actual use of the architectural artifacts. The enterprise architect's role in the collaboration decision making and the participatory design processes is that of a facilitator to guide the design of a "dynamic architecture" that meets the needs of the enterprise in the short term, and which can be reconfigured and scaled to meet the future needs of the enterprise. An agile EAM method should therefore provide the EAM function with appropriate decision making and governance mechanisms that result in participation and commitment of all stakeholders, and in decisions that benefits the enterprise as a whole rather than individual domains.

EE4: The requirements for an EAM endeavor could be changing rapidly (volatile), difficult to communicate (identify), difficult to understand (complex), or could represent different things to different stakeholders (discordance). The nature of EA requirements does not only make the definition of what represents "done" very difficult but also makes the definition of the criteria for goal fulfillment; for example timeliness, cost, and the percentage of changes to be accommodated; very problematic. However, for EAM endeavors to demonstrate value and the extent of success, these criteria for goal fulfillment need to be defined for each endeavor, and possibly refined to accommodate changing contingencies. As mentioned in EE3, EAM involves the coordination of the various EA domain efforts in order to rectify the potentially conflicting EA requirements from diverse stakeholders and convert them into EA products.

Furthermore, the EAM endeavor, like any other managerial endeavor, relies on skilled teams to translate EA requirements to architectural artifacts and to manage the processes involved. Skills requirements may vary from one EAM endeavor to another and in relation to the criteria for goal fulfillment for the particular endeavor. For instance, for the same EAM effort, the skills requirement when the focus is on timeliness might vary from the skills requirement when the focus is on cost efficiency or the ability to rapidly respond to changing requirements. EAM function should therefore facilitate the identification of skill requirements based on EA requirements and success criteria, and formulate EAM teams accordingly. For example, by recommending training for internal EAM team members or sourcing for skills from outside. Also, the situational dependence of EAM function makes a "one-size-fits-all" approach to EAM impossible. Methods and tools employed during an EAM endeavor are selected, adapted, or even developed to suit the situations of the EAM endeavor. Such methods and tools could include project management methods, change management methods, planning tools, and requirement management and modeling tools. An EAM method should therefore support a wide variety of tools and allow the EAM function to select and use tools and methods that are suitable for the particular EAM endeavor. To lend support to the agility of the EAM function, the agile EAM Method should therefore promote the development of skilled teams and adaption of methods and tools, and provide mechanisms for defining EA requirements and criteria for project goal fulfillment.

EE5: Communication of architectural artifacts and information to stakeholders is an important role of the EAM function. However, the multiplicity of stakeholders coupled with the quest to establish common understanding amongst these stakeholders make the effective communication of architectural artifacts difficult. Views and viewpoints are usually used as a remedy in EA. They enable stakeholders to see the architecture from a particular set of concerns. Beyond the views and viewpoints, the EAM function needs to establish a common view of the whole architecture to which all the stakeholders at a particular level could relate. This view should be void of excessive details but comprehensive enough to allow the different stakeholders grasp a holistic view of the architecture, and the implications for their various EA domains or enterprises. The common view then forms the basis for stakeholder collaboration, and stimulates detail design and development efforts within the various EA domains. To achieve this, the EAM function needs to create a set of principles that governs the representation of architectural artifacts, and to create a common language or a common frame of reference that enables the effective communication of these artifacts. To this end, an agile EAM method should support appropriate representation of architectural artifacts and the communication of architectural documentations across the enterprise in a common language.

EE6: The enterprise's abilities to create and absorb, codify and articulate, integrate and reconfigure knowledge have been found to shape its ability to respond to rapidly changing environments. This knowledge is embedded in the components of the enterprise for instance the enterprise's procedures, and capabilities. EA deals with the design, representation and evolution of these enterprise components and can therefore contribute to enterprise knowledge. An integrated EA repository offers the EAM function the opportunity to store and re-use EA descriptions, best practices and lessons for instance during enterprise awareness creation or enterprise (re)design efforts. In this way, inanimate things including architectural models are allowed to participate in the (re)design process. Enterprise awareness (or cognition) of EA has been identified in the EA literature as one of the antecedents of sustained stakeholder engagement in EA, actual use of EA artifacts, and anchorage of EAM within the enterprise. An agile EAM method should therefore provide the EAM function with mechanisms to develop and maintain an integrated architecture repository for knowledge management and enterprise awareness initiatives.

EE7: The three EA concepts; Enterprise Architecture, Enterprise Architecting, and Enterprise Architecture Management that were presented earlier in this thesis, when taken together, could represent a dynamic capability in line with Pavlou & El Sawy (2011). The content nature of EA represents the "existing operational capabilities" (as-is architecture) and the "reconfigured operational capabilities" (to-be architecture); the process nature of EA (that's Enterprise Architecting) represents the "integrating capability"; and EAM provides the capabilities that sense changes in the enterprise environment, foster learning, and coordinate the various enterprise architecting (integrating capability) efforts within the enterprise. That is to say, the EAM function acts as a sensing capability, a learning capability, and a coordinating capability. However, because dynamic capabilities do not by themselves result in an agile enterprise but the continuous application of dynamic capabilities does, the EAM function needs to deploy its capabilities to continuously transform the enterprise in response to changing contingencies. An agile EAM method should provide the EAM function the ability to do so, by acting as a dynamic capability and supporting the enhancement of agility capabilities and development of agility providers of the enterprise.

EE8: Finally, for the EAM and the EAM function to yield the results that the enterprise expects from them, the EAM function needs to be positioned as a strategic endeavor that involves all aspects of the enterprise. EAM endeavor should not be seen as an add-on activity but it should be regarded as an integral component of how the enterprise does business. It should be embedded into the daily routines of the enterprise and given a strategic mandate. To achieve this end, the EAM function needs to build its methods, mechanisms, and tools based on the enterprise's methods, mechanisms and tools. To support the EAM function along this line, an agile EAM method should promote the anchoring/integration of its mechanism and tools into the culture of the organization.

5.4 Evaluation of the Essential Elements of an Agile EAM Method

This sub-section marks the first iteration point in the DSR process. The suggested EEs are evaluated for completeness and for relevance. The evaluation for completeness is done by relating the EEs to extant literature, the CSFs of EAM, and the agility providers and agility capabilities of an enterprise. Also, through an interview process, the suggested EEs are evaluated for relevance, especially to practice. The interview results are presented and analyzed to inform the next step in the DSR process.

5.4.1 Evaluation for Completeness

The evaluation of the EEs of an agile EAM for completeness is done from three perspectives. The evaluation is done firstly, against the extant EA literature on the essential components of EAM; secondly, against the list of Critical Success Factors of EAM; and finally, against the agility providers and agility capabilities of the enterprise. This evaluation does not however conclude that the EEs of an agile EAM method that have been proposed in this thesis are all-inclusive but rather provides indication that the thesis has acknowledged prior knowledge. The evaluation is presented in Table 16**.** Essential building blocks or components of an EAM have been researched in the EA literature as presented in sub-section 2.2.11. Column 2 of Table 16 presents a mapping of these elements in extant literature against the EEs of an agile EAM proposed in this thesis. Further, the EEs of an Agile EAM Method construct within the conceptual model for this thesis (see Figure 5 in section 4.3), is to support the provision of the Critical Success Factors of EAM. Column 3 of Table 16 below represents a proposed relation between the EEs of an Agile EAM method and the CSF of EAM.

Finally, the Agile EAM method is to promote both the agility of EAM function and the agility of the enterprise at large. In this regard, the proposed EEs of an Agile EAM method should strengthen the Agility Providers to achieve the Agility Capabilities with which the enterprise senses and appropriately responds to changes in its environment (Agility Drivers). Agility Drivers, Agility Capabilities and Agility Providers were discussed in section 2.1.3. Colum 4 of Table 16 presents a proposed relationship between the EEs of an Agile EAM method and Agility Capabilities (AC) and Agility Providers (AP). There exists no clear one-to-one relationship between the EEs and the agility capabilities and the providers partly because, as noted by Zhang & Sharifi,(2000) agility capabilities are related and the enhancement of one capability might enhance other agility capabilities as well.

Table 16: The evaluation of the EEs for completeness

Essential Elements of Agile EAM	Extant EA Literature	CSFs of EAM	Agility Capabilities & Providers
EE1: An Agile EAM Method should provide a mechanism to develop and continuously refine EA vision, goals, purpose, principles, and scope to accommodate changes in enterprise competitive bases, and turbulence in the external and internal environment	Armour, Kaisler, & Liu (1999a, 1999b), Armour & Kaisler (2001) Wagter et al (2005, pp. 53–56), Buckl, Dierl, Matthes, & Schweda (2010), Buckl et al (2011), Aier, Gleichauf, & Winter (2011), Ahlemann et al(2012, pp. 38–55)	Scoping and purpose / Clearly defined vision and goals / Assessment and evaluation / Change management / Clearly stated EA principles	**AC:** Responsiveness/ Customer focus **AP:** Customer/market sensitivity/ change
EE2: An Agile EAM Method should maintain appropriate balance between long-term and short term orientation and satisfy EA stakeholders through early, continuous and incremental delivery of effective and efficient EA products towards achieving a (shifting) long term goal/strategies	Armour, Kaisler, & Liu (1999a, 1999b), Armour & Kaisler (2001), Pulkkinen & Hirvonen (2005), Wagter et al (2005, pp. 53–56), Buckl, Matthes, & Schweda (2009), Buckl, Dierl, Matthes, & Schweda (2010), Buckl et al (2011), Bente et al (2012, pp. 27–28), Ahlemann et al(2012, pp. 38–55)	Intertwine planning and execution / Development Methodology and tools / Business/Strategy driven approach / Change management / Adequate documentation(e.g. of "as-is" and "to-be" architecture)	**AC:** Responsiveness /Flexibility / Quickness /Robustness **AP:** Technology / Innovation / Information systems / Change / Customer/market sensitivity/ Quality
EE3: An Agile EAM Method should provide an appropriate decision making and governance mechanisms (e.g. collaborative decision making and consensus building) that result in participation and commitment of all stakeholders, and decisions that benefits the enterprise as a whole rather than individual domains	Pulkkinen & Hirvonen (2005), Wagter et al (2005, pp. 53–56), Buckl, Matthes, & Schweda (2009), Buckl, Dierl, Matthes, & Schweda (2010), Aier, Gleichauf, & Winter (2011), Bente et al (2012, pp. 27–28), Ahlemann et al(2012, pp. 38–55),	Commitment and Sponsorship (especially from top management) / Communication and Common language / Business/Strategy driven approach / IT investment and Acquisition strategy / Stakeholder participation and involvement	**AC:** Proactiveness/ Responsiveness / Knowledge-driven enterprise **AP:** Collaborative relationship / Team building
EE4: An Agile EAM Method should promote the development of skilled	Buckl et al (2011), Ahlemann et al(2012, pp. 38–55),	Development Methodology and tools / Effective project & pro-	**AC:** Competency

Essential Elements of Agile EAM	Extant EA Literature	CSFs of EAM	Agility Capabilities & Providers
teams and adaption of methods and tools, and provide mechanisms for defining EA requirements and criteria for project goal fulfillment		gramme mgt. / Skilled team, training & education	**AP:** Competency /Team building / Education /People & Welfare
EE5: An Agile EAM Method should support appropriate representation of architectural artifacts and communicate architectural documentations across the enterprise in a common language	Wagter et al (2005, pp. 53–56), Rouhani et al (2008), Buckl, Matthes, & Schweda (2009), Buckl, Dierl, Matthes, & Schweda (2010), Ahlemann et al(2012, pp. 38–55)	Communication and Common language / Enterprise awareness (cognition of EA) / Integrated repository / Adequate documentation(e.g. of "as-is" and "to-be" architecture)	**AC:** Competency/ Knowledge-driven enterprise **AP:** Competency / Education
EE6: An Agile EAM Method should provide mechanism to develop and maintain an integrated architecture repository for knowledge management and enterprise awareness initiatives	Aier, Gleichauf, & Winter (2011), Bente et al (2012, pp. 27–28), Ahlemann et al(2012, pp. 38–55),	Skilled team, training & education / Communication and Common language /Organizational culture/ Enterprise awareness (cognition of EA) / Integrated repository / Adequate documentation(e.g. of "as-is" and "to-be" architecture)	**AC:** Knowledge-driven enterprise /Core competency mgt. **AP:** Information systems / Education
EE7: An Agile EAM Method should act as a dynamic capability (e.g. as sensing, learning, integration, and coordination capabilities), and support enhancement of agility capabilities and development of agility providers of the enterprise	Rouhani et al (2008), Buckl, Matthes, & Schweda (2009), Aier, Gleichauf, & Winter (2011), Bente et al (2012, pp. 27–28), Ahlemann et al(2012, pp. 38–55),	Quality EA product (e.g. models and artifacts) /Business/Strategy driven approach /Change management /Enterprise awareness (cognition of EA) / Integrated repository	**AC:** Capability for re-configuration /Virtual enterprise/ Core competency mgt. / competency **AP:** Integration / Relationship with suppliers/customers/ competitors /Process integration
EE8: An Agile EAM Method should promote the anchoring/integration of its mechanisms and tools into the culture of the organization	Wagter et al (2005, pp. 53–56), Ahlemann et al(2012, pp. 38–55),	Organizational culture	AC: Competency / Core competency management AP: People /Organization

5.4.2 Evaluation for Relevance

Here, the processes employed to evaluate the relevance of the suggested EEs are presented. The processes included: preparation (operationalization of the EEs and questionnaire development); conducting the interview (selection of the interviewees and the interview process); presentation of the interview results; and the analysis of the interview results.

Operationalization of the EEs and Questionnaire Development

To facilitate the interview process, the EEs of an Agile EAM method were operationalized into sub-elements or sub-components whose relevance were examined on a *likert scale* of 1-5; with 1 meaning "Not Essential at All" and 5 meaning "Very Essential". There were 31 sub-elements. In addition, to the operationalized EEs, the interview also collected some background information that helped to situate the interview results within context. The full questionnaire that was used during the interview comprised of the operationalized EEs and the background questions.

The Interview Process

The interviews were conducted from 15th July, 2015 to 9th October, 2015 and involved 8 practitioners and academics. The practitioners were consultants or corporate managers who were directly involved in enterprise architecture work, and the academicians were researchers who have been and were still involved in enterprise architecture research as at the time of the interview. Mails were sent to a total of twelve (13) potential interviewees selected at random. Ten of them responded, and eight (8) of them subsequently, were involved in the interview process. The other two were not available: one was on a sabbatical, and the other was retired. Out of the eight interviewees, the author of this thesis had no prior relationship with seven (7) of them. He had a student-guess lecturer relationship with one of the interviewees (Interviewee G in the Table 17 below).

Some of the interviews were conducted during face-to-face meetings either in the interviewee's office or in a common room in the Faculty of Information Technology (University of Jyvaskyla), whilst others were done online via Skype or Adode Connect (AC). The research proceedings in all cases were recorded using an audio recording device and later transcribed.

The interviewees were allowed freedom to express their experiences, ask questions for clarification or verify a source of information, and momentarily digress in attempt to provide support to their opinion on the relevance of the EEs of EAM method. Therefore, though each of the interviews was scheduled to last for an hour, the actual durations ranged from two hours forty-five minutes to fifty three minutes. Other factors that affected the duration of the interview include how well the interviewee was able to familiarize himself or herself with the areas of concern, the schedule of the interviewee, and how elaborate and reflective the interviewee was. Table 17 below provides the summery of the interview processes.

Table 17: Summary of Interview Process

Interviewee	Job Cat.	Position	Exp. (Yrs)	Date /15	Mode	Media	Duration
Int.A	Consultant	Bus. Architect Info. Architect	Above 10	15/07	Face-to-face	n/a	2:42:03
Int.B	Consultant	Ent. Architect Bus. Architect	Above 10	07/08	Online	Skype	1:49:54
Int.C	Corporate Manager	C-Level (CIO)	9-10	10/08	Face-to-face	n/a	1:04:08
Int.D	Consultant	Ent. Architect Bus. Architect	Above 10	31/08	Online	Skype	53:32:00
Int.E	Researcher	Ent. Architect	6-8	14/09	Face-to-face	n/a	1:23:01
Int.F	Researcher Consultant	Bus. Architect	6-8	17/09	Face-to-face	n/a	1:07:16
Int.G	Corporate Manager	C-Level (VP)	Above 10	9/10	Online	Skype	1:09:43
Int.H	Corporate Manager	Ent. Architect	6-8	30/10	Online	Adobe Connect	1:11:17

Interview Results

All the eight interviewees responded to the foundation questions and the EEs. Each of the interviewees responded to all the 31 sub-elements (N=31) as shown in Table 9 . Appendix 1 shows individual rating, mean, and standard deviation for each of the sub-elements, and the Essential Elements. Table 18 provides a summary of the ratings, mean, and standard deviation for each of the EEs. EE1 has the highest mean (m = 4.31, sd = 0.26) and EE5 has the lowest mean (m = 3.53, sd = 0.34). With regards to the interviewees, interviewee E and interviewee G gave the highest ratings (mean = 4.06, sd = 0.997) and (mean = 4.06, sd = 0.964) respectively; and interviewee D gave the lowest rating (mean = 3.71, sd = 0.902) as shown in Figure 9. The total mean, calculated as the mean of the mean ratings of the respondents is mean = 3.91, sd = 0.136.

Apart from generating quantitative data, the interview process also generated a lot of descriptive (qualitative) data that is useful in this thesis, especially in the analysis of the interview results, and during the design of the Agile EAM method.

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
Interviewee A	31	2,00	5,00	3,8065	,79244
Interviewee B	31	2,00	5,00	3,8387	,93441
Interviewee C	31	2,00	5,00	3,8387	,86011
Interviewee D	31	2,00	5,00	3,7097	,90161
Interviewee E	31	2,00	5,00	4,0645	,99785
Interviewee F	31	2,00	5,00	4,0323	,94812
Interviewee G	31	1,00	5,00	4,0645	,96386
Interviewee H	31	2,00	5,00	3,8710	,80589
Valid N (listwise)	31				

Figure 9: Summary of average rating per interviewee

Table 18: Mean ratings for each of the EEs per interviewee

	Essential Elements	**Int.A**	**Int.B**	**Int.C**	**Int.D**	**Int.E**	**Int.F**	**Int.G**	**Int.H**	**Mean**	**SD**
EE1	Mechanism to develop and refine EA vision, goals, purpose, principles & Scope	4.00	4.00	4.50	4.50	4.50	4.50	4.50	4.00	4.31	0.26
EE2	Maintain balance between short & long term orientation	3.75	4.25	3.50	4.00	3.75	4.00	3.75	4.50	3.94	0.32
EE3	Provide decision making & governance mechanisms to promote stakeholder commitment	4.33	4.00	4.67	4.00	4.33	4.67	3.00	3.67	4.08	0.56
EE4	Promote development of skilled teams, adaption of methods & tools, and definition of requirements	3.60	3.60	3.40	3.80	3.80	3.60	3.60	3.40	3.60	0.15
EE5	Support appropriate representation of artifacts & communicate documentation in a common language	3.75	3.50	3.25	3.25	3.50	3.25	4.25	3.50	3.53	0.34
EE6	Mechanism to develop and maintain an integrated architecture repository	3.75	4.00	3.75	3.75	4.00	4.25	5.00	4.00	4.06	0.42
EE7	EAM method as a dynamic capability and support agility capabilities, and agility providers	3.75	3.50	3.75	3.25	5.00	4.50	4.00	4.50	4.03	0.59
EE8	Agile EAMM should promote the anchoring/integration of its mechanisms into the culture of the organization	3.67	4.00	4.33	3.00	3.67	3.67	4.33	3.33	3.75	0.46
	Total Averages	3.83	3.86	3.89	3.69	4.07	4.05	4.05	3.86	3.91	0.13

Analysis of Interview Results

In general, the mean values of 4 of the EEs are between Somewhat Essential and Essential (3.53 – 3.94), and the mean values of the other 4 of the EEs are between Essential and Very Essential (4.03 – 4.31). The essential elements are closer to 4.0; that is Essential; than they are to either Somewhat Essential or Very Essential. Also, the total mean rating for the interviewees (mean = 3.91, sd = 0.136) indicates that, in general each of the interviewees perceives the EEs to be approximately 4, that is Essential.

Aside the general inferences made above, an analysis was performed to examine the extent to which the individual interviewees agree by calculating the Intraclass Correlation Coefficient (ICC) using SPSS. Since each of the eight interviewees responded to each of the 31 sub-elements, and the research intends to generalize the result beyond the interviewees so as to form the basis for the method construction, the ICC was computed using Two-way Random, Absolute Agreement and Coefficient Interval of 95% in SPSS. The ICC Single Measures and Average Measures were .120 and .522 respectively. The two measures were statistically significant ($p < .001$). According to LeBreton & Senter's standards for interpreting interrater agreements, there exists a moderately strong agreement amongst the interviewees (2007, p. 22).

The moderately strong agreement that existed amongst the interviewees might have been informed by several factors including; the small number of raters (8 interviewees) (LeBreton & Senter, 2007, p. 14), and the situational specific nature of EAM (e.g. Buckl, Schweda, & Matthes, 2010; TOGAF, 2009, p. 56). To understand the source of discrepancy amongst the interviewees beyond the statistical values, qualitative data on the sub-elements with high standard deviations (e.g. EE2d, EE3c, EE4a, EE4b, EE6c, EE7c, EE8a and EE8b) were analyzed. For brevity, the interviewees will be referred to here with only their corresponding letters (e.g. A, instead of Interviewee A). In other sections, the full label will be used.

Concerning EE2d (Duration of short term goals should be at most 3 months), whilst B stated that "it's not a short-term goal if it's half a year or a year or more", E and F believed that 6 months is the minimum in the public administration. For instance F stated that "…change takes time, the corporation is big and the pace of the change in municipality is very very difficult…because it is regulated by the law…" G believed that "…it's a case by case thing" however he did add that "…if this is about keeping stakeholders happy…that they can see something…some concrete progress,... because agile it is the practice to demonstrate actual, I think that … yeah at most 3 months". C was of the view that "3 months sounds good so… we can then speak about the quartiles…" however he concluded that "…but I don't know how important this is".

Another sub-element that generated differences in view is EE3c (promote co-evolution of domain strategies). A, B, C, E, F and H agreed that business strategy and IT strategy should at least lead and support each other. For instance B stated "…it should be always so", whilst C stated "…for example if there are some kind of very good innovation that will make some kind of business to company so ….it is essential." However, G argued that "… if any function has any business significance, then it will become part of the business strategy" and so according to him, "IT cannot have a strategy that is not derived from business strategies".

EE4a (Promote development of skilled teams) and EE4b (Support wide range of project management methods and architecting tools) also generated divergent views. Regarding EE4a, B, F, C, and E believed that it was at least essential for the EAMM to support the development of skilled teams. However, A,D,G, and H held a contrary view. For instance, H stated "… yes of course it's good that there is much enterprise architecture capability and skills … in the organization" however he added " … I think that we prioritize what agile enterprise architecture management should do and this is maybe not among them…the most important things". Regarding EE4b, E stated "… if the tools are correct and we are happy with them, we can only manage with one or two methods and tools… ". The view of H supports that of E. H believed "…the wider the range of many methods you have the more chaotic it gets …" On the contrary, G believed that EAMM should support a wide range of methods in order to allow the enterprise to freely choose methods and tools. For instance he stated "…enterprise architecture doesn't dictate that what is your project or programme management methodology…it should be able to pick your management…"

On the issue of maintaining the EA repository with commitment from all levels of the enterprise (EE6c), most of the interviewees did agree that EE6c was at least essential. However, others believed it was either somewhat essential (D and H) or not essential (E). On one hand, G did state "… the business has to be reliant on the enterprise architecture artifacts…and really the commitment has to come from the CEO of the company…" B seemed to share G's view by stating "…if there is not any commitment, it [EA repository] is technically affected… it can become an archive…" He likened an archive to an EA repository whose contents are not used or re-used. On the other hand, E said "…I don't see that all the levels of the organization should be… even aware that we have some architecture repository…" Whilst H momentarily was unsure what the situation should be by stating "…but I wonder if also the senior management…like highest level of the enterprise is so interested in the repository…", he later concluded "…I think senior management is not so interested if we have repository or not…the commitment to maintaining the repository is for the expert level personnel…"

EE7c (coordinate allocation of resources and create an environment of CSFs) received one of the lowest ratings largely because of its structure; a combination of coordination role, resource allocation, and provision of CSFs. Most interviewees concentrated on the resource allocation aspect. For instance, B was unsure "…if enterprise architecture method is the right place to do this kind of allocation…"; G stressed "…this's about allocation of resources which is not the purpose of enterprise architecture management methodology"; and H stated "…those in the enterprise architecture, they are not so … involved in the resource management of the projects". However, F believed that EAMM should coordinate the allocation of resources but she was not sure if EAMM should provide the environment of CSFs. She stated "…I think it should coordinate the allocation of resources, but I'm not sure of these critical success factors…"

The interviewees expressed divergent views on EE8a (EAM mechanisms embedded into daily routines of organizations) and EE8b (Mechanisms to promote education of all stakeholders on EA values and principles). Concerning EE8a, B made an argument that mechanism or method that is good in one situation or in one organization might not be good in another and so did not regard EE8a as essential. Further, F claimed being agile means "…making focused attacks…but embedded is like for corporate governance" which is a "…heavy thing. It is not agile". Contrary to the views of F and B, other interviewees (e.g. C, G, and H) expressed rather positive views from lived experiences. According to C, "…the best EA work is that kind of work that people … you don't understand that you are doing EA work; they do their normal work and they use EA tools in that work". H shared his experience of EA in his institution. He said;

> "We started with our EA initiative six to seven years ago and it was quite separate. But now adays, it's more and more integrated into planning, into annual planning, into project management… I think the good thing is that many people…they are actually dealing with enterprise architecture without knowing it themselves because the method itself for enterprise architecture as a concept or as a term is not that important but it's important that our enterprise architecture principles for example are used quite widely and of course they are…the projects are assessed against those principles so forth and I think that it's good, it should be…"

Similarly, G concluded "…if it is not part of the normal management work, then it's something separate that will be dropped at some point in time. It will be an overhead." However, G regarded EE8b as internal marketing and said "…I do not consider this internal marketing or internal promotion as part of enterprise architecture management methodology; it is just management". E affirmed his early position that top management should not be bothered with EA principles and values. Meanwhile, B, C and F regarded it as at least essential.

Clearly, the views of the interviewees were influenced largely by their experiences and the context (e.g. the academia, corporate management positions in public sector or private sector, consultancy positions in different sectors, and the maturity of EA within the environment in which they interact) in which they formed such experiences. Nevertheless, the disparity in views might also be affected by the compound nature of the sub-elements. Some of the sub-elements contained multiple concepts and so the interviewees had to make compromises in the cases where they perceived one concept more essential than the other. For instance, to sub-element EE7c, F said "…this is the result of two things... so I have to make compromise, like somewhat essential." The above suggests that the sub-elements be made more atomic and another iteration of evaluation made involving a wider audience, e.g. through an online survey.

However, from the statistical point of view presented above, and from the qualitative data generated from the interviews, the interviewees do moderately agree that the suggested elements are essential. Therefore, the thesis proceeds to the next iteration of construction: the construction of the agile EAM method.

5.5 Chapter Summary

In this chapter, the theoretical foundations from Chapter 2 and Chapter 3 were used to construct the Essential Elements of an Agile EAM Method. Firstly, the "probable theoretical glue" was used to adapt agile and lean principles and values for EAM. Secondly, the adapted agile and lean principles and values, and the essential elements/components identified in extant literature were consolidated into the Eight Essential Elements of Agile EAM Method. Each of the EEs was then briefly explained. The suggested EEs of an Agile EAM method were evaluated for completeness by relating the suggested EEs to extant literature, the CSFs of EAM, and agility providers (AP) and agility capabilities (AC) of an enterprise; and for relevance through an interview process that involved 8 interviewees.

It was noted that collectively the interviewees moderately agreed that each of the 8 EEs was essential. Based on this agreement, the thesis proceeds to the next iteration of "design and develop"; that is the construction of the Agile EAMM in Chapter 6.

6 CHAPTER 6 CONSTRUCTION OF THE AGILE ENTERPRISE ARCHITECTURE MANAGEMENT METHOD

This chapter presents the second iteration in the method design (DSR) process. It is dedicated to the construction, demonstration and evaluation of the Agile EAM method in fulfillment of research question three: "how could these essential elements be conveyed in the form of a method?" In this regard, mechanisms and processes are proposed as embodiments of the EEs and the knowledge acquired from the evaluation of the EEs. Also, guidelines for instantiating the Agile EAM method are provided.

Furthermore, the compatibility of the Agile EAM method with a widely used EA development standard (the TOGAF's Architecture Development Method – ADM) is demonstrated with focus on how the Agile EAM process can contribute to the agility of the ADM, and how the proposed Agile EAM method can be employed to coordinate multiple ADMs at different levels of the enterprise. Following the findings in Peffers et al. (2012), a hypothetical scenario is used to conceptually demonstrate and evaluate the efficacy of the method in supporting the agility of the EAM function and the agility of the enterprise as a whole.

6.1 EA Domains and Enterprise Levels

As discussed in Section 2.2.7 there seem to be no agreement in research and practice on the set of EA domains that an enterprise should have. This is partly because each enterprise is different and ought to emphasize different aspects based on its peculiar circumstances. In this regard, meta-domains including those presented in section 2.2.7 provide a starting point for enterprises but are not entirely prescriptive. The method being constructed in this thesis does not suggest any set of EA domains upfront; the identification of the EA domains will be done at the time of instantiating the method within an enterprise. However, for representation purposes the thesis adopts the set of EA domains proposed in Hirvonen & Pulkkinen (2004).

Concerning the discussions on the levels of EAM endeavor, the thesis suggests four levels from three different perspectives as presented in Table 19 below. The "Enterprise Levels" refers to a hypothetic structure of the enterprise. The "EA Components Levels" are the EA components that are needed at each Enterprise Level, and the "EA Management Levels" are the EAM levels at which the various EA components are produced. The thesis recognizes that during instantiation the number of levels might reduce depending on the peculiar characteristics of the enterprise. For instance, in an enterprise where there are no different business domains (or business segments), any two adjacent levels, e.g. the Enterprise and Operations levels, could be merged; and likewise their corresponding EA component levels and EAM levels.

Table 19: Levels of EAM Endeavor

Enterprise Levels	**EA Components Levels**	**EA Management Levels**
Enterprise	Enterprise Architectures	EAM Strategy
Operations	Enterprise Capabilities	EAM Portfolio
Business Process	Enterprise Solutions	EAM Programs
Business Activity	Enterprise Features	EAM Projects

6.2 EA Components as "IT-Enable Resources"

The Agile EAM method being constructed in this thesis hinges on collaborative efforts (EE3) amongst the various EA domains at all levels of the enterprise (Enterprise levels). It considers the EA component at a particular enterprise level as an "IT-enabled resource" formed out of synergistic relationships amongst all the EA domain components at that level in a way that makes the ensuing EA components have emergent capabilities that are more desirable than the individual capabilities of the EA domain components put together (Nevo & Wade, 2010). Using the EA domains proposed by Hirvonen & Pulkkinen (2004) together with the Enterprise components Levels, Figure 10 exemplifies how the EA components are built out of synergistic relationships amongst the various EA domain architecture components. On the one hand, the addition sign (+) used here does not merely refer to the aggregation of the various EA domain architecture components but reflects the totality of the ensuing EA component; that is the sum of the EA domain components and the relationships amongst them. For example, the Enterprise Architecture is the combination of the Business Architecture, Information Architecture, Information Systems Architecture, Technology Architecture and the relationships amongst them. Equation 1 reflects this thinking. On the other hand, the addition sign does not indicate close or rigid coupling that restricts the ability to change the various sub-components or to evolve them, but represents loose coupling that allows modular reconfiguration of the constituent components.

EA Domains / EA Components	Business Architecture	Information Architecture	Information Systems Architecture	Technology Architecture
Enterprise Architectures	= Business Architecture	+ Information Architecture	+ Information Systems Architecture	+ Technology Architecture
Enterprise Capabilities	= Portfolio of Business Processes (business capability)	+ Information required and produced by the business capability (Information Capability)	+ Portfolio of applications that supports the Information capability (IS Capability)	+ Technologies that support the IS Capability (Technology Capability)
Enterprise Solutions	= Business Processes	+ Information needed for & produced from the business process	+ Application(s) that support Business process	+ Technologies that support the applications in
Enterprise Features	= Business Activities	+ Information required and produced by the activity	+ Application (application functionalities) needed to support the activity	+ Technologies needed to support the application

Figure 10: EA Components as IT-Enabled Resources

Equation 1: EA Components as IT-Enabled Resources

$$\text{EA Component}(l) = \sum_{k=1}^{n} (\text{EA Domain Components } (k) + \text{Synergy })$$

Where *l* represents the enterprise level, n represents the number of EA domains, and k represents a particular domain.

6.3 **Collaborative Decision Making and Governance Mechanisms**

Building on the discussions in session 6.2 above, the thesis discusses the collaborative decision making and governance mechanisms (EE3) that underpin the Agile EAM method being constructed. Again, for illustration purposes the thesis employs the EA domains proposed in Hirvonen & Pulkkinen (2004) and combines that with the Enterprise Levels as illustrated in Figure 11**.** In addition to the EA domains is a *virtual* domain that is referred to in this thesis as the *"Big-Picture"*. The word virtual is used here because the Big-Picture is not an actual EA domain; it only reflects the collective vision, strategies, plans, goals, principles, and efforts of the actual EA domains. Each of the enterprise levels has its own Big-Picture. The figure 8 below represents an extension of the EAM Grid proposed by Hirvonen & Pulkkinen (2004) to reflect the “big-picture “and the set of enterprise levels that have been suggested in this thesis, and to guide the design and development of the EA component as an IT-enabled resource. The Extended EAM Grid provides a guide to the enterprise, and should be instantiated to reflect the peculiarities of the enterprise; for instance, the set of EA domains, and enterprise levels.

	"Big-picture"	"EA Domain Efforts"			
Enterprise Level	Enterprise level EA Vision, Strategies, Policies & Principles	Enterprise Level business vision, strategies, policies & principles	Enterprise Level vision, strategies, policies & principles for strategic information management	Enterprise Level vision, strategies, policies & principles for Information Systems (IS) management	Enterprise Level vision, strategies, policies & principles for technology investments
Operations Level	Strategies & Principles, and operations performance criteria , and guidelines for Enterprise Capabilities	Portfolio of business processes and Strategies, and criteria for performance measurement	Operations Strategy for information management (including security) in support of efficient and flexible business process portfolio and information value chain	Operations Strategy for IS (application portfolio) acquisition, and IS principles and guidelines e.g. for integration, interoperability, scalability & maintainability	Operations Strategy for technology acquisition, and technology principles and guidelines e.g. for integration, interoperability, scalability & maintainability
Business Process Level	Strategies & principles, and performance criteria and guidelines for Enterprise Solutions	Business processes; guidelines e.g. for business process creation, improvement & retirement; and criteria for performance measurement	Information assets & Information sources; principles e.g. for availability, integrity, accessibility, usability and security; and business process - information requirement mapping	Principles on application characteristics e.g. functionality, architecture, modularity, interoperability & scalability; and criteria for application development success	Technology infrastructure e.g. platforms and networks;
Business Activity Level	Strategies & Principles, performance criteria , and guidelines for Enterprise Features	Activities & routines; resource requirements; guidelines e.g. for activity creation, improvement & retirement; and performance criteria	Information requirement for business activities & routines	Applications (application requirements) for business activities & information; requirement management (including traceability, prioritization)	Technology assets to support activity, information, and application

Figure 11: An Extension of the EAM Grid in Hirvonen & Pulkkinen (2004)

6.3.1 **Collaborative Decision Making Mechanism**

EAM decisions are made through collaborative efforts amongst the various domains. At each enterprise level, the various EA domains (their representatives) come together to make decisions on a particular EA endeavor and form a common understanding about what is to be done and the implications for the various EA domains. A high level description of the common understanding is developed and called the *big-picture* for that enterprise level. During this decision making process, the participants consider the totality of the EA endeavor and the relationships amongst the anticipated EA domain components, and try to optimize the whole rather than try to achieve local optimization. The role of the enterprise architect is to facilitate the collaborative process, provide architecture guidelines and to coordinate the various EA domain efforts towards reaching the goals of the *big-picture*. Each of the EA domains uses the *big-picture* as the reference point for more elaborate architecture efforts. When the big-picture goal at a particular enterprise level has implications for the enterprise level above or below it, the implications are communicated to the enterprise architect role at the affected enterprise level. The enterprise architect takes an appropriate action – e.g. convenes a *big-picture* meeting. Also, when there is a change at the enterprise level or in an EA domain, the *big-picture* is reviewed to reflect the change and the implications for the EA domains are incorporated into the EA domain architecture efforts. Therefore, on the one hand, the *big-picture* is formed by collaborative efforts of the EA domains, and on the other, the *big-picture* ensures alignment amongst the various EA domains and governs their co-evolution over time. The *big-picture* meetings are either periodic or triggered by a change (agility driver) that requires the attention of multiple EA domains. It is recommended that the periodic *big-picture* meetings are held between once a week to once a month depending on the peculiarity of the enterprise and the enterprise level at which the meeting is held. The frequency of the *big-picture* meetings is determined during the instantiation of the method in an enterprise. The collaborative decision making mechanism described here is illustrated in Figure 12 below.

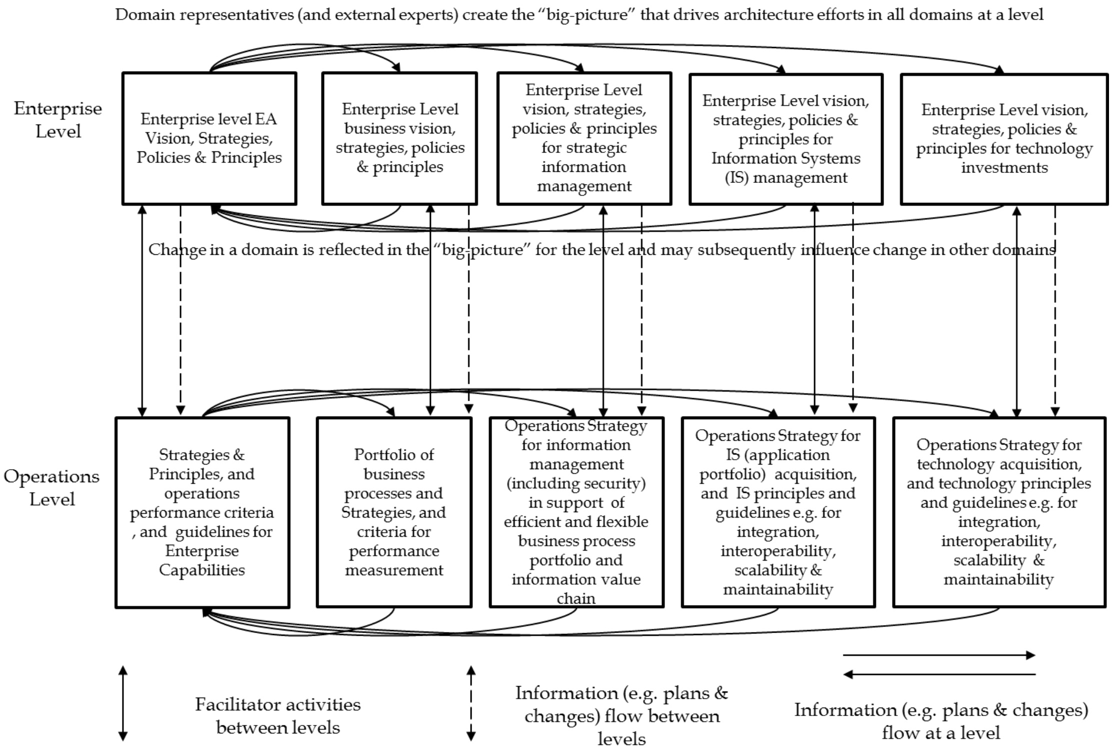


Figure 12: Exemplifying collaborative decision making mechanism

6.3.2 EA Visions, Strategies, Goals, Principles, and Plans

EA visions provide the long-term direction for the EA endeavor; the strategies are the means by which the long-term vision is to be achieved (e.g. through the use of short-term goals); the goals are the anticipated concrete gains that are to be achieved towards reaching the EA vision; the plans provide the arrangement of the EA efforts, both by time and by sequence, that enables the attainment of the visions, strategies and goals; and the principles are the philosophies, and rules that guide the development and evolution of the enterprise architecture. All these are subject to the peculiarity of the enterprise for example its structure, size, level of maturity in EA, and the contingencies that prevail in its internal and external environment. Therefore, as suggested in EE1, the EA visions, strategies, goals, principles, and plans (including purpose and scope) should be continuously refined to accommodate changes in the competitive bases of the enterprise.

Early on, it was proposed that the EA components serve as IT-enabled resource born out of collaborative efforts amongst various EA domain components (see subsections 6.2. and 6.3.1). Building on this proposal, EA visions, strategies, goals, principles and plans should also be viewed respectively as the big-picture vision, strategies, goals, principles and plans; and should be formed and reviewed by means of the collaborative decision making mechanism described in 6.3.1 above. For example, in this way, EA principles can be viewed as meta-principles that guide the development and refinement of architecture principles within the EA domains.

6.3.3 **Governance Mechanism**

EA governance mechanisms as discussed in (section 2.2.9) are related to decision making, change management, and the development of guidelines and architecture principles at the enterprise levels to enact and sustain the relevance of EA endeavors over time. EAM governance in the method being developed is decentralized down the enterprise levels meaning; each enterprise level has some level of autonomy that allows it to be self-governing; but centralized for each enterprise level and implemented through the *big-picture* at that level.

The EAM governance mechanism proposed here provides a context in which the respective domain governance mechanisms are coordinated and co-evolved. During early big-picture meetings following the instantiation of the method, the various EA domains agree on a set of control processes, principles, standards, roles and responsibilities based on the various enterprise governance mechanisms (e.g. corporate governance, and EA domain governance mechanisms) that already exist within the enterprise. After the development of the EA governance mechanism, it serves as a guide for the coordination and evolution of the various EA domain governance mechanisms. Therefore, the ensuing EA governance mechanism should not be too restrictive that it does not allow for the evolution of the EA domain governance mechanism, or too vague that it loses its essence; that is to coordinate and govern the co-evolution of the various EA domain governance mechanisms, and the EA components over time. The EA governance mechanism is reviewed collectively during *big-picture* meetings when the need arises.

Down the enterprise levels, an upper level EA governance mechanism provides a guideline and context for the development of a more detail EA governance mechanism at a lower level. This allows for the governance mechanisms to be consistent across EA domains, and down the enterprise levels. Also, just as changes in EA domains governance mechanisms could influence changes in the EA governance mechanism and vice-versa at an enterprise level; changes in a lower level EA governance mechanism could inform changes in an upper level EA governance mechanism. In this way, the development and refinement of EA governance mechanisms occur vertically (top – down and bottom - up) and horizontally (EA – to – EA domain, and EA domain – to – EA).

6.3.4 **EA Change Management Mechanism**

Change management is closely related to EA governance. In the ensuing method, change management, like EA governance, is decentralized down the enterprise level and centralized at each enterprise level. Change management at an enterprise level is done through the *big-picture* meetings (either periodic or triggered). For instance, when there is a change to be made in one EA domain, at a *big-picture* meeting, the change is reviewed in the context of the *big-picture* and the implications for the other domains are evaluated. The reviewed *big-picture* then forms the basis for further work in the various EA domains. In this way, the *big-picture* enforces alignment amongst the various EA domain efforts and supports their co-evolution. However, changes that are *local* (do not have effect on the *big-picture*) to an EA domain are managed locally within the EA domain. This arrangement is in line with the view of Interviewee G "... so that you fit your change management process to the level of visibility of the change."

Also, to enable the agility and the "lightness" of the change management process, the change management efforts should be focused on up-coming and on-going EAM endeavors whilst keeping an eye on the effect of the changes on the big-picture and future EAM endeavors. Detail change management processes for a future EAM endeavor should be postponed till the endeavor is proximate enough. The EA change management mechanism should build on the existing change management mechanisms within the enterprise, and similar to governance, changes should be managed horizontally and vertically.

6.4 The Agile EAM Method

This sub-section develops the structure of the Agile EAM Method (The Agile EAM Structure" within which the proposed EAM mechanisms are employed. The word "Structure" is used here to avoid the confusion that might ensue by using "framework" because of the totemic status that "framework" has attained within the EA community. The Agile EAM Structure is developed based on a relationship drawn between dynamic capabilities and EAM. Further, an Agile EAM process and procedure for integrating EA components are proposed.

6.4.1 Enterprise Architecture Management as a Dynamic Capability

In view of the relationship established amongst enterprise architecture (specifically "as-is" and "to-be" architectures), enterprise architecting, and enterprise architecture management in section 2.2.3, the thesis proposes EAM as a dynamic capability **(EE7)** and reflects Figure 1 in the lights of Figure 3 to form the basis for the structure of the agile EAM method being developed in this thesis. Accordingly, the "As-is" architecture can be viewed as existing operational capabilities, the "To-be" architecture as reconfigured operational capabilities, enterprise architecting as integrating dynamic capability, and EAM as sensing, learning and coordination dynamic capabilities.

As represented in the Figure 13 below, the sensing capability of the EAM is employed to continuously scan the internal and external environments of the enterprise for internal and external stimuli (change driver) to which the enterprise should respond. These agility drivers trigger the review of existing operational capabilities (to-be architecture) for misfit and make recommendations for appropriate response (e.g. see Overby et al (2005) and also 2.1.1). This could result in several learning opportunities for example: first, learning about the agility drivers; second, learning about whether there is misfit; third learning about what caused the misfit; and fourth; learning about what needs to be done to restore the existing operational capability (as-is architecture) to fitness. When this is carried out in a collaborative manner, example as described in 6.3.1, it could establish common understanding amongst relevant stakeholders and also contribute to organizational learning and organization knowledge.

When a consensus is reached on the appropriate response to be taken, the integration dynamic capability (i.e. enterprise architecting) is evoked to integrate the common understanding that has been formed, individual knowledge, and existing organizational knowledge in the repository (EA repository) into the existing operational capability (as-is architecture) to reconfigure it into a reconfigured operational capability (to-be architecture).

Further, since enterprise architecture is seen as a representation of a complex system (the enterprise), a change in one aspect might lead to predictable and/or unpredictable changes in other areas as well. Therefore, there is the need to coordinate efforts amongst the constituent parts in order to reach a desired goal without creating significant drawbacks elsewhere; that is to promote the optimization of the whole rather than local optimization. In this regard, the EAM serves as a coordinating dynamic capability by establishing linkage amongst the various architecting efforts, tasks and resources, as well as fostering synergies amongst the ensuing EA domain components to form IT-enabled resources with desired emergent capabilities.

The reconfigured operational capabilities (as-is architecture) become the new operational capabilities pending a periodic reconfiguration or a reconfiguration triggered by an agility driver. The knowledge created (e.g. architecture descriptions, lessons learnt, etc.) during the process is stored in the EA repository and used for future EA endeavors and organizational awareness creation initiatives **(EE6)**. The EA repository provides support to the sensing, learning, integration and coordination capabilities. The EAM conceptualized as a dynamic capability provides sensing and response mechanisms that contribute to the agility of the firm (see Nazir & Pinsonneault, 2012; Overby et al., 2005; Seo & La Paz, 2008).

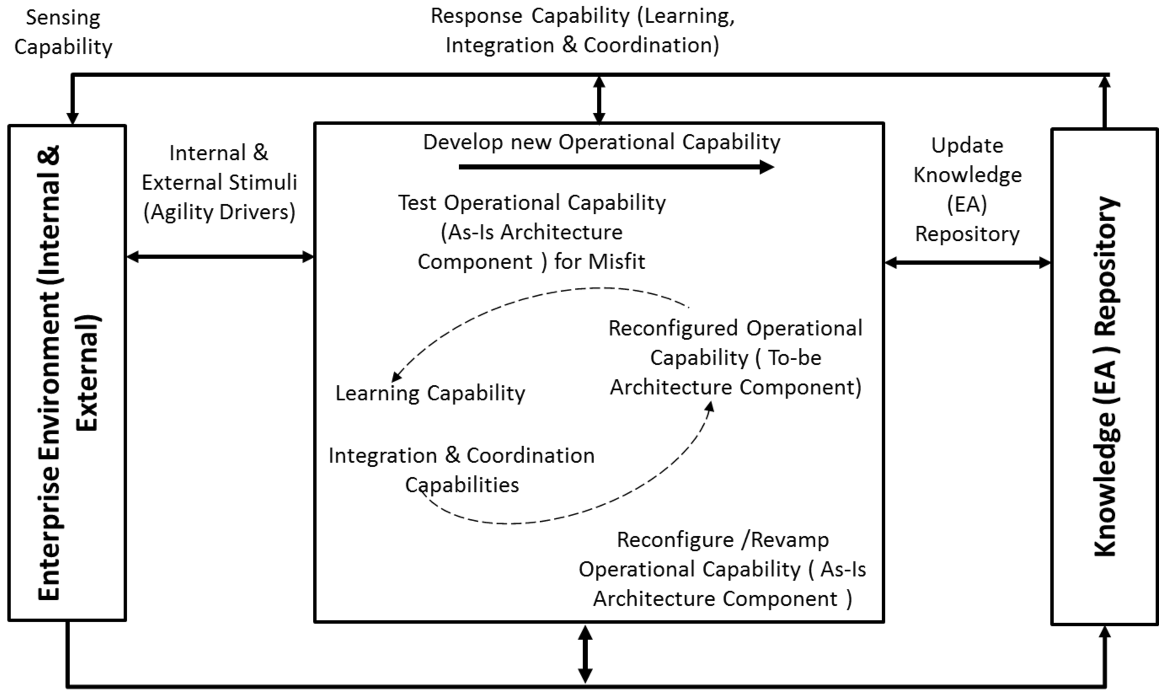


Figure 13: EAM as a Dynamic Capability

6.4.2 The Structure of the Agile EAM Method

As proposed in 6.1, EAM efforts occur at different levels of the enterprise. These are strategy, portfolio, program and project. The concept described in 6.4.1 and represented in Figure 13 is therefore replicated for each of the EA management levels and illustrated in Error! Reference source not found. The EA components roduced at each of the EA management levels is indicated to the right. The proposed structure is informed by four major reasons: first, the need to have targeted EA efforts towards reconfiguring or creating new specific EA components; second, the need to pursue fluxing especially through dynamic capabilities and modular reconfiguration (see Nissen (2014) and also section 3.3.3); third, the need to provide a guide for integration of the various EA components; and forth, the need to represent and communicate the EA in the detail, at the level of abstraction, and with the language that are relevant to different stakeholders.

At the EAM Strategy Level, the EAM efforts relate to the evaluation of the EA for misfit, and revamping the EA to meet the contingencies of the enterprise. In organizations that have multiple business domains, the efforts at the EAM strategy level reconfigure the various business domain architectures or segment architectures(see TOGAF(2009)) in a way that creates synergy amongst them; for example by sharing enterprise capabilities across the various business domain (or segment) architectures. Also, during opportunity explorations, the enterprise could create new EA for the new business by quickly combining enterprise capabilities to create the new business domain (or enterprise). When there are needs to restore enterprise capabilities to fitness or create new ones, a request is sent to the EAM portfolio level to get that done. Unnecessary business domain architectures, e.g. resulting from the closure of a business domain or from a merger, are retired to keep the EA lean.

At the EAM Portfolio level, the EAM efforts are geared towards detecting misfit and restoring fitness in enterprise capabilities, creating new enterprise capabilities from modular reconfiguration of enterprise solutions, and retiring unnecessary ones. When there are misfit enterprise solutions detected in the enterprise capabilities, a request is sent to the EAM Programs level to reconfigure the misfit enterprise solution or create a new one.

At the EAM program level, output of related EA projects (i.e. enterprise features) are combined to form enterprise solutions. When these solutions become misfit, they are revamped or retired, and new ones are created through modular reconfiguration of enterprise features. When an enterprise feature is found to be misfit, a request is sent to the project team to revamp or develop a new one. The EAM project level is the lowest level, at which EA features are developed, revamped, and retired.

Also, as illustrated in Figure 14, the agility drivers could occur at any of the levels, requiring targeted efforts at that level. The EAM efforts triggered at the various levels must be viewed in line with the EA strategy and the *big-picture* at that level and how it relates to the EA efforts at other enterprise levels. Knowledge (e.g. architecture descriptions, models, lessons learnt, prototypes, etc.) produced from EAM efforts at all the enterprise levels are stored in the EA repository. Hence, the knowledge in the repository is updated by those who produced it. For instance, updates on EA features are done by the EAM project team, updates of EA solutions are done by the EAM programs team, and so on. This arrangement is supported by practices in some enterprises. For instance, recounting his lived experience, Interviewee B stated"… it's [updating the EA repository is] done in different levels and it's maintained by the persons who are responsible in that level..."

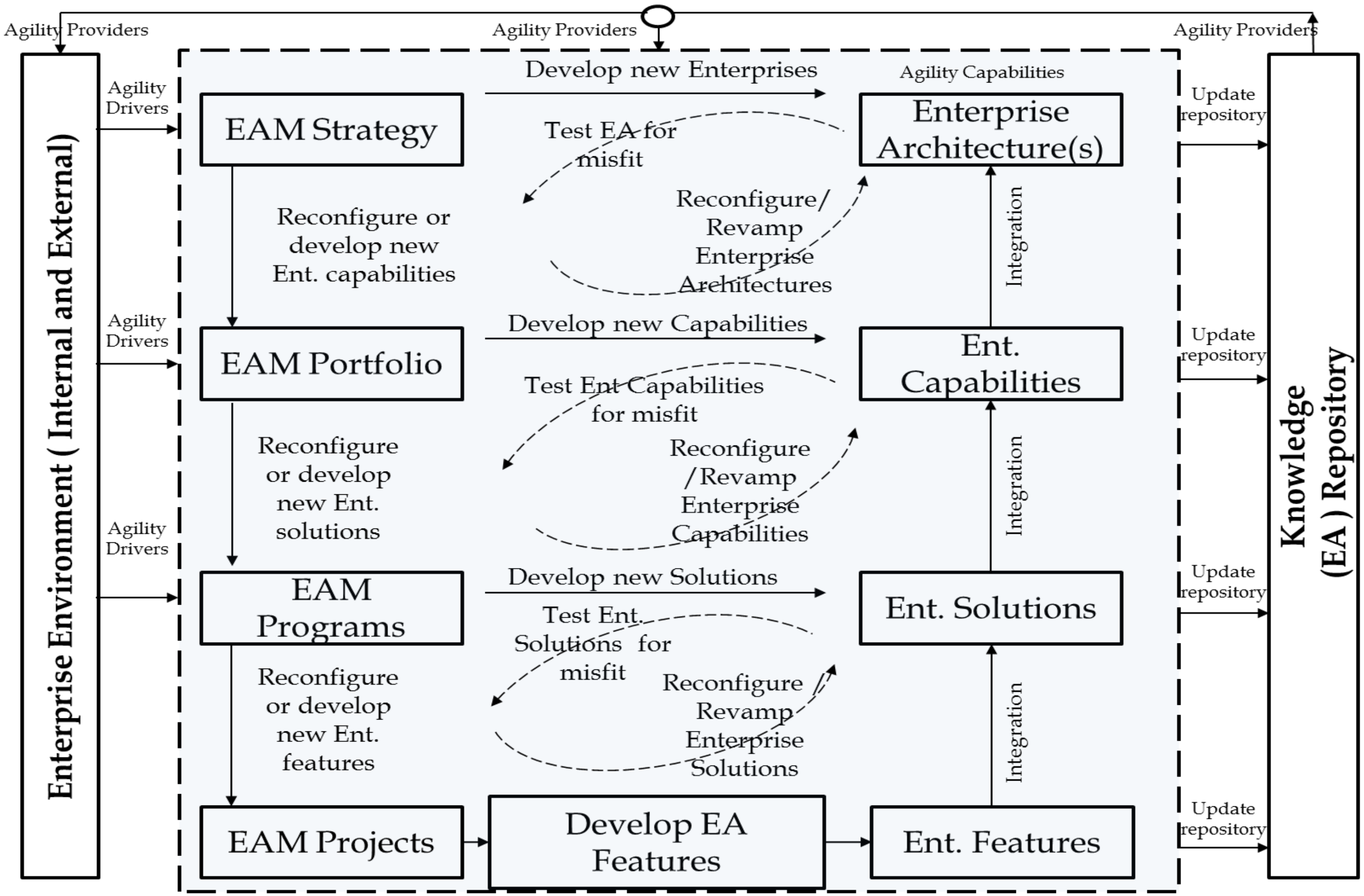


Figure 14: The structure of the Agile EAM Method

6.4.3 **Integration of the EA components**

As discussed in 6.2 above, EA components, in the method being described here, are conceived as IT-enabled resources that are born out of the synergistic relationship amongst the various EA domain components. These components are formed by the EA management efforts at the different enterprise levels, and EA components at lower enterprise level are modularly configured to form EA components at higher enterprise levels. This arrangement calls for two major considerations during the architecting of these EA components: modularity and the ability to reconfigure (redesign) the EA components during use (see section 3.3.2.)

Also, to ease traceability, it is proposed that each of the EA components carries four basic pieces of information as shown in Figure 15. These pieces of information include: the level of agility drivers that trigger the development, reconfiguration or expiration of the EA component; the type of component it is, the EA components that it is made up of; and the EA components to which it is related.

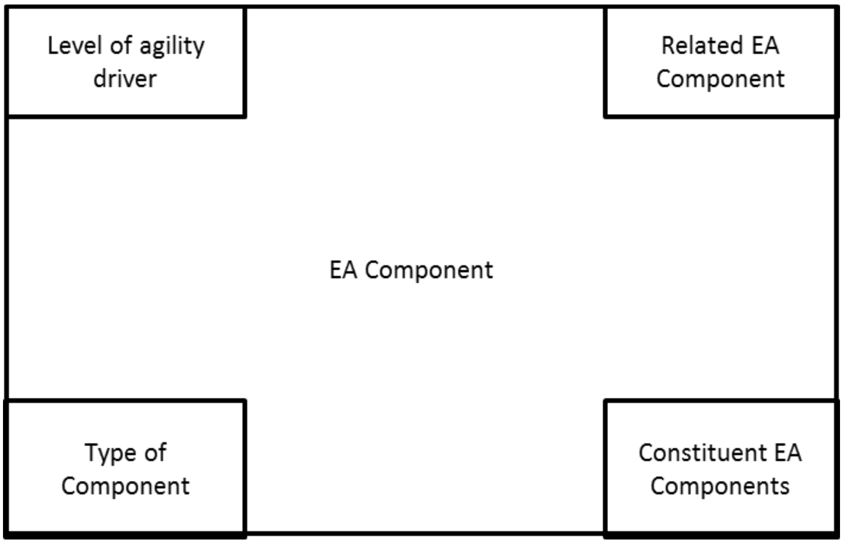


Figure 15: Structure of the EA Component

Furthermore, Figure 16 illustrates how the various EA components integrate together. The agility drivers could be a new change in the environment or a request from an upper enterprise level. For instance, agility drivers to an enterprise solution (at the enterprise business process level) could either be from a change in market demand that triggers the development of a new enterprise solution or renewal of an existing one; or a request from the enterprise operations level to develop a new, or revamp an existing enterprise solution.

The type of the EA component (bottom left corner) could be either ad-hoc or main. Ad-hoc EA components are used to handle agility drivers that are perceived to be temporary but necessary to be responded to through managed improvisation. These ad-hoc EA components are either retired when the agility drivers to which they respond are no more, or they are integrated into the EA as main EA components when the agility drivers to which they respond are perceived to have become persistent. For instance, during the formation of a virtual enterprise, the constituent enterprises could develop ad-hoc enterprise capabilities and contribute them to the formation of such a virtual enterprise. When the virtual enterprise does no more exist, the ad-hoc enterprise capabilities are retired. However, when the virtual enterprise remains persistent or evolves into an enterprise, then the ad-hoc enterprise capabilities are integrated as main enterprise capabilities within the new enterprise.

The "constituent EA components" refer to the lower level EA components that make up the EA component. For instance, the enterprise architecture has enterprise capabilities as its constituents; an enterprise capability has enterprise solutions as its constituents, and so on. The "Related EA Components" refers to EA components to which the EA component relates to. For instance, an enterprise solution could be related to multiple enterprise capabilities. Likewise, an enterprise capability could be related to the enterprise architectures of multiple business domains.

These pieces of information are relevant for traceability of the various EA components, as well as the modeling of the EA in a way that makes it "collapsible" and presented to different stakeholders. For instance, at the enterprise level, stakeholders might be interested in how the various business domain (segment) architectures relate to each other and how they can leverage enterprise capabilities; whilst, at the enterprise operations level, stakeholders might be interested in seeing the EA as interrelated enterprise capabilities and how they leverage enterprise solutions amongst themselves. In this way, the EA could be seen at each enterprise level as a representation of the EA components at that enterprise level, the interrelationship between them and their environment and the principles that govern their evolution over time. Obviously, this has implications for EA modeling and representation tools that could be used in order to be able to derive these different architectures from unified single enterprise architecture. However, depending on peculiarities of the enterprise; including level of complexity, EA maturity, the need for agility, dynamics in its EA, etc., other tools with lesser capabilities could be used.

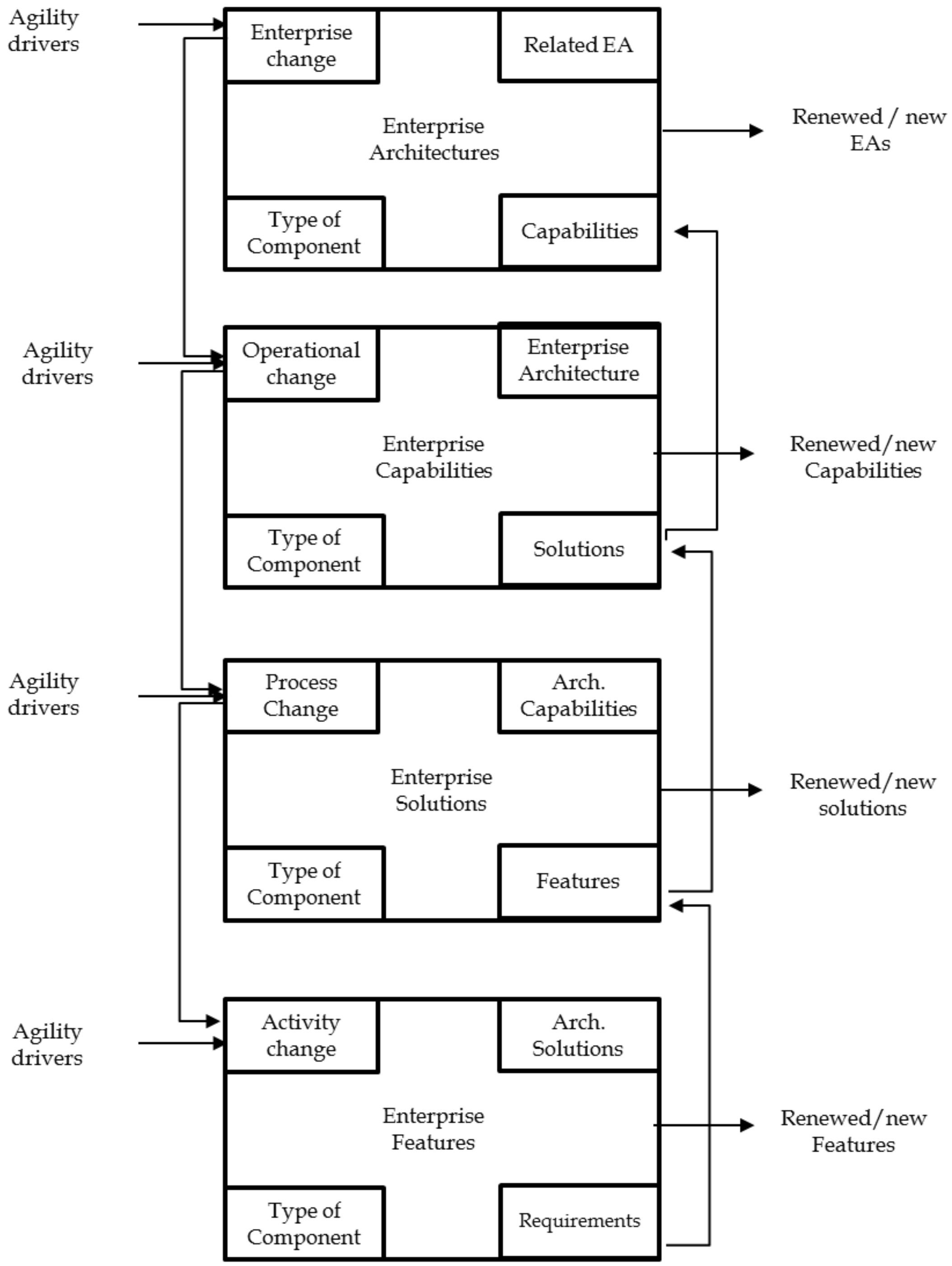


Figure 16: Integration of EA Components

6.4.4 **Agile Enterprise Architecture Management Process**

A description of the agile EAM structure was presented in section 6.4.2 and the integration of the EA components was also presented in 6.4.3. This section builds on these descriptions and presents a process by which the agile EAM process is carried out within the agile EAM structure.

Based on continuous improvement (CI) concepts presented in section 3.2.4 and the collaborative decision making mechanism presented in 6.3.1, the thesis proposes an agile EAM process illustrated in Figure 17 below. The process is built around Deming's Plan-Do-Check-Act cycle for continuous improvement. At the "Plan" stage, the stakeholders from the various EA domains through a collaborative process arrive at a consensus called the big picture. The big-picture provides the common understanding on what is to be accomplished at the end of the cycle (e.g. the development of an EA component); the goal fulfillment criteria (e.g. time, scope, cost, extent of change to be accommodated, etc.); the architecture principles to be adhered to (e.g. the level of modularity, interoperability, scalability etc.), etc. The big-picture is then used for further planning in the various EA domains.

At the "DO" stage, the various domains carry out their respective tasks towards achieving the common goal that has been agreed on in the *big-picture*: for instance, the development of a new EA domain component; reconfiguration of an existing EA domain component; modeling or documenting of a new component etc. At the "Check" stage, the various EA domains compare their actions and developed EA domain components to their various EA domain plans and the *big-picture*. Variations, lessons learned, and suggested next lines of action are noted. At the "Act" stage, the various EA domains convenes again to integrate their various EA domain efforts to reach the common goal, share lessons learned, and agree on the next lines of action. The realized artifacts e.g. EA component, lessons and models, are made available in the EA repository. Requests to reconfigure or create new EA components can also be generated and added to the "backlog" of a lower enterprise level. The cycle is repeated again to either improve on a particular *big-picture* goal, or to take on a new big-picture goal for the next EAM effort.

Each of the cycles should have a unique name that describes what the cycle seeks to achieve. For instance, if the EAM effort is towards revamping a Customer Relations Management (CRM) capability or integrating an information system and a customer management process (CMP) to form an IS-enabled CMP, the cycles could be named as "Revamping of CRM Capability" and "Integration of IS and CMP" respectively. This naming gives focus to the EAM endeavor and also reminds stakeholders of the *big-picture* goal that the particular cycle seeks to achieve.

Through this process, the EAM function will be able to coordinate architecture efforts or activities that are being carried out in multiple domains, leverage resources amongst the activities, and form synergy amongst the ensuing EA domain components to form IT-enabled resources i.e. the EA components. The EAM process fosters discrete, incremental and continuous development of the EA components.

Apart from coordinating efforts amongst EA domains, the EAM process proposed here could be applied at other levels of architecture efforts. In enterprises that have multiple business domains or segments, the EAM process could be used to coordinate architecture efforts amongst the various business domains or segments as illustrated in appendix 2. For example, at the enterprise level, the EAM process could coordinate segment architecture efforts and establish synergy amongst segment architectures to form the enterprise architecture. The EAM process could also be used within EA domains too, as represented in appendix 3. For instance, in cases where an EA domain consists of sub-domains, the EAM process suggested here could provide a means of coordinating the efforts of the various sub-domains to ensure that the ensuing EA domain component represents the collective interest and optimal solution for that EA domain.

Furthermore, during EA development, the proposed EAM process could be used to coordinate the efforts of multiple development teams, for instance agile teams. As represented in appendix 4, at the PLAN stage, the various development teams and other stakeholders meet to decide on the big-picture, the immediate set of architecture artifacts to be developed, and the action to be taken by each team (e.g. develop a unified Product backlog). Based on the actions to be taken, the various teams are assigned prioritized list of items to be developed during a particular cycle (e.g. updating the sprint backlogs). At the DO stage, the various teams develop the assigned artifacts (e.g. sprinting) and hold team level meetings (e.g. daily sprints). At CHECK stage, the teams review the developed artifacts against plan or requirements and document lessons learned and next lines of action (e.g. Sprint review). At the ACT stage, the teams and other stakeholder come together again, integrate and review the developed EA artifacts against the big-picture goal, share lessons learned, and decide on next lines of action (e.g. Sprint retrospective). During development, the teams could use different methods for example off-the-shelf methods such as Waterfall, Spiral and Scrum; tailored versions of off-the-shelf methods; or combination of off-the-shelf methods.

In effect, the EAM process can be used to coordinate EA related activities at different levels within the EAM structure. The appropriation of the EAM process, however, is based on the peculiarity of the enterprise and the EAM effort at hand.

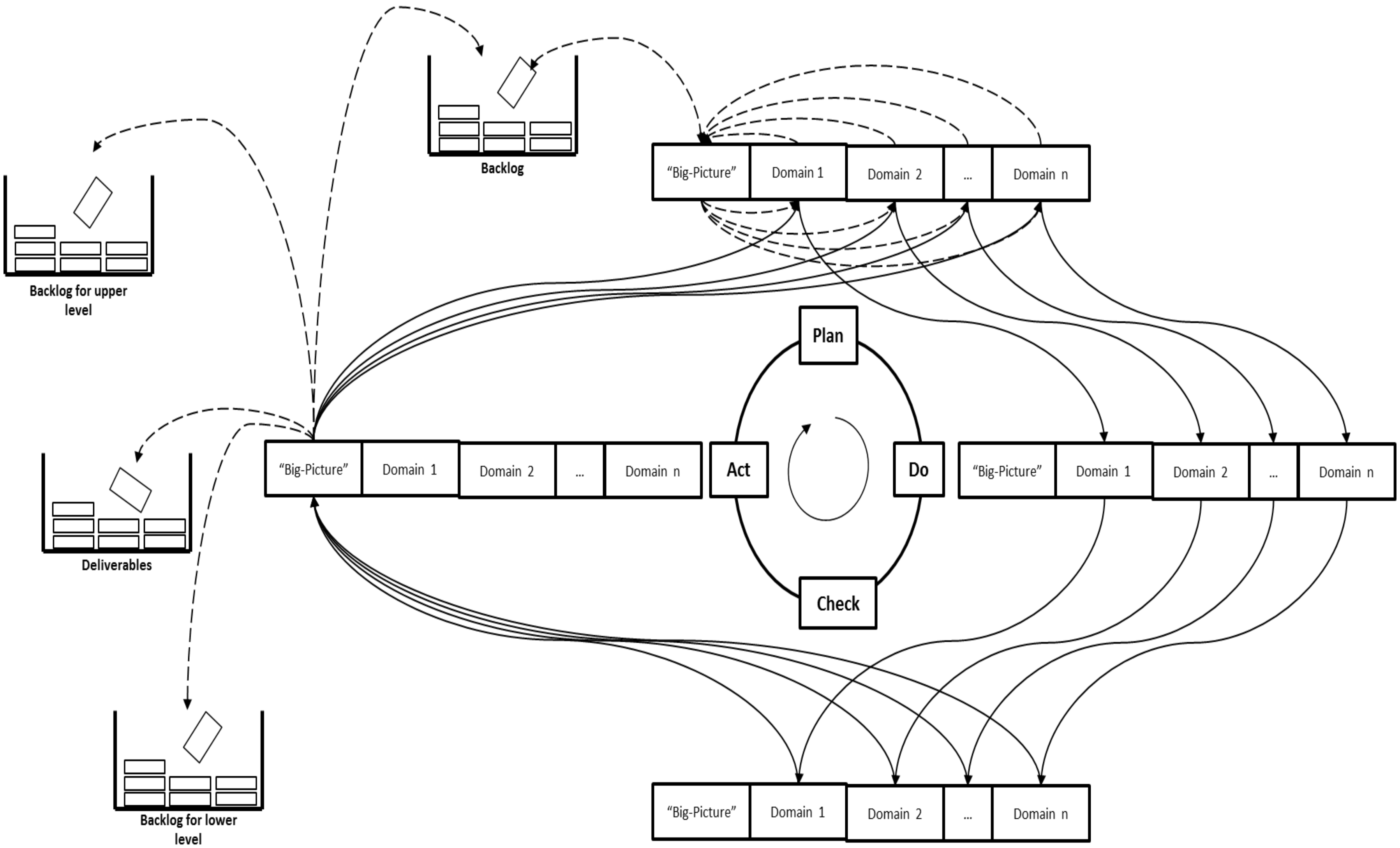


Figure 17: The Agile Enterprise Architecture Management Process

6.5 Instantiation of the Agile EAM Method

This sub-section provides guideline on how the Agile EAM Method could be instantiated in an enterprise. Instantiation allows the enterprise to develop a unique version of the Agile EAM method to fit its context. The guideline covers the selection of EA domains and enterprise levels, representation of the EA component as "IT-Enabled Resources", the appropriation of EAM mechanisms and roles, and the development and enactment of EA principles and guidelines.

It is recommended that the instantiation occurs at a level as high up the enterprise as possible; preferably at the highest level of the enterprise. The instantiation is done through a preliminary "big-picture" meeting at the level of the enterprise that the method is being instantiated. The preliminary big-picture meeting should consist of decision makers from different aspects of the enterprise whose decisions can be binding on the enterprise and easily enforced. After the initial instantiation, the preliminary team is replaced by the actual team at that level and the teams for the other levels are formed and enacted. The instantiation and development of an enterprise's version of the Agile EAM method should be done progressively. There is also the likelihood that during the instantiation of the method within an enterprise, there could be enterprise specific CSFs or mechanisms that needs to be taken into consideration. These organization specific characteristics might add to the proposed EEs, and or the CSFs that have been found from extant literature.

6.5.1 Selection of EA Domains and Enterprise Levels

The set of EA domains that an enterprise has or might want to adopt probably depends on many factors including the maturity of EA, the sector of the organization, the knowledge and experience of the EA lead, the EA framework adopted, regulation enforcing EA, the strategic intent of EA etc. At the time of instantiating the Agile EAM Method, the enterprise decides on the set of EA domains that is more suitable for it. In larger enterprises (e.g. governments, and multinationals companies), this process might involve consolidating the existing EA domains across multiple business domains (segments). In some cases the existence of different sets of EA domains across different business domains might be inevitable. For example, in the case of a large enterprise with business domains in different countries, the various business domains might have to adhere to different national policies e.g. on information security. Another case that might make the existence of different EA domains across different business domains in an enterprise inevitable is when the different business domains are into different industries e.g. one is into manufacturing, and another is into services. In such cases, the enterprise should decide on unify the EA domains or use meta-domains at the enterprise level and allow the various business domains to have their unique set of EA domains. The enterprise could reach this decision by answering the questions: how different are the EA domains across the various business domains?; to what extent do or should the various business domains leverage EA components?; and what strategic benefits and limitations will unifying the EA domains bring?. Collaboratively, the enterprise should select a set of EA domains that will enable it derive optimum enterprise level benefits from EA.

Also, at instantiation, the enterprise should select the EAM levels that are deemed appropriate. The EAM levels could be aligned to existing enterprise structures e.g. organizational levels, decision making levels, and project management levels. Nevertheless, the enterprise could use the enterprise levels, EAM levels suggested in sub-section 6.1 above as a starting point. When the enterprise levels and EAM levels are selected, the EA components levels are aligned to them.

6.5.2 **Representation of EA Components as "IT-Enabled Resources"**

Following the selection of the EA domains and enterprise levels, the enterprise should represent the EA Components as "IT-Enabled Resources" as shown in Figure 10**.** This representation guides future architecture efforts for instance, the specific architecture components to be produced or revamped. Furtherance to this, the big-picture team should collaboratively develop and populate the Extended EAM Grid for the enterprise.

6.5.3 **Appropriation of EAM Mechanisms and Role**

The suggested EAM Mechanisms for example decision making, change management and governance mechanisms are instantiated using existing structures. For instance, the EAM governance mechanism is developed within the enterprise in such a way that allows the other governance mechanisms e.g. corporate, business, and IT governance mechanisms, to function efficiently. At instantiation, the various governance mechanisms should be reviewed and the EAM governance mechanism designed as a meta-governance mechanism that binds the various governance mechanisms together. Likewise, other mechanisms, including change management and decision making mechanisms should be developed as described sub-section 6.3, preferably by leveraging existing structures as much as possible.

The roles and responsibilities that are associated with these mechanisms should also be enacted by using existing roles and resources within the enterprise. For instance, each EA domain should have a lead at each enterprise level who represents the EA domain at the "big-picture" at that enterprise level. Also, there should be a lead for each enterprise level who facilitates the architecture work at that enterprise level, and represents the enterprise level at a lower level (e.g. to explain higher level architecture changes that need to be addressed in lower level EA components) and at a higher level (e.g. to explain lower level architecture change that needs higher level permission). This role can be played by multiple people depending on the issue at hand. For instance, the general EAM facilitation role could be performed by the enterprise architect. Further, depending on the issues to be explained to the upper enterprise level or the lower enterprise level, the enterprise architect or an EA domain lead could acts as the facilitator. For instance, if the issue is mainly architectural issues, then the enterprise architect could lead. However, if the issue is more technical (business and IT), then the lead for the EA domain requiring the change could act as the facilitator.

6.5.4 Development and Enactment of EA Principles and Guidelines

The principles and guidelines for EA should be developed as meta-principles that bind the other architecture principles e.g. business architecture, and information systems architecture principles together. At instantiation, the various EA domain principles should be discussed and a set of EA principles developed that embody the EA domain principles at a higher level such that consistency is maintained amongst the EA domain principles even as they evolve. For example, EA principles should specify: how the various EA domain and segment efforts should relate to the enterprise strategies; the extent of modularity, scalability, and stability of EA components; how detail, coherent and complete should EA components be described and communicated at each level; how changes to the big-picture should be managed in order to maintain alignment amongst the various EA domains efforts; the conditions under which the EA components could be changed considerably; and guidelines for developing and managing ad-hoc EA components.

6.6 Demonstration and Evaluation of the Agile EAM Method

The demonstration and evaluation of the agile EAM Method is presented in this sub-section. The compatibility of the Agile EAM method with the Architecture Development Method of one of the most widely used EA frameworks (TOGAF) is discussed. It is envisaged that the Agile EAM method improves the agility of the ADM, or any other architecture development method that can be adapted to fit the Agile EAM process. The section also presents a scenario analysis that conceptually demonstrates and evaluates the efficacy of the Agile EAM method in addressing the main motivation of the thesis; the construction of an EAM method to improve the agility of the EAM function and the agility of the enterprise as a whole.

6.6.1 Demonstration of Compatibility with TOGAF's Architecture Development Method (ADM)

TOGAF is one of the most widely used EA frameworks. It has an architecture development method (ADM) that consists of ten constructs; a preliminary, a cycle of Phases from A to H; and a requirement management construct that is central to the Phases (TOGAF, 2009, pp. 67–209).

The preliminary defines the organizational context, identifies major stakeholders and their requirements, defines the business directives for the architecture work, and secures the commitment of stakeholders to the architecture work. Phase A (Architecture Vision) is concerned with establishing the vision and scope of the architecture work and obtaining permissions. Following Phase A, Phase B (Business Architecture) develops the business architecture to support the vision developed in Phase A. Phase C (Information Systems Architectures) develops the data and application architectures to support the business with complete, consistent, stable and understandable data, and the system that is needed to process such data. Phase D (Technology Architecture) defines the technology architecture that defines the physical realization of architectural solutions specified in Phases A to C. Phase E (Opportunities and Solutions) defines the projects, programs, and portfolios needed to deliver the target architecture. At Phase E, all the architectural artifacts developed in previous phases are consolidated, and co-existence and interoperability issues are examined and clarified. Phase F (Migration Planning) is concerned with creating migration plans to realize the transition architectures defined in Phase E. Phase G provides architectural oversight of the implementation whilst Phase F provides procedures for managing changes to the new architecture. Lastly, the Architecture Requirement Management construct defines processes for identifying, storing, and managing architecture requirements throughout the ADM. The ADM is illustrated as a cycle and is iterative over the whole cycle, between phases, and within phases (TOGAF, 2009, p. 53). It has four main iterations between phases namely; Architecture Context Iteration (Preliminary and Phase A), Architecture Definition Iteration (Phases B to F), Transition Planning Iteration (Phases E and F) and Architecture Governance Iteration (Phases G and H) (2009, p. 223). The TOGAF ADM is regarded in the academia and practice as an agile EA development method mainly because of these iterations. For instance, Buckl et al (2011) considers the ADM in TOGAF 9 as a related method for agile EAM, and Interviewee B claimed "…after all if you are thinking of now a days, even TOGAF 9 is agile method in a strange sense... It's agile because there are those turn-backs".

The Agile EAM Method could support the ADM and add value in terms of agility to the ADM, if the ADM is implemented within the Agile EAM process. Following the provision within TOGAF 9 to adapt the ADM to an enterprise-specific ADM (2009, p. 56), and the guidance to implement the ADM at multiple levels within an enterprise (2009, p. 223), the thesis demonstrates an implementation of the ADM within the Agile EAM Process. The Preliminary, Phase A, and Phase E are collaboratively considered by the various EA domain leads during big-picture meetings. In this way, the organizational context, constraints, business directives, criteria for goal fulfillments, target architecture, and transition architectures are collectively considered and the implications for the various EA domains are specified. Architectural efforts within the various EA domains are therefore focused on producing targeted architecture artifacts towards delivering a specific business directive. Also other issues including interoperability and co-existence issues are discussed upfront to govern detail work within the various EA domains.

Furthermore, Phases B to D are planned in parallel and iteratively using the decisions reached at the big-picture meeting. Following the planning, the architecture artifacts are developed and reviewed respectively during the DO and CHECK stages of the Agile EAM process. Phase F is done at the ACT stage of the Agile EAM process. If necessary, another cycle is executed to integrate the architectural artifacts produced into the EA. Phases G and H, and Requirements Management are inherent within the Agile EAM Structure, and the Agile EAM process, and are therefore executed as day to day EAM mechanisms. In this regard, the ADM becomes intervention focused. A business imperative is collaboratively reviewed; a target architecture and a set of architectural solutions (transition architectures) are conceived and prioritized; then each architectural solution is taken and developed through the DO, CHECK and ACT stages of the Agile EAM Process. That is, architectural endeavors in the various EA domains during a cycle of the Agile EAM Process are focused on delivering the chosen or targeted architectural solution(s). Furthermore, architectural issues e.g. alignment, interoperability and consistency, associated with the architectural solutions are considered upfront and enforced throughout the process, and not after the EA domain architectures (Phases B to D) have been developed.

Also, the ADM could be deployed within the Agile EAM Structure proposed in this thesis, especially when it is adapted to fit the Agile EAM process. In that way, the coordinating capability of EAM, as explicated in the Agile EAM Structure, is employed to coordinate the efforts of multiple adapted ADMs at different levels of the enterprise. The EA outputs (the "building blocks" as in TOGAF) of the ADM are aligned to the EA components at the different enterprise levels, which further makes the visioning, planning, development and governance of the adapted ADM processes at the different enterprise levels more focused.

6.6.2 **Illustrative Scenario Analysis with a Hypothetical Case**

The hypothetical case used here is from Sessions(2007). The full case description is provided in appendix 5. In summary, MedAMore, a chain of drug stores, developed a system (MAM) that enabled it to efficiently execute its operations. However, when MedAMore expanded by acquiring additional regional chains MAM was found to be restrictive and unable to cope with the expansion. It created discourse between the business and IT groups. For example, the IT group made expensive IT investments that were ignored by the business group. MedAMore seeks to use EA as a vehicle to unite its business and IT groups, and delivery more cost effective systems that will enable business agility. Bret (the Vice President of Business) and Irma (CIO) were tasked to develop a common EA (called MAM-EA), and to make it work.

In order to use the Agile EAM method developed in this thesis, Bret and Irma should constitute the preliminary "big-picture" team and instantiate the method at the highest enterprise level. They should then identify the EA domains e.g. business, information, application and infrastructure EA domains as implicated in the scenario; and the enterprise levels. Further, they need to populate the *EA component as IT –enabled resources* chart and the Extended EAM Grid for MedAMore to the extent possible. Thereafter, Bret and Irma need to form the actual big-picture team at the highest level by appointing an enterprise architect and leads to the various EA domains, and bringing them together to formulate the EA visions and strategies (long-term and short-term), EAM mechanisms (e.g. governance, change management, decision making, etc.) and EA principles (e.g. scalability, modularity, consistency, completeness etc.). The enterprise architect might be sourced from inside e.g. by training existing team members, or sourced from outside. Bret and Irma remain part of the team and provide oversight responsibilities. They will also coordinate the participation of other stakeholders, as and when the need arises.

The team then decides on the next steps of actions to be taken for instance, document the reference EA and EA Components as per the *EA components as IT-enabled resources* chart, describe the target architecture, and identify the transition architectures needed to achieve the target architecture. Concrete deliverables are identified and prioritized, and projects, programs and portfolios are instituted to deliver them. The Agile EAM process is initiated for each of the endeavors to coordinate the architecture efforts across the various EA domains. Similarly, EAM teams are constituted at other levels of the enterprise to handle EAM endeavors at the various enterprise levels.

In cases of strategic endeavors, for instance the acquisition of new regional chains, such endeavors are discussed at big-picture meetings at the highest levels. The peculiarities of the endeavors may include inheriting new processes, procedures and systems, and satisfying regional requirements. The architectural work needed to support such endeavors is considered and appropriate actions are collectively taken. For instance, appropriate actions could include the reconfiguration of existing or the development of new EA components, and the re-design of inherited processes, procedures and systems to adhere to the EA principles of MedAMore. Targeted EA projects are enacted at different enterprise levels in response to change drivers in MedAMore's internal or external environments e.g. the need to bring newly acquired regional chains online, and the need to share data (information) across the growing number of regional chains.

The description of EA components, lessons learned and other architecture artifacts are stored in the EA repository and are communicated to relevant stakeholders across the enterprise to improve enterprise awareness creation and learning. The EA repository items are re-used during future EAM endeavors to support the agility of MedAMore's transformation endeavors; for instance, expansion (through the acquisition of new regional chains), and enhancement of internal efficiency. The establishment and enforcement of EA principles will promote the scalability, modularity and stability of the MAM and the MAM-EA.

Furthermore, the collaborative nature of the mechanisms in the Agile EAM method will not only promote the alignment and co-evolution of business and technical solutions, but it will also unite the business and technical teams. Lastly, the nature of the Agile EAM process will promote the continuous improvement of the EA components e.g. warehouse management capability, information sharing capability, and order processing solutions; as well as the continuous improvement of the EAM processes and mechanisms.

Conceptually, the use of the Agile EAM method at MedAMore has the propensity to improve the agility of the EAM function within MedAMore as well as the agility of MedAMore as a whole. However, the Agile EAM Method, and its mechanisms need to be evaluated in a real enterprise to prove the practical efficacy of the method.

6.7 Chapter Summary

This chapter was dedicated to the design, construction, demonstration and evaluation of the Agile EAM method in fulfillment of research question three: "how could these essential elements be conveyed in the form of a method?" The Agile EAM method proposes: first, that through a collaborative effort, EA components should be viewed and developed as IT-enabled resources, and EAM mechanisms should be developed based on existing organizational structures in a way that brings relevant stakeholders to engage in consensus building and participatory design processes(EE3 & EE8); second, an EAM structure that enables EAM to provide sensing and learning capabilities, and to coordinate (coordination capability) knowledge integration (Enterprise Architecting) efforts (EE7); third, an EA repository as part of the EAM structure that allows EA artifacts including lessons learnt to be stored and re-used for enterprise awareness, and further enterprise architecting efforts(EE6); fourth, an Agile EAM process as a mechanism to develop and refine both long and short term EAM visions, strategies, goals, purpose and criteria for goal fulfillment, and to continuously and incrementally deliver EA artifacts (EE1, EE2 & EE4); and fifth, a structure of an EA component and a guide for the integration of EA components that allow the enterprise architecture at a particular enterprise level to be represented in terms of the EA components at that level (EE5).

The compatibility of the Agile EAM method with TOGAF's ADM was demonstrated. The focus was on how the Agile EAM process can contribute to the agility of the ADM, and how the Agile EAM Structure can be employed to coordinate multiple ADMs at different levels of the enterprise. Further, following the recommendations of Peffers et al. (2012), a hypothetical scenario was used to demonstrate and evaluate the efficacy of the method in supporting the agility of the EAM function and the agility of the enterprise as a whole.

7 Chapter 7 Discussions and Conclusion

This chapter presents the summary of the whole research including the research processes and findings (outputs). How the three research questions were answered in the thesis are discussed. The chapter also presents the limitations of the research and suggests possible areas for future research.

7.1 Discussions

Enterprise architecture has the ability to contribute to the agility of the enterprise if it is conceived, developed, maintained, and evolved in a way that senses the agility drivers and strengthens the agility providers and agility capabilities of the enterprise. The EAM function within enterprises therefore needs a method that can enable it to manage EA in a manner that supports the agility of the enterprise. Prior research is either focused on the agility of the EA or the agility of the EAM function without explicit relation to the agility of the enterprise as a whole. The motivation for this thesis was to develop an EAM method that supports the agility of the EAM function and the agility of the enterprise as a whole. Such a method was referred to in this thesis as the Agile EAM Method (Agile EAMM).

To construct such a method, the thesis needed to answer three main questions. The first was to find out the factors that influence the agility of the enterprise; the second was to find out the essential elements that such a method need to possess in order to support the agility of the EAM function and the agility of the enterprise; and the third was to convey such essential elements in a form of a method, called the Agile EAMM. Since the thesis was concerned with the development of an Information Systems artifact in the form of a method, the Design Science Research (DSR) approach was adopted to guide the research efforts.

Base knowledge of the field and for the method design was established by reviewing prior literature on enterprise agility, enterprise architecture, agile and lean principles and values, and suggested theoretical glue that made it possible for the agile and lean principles to be adapted for EAM. From the review of literature on enterprise agility, it was found that agility is multifaceted and can be defined at many levels (e.g. software team level, operations level, and enterprise level), and applied in diverse contexts (e.g. new product development and manufacturing). Also, the situations in which an enterprise finds itself at a particular moment could have significant implications on what represents agility to that enterprise. Consequently, agility has diverse definitions in prior literature. For the purpose of this thesis, a definition was formed for enterprise agility that takes cognizance of the different needs that drive enterprises to want to be agile (competitive bases), and the different enterprise efforts towards achieving agility (e.g. reconfiguration of enterprise capabilities). Further, the literature review on enterprise agility led to answering of research question 1: what factors influence the agility of the enterprise. From extant literature, such factors are agility drivers, agility providers, and agility capabilities. Agility drivers (e.g. changing customer demands) are changes in the enterprise environment that forces the enterprise to change, or want to be agile; agility capabilities (e.g. responsiveness and competency) provide the enterprise the ability to respond to the agility drivers; and the agility providers (e.g. relationship with partners, market sensitivity, and enterprise integration) are the means by which the agility capabilities are achieved. IT is also known to have influence on these factors, for instance by enabling sensing capabilities with which the agility drivers are sensed, and response capabilities with which the enterprise integrates its agility providers and agility capabilities in response to the agility drivers.

Literature review on EA established a relationship amongst enterprise architecture, enterprise architecting, and EAM. EAM was conceived as the managerial endeavor that provides the necessary organizational context (e.g. processes, coordination, tools, structures, resources, relationships, and critical success factors) to enable enterprise architecting efforts to continuously transform the "as-is" architecture towards a "to-be" architecture in response to the increasingly dynamic enterprise environment. Further, literature on EA issues (e.g. organizational benefit of EA, challenges of EA, CSFs of EAM, and EA governance) was examined to understand the "state of affairs" on each of them. Then an attempt was made to answer research question two by performing a systematic search of literature on the components or elements of EAM. Ten sources were selected and reviewed for such components. It was apparent that, though the sources listed some components of EAM, the agility of the EAM function and its relation to the agility of the enterprise has sparingly been treated in the EA literature. A preliminary list of components of EAM was formed to lay the foundation for further work towards answering research question two.

Lean and Agile Principles and Values have started to gain attention in the EA community especially in relation to agile EA and agile EAM. Therefore, attention was turned to the Lean and Agile Principles and Values to understand them, and apply them to EAM. However, from the IS literature, there are hindrances in scaling up agile values and principles, and also in applying them to fields outside software development. It is therefore advised in the IS literature to employ some foundation theories (theoretical glue) that make the application of these principles and values outside software development possible. In this thesis, such justification theories (probable theoretical glue) were suggested for each of the lean and agile values and principles, with focus on EAM agility and enterprise agility. Chapters two and three were dedicated to providing the theoretical foundation for the design process, and thus provided the "foundation theory" construct in the conceptual framework for this thesis as represented in Figure 18.

After laying the theoretical foundation in chapters two and three, the first set of design artifacts was produced. The probable theoretical glue was used to adapt the lean and agile principles and values for EAM, taking into consideration the preliminary list of EAM components in chapter 2. Four agile EAM values, five agile EAM principles, five lean EAM principles, and seven lean EAM wastes were produced. These intermediate artifacts correspond to the "adapted agile principles and values for EAM" and the "adapted lean principles and wastes for EAM" constructs as shown in Figure 18**.** Further, the lean and agile principles were consolidated into the eight essential elements of an agile EAM method (EEs), representing the "the essential elements of an agile EAM" construct in Figure 18. The EEs were then evaluated for relevance by using a semi-structured interview involving eight interviewees. On the average, there existed a moderately strong agreement (ICC = .522, $p < .001$) amongst the interviewees that each of the EEs was essential (mean = 3.91, sd = 0.136). The analysis of qualitative data from the interview showed that the differences in views amongst the interviewees could be attributed to their lived experiences partly informed by the type of organization and their positions. The EEs were also evaluated against those in extant EA literature, and against the CSFs of EAM, agility capabilities, and agility providers.

Based on the evaluation, the second iteration (the construction of the Agile EAM method) in the DSR process was initiated. EAM was conceptualized as a dynamic capability (EE7) that provides sensing and learning dynamic capabilities, and coordinates (coordinating capability) enterprise architecting (integration capability) efforts towards developing and maintaining an agile enterprise. This concept was replicated at the different levels of the enterprise to form the structure of the Agile EAM method (the "Agile EAM Method" construct in Figure 18). Also, within the structure are the EAM components and the EAM levels at which they are produced. The EAM components are IT-enabled resources formed out of synergistic relationship amongst EA domain components; and the EAM levels are the EAM efforts that create and manage the EA components. Therefore, at each enterprise level, the EA could be seen as a representation of the EA components at that enterprise level, the interrelationship between them and their environment, and the principles that govern their evolution over time (EE5). The structure of the agile EAM method also consists of an integrated repository that stores architecture knowledge (e.g. EA components and lessons learned) for enterprise awareness endeavors and re-use during future EAM endeavors (EE6).

Further, the method hinges on collaborative decision making and participatory design processes (EE3) that result in the establishment of a common understanding (EE5) and the creation of the "big-picture" at a particular enterprise level. In view of this, EAM mechanisms (including those for governance, change management, EAM visioning and planning) were proposed (EE1, EE3); representing the "EAM mechanisms" construct in Figure 18. These mechanisms are instantiated based the EA domain and enterprise mechanisms (EE8) that exist within the target organization. An agile EAM process has also been proposed. The agile EAM process is formed around the Deming Cycle consisting of Plan, Do Check/Study and Act stages. It enables the EAM function to coordinate the development of the "big-picture", and the various EA domain efforts towards the incremental and iterative development of the EA at the various levels of the enterprise (EE2). Consequently, the agile EAM process provides the ability to coordinate efforts amongst different EA domains, business domains (segments), and different project teams; and allows for the selection and use of different tools and methods (EE4). It can be employed at different levels within the structure of the agile EAM method and for different endeavors (e.g. creating new EA component or reconfiguring an existing one; updating the repository; evaluating existing EA components for fitness; retiring irrelevant EA components; and integrating EA components into the EA). At instantiation, detail information concerning the agile EAM process (e.g. duration for each cycle, and EA domains) are decided. Also, during the instantiation of the method within an enterprise, there could be enterprise specific CSFs or mechanisms that could further inform the EEs and CSF of EAM, hence, the dotted lines from the mechanisms back to the "essential elements of an agile EAM" and the " CSFs of EAM" constructs as shown in Figure 18. Indeed, the agile EAMM that has been proposed integrates the various EEs towards supporting the agility of the EAM function (the "Enterprise benefit" construct in Figure 18).

In line with the DSR process model employed, the agile EAMM was demonstrated and evaluated against the objectives of the thesis. First, a demonstration of how the agile EAM could contribute value, in the form of agility, to the EA development process (e.g. TOGAF ADM) was done. The ADM's Preliminary, and phases A and E are collaboratively considered at the big-picture to form common understanding of the to-be architecture and prioritize transition architectures and solutions needed to get there. The implication for each of the EA domain is decided and agreed upon. An architecture endeavor is selected and Phases B to D are done in parallel and iteratively to develop the various EA domain plans necessary to reach the intended goals. The plans are implemented through the Do, Check and Act stages of the agile EAM process to develop the intended architectural artifact. Phases G and H, and Requirements Management are implemented within the structure of the Agile EAMM. Conceptually, deploying the ADM within the agile EAM process makes the ADM more targeted and agile.

Also, an illustrative scenario was used to evaluate the efficacy of the method. The focus of the hypothetical case was to establish alignment between

Business and IT, unite IT and Business teams, and use EA to enable IT to support business agility. The collaborative nature of the agile EAMM was employed to unite and form common understanding between business and IT teams, and to align business and IT; the targeted interventions were employed to develop or reconfigure EA components in response to changing business needs; and the continues improvement nature of the agile EAM process was employed to keep the enterprise in fit with its environment. Indeed, conceptually, the use of the agile EAM method has the propensity to improve the agility of the EAM function within the hypothetical case company, as well as the agility of the hypothetical case company as a whole (the "Enterprise Benefits" construct in Figure 18).

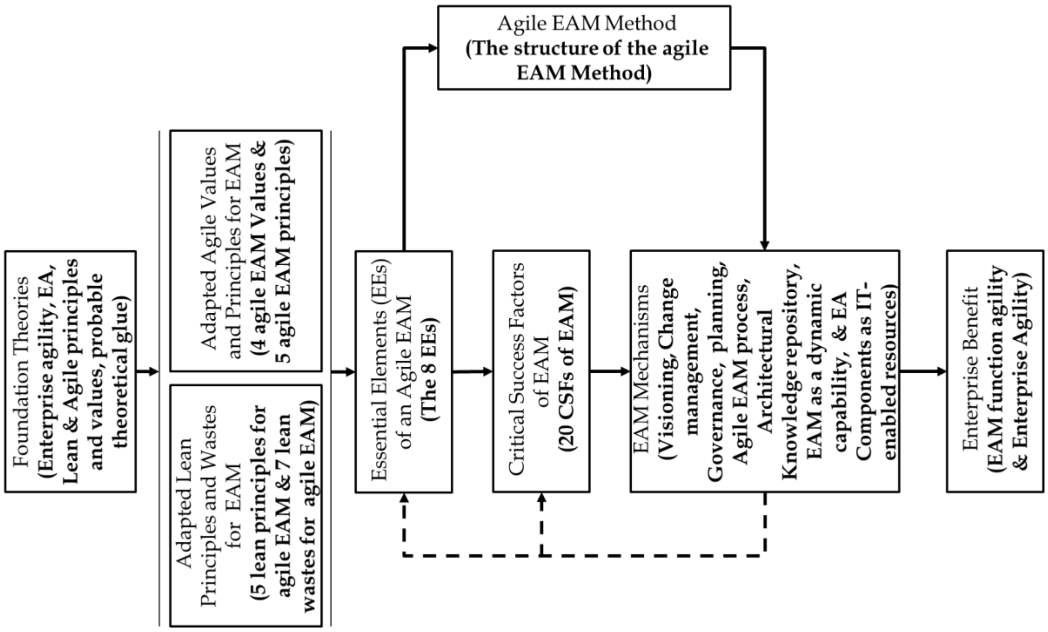


Figure 18: Realized conceptual model of the research

7.2 **Limitations**

There are three main limitations with the thesis; two of them are related to the process, and the third is related to the agile EAMM. Firstly, the research process included an interview with 8 interviewees from Finland. Most of the interviewees were either employees in the public sector, or consultants in the public sector. The interview response therefore lacked perspectives from the arguably more dynamic private sector, especially smaller firms.

Secondly, the agile EAM method has not been demonstrated and evaluated within an enterprise yet. The efficacy of the method therefore remains conceptual at this point.

Thirdly, the agile EAMM assumes that enterprises are willing to collaboratively engage business and IT in the EAM process, and thus overlooks the possible restrictions that could emanate from organizational structures, cultures and politics. Conceptually, the adoption and adaption of the agile EAM method will be more profitable in enterprises that are willing to engage all stakeholders, from business and IT, to form a shared understanding of the architectural needs of the enterprise, and find solutions that will benefit the whole enterprise rather than individual silos at the detriment of enterprise-wide benefits. Nevertheless, other enterprises can adopt and adapt individual mechanisms that fit their organizational situations.

7.3 Further Research

Firstly, the various EEs represent areas for further research. For instance, a further study into how the collaboration between the various EA domains could be enacted and sustained over the long term will add value to the instantiation and utility of the agile EAMM.

Secondly, further evaluation of the proposed EEs in wider context e.g. through online survey will contribute to the justification and enhancement, or rebuttal of some of the proposed mechanisms, and the appropriation of the method during instantiation.

Thirdly, the method, its mechanisms, and guideline for instantiation remain conceptual and lack evaluation within real enterprises. An empirical research, either by action research or DSR (instantiation), would be valuable in evaluating the efficacy of the method. The empirical research or other parallel research could also investigate the enterprise level characteristics of the enterprises in which the agile EAMM worked, and those in which the agile EAMM did not work.

7.4 Conclusions

As enterprises are exposed to continuously changing environments fueled by factors including; changing market requirements, regulations, advancements in technologies, and competitor actions; they are forced to develop capabilities that enable them to sense and respond to these changes in ways that promote their sustainability and competitiveness over time. One of these capabilities is enterprise agility. EA has been identified to possess the ability to drive organizational impact, for example, in the form of enterprise agility. Indeed, EA has the propensity to promote enterprise agility provided it is managed and guided by a methodic approach in a way that strengthens the capabilities of the enterprise that enables it to sense and respond to changes in the business environment. However, prior research efforts at developing the building blocks or components of Enterprise Architecture Management (EAM) do not either consider "agility" as an attribute of the EAM itself or relate the components that promote the agility of the EAM function to the factors that promote the agility of the enterprise as a whole

This thesis aims at filling this gap in literature by developing an enterprise architecture management (EAM) method that enables the agility of the EAM function, and supports the agility of the enterprise as a whole. To do this, the thesis answered three main questions: firstly, what are the factors that influence the agility of the enterprise?; secondly, what essential elements or components should an enterprise architecture management method possess in order to support the agility of the EAM function and the agility of the enterprise as a whole?; and finally, how could these essential elements be conveyed in the form of a method?

The first question was answered via the review of literature mostly on agile manufacturing where the term "agile enterprise" emanated. There are three factors that influence the agility of the enterprise; agility drivers, agility capabilities, and agility providers. The second question was answered by: first, reviewing literature on enterprise agility, enterprise architecture, Lean and Agile principles and values, and a set of proposed theoretical glue to form a theoretical foundation; second, using the theoretical foundation to adapt agile principles, and lean principles for EAM; and third, consolidating the adapted agile and lean principles into the eight Essential Elements of Agile EAM Method. The essential elements were evaluated for relevance through a semi-structured interview with 8 interviewees consisting of practitioners and researchers. The third, question was answered by constructing a method, called the agile EAMM, that coveys these EEs towards supporting the agility of the EAM function and the agility of the enterprise. The agile EAMM was demonstrated to be compatible with TOGAF's ADM, and evaluated using an illustrative scenario. Conceptually, the method has the ability to support the agility of the EAM function and the agility of the enterprise. The thesis succeeded in answering all the three research questions that it aimed at answering, and filled a gap in literature by constructing the EEs that add agility to the EAM function; and the agile EAMM that supports the agility of the EAM function and the agility of the enterprise as a whole.

REFERENCES

Abbas, N., Gravell, A. M., & Wills, G. B. (2008). Historical roots of Agile methods: where did "Agile thinking" come from? In *Agile Processes in Software Engineering and Extreme Programming* (pp. 94–103). Springer.

Abrahamsson, P., Conboy, K., & Wang, X. (2009). �Lots done, more to do�: the current state of agile systems development research.

Ahlemann, F., Stettiner, E., Messerschmidt, M., & Legner, C. (2012). *Strategic enterprise architecture management: challenges, best practices, and future developments*. Springer Science & Business Media.

Aier, S. (2014). The role of organizational culture for grounding, management, guidance and effectiveness of enterprise architecture principles. *Information Systems and E-Business Management*, *12*(1), 43–70.

Aier, S., Fischer, C., & Winter, R. (2011). Construction and evaluation of a meta-model for enterprise architecture design principles.

Aier, S., Gleichauf, B., & Winter, R. (2011). Understanding Enterprise Architecture Management Design-An Empirical Analysis. In *Wirtschaftsinformatik* (p. 50).

Aier, S., & Schelp, J. (2010). A reassessment of enterprise architecture implementation. In *Service-Oriented Computing. ICSOC/ServiceWave 2009 Workshops* (pp. 35–47). Springer.

Aladwani, A. M. (2001). Change management strategies for successful ERP implementation. *Business Process Management Journal*, *7*(3), 266–275.

Alliance, A. (2001). Agile manifesto. *Online at Http://www. Agilemanifesto. Org*, *6*(6.1).

Amabile, T. M., & Kramer, S. J. (2011). The power of small wins. *Harvard Business Review*, *89*(5), 70–80.

Ambrosini, V., Bowman, C., & Collier, N. (2009). Dynamic capabilities: an exploration of how firms renew their resource base. *British Journal of Management*, *20*(s1), S9–S24.

Anand, G., Ward, P. T., Tatikonda, M. V., & Schilling, D. A. (2009). Dynamic capabilities through continuous improvement infrastructure. *Journal of Operations Management*, *27*(6), 444–461.

Armour, F. J., & Kaisler, S. H. (2001). Enterprise architecture: Agile transition and implementation. *IT Professional*, *3*(6), 30–37.

Armour, F. J., Kaisler, S. H., & Liu, S. Y. (1999a). A big-picture look at enterprise architectures. *IT Professional*, *1*(1), 35–42.

Armour, F. J., Kaisler, S. H., & Liu, S. Y. (1999b). Building an enterprise architecture step by step. *IT Professional*, *1*(4), 31–39.

Aziz, S., Obitz, T., Modi, R., & Sarkar, S. (2005). Enterprise Architecture: A Governance Framework. *Part I: Embedding Architecture into the Organization. InfoSyS Technologies Ltd*.

Bagozzi, R. P., Yi, Y., & Phillips, L. W. (1991). Assessing construct validity in organizational research. *Administrative Science Quarterly*, 421–458.

Ball, M. O., Chen, C.-Y., Hoffman, R., & Vossen, T. (2001). Collaborative decision making in air traffic management: Current and future research directions. In *New Concepts and Methods in Air Traffic Management* (pp. 17–30). Springer.

Barki, H., & Hartwick, J. (1989). Rethinking the concept of user involvement. *MIS Quarterly*, 53–63.

Barnett, W., Presley, A., Johnson, M., & Liles, D. (1994). An architecture for the virtual enterprise. In *Systems, Man, and Cybernetics, 1994. Humans, Information and Technology., 1994 IEEE International Conference on* (Vol. 1, pp. 506–511). IEEE.

Baroudi, J. J., Olson, M. H., & Ives, B. (1986). An empirical study of the impact of user involvement on system usage and information satisfaction. *Communications of the ACM*, *29*(3), 232–238.

Beer, M., Voelpel, S. C., Leibold, M., & Tekie, E. B. (2005). Strategic management as organizational learning: Developing fit and alignment through a disciplined process. *Long Range Planning*, *38*(5), 445–465.

Bente, S., Bombosch, U., & Langade, S. (2012). *Collaborative Enterprise Architecture: Enriching EA with lean, agile, and enterprise 2.0 practices*. Newnes.

Bessant, J., Caffyn, S., & Gallagher, M. (2001). An evolutionary model of continuous improvement behaviour. *Technovation*, *21*(2), 67–77.

Bessant, J., Caffyn, S., Gilbert, J., Harding, R., & Webb, S. (1994). Rediscovering continuous improvement. *Technovation*, *14*(1), 17–29.

Bessant, J., & Francis, D. (1999). Developing strategic continuous improvement capability. *International Journal of Operations & Production Management*, *19*(11), 1106–1119.

Bhasin, S., & Burcher, P. (2006). Lean viewed as a philosophy. *Journal of Manufacturing Technology Management*, *17*(1), 56–72.

Bjögvinsson, E., Ehn, P., & Hillgren, P.-A. (2012). Design things and design thinking: Contemporary participatory design challenges. *Design Issues*, *28*(3), 101–116.

Boehm, B. (2002). Get ready for agile methods, with care. *Computer*, *35*(1), 64–69.

Boster, M., Liu, S., & Thomas, R. (2000). Getting the most from your enterprise architecture. *IT Professional*, *2*(4), 43–51.

Bradley, R. V., Pratt, R. M., Byrd, T. A., Outlay, C. N., & Wynn Jr, D. E. (2012). Enterprise architecture, IT effectiveness and the mediating role of IT alignment in US hospitals. *Information Systems Journal*, *22*(2), 97–127.

Broadbent, M., & Kitzis, E. S. (2005). The new CIO leader. *Harvard Business School Press, Boston, Massachusetts*.

Bryman, A. (2012). *Social research methods*. Oxford university press.

Buchanan, R. D., & Soley, R. M. (2002). Aligning enterprise architecture and IT investments with corporate goals. *OMG Whitepaper, Object Management Group, Needham*.

Buckl, S., Dierl, T., Matthes, F., & Schweda, C. M. (2010). Building blocks for enterprise architecture management solutions. In *Practice-driven research on enterprise transformation* (pp. 17–46). Springer.

Buckl, S., Matthes, F., Monahov, I., Roth, S., Schulz, C., & Schweda, C. M. (2011). Towards an agile design of the enterprise architecture management function. In *Enterprise Distributed Object Computing Conference Workshops (EDOCW), 2011 15th IEEE International* (pp. 322–329). IEEE.

Buckl, S., Matthes, F., & Schweda, C. M. (2009). A viable system perspective on enterprise architecture management. In *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on* (pp. 1483–1488). IEEE.

Buckl, S., Schweda, C. M., & Matthes, F. (2010). A design theory nexus for situational enterprise architecture management. In *Enterprise Distributed Object Computing Conference Workshops (EDOCW), 2010 14th IEEE International* (pp. 3–8). IEEE.

Byrd, T. A., & Turner, D. E. (2000). Measuring the flexibility of information technology infrastructure: Exploratory analysis of a construct. *Journal of Management Information Systems*, *17*(1), 167–208.

Caffyn, S. (1999). Development of a continuous improvement self-assessment tool. *International Journal of Operations & Production Management*, *19*(11), 1138–1153.

Carl, J. W., & Colombi, J. M. (2007). *Seven Secret Tips to Build Intelligent Enterprise Architectures*. DTIC Document.

Carroll, J. M., & Rosson, M. B. (2007). Participatory design in community informatics. *Design Studies*, *28*(3), 243–261.

Chen, D. Q., Mocker, M., Preston, D. S., & Teubner, A. (2010). Information systems strategy: reconceptualization, measurement, and implications. *MIS Quarterly*, *34*(2), 233–259.

Cho, C.-S., & Gibson Jr, G. E. (2001). Building project scope definition using project definition rating index. *Journal of Architectural Engineering*, *7*(4), 115–125.

Cho, H., Jung, M., & Kim, M. (1996). Enabling technologies of agile manufacturing and its related activities in Korea. *Computers & Industrial Engineering*, *30*(3), 323–334.

Chuang, C.-H., & van Loggerenberg, J. (2010). Challenges facing enterprise architects: A south african perspective. In *System Sciences (HICSS), 2010 43rd Hawaii International Conference on* (pp. 1–10). IEEE.

Cockburn, A., & Highsmith, J. (2001). Agile software development: The people factor. *Computer*, (11), 131–133.

Collis, D. J. (1994). Research note: how valuable are organizational capabilities? *Strategic Management Journal*, *15*(S1), 143–152.

Committee, I. S. O. E., & others. (2007). *ISO/IEC 42010: 2007-Systems and software engineering–Recommended practice for architectural description of software-intensive systems*. Technical report, ISO.

Conboy, K. (2009). Agility from first principles: reconstructing the concept of agility in information systems development. *Information Systems Research*, *20*(3), 329–354.

Davenport, T. H., & Short, J. E. (1990). The New Industrial Engineering: Information Technology And Business Process Redesign. *MIT Sloan Management Review*, *31*(4), 11.

Delone, W. H., & McLean, E. R. (2003). The DeLone and McLean model of information systems success: a ten-year update. *Journal of Management Information Systems*, *19*(4), 9–30.

Drnevich, P. L., & Kriauciunas, A. P. (2011). Clarifying the conditions and limits of the contributions of ordinary and dynamic capabilities to relative firm performance. *Strategic Management Journal*, *32*(3), 254–279.

Ehn, P. (2008). Participation in design things. In *Proceedings of the Tenth Anniversary Conference on Participatory Design 2008* (pp. 92–101). Indiana University.

Engelsman, W., Quartel, D., Jonkers, H., & van Sinderen, M. (2011). Extending enterprise architecture modelling with business goals and requirements. *Enterprise Information Systems*, *5*(1), 9–36.

Espinosa, J. A., Boh, W. F., & DeLone, W. (2011). The organizational impact of enterprise architecture: a research framework. In *System Sciences (HICSS), 2011 44th Hawaii International Conference on* (pp. 1–10). IEEE.

Farwick, M., Breu, R., Hauder, M., Roth, S., & Matthes, F. (2013). Enterprise architecture documentation: Empirical analysis of information sources for automation. In *System Sciences (HICSS), 2013 46th Hawaii International Conference on* (pp. 3868–3877). IEEE.

Ferns, D. C. (1991). Developments in programme management. *International Journal of Project Management*, *9*(3), 148–156.

Finney, S., & Corbett, M. (2007). ERP implementation: a compilation and analysis of critical success factors. *Business Process Management Journal*, *13*(3), 329–347.

Fischer, C., Winter, R., & Aier, S. (2010). What Is an Enterprise Architecture Principle? In *Computer and Information Science 2010* (pp. 193–205). Springer.

Fischer, R., Aier, S., & Winter, R. (2007). A Federated Approach to Enterprise Architecture Model Maintenance. *Enterprise Modelling and Information Systems Architectures*, *2*(2), 14–22.

Foorthuis, R., van Steenbergen, M., Brinkkemper, S., & Bruls, W. A. (2015). A theory building study of enterprise architecture practices and benefits. *Information Systems Frontiers*, 1–24.

Foorthuis, R., van Steenbergen, M., Mushkudiani, N., Bruls, W., Brinkkemper, S., & Bos, R. (2010). On Course, but not There Yet: Enterprise Architecture Conformance and Benefits in Systems Development. In *ICIS* (p. 110). Citeseer.

Gregor, S., & Hevner, A. R. (2013). Positioning and presenting design science research for maximum impact. *MIS Quarterly*, *37*(2), 337–356.

Hartwick, J., & Barki, H. (1994). Explaining the role of user participation in information system use. *Management Science*, *40*(4), 440–465.

Helfat, C. E., & Winter, S. G. (2011). Untangling dynamic and operational capabilities: Strategy for the (N) ever-changing world. *Strategic Management Journal*, *32*(11), 1243–1250.

Henderson, J. C., & Venkatraman, N. (1993). Strategic alignment: Leveraging information technology for transforming organizations. *IBM Systems Journal*, *32*(1), 4–16.

Hervani, A. A., Helms, M. M., & Sarkis, J. (2005). Performance measurement for green supply chain management. *Benchmarking: An International Journal*, *12*(4), 330–353.

Hevner, von A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, *28*(1), 75–105.

Hines, P., Holweg, M., & Rich, N. (2004). Learning to evolve: a review of contemporary lean thinking. *International Journal of Operations & Production Management*, *24*(10), 994–1011.

Hirvonen, A. P., & Pulkkinen, M. (2004). A practical approach to EA planning and development: the EA management grid. In *7th International Conference on Business Information Systems*. Citeseer.

Hong, K.-K., & Kim, Y.-G. (2002). The critical success factors for ERP implementation: an organizational fit perspective. *Information & Management*, *40*(1), 25–40.

Hoogervorst, J. (2004). Enterprise architecture: Enabling integration, agility and change. *International Journal of Cooperative Information Systems*, *13*(03), 213–233.

Iivari, J. (1992). The organizational fit of information systems. *Inf. Syst. J.*, *2*(1), 3–29.

Iivari, J. (2007). A paradigmatic analysis of information systems as a design science. *Scandinavian Journal of Information Systems*, *19*(2), 5.

Iivari, J., & Venable, J. (2009). Action research and design science research–seemingly similar but decisively dissimilar. In *17th European Conference on Information Systems* (pp. 1–13).

Innes, J. E. (2004). Consensus building: Clarifications for the critics. *Planning Theory*, *3*(1), 5–20.

Innes, J. E., & Booher, D. E. (1999). Consensus building and complex adaptive systems: A framework for evaluating collaborative planning. *Journal of the American Planning Association*, *65*(4), 412–423.

ISO/IEC 42010:2007, I. (2007). ISO/IEC 42010: Systems and Software Engineering-Recommended Practice for Architectural Description of Software-Intensive Systems. *International Organization for Standardization, Geneva, Switzerland*.

Iyer, B., & Gottlieb, R. (2004). The Four-Domain Architecture: An approach to support enterprise architecture design. *IBM Systems Journal*, *43*(3), 587–597.

Jarke, M., & Pohl, K. (1994). Requirements engineering in 2001:(virtually) managing a changing reality. *Software Engineering Journal*, *9*(6), 257–266.

Jonkers, H., Lankhorst, M. M., ter Doest, H. W., Arbab, F., Bosma, H., & Wieringa, R. J. (2006). Enterprise architecture: Management tool and blueprint for the organisation. *Information Systems Frontiers*, *8*(2), 63–66.

Jonkers, H., van Burren, R., Arbab, F., De Boer, F., Bonsangue, M., Bosma, H., … others. (2003). Towards a language for coherent enterprise architecture descriptions. In *Enterprise Distributed Object Computing Conference, 2003. Proceedings. Seventh IEEE International* (pp. 28–37). IEEE.

Jugel, D., & Schweda, C. M. (2014). Interactive functions of a cockpit for enterprise architecture planning. In *Enterprise Distributed Object Computing Conference Workshops and Demonstrations (EDOCW), 2014 IEEE 18th International* (pp. 33–40). IEEE.

Kähkönen, T. (2004). Agile methods for large organizations-building communities of practice. In *Agile Development Conference, 2004* (pp. 2–10). IEEE.

Kaisler, S. H., Armour, F., & Valivullah, M. (2005). Enterprise architecting: Critical problems. In *System Sciences, 2005. HICSS'05. Proceedings of the 38th Annual Hawaii International Conference on* (p. 224b–224b). IEEE.

Kaiya, H., Shinbara, D., Kawano, J., & Saeki, M. (2005). Improving the detection of requirements discordances among stakeholders. *Requirements Engineering*, *10*(4), 289–303.

Kamogawa, T., & Okada, H. (2005). A framework for enterprise architecture effectiveness. In *Services Systems and Services Management, 2005. Proceedings of ICSSSM'05. 2005 International Conference on* (Vol. 1, pp. 740–745). IEEE.

Kane, G. C., & Alavi, M. (2007). Information technology and organizational learning: An investigation of exploration and exploitation processes. *Organization Science*, *18*(5), 796–812.

Kanellis, P., Lycett, M., & Paul, R. (1999). Evaluating business information systems fit: from concept to practical application. *European Journal of Information Systems*, *8*(1), 65–76.

Keil, M., Cule, P. E., Lyytinen, K., & Schmidt, R. C. (1998). A framework for identifying software project risks. *Communications of the ACM*, *41*(11), 76–83.

Kilpatrick, J. (2003). Lean principles. *Utah Manufacturing Extension Partnership*, 1–5.

Kluge, C., Dietzsch, A., & Rosemann, M. (2006). How to realise corporate value from enterprise architecture. In *ECIS* (pp. 1572–1581).

Lange, M., Mendling, J., & Recker, J. (2015). An empirical analysis of the factors and measures of Enterprise Architecture Management success. *European Journal of Information Systems*.

Lankhorst, M. M. (2004). Enterprise architecture modelling—the issue of integration. *Advanced Engineering Informatics*, *18*(4), 205–216.

Lapalme, J. (2012). Three schools of thought on enterprise architecture. *IT Professional*, (6), 37–43.

LeBreton, J. M., & Senter, J. L. (2007). Answers to 20 questions about interrater reliability and interrater agreement. *Organizational Research Methods*.

Lee, G., & Xia, W. (2010). Toward agile: an integrated analysis of quantitative and qualitative field data. *Management Information Systems Quarterly*, *34*(1), 7.

Leffingwell, D., Yakyma, A., Jemilo, D., Knaster, R., Shalloway, A., & Oren, I. (2011). *Scaled Agile Framework*. (Windows Phone Central, Ed.). Retrieved from http://www.scaledagileframework.com/

Leist, S., & Zellner, G. (2006). Evaluation of current architecture frameworks. In *Proceedings of the 2006 ACM symposium on Applied computing* (pp. 1546–1553). ACM.

Lim, N., Lee, T., & Park, S. (2009). A comparative analysis of enterprise architecture frameworks based on EA quality attributes. In *Software Engineering, Artificial Intelligences, Networking and Parallel/Distributed Computing, 2009. SNPD'09. 10th ACIS International Conference on* (pp. 283–288). IEEE.

Lindström, \AAsa. (2006). On the syntax and semantics of architectural principles. In *System Sciences, 2006. HICSS'06. Proceedings of the 39th Annual Hawaii International Conference on* (Vol. 8, p. 178b–178b). IEEE.

Lucke, C., Bürger, M., Diefenbach, T., Freter, J., & Lechner, U. (2012). Categories of Enterprise Architecting Issues-An Empirical Investigation based on Expert Interviews. *DC Mattfeld & S. Robra-Bissantz(Eds.), Multikonferenz Wirtschaftsinformatik*, 999–1010.

Lucke, C., Krell, S., & Lechner, U. (2010). Critical issues in enterprise architecting–a literature review.

Luck, R. (2003). Dialogue in participatory design. *Design Studies*, *24*(6), 523–535.

Lu, Y., & Ramamurthy, K. (2011). Understanding the link between information technology capability and organizational agility: An empirical examination. *Mis Quarterly*, *35*(4), 931–954.

Lycett, M., Rassau, A., & Danson, J. (2004). Programme management: a critical review. *International Journal of Project Management*, *22*(4), 289–299.

Lyytinen, K., & Rose, G. M. (2006). Information system development agility as organizational learning. *European Journal of Information Systems*, *15*(2), 183–199.

March, J. G. (1991). Exploration and exploitation in organizational learning. *Organization Science*, *2*(1), 71–87.

March, S. T., & Smith, G. F. (1995). Design and natural science research on information technology. *Decision Support Systems*, *15*(4), 251–266.

Markus, M. L., & Robey, D. (1983). The organizational validity of management information systems. *Human Relations*, *36*(3), 203–225.

Mathiassen, L., Saarinen, T., Tuunanen, T., & Rossi, M. (2007). A contingency model for requirements development. *Journal of the Association for Information Systems*, *8*(11), 569–597.

McKeen, J. D., Guimaraes, T., & Wetherbe, J. C. (1994). The relationship between user participation and user satisfaction: an investigation of four contingency factors. *Mis Quarterly*, 427–451.

Miles, R. E. (1984). Fit, failure, the hall of fame. *California Management Review (pre-1986)*, *26*(000003), 10.

Moen, R., & Norman, C. (2006). *Evolution of the PDCA cycle*.

Myers, M. D., & Newman, M. (2007). The qualitative interview in IS research: Examining the craft. *Information and Organization*, *17*(1), 2–26.

Nabukenya, J., Bommel, P. van, & Proper, H. (2006). Collaborative policy–making processes.

Närman, P., Johnson, P., & Nordström, L. (2007). Enterprise architecture: A framework supporting system quality analysis. In *Enterprise Distributed Object Computing Conference, 2007. EDOC 2007. 11th IEEE International* (pp. 130–130). IEEE.

Naylor, J. B., Naim, M. M., & Berry, D. (1999). Leagility: integrating the lean and agile manufacturing paradigms in the total supply chain. *International Journal of Production Economics*, *62*(1), 107–118.

Nazir, S., & Pinsonneault, A. (2012). IT and firm agility: an electronic integration perspective. *Journal of the Association for Information Systems*, *13*(3), 150–171.

Nevo, S., & Wade, M. (2011). Firm-level benefits of IT-enabled resources: A conceptual extension and an empirical assessment. *The Journal of Strategic Information Systems*, *20*(4), 403–418.

Nevo, S., & Wade, M. R. (2010). The formation and value of it-enabled resources: Antecedents and consequences. *Management Information Systems Quarterly*, *34*(1), 10.

Niemi, E. (2008). Enterprise architecture benefits: Perceptions from literature and practice. *Evaluation of Enterprise and Software Architectures: Critical Issues, Metrics and practices:[AISA Project 2005-2008]/Eetu Niemi, Tanja Ylimäki & Niina Hämäläinen (eds.). Jyväskylä: University of Jyväskylä, Information Technology Research Institute, 2008.-(Tietotekniikan Tutkimusinstituutin Julkaisuja, ISSN 1236-1615; 18). ISBN 978-951-39-3108-7 (CD-ROM).*

Nikpay, F., Selamat, H., Rouhani, B. D., & Nikfard, P. (2013). A Review of Critical Success Factors of Enterprise Architecture Implementation. In *Informatics and Creative Multimedia (ICICM), 2013 International Conference on* (pp. 38–42). IEEE.

Nissen, M. E. (2014). Organization design for dynamic fit: A review and projection. *Journal of Organization Design*, *3*(2).

Okoli, C., & Schabram, K. (2010). A guide to conducting a systematic literature review of information systems research. *Available at SSRN 1954824*.

Op't Land, M., & Proper, H. (2007). Impact of principles on enterprise engineering. In *Relevant rigour–Rigorous relevance: Proceedings: 15th European Conference on Information Systems (ECIS 2007), St. Gallen, Switzerland, June 2007*. University of St. Gallen: St. Gallen, Switzerland.

Overby, E., Bharadwaj, A., & Sambamurthy, V. (2005). A framework for enterprise agility and the enabling role of digital options. In *Business Agility and Information Technology Diffusion* (pp. 295–312). Springer.

Pandza, K., Horsburgh, S., Gorton, K., & Polajnar, A. (2003). A real options approach to managing resources and capabilities. *International Journal of Operations & Production Management*, *23*(9), 1010–1032.

Pavlou, P. A., & El Sawy, O. A. (2011). Understanding the elusive black box of dynamic capabilities. *Decision Sciences*, *42*(1), 239–273.

Peffers, K., Gengler, C. E., & Tuunanen, T. (2003). Extending critical success factors methodology to facilitate broadly participative information systems planning. *Journal of Management Information Systems*, *20*(1), 51–85.

Peffers, K., Rothenberger, M., Tuunanen, T., & Vaezi, R. (2012). Design science research evaluation. In *Design Science Research in Information Systems. Advances in Theory and Practice* (pp. 398–410). Springer.

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, *24*(3), 45–77.

Peterson, R. (2004). Crafting information technology governance. *Information Systems Management*, *21*(4), 7–22.

Politi, M. C., & Street, R. L. (2011). The importance of communication in collaborative decision making: facilitating shared mind and the management of uncertainty. *Journal of Evaluation in Clinical Practice*, *17*(4), 579–584.

Poppendieck, M. (2011). Principles of lean thinking. *IT Management Select*, *18*.

Poppendieck, M., & Poppendieck, T. (2003). *Lean software development: an agile toolkit*. Addison-Wesley Professional.

Pulkkinen, M. (2006). Systemic management of architectural decisions in enterprise architecture planning. four dimensions and three abstraction levels. In *System Sciences, 2006. HICSS'06. Proceedings of the 39th Annual Hawaii International Conference on* (Vol. 8, p. 179a–179a). IEEE.

Pulkkinen, M., & Hirvonen, A. (2005). Ea planning, development and management process for agile enterprise development. In *System Sciences, 2005. HICSS'05. Proceedings of the 38th Annual Hawaii International Conference on* (p. 223c–223c). IEEE.

Putnik, G. D., Browaeys, M.-J., & Fisser, S. (2012). Lean and agile: an epistemological reflection. *The Learning Organization*, *19*(3), 207–218.

Quinn, R. D., Causey, G. C., Merat, F. L., Sargent, D. M., Barendt, N. A., Newman, W. S., … others. (1996). Design of an agile manufacturing workcell for light mechanical applications. In *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on* (Vol. 1, pp. 858–863). IEEE.

Rapoport, R. N. (1970). Three dilemmas in action research with special reference to the Tavistock experience. *Human Relations*, *23*(6), 499–513.

Redström, J. (2008). RE: Definitions of use. *Design Studies*, *29*(4), 410–423.

Richardson, G. L., Jackson, B. M., & Dickson, G. W. (1990). A principles-based enterprise architecture: Lessons from Texaco and Star Enterprise. *MIS Quarterly*, 385–403.

Robey, D., Ross, J. W., & Boudreau, M.-C. (2002). Learning to implement enterprise systems: An exploratory study of the dialectics of change. *Journal of Management Information Systems*, *19*(1), 17–46.

Rood, M. A. (1994). Enterprise architecture: definition, content, and utility. In *Enabling Technologies: Infrastructure for Collaborative Enterprises, 1994. Proceedings., Third Workshop on* (pp. 106–111). IEEE.

Roth, S., Hauder, M., Farwick, M., Breu, R., & Matthes, F. (2013). Enterprise Architecture Documentation: Current Practices and Future Directions. In *Wirtschaftsinformatik* (p. 58).

Rouhani, B. D., Shirazi, H., Nezhad, A. F., & Kharazmi, S. (2008). Presenting a framework for agile enterprise architecture. In *Information Technology, 2008. IT 2008. 1st International Conference on* (pp. 1–4). IEEE.

Saat, J., Aier, S., & Gleichauf, B. (2009). Assessing the complexity of dynamics in enterprise architecture planning-lessons from chaos theory. *AMCIS 2009 Proceedings*, 808.

Sambamurthy, V., Bharadwaj, A., & Grover, V. (2003a). Shaping agility through digital options: Reconceptualizing the role of information technology in contemporary firms. *MIS Quarterly*, 237–263.

Sambamurthy, V., Bharadwaj, A., & Grover, V. (2003b). Shaping agility through digital options: Reconceptualizing the role of information technology in contemporary firms. *MIS Quarterly*, 237–263.

Sambamurthy, V., Wei, K.-K., Lim, K., & Lee, D. (2007). IT-enabled organizational agility and firms' sustainable competitive advantage. *ICIS 2007 Proceedings*, 91.

Schmidt, C., & Buxmann, P. (2011). Outcomes and success factors of enterprise IT architecture management: empirical insight from the international financial services industry. *European Journal of Information Systems*, *20*(2), 168–185.

Schultze, U., & Avital, M. (2011). Designing interviews to generate rich data for information systems research. *Information and Organization*, *21*(1), 1–16.

Seddon, P. B. (1997). A respecification and extension of the DeLone and McLean model of IS success. *Information Systems Research*, *8*(3), 240–253.

Sein, M., Henfridsson, O., Purao, S., Rossi, M., & Lindgren, R. (2011). Action design research.

Seo, D., & La Paz, A. I. (2008). Exploring the dark side of IS in achieving organizational agility. *Communications of the ACM*, *51*(11), 136–139.

Seppänen, V. (2014). From problems to critical success factors of enterprise architecture adoption.

Sessions, R. (2007, May). A Comparison of the Top Four Enterprise Architecture Methodologies. Retrieved from https://msdn.microsoft.com/en-us/library/bb466232.aspx

Sharifi, H., & Zhang, Z. (1999). A methodology for achieving agility in manufacturing organisations: An introduction. *International Journal of Production Economics*, *62*(1), 7–22.

Sherehiy, B., Karwowski, W., & Layer, J. K. (2007). A review of enterprise agility: Concepts, frameworks, and attributes. *International Journal of Industrial Ergonomics*, *37*(5), 445–460.

Simon, D., Fischbach, K., & Schoder, D. (2013). An exploration of enterprise architecture research. *Communications of the Association for Information Systems*, *32*(1), 1.

Simon, D., Fischbach, K., & Schoder, D. (2014). Enterprise architecture management and its role in corporate strategic management. *Information Systems and E-Business Management*, *12*(1), 5–42.

Sokovic, M., Pavletic, D., & Pipan, K. K. (2010). Quality improvement methodologies–PDCA cycle, RADAR matrix, DMAIC and DFSS. *Journal of Achievements in Materials and Manufacturing Engineering*, *43*(1), 476–483.

Spinuzzi, C. (2005). The methodology of participatory design. *Technical Communication*, *52*(2), 163–174.

Stelzer, D. (2010). Enterprise architecture principles: literature review and research directions. In *Service-Oriented Computing. ICSOC/ServiceWave 2009 Workshops* (pp. 12–21). Springer.

Strong, D. M., & Volkoff, O. (2010). Understanding organization-enterprise system fit: a path to theorizing the information technology artifact. *MIS Quarterly*, *34*(4), 731–756.

Tallon, P. P., & Pinsonneault, A. (2011). Competing perspectives on the link between strategic information technology alignment and organizational agility: insights from a mediation model. *Mis Quarterly*, *35*(2), 463–486.

Tamm, T., Seddon, P. B., Shanks, G., & Reynolds, P. (2011). How does enterprise architecture add value to organisations. *Communications of the Association for Information Systems*, *28*(1), 141–168.

Tang, A., Han, J., & Chen, P. (2004). A comparative analysis of architecture frameworks. In *Software Engineering Conference, 2004. 11th Asia-Pacific* (pp. 640–647). IEEE.

Teece, D. J., Pisano, G., & Shuen, A. (1997). Dynamic capabilities and strategic management.

Teo, T., & King, W. (1999). An empirical study of the impacts of integrating business planning and information systems planning. *European Journal of Information Systems*, *8*(3), 200–210.

TOGAF, V. 9. (2009). 9, The Open Group Architecture Framework (TOGAF). *The Open Group*, *1*.

Tseng, Y.-H., & Lin, C.-T. (2011). Enhancing enterprise agility by deploying agile drivers, capabilities and providers. *Information Sciences*, *181*(17), 3693–3708.

Urbaczewski, L., & Mrdalj, S. (2006). A comparison of enterprise architecture frameworks. *Issues in Information Systems*, *7*(2), 18–23.

Van der Raadt, B., Bonnet, M., Schouten, S., & Van Vliet, H. (2010). The relation between EA effectiveness and stakeholder satisfaction. *Journal of Systems and Software*, *83*(10), 1954–1969.

Van Hoek, R. I. (2000). The thesis of leagility revisited. *International Journal of Agile Management Systems*, *2*(3), 196–201.

Varva, T. (1997). *Improving your measurement of Customer Satisfaction*. ASQ Quality Press, Milwaukee, WI.

Vázquez-Bustelo, D., Avella, L., & Fernández, E. (2007). Agility drivers, enablers and outcomes: empirical test of an integrated agile manufacturing model. *International Journal of Operations & Production Management*, *27*(12), 1303–1332.

Verona, G., & Ravasi, D. (2003). Unbundling dynamic capabilities: an exploratory study of continuous product innovation. *Industrial and Corporate Change*, *12*(3), 577–606.

Vink, P., Imada, A., & Zink, K. (2008). Defining stakeholder involvement in participatory design processes. *Applied Ergonomics*, *39*(4), 519–526.

Von Alan, R. H., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, *28*(1), 75–105.

Wagter, R., Van Den Berg, M., Luijpers, J., & Van Steenbergen, M. (2005). *Dynamic enterprise architecture: how to make it work*. John Wiley & Sons.

Wan, H., Luo, X., Johansson, B., & Chen, H. (2013). Enterprise architecture benefits: the divergence between its desirability and realizability. In *14th International Conference on Informatics and Semiotics in Organizations (ICISO2013, IFIP WG 8, 1 Working Conference)*. SciTePress.

Whetten, D. A. (1989). What constitutes a theoretical contribution? *Academy of Management Review*, *14*(4), 490–495.

Winter, R., & Aier, S. (2011). How are Enterprise Architecture Design Principles Used? In *Enterprise Distributed Object Computing Conference Workshops (EDOCW), 2011 15th IEEE International* (pp. 314–321). IEEE.

Winter, R., & Fischer, R. (2006). Essential layers, artifacts, and dependencies of enterprise architecture. In *Enterprise Distributed Object Computing Conference Workshops, 2006. EDOCW'06. 10th IEEE International* (pp. 30–30). IEEE.

Winter, R., & Schelp, J. (2008). Enterprise architecture governance: the need for a business-to-IT approach. In *Proceedings of the 2008 ACM symposium on Applied computing* (pp. 548–552). ACM.

Winter, S. G. (2003). Understanding dynamic capabilities. *Strategic Management Journal*, *24*(10), 991–995.

Wu, C. W., & Chen, C. L. (2006). An integrated structural model toward successful continuous improvement activity. *Technovation*, *26*(5), 697–707.

Yang, S., & Seddon, P. (2004). Benefits and key project success factors from enterprise systems implementations: lessons from Sapphire 2003. *ACIS 2004 Proceedings*, 27.

Ylimäki, T. (2008a). Potential critical success factors for enterprise architecture. *Evaluation of Enterprise and Software Architectures: Critical Issues, Metrics and practices:[AISA Project 2005-2008]/Eetu Niemi, Tanja Ylimäki & Niina Hämäläinen (eds.). Jyväskylä: University of Jyväskylä, Information Technology Research Institute, 2008.-(Tietotekniikan Tutkimusinstituutin Julkaisuja, ISSN 1236-1615; 18). ISBN 978-951-39-3108-7 (CD-ROM).*

Ylimäki, T. (2008b). Towards a Generic Evaluation Model for Enterprise Architecture. *Evaluation of Enterprise and Software Architectures: Critical Issues, Metrics and practices/Eetu Niemi, Tanja Ylimäki & Niina Hämäläinen (eds.). Jyväskylä: University of Jyväskylä, Information Technology Research*

Institute, 2008.-(Tietotekniikan Tutkimusinstituutin Julkaisuja, ISSN 1236-1615; 18). ISBN 978-951-39-3108-7 (CD-ROM).

Yusuf, Y. Y., Sarhadi, M., & Gunasekaran, A. (1999). Agile manufacturing:: The drivers, concepts and attributes. *International Journal of Production Economics*, *62*(1), 33–43.

Zachman, J. A. (1996). Concepts of the framework for enterprise architecture. *Los Angels, CA*.

Zahra, S. A., Sapienza, H. J., & Davidsson, P. (2006). Entrepreneurship and dynamic capabilities: a review, model and research agenda*. *Journal of Management Studies*, *43*(4), 917–955.

Zhang, Z., & Sharifi, H. (2000). A methodology for achieving agility in manufacturing organisations. *International Journal of Operations & Production Management*, *20*(4), 496–513.

Zhang, Z., & Sharifi, H. (2000). A methodology for achieving agility in manufacturing organisations. *International Journal of Operations & Production Management*, *20*(4), 496–513.

Zhang, Z., & Sharifi, H. (2007). Towards theory building in agile manufacturing strategy�a taxonomical approach. *Engineering Management, IEEE Transactions on*, *54*(2), 351–370.

Zhong Liu, A., & Seddon, P. B. (2009). Understanding how project critical success factors affect organizational benefits from enterprise systems. *Business Process Management Journal*, *15*(5), 716–743.

Zollo, M., & Winter, S. G. (2002). Deliberate learning and the evolution of dynamic capabilities. *Organization Science*, *13*(3), 339–351.

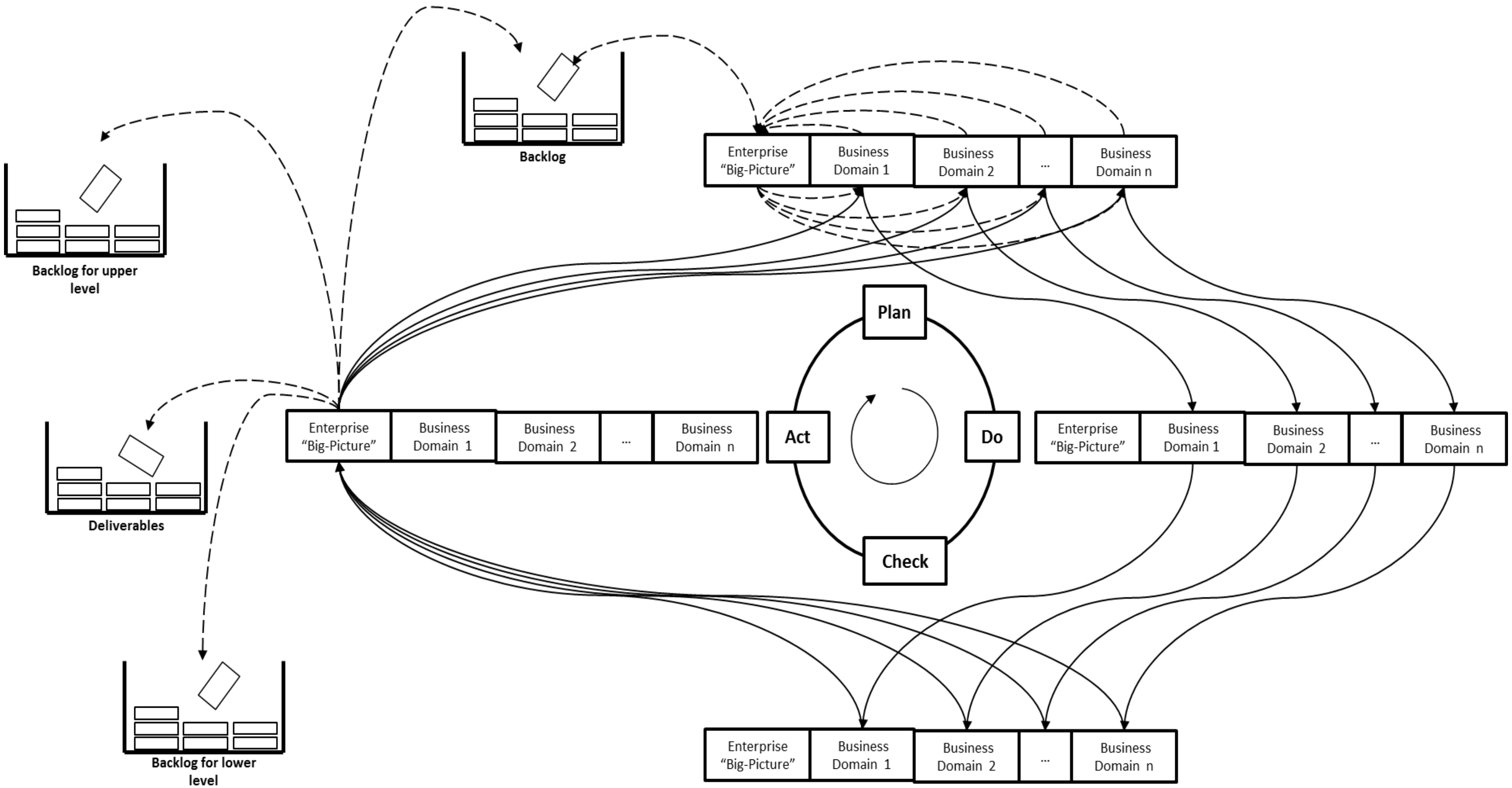
APPENDIX 1 SUMMARY OF QUANTITATIVE RESULTS FROM INTERVIEW

	Essential Elements	Int.A	Int.B	Int.C	Int.D	Int.E	Int.F	Int.G	Int.H	Mean	SD
EE1	**Mechanism for to develop and refine EA vision, goals, purpose, principles & Scope**	**4.00**	**4.00**	**4.50**	**4.50**	**4.50**	**4.50**	**4.50**	**4.00**	**4.31**	0.26
EE1a	Guide development & refinement of EA vision & goals/strategies	5	4	5	5	5	5	5	4	4.75	0.46
EE1b	Guide development & refinement of EA purpose & scope of specific EA intervention	4	4	4	5	4	5	5	4	4.38	0.52
EE1c	Guide development & refinement of EA design principles of specific EA intervention	3	4	4	3	4	4	4	4	3.75	0.46
EE1d	Refine EA visions, goals, purpose, principles & scope to accommodate changes	4	4	5	5	5	4	4	4	4.38	0.52
EE2	**Maintain balance between short & long term orientation**	**3.75**	**4.25**	**3.50**	**4.00**	**3.75**	**4.00**	**3.75**	**4.50**	**3.94**	0.32
EE2a	Guide the achievement of long-term goals via short-term goals	4	3	5	5	5	4	4	5	4.38	0.74
EE2b	Continuously improve EA products & artifacts to meet stakeholder needs	5	4	3	4	4	5	4	5	4.25	0.71
EE2c	Respond to temporary shift in EA goals with managed ad-hoc EA products	4	5	3	3	4	4	3	5	3.88	0.83
EE2d	Duration of short term goals should be at most 3 months	2	5	3	4	2	3	4	3	3.25	1.04
EE3	**Provide decision making & governance mechanisms to promote stakeholder commitment**	**4.33**	**4.00**	**4.67**	**4.00**	**4.33**	**4.67**	**3.00**	**3.67**	**4.08**	0.56
EE3a	Employ cross functional decagons making mechanisms at each level of the enterprise	4	3	5	5	5	5	5	3	4.38	0.92
EE3b	Domain specific planning at a level should be based on collective decision at that level	4	4	5	4	4	5	3	4	4.13	0.64
EE3c	Promote co-evolution of domain strategies	5	5	4	3	4	4	1	4	3.75	1.28
EE4	**Promote development of skilled teams, adaption of methods & tools, and definition of requirements**	**3.60**	**3.60**	**3.40**	**3.80**	**3.80**	**3.60**	**3.60**	**3.40**	**3.60**	0.15

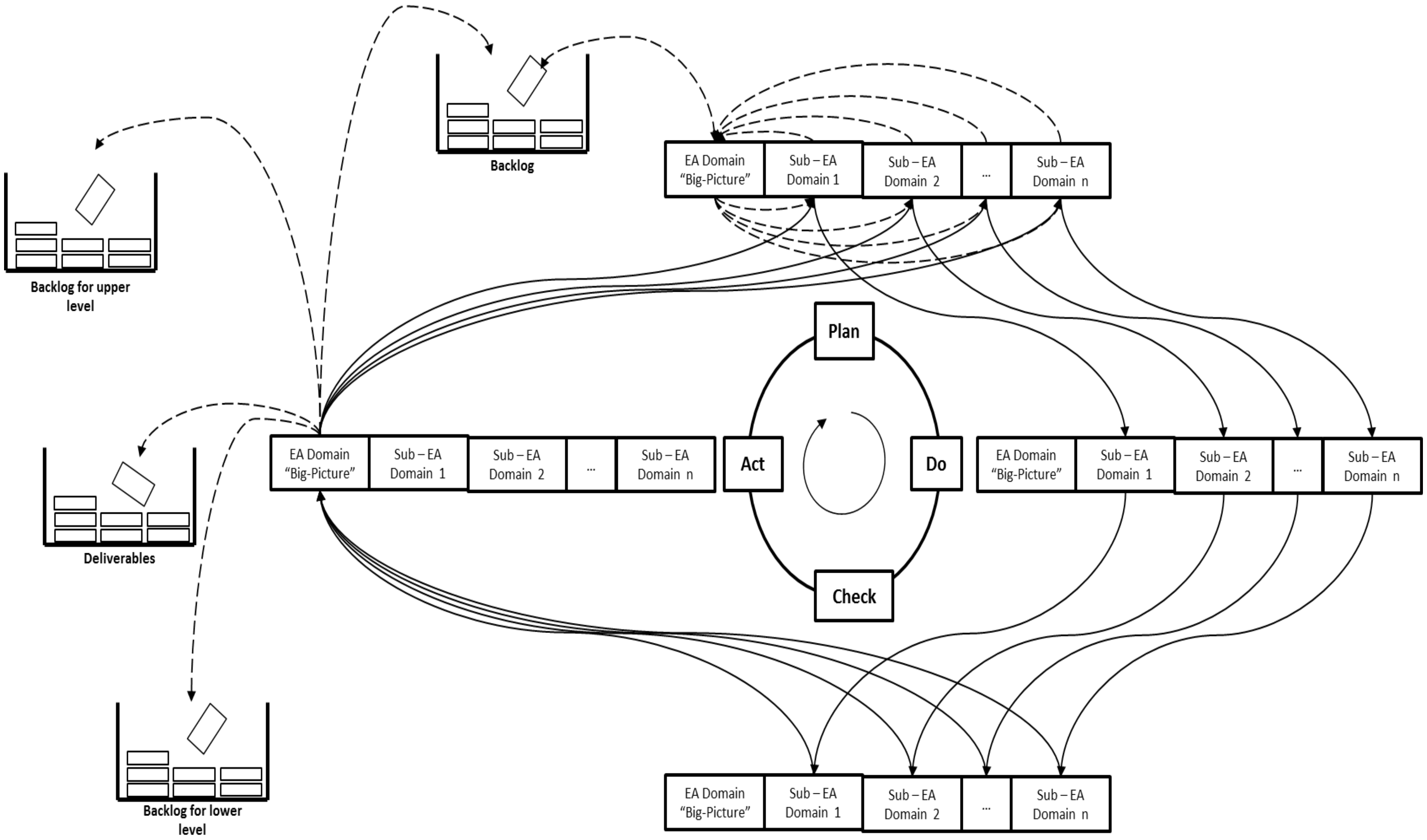
	Essential Elements	Int.A	Int.B	Int.C	Int.D	Int.E	Int.F	Int.G	Int.H	Mean	SD
EE4a	Promote development of skilled teams	3	5	4	3	4	5	2	3	3.63	1.06
EE4b	Support wide range of project management methods and architecting tools	4	2	4	5	3	3	4	2	3.38	1.06
EE4c	Requirements should be contingency based and described at each level with details at lower levels	4	4	3	4	4	4	4	3	3.75	0.46
EE4d	Specify criteria for goal fulfillment for each EA endeavor	3	4	3	4	4	4	4	5	3.88	0.64
EE4e	Manage changes to requirements and success criteria with a change management process	4	3	3	3	4	2	4	4	3.38	0.74
EE5	**Support appropriate representation of artifacts & communicate documentation in a common language**	**3.75**	**3.50**	**3.25**	**3.25**	**3.50**	**3.25**	**4.25**	**3.50**	**3.53**	0.34
EE5a	Promote establishment of common frame of reference and terminologies	3	5	3	3	4	4	4	4	3.75	0.71
EE5b	EA products at a level should be represented with appropriate details for stakeholders at that level	4	3	5	5	5	3	5	4	4.25	0.89
EE5c	Changes to terminologies, EA products, and projects should be communicated across the organization	4	4	2	3	2	3	4	3	3.13	0.83
EE5d	Guide development and refinement of representation principles for specific EA endeavors	4	2	3	2	3	3	4	3	3.00	0.76
EE6	**Mechanism to develop and maintain an integrated architecture repository**	**3.75**	**4.00**	**3.75**	**3.75**	**4.00**	**4.25**	**5.00**	**4.00**	**4.06**	0.42
EE6a	Architecture repository should be developed as knowledge repository	4	4	4	4	5	3	5	4	4.13	0.64
EE6b	Architecture repository should serve as a source of knowledge for revamping EA products	4	4	3	4	4	4	5	4	4.00	0.53
EE6c	Architecture repository should be maintained with commitment from all levels of the enterprise	4	4	4	3	2	5	5	3	3.75	1.04
EE6d	Mechanism that allows stakeholders responsible for an EA product to make changes to the product	3	4	4	4	5	5	5	5	4.38	0.74

	Essential Elements	Int.A	Int.B	Int.C	Int.D	Int.E	Int.F	Int.G	Int.H	Mean	SD
EE7	**EAM method as a dynamic capability and support agility capabilities, and agility providers**	**3.75**	**3.50**	**3.75**	**3.25**	**5.00**	**4.50**	**4.00**	**4.50**	**4.03**	0.59
EE7a	Stakeholders should be allowed to contribute their individual knowledge to promote collective goals	3	4	5	3	5	5	4	4	4.13	0.83
EE7b	Lessons learnt should integrated into existing, or be used when creating EA products and processes	3	4	4	4	5	5	4	4	4.13	0.64
EE7c	Coordinate allocation of resources, and create an environment of CSF	4	2	3	2	5	3	3	5	3.38	1.19
EE7d	Mechanism to develop new or revamp existing EA products during strategic endeavors	5	4	3	4	5	5	5	5	4.50	0.76
EE8	**Promote the anchoring/integration of its mechanisms into the culture of the organization**	**3.67**	**4.00**	**4.33**	**3.00**	**3.67**	**3.67**	**4.33**	**3.33**	**3.75**	0.46
EE8a	EAM mechanism embedded into daily routines of organization	5	2	5	3	5	2	5	4	3.88	1.36
EE8b	Mechanism to promote education of all stakeholders on EA values and principles	4	5	4	3	2	5	3	3	3.63	1.06
EE8c	Top management should lead organizational change processes induced by EA.	2	5	4	3	4	4	5	3	3.75	1.04

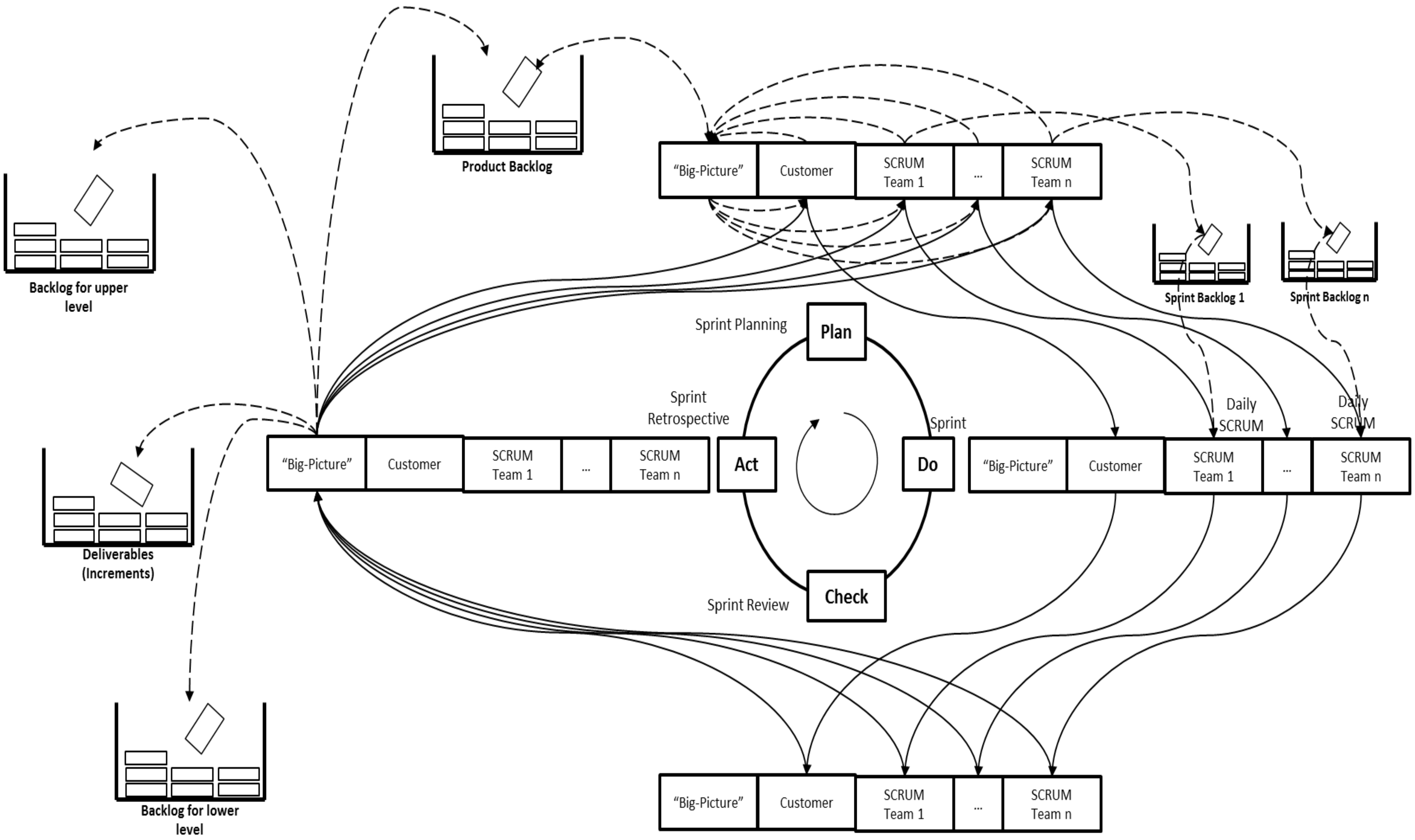
APPENDIX 2 THE AGILE EAM PROCESS FOR MULTIPLE BUSINESS DOMAIN ARCHITECTURE



APPENDIX 3 THE AGILE EAM PROCESS FOR INTRA-EA DOMAIN ARCHITECTURE



APPENDIX 4 THE AGILE EAM PROCESS FOR MULTIPLE DEVELOPLENT (E.G. SCRUM) TEAMS



APPENDIX 5 ILLUSTRATIVE SCENARIO

MedAMore is a chain of drug stores. It started as a regional chain in 1960. In 1995, it developed an innovative software system that enabled it to run drug stores very efficiently. It called this system MedAManage, or MAM. MAM incorporated some innovate business ideas, such as patient-relationship management, inventory management, automated insurance billing, and even utility optimization.

MAM consisted of three programs: MAM/Store, which ran on a small computer at a drug store; MAM/Warehouse, which ran on a server in a regional warehouse; and MAM/Home, which ran on a large server at the home office. These three programs communicated through files that were transferred from one location (for example, a store) to another (for example, a regional warehouse). When reliable communications lines existed, file transfers could occur through FTP. The system was also flexible enough to accommodate transfers through courier, where necessary.

By 2000, MedAMore was doing quite well—in part, because of the cost-cutting moves enabled by the MAM system. MedAMore decided to begin expansion. To do this, it purchased three regional chains. With these purchases, MedAMore extended its reach through the southeast quadrant of the U.S.

By 2002, it was clear that the same software systems that had initially fueled MedAMore's success were now hampering its future. Some of the problems MedAMore was running into were the following:

- MAM/Store required regional specializations. For example, different insurance plans needed to be supported in different regions, and these all required changes to the MAM/Store module.
- The regional warehouses that had been acquired through acquisition each had different ways of receiving orders from the retail stores and different procedures from ordering supplies from the wholesalers. Each of these differences required changes to the MAM/Warehouse module.
- The file-transfer approach to information sharing that had worked so well when MedAMore consisted of 30 drugstores, one regional warehouse, and one home office were turning out to be difficult to coordinate among 200 drugstores, four regional warehouses, two geographic offices, and one home office. Files were often delivered late, sometimes not at all, and occasionally multiple times. This made it difficult for the home office to access reliable, up-to-date financial information, especially in the areas of sales and inventory.

It was clear to MedAMore management that the MAM system needed many enhancements. However, upgrading this system was difficult. Each of the three modules (store, warehouse, and home office) was huge, inefficient, and cumbersome, and each included functionality for everything that each entity might need. The modules had grown to over 1 million lines of code each. It was difficult to change one function without affecting others. All of the functions accessed a single database, and changes to one record definition could ripple through the system in an unpredictable fashion. Changing even a single line of code required a rebuild of the entire multimillion-line module.

MedAManage had become MedANightmare. Debugging was difficult. Software builds were torturous. Installing new systems was hugely disruptive. These technical problems soon created internal conflicts within the home office of MedAMore. The business side of MedAMore wanted to acquire two more regional chains, but IT was still struggling to bring the existing acquisitions online. This resulted in a rapidly growing divide between the business and the technical sides of MedAMore. **The business side saw IT as reducing business agility**. **The technical side saw the business side as making impossible demands and blamed it for refusing to consult IT before entering into acquisition discussions.** The distrust had reached such a point that, by 2005, the CIO was no longer considered part of the executive team of MedAMore. The business side distrusted IT and tried to circumvent it at every opportunity. The technical side built its IT systems with little input from the business folks. Several large and expensive IT initiatives were ignored by the business side and were eventually abandoned.

By 2006, MedAMore was in crisis. It clearly needed to revamp its technical systems to make them easier to specialize for regional requirements. This was going to be an expensive proposition, and MedAMore couldn't afford for the effort to fail.

Just as importantly, MedAMore also had to rebuild its internal relationships. The constant bickering and distrust between business and IT was affecting morale, efficiency, and profitability. A company that only five years earlier was an industry leader in profitability—in large part, because of its innovative use of IT—was now struggling to stay out of the red—in large part, because of the inflexibility of those same IT systems.

Cath, the CEO of MedAMore, desperately needed a solution. At a CEO conference, she heard how many of her peers were using enterprise architectures to build stronger partnerships between their technical and business groups and deliver more cost-effective IT systems that enabled business agility. Cath decided that this approach merited further investigation. She asked Irma, her CIO, to prepare a recommendation on the use of an enterprise architecture within MedAMore. Irma was impressed with the approach, but recognized that any such initiative needed to be driven from the top and needed to involve the business side from the start. On Irma's recommendation, Cath called a meeting with Bret, the Vice-President of Business, and Irma. Cath announced that she had decided to create a common enterprise architecture for MedAMore that would unite its technical and business people. This common enterprise architecture would be named MedAMore-Enterprise Architecture, or MAM-EA. After it was completed, MAM-EA would drive all new IT investment and ensure that every dollar invested in IT was delivering the maximum value to the business.

Cath knew that MAM-EA was a bet-the-company decision for MedAMore. The MAM-EA vision had to work. Cath was depending on Bret (the business side) and Irma (the IT side) to make it work.