RHODES UNIVERSITY

Where leaders learn

A Framework for the Development and Measurement of Agile Enterprise Architecture

A thesis submitted in fulfilment of the requirements for the degree of

MASTER OF COMMERCE

OF

RHODES UNIVERSITY

BY

BOKANG MTHUPHA

Department of Information Systems

January 2012

Abstract

Enterprise architecture is the high-level design of the entire business, facilitated by enterprise architects.

„Agile enterprise architecture" is the term used in enterprise architecture to describe an architecture that caters for future unknowns, enabling change to occur rapidly without undue resource utilization, yet in a controlled manner and with minimal adverse impact.

Some enterprise architects still use outdated, rigid approaches to enterprise architecture which are incompatible with today"s business environment. In addition, there is limited research into methods that can be applied to measure the agility of enterprise architecture.

The current environment is such that there is a need for a more agile approach to developing and measuring enterprise architecture.

This work will lead to the creation of a Framework for The Development and Measurement of Agile Enterprise Architecture.

In support of the main goal of the development of the framework, a literature review will be conducted focusing on the necessary sub-goals of the research. The first sub-goal of the literature review is to develop a comprehensive definition for enterprise architecture (referred to as EA), as well as discover how it is currently practiced. Thereafter, the literature review will investigate a comprehensive definition for agility and research why it is emerging as a critical topic. The next chapter of the literature review will research how agility fits within the context of EA, uncovering a comprehensive definition for agile EA and the best practices in agile EA development. The final chapter of the literature review will investigate suitable measurement techniques that can be used to assess the level of agility of EA.

On completion of the literature review, a preliminary framework will be created using the most important contributions from the literature.

An empirical study will be conducted to explore the definitions for EA, agility, agile EA, the methods to measure the agility of EA and the concepts for the development of agile EA summarised in the preliminary framework.

Data analysis follows an interpretive and qualitative approach based on four case studies through interviews with systems experts in four South African organizations in one province.

Each interviewee was interviewed once. The initial interview with a Principal Consultant on Enterprise Strategy Consulting at organisation # 1 formed the basis of an exploratory study; the results of which were used to refine the research instrument and preliminary framework. Thereafter, a more rigorous empirical study focused on interviews with the Chief Architect, Senior Manager in Advisory Services and an Enterprise Architect at organizations 2, 3 and 4 respectively was conducted.

The research follows an inductive approach to capture the interpretive experiences of participants and develop theoretical propositions from them.

Following the exploratory pilot study it became necessary to make changes to the preliminary framework and initial survey instrument created. Thereafter, the empirical study consisting of the remaining three cases was conducted to test the important aspects of the framework and literature definitions. The analysis of the results of the empirical study prompted further changes to the theoretical framework and definitions created.

The interviews conducted with each of the organizations confirmed the factors for agile EA development as well as the effectiveness of the definitions created in the literature review. The research uncovered that the need for a more agile approach to developing EA and a way to measure the level of agility of EA has become more and more significant in organizations. This begins with a better understanding of EA, agility, how agility fits within the context of EA, as well as appropriate methods to measure agility.

Acknowledgements

First and foremost, I offer my sincerest gratitude to my supervisor, Mr. John McNeill, who has supported me though out my thesis with his patience and knowledge whilst allowing me the room to work in my own way. I attribute the level of my Masters degree to his encouragement and effort and without him, this thesis would not have been completed or written. One simply could not wish for a better or friendlier supervisor.

I would also like to thank my parents, brother and aunt; Drs. O., G. and T. Mthupha and Mrs. P. Ncube for providing me with both moral and financial support throughout all my studies at University.

To all the respondents who participated in this study, I feel honoured and privileged to have been able to discuss the intricacies of agile enterprise architecture development in your organizations. I enjoyed our discussions immensely and appreciated your interest and feedback.

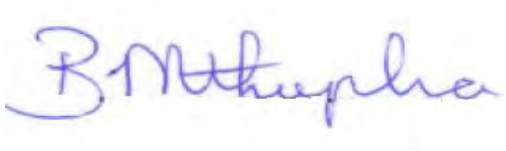
I would be remiss if I did not mention the fine Hamilton building staff who aided me for many years. From the Head of Department, to the cleaning staff; the smooth running and speed at which problems are dealt with is a testament to your efforts. The Department of Information Systems has provided the support and equipment I have needed to produce and complete my thesis and IBM awarded me a generous bursary to fund my studies.

I have had the privilege of "growing up" in the department and attribute my wealth of diverse academic, professional and leadership experiences to the time I have spent here. I am eternally grateful, as I believe these experiences have developed me into the ideal candidate for the positions I apply for and on completion of my studies, I will be going on a clear objective to uphold the stature of Rhodes University by being a sterling ambassador who would embody our motto, "*Where leaders learn"*.

Studying in post-Apartheid South Africa has shown me that opportunity is not something to take for granted. Fulfilling ones potential is not only a personal duty but also a duty to society as a whole.

Declaration

I acknowledge that all references are accurately recorded and that, unless otherwise stated, all work contained herein is my own



Bokang Mthupha

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction

This chapter introduces the research.

The discussion will include a presentation of the research context, as well as the statement of the problem. The results of the research are also presented, together with details of the organization of the thesis.

1.1 Introduction

According to Madni (2008: 49), competitive market forces continue to challenge complex system development initiatives. Research conducted by Edwards (2008: 1) indicates that some enterprise architects still use outdated, rigid approaches to enterprise architecture which are incompatible with today"s business environment. As such, development processes need to go beyond being just flexible (i.e., easy to change) to being agile.

This chapter provides an introduction to the research study. The research context is first explored, followed by the goals of the research with questions that guide the study. The research methodology adopted is also briefly described, followed by a summary of the results and finally the thesis organization is outlined.

1.2 The Problem and its Setting

This section introduces the research problem and its setting.

Giachetti (2009: 39) describes enterprise architecture (referred to as EA) as the high-level design of the entire business. It describes the structure of the business processes, how they are coordinated with each other and how technology supports them. It helps us understand the business" complexity by showing how all the different systems are linked together.

This is a simple definition to start the discourse. The term enterprise architecture however lacks a consistent definition throughout the IT industry. For example, Stevenson (1995) in his belief that EA has a greater role to play than simply at a business process level describes it as a master plan which acts as an integrating force between aspects of business planning such as goals, visions, strategies and governance principles; aspects of business operations such as business terms, organization structures, processes and data; aspects of automation such as

application systems and databases; and the enabling technological infrastructure of the business such as computers, operating systems and networks.

Khosrowpour (1996: 132) suggests that there is a parallel between EA design and city planning. Nolan and Mulryan (1987:63) acknowledge this relationship and add that city planners must design in the face of many unknowns, such as future transportation technologies, changing work, living, and commuting patterns, and so on. As a result of this level of planning, major cities are able to accommodate new technologies for transportation and communication which remain viable for hundreds of years, and which make a major contribution to each city's brand of urban culture.

Saint-Exupery in one of his novels wrote: “As for the future, your task is not to foresee, but to enable it.” In EA as in city planning it is futile to attempt to foresee every possible future change. The architecture must rather provide the capability to enable change to occur rapidly; without undue resource utilization, yet in a controlled manner and with minimal adverse impact (Stevenson, 1995).

„Agile Architecture“ is the term used in EA to describe an architecture that caters for future unknowns, enabling change to occur rapidly without undue resource utilization, yet in a controlled manner and with minimal adverse impact (Stevenson, 1995).

Edwards (2008: 1) believes there are many different threads of concern influencing the need for an agile EA. He favors an adaptive (agile) versus a predictive approach to enterprise architecture design- The potential problem here is that enterprise architects use a large plan up front, rigid process (predictive), instead of a time-boxed, continuous risk-driven improvement process (adaptive) to manage the EA practice and its output.

Flower (2000) acknowledges this approach and adds that it would also overcome the potential problem of a slow project process. When done effectively, this achieves a pragmatic balance between business and architectural priorities while delivering both with agility (Madison, 2010:42).
Kutnick (2006) suggests that an agile architecture will allow an organization to sense environmental change and respond efficiently and effectively to that change.

Nair (2008) adds that one needs to know the current state of the architecture in order to be able to assess the impact of a change so much as to know what is going to change, when,

where, how and why it changes. The first and most important step in this direction is to understand the organization‟s „agility requirements‟. When architecture requirements are gathered, there is a common practice to neglect the need of capturing a set of agility requirements.

But the challenging question is how one can identify agility requirements. There are a number of different viewpoints on this topic. However, one approach that can be taken is to measure the level of agility of an organization relative to a desired level of agility set out in an agility framework (Nair, 2008). The gap between the organization‟s current practices and those set out in the framework will signify the organisation‟s agility requirements.

Allen (2009) supports the belief that agility should be measured and adds that often, the term „agility‟ is bandied about, without real meaning. For example, it is common practice to measure the various quality attributes of a piece of software. Performance comes in the form of response time; capacity is expressed, among other things, in terms of bytes of storage; security is expressed in terms of freedom from malicious software attacks, and so forth. One can argue that the same should be done in terms of agility, not to look at it as just a good thing as many seem to assume, like happiness or moral goodness. It should be defined and measured, as with performance, capacity, and security. Due to its importance in addressing ever-demanding and more complex business requirements, it is imperative to define and measure agility.

Although there are numerous frameworks for those wishing to practice enterprise architecture, an examination of the most common frameworks by the researcher revealed that there is no accepted industry standard or widely used method that exists describing the best way to go about being agile (ChiefInformationOfficerCouncil, 2007; James, Handler, Lapkin and Gall, 2005; Perks and Beveridge, 2003; The Open Group, 2003; U.SDepartmentofDefence, 1986; Zachman, 1987). A comparison of the top four EA methodologies by Sessions (2007) and similar work by Tang and Han (2004) did not reveal anything to the contrary. In addition, the literature (ChiefInformationOfficerCouncil, 2007; James, Handler, Lapkin and Gall, 2005; Perks and Beveridge, 2003; Sessions, 2007; Tang and Han, 2004; The Open Group, 2003; U.SDepartmentofDefence, 1986; Zachman, 1987) showed that there is no accepted industry standard or widely used method for measuring the agility of EA.

The motivation behind the research arises out of a number of factors:

- The lack of comprehensive definition for EA and how it is practiced. Varying views on a definition for EA lead to a lack of unified knowledge and presence of varying cultures, and policies that prevent efficient management of enterprise processes, innovation and building software (Shuja, 2010: 1).
- As with EA, the IT industry does not collectively agree on what agility means, or why it is important in the current business environment.

- Some enterprise architects still use outdated, rigid approaches to EA which are incompatible with today‟s business environment. Their approach is out of a lack of understanding of how agility fits within the context of EA and a belief that EA is done for agility and cannot be done with agility (Edwards, 2008: 1).

- Although there are a number of architecture frameworks that exist, there is no accepted industry standard or method for agile EA describing the best way to go about the process of being agile.

- Identifying the level of agility is the first step in understanding how much agility should be introduced and where; in order for the organization to survive or outperform its competitors. Agility within the context of EA is a relatively new area and therefore there is limited research into a method and guidelines describing how the level of agility of EA can be measured. That does not stop an individual or team from using any available methods for determining the level of agility of EA. However, a lack of a comprehensive, standardized process leaves the determination of the current level of agility open to the interpretation of those engaged in the EA effort which could lead to the use of methods that are in some cases ineffective.

The research is being undertaken in an attempt to there being comprehensive definitions for EA, agility and agile EA, in order for those involved in the EA effort to be fully aware of what EA is, as well as what it means to be agile. The research is also being undertaken in order for there to be a framework containing a set of guidelines or best practices and an EA agility measurement method that is architecture framework agnostic, allowing it to be used in conjunction with any other architecture frameworks that an organization may be using and that will ensure that EA agility is achieved effectively and efficiently.

1.3 Goals of the Research

This section describes what the research is intending to achieve by defining the main problem and sub-problems of the research.

This research proposes to develop a Framework for The Development and Measurement of Agile Enterprise Architecture.

In support of the development of the framework, the research will develop a comprehensive definition for enterprise architecture as well as discover how it is currently practiced. It will also investigate a comprehensive definition for agility and why it is emerging as a critical topic. Under the title „Agile Enterprise Architecture", the research also aims to discover how agility fits within the context of EA, uncovering a comprehensive definition for agile EA and the best practices in agile EA development.

The final goal of the literature review research is to investigate suitable measurement techniques that can be used to assess the level of agility of EA.

1.3.1 Problem

The main problem of the research is to analyse and evaluate the extent of understanding of enterprise architecture, agility and agile EA, the methods that can be used to measure the agility of EA as well as the factors affecting the successful implementation of agile EA.

This will be explored initially through a literature review and thereafter, an empirical study will be conducted to further explore the findings from the literature.

1.3.2 Sub-Problems

The following research questions guide the research study:

- *What is the perception about what enterprise architecture is?*
  This research is initially driven by a literature review to uncover researchers" views of EA and thereafter, a further exploration of how practitioners view EA through semi-structured interviews with four selected organisations in one South African province.

- *What is the perception of the meaning of agility and agile EA?*

  This research is initially driven by a literature review to uncover researchers‟ views and thereafter a further exploration of how practitioners view agility and agile EA through semi-structured interviews with four selected organisations in one South African province

- *What is the perception of the methods can be used to measure the agility of EA?*

  This research is initially driven by a literature review to uncover researchers‟ views and thereafter a further exploration of any methods that practitioners believe can be used to measure the agility of EA through semi-structured interviews with four selected organisations in one South African province.

- *What is the perception of the factors affecting the successful implementation of agile EA?*

  A framework will be developed through a critical review of the important considerations for agile EA uncovered from the literature review. Thereafter these considerations will be further explored though semi-structured interviews with four selected organisations in one South African province in order to expand on existing theory and to explore new theoretical concepts.

  The resulting framework will provide a set of successful agile EA development factors for South African organisations aiming to develop their IT architecture in an agile manner.

1.4 Research Methodology

This section describes the research methodology employed in this thesis.

Yin (2003: 20) states that every type of research should have an implicit, if not explicit research design. This will guide the researcher in the process of collecting, analyzing and interpreting observations. The topics that need to be addressed are what research questions need to be studied, what data are relevant, what data to collect, and how to analyze the results (Yin, 2003: 21).

This research investigates the criteria for the development of agile EA as well as a measurement method that can be applied to measure the agility of an organization‟s EA.

The qualitative and interpretive case study research methodology is chosen to enable an understanding of the phenomenon agile enterprise architecture with an exploratory strategy to identify the factors that affect the development of agile EA.

Interpretive research sees human behavior as the outcome of the subjective interpretation of the environment (Rowlands, 2005: 81). The development of agile EA is assumed not to be an objective phenomena with known properties or dimensions. Interpretive research is thus consistent and compatible with the epistemological assumptions that the world and reality are interpreted by people in the context of social and historical practices (Rowlands, 2005: 83). Experience of the world is subjective and best understood in terms of individual meanings rather than a researcher‟s objective definitions (Rowlands, 2005: 83).

Qualitative research will be conducted by means of semi-structured interviews guided by qualitative survey questions with systems experts in four companies. As the investigation of agility within the process of EA is a new phenomenon; the only criteria for participation in the empirical study is that an organization has a formal enterprise architecture practice. These sources will be employed to gather the data with which to synthesize a framework that will describe the best practices in agile EA and how it can be measured (Yin, 2003:7).

1.4.1 Literature Review

A literature review will be conducted focusing on:

- An analysis of the definitions of EA, describing how it is currently practiced, who is involved and why an organization would embark on an EA effort.
- An analysis of the definitions of agility; uncovering the meaning of agility which will provide a scope of the topic and why it is emerging as a critical topic.
- An analysis of agile architecture, which will present an overview of the topic; explaining how agility fits in the context of EA, the challenges with current methods for developing EA which favour an agile approach, as well as how agile software development principles can be applied to EA to ensure an agile EA. Thereafter, the best practices in agile architecture will be discussed, highlighting how agile EA can be developed as well as the role of the enterprise architect in an agile EA environment showing how different it is in comparison to traditional EA practice.
- An analysis of suitable measurement techniques that can be applied to measure the agility of EA. The discussion will involve an introduction of why agility should be

measured; providing the scope of agility measurement and thereafter go onto an evaluation of both informal and formal methods of measurement.

1.4.2 Preliminary Framework and Pilot Study

- An initial list of factors affecting the development of agile EA is developed into a preliminary framework.
- Questions and key concepts are extracted from the preliminary framework and transformed into an open survey which will be used as a guide in semi-structured interviews with participants in four selected South African organizations.

A pilot study is conducted with a participant from one of the organizations, who assists in the preliminary testing of the framework, which is refined and modified as necessary.

1.4.3 Empirical Work

- A theoretical framework is then developed based on the previous literature as well as emerging issues identified in the pilot study.
- The survey instrument is then refined as needed and used in the collection of data through semi-structured interviews with the remaining participants in each of the three organizations.

1.4.4 Findings and Analysis

- Once data collection in the field is complete, it is analyzed in each of the cases and reports are written identifying the emerging categories of data.
- Adaptations are made to the theoretical framework which enables an exploration of the factors required for the successful development of agile EA.

1.5 Summary of Results

This section summarizes the results of the empirical research.

The framework for the development and measurement of agile EA comprises thirty four factors. It was concluded that that a better understanding of EA and how agility fits within the context of EA (agile EA) is required.

Before an organization can successfully develop agile EA, it needs to achieve a common understanding of the terms „enterprise architecture“, „agility“ and „agile enterprise architecture.“

The definitions created within the literature review for EA, agility and agile EA were comprehensive enough to encompass those provided by the respondents in the empirical study and were therefore deemed sufficient. From the results gathered, it is clear that agility is of growing importance to organizations. Agility within the context of EA is a concept which EA teams are slowly taking into account and realising as an important factor that has to be measured. The organizations investigated in this study generally embrace the principals for the development of agile EA and believe that it is the clear future direction of their EA efforts.

However, the agile principle in the theoretical framework that promoted developers using code to evaluate alternatives was deemed to not work in practice, as it will result in budget and project time overruns. Concerns are evident in the lack of team commitment at the start of an EA project and that organizations find it difficult to have a clear cut deliverable and choose an approach to achieve that deliverable.

In addition, it was found that the EA team should not only consist of architects, but also developers whose responsibility will be working with the architects and coding the models developed. The inclusion of developers in the EA team also led to the definition for agile EA created in the literature review being modified to reflect this change.

In the end, while there may be an attempt to introduce elements of agility into the EA process to develop architecture in an agile manner, the success of these additions all add up to effective leadership and managing.

1.5 Thesis Organization

This section describes how the thesis is organized.
This thesis is organized into 14 Chapters:

Chapter 1: Introduction

This chapter introduces the research. The discussion includes a presentation of the research context, the statement of the problem as well as the research methodology. The results of the research are also presented, together with details of the organization of the thesis.

Chapter 2: Enterprise Architecture

The aim of this chapter is to provide a comprehensive definition of EA, describing how it is currently practiced, who is involved and why an organization would embark on an EA effort.

Chapter 3: Agility

The aim of this chapter is to provide a fundamental overview of agility. A broad view of the topic will be taken; uncovering the meaning of agility which will provide a scope of the topic and why it is emerging as a critical topic.

Chapter 4: Agile Enterprise Architecture

The aim of this chapter is to introduce „agile architecture". The discussion will include the scope of agile architecture which will present an overview of the topic; explaining how agility fits in the context of EA and providing a comprehensive definition for agile EA, the challenges with current methods for developing EA which favour and agile approach, as well as how agile software development principles can be applied to EA to ensure an agile EA. Thereafter, the best practices in agile architecture will be discussed, highlighting how agile EA can be developed as well as the role of the enterprise architect in an agile EA environment showing how different it is in comparison to traditional EA practice.

Chapter 5: Measurement Techniques

The aim of this chapter is to discuss suitable measurement techniques that can be applied to measure the agility of EA. The discussion will involve an introduction of why agility should be measured; providing the scope of agility measurement and thereafter go onto an evaluation of both informal and formal methods of measurement.

Chapter 6: Preliminary Model

The aim of this chapter is to sum up the theoretical work by defining a framework and guidelines for the implementation of agile EA as well as a measurement method that can be used to assess the level of agility within an organization.

Chapter 7: Research Methodology

The aim of this chapter is to present the research methodology adopted to explore the extent of understanding of EA, agility and agile EA, methods used to measure the agility of EA and the factors for the successful development of agile EA in organizations. The research questions, research method, unit of analysis, research instrument, the approach to analyzing data and the research design are discussed.

Chapter 8: Exploratory Pilot Study

The aim of this chapter is to present the details of an exploratory pilot study conducted through an interview with a systems expert at organization # 1. The extent of understanding of enterprise architecture, agility and agile EA, the methods used to measure the agility of EA as well as the factors affecting the development of agile EA at the organization are explored.

Chapter 9: Theoretical Framework

The aim of this chapter is to present the adjustments to the preliminary framework based on the results from the exploratory pilot study and proposes a new theoretical framework that identifies the key factors that affect the development of agile EA in an organization.

Chapter 10: Design of the Empirical Study

The aim of this chapter is to detail the design of the empirical study. The empirical study is intended to further explore the theoretical framework proposed in Chapter 9. The design of the survey instrument is discussed in detail.

Chapter 11: Analysis of the Results of the Empirical Study

The aim of this chapter is to present the analysis of the results of the empirical study and to discuss the impact that they have on the research and the theoretical framework. A comparative factor analysis is made of the extent of understanding of enterprise architecture, agility and agile EA; the methods used to measure the agility of EA as well as each of the factors affecting the development of agile EA. A holistic analysis of the results is also provided.

Chapter 12: Revisions to the Theoretical Framework

The aim of this chapter is to present the revisions to the theoretical framework based on the results of the empirical study and their analysis in Chapter 11.

Chapter 13: Conclusion

The aim of this chapter is to conclude the research by identifying the most significant contributions of the research, limitations and by suggesting areas of future work.

Chapter 2: Enterprise Architecture

Chapter one provided background for the research study.

The aims and objectives of this chapter are to provide a comprehensive definition of EA, describing how it is currently practiced, who is involved and why an organization would embark on an EA effort.

A wide range of literature describing the various views is explored to develop a comprehensive definition of EA providing the scope of EA and also highlighting the importance of a consistent definition for EA. Thereafter, the current state of EA will be explored describing how EA is currently practiced, the participants of the process as well as the potential impact it can have on an organization. The chapter will end with a summary of the findings from the EA literature.

It was concluded that there is a need for a consistent and comprehensive definition of EA, as well as active participation from all the people within the organization. EA is both a management program consisting of phases and a documentation method that when executed correctly touches every part of the enterprise and brings with it numerous benefits. It is a never ending program as the requirements of the business will continue to evolve. The process of EA is driven by enterprise architects and needs support from other members of the organization. Although there are numerous reasons why organizations would embark on an EA program, IT-business alignment was cited as the top driver.

2.1 Introduction

This section presents a review of the many definitions of EA, showing how the definitions have evolved and how EA now has a greater role to play within the organization. The section ends with the presentation of a comprehensive definition that encompasses all the definitions provided in the review and that will be used throughout this thesis.

The term „enterprise architecture" is sometimes used in technology strategy and management meetings of a company. However, when inquiry is made into the meaning of this term and its practice, there are as many views as there are individuals. Varying views on EA lead to lack of unified knowledge (knowledge required to relate systems, processes, and people in different enterprises within an enterprise, understanding the dependencies that exist between

them, and relating them to the overall strategic goal of the enterprise) and presence of varying cultures, and policies that prevent enterprises from managing the process of innovation, building software and policy solutions to resolve critical business challenges, and integrating those solutions across the enterprise in an effective and efficient manner (Shuja, 2010: 1).

According to Shuja (2010: 1), EA is sometimes confused and other times misinterpreted by both practitioners as well as researchers.

To understand the EA discipline, it is important to understand its heritage (Hjort- Madsen, 2005). The Open Group (2003) defines an enterprise as a collection of organizations that have a common set of goals and/or a bottom line. In that sense, an enterprise can be a government agency, a whole corporation, a division of a corporation, a single department, or a chain of geographically distant organizations linked together by common ownership. It is made more complex by defining supply chains of partners, customers and suppliers.

The Oxford English Dictionary defines architecture as, "The art or science of building, thing built, structure; style of building; construction," (Anderson and Backhouse, 2008: 1). According to Harmon (2003:1), the term „architecture" has been used for many years within the Information Systems (referred to as IS) community to refer to various types of overviews that provide guidance to software systems and applications developers. In the mid-Nineties, the term began to be used by business managers, especially those involved in enterprise planning and in business process reengineering projects, to describe an overview of the business. For example, some managers began to refer to a high-level description of all of the core business processes in an organization as, „Business Process Architecture."

Just as builders would not undertake the construction of a house or an office building without an architecture, documented in various blueprints, so software developers should not undertake the development of software systems without a detailed plan, documented with software „blueprints" of various kinds (Harmon, 2003: 1).

To that end, „architecture" is defined in the software development context by IEEE standard 1471-2000 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 (Land, 2003:2).

Today has seen the success of a movement started in the early nineties among both business managers and IS managers to use the term „enterprise architecture" to refer to a broad description of all of the key elements and relationships that make up an organization. Increasingly, when managers talk about the alignment between business processes, goals, IS applications and middleware systems, they rely on enterprise architecture to define how the business-IS alignment should be achieved (Harmon, 2003: 1).

EA as such is not a new discipline as it replaces the systems-level approaches to IT resource development that have dominated the last several decades and have left many organizations with stovepipe and duplicative IT systems (Bernard, 2004: 32).

According to Gartner (2008a: 1), EA means significantly different things to different organizations (or even among different constituencies in the same organization). Enterprise architects should use this definition to help articulate what enterprise architecture is within their organization.

Tarabanis, Peristeras and Fragidis (2001) view EA more at a data modeling level, using the concept to create an Integrated Process & Data Repository for Public Administration (referred to as PA) that could serve as a knowledge base for all PAs as well as structure the processes and data in a Generic Process & Data Model.

However, according to some researchers, EA should have a greater role to play in aligning technologies with businesses to achieve competitiveness.

Giachetti (2009: 39) subscribes to this school of thought and describes EA as the high-level design of the entire business that helps us understand the business" complexity by showing how all the different systems are linked together. In this definition, a design of the entire business is intended, yet in practice much of the discussion of EA takes place within the software engineering community. On the other hand, in truth, EA should be and is intended to be for the entire business. The software component is only one part of the EA which is why industrial engineers and other professionals should be aware of and participate in the development of EA.

It appears from their definitions that Giachetti (2009: 39) and Tarabanis, *et al*. (2001) assume that implementation is not within the scope of EA. In fact, that is where it should play a central role in order to ensure successful implementation of the business strategy that

represents corporate goals. Therefore, even if one was to take this view of EA which excludes implementation, the likely outcome is to be left with dealing with the most challenging part of implementing the strategy on your own.

Spewak (1992: 1) takes implementation into account, referring to EA as "Enterprise Architecture Planning" and defining it as, "the process of defining architectures for the use of information in support of the business and the plan for implementing those architectures."

Although the definition provided by Spewak (1992: 1) includes implementation within its scope, the author fails to elaborate on his meaning of „architectures." In addition, since the definition seems to focus on defining those architectures for the use of information in support of the business, it may appear that the author agrees with Tarabanis, *et al*. (2001), viewing EA more at a data modeling level.

A better description is provided by Edwards (2006:1) who defines EA in the simplest of terms by putting the definitions of the two words together: "The fundamental organization of all enterprise systems, the components and their relationships to each other and the environment and all principles governing the enterprise systems design and evolution."

While succinct, such an approach to EA does not take into account the core components (cultures, people, and processes within both operations as well as innovation groups of an enterprise) required to ensure strategic delivery of enterprise-wide solutions and therefore presents a perfect recipe for failure (Shuja, 2010: 2).

A more exhaustive definition is provided by Schekkerman (2005: 18) who sees EA from a holistic point of view: "Enterprise Architecture is a complete expression of the enterprise; a master plan which acts as a collaboration force between aspects of business planning such as goals, visions, strategies and governance principles; aspects of business operations such as business terms, organization structures, processes and data; aspects of automation such as information systems and databases; and the enabling technological infrastructure of the business such as computers, operating systems and networks."

As shown by the definitions provided, the majority of researchers in EA literature appear to focus on EA as a noun. It has been found that when people focus on EA as a noun, they focus on the outputs and are more concerned about producing a predefined set of deliverables, rather than about meeting the strategic imperatives of the enterprise (Gartner, 2008a: 2).

Research conducted by Bernard (2004: 33) views EA as course of action, rather than a noun and stresses that EA is both a management program and a documentation method that together provides an actionable, coordinated view of an enterprise‟s strategic direction, business processes, information flows, and resource utilization.

The recommendation provided by Gartner research into a comprehensive, consistent definition for the term „enterprise architecture‟ is to view EA as a process, rather than on the deliverables that are produced. Their definition of EA has focused on the verb, emphasizing that EA is a translation process; the environmental trends affecting the enterprise and its business strategies are articulated and examined so that the optimal future-state vision can be derived (Gartner, 2008a: 2).

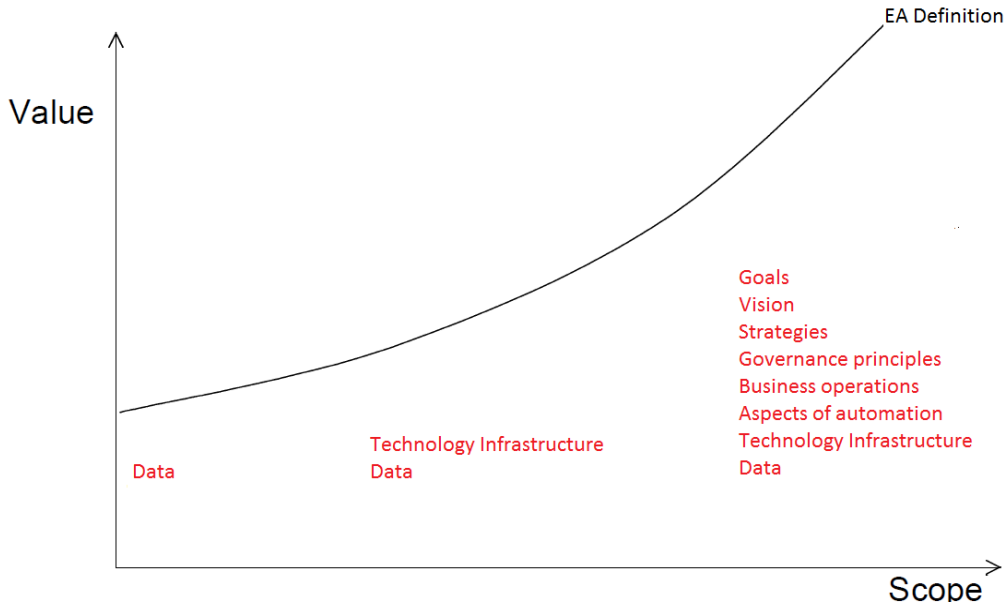
This is important as it has often been seen that some researchers and those involved in the EA effort tend to refer to the collection of artifacts that are produced as the „enterprise architecture.‟ The process by which those artifacts are derived and applied is far more important than the artifacts themselves. By focusing the definition of EA on the process that creates, applies and maintains the artifacts, the proper emphasis is maintained. This single-minded focus on deliverables is a mistake because it can lead to mountains of artifacts (requirements, models, principles, guidelines, standards) that are not necessarily connected to the strategic imperatives of the enterprise and are therefore not leveraged across the organization.

Sessions (2006: 4) believes that if adding business value is not the ultimate goal, then the energy put into EA has been badly misplaced.

Looking at EA as a process will ensure that the outputs of the process; the requirements, principles and models will describe the optimal future state, provide an analysis of the gaps between the future state and the current state, and present road maps that support the evolution of the enterprise to the future state by closing the gaps (Gartner, 2008a: 2).
The contemporary literature does not offer a consistent definition of EA. However, as is illustrated with the definitions above, some consensus can be detected in that the various authors all centre their definitions on creating an overview of the organization showing the alignment of the business and technology with the aim to make the IT more business driven. Further, some definitions include processes in relation to management and governance, which will assist the organization in creating a dynamic EA (Hansen, 2006: 8).

Since its inception, it is clear from the literature that the EA definition has evolved. Its purpose, which is to bring change within an organization, has however remained the same. The growth in scope of the definition of the term has meant that if executed correctly, EA will positively affect a larger part of the organization than in the past (Gartner, 2008b: 4). Based on the definitions of EA uncovered in the literature, Figure 1 below shows that the increasing scope of EA continues to increase the number of areas that could positively be affected by its undertaking; which as a result has increased the value it could potentially bring to an organization.

Figure 1: The Increasing Value of Enterprise Architecture

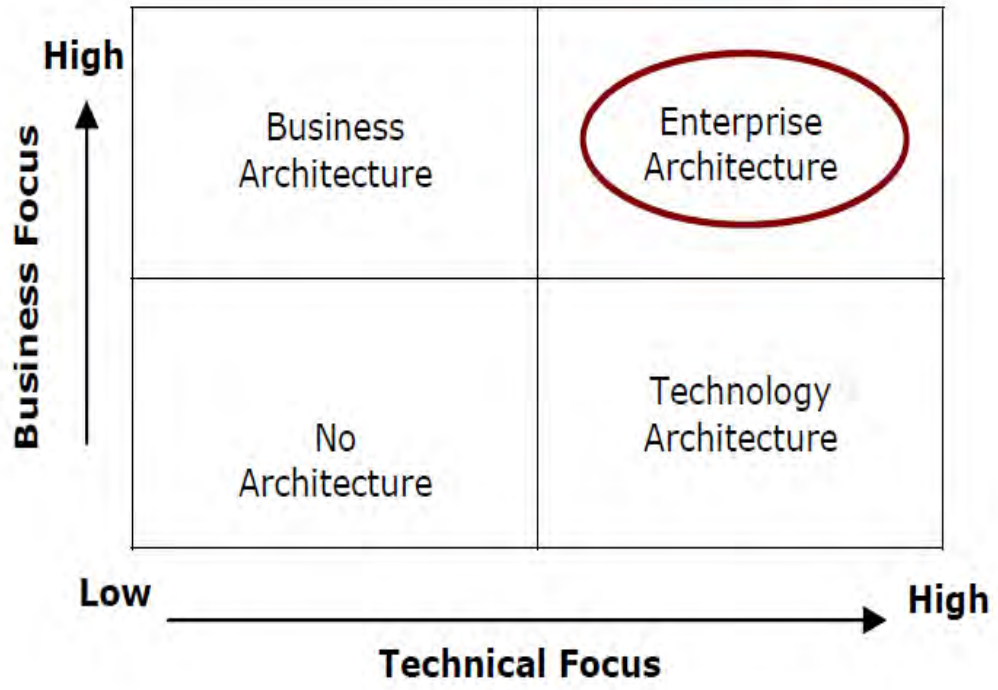


(Bredemeyer Consulting, 2003: 1)

The definitions show that the EA literature today takes a more holistic perspective on the management of IT than has previously been seen, but many of the concepts used today are based on ideas and concepts developed in the earlier system-level approach (Hjort-Madsen, 2005).

As Figure 2 on the next page shows, the essence of EA is that it focuses equally on business, technology, and their interrelationship (Hansen, 2006: 7).

Figure 2: The Focus of Enterprise Architecture



(Hansen, 2006: 8)

As mentioned previously, numerous, conflicting interpretations of the term „enterprise architecture" cause confusion and obstruct the benefits that a common understanding of the concept enables (Gartner, 2008a: 1).

The Gartner definition that views enterprise architecture as a process ensures that value will be added as a result of the process. The definition is also consistent with the work to be conducted in later chapters of this thesis; uncovering a method to evaluate the level of agility present in an organization"s process to developing enterprise architecture. In addition, it is a comprehensive definition that covers multiple dimensions of the subject; what EA is, the scope of EA, what is achieved after its execution and the benefits of its implementation (Gartner, 2008a: 2) as well as the business justification for undergoing the process (Sessions, 2006: 3).

According to Shuja (2010: 1), when confronted with EA, it is important to clarify and refine the definition in order to communicate its meaning in a particular situation.
Within the context of this research, the Gartner (2008a) definition for enterprise architecture will be used:

Enterprise Architecture is the process of translating business vision and strategy into effective enterprise change by creating, communicating and improving the key requirements, principles and models that describe the enterprise's future state and enable its evolution. The scope of the enterprise architecture includes the people, processes, information and technology of the enterprise, and their relationships to one another and to the external environment. Enterprise architects compose holistic solutions that address the business challenges of the enterprise and support the governance needed to implement them (Gartner, 2008a: 2).

The next section will discuss the current state of enterprise architecture.

2.2 The Current State of Enterprise Architecture

The previous section presented the various definitions of EA showing how it has evolved and concluded with a comprehensive definition for EA developed from the literature review. This section provides insight into how EA is currently practiced (section 2.2.1), as well as who is involved in its practice (section 2.2.2). Section 2.2.3 describes the benefits of successful EA explaining why an organization would embark on an EA program.

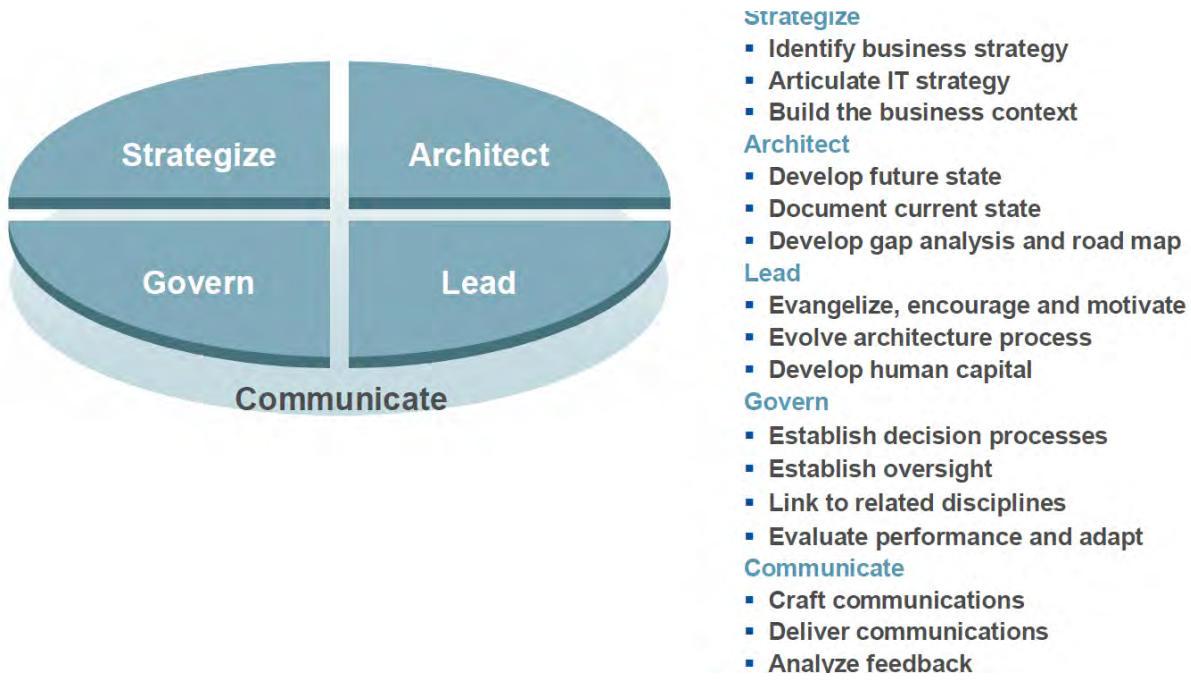
To understand the practice of EA, it is important to differentiate between the process of enterprise architecture (EA) and the role of the enterprise architect. EA should be viewed as a process whose most important deliverable is change (Gartner, 2008a: 1).

2.2.1 The Process of Enterprise Architecture

This subsection of current state of enterprise architecture describes how EA is currently practiced by describing the EA process and the activities at each stage of the process.

According to Gartner (2008: 3), undertaking EA is a process facilitated by enterprise architects with a repeating lifecycle that is constantly evolving. This process can be presented in the context of the EA activity cycle illustrated in Figure 3 on the next page (Gartner, 2007: 2).

Figure 3: The Enterprise Architecture Activity Cycle



(Gartner, 2007: 3)

2.2.1.1 Strategize

Strategizing encompasses identifying the organization‟s strategic direction, goals, and initiatives, and it provides a clear description of the contribution that IT will make in achieving these goals (Bernard, 2004).

Gartner (2008b: 4) agrees with this view of strategy, adding that a solid EA should not only provide clarity on the organization‟s direction, but also offer a bridge between strategy and implementation by providing models, guidance, road maps and artifacts that enable business and IT to make better decisions, as well as to achieve the enterprise change agenda.

Hansen (2006: 23) views the movement from strategy to implementation from an architectural standpoint; stating that it involves evaluating the current architecture view of „as-is‟ strategies, processes and resources and describing the desired future architecture view of „to-be‟ strategies, processes and resources and creating a management plan to move from the current to the future view.

Dziewulski, Iannuzzi, Menzel, Ramchandi and Rothschild (2003: 20) recognize the importance of strategizing, referred to as „Planning the EA Program‟ in their research. They

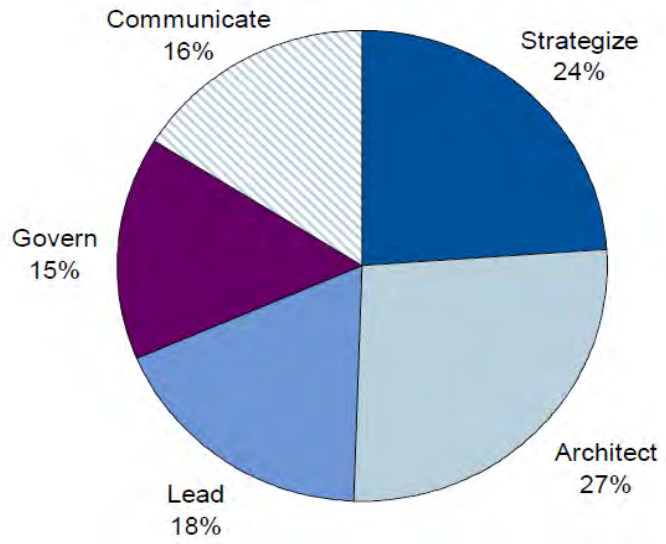
however believe executive buy-in and support (sponsorship) should be obtained before any planning occurs.

Harmon (2003: 10) agrees that nothing happens until senior management agrees that EA is needed. Thus, everything starts with an internal sales effort on the part of those who see the advantages of EA. Since EA is often a responsibility of the IT organization, the CIO or some other senior IS manager often leads the effort to sell other senior managers on the advantages of creating and maintaining EA.

Dziewulski, *et al*. (2003: 7) add that after obtaining executive buy-in, strategizing requires building a foundation that focuses on implementation of roles and responsibilities, defining the scope of the architecture effort, and providing the necessary resources to effectively develop the architecture products. The result is the development of the business context, which includes trends, common requirements and principles and simply makes that which is architected explicitly business-driven.

Devoid of good business context, enterprise architecture programs tend to languish. It is no surprise that while all aspects of the EA activity cycle are of great importance, the „Strategize" component which leads to the development of the business context is a large focus of EA team effort as Figure 4 below illustrates (Gartner, 2007: 3).

Figure 4: Distribution of Team Effort by Components of the Activity Cycle



(Gartner, 2007: 9)

2.2.1.2 Architect

The „Architect" component of the activity cycle entails designing the future state and articulating and base lining the current state. Additionally, this is where gaps between the current and future states are identified, and a migration plan or road map is developed (Gartner, 2007: 5).

In their research into advancing EA maturity, Dziewulski, *et al*. (2003: 7) refer to this step of the EA process as, "Define an Architecture Process and Approach." They point out that although overall goals and objectives of the EA effort are typically established very early in the program; some additional scoping is done as part of this phase. Specifically, scope definition considerations should cover:

- Relative emphasis on Business and Technology aspects.
- Relationship between Business and Technology drivers.
- Scope and depth of „as-is" architecture definition.
- Scope and breadth of „to-be" architecture definition.
- Scope and depth of architecture transition plans.

According to DiMario, Cloutier and Verma (2008: 19), what is known as an EA framework helps in the organization of architectural information. These EA frameworks are used to define what the EA program will document and can thus be viewed as a logical structure for classifying, organizing and managing complex information (Hansen, 2006: 9).

Each organization needs to determine what types of documents and models it will include in its EA. The result defines that organization"s framework. Most organizations start with one of the widely used frameworks like the Zachman Framework, and then tailor it to their specific needs. Each organization that elects to rely on EA must necessarily decide on the specific types of documents and diagrams they intend to keep track of with their architecture. Many begin with a subset simply to limit the time required to establish all the necessary relationships, in order to get an initial EA in place quickly. Later, when the initial architecture is established, they expand the framework and commit to the addition of new information (Harmon, 2003: 11).

2.2.1.3 Lead

Leading consists of changing the culture of the organization to embrace EA, evolving the architecture process commensurate with the organization's needs and capabilities, and developing or evolving the enterprise architecture team and constituents (Gartner, 2007: 5).

Dziewulski, *et al*. (2003: 15) in their research identify the „Lead" phase of the Gartner (2007) EA process in their „Develop the Target Enterprise Architecture" phase and add that changing the culture of the organization to embrace EA involves identifying thought leader(s) and providing appropriate training. A thought leader is a person, or people with the vision to understand the overall purpose and needs of the EA and who can envision a way to embody it in a practical, working architecture and communicate that understanding to the rest of the organization. Thought leadership is not something that is easily taught. However, training can be provided that shows how to recognize emerging thought leaders, nurture their development, and utilize their talents. This may be included as part of project management training, personnel leadership training, or other suitable forums (Dziewulski, *et al*., 2003: 16).

2.2.1.4 Govern

A governance process is crucial to ensure the effectiveness in an organization"s EA. EA governance centres around creating and making sure that the EA processes and structures are followed and EA governance is thus a key aspect of ensuring positive EA performance (Hansen, 2005: 18).

Harmon (2003: 6) believes that this is the role of an Enterprise Architecture Committee which is responsible for maintaining the organization"s architecture and for initiating and prioritizing all changes in the architecture. This isn"t to suggest that this group must create the architecture, but only that they must pull all the pieces together and maintain them so that anyone else can access the architecture. They receive inputs from two groups. First, they receive inputs from the strategy committee when they decide that the organization needs to set new strategies or change existing goals. At the same time, the Enterprise Architecture Committee receives inputs from a variety of line managers when they decide that processes they manage aren"t performing properly.

Gartner (2008b: 8) recognizes the importance of an adequate governance structure. They have found that in many organizations, governance initiatives and decisions are defined without a link or even knowledge of EA, and vice versa. IT management is working on IT governance

and on partnering with people in the business on IT projects supported by a governance model. Meanwhile, enterprise architects may be separately focused on EA processes and practices, and on engaging business leaders to ensure that the evolving EA reflects and supports the business strategies and goals. The result is that diverse roles and responsibilities are often misunderstood, resulting in significant overlaps, poor investment decisions, wasted resources and miscommunications.

Dziewulski, *et al*. (2003: 4) suggest that development of a successful EA requires active participation by both the agency business units and Information Management/Information Technology (IM/IT) organizations. An effective, visible executive sponsorship of the EA program goes a long way towards ensuring that an agency commits the right level and type of resources to conduct a successful EA program. The authors (Dziewulski, *et al*., 2003: 4) believe that after developing the target enterprise architecture, it is important to develop a sequencing plan which ties the architecture into Capital Planning Investment Control (CPIC) Process. This structure will ensure that IT is tied directly to business objectives and is managed effectively.

2.2.1.5 Communication

According to Gartner (2007: 5), EA teams continue to under invest in communication planning efforts. Organizations know that it needs to get done, and is critical for success, but in some cases the execution does not match the plan, and indeed, the plan does not match the intent to communicate. Communication is often highlighted as the most time-consuming area, particularly for the chief architect. Organizations undertaking an EA effort that engage in serious communication regularly report greater success and higher maturity in EA.

Effective communication plans should emphasize promoting and publishing success stories throughout the organization. This is particularly important during the startup phase of an EA program, when participants struggle to find direction, make progress, and above all, to see results. Effective communication will result in a greater chance of organizational buy-in and support (Dziewulski, *et al*., 2003: 9).

Dziewulski, *et al*. (2003: 9) add that in order for the EA process design objectives to be implemented properly, a quality assurance program including independent verification and validation (IV&V) processes must be incorporated into the EA process as early as the initial planning stage. The criteria for goodness and quality should be understood as well as the

targeted deliverables, models, repositories and artifacts. Reviewing these ultimately will be the means for IV&V of the architecture providing necessary feedback that will uncover the success or failure or EA process activities.

It is clear from the literature on the process of EA that while various authors have slightly differing views on the process of EA in terms of when certain activities are to be done as well as the names for the different stages, there are some common themes that emerge. Despite the slight differences in approaches, the literature leads me to identify that the process of EA consists of an:

- EA Foundation Stage which occurs at the beginning of the EA process and in which the organization must perform a needs analysis and EA cost benefit analysis. This should help them establish to what extent EA will assist them in solving any problems and/or assist them in achieving their goals. It is important that the organization specifically define their goals as well as how they intend to implement EA and the expected results. The organization should also engage in communication planning at this stage in order to set up effective communication structures that will be used throughout the EA process. Based on these conclusions, it will be an advantage to obtain the top management‟s buy-in given indications from a business case (Bernard, 2004; Dziewulski, *et al*., 2003: 7; Gartner, 2008b: 4; Hansen, 2006: 23).

- EA Approach Stage in which if an organization decides to commence on an EA program based on conclusions from the foundation stage, the top management should set-up a project sponsor(s) or thought leader(s) and an executive body for the EA project (Dziewulski, *et al*., 2003: 15; Hansen, 2006: 65). In this way, top management is involved in the high-level decisions that need to be taken (Hansen, 2006: 65). Thereafter, architecting activities may be conducted by the EA team. This team should be lead and governed by the thought leaders and the executive body (Dziewulski, *et al*., 2003: 15; Hansen, 2006: 65).

- EA Governance (EAG) and Management Stage which focuses directly on establishing and implementing EA in an organization as well as on setting the policy for how EA subsequently should be run. EAG defines the systems, structures and responsibilities within which the EA and affected people must operate (Hansen, 2006: 35). This stage

occurs shortly after the election of the project sponsor (s) and executive body and continues throughout the process of EA.

- EA Maturity and Measurement Stage in which the current maturity of the organization and the value added by the EA project is measured after the completion of a project iteration for a feature that can be tested and thereby evaluating its justification (Schekkerman, 2005). Good intentions and a good start are not measures of success. What matters in the end is completion that delivers performance and results (Hansen, 2005: 42).

- EA Extension and Maintenance Stage in which once the initial enterprise architecture is established, the EA team should proceed to extend and maintain the architecture (Harmon, 2003: 12). The ChiefInformationOfficerCouncil (2001) believes that the primary purpose of an EA is to inform, guide, and constrain the decisions for the enterprise, especially those related to IT investments. The true challenge of enterprise engineering is to maintain the architecture as a primary authoritative resource for enterprise IT planning.

2.2.2 The Role of the Enterprise Architect and the EA Support Team

This subsection of current state of enterprise architecture describes who is involved in the process of EA as well as their role.

The Enterprise Architect

Anderson and Backhouse (2008: 2) view EA as a high-level, strategic process and approach designed to help senior managers deal with complexity and plan for change in the enterprise. The use of the word „architecture" conjures up images of construction of beautiful buildings, and this is deliberate. Like that of the traditional architect, the office of the enterprise architect is replete with blueprints, views, plans, drawings and models. But the edifice under construction is not a physical building, but the far more intangible construction that is the enterprise; the socio-technical make-up of an entire organization.

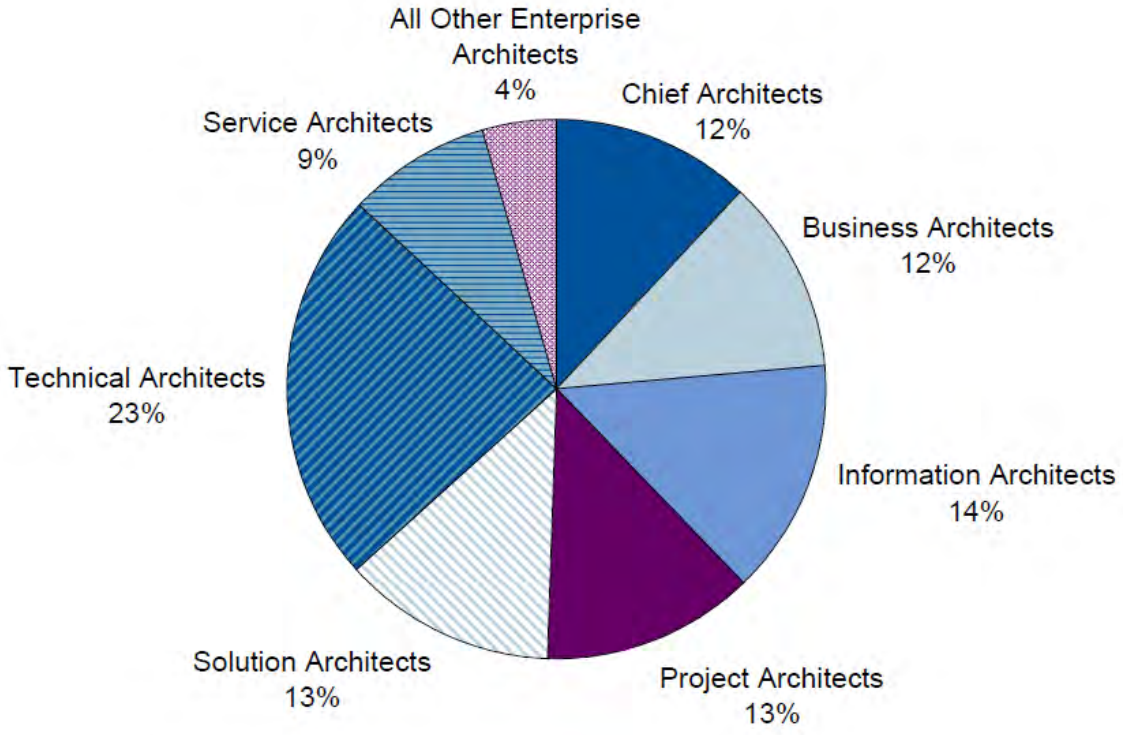
The Gartner (2008:2) definition has adopted an "aspect-oriented" approach, which proposes that the architecture is a collection of viewpoints, including business, technology, and information.

According to Anderson and Backhouse (2008: 2), the architect‟s role is to analyze how it all fits together, to communicate models and views that fit and to ask whether Information and Communications Technology (ICT) infrastructure and business processes work together holistically to deliver an organization‟s strategic plans and overall mission. Architects operate at the level where business objectives, operations, processes and management interact with information systems and ICT.

The viewpoints articulated in the aspect-oriented approach relate to different types of architectures developed by the respective architects specialized in that particular area. For example, using information provided within and external to the organization, the business architect initiates new ventures or leads business innovation. This involves designing a business architecture or business model, which communicates to the rest of the organization a sustainable balanced business system for lasting success. The effectiveness of information assets is leveraged by the creation and management of information architecture by information architects. Technical architects ensure that the infrastructure needs, such as the technical environment, the software environment, as well as the networking within a company are met. The solution architect facilitates the process in which each of these viewpoints is further intersected to form the organization's solution architecture; the guiding framework for implementing future-state projects that close the gap between the current and desired future state (Gartner, 2008b: 5).

Figure 5 on the next page illustrates research conducted by Gartner (2007) into the composition of EA teams and shows that a variety of enterprise architects, specialized in different areas are required in an EA team to ensure that an organization‟s EA is created and managed effectively.

Figure 5: Distribution of Architects by Type



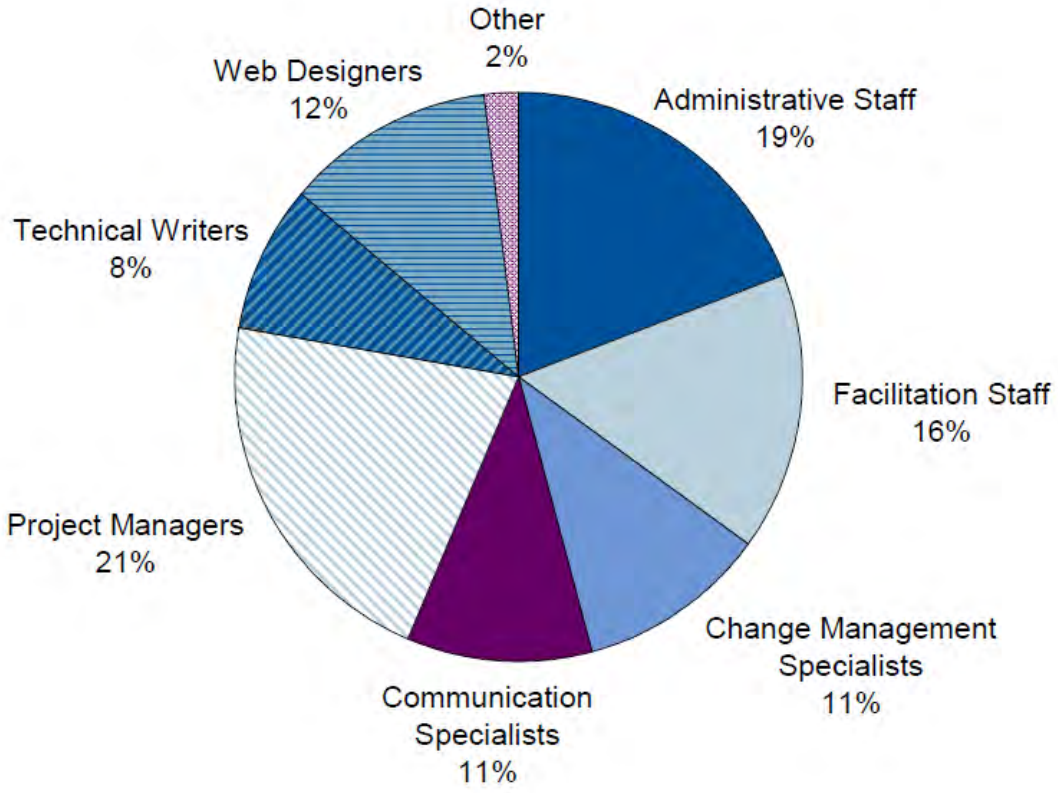
(Gartner: 2007: 7)

In essence, the enterprise architects attempt to capture the essentials of a business and answer the question as to whether the organization‟s information technology is in the parlance of another age, a smooth, well-oiled machine (Anderson and Backhouse, 2008: 2).

The EA Support Team

In order for the activities of the EA team to be successful, they need to be supported by other members of the organization. Figure 6 on the next page shows the results of research conducted by Gartner (2007) into the composition of the EA support team.

Figure 6: Distribution of EA Team Support Staff



(Gartner: 2007: 8)

People often confuse the process of enterprise architecture with the role of the enterprise architect. This leads to a great deal of confusion, because the process as illustrated by Figure 6 is far-reaching in scope and, when done properly, touches everything in the enterprise (Gartner: 2008a: 3)

If one confuses the role of the architect with the process, then one might believe that the enterprise architecture team is trying to own all the strategic decision making of the enterprise. In reality, enterprise architects (who typically drive the EA program) are facilitators and do not have decision rights over all the critical issues of the enterprise (Gartner: 2008a: 3).

2.2.3 The Benefits of Successful EA

This subsection of the current state of enterprise architecture describes the benefits of successful EA which explain why an organization would embark on an EA program.

EA should be an actionable, practical process and approach, facilitated by skilled enterprise architects (Gartner, 2008b: 5). Sessions (2006: 3) believes that when EA works well it is a

tremendous asset that results in many benefits including decreased costs, increased revenues, improved processes and expanded business opportunities.

Hansen (2006: 9) and Shuja (2010: 3) in their research describe how a successful EA program will dynamically guide an organization‟s IT to support the business thus reaching for a common goal in which business, information, data, and technology are integrated, which will in turn provide a positive shift in the organization‟s position.

This is not just a one-shot change, but continuous, sustainable change. For example, it is not enough just to clean up the technical chaos that has grown up over the years; EA must help prevent the chaos from returning. It is not enough to simplify an application portfolio in a year; EA must help maintain simplicity in future years. (Gartner, 2008b: 4).

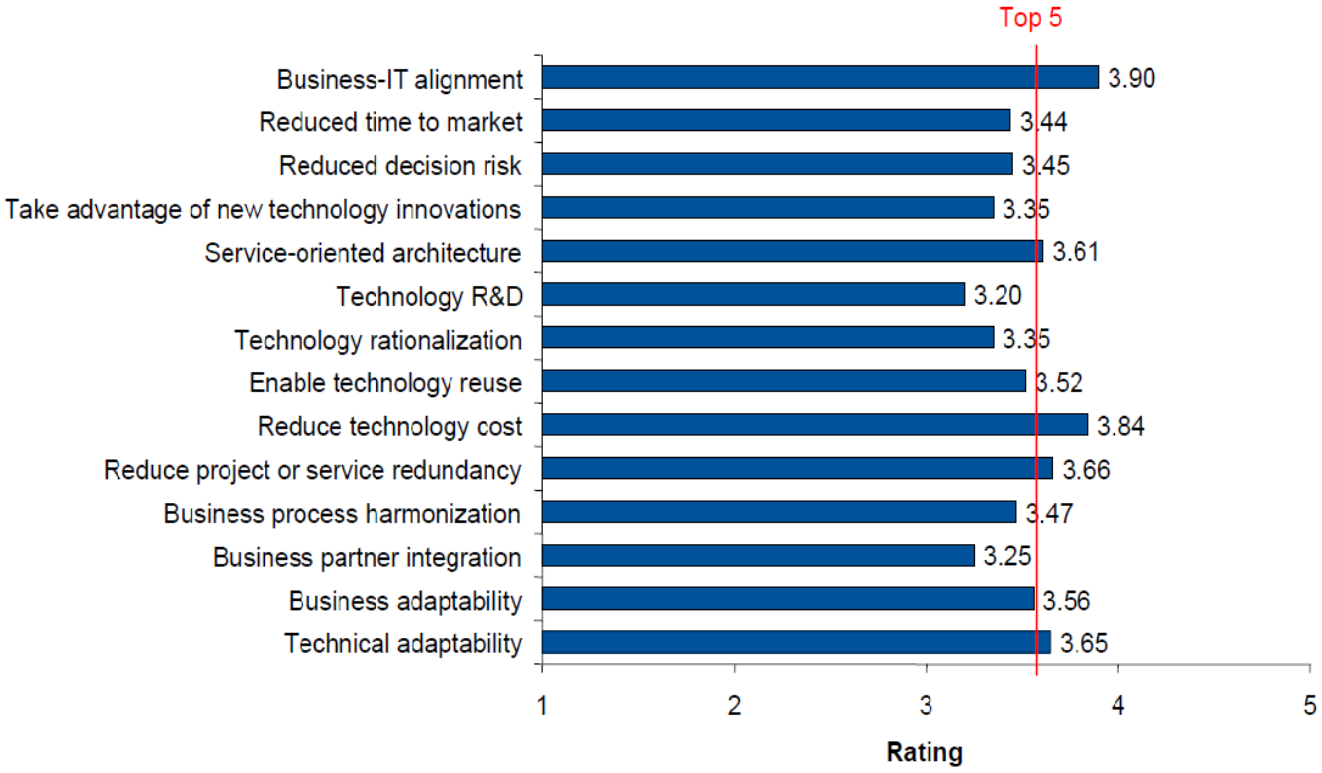
The ChiefInformationOfficerCouncil (2001) believes that EA offers tangible benefits to the enterprise and those responsible for evolving the enterprise. They present a more detailed list of benefits of EA stating that EA can:

- Capture facts about the mission, functions, and business foundation in an understandable manner to promote better planning and decision making.
- Improve communication among the business organizations and IT organizations within the enterprise through a standardized vocabulary.
- Provide architectural views that help communicate the complexity of large systems and facilitate management of extensive, complex environments.
- Focus on the strategic use of emerging technologies to better manage the enterprise‟s information and consistently insert those technologies into the enterprise.
- Improve consistency, accuracy, timeliness, integrity, quality, availability, access, and sharing of IT-managed information across the enterprise.
- Support the process of selecting appropriate EA tools by providing a tool for assessment of benefits, impacts, and capital investment measurements and supporting analyses of alternatives, risks, and tradeoffs.
- Highlight opportunities for building greater quality and flexibility into applications without increasing cost.
- Achieve economies of scale by providing mechanisms for sharing services across the enterprise.
- Expedite integration of legacy, migration, and new systems.

- Ensure legal and regulatory compliance.

As shown by Figure 7 below, enterprises undertake the process of enterprise architecture for a variety of reasons. In 2006, an analysis of more than 200 sets of EA program maturity data identified business context as a critical component of enterprise architecture. Additionally, business-IT alignment was cited as the top driver for enterprise architecture. Business context is the starting point for using enterprise architecture to align business and IT (Gartner, 2007: 3).

Figure 7: Primary Drivers for Enterprise Architecture



(Gartner, 2007: 4)

As shown above, some want to reduce technology diversity; others want to simplify a complex application landscape. Still others want to improve time to market for new products, or to take a more strategic approach to a transformation initiative (Gartner: 2007: 4).

Whatever the reason for initiating an enterprise architecture program, there is one common objective: change. The enterprise wants to perform better by doing things differently, and it expects the EA program to effect that change. All the future-state models, principles and road maps will be for naught unless they are actually implemented. This requires a robust

governance mechanism that will ensure that EA guidance is followed and that there is strong integration with the IT strategy, enterprise program management and portfolio management functions, in order to ensure that common strategic goals are shared (Gartner, 2008b: 6).

As part of this thesis work involves developing a comprehensive definition for EA, EA is defined as the process of translating business vision and strategy into effective enterprise change by creating, communicating and improving the key requirements, principles and models that describe the enterprise's future state and enable its evolution. The scope of the enterprise architecture includes the people, processes, information and technology of the enterprise, and their relationships to one another and to the external environment. Enterprise architects compose holistic solutions that address the business challenges of the enterprise and support the governance needed to implement them (Gartner, 2008a: 2).

2.3 Conclusion

There are numerous and conflicting definitions for the term „enterprise architecture". Varying views lead to a lack of unified knowledge, presence of varying cultures, and policies that prevent efficient management of enterprise processes, innovation and building software (Shuja, 2010: 1). EA is both a management program consisting of phases that overlap and a documentation method supported by an EA framework that together provides an actionable, coordinated view of an enterprise"s strategic direction, business processes, information flows, and resource utilization (Bernard, 2004: 33).

There is no such thing as finished EA, as the requirements of the business will continue to evolve. Instead, the EA is seen as a living set of documents that guides the use of technology. It is more analogous to a city plan than to a building blueprint (Sessions, 2006: 5).

But perhaps even more important than the particular method or framework is the role of the architect. As EA continually evolves, EA is not a project that is ever completed. Leading-edge enterprise architects aided by an organization support team respond to changing requirements (business, people, process, information and technology issues), the realization that the future state becomes a new current state and update EA content and plans accordingly (Anderson and Backhouse, 2008: 2; Gartner, 2008b: 10).

Although there are a number of reasons why an organization would embark on an EA effort, Figure 7 showed business-IT alignment as the top driver for EA. This is hardly surprising, as

organizations rely heavily on IT for their business functions. In an ever-changing business environment, there is a need for organizations to sense environmental change and respond accordingly (Stevenson, 1995). Fluctuations in customer preferences, regulatory compliance, information needs, etc. require an organization to respond in a timely fashion if it is to stay ahead of its competitors. In fact, the current hostile business environment is such that the ability for a company to respond to changing market conditions promptly is now necessary for its survival (Kutnick, 2006).

This chapter provided a comprehensive definition of EA, describing how it is currently practiced, who is involved and why an organization would embark on an EA effort.

Edwards (2008: 3) stresses the importance of an organization to overcome the problem of not being effective because they are reactive or "fire-fighting" instead of being pro-active and organized. He suggests that the solution to this is to be adaptive; as the business changes to meet its needs, so IT should change to support the business; IT should be aligned with business.

Nair, (2008) highlights that the ability to react to upcoming changes effectively and efficiently, under control and in an organized manner is a concept known as „agility," which will be the subject of discussion in the next chapter.

Chapter 3: Agility

Chapter two explored EA; this involved developing a consistent definition for the term, „enterprise architecture", discovering how it is currently practiced, who is involved in the process and the benefits of successful practice.

The aim of this chapter is to provide a fundamental overview of agility.

A broad view of the topic will be taken; uncovering the meaning of agility which will provide a scope of the topic and why it is emerging as a critical topic. The chapter will end with a summary of the findings from the agility research.

It was concluded that agility is the ability for an organization to sense environmental change and respond effectively and swiftly to that change. It is important to have a comprehensive definition for agility which everyone within the organization understands. Agility can be introduced in both the end product produced by an organization and also the process that results in the product. Agility is a characteristic that organizations are required to possess in order to survive in today"s volatile environment.

3.1 Introduction

This section presents an overview of the definitions of agility. The section ends with the presentation of a comprehensive definition that encompasses all the definitions provided in the review and that will be used throughout this thesis.

Lin*, et al*. (2005: 353) have found change to be one of enterprises" major characteristics in this new competitive era. Modern organizations are compelled to react quickly in accordance with the kind of dynamic demands of the customers, which is referred to as agility (Vinodh, *et al*., 2010: 809).

According to Christian *et al*. (2001), some traditional companies already have elements of agility because the realities of a competitive environment dictate these changes (e.g. in sectors such as automobiles, food, textiles, chemicals, precision engineering and general engineering). This is however, usually outside any strategic vision and is approached in an ad-hoc fashion. The lack of a systematic approach to agility does not allow companies to develop the necessary proficiency in change, a prerequisite for agility (Lin *et al*., 2006).

Lin, *et al*. (2005: 354) believe that in embracing agile enterprise, there are important questions to be asked, "What precisely is agility? How can one assist in achieving and enhancing agility effectively?" Answers to such questions are critical to the practitioners and to the theory of agile enterprise design.

Research conducted by Gartner (2006: 2) has established that the term „agility" lacks a consistent meaning across the IT industry. For every person who believes it is simply a term associated with new ways to develop systems (for example, agile methods), there is another who sees it as the label for successful supply chain management. Allen (2009) agrees, adding that often, the term „agility" is bandied about, without real meaning. In his agility research; Kidd (2000) highlights that agility is a confusing area and refers to the current definitions for agility as, "fuzzy." Lin*, et al*. (2005: 353) and Tsourveloudis and Valavanis (2002: 329) refer to the present agility research definitions and measurement techniques as ill-defined and vague. The industry talks of creating, enhancing, improving and leveraging agility as if all those involved understand its meaning and how to go about the process. This is clearly false. The IT industry does not collectively agree on what agility means, nor does it know the best way to go about the process of becoming agile. This is the problem when IT adopts a word that already connotes and denotes a world of pre-existing images of what it means to be agile: dancers, athletes, wild animals, watchmakers, surgeons and so on. It is not shocking that enterprises want to be agile. What is shocking is that very few of them know what it means to be agile. One can imagine a meeting where agility is exhorted as the next big goal, and where the meeting participants run in ten different directions. Some want to investigate new development approaches, others want to tune the supply chain and the majority simply recall childhood memories of gazelles leaping on the African plains as their only anchor points to this critical topic. Without definition, coordination and a shared vision, exhorting IT to be agile is an exercise in futility (Gartner, 2006: 2).

It can be concluded from the literature above that the first challenge for agility is definitional.

Nair (2008) provides a general description of agility; stating that it is the ability to react to upcoming changes effectively and efficiently.

Agility is more formally defined in a business context by Tsourveloudis and Valavanis (2002: 330) as the ability of an enterprise to operate profitably in a rapidly changing and

continuously fragmenting global market environment by producing high-quality, high-performance, customer-configured goods and/or services.

Research conducted by Gartner (2006: 2) into the meaning of agility in an IT context is synonymous with the above definition, adding that an organization is able to operate profitably under such unstable conditions by sensing environmental change and responding efficiently and effectively to that change.

The above definitions give the impression that agility can be achieved by only focusing on improving the operations of the organization. However, the agility of an organization also has very much to do with the people within the organization (Seitz, 2008: 10).

Tsourveloudis and Valavanis (2002: 330) agree that with respect to agility; people have a large role to play in an organization; and add that agility is the outcome of not only technological achievement, advanced organizational and managerial structure and practice, but also a product of human abilities, skills, and motivations. The latter is one of the main differences between *agility* and *flexibility* in the business context.

The authors (Tsourveloudis and Valavanis, 2002: 330) go on to explain the difference between agility and flexibility; In manufacturing terms, flexibility refers to product(s) range using certain (production) strategies, while agility refers to quick movement (change) of the whole enterprise in a certain direction. Flexibility normally refers to the capabilities of a factory floor to rapidly change from one task or from one production route to another, including the ability to change from one situation to another, with each situation not always defined ahead of time. Agility refers to the strategic ability of an enterprise to adapt and accommodate quickly unplanned and sudden changes in market opportunities and pressures, thus, in this sense it is wider than flexibility.
It has also been observed by Madni (2008) that agility is something that can be and in some instances is introduced in both the products that are manufactured or services provided and the processes that an organization executes to produce these products and or services.

A discussion about process and product agility occurs below.

3.1.1Process versus Product Agility

This subsection of the overview of agility highlights the difference between process and product agility, explaining how the two can exist separately or at the same time.

According to Madni (2008: 50), agility needs to be introduced judiciously because it always comes at a cost, i.e. increased complexity. Therefore, introducing it in the process and/or product should be with good justification. An agile process embraces change as a natural consequence of innovative work. It keeps options open as late as possible in the product lifecycle to exploit opportunities and breakthroughs without having to pay a steep price. It is especially advantageous in those circumstances in which significant uncertainties exist during product development as a result of factors such as changing customer requirements, immature technologies or immature manufacturing processes. The key benefit of an agile process is that it affords the opportunity to rethink/modify solutions and concepts in order to exploit new developments and findings. With proper up front infrastructure investment, it also enables cost-effective manufacturing and delivery of the right product regardless of build volume.

An agile product, on the other hand, can respond to unexpected changes in the environment after being deployed through dynamic restructuring/reconfiguration. Such reconfiguration is often required to adapt to changing user requirements, new regulatory measures, or new competitors or threats. An agile product is appropriate in those circumstances in which not all uncertainties can be resolved prior to product deployment. As a result, the system/product needs to rapidly and cost-effectively respond to changes in the environment or capitalize on fast moving opportunities during operation. Typical sources of uncertainty that require such adaptation are changes in mission/operational environment and new customer or user requirements. The key advantages of an agile product are the ability to rapidly and cost effectively adapt to or exploit changes in customer/mission demands, or respond to uncertainties and unexpected conditions in the operational environment after being deployed (Madni, 2008: 50).

Figure 8 on the next page summarizes the differences between process and product agility.

Figure 8: Process agility versus product agility

Comparison Factors	Process Agility	Product Agility
Emphasis	accommodate and exploit vital new information as late as possible during development and even after deployment	respond effectively to changes in customer requirements or mission demands after being fielded
Defining Characteristic	flexibility incorporated in the development process for rapid, cost-effective adaptation	system can cost-effectively change/be changed in operational setting
Uncertainty	most uncertainty resolved prior to product release; however, process remains flexible throughout lifecycle (e.g., five years after deployment; it turns out that custom components can be replaced with COTS)	substantial (e.g., demand evolution, changes in customer functional requirements); needs to be addressed during product operations
When Needed	changing customer criteria, immature technologies, low-to-medium manufacturing readiness levels	long lifecycle (>10 years), significant replacement costs (i.e., to build a new system)
Penalty	increase in process complexity	increase in product complexity, introduction of additional interfaces, higher costs

(Madni, 2008: 52)

The literature (Gartner, 2006: 2; Lin, *et al*. 2005: 354) stressed the importance of there being a comprehensive definition for agility which encompassed what precisely agility is, as well as gave insight into how one can assist in achieving and enhancing agility effectively.

From the literature presented, agility is defined within the context of this research as the ability of an enterprise to operate profitably in a rapidly changing and continuously fragmenting global environment by producing high quality, high performance customer-configured goods and/or services and by sensing environmental change and responding efficiently and effectively to that change. It is the outcome of technological achievement, advanced organizational and managerial structure and practice as well as a product of human abilities, skills and motivations (Gartner, 2006: 2; Tsourveloudis and Valavanis, 2002: 330).

This is an ample definition to start the discourse, but a concept such as agility needs flesh beyond the bare-bones definition. The remainder of the agility research explains why it is emerging as a critical topic (Gartner, 2006: 2).

3.2 Why Agility is emerging as a Critical Topic

The previous section presented a comprehensive definition for agility and explained the difference between process and product agility. This section explains why agility is emerging as a critical topic.

The need for agility for competitiveness has traditionally been associated with the supply chains that provide and manufacture innovative products, such as high-technology industry products characterized by shortened life-cycles, a high degree of market volatility, uncertainty in demand, and unreliability in supply. Similarly, traditional, more slow-moving industries face such challenges in terms of requirements for speed, flexibility, increased product diversity and customization. Consequently, the need for agility is becoming more prevalent (Jain, *et al*., 2008: 367).Agile enterprise whereby an organization can change and adapt quickly to changing circumstances is increasingly viewed as a winning strategy (Lin, *et al*., 2005: 353).

Edwards (2008: 3) highlights the importance of an organization to overcome the problem of not being effective because they are reactive or "fire-fighting" instead of being pro-active and organized. He suggests that the solution to this is to be adaptive (agile) and not predictive, under control and in an organized manner.

Seitz (2008: 16) agrees that organizations should adopt an agile approach; mentioning in his research that a key business driver for organizations is responding quickly to changing market conditions. While agility gives companies an important competitive advantage during stable market conditions, it becomes essential during periods of volatility and uncertainty. Agility also allows them to transform their businesses into a single agile entity capable of extending its leadership position and global footprint.

Prahalad (2009: 80) adds that over the years managers have developed tools and techniques to overcome challenges ranging from inconsistent quality to stagnant productivity. Now what they need is a system for addressing volatility. How does a chemical company, for example, cope with oil prices that bounce from $50 a barrel to $150 and back in 18 months?

As market globalization raises competitive pressures worldwide, one essential requirement for enterprise survival is continuous ability to meet customer needs and demands. Market needs cause unceasing changes in product(s) life cycle, shape, quality, and price (Tsourveloudis and

Valavanis, 2002: 329). Changes in consumer sentiment, such as "going green" or shunning conspicuous consumption, can rewrite established patterns of demand in a remarkably short time. Volatility is here to stay. This forces managers to harmonize two critical capabilities. On the one hand, strategic clarity and consistency; on the other, agility and resilience in operations. This may seem counterintuitive, but organizations can handle extreme change only when they can address it within a clear strategic framework. Otherwise, companies can only wait and react (Prahalad, 2009: 80).

According to Lin *et al*. (2005: 353) many companies are facing constantly increasing competition stimulated by technological innovations, changing market environments and changing customer demands. This critical situation has led to a major revision in business priorities, strategic vision, and in the viability of conventional and even relatively contemporary models.

Some aspects of managing in a volatile environment, such as focusing on operational efficiency and staying liquid are givens. But more important are the abilities to scale up and down and reconfigure resources rapidly. Here, there are lessons to draw from volatility-prone India, with its shifting regulations, spurts of growth, capital shortages, and challenging supply base (Prahalad, 2009: 80). Within the software development context, research conducted by Schmidt, Lyytinen, Keil, and Cule (2001) highlights that the unprecedented rate of change in business and technology has made it increasingly difficult for software teams to determine user requirements and respond to their changes.

Scaling up and down quickly requires a renewed focus on the breakeven point of businesses. Can a company break even at 30% to 40% of capacity utilization? If the answer becomes yes, this creates a lot of room for dealing with shifts in demand. But to get to that point, managers have to focus on capital intensity and fixed costs (Prahalad, 2009: 80).

Not long ago, Airtel, India's largest telecom carrier, with 100 million subscribers, saw opportunities for growth but was starved for cash. Typically, telecom carriers invest in capacity, but Airtel outsourced IT management to IBM and leased network capacity from Ericsson and Nokia. By paying for only the capacity it needed, Airtel converted fixed costs into variable costs. So the company shrank its capital intensity and gained the ability to scale up rapidly. It can also scale down without the penalties a capital-intensive telecom carrier

would face. Airtel may now be the world's lowest-priced carrier (at less than 2 cents a minute) and is one of the most profitable (Prahalad, 2009: 80).

Indian IT providers have to hit a number of moving targets. As customer requirements change, the mix of skills needed to serve them changes, too. These companies must also expand the range of services they offer to capture emerging business opportunities. Further, they have to juggle an overlapping series of projects of differing durations. Many of these companies have found ways to handle all this flux without painful restructuring (Prahalad, 2009: 80).

One of their rules of thumb guards against sudden shifts in demand: Don't allow a single customer to contribute more than 5% of revenues. Many IT companies make it a goal to draw clients from a variety of sectors. Managing their client portfolios this way offers a first level of protection (Prahalad, 2009: 80).

To better handle the constant project turnover, employees are cross-trained in many different skills. This requires an arsenal of training programs. Employees are regularly tested, and the hallmark of the best of them is the ability to learn quickly (Prahalad, 2009: 80).

Having this much flexibility in staff, and within each staffer, forces these companies to equip their managers with instant access to data on what each employee can do and where they are physically and in terms of the finish date of their current assignment. All employees know they will be moved from one assignment to another; and in many cases across the world. It becomes the cultural expectation (Prahalad, 2009: 80).

Lee and Xia (2010) believe that the same is true in the software development environment. As business and technology environments change at an unprecedented rate, software development agility to respond to changing user requirements has become increasingly critical for software development performance. Agile software development approaches, which emphasize sense-and-respond, self-organization, cross-functional teams, and continuous adaptation, have been adopted by an increasing number of organizations to improve their software development agility.

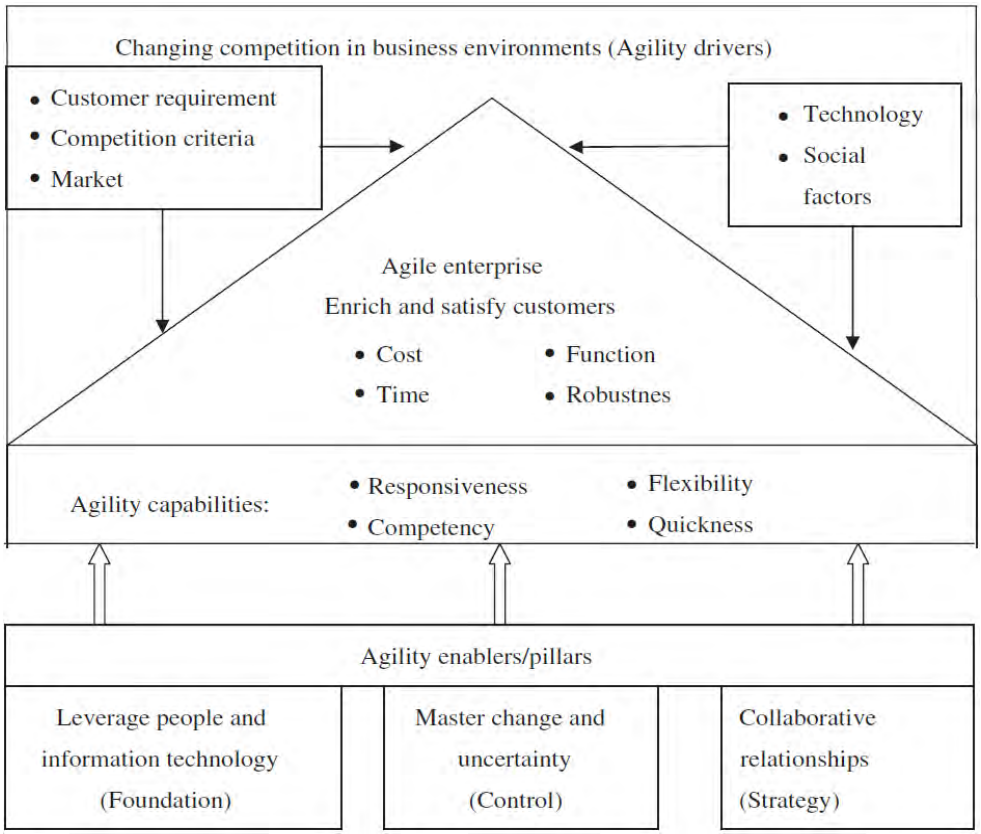
In this volatile world, more and more companies will strive to become "Velcro organizations" in which people and capacity can be rearranged and recombined creatively and quickly

without major structural change. The winners won't stop focusing on quality, cost, and efficiency, but they'll be paying a lot more attention to agility, too (Prahalad, 2009: 80).

In an increasingly competitive market, there is a need to develop and improve organizational flexibility and responsiveness. In the past decade, most companies adopted restructuring and re-engineering in response to challenges and demands; however, these were not always successful. Agile enterprise addresses new ways of running companies to react quickly and effectively to changing markets, driven by customized products and services. Agile enterprise in general can provide lower manufacturing costs, increase market share, satisfy customer requirements, facilitate the rapid introduction of new products, eliminate non-value added activities and increase company‟s competitiveness. Thus, agile enterprise has been advocated as the 21st century‟s enterprise paradigm, and is seen as the winning strategy to become national and international leaders in an ever increasing competitive market of fast changing customer requirements. However, the ability to build agile enterprise has not developed as rapidly as anticipated, because the development of technology to manage agile enterprise is still under way. Even in the software development community, a common cause of disaster is that the end product is precisely what the customer originally ordered. In a world moving at internet speed, a customer's objectives are constantly being revised, so programmers have to be able to hit a moving target (Lin *et al*., 2005: 354).

The purpose of agile enterprise is to enrich/ satisfy customers and employees. An enterprise essentially possesses a set of capabilities so as to make appropriate responses to changes taking place in its business environment. However, the business conditions in which many companies find themselves are characterized by volatile and unpredictable demand; thus, the increasing urgency of pursuing agility. To be truly agile, an enterprise should possess a number of distinguishing agile enablers. From a review of the normative literature, Lin *et al*. (2005: 355) have developed a conceptual model of agile enterprise, as shown in Figure 9 on the next page, culminating in many research propositions.

Figure 9: Conceptual Model for agile Enterprise



(Lin *et al*., 2005: 356)

The main driving force behind agility is change. Even through change is nothing new; today"s change is taking place at a much faster speed than ever before. Turbulence and uncertainty in the business environment have become the main causes of failure in the manufacturing industry. The number of changes and their type, specification or characteristic cannot be easily determined and are probably indefinite (Lin *et al*., 2005: 355).

It is evident that different enterprises with different characteristics and in different circumstances experience different changes that are specific and perhaps unique to them. But there are common characteristics in changes that occur, which can bring about a general consequence for every enterprise. Lin *et al*. (2005: 356) believe the general areas of change in business environment are categorized as follows:

- Market volatility caused by growth of the niche market, increasing new product introduction and product lifetime shrinkage;

- Intense competition caused by a rapidly changing market, increasing cost pressure, international competitiveness and short development of new products;
- Customer requirements‟ changes caused by demand for customization, quality expectation increase and quicker delivery time;
- Accelerating technological change caused by the introduction of new and efficient production facilities, and system integration (hardware and software); and
- Change in social factors caused by environmental protection, workforce/workplace expectations and legal pressures.

Agile enterprises are concerned about change, uncertainty and unpredictability within their business environment and make appropriate responses. Therefore, agile enterprises require a number of distinguishing capabilities or „„fitness‟‟‟ to deal with the change, uncertainty and unpredictability within their business environment. These capabilities consist of four principle elements:

- Responsiveness, which is the ability to identify changes and respond quickly to them, reactively or proactively, and recover from them;
- Competency, which is the ability to efficiently and effectively reach enterprises‟ aims and goals;
- Flexibility/adaptability, which is the ability to process different processes and achieve different goals with the same facilities; and
- Quickness/speed, which is the ability to carry out activity in the shortest possible time (Lin *et al*., 2005: 356).

Furthermore, underpinning these fours principles should be a methodology to integrate them into a coordinated, interdependent system, and to translate them into strategic competitive capabilities. These must be taken into account if an organization is to carry out agile enterprise. Achieving agile enterprise requires responsiveness in strategies, technologies, people, business processes and facilities. Thus all areas of the company need to have some agility providers to effectively respond to changing market requirements (Lin *et al*., 2005: 356).

As part of this thesis work involves developing a comprehensive definition for agility, agility within the context of this thesis is defined as the ability of an enterprise to operate profitably in a rapidly changing and continuously fragmenting global environment by producing high

quality, high performance customer-configured goods and/or services and by sensing environmental change and responding efficiently and effectively to that change. It is the outcome of technological achievement, advanced organizational and managerial structure and practice as well as a product of human abilities, skills and motivations (Gartner, 2006: 2; Tsourveloudis and Valavanis, 2002: 330).

3.3 Conclusion

It is important for an organization to have a consistent definition for agility as well as a common understanding of what it means to be agile. This will allow them to be in the best position to sense environmental change and respond effectively to that change (Gartner, 2006: 2; Tsourveloudis and Valavanis, 2002: 330). Agility is the ability to react to upcoming changes effectively and efficiently (Nair, 2008). It is the outcome of not only technological achievement, advanced organizational and managerial structure and practice, but also a product of human abilities, skills, and motivations (Tsourveloudis and Valavanis, 2002: 330).

In this volatile world, more and more companies will strive to become "Velcro organizations" in which people and capacity can be rearranged and recombined creatively and quickly without major structural change. The winners won't stop focusing on quality, cost, and efficiency, but they'll be paying a lot more attention to agility, too (Prahalad, 2009: 80).

This chapter provided a fundamental overview of agility. The discussion involved developing a comprehensive definition as well as uncovering the meaning of agility and explained why it is emerging as a critical topic.

Part of this thesis work also involves explaining how agility fits within the context of enterprise architecture (EA). Agility within the context of EA is a relatively new area; therefore it becomes necessary to establish the relationship between EA and agility which will be the subject of discussion in the next chapter.

Chapter 4: Agile Enterprise Architecture

Chapter three explored the concept of agility. This involved developing a comprehensive definition for the term, as well as uncovering why it is emerging as a critical topic.

The aim of this chapter is to introduce „agile enterprise architecture".

The discussion will include the scope of agile enterprise architecture which will present an overview of the topic; explaining how agility fits in the context of EA and developing a comprehensive definition for agile enterprise architecture, the challenges with current methods for developing EA which favour an agile approach, as well as how agile software development principles can be applied to EA to ensure an agile EA. Thereafter, the best practices in agile architecture will be discussed; highlighting how agile EA can be developed as well as the role of the enterprise architect in an agile EA environment showing how different it is in comparison to traditional EA practice. The chapter will end with a summary of the findings from the literature.

It was concluded that the volatile environment in which companies find themselves has spurred the need for a more agile approach to developing EA. Agility within the context of EA is a relatively new area and this leads to a lack of understanding of how it can be achieved. There is therefore a need for a comprehensive and consistent definition for agile EA within an organization. As EA is a process, in order to ensure that agile EA is developed, agility characteristics should not only be embedded in the end products created by the process of architecting, as is the traditional practice, but also in the process itself. Commenting on agile software development practices from an EA perspective proves that agility can be introduced into the process of EA. There are also agile architecture principles which an organization can adhere to when developing their EA. These principles do not recognize the traditional role of the systems architect. Since agile focuses on harnessing the power of the collective team, rather than any one individual, the system architect no longer dictates technical direction. While an agile approach to EA comes with it many advantages, there are also some disadvantages which must not be overlooked.

4.1 Introduction

The aim of this section is to provide an understanding of how agility fits within the context of EA.

Gartner (2008b: 5) asserts that EA should be an actionable, practical process and approach, facilitated by skilled enterprise architects. As the name implies, EA looks after the architecture for the whole enterprise. This includes Business Architecture, Information Services Architecture and Technology Infrastructure Architecture. The products of the EA process are artifacts that describe the organization"s structure and processes.

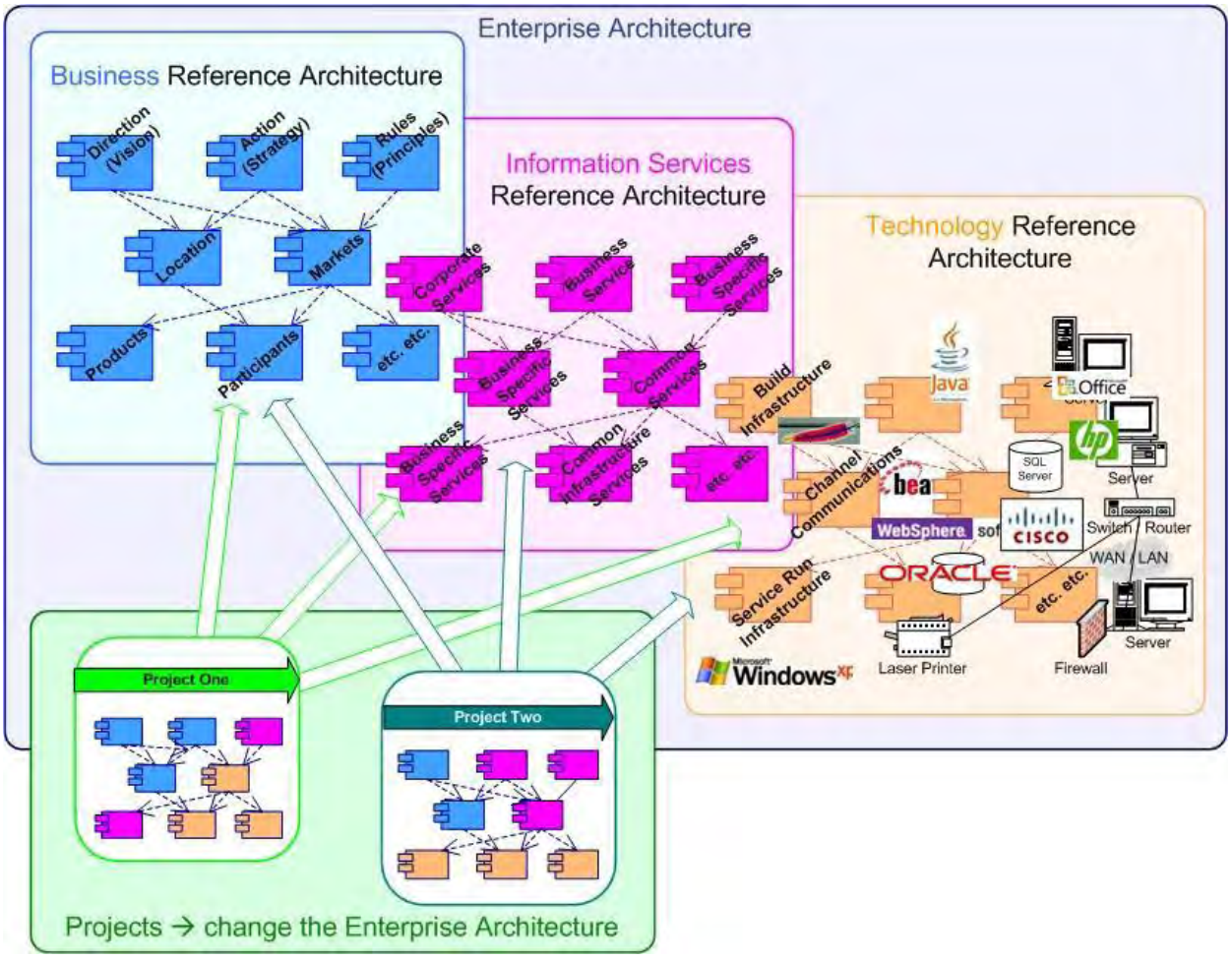
According to Madni (2008: 49), competitive market forces continue to challenge complex systems development initiatives. As such, development processes need to go beyond being just flexible (i.e., easy to change) to being agile. The key characteristics of an agile process are flexibility (i.e., ability to respond to expected change), context-awareness (i.e., ability to know what is happening), efficiency(i.e., short test cycle times, optimal resource utilization) and adaptability(i.e., ability to react rapidly and cost effectively to unexpected change). These characteristics collectively contribute to defining agility as the ability to exploit and thrive in the face of expected or unexpected change.

Stevenson (1995) believes that in this sense, each architecture effort, if done as a part of a fresh full life-cycle project, is meant to deliver exactly this purpose. „Agile Architecture" is the term used to describe an architecture that caters for future unknowns, enabling change to occur rapidly without undue resource utilization, yet in a controlled manner and with minimal adverse impact.

Nair (2008) adds that one needs to know the current state of the architecture in order to be able to assess the impact of a change so much as to know what is going to change, when, where, how and why it changes. Even though any project may deliver the best and most reliable product set for the particular purpose, it may harm the overall enterprise architecture. For example a new project may conflict with some existing business processes, add a new vendor database product into the mix or add another application server instead of using the existing one. This could turn out to be costly and difficult to support and manage in the longer term (Edwards, 2006: 2).

Figure 10 below shows a diagrammatic representation of different architecture domains that could occur within an organization; namely Business, Information and Technology architecture and how a successful project could add or change some sub-set of all the business systems, software and technology architecture in the enterprise (Edwards, 2006: 2).

Figure 10: Enterprise Architecture within the Organization



(Edwards, 2006: 2)

So to have a good architecture is undoubtedly the first step of being agile. In other words, the greatest objective of EA is to assist agility. Going by this logic, one may tend to think that „agile architecture" is badly phrased; if not that it is an absolute oxymoron (Nair, 2008).

However, as signified by Nair (2008), the term „agile architecture" in this instance is used to stress the importance of an architecture that is optimized for changes. So, while capturing the „as-is" architecture is implicitly and arguably agility-driven, defining the „to-be" state is much more than that. In the latter case, one strives to devise an architecture that is to respond to changes effectively and efficiently, rather than to pull the current architecture together to fine-

tune understanding of the system(s) to handle potential changes effectively and efficiently. There should rather be a focus on this future-state modeling when talking about agile architecture. The lack of a systematic approach to agility does not allow companies to develop the necessary proficiency in change, a prerequisite for agility (Lin *et al*., 2006).

In the same vein, Madni (2008: 49) identifies the characteristics of an agile product are extensibility (i.e., ability to add new functionality); modularity (i.e., ability of one or more modules to be substituted without disrupting the integrity of the rest); and reconstitutability (i.e., ability to recombine, or substitute modules in the face of a fault or failure). An agile development process is required when there is a pressing need for both flexibility and speed in upstream processes (e.g., conceptualizing, designing, implementing) in the system‟s lifecycle. An agile process embraces change as a natural consequence of innovative work. It keeps options open as late as possible in the product lifecycle to exploit opportunities and breakthroughs without having to pay a steep price.

An agile end product is required when it is not possible to predict future demand or functional requirements with high confidence and when the product/system is expected to be long-lived. Agility in the end product implies embedding agility characteristics in the end product, which means that the end product or system is architected for extensibility and substitutability, and the processes required to implement such extensions and make such substitutions are also embedded in the system (Madni, 2008: 50).

The above definitions give the impression that agility can be achieved by only focusing on improving the operations of the organization. However, the agility of an organization also has very much to do with the people within the organization (Seitz, 2008: 10). Tsourveloudis and Valavanis (2002: 330) agree that with respect to agility; people have a large role to play in an organization; and add that agility is the outcome of not only technological achievement, advanced organizational and managerial structure and practice, but also a product of human abilities, skills, and motivations

It was established in chapter two, that while some authors view EA as a noun (product), the view taken in this thesis is to view EA as a verb (a process) which results in artifacts (nouns) created. The relationship between agility and EA discussed will therefore be related to the process of EA.

Agile enterprise architecture is therefore defined within the context of this research as the systematic process of adhering to agile development principles while translating business vision and strategy into effective enterprise change by flexibly creating, communicating and improving key requirements, principles and models. The models should have agile characteristics embedded in them, describe the enterprise‟s future state while keeping options open as late as possible and enable its evolution. The scope of agile enterprise architecture includes the people, processes, information and technology of the enterprise, and their relationships to one another and to the external environment. Enterprise architects compose holistic solutions that address the business challenges of the enterprise and support the governance needed to implement them (Gartner, 2008a: 2; Lin *et al*., 2006; Madni, 2008: 50; Tsourveloudis and Valavanis; 2002: 330).

Research conducted by Edwards (2008: 1) indicates that some enterprise architects still use outdated, rigid approaches to EA which are incompatible with today‟s business environment. Their approach is out of a lack of understanding of how agility fits within the context of EA and leads to a number of challenges which are discussed in the next section.

4.2 Challenges with Current Approaches

The previous section introduced the term „agile enterprise architecture,‟ explaining how agility fits within the context of EA and presenting a comprehensive definition. This section highlights the challenges with current (more rigid) approaches to EA and makes the case for agile architecture.

There are many different threads of concern influencing the need for an agile EA. Concerns are looked at from various different viewpoints, but each view considers EA as a whole:

1. **A Predictive approach** – Leffingwell (2007: 1) has seen the potential problem here to be that enterprise architects use a large plan up front, rigid process (predictive), instead of a time boxed, continuous, risk-driven improvement process (adaptive) to manage the EA practice and its output. This approach to EA diminishes the agility provided by the adoption of more agile development methods like SCRUM and XP in software development within the enterprise.
2. **Ill-informed Project teams** - Shirazi, *et al.* (2009: 183) have discovered that in some instances development teams do not know that the EA exists and therefore do not

structure their development efforts in a way that follows the enterprise architecture. This arises in organizations where developers don‟t work with the enterprise architects.

3. **Outdated architecture** - Ambler (2009) concludes that enterprise architects spend a great deal of time trying to cater for every possible event that could occur. A critical concept is that the EA models and documents just need to be good enough; they don‟t need to be perfect. It is naive to assume that perfect artifacts can be produced. Furthermore, even if a development team managed to create perfect artifacts they‟d be out of date the day after they were published because something within the business or technical environment would change.
4. **No Duality of process (lack of separate cycles)**– The traditional project process is slowed down (i.e. making it non-Agile) just to consider all the views EA require. The solution proposal is to split the lifecycles into two separate adaptive process cycles. One for projects and a separate one for EA, both adaptive in manner, both using their own phases and Iterations (Edwards, 2006: 2).
5. **Lack of traceability between EA and Software development** – Edwards (2006: 2) sees a potential problem of multiple projects duplicating work and effort, by taking project time to build up an understanding about what already exists, exactly what will be impacted, external dependencies, etc. If traceability exists from the EA to the projects, then this waste of time could be minimized.

The benefits of agile methods are becoming more obvious and compelling (Leffingwell, *et al.,* 2008: 1).

Giachetti (2009: 39) points out that although EA should be and is intended to be for the entire business, in practice much of the discussion of EA takes place within the software development community.

One might wonder; if Agile is about better and more efficient ways of developing software, why would anyone want to make EA more agile? After all, EA is not directly involved with actually developing software (dependent up on the EA practice); EA is only really concerned about the big picture. It is concerned about what already exists in the enterprise and what should exist once any one particular project is complete. The simple answer is; it‟s the principles that Agile and Adaptive software development stand for, that make sense to follow as an EA set of processes and practice (Edwards, 2006: 3).

Edwards (2006: 3) suggests that these can be explored in terms of the Agile Manifesto and the Agile-Adaptive Project Management Declaration of Independence (DOI); looking at their principles and commenting on them from an EA point of view as to whether the principles work within an EA context. This discussion occurs in the next section.

4.3 Agile Enterprise Architecture vs. Software Development

The previous section discussed the challenges with current approaches to EA making the case for agile EA. This section discusses the principles of The Agile Manifesto and the Agile-Adaptive Project Management Declaration of Independence and comments on them from an EA perspective showing how their principles can be applied to EA and that agility in the process of developing EA has a place in the software development community where much of the discussion of enterprise architecture occurs.

4.3.1 The Agile Manifesto Principles

- **Individuals and interactions over processes and tools**

Edwards (2006: 3) and Fowler and Highsmith (2001) believe that people collaborate in much more of a real-time basis, compared with tools and process, which tend to imply more of a delayed response time. This does not imply that process and tools must not be used, but collaboration and interaction using the people skills should be done first; then the models can be refined using the tools and processes. The most important factors that need to be considered are the people and how they work together because if that is not focused on, the best tools and processes won't be of any use (Ambler, 2006). Sutherland (2010) believes that individuals and interactions are essential to high performing teams. Fowler and Highsmith (2001) add that physical documentation has heft and substance, but the researchers (Fowler and Highsmith, 2001) believe that writing is a difficult and inefficient communication medium. They (Fowler and Highsmith, 2001) use it because they have to, but most project teams can and should use more direct communication techniques.

- **Working „software" over comprehensive documentation**

Any good architect would have a working architecture over comprehensive documentation. This implies being risk driven and attacking the highest risk to the enterprise first. In many companies they do not even have the concept of EA, so they have already defaulted to practicing this principle by fire-fighting; but an enterprise can only take it to the next level of efficiency by reducing complexity with more comprehensive models (Edwards, 2006: 3).

Rosenblatt (2009) agrees, and believes such an approach will deliver value to the customer as soon as possible. The researcher (Rosenblatt, 2009) stresses the importance of creating value as soon as possible by stating that, "Above anything else we do; above code formatting, testing, meetings, planning, sometimes breathing – our number1 goal is valuable software." Fowler and Highsmith (2001) re-iterate that in a practical context this means for example, delivering working software frequently, from a couple of weeks to a couple of months, with a preference for the shorter timescale.

- **Customer collaboration over contract negotiation**

In most cases, it takes time to build up trust and prove the EA practice‟s worth. This can only be done by showing real added value and tangible benefits in monthly increments. No business will sit back and wait for a year while some team of people ferret away building up something they do not understand, costs a lot and appears to offer no value, even if it has been put down in some form of contract (Edwards, 2006: 3).Fowler and Highsmith (2001) explain that many individuals may want to buy software the way they buy a car. They have a list of features in mind, they negotiate a price, and they pay for what they asked for. This simple buying model is appealing, but for most software projects, it doesn't work.

Agile developers should respond with a radical change in their concept of the requirements process. For a start, they shouldn‟t expect a detailed set of requirements to be signed off at the beginning of the project; rather, they should see a high-level view of requirements that is subject to frequent change. Clearly, this is not enough to design and code, so the gap is closed with frequent interaction between the business people and the development team (Fowler and Highsmith, 2001). Ambler (2006) points out that only customer can tell a development team what they require. Over the past two decades, project success rates have more than doubled worldwide. This is attributed to smaller projects and frequent deliveries, which allow the customer to provide feedback on working software at regular intervals (Sutherland, 2010).

- **Responding to change over following a plan**

To be successful the development team will need to be adaptive as people change their priorities for a variety of reasons (Ambler, 2006; Edwards, 2006: 4). It is important to have a plan for the short term (measured in weeks) and stick to it as much as possible, but respond to change within a risk managed environment. This does not mean that there is no big picture plan; it is just not in great detail (Edwards, 2006: 4). Fowler and Highsmith (2001) believe this requires that agile teams must refine and reflect as the project goes along, constantly

improving their practices in their local circumstances. Sutherland (2010) adds that responding to change is essential for creating a product that will please the customer and provide business value.

4.3.2 Agile-Adaptive Project Management Declaration of Independence (DOI)

- **We increase return on investment by making continuous flow of value our focus -** Many executives battle to see the value EA brings to the organization. EA should take on and deliver value as early as possible to ensure its continued existence (Edwards, 2006: 4). Cockburn (2005) suggests focusing on the value that is being created and watching the flow of increase in value. The development team should make the unit of value in the flow small, in some ideal world a single unit, what the manufacturing people call continuous flow (Cockburn, 2005).

- **We deliver reliable results by engaging customers in frequent interactions and shared ownership -** Once the business see the value and share ownership, the results become more reliable as they become engaged and open up (Edwards, 2006: 4). Cockburn (2005) stresses the need for the organisation to strive for shared ownership and for the development team to engage with the customers, checking in with the customers frequently during the development process.

- **We expect uncertainty and manage for it through iterations, anticipation, and adaptation -** Contrary to what is understood today, Fowler and Highsmith (2001) believe that form does not follow function: Form follows failure. The form of made things is always subject to change in response to their real or perceived shortcomings, their failures to function properly. The, "form follows function" idea has misled architects into believing that they could predict how buildings would actually be used, when in fact the best designs emerge from iterative development and use rather than from early plans (Fowler and Highsmith, 2001). Uncertainty is the only certainty in business today, with the possible exception that uncertainty should not stem from poor iteration planning, a culture of fire-fighting and bad management practice, but rather from controlled rapid business change (Edwards, 2006: 4). Highsmith (2009) explains that the ability to respond to change drives competitive advantage. Teams must adapt, but they can't lose track of the ultimate goals of the project. These teams should also constantly evaluate progress, whether adapting or anticipating (Highsmith, 2009).

- **We unleash creativity and innovation by recognizing that individuals are the ultimate source of value, and creating an environment where they can make a difference -** EA teams have people and they should be recognized as the biggest drivers of success on a project (Edwards, 2006: 4; Highsmith, 2009). Software development consists of nothing but invention and communication, the two hallmarks of human individuals (Cockburn, 2005).

- **We boost performance through group accountability for results and shared responsibility for team effectiveness -** EA Practice shares much responsibility in general, the teams are typically small, are far reaching and have the same potential to have performance boosted and be effective (Edwards, 2006: 4).

- **We improve effectiveness and reliability through situationally specific strategies, processes and practices -** The EA team in this sense benefit the organization by taking this concept to the enterprise level, because while project teams each improve effectiveness on a particular project, the EA team manage this concept across all projects, business services, systems, information and technologies to improve effectiveness and reliability for the whole enterprise (Edwards, 2006: 4).

It is clear that agile software development principles can be applied to EA. Agile approaches to EA therefore have a place in the software development community where the discussion about EA mostly occurs. The next section will discuss the current approaches to developing agile architecture.

4.4 Current Best Practices in Agile Architecture Development

The previous section provided a description of how agile software development principles can be applied to EA, showing that agility in the process of developing EA has a place in the software development community where much of the discussion of enterprise architecture occurs. This section discusses the principles associated with agile EA development.

Leffingwell, Martens and Zamora (2008: 3) define a set of governing principles for the development and maintenance of agile, enterprise-class architectures. Most contributions to the world of agile development define a set of key principles, and architecture development should be no exception. The following build on agile software development principles.

However, they have been re-structured, and some revised, to better meet the needs of agile architects.

These principles are:

Principle #1 The teams that code the system design the system

Teams themselves are empowered to define, develop and deliver software, and they are held accountable for the results. From a management perspective, in order for teams to be held accountable, they must be allowed to make the decisions required to support that accountability. If not, they will be held accountable for decisions made by others, and that is an ineffective and de-motivating model for team performance. While this seems axiomatic, responsibility in earlier practice was different, as Table 1 shows (Leffingwell, *et al.,* 2008: 3).

Table 1: Responsibility and accountability in the pre- and post-Agile world

	Pre Agile	Post Agile
Management	• Determines market needs and features • Communicates vision •Product managers determine requirements •Architects determine architecture •Management determines schedule and commits on behalf of team • Accountable for the results	• Determines market needs and features • Communicates vision • Eliminates impediments for the team •Accountable for empowering teams to deliver
Team	• Inherits the plan • Inherits the architecture • Left "holding the bag" and executes on a "best efforts" basis	•Determines the requirements • Determines the architecture • Determines the schedule in terms of iterations and releases • Commits on behalf of themselves • Accountable for the results

(Leffingwell, *et al.,* 2008: 4)

This level of responsibility was further compounded by time-sequenced waterfall activities. Typically, there was a single, up-front planning phase intended to accommodate the inherent risk of the project, perform all the necessary design and predict sequenced-task dependencies,

running out as far as a year. Because it happened up-front, the planning phase was decoupled from the lessons learned later during implementation. Once the lessons were learned, it was too late to do anything to impact the schedule, except to apologize for the fact that the schedule was not met. From a technical perspective, architectural decisions are most optimally made by the coders, the technical leads and team-based architects because they are closest to the implementation and often have the best data available to make such a decision (Leffingwell, *et al.,* 2008: 4).

Ambler (2010) adds that the architecture must be based on requirements. Practicing active stakeholder participation is critical to success when it comes to identifying architectural requirements. Requirements come from project stakeholders, not developers. Good sources for technical architecture requirements will include the users and their direct management as they will often have some insight into technical requirements and constraints. Operations staff will also have requirements pertaining to the deployment architecture. The best sources for business-oriented requirements are the users and their managers. Senior management within the organization will have insights that may lead to potential change cases for the system.

In order not to slow project progress, the development team should create several architecture models in parallel. When working on the technical aspects of the architecture they should base it on technical requirements, constraints, and possibly change cases. Similarly, when working on business aspects of the architecture, potentially identifying software subsystems or business components, they will likely need to focus on essential use cases or user stories that describe critical usage requirements and potentially the key business rules applicable to the system (Ambler, 2010).

A common mistake that architecture teams (or for smaller projects the architecture owner) will make is to ignore existing and pertinent artifacts, such as network or deployment diagrams that describe the organization‟s existing technical infrastructure, enterprise-level business models (use case models, process diagrams, workflow diagrams, corporate business rules, and so on), or corporate deployment standards (for workstations, branch offices, etc.) that the system is expected to conform to. Although existing artifacts may be out of date or simply not apply to the project, the team should at least make an effort to examine them and take advantage of the existing work wherever possible. A little bit of reading or discussion

with the right people is likely to save significant effort later on. In other words, the team should reuse existing artifacts whenever possible (Ambler, 2010).

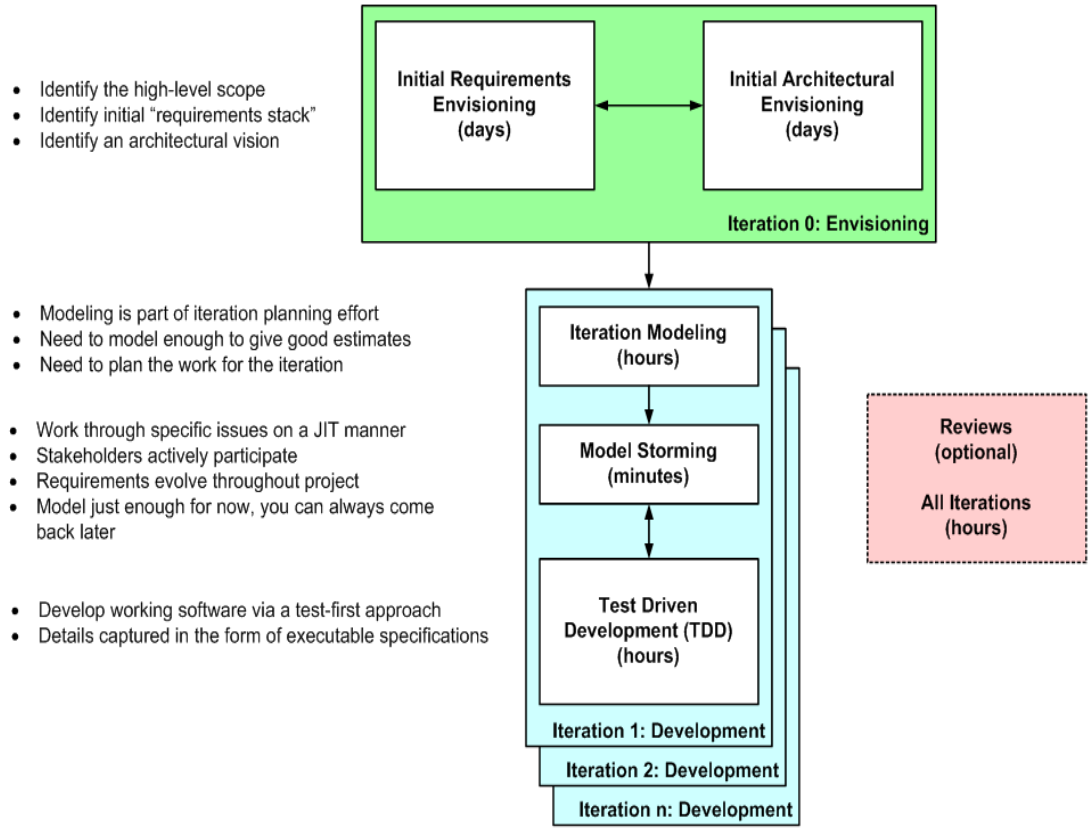
Principle #2 Build the simplest architecture that can possibly work

Modern systems are complex and require enterprise architects to consider a range of views in their architecture. Although they take different approaches, a multi-view strategy is a fundamental concept in modern architectural frameworks such as the Zachman Framework, TOGAF, 4+1, and so on. Each of these frameworks have very good reasons for their choice of views, they all seem to work well in practice and they can all be approached in an agile manner. This allows enterprise architects to review their options and pick the architectural framework which best reflects the culture of the organization (Ambler, 2010).According to (Leffingwell, *et al.,* 2008: 4), the only way to manage a large distributed system is to keep things as simple as possible. Fowler and Highsmith (2001) agree, referring tosimplicity as, "the art of maximizing the amount of work not done." Any software development task can be approached with a host of methods. In an agile project, it is particularly important to use simple approaches, because they are easier to change. It is easier to add something to a process that is too simple than it is to take something away from a process that is too complicated. Hence, there's a strong taste of minimalism in all the agile methods. The rule of thumb is to include only what everybody needs rather than what anybody needs, to make it easier for teams to add something that addresses their own particular needs (Fowler and Highsmith, 2001).

Things can be kept simple by making sure there are no hidden requirements and hidden dependencies in the design. Technology should be cut to the minimum needed to solve the problem at hand. The organization is not aided by creating artificial and unneeded layers of complexity.

Figure 11 below depicts the lifecycle of Agile Model Driven Development (AMDD). During "iteration 0", the first iteration of an agile project, there is a need to ensure that the project is organized and going in the right direction. Part of that effort is the initial requirements envisioning and architecture envisioning so that the development team is able to answer critical questions about the scope, cost, schedule, and technical strategy of the project (Ambler, 2010).

Figure 11: The Agile Model Driven Development (AMDD) lifecycle for software projects



(Ambler, 2010)

From an architectural point of view, during iteration 0 the goal is identify a potential technical direction for the team as well as any technical risks which will potentially be faced (risks which should be addressed by proving it with code). At this point there is no need for a detailed architectural spec; in fact creating such a spec at the beginning of a software development project is a very big risk (Ambler, 2010).

Instead, the details should be identified on a just-in-time (JIT) basis during iterations via initial iteration modeling at the beginning of each iteration, or by modeling storming throughout the iteration. The end result is that architecture emerges over time in increments, faster at first because of the greater need to set the foundation of a project, but still evolving over time to reflect the greater understanding and knowledge of the development team. This follows the practice to model in small increments and reduces the technical risk of the project. Cockburn (2005) agrees on the benefits of modeling in small increments, referring to

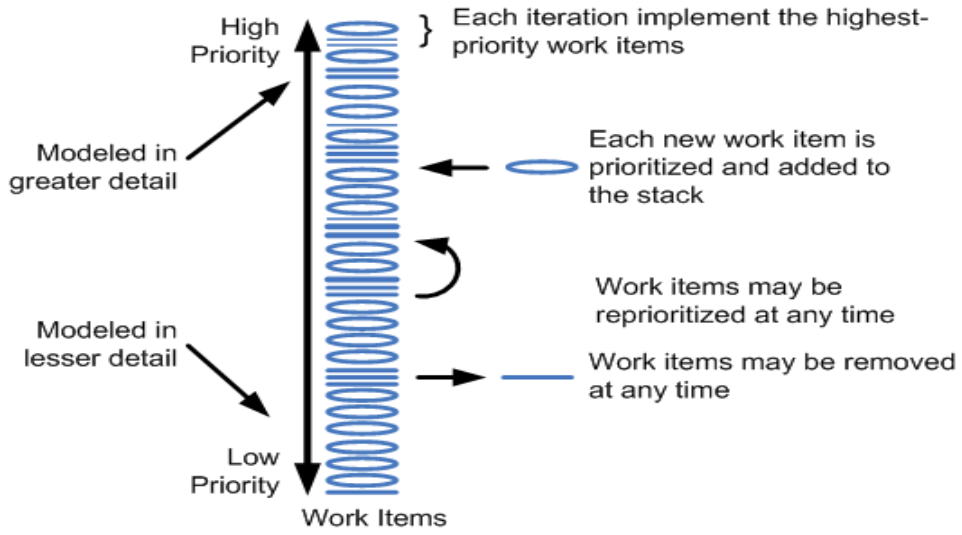
increments as, "process miniatures" and adds that the value of a process miniature is that everyone can look at the whole process and see how the various parts connect.

The development team always has a firm and proven foundation from which to work. In other words, the development team should think about the future, but wait to act (Ambler, 2010).

Principle #3 When in doubt, code it out

Ambler (2010) views a model as merely an abstraction, one that may appear to be very good but may not actually be in practice. Agile, with its highly iterative experience and code-based emphasis, allows developers to simply rely on their coding skills to move efficiently through the decision-making process. This is helpful when selecting a design alternative or a high-impact infrastructure implementation choice. It is important to obtain approval from the stakeholders as it is their money being spent in this effort. Figure 12 shows a discipline approach to prioritizing requirements and shows that these requirements can be changed, removed or more added at any time during the development process.

Figure 12: Disciplined agile change management process



(Ambler, 2010)

This principle is a reminder that when there is a tough decision to be made, developers can always turn to a rapid evaluation in code (Leffingwell, *et al.,* 2008: 5).

Principle #4 They build it, they test it

According to Leffingwell, *et al.* (2008: 5), agile is renowned for forcing testing early in the lifecycle of the development process. Many agile thought leaders implemented unit testing and acceptance testing frameworks into the base agile technical practices. Concurrent testing is a cornerstone practice of agile, and is a primary reason why quality is significantly higher

in agile, without sacrificing developer productivity. Because testing represents complexity at its highest level, the team that codes the system should be the team that determines how to test the system. With the complexity of today‟s automation frameworks, developers are likely to be directly involved in applying testing automation. It is the responsibility of the development teams to develop, test; and maintain a system-testing framework that continually assesses the system‟s ability to meet its architectural and functional requirements. This responsibility cannot be given to any other testing resource or outsourced function.

Principle #5 The bigger the system, the longer the runway

At the release level (internal or external), value delivery focuses on delivering the features customers need .That ability allows agile enterprises to communicate expectations to customers, whose businesses depend on new software releases. One of the key benefits of agile is that the team meets its commitments, and the software actually works. But even experienced agile teams occasionally have trouble completing iterations. In general, that can be acceptable, as a team that reliably completes 100 percent of the stories may not be stretching enough to meet the demands of the marketplace. Furthermore, so long as the team is able to self-correct effectively, it also encourages a level of acceptable risk taking (Leffingwell, *et al.,* 2008: 6).

However, when an iteration is missed badly (<50 percent of story completion and failure to deliver even the highest-priority stories), then the release itself may be at risk. In those cases, there is typically a serious architectural work at play, and the team simply underestimated the time it would take for a significant refactor or to lay in a new foundation. This leads to the conclusion that an agile team‟s ability to meet value delivery commitments is far more reliable when the foundation for the new features is already in place. This is why the need for the continuous build out of system infrastructure (architectural runway) must be in place to deliver features on the product roadmap as a mechanism for decreasing the risk of missed commitments (Leffingwell, *et al.,* 2008: 6).

For smaller teams, infrastructure to support a single iteration or release cycle may be all the runway that is needed. It may be much more efficient for those teams to be wrong initially, and then refactor the application, than it is to invest time up front trying to discover the undiscoverable. For larger teams and systems, however, building and refactoring infrastructure takes longer than a single short release cycle. Because of this, it is necessary to build most features for a particular release on existing infrastructure. This requires some

additional foresight and investment in more runway. Without additional runway, the team won‟t be able to reliably "land" each release on schedule (Leffingwell, *et al.,* 2008: 6).

Principle #6 System architecture is a role collaboration

According to Ambler (2010), everyone on the team should be responsible for architecture. This is in line with the principle of shared ownership advocated by Cockburn (2005) and Edwards (2006: 4) in their research into the principles of the Agile Manifesto. Architecture is far too important to leave in the hands of a single person no matter how bright they are; therefore architecture should be a team effort. In some cases on a small project team, for example fifteen people or less, it is good practice to include all of the developers as this allows everyone involved to have their say in the architecture. This increases everyone‟s understanding and acceptance of the architecture because they worked on it together as a team. It also increases the chance that developers are willing to change aspects of the architecture when the architecture proves insufficient, or perhaps it doesn‟t scale as well as initially thought, because it is the group‟s architecture and not just theirs. When something is developed by a single person it becomes "their baby" and nobody likes to hear that their baby is ugly; when a problem is found with their architecture they are likely to resist any criticisms of it. When an architecture is developed by the entire team then people are often far more willing to rethink their approach because it is a team issue and not a personal issue.

However, Ambler (2010) identifies two basic problems with the, "everyone owns the architecture," strategy:

1. **Sometimes people don't agree**. This strategy can fall apart dramatically when the team doesn't come to agreement; hence there is a need for someone to be in an architecture owner role to facilitate agreement.
2. **It doesn't scale**. When the team is large or geographically distributed, the team will have to be organized into a team of sub-teams. Architecture at scale requires a coordinating body in such situations.

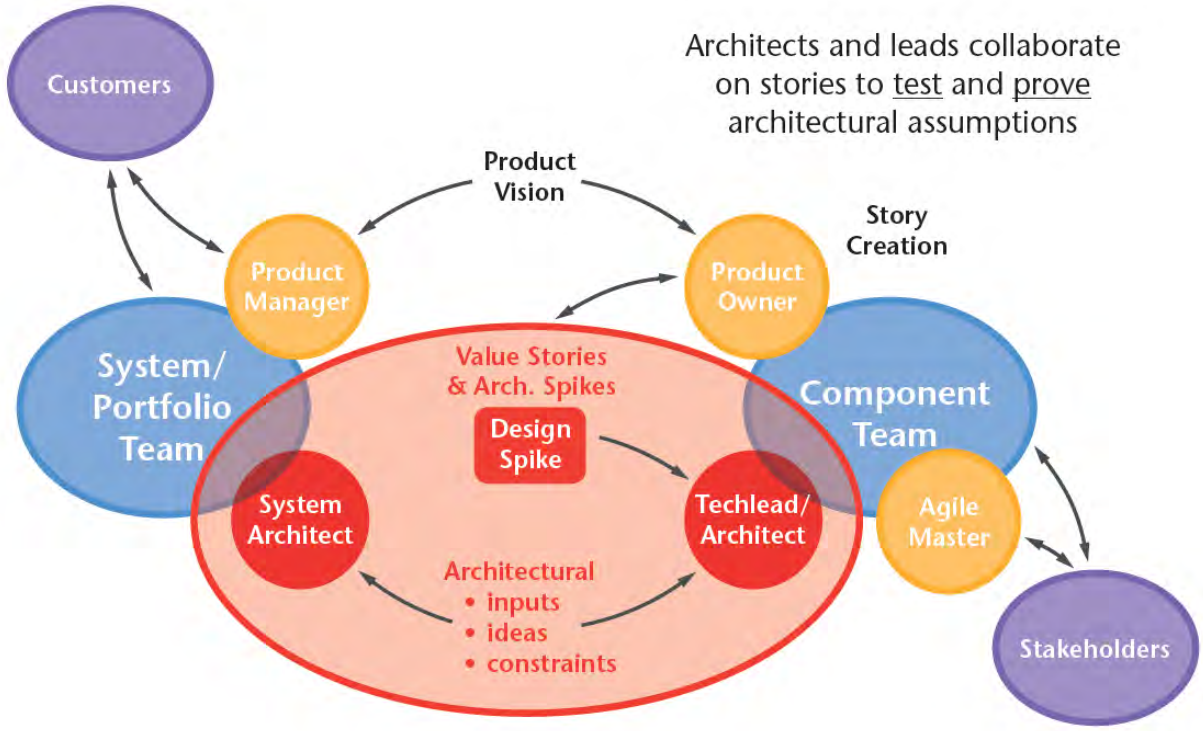
For any reasonably complex system the development team will need to invest some time architecting it. It is necessary to do some up front architecture envisioning to get the project started in the right direction and then the architecture will need to evolve from there.

Many agile teams find that they need someone in the role of architecture owner, often the most technically experienced person on the team, who is responsible for facilitating the

architectural modeling and evolution efforts. Just like Scrum's product owner role is responsible for the team's requirements, the architecture owner is responsible for the team's architecture. Architecture owner is different than the traditional role of architect. In the past, the architect would often be the primary creator of the architecture and would be one of the few people who worked on it. They would often develop the architecture and then present it to, or more accurately force it upon, the development team. An architecture owner collaboratively works with the team to develop and evolve the architecture. Although they are the person with the final decision-making authority when it comes to the architecture, those decisions should be made in a collaborative manner with the team. Effective architecture owners are developers experienced in the technologies that the organization is working with and have the ability to work on architecture spikes to explore new strategies. They should also have a good understanding of the business domain and have the necessary skills to communicate the architecture to developers and to other project stakeholders (Ambler, 2010). Cockburn (2005) believes that an effort to achieve increased collaboration and ownership should be extended to the stakeholders as well. When the customer/user/sponsor goes from being on the other side of the table to the same side, all kinds of good things start to happen, such as increased feedback, engagement, and satisfaction (Cockburn, 2005).

Figure 13 on the next page shows that system architecture is a role collaboration between the system architect and technical leads of the component teams that write the code. These system-level teammates work together with the component teams to decide what the architecture will look like. And when-in-doubt, they-code-it-out with a series of design spikes inside iteration or release boundaries. With the support of the product owner, design spikes are mixed in the backlog, based on the priorities the team feels is appropriate. This is one of the reasons effective product owners often have a high degree of technical experience. Using this model, a consensus emerges as to how to build the system that is about to be deployed (Leffingwell, *et al.,* 2008: 7).

Figure 13: Systems Architecture is a Role Collaboration



(Leffingwell, *et al.,* 2008: 7)

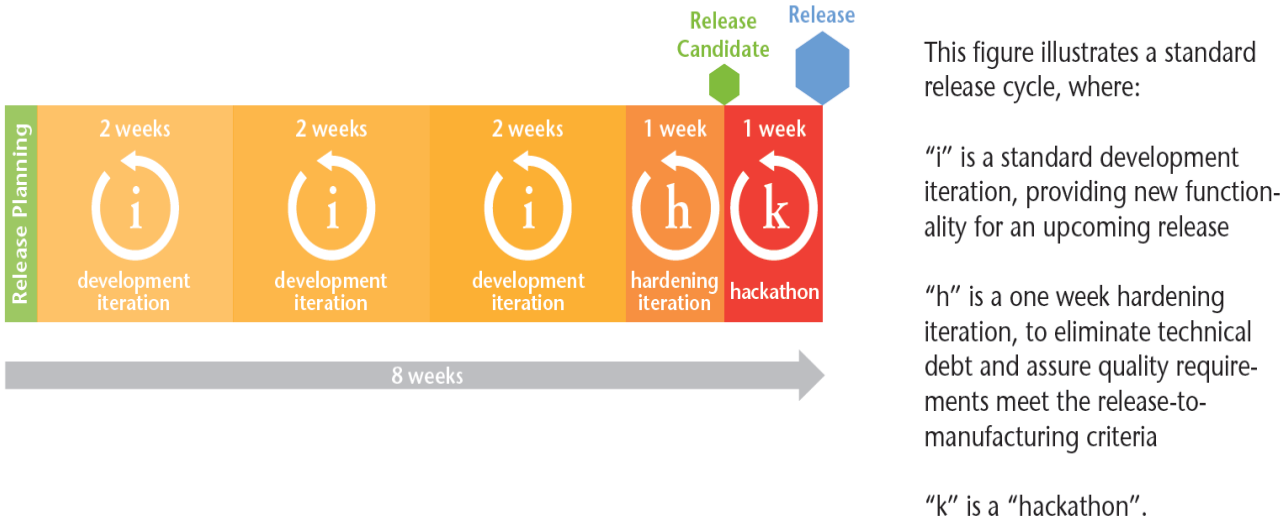
Principle #7 There is no monopoly on innovation

Agile practices provide a disciplined, production-like ability to reliably meet commitments and rapidly evolve a system to meet existing customer requirements. But there is a downside as well. If organizations are not careful, the "tyranny of the urgent" may keep them focused only on near-term deliverables. There is a need to introduce innovation in such a model. Mature agilists put processes in place to assure that innovation is not just incremental and near term. Fowler and Highsmith (2001) believe that this can be achieved by the development team acquiring a mindset and processes in which they welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.Some of the innovation comes from empowering system architects as part of their advanced guard. They can be exploring new technologies, patterns and techniques that will help them innovate (Leffingwell, *et al.,* 2008: 8).

But agile architecture provides a team-centric model, in which the architects alone are not relied upon as the sole source of such innovation. In fact, the team-centric model can foster

innovation at an even greater pace than that generally seen in traditional software organizations. That is because true innovators innovate at all stages of their career, and the team-centric model enables these people to flourish and contribute beyond what their level of experience may imply. One way to foster innovation at the team level is by judicious backlog management that includes spikes for refactoring, design and exploration of new ideas. This can work quite well, but even more explicit models have been put into use. For example, at Rally Software Development, where they built their SaaS Agile Product management solution in a highly agile fashion (with rarely a missed or delayed release commitment), they evolved to an advanced development cadence as illustrated in Figure 14 below (Leffingwell, *et al.,* 2008: 8).

Figure 14: An Iteration and Release cadence with one Innovation "hackthon" per Release



(Leffingwell, *et al.,* 2008: 8)

The hackathon is designed to foster innovation at the team level. The rules of the hackathon are simple: Any team member can explore any technology area in any way they want, as long as there is some correlation to the company"s mission. This gives the team some mental down time to reflect, think and experiment outside of the everyday rigor and pressures of the iteration and release cycle (Leffingwell, *et al.,* 2008: 9). Reflection will allow the development team to be constantly improving its practices in its local circumstances (Fowler and Highsmith, 2001). In addition, Sutherland (2010) suggests that during the exchange of ideas, each individual of the team should be treated as an equal and their opinion respected, even if it conflicts with another team members point of view. Innovation occurs only with the free interchange of conflicting ideas (Sutherland, 2010).

It is evident from the literature that the traditional role of the enterprise architect has no place in the agile environment. As mentioned previously by Ambler (2010), in the past, the architect would often be the primary creator of the architecture and would be one of the few people who worked on it. They would often develop the architecture and then present it to, or more accurately force it upon, the development team. The next section discusses the role of the enterprise architect in an agile enterprise.

4.5 The Role of the Enterprise Architect in the Agile Enterprise

The previous section discussed the principles of agile architecture. This section provides insight into the role of the enterprise architect in an agile environment.

Historically, agile architecture was a primary function of the system architect. But the most common agile methods don‟t define or even support such a role. Since agile focuses on harnessing the power of the collective team, rather than any one individual, the system architect no longer dictates technical direction (Leffingwell, *et al.,* 2008: 2).

While these system architects have decades of technical experience, this expertise has most likely taken place outside of the agile process, and they may not understand the construct of building refactorable code. Indeed, they may view the practice as unnecessary rework, and might not support the agile model. System architects may also be concerned about the potential architectural entropy of all the newly empowered and energized agile teams. They may also have strong opinions about the software development practices teams employ. Failure to bring these key stakeholders on board to the agile development paradigm could quickly kill the entire initiative. A battle between agile teams and system architects should be avoided at all costs, for there will be no winner in that fight. Therefore, it is definitely in the best interest of agile development to include system architects in the agile process, and their input should be highly valued by the team (Leffingwell, *et al.,* 2008: 3).

There are substantial benefits when agile EA is effectively practiced, provided that development is not slowed and the team does not capitulate to the waterfall design phases of the past. There are also some disadvantages to agile EA that must not be overlooked. The next section lists these benefits that an organization can look forward to and disadvantages that they must take into consideration before adopting agile EA.

4.6 The Benefits and Disadvantages of Agile Enterprise Architecture

The previous section discussed the role of the enterprise architect in the agile enterprise. This section provides insight into the benefits and disadvantages of an agile approach to developing EA.

According to Ambler (2009), the agile approach has, but is not limited to the following benefits and disadvantages:

Benefits of the Agile Approach:

- The development team can quickly discover whether or not their ideas work, and if so then how well.
- The chance that project teams understand the architecture is improved because they work face-to-face.
- Ideas are cross-fertilized, particularly technical ones, across teams, quickly sharing good ideas and strategies.
- There is a greater that a common infrastructure, both technical and business, will be built and reused over time because the project teams will be working towards the enterprise architecture.
- Experience is gained in the tools and technologies that the project teams work with, as well as the business domain itself, improving your own understanding of what it is that you‟re architecting.
- Concrete feedback is obtained that can be acted upon to improve the architecture, enabling it to evolve over time to meet the needs of the organization.
- The stakeholders have confidence in the development effort because they can see results from each development iteration.
- Application developers and agile database administrators on the project teams are mentored in modeling and architecture, improving their skill sets.
- The overall data management (including Master-Data Management (MDM)), security management and network management efforts support and enhance the development team‟s efforts instead of hinder them.

Potential Problems with the Agile Enterprise Architecture Approach:

- It does not include an explicit way to ensure compliancy (although having enterprise architects embedded on the teams goes a long way towards this).
- It depends on people being responsible.
- It requires that the team actively strive to keep things simple.
- It requires the team to accept an agile approach to modeling and documentation.

Part of this thesis work involves developing a comprehensive definition for agile EA.Agile EA is defined within the context of this research as the systematic process of adhering to agile development principles while translating business vision and strategy into effective enterprise change by flexibly creating, communicating and improving key requirements, principles and models. The models should have agile characteristics embedded in them, describe the enterprise‟s future state while keeping options open as late as possible and enable its evolution. The scope of agile enterprise architecture includes the people, processes, information and technology of the enterprise, and their relationships to one another and to the external environment. Enterprise architects compose holistic solutions that address the business challenges of the enterprise and support the governance needed to implement them (Gartner, 2008a: 2; Lin *et al*., 2006; Madni, 2008: 50; Tsourveloudis and Valavanis; 2002: 330).

4.7 Conclusion

The volatile environment in which companies find themselves has spurred the need for a more agile approach to developing EA. Agility within the context of EA is a relatively new area and this leads to a lack of understanding of how it can be achieved. There is therefore a need for a comprehensive and consistent definition for agile EA within an organization.

As EA is a process, in order to ensure that agile EA is developed, agility characteristics should not only be embedded in the end products created by the process of architecting, as is the traditional practice, but also in the process itself. Commenting on agile software development practices from an EA perspective proves that agility can be introduced into the process of EA. There are also agile architecture principles which an organization can adhere to when developing their EA. These principles do not recognize the traditional role of the systems architect. Since agile focuses on harnessing the power of the collective team, rather

than any one individual, the system architect no longer dictates technical direction. While an agile approach to EA comes with it many advantages, there are also some disadvantages which must not be overlooked.

The agile architectural approach described in this chapter is markedly different from what a lot of organizations are currently doing today. Table 2 below compares and contrasts the architectural practices that are commonly found in many organizations with their agile counterparts. Clearly, there is a big difference.

Table 2: Comparing Common and Agile Architectural Practice

Common Practice	Agile Practice
Architects are held in high esteem and are often placed, or even worse place themselves, on pedestals	Agile architects have the humility to admit that they don't walk on water
Architects are too busy to get their hands dirty with development	Agile architects are active members of development teams, developing software where appropriate and acting as architectural consultants to the team
Architecture models are robust to enable them to fulfill future requirements	Agile architects have the humility to admit that they can't predict the future and instead have the courage to trust they can solve tomorrow's problem tomorrow
The goal is to develop a comprehensive architecture early in a project	You evolve your architecture incrementally and iteratively, allowing it to emerge over time
Well-documented architecture model(s) are required	Travel light and focus on navigation diagrams that overview your architecture, documenting just enough to communicate to your intended audience
Architecture model(s) are communicated only when they are "suitable for public consumption"	Architecture model(s) are displayed publicly, even when they are a work in progress, to promote feedback from others
Architecture reviews are held to validate your model(s) before being put into use	Architectures are proved through concrete experiments

(Ambler, 2010)

The agile approach works because of its focus on people working together effectively as a team. These people keep things as simple as possible, developing the architecture in an iterative and incremental manner. Agile modeling recognizes that people are fallible, that they aren‟t likely to get the architecture right to begin with and therefore need the opportunity for acting on feedback from implementation efforts. When agile architects are productive members of the development team, and when the development team has been involved with the architectural efforts to begin with, then comprehensive documentation isn‟t needed by them. Agile architects have the courage to focus on solving today‟s problem today and trusting that they can solve tomorrow‟s problem tomorrow and the humility to recognize that they cannot accurately predict the future and therefore choose not to overbuild their architectures (Ambler, 2010).

The approach described in this chapter works incredibly if the development team is willing to embrace it (Ambler, 2009). The most important take away point is that it is all about people helping teams of teams build reliable, extensible, enterprise-class systems in an agile manner (Leffingwell, *et al.,* 2008: 8). Fancy tools based on theoretically sound frameworks, meta-models, or modeling languages are great to have but they will not do anything if the EA team does not use them. It‟s all about people. Sophisticated models and documents are interesting to create, but they offer little value if nobody reads them. It‟s all about people (Ambler, 2009).

This chapter provided a comprehensive definition for agile enterprise architecture. In addition, it outlined the importance of understanding how agility fits within the context of enterprise architecture (EA), the challenges with current methods for developing EA which favour an agile approach, as well as how agile software development principles can be applied to EA to ensure an agile EA. Thereafter, the best practices in agile architecture were discussed, highlighting how agile EA can be developed as well as the role of the enterprise architect in an agile EA environment showing how different it is in comparison to traditional EA practice.

This is an important step on the road to developing a "Framework for the Development and Measurement of Agile Enterprise Architecture."The last step is to uncover suitable measurement techniques that can be used to evaluate the agility of EA, which will be the subject of discussion in the next chapter.

Chapter 5: Measurement Techniques

Chapter four explored the concept of agile architecture. This involved describing how agility fits within the context of EA and the current best practices to developing agile EA.

Agility within the context of EA is a relatively new area and therefore there is limited research into a method and guidelines describing how the level of agility of EA can be measured.

The aim of this chapter is to discuss suitable measurement techniques that can be applied to measure the agility of EA.

The discussion will involve an introduction of why agility should be measured; providing the scope of agility measurement and thereafter go onto an evaluation of both informal and formal methods of measurement. The chapter will end with a summary of the findings from the literature review.

It was concluded that agility is an architectural characteristic that is worth getting prime attention – expressly and systematically. Agility should be measured, as it is at times found to be part of a big decision which an organization has to make. The informal measurement techniques discussed in this chapter involve developing an intuitive measurement habit; allowing an individual to reduce uncertainty about a subject and obtain an understanding of what to measure before more formal techniques can be employed to provide a more accurate measurement. The formal measurement techniques discussed are borrowed heavily from the manufacturing environment where, due to fluctuations in prices, customer demands, etc., uncertainty is the norm. These companies have had to be agile in order to survive and have had to develop methods to assess agility in order to know when agility is appropriate for a particular problem, where agility holds the highest payoff, to what degree agility is needed, and how to introduce agility in the process and/or product at the right points to achieve desired outcomes.

5.1 Introduction

This section discusses why it is important to measure agility.

Kidd (1994) explains that agility can be something that companies achieve without realizing it, or it can relate to issues that are difficult to quantify. The nature of the competencies implied by agility is such that they would be better considered as intangibles, similar to intellectual property, company specific knowledge, skills, expertise, etc.

Hubbard (2007: 2) believes that there is a costly myth that permeates many organizations today: that certain things can‟t be measured. The widely held belief must be a significant drain on the economy, public welfare, the environment, and even national security. „Intangibles‟ such as agility are frequently part of some critical business or government policy decision. Often, an important decision requires better knowledge of agility, but when an executive believes something to be immeasurable, attempts to measure it will not even be considered. Often a client would dismiss agility as completely beyond measurement when undertaking a major new investment or policy decision.

As a result, decisions are less informed than they could be. The chance of error increases. Resources are misallocated, good ideas are rejected, and bad ideas are accepted. Money is wasted. In some cases life and health are put in jeopardy. The belief agility might be impossible to measure is sand in the gears of the entire economy (Hubbard, 2007: 1).

Allen (2009) supports the belief that agility should be measured and adds that often, the term "agility" is bandied about, without real meaning.

Identifying the level of agility is the first step in understanding how much agility should be introduced and where; in order for the organization to survive or outperform its competitors. That does not stop an individual or team from using any available methods for determining the level of agility of EA. However, a lack of a comprehensive, standardized process leaves the determination of the current level of agility open to the interpretation of those engaged in the EA effort which could lead to the use of methods that are in some cases ineffective.

The next section details some of the informal and formal measurement techniques that can be applied to measure agility.

5.2 Measurement Methods

The previous section discussed why it is important to measure agility. This section discusses informal and formal methods of measurement that can be used to measure agility.

5.2.1 Informal Measurement

This subsection of measurement methods discusses informal methods of measurement that may be applied to measure agility.

Hubbard (2007: 17) in his measurement research found that company executives often say, "We can't even begin to guess at something like that." They dwell ad infinitum on the overwhelming uncertainties. Instead of making any attempt at measurement, they prefer to be stunned into inactivity by the apparent difficulty in dealing with these uncertainties. There are a lot of things that one may not know, but the question to be asked is how much is known. Other managers might object: "There is no way to measure that thing without spending millions of dollars." As a result, they opt not to engage in a smaller study, even though the costs might be very reasonable because such a study would have more error than a larger one. Yet perhaps even this uncertainty reduction might be worth millions, depending on the size and frequency of the decision it is meant to support.

As the next parts of this subsection will show, useful observations can tell someone something they didn't know before, even on a budget if they approach the topic with just a little more creativity and less defeatism.

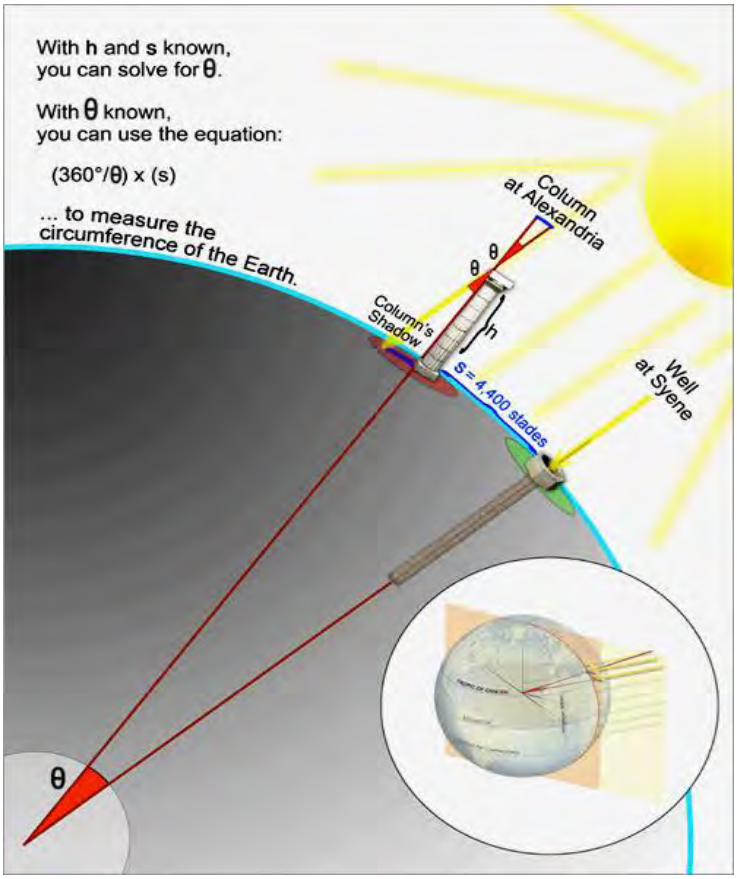
5.2.1.1How an Ancient Greek Measured the Size of Earth

The first mentor of measurement did something that was probably thought by many in his day to be impossible. An ancient Greek named Eratosthenes (276 BC–194 BC) made the first recorded measurement of the circumference of Earth (Hubbard, 2007: 8).

Eratosthenes didn't use accurate survey equipment, and he certainly didn't have lasers and satellites. He didn't even embark on a risky and probably lifelong attempt at circumnavigating Earth. Instead, while in the Library of Alexandria, he read that a certain deep well in Syene, a city in southern Egypt, would have its bottom entirely lit by the noon sun one day a year. This meant the sun must be directly overhead at that point in time. But he also observed that at the same time, vertical objects in Alexandria, almost straight north of Syene, cast a shadow.

This meant Alexandria received sunlight at a slightly different angle at the same time (Hubbard, 2007: 9).

Figure 15: The Eratosthenes Experiment



(Alfred, 2008)

Eratosthenes recognized that he could use this information to assess the curvature of Earth. He observed that the shadows in Alexandria at noon at that time of year made an angle that was equal to an arc of one-fiftieth of a circle. Therefore, if the distance between Syene and Alexandria was one-fiftieth of an arc, the circumference of Earth must be 50 times that distance. Modern attempts to replicate Eratosthenes‟s calculations vary by exactly how much the angles were, conversions from ancient units of measure, and the exact distances between the ancient cities, but typical results put his answer within 3% of the actual value. Eratosthenes‟s calculation was a huge improvement over previous knowledge, and his error was less than the error modern scientists had just a few decades ago for the size and age of the universe (Hubbard, 2007: 9).

Here is the lesson for business: Eratosthenes made what might seem an impossible measurement by making a clever calculation on some simple observations. If asked how a measurement of the circumference of the earth can be made, individuals are most likely to identify one of the „„hard ways"" to do it (e.g., circumnavigation). But Eratosthenes in fact may not have even left the vicinity of the library to make this calculation. One set of observations that would have answered this question would have been very difficult to make, but his measurement was based on other, simpler, observations. He wrung more information out of the few facts he could confirm instead of assuming the hard way was the only way (Hubbard, 2007: 9).

5.2.1.2 Estimating: Be Like Fermi

Another example from outside business that might inspire measurements within business is Enrico Fermi (1901–1954), a physicist who won the Nobel Prize in physics in 1938. He had a well-developed knack for intuitive, even casual-sounding measurements. One renowned example of his measurement skills was demonstrated at the first detonation of the atom bomb, the Trinity Test site, on July 16, 1945, where he was one of the atomic scientists observing the blast from base camp. While final adjustments were being made to instruments used to measure the yield of the blast, Fermi was making confetti out of a page of notebook paper. As the wind from the initial blast wave began to blow through the camp, he slowly dribbled the confetti into the air, observing how far back it was scattered by the blast (taking the farthest scattered pieces as being the peak of the pressure wave). Fermi concluded that the yield must be greater than 10 kilotons. This would have been news, since other initial observers of the blast did not know that lower limit. After much analysis of the instrument readings, the final yield estimate was determined to be 18.6 kilotons. Like Eratosthenes, Fermi was aware of a rule relating one simple observation; the scattering of confetti in the wind to a quantity he wanted to measure. The value of quick estimates was something Fermi was familiar with throughout his career (Hubbard, 2007: 9).

He was famous for teaching his students skills at approximation of fanciful-sounding quantities that, at first glance, they might presume they knew nothing about. The best-known example of such a „„Fermi question""was Fermi asking his students to estimate the number of piano tuners in Chicago. His students; science and engineering majors would begin by saying that they could not possibly know anything about such a quantity. Of course, a solution would be to simply do a count of every piano tuner perhaps by looking up advertisements, checking

with a licensing agency of some sort, and so on. But Fermi was trying to teach his students how to solve problems where the ability to confirm the results would not be so easy. He wanted them to figure out that they knew something about the quantity in question. He would start by asking them to estimate other things about pianos and piano tuners that, while still uncertain, might seem easier to estimate. These included the current population of Chicago (a little over 3 million in the 1930s to 1950s), the average number of people per household (2 or 3), the share of households with regularly tuned pianos (not more than 1 in 10 but not less than 1 in 30), the required frequency of tuning (perhaps 1 a year, on average), how many pianos a tuner could tune in a day (4 or 5, including travel time), and how many days a year the turner works (say, 250 or so). The result would be computed:

Tuners in Chicago = Population / people per household
X percentage of households with tuned pianos
X tunings per year/ (tunings per tuner per day
X workdays per year)

Depending on which specific values chosen, one would probably get answers in the range of 20 to 200, with something around 50 being fairly common. When this number was compared to the actual number (which Fermi might get from the phone directory or a guild list), it was always closer to the true value than the students would have guessed. This may seem like a very wide range, but consider the improvement this was from the „„How could we possibly even guess?"" attitude his students often started with (Hubbard, 2007: 10). This approach also gave the estimator a basis for seeing where uncertainty came from. Was the big uncertainty about the share of households that had tuned pianos, how often a piano needed to be tuned, how many pianos can a tuner tune in a day, or something else? The biggest source of uncertainty would point toward a measurement that would reduce the uncertainty the most.
A Fermi question is not yet quite a measurement. It is not based on new observations. It is really more of an assessment of what is already known about a problem in such a way that it can get you in the ballpark (Hubbard, 2007: 11).

The lesson for business is to avoid the quagmire that uncertainty is impenetrable and beyond analysis. Instead of being overwhelmed by the apparent uncertainty in such a problem, one should start to ask what things about it are known. As will be seen later, assessing what is currently known about a quantity is a very important step for measurement of those things that do not seem as if they can be measured at all (Hubbard, 2007: 11).

5.2.1.3 Experiments: Not Just for Adults

Another person who seemed to have a knack for measuring her world was Emily Rosa. Although Emily published one of her measurements in the Journal of the American Medical Association (JAMA), she did not have a PhD or even a high school diploma. At the time she conducted the measurement, Emily was a 9-year-old working on an idea for her fourth-grade science fair project. She was just 11 years old when her research was published, making her the youngest person ever to have research published in the prestigious medical journal and perhaps the youngest in any major, peer-reviewed scientific journal (Hubbard, 2007: 12).

In 1996 Emily saw her mother, Linda, watching a videotape on a growing industry called „„therapeutic touch,"" a controversial method of treating ailments by manipulating the patients" „„energy fields."" While the patient lay still, a therapist would move his or her hands just inches away from the patient"s body to detect and remove „„undesirable energies,"" which presumably caused various illnesses. Emily suggested to her mother that she might be able to conduct an experiment on such a claim (Hubbard, 2007: 13). Linda, who was a nurse and a long-standing member of the National Council Against Health Fraud (NCAHF), gave Emily some advice on the method. Emily initially recruited 15 therapists for her science fair experiment. The test involved Emily and the therapist sitting on opposite sides of a table. A cardboard screen separated them, blocking each from the view of the other. The screen had holes cut out at the bottom through which the therapist would place her hands, palms up, and out of sight. Emily would flip a coin and, based on the result, place her hand four to five inches over the therapist"s left or right hand. The therapists, unable to see Emily, would have to determine whether the girl was holding her hand over their left or right hand by feeling for the girl"s energy field. Emily reported her results at the science fair and got a blue ribbon, as everyone else did. Linda mentioned Emily"s experiment to Dr. Stephen Barrett, whom she knew from the NCAHF. Barrett, intrigued by both the simplicity of the method and the initial findings, mentioned it to the producers of the TV show Scientific American Frontiers shown on the Public Broadcasting System. In 1997 the producers shot an episode on Emily"s method, and Emily recruited 13 more therapists for the show, for a total of 21 (Hubbard, 2007: 13).

The 21 therapists made a total of 280 individual attempts to feel Emily"s energy field. They correctly identified the position of Emily"s hand just 44% of the time. Left to chance alone, they should get about 50% right with a 95% confidence interval of +/_16%. (If you flipped

280 coins, there is a 95% chance that between 44% and 66% would be heads.) So the therapists may have been a bit unlucky (since they ended up on the bottom end of the range), but their results are not out of bounds of what could be explained by chance alone. In other words, people "uncertified" in therapeutic touch could have guessed and done as well as, or better than the therapists (Hubbard, 2007: 13).

The reason intangibles seem intangible is almost never for lack of the most sophisticated measurement methods. Usually things that seem immeasurable in business reveal themselves to much simpler methods of observation, once we learn to see through the illusion of immeasurability. In this context, Fermi"s value to us is in how we determine our current state of knowledge about a thing as a precursor to further measurement (Hubbard, 2007: 17).

Unlike Fermi"s example, Emily"s example is not so much about initial estimation since her experiment made no prior assumptions about how probable the therapeutic touch claims were. Nor is it about using a clever calculation instead of infeasible observations, like Eratosthenes. Her calculation was merely based on standard sampling methods and did not itself require a leap of insight like Eratosthenes"s simple geometry calculation. But Emily does demonstrate that useful observations are not necessarily complex, expensive, or even, as is sometimes claimed, beyond the comprehension of upper management even for ephemeral concepts like touch therapy or strategic alignment (Hubbard, 2007: 18).

As useful as these lessons are, there is still room to build even further in order to learn ways to assess the current uncertainty about a quantity (Hubbard, 2007: 18).

The benefits of seeing the world through „calibrated" eyes that see everything in a quantitative light have been a historical force propelling both science and economic productivity. Humans possess a basic instinct to measure, yet this instinct is suppressed in an environment that emphasizes committees and consensus over making simple observations. It simply won"t occur to many managers that an „„intangible"" can be measured with simple, cleverly designed observations (Hubbard, 2007: 40).

There are several misconceptions about measurement and what it means. Many have been exposed to basic concepts of measurement in, say, a chemistry lab in high school, but it is unlikely that much has been learned besides the idea that measurements are exact and apply only to the obviously and directly observable quantities. University statistics, however, probably help to confuse as many people as they inform. In the workplace, professionals at all

levels in all fields are inundated with problems that don‟t have the neatly measurable factors seen in high school and university problems (Hubbard, 2007: 40).

The techniques discussed in this subsection of measurement methods involve developing an intuitive measurement habit; allowing an individual to reduce uncertainty about a subject before more formal techniques can be employed to provide a more accurate measurement. The next section will involve a discussion of some of the formal methods for measuring agility.

5.2.2 Formal Measurement Methods

This subsection of measurement methods discusses some of the formal approaches that can be applied to measuring agility. The discussion includes an approach known as AgileTecting which uses agility metrics to assess the level of agility of the process and the products derived from the process, a goal based agility assessment which involves the definition of agile goals and an agility evaluation technique based on fuzzy logic.

5.2.2.1AgileTecting

According to Madni (2004: 50) competitive market forces continue to impose challenging trade-offs in the development of complex systems and products. On the one hand, there is great pressure to incorporate the latest breakthroughs. On the other hand, system reliability cannot be compromised. Agile systems architecting is concerned with balancing these competing requirements while determining where and how agility should be introduced to achieve a competitive advantage and/or requisite adaptability to changing circumstances. Agility in the development process is needed to rapidly and cost-effectively exploit technology breakthroughs for improving process efficiencies and costs.

AgileTecting is a principled approach for assessing when agility is appropriate for a particular problem, where agility holds the highest payoff, to what degree agility is needed, and how to introduce agility in the process and/or product at the right points to achieve desired outcomes. AgileTecting emphasizes the fact that the introduction of agility in an organization has to be accompanied by a corresponding change in organization and culture. AgileTecting is especially well suited for architecting complex, long-lived systems. A complex system is one in which overall system performance cannot be predicted from merely the “sum of the parts.” It consists of many types of components and connections, which may both change dynamically (Madni, 2008: 50).

Agility metrics are key to assessing the agility of both processes and products. To be of value, these metrics need to be relevant, measurable, and quantitative. They should be able to convey the impact of agility on both the process and product in business terms. Examples of process and product agility metrics are provided in table 3 below (Madni, 2008: 50). Within the context of EA, a model would relate to a diagram created by an architect. Whereas, a module would relate to a component within that diagram for example a software system.

Table 3: Process and Product Agility Metrics

Metric	Description	Process/Product Metric
Reuse	– ensure models are defined such that they can be employed in multiple applications. – make up front infrastructure investment to decouple unit cost from volume	Product
Continuous, Incremental Deployment	– provide for introduction of process change throughout the product lifecycle to opportunistically exploit new breakthroughs (in product and process)	Process and product
Dynamic Capacity Adaptation	– adapt capacity to demand in both process (i.e., manufacturing line) and product (i.e., end product in operational environment	Process and product
Individual Preference Support	– achieve customization of individual products at mass production efficiencies (e.g., automobile dealers)	Process and Product

Option Space Preservation	– avoid / circumvent premature closure of process and product option space to exploit new breakthroughs in process and technologies – postpone/defer decisions and "binding" among models as late as feasible to exploit the most recent information	Process and product
Plug-and-Play	– ensure that models share defined interaction and interface standards and can be easily inserted/removed	Product
Self-organization	– module relationships are self-determined with module interaction being self-adjusting or negotiated	Product
Dynamic Composability	– ensure functionality can be dynamically composed in response to mission objectives rather than hardwired	Product
Empowered "edges"	– ensure that lowest and farthest nodes in the net have the power to act autonomously in specific circumstances	Product
Evolvability	– ensure functionality and interfaces can cost-effectively evolve with new operational requirements and interface standards	Product

Functional Redundancy	– ensure multiple different ways to accomplish the same function to achieve fault protection without physical redundancy	Process
Organization Flattening	– introduce peer-to-peer relationships among nodes with concurrent interactions where needed	Process

(Madni, 2008: 51)

AgileTecting is a transformational process that is applied to both processes and products to make them agile to the degree needed. Both organizational and cultural change are part and parcel of this transformation. In subsequent paragraphs, two examples of AgileTecting are presented. The first is the transformation of a flexible manufacturing line into an agile manufacturing line. The second is the transformation of traditional enterprise architecture into scalable, extensible architecture through the use of an agile Service-Oriented Architecture (SOA) strategy (Madni, 2008: 51).

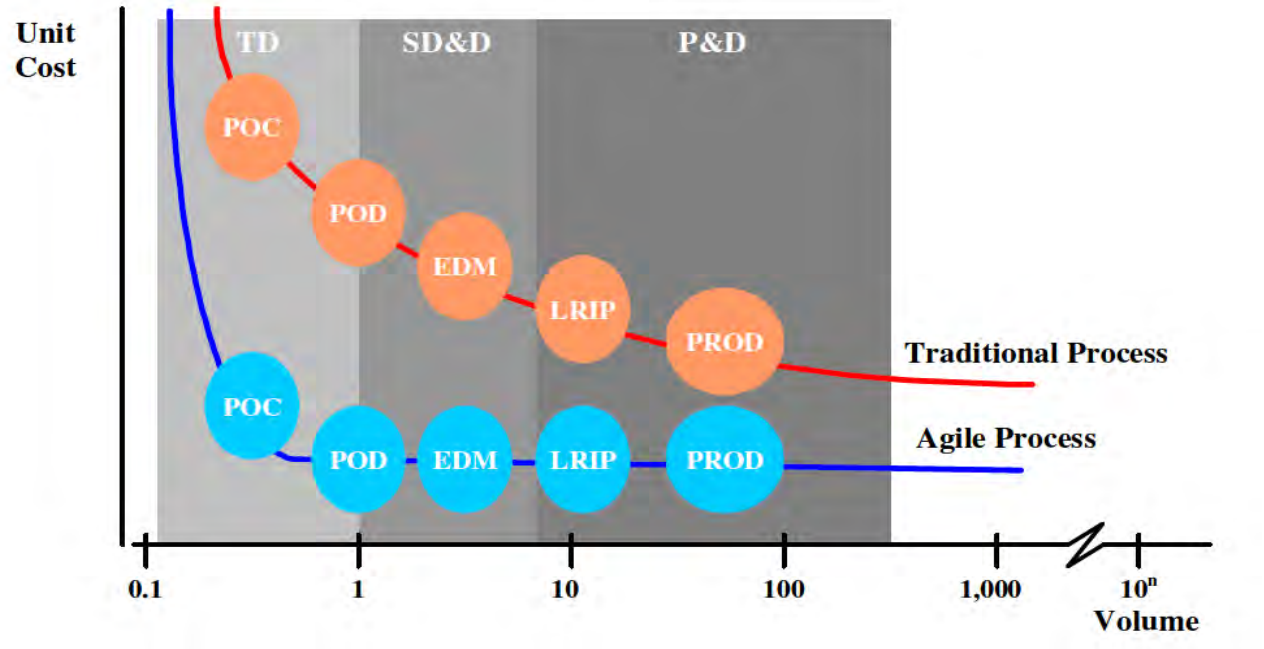
A flexible manufacturing line is easy to change but is not cost-effective for low build volume (i.e., unit cost is highly sensitive to build volume). An AgileTecting solution to this problem is as follows:

- Identify idle times in manufacturing line.
- Identify other company products that the manufacturing line could support during idle times with investment in infrastructure and proper tooling/equipment.
- Perform cost-benefit analysis to justify whether such investment is justified.
- Invest in necessary infrastructure/equipment, if justified.
- Employ manufacturing line for multiple products thereby eliminating idle times (i.e., keeping the line perpetually busy).

The outcome of this transformation is an agile manufacturing line. The increase in "effective build volume" of manufacturing line decreases unit cost for each product regardless of build volume. The net result is that unit cost is decoupled from build volume for each product supported by the manufacturing line by exploiting "volume effects" across multiple products.

The AgileTecting principle employed in this transformation is that of reuse. It is the reuse of the manufacturing line that achieves effective decoupling of unit cost from build volume (Madni, 2008: 51).

Figure 16: Agile manufacturing decouples unit cost from volume



(Madni, 2008: 53)

The second problem is concerned with transforming an enterprise IT architecture to overcome its scalability and extensibility limitations. The AgileTecting solution in this case is a Service-Oriented Architecture (SOA) whose tenets are to provide/enable:

- High assets reuse;
- Rapid, inexpensive replacement of computation modules (i.e., services);
- Rapid process reconfiguration (termed "orchestration" within an enterprise and "choreography" among separate enterprise systems);
- Easy extensibility to incorporate new organization elements, services, data stores;
- Cutting across "stovepipe" systems enabling all information relevant to a decision or action to be automatically assembled and provisioned to a service;
- (Re) configuration of process flow by business personnel rather than IT experts

(Madni, 2008: 54).

To achieve this transformation requires: identifying reusable IT services/components; mapping them to business processes; decoupling communications from services; and employing standard communication protocols. The outcome of this transformation is unbounded extensibility and scalability, and on-demand process adaptation through dynamic services orchestration. However, it is important to recognize that SOA adds complexity in two areas: *serializability*; making sure the transactions are entered into database(s) in the right order; and *atomicity*, ensuring that a series of related operations all occur, or do not occur. Despite the potential complications in SOA implementation, the transformation from a non-scalable, non-extensible IT architecture to an SOA is often times well worth the trouble (Madni, 2008: 54).

AgileTecting is a methodology for deciding whether or not agility is needed in the process and/or product, where it is needed, to what degree it is needed, and in what form it is needed. A clear distinction is made between an "agile system" which is a product, and "agile architecting" which is a process. This methodology emphasizes that not every problem situation requires agility and that agility invariably comes at a cost (e.g., increased complexity). It clarifies the fact that it is possible to have a flexible process that is not agile; however, every agile process is flexible given that flexibility is a key component of agility. The key challenge is in determining where agility is needed and to what end (Madni, 2008: 55).

5.2.2.2Goal based agility assessment

In software development there seems to be a general feeling in the agile community that if you follow all the practices associated with your chosen method then you are by definition agile. While this may be true of agile methods such as XP or Scrum which have defined a set of practices that have emergent properties such that the team becomes agile as a result of the process, it is still possible to use XP or Scrum without gaining much in terms of agility. There is much talk in the agile community of improving the development process but most of the improvements are anecdotal. There have been attempts at measuring and proving the efficacy of agile software development methods versus traditional methods. These studies have shown that agile methods are at least as good as traditional methods. There is talk of metrics which some people unfortunately frown upon because of the political connotations, but nevertheless metrics don't measure how agile you are. This part of the formal measurement methods defines a technique to assess agility through goals and using some examples shows how to create agile goals. The technique won't compare an organization to another, at least not

directly. Rather it is a means to measure the relative performance of an organization (Lappo& Andrew, 2008: 1).

The only practical way of determining whether agile methods actually make a difference to the software development process is by measuring the process and seeing whether agile methods actually make a difference. The question is how to make the measurements and what to measure. Once these questions have been answered it is possible to have some clarity about what makes a difference in the environment. While it is possible to use the technique proposed in this section to make a comparison between teams within an organization, it is difficult as often one is not comparing like with like. However, other peoples performances are useful as a guide to what can be achieved (Lappo& Andrew, 2008: 2).

There is a large body of work concerning software metrics. Most of this work is useful for the long term analysis of trends and comparative studies. Metrics are sometimes used during planning and bidding. However, metrics are not much use for assessing agility. Goal based agility assessment does not concern itself much with metrics in a belief that most metrics are measuring artifacts of the process such as lines of code or code complexity. Most environments collect some sort of metrics even if it is only hours spent on project tasks, requirements tested or defect rates. While these may be useful, especially the last two, the collection of hours on tasks is often a fantasy of the developer or manipulated by political necessities (Lappo& Andrew, 2008: 2).

Rather than just gather metrics such as lines of code, code complexity, function point or quality metrics there is a need for measurements that are related to business needs with an agile perspective. For instance, while code complexity analysis sheds light on the complexity of code, it does not give any idea of whether the code is easy to change in practice and hence having the potential to be agile. Code may also not be complex, but it may still be difficult being agile because of the process or because of the attitude or experience of the people working on the project. The point being that low level measurements of process artifacts don't necessarily mean anything at a higher process level. Using the measurable goals described below it is possible to define a set of goals for a team that are directly related to agile principles such as frequent delivery of software. Goals differ from metrics principally in that they attempt to be free from the details of the process, so that a goal to be responsive to change, for example, doesn't care about metrics like lines of code or code complexity. The

other reason to differentiate goals from metrics is because a goal implies thought about where the organization wants to be, rather than where they are now (Lappo& Andrew, 2008: 2).

The first step is defining a set of measurable goals for the process, environment, tools and software quality in conjunction with the project stakeholders (this includes developers). Then there is a need to determine what the current state of these goals are, agree on a future value with the project stakeholders, and take steps to achieve the agreed values. The process of achieving the goals should of course be iterative, with regular reviews on progress and the goals themselves. The cost and benefit of change should also be considered thereby preventing over or under investment. The goals, ideally, should be method agnostic, that is, they shouldn‟t be defined in terms of particular practices used to achieve agility or in terms of the artifacts of the process as this will stop method innovation and cause a lot of argument about favourite best practices (Lappo& Andrew, 2008: 3).

Numerous management techniques exist for improving processes, but perhaps the most interesting one to use at an early stage when investigating possible improvements is value stream mapping as used by the lean community. Value stream mapping produces a timeline for a complete process and determines those steps which add value to the process. Subsequent work entails eliminating steps that don't add value and eliminating process delays.
For example, the production release may have a number steps that cause unreasonable delays which could easily be eliminated or automated (Lappo& Andrew, 2008: 3).

It is useful to categorize goals to help define them and focus the mind on what goals are necessary. This research proposes four categories as follows (Lappo& Andrew, 2008: 4):

- **Process** goals associated with the software development process and the process practices used (Lappo& Andrew, 2008: 4).
- **Environment** goals associated with the environment the process runs in. These are mainly organizational and people oriented goals (Lappo& Andrew, 2008: 4).
- **Tools** goals associated with the tools used to develop the software (Lappo& Andrew, 2008: 4).
- **Software** goals associated with the design and quality of the software. How the software has been designed can have a big impact on agility and of course if the software is full of bugs or only manual testing is performed then again agility will be constrained (Lappo& Andrew, 2008: 4).

Goals are defined using the technique described by Gilb (1988: 135):

Name

This a short name for the goal to make it easy to remember and discuss. It is also used for cross referencing to other goals. For example, "Rate Of Change". Names are preferred over numbers as they are easier to remember and have more meaning (Lappo& Andrew, 2008: 4).

Test

The goal needs to be measured in some way. This defines the test to measure the goal and its scale. The test is the most important part of defining a goal. Tests should be quantitative when possible, but it is appreciated that some things are difficult to measure, such as knowledge transfer, so qualitative assessments can be used (Lappo& Andrew, 2008: 5).

For example, "Rate of Change" could be measured by running a query on the change management system to determine how many changes have been released to production over a given period. The scale could be changes per month (Lappo& Andrew, 2008: 5).

Benchmark

This is an actual measurement taken in the field. It could be data from within the own organization but is more likely to be a measure taken from the best organization in the same line of business. In other words, it is the benchmark to compare the organization against. This field is optional as the data may not be available or the "lean" approach which is to strive for perfection and ignore benchmarks has been taken (Lappo& Andrew, 2008: 5).

Now

Now simply states what the current measurement is. For example, the "Rate of Change" goal could be 1 change released per month (Lappo& Andrew, 2008: 5).

Worst

It is recognized that some goals may be difficult to achieve so this defines the lowest expected improvement in the goal. For example, the "Rate of Change" goal could have a worst case improvement of 2 changes per month (Lappo& Andrew, 2008: 5).

Planned

This is the planned level of the goal. For example, the "Rate of Change" goal could have a planned value of 20 changes per month (Lappo& Andrew, 2008: 5).

Planned Date

The planned date defines when you expect to achieve your planned or worst case goal (Lappo& Andrew, 2008: 5).

Net Benefit

Changes should not be made to an organization unless there is some idea of the net benefit of the goal, where net benefit is the potential value of the goal minus its implementation cost (Lappo& Andrew, 2008: 5). Value is a difficult thing to define and measure and even more difficult to predict. It also dangerous as one may oversell the benefit of a goal and raise expectations too high. Some goals may have intangible values. In this case it is best to simply list the benefits and costs. The cost is only an estimate as it is difficult to predict what the costs will be. For instance, a new environment may not be suitable for some people and they may leave; forcing the organization to replace them and train their replacements (Lappo& Andrew, 2008: 6).

Some goals don't add much value or the cost of achieving the goal is prohibitive in which case the goal should be dropped. The net benefit serves as a means of checking whether it‟s worth implementing this goal (Lappo& Andrew, 2008: 6).

Owner

All goals must be owned by an individual, this person should preferably be in the management team or steering committee. The owner is responsible for ensuring the goal is achieved but not necessarily implementing the goal, as this may be carried out by someone else (Lappo& Andrew, 2008: 6).

Notes

This is simply further notes of explanation which can include a reference to further information that may be relevant. It is optional (Lappo& Andrew, 2008: 6). However it should be possible to assess a particular practice in terms of the impact it has on agility (Lappo& Andrew, 2008: 3).

With a little thought it is possible to define a number of measurable goals which will help achieve greater agility, where of course agility is defined by the goals. Any number of management techniques can be used to achieve the goals with value stream mapping being particularly useful during analysis. An organization using this technique no longer has to worry about whether they are doing all the recommended XP practices in order to be agile

(whatever they may be at the time). If the agile goals satisfy the project stakeholders then the organization can consider itself agile. However, there is nothing stopping an organization from investigating what kind of agility scores their competitors are achieving and attempting to better them or taking the lean approach and simply aiming to be the best. By measuring what they are doing and setting goals for the future, an organization has the opportunity to achieve those goals (Lappo& Andrew, 2008: 8).

5.2.2.3Agility evaluation using fuzzy logic

As complexity of the market and production increases on a global scale, new integrated supply chain objectives, drivers, performance indicators and boundary conditions are being defined within the framework of agile manufacturing. Whilst the needs of integrated supply chain networks have been to a large extent identified, there is a lack of suitable and commercially available tools to satisfy these. Therefore, a new generation of tools should be developed and the existing tools significantly enhanced to support decision-making processes and to deliver the required solutions to extended businesses. Current approaches to the design and construction of supply chain systems lead to fixed interdependencies between valuable resources. This constrains the resource reuse and the agility of systems, often preventing close alignment between system behaviour and business process requirements (Jain, *et al*., 2008: 368).

According to Tsourveloudis and Valavanis (2002: 331), manufacturing systems engineering lacks analytic and closed-form mathematical solutions albeit in the simplest possible cases. Since manufacturing systems are operated and managed by people, it is necessary to record and utilize human knowledge and perceptions about agility and its factors (parameter quantification and measurement). Algebraic formulae fail in putting together the various dimensions of agility coupled with the human perception of agility. To overcome such problems, the key idea is to model human inference, or equivalently, to imitate the mental procedure through which experts (managers, engineers, operators and researchers) arrive at a value of agility by reasoning from various sources of evidence. To quantify agility, managers and operators, frequently use verbal or linguistic values, such as low, average, about high and so on. Thus, a valid and suitable candidate solution to the problem of measuring enterprise agility should be based on fuzzy logic. Regardless of the structure of each measure, it is important to establish basic principles, which should be satisfied by any such agility measure. It is postulated that any practical agility metric should provide a situation specific measurement by taking into account the particular characteristics of the system/enterprise

under study, and allow for comparisons among different installations. Further, it should incorporate all the relevant to agility accumulated human knowledge/expertise by focusing on specific observable measuring parameters that may be defined.

In view of the above statements, the proposed agility measurement scheme is:

- *Direct*: it focuses on the observable operational characteristics that affect agility (direct measurement), such as product variety, versatility, change in quality, networking etc., and not on the effects of agility (indirect measurement) such as, increased assets or profits, short delivery times, customer satisfaction, etc.

- *Knowledge-based*: it is based on the expert knowledge accumulated from the operation of the system under examination, or on similar systems. The measure is capable of handling both numerical and linguistic data, resulting in precise/crisp (e.g., agility = 0.85) and/or qualitative (e.g., high agility) measurements.

- *Holistic*: it combines all known dimensions of agility. Agility is a multidimensional notion, observable in almost all hierarchical levels of an enterprise. For quantification purposes, it is categorized into several distinct (enterprise) infrastructures.

Tsourveloudis and Valavanis (2002: 332) believe the essential concept in agile manufacturing is the integration of organization, people, and technology into a coordinated interdependent system, which responds rapidly to changes. The proposed measuring approach involves all the founding concepts of agility expressed, for the sake of analysis, in the following divisions/infrastructures:

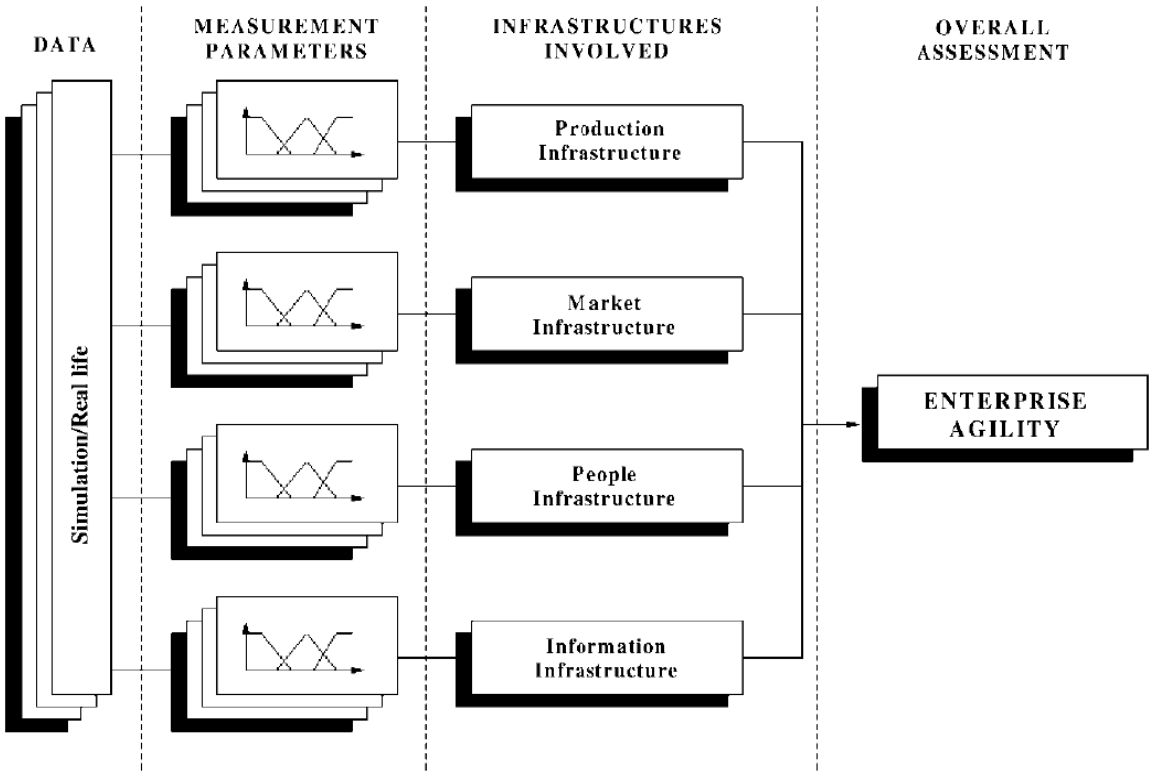
• *Production infrastructure*: Deals with plant, processes, equipment, layout, material handling, etc. It can be measured in terms of time and cost needed to face unanticipated changes in the production system.
• *Market infrastructure*: Deals with the external enterprise environment, including customer service and marketing feedback. It may be measured by the ability of the enterprise to identify opportunities, deliver, upgrade products/enrich services, and expand.

• *People infrastructure*: Deals with the people within the organization. The level of training and motivation of personnel may measure it.

• *Information infrastructure*: Deals with the information flow within and outside the enterprise. It may be measured by the ability to capture, manage, and share structured information to support the area of interest.

The key idea of this approach is to combine all infrastructures and their corresponding operational parameters as shown in Figure 17, to determine the overall agility. The value of agility is given by an approximate reasoning method taking into account the knowledge that is included in simple IF–THEN rules. This is implemented via multi-antecedent fuzzy IF–THEN rules, which are conditional statements that relate the observations concerning the allocated divisions (IF-part) with the value of agility (THEN-part) (Tsourveloudis and Valavanis, 2002: 332).

Figure 17: The Architecture of the proposed assessment of agility



(Tsourveloudis and Valavanis, 2002: 333)

An example of such a rule is:

IF the agility of *Production* Infrastructure is *Low*

AND the agility of *Market* Infrastructure is *Average*

AND the agility of *People* Infrastructure is *Average*

AND the agility of *Information* Infrastructure is *Average*

THEN the overall *Enterprise* agility is *About Low,*

WHERE *Production*, *Market*, *People*, *Information* infrastructures and *Enterprise agility* are the linguistic variables of the above rule, i.e., variables whose values are linguistic terms such as, *Low*, *Average*, *About Low*, rather than numbers. These linguistic ratings are represented with fuzzy sets having certain mathematical meaning represented by appropriate membership functions (Tsourveloudis and Valavanis, 2002: 334).

Since the impact of all individual infrastructures on the overall manufacturing agility is hard to be analytically computed, fuzzy rules are derived to represent the accumulated human expertise. In other words, the knowledge concerning agility, which is imprecise or even partially inconsistent, is used to draw conclusions about the value of agility by means of simple calculus (Tsourveloudis and Valavanis, 2002: 334).

5.3 Conclusion

Agility is an architectural characteristic that is worth getting prime attention – expressly and systematically. Agility should be measured, as it is at times found to be part of a big decision which an organization has to make. Although it is deemed intangible, the lesson for business is to avoid the quagmire that uncertainty is impenetrable and beyond analysis. Instead of being overwhelmed by the apparent uncertainty in such a problem, one should start to ask what things about it are known. Useful observations can tell someone something they didn‟t know before, even on a budget if they approach the topic with just a little more creativity and less defeatism.

The informal measurement techniques discussed in this chapter involve developing an intuitive measurement habit; allowing an individual to reduce uncertainty about a subject and obtain an understanding of what to measure before more formal techniques can be employed to provide a more accurate measurement.

The formal measurement techniques discussed are borrowed heavily from the manufacturing environment where, due to fluctuations in prices, customer demands, etc., uncertainty is the norm. These companies have had to be agile in order to survive and have had to develop methods to assess agility in order to know when agility is appropriate for a particular problem, where agility holds the highest payoff, to what degree agility is needed, and how to introduce agility in the process and/or product at the right points to achieve desired outcomes.

This chapter outlined the importance of measuring agility and described both informal methods that can be used to reduce uncertainty as well as formal methods to measure agility.

As the conclusion of this chapter marks the end of the literature review, the next chapter will sum up the theoretical work by defining a framework and guidelines that can be used for the development and measurement of agile EA.

Chapter 6: Enterprise Architecture Agility Implementation Guidelines

Chapter five explored suitable measurement techniques that can be applied to measure the agility of EA. The discussion involved an introduction of why agility should be measured; providing the scope of agility measurement and an evaluation of both informal and formal methods of measurement.

The aim of this chapter is to sum up the theoretical work by defining a framework and guidelines for the implementation of agile EA, as well as a measurement method that can be used to assess the level of agility within an organization.

It was concluded that the IT industry could benefit from the development of a framework containing a set of guidelines or best practices and an EA agility measurement method that is architecture framework agnostic; allowing it to be used in conjunction with any architecture frameworks that an organizations may be using and that will ensure that EA agility is achieved effectively and efficiently. The preliminary framework guidelines presented in this chapter were developed from the concepts which relate to EA agility uncovered in previous chapters of the literature review. The measurement technique used to measure EA agility involved an adaptation of the research conducted by Tsourveloudis and Valavanis (2002: 331) into fuzzy logic as well as an intuitive measurement habit as suggested by Hubbard (2007: 11). The five stages of the framework have a dual function – first as guidelines for creating agile EA, but also as a framework for evaluating and measuring the agility of EA.

6.1 Introduction

This section will provide an introduction to the EA agility implementation guidelines. The discussion will include the origin of the proposed framework and highlight how it is different from the architecture frameworks that are currently available to practitioners of EA.

Some enterprise architects still use outdated, rigid approaches to EA which are incompatible with today‟s business environment. Their approach is out of a lack of understanding of how agility fits within the context of EA and a belief that EA is done for agility and cannot be done with agility (Edwards, 2008: 1).

In the same vein, it has been found that some authors view EA as a noun (product). It has been found when the focus of EA is on the noun, the EA team focuses on creating mountains of artifacts as opposed to meeting the strategic goals of the enterprise. The view taken in this thesis is to view EA as a verb (a process) which results in artifacts (nouns) created. In this way, the proper focus is maintained.

Therefore, as discussed previously in chapter 4, agile enterprise architecture is defined within the context of this research as the systematic process of adhering to agile development principles while translating business vision and strategy into effective enterprise change by flexibly creating, communicating and improving key requirements, principles and models. The models should have agile characteristics embedded in them, describe the enterprise‟s future state while keeping options open as late as possible and enable its evolution. The scope of agile enterprise architecture includes the people, processes, information and technology of the enterprise, and their relationships to one another and to the external environment. Enterprise architects compose holistic solutions that address the business challenges of the enterprise and support the governance needed to implement them (Gartner, 2008a: 2; Lin *et al*., 2006; Madni, 2008: 50; Tsourveloudis and Valavanis; 2002: 330).

Although there are numerous frameworks for those wishing to practice enterprise architecture, there is no accepted industry standard or widely used method describing the best way to go about being agile (The Open Group, 2003). In addition, the focus of the frameworks for creating agile enterprise architecture is on introducing agility in the end product created by the process of architecting and not on the process itself.

The IT industry could therefore benefit from the development of a framework containing a set of guidelines or best practices and an EA agility measurement method that is architecture framework agnostic; allowing it to be used in conjunction with any architecture frameworks that an organizations may be using and that will ensure that EA agility is achieved effectively and efficiently. As it will be architecture framework agnostic, the framework can be seen as a Meta-process methodology. All of the methodologies for agile software development that have been introduced during the last years can be categorized as meta-process methodologies or specific development methods. Meta process methodologies do not describe specific development approaches, but rather focus on general procedures to support the development team in establishing a project-specific approach.

The Framework is focused on a goal based agility assessment. Lappo and Andrew (2008: 2) believe that using measurable goals makes it possible to define a set of goals for a team that are directly related to agile principles such as frequent delivery of software. Goals differ from metrics principally in that they attempt to be free from the details of the process, so that a goal to be responsive to change, for example, doesn't care about metrics like lines of code or code complexity. The other reason to differentiate goals from metrics is because a goal implies thought about where the organization wants to be, rather than where they are now.

In their research on a goal based agility assessment, Lappo and Andrew (2008: 2) identify four categories of goals; process, environment, tools and software goals. As mentioned, this research focuses on agility of the process of EA. Therefore, process goals associated with the software development process and the EA process practices will be the focus of the framework (Lappo and Andrew, 2008: 4).

The framework and guidelines (goals/factors) derived in this chapter are a result of the discussions in the previous chapters, and, as such, sum up the main theoretical and best practice contributions of this thesis and my evaluation of those. They approach the EA discipline from various angles such as EA definition, EA methodology, participants in the EA process and agility within the context of EA. They also take into account process agility metrics and agile software development practices.

In defining a set of measurable goals for their four categories, Lappo and Andrew (2008: 3) highlight the need to determine what the current state of these goals are, agree on a future value with the project stakeholders, and take steps to achieve the agreed values. The process of achieving the goals should of course be iterative, with regular reviews on progress and the goals themselves. An organization wishing to engage in agile EA will be able to do so by conducting an initial measurement to evaluate their current level of agility obtaining a score and assess the gap between the organization‟s current state and that of the framework. Thereafter, the organization will be able to develop a roadmap which will take the EA practice from the current state, to the desired state in the framework.

The measurement method is an adaptation from a technique researched by Tsourveloudis and Valavanis (2002: 331) known as fuzzy logic. To quantify agility, managers and operators, frequently use verbal or linguistic values, such as low, average, about high and so on. Thus, a valid and suitable candidate solution to the problem of measuring EA agility should be based

on fuzzy logic (Tsourveloudis and Valavanis, 2002: 331). The measurement method in the framework will assign a weighting to each goal/criterion according to its overall importance in the development of agile EA uncovered in the previous literature. The method to assign a weighting to these criterion according to their perceived importance in the literature is synonymous with the work done by Hubbard (2007: 11) into developing an intuitive measuring habit to reduce the uncertainty of a quantity deemed "intangible" and therefore perceived as immeasurable. It is not based on new observations; it is really more of an assessment of what is already known about a problem in such a way that an intelligent estimation can be made and uncertainty about the subject reduced (Hubbard, 2007: 11).

To measure their agility score, an organization will evaluate their EA effort in order to see which of the criteria they currently meet. They will then use the measurement method within the framework to add their score together in proportion to the weighting of each criterion to determine the overall score of EA agility. Using linguistic values in this case wouldn"t result in a meaningful measurement for overall EA agility. The values low average and high will therefore be represented by the numerical values 1, 3 and 5 respectively. This will allow the values for each criterion to be added together to derive an overall score for EA agility.

According to Tsourveloudis and Valavanis (2002: 331), regardless of the structure of a measurement, it is important to establish basic principles, which should be satisfied by any such agility measurement. Further, it should incorporate all the relevant to agility accumulated human knowledge/expertise by focusing on specific observable measuring parameters that may be defined.

The proposed agility measurement scheme is:

- *Direct*: it focuses on the observable operational characteristics that affect EA agility (direct measurement), such as creating several architecture models in parallel, building the simplest architecture that can work etc., and not on the effects of agility (indirect measurement) such as, increased assets or profits, short delivery times, customer satisfaction, etc.

- *Knowledge-based*: it is based on the expert knowledge accumulated from previous research conducted by expert researchers and practitioners of enterprise architecture, agility, agile EA and measurement techniques.

- *Holistic*: it combines all known factors that affect EA agility that have been researched in previous chapters.

The proposed guidelines are different in that although there are a number of architecture frameworks that provide guidance on developing EA, there is no accepted industry standard or method for agile EA; describing the best way to go about the process of being agile. In addition, there is no widely available method for measuring the current level of EA agility within an organization. Agility within the context of EA is a relatively new area and therefore there is limited research into a method and guidelines describing how the level of agility of EA can be measured. That does not prevent an individual or team from using any available methods for determining the level of agility of EA. However, a lack of a comprehensive, standardized process leaves the determination of the current level of agility open to the interpretation of those engaged in the EA effort which could lead to the use of methods that are in some cases ineffective.

The next section will present the preliminary framework, guidelines and measurement method developed from the literature review.

6.2 Framework for the Development and Measurement of Agile Enterprise Architecture

The previous section provided an introduction to the EA agility implementation guidelines. The discussion included the origin of the proposed framework and highlighted how it is different from the architecture frameworks that are currently available to practitioners of EA.

This section discusses the framework containing a set of guidelines or best practices for agile EA and an EA agility measurement method that is architecture framework agnostic that will ensure that EA agility is achieved effectively and efficiently. The section will begin with a presentation of the different components/stages of the preliminary framework for the development of agile EA as well as the factors to be considered at each stage. The section will end with a presentation of the complete preliminary framework for the development and measurement of agile EA.

It was clear from the literature in previous chapters on the process of EA that while various authors have slightly different views on the process of EA in terms of when activities are to be done as well as names for different stages, there are some common themes that emerge.

This has led to the identification of five main stages in the process of EA. The framework is therefore split into three main stages; an EA Foundation Stage, an EA Approach Stage and an EA Extension and management Stage and also contains two supporting stages; an EA Maturity and Measurement Stage as well as an EA Governance and Management Stage that occur throughout the three main stages. Each stage contains considerations/factors that must be taken into account when attempting to develop agile EA. These considerations have a value in brackets associated with them which relates to their overall importance in the development of agile EA; low = 1, medium = 3 and high = 5. An organization wishing to measure their current level of agility will evaluate their EA effort in order to see which of the factors for the development of agile EA they currently meet. They will then add their score together in proportion to the weighting of each factor to determine the overall score of EA agility. Thereafter, when the areas of improvement have been identified by viewing which of the requirements in the framework are not currently met, the organization will be able to develop a roadmap which will take the EA practice from the current state, to the desired state in the framework.

The Framework is intended to be used by an organization that wishes to develop agile EA. The next subsections will discuss the guidelines at each stage presented in the framework for the development of agile EA. Each heading will provide the guidelines for each stage of development; and thereafter present a summary of the guidelines and their associated weighting in the form of an illustration which will form part of the overall framework for the development and measurement of agile EA.

6.2.1 EA Foundation Stage

This subsection of the framework guidelines discusses the activities that occur at the EA foundation stage.

The EA Foundation Stage occurs at the beginning of the EA process and it is in which the organization must perform a needs analysis and EA cost benefit analysis. It will be an advantage to obtain the top management‟s buy-in. This should help establish to what extent EA will assist in solving any problems and/or assist achieving the goals. The EA team should have a common definition for EA and agility and understand how agility fits within the context of EA. It is important that management communicate their vision and translate that vision into clearly defined goals as well as state how they intend to implement EA and the expected results. The organization should also engage in communication and governance

planning at this stage in order to set up effective communication and governance structures that will be used throughout the EA process (Bernard, 2004; Dziewulski, *et al*., 2003: 7; Gartner, 2008b: 4; Hansen, 2006: 23).

Although all of the factors are important, the activities at this stage are rated 1, indicating low importance in the development of agile EA. They have been included in the framework as they are necessary steps, but introduce very little levels of agility and instead are preliminary steps that set up an environment for the development of agile EA.

Figure 18 below shows a summary of the activities of the EA foundation stage.

Figure 18: The EA Foundation Stage

EA FOUNDATION STAGE
Needs and cost benefit analysis ***(1)***
Top management buy-in ***(1)***
Common EA definition ***(1)***
Common agility and agile EA definition ***(1)***
Management should communicate vision ***(1)***
Management should define goals ***(1)***
Management should define expected results ***(1)***
Management should engage in communication planning ***(1)***
Management should develop governance structures ***(1)***

(Own contribution)

6.2.2 EA Approach Stage

This subsection of the framework guidelines discusses the activities that occur at the EA Approach stage.

The EA Approach Stage occurs if an organization decides to commence on an EA program based on conclusions from the foundation stage. Top management should set-up a project sponsor and an executive body for the EA organization (Dziewulski, *et al*., 2003: 15; Hansen, 2006: 65). In this way, top management will be involved in the high-level decisions that need to be taken (Hansen, 2006: 65). Thereafter, architecting activities may be conducted by the EA team. This team should be lead and governed by the sponsor and the executive body (Dziewulski, *et al*., 2003: 15; Hansen, 2006: 65).

Leffingwell, *et al.* (2008: 3) stress the importance of the team committing to the project knowing that they are accountable for results. People collaborate better in much more of a

real-time basis, compared with tools and process, which tend to imply more of a delayed response time. This does not imply that process and tools must not be used, but collaboration and interaction using the people skills should be done first; then the models can be refined using the tools and processes.

According to Ambler (2010), everyone on the team should be responsible for architecture. Architecture is far too important to leave in the hands of a single person no matter how bright they are; therefore architecture should be a team effort. It is important to collaborate with customers in order to build up trust and prove the EA project‟s worth. This can only be done by showing real added value and tangible benefits in scheduled increments. The scope of the EA program should be clearly defined to ensure that all those involved know the end result and what is expected of them. Situational specific strategies, processes and practices should be developed to address the architecture requirements within the desired scope.

Agile practices provide a disciplined, production-like ability to reliably meet commitments and rapidly evolve a system to meet existing customer requirements. But there is a downside as well. If organizations are not careful, the "tyranny of the urgent" may keep them focused only on near-term deliverables. There is a need to introduce innovation in such a model. Mature agilists put processes in place to assure that innovation is not just incremental and near term. Some of the innovation comes from empowering system architects as part of their advanced guard. They can be exploring new technologies, patterns and techniques that will help them innovate (Leffingwell, *et al.,* 2008: 8; Madni, 2008: 50).

Ambler (2010) and Madni (2008: 50) state that a common mistake that architecture teams (or for smaller projects the architecture owner) will make is to ignore existing and pertinent artifacts, such as network or deployment diagrams that describe the organization‟s existing technical infrastructure, enterprise-level business models (use case models, process diagrams, workflow diagrams, corporate business rules, and so on), or corporate deployment standards (for workstations, branch offices, etc.) that the system is expected to conform to. Although existing artifacts may be out of date or simply not apply to the project, the team should at least make an effort to examine them and take advantage of the existing work wherever possible. A little bit of reading or discussion with the right people is likely to save significant effort later on. In other words, the team should reuse existing artifacts whenever possible.
Leffingwell, *et al.* (2008: 4) highlight the importance of determining the schedule in terms of iterations and releases referred to by Madni (2008: 50) as, "Incremental Deployment" in his

research. Ambler (2010) adds that this ensures that the architecture is developed in increments and proper practices to test the value added after each iteration are put into place, ensuring that the benefits of the EA program are realized quickly. In order not to slow project progress, the development team should create several architecture models in parallel.

Edwards (2006: 3) believes that a good architect would have a working architecture over comprehensive documentation. This implies being risk driven and attacking the highest risk to the enterprise first. Ambler (2010) agrees, stating that technology should be cut to the minimum needed to solve the problem at hand. The organization is not aided by creating artificial and unneeded layers of complexity.

Ambler (2010) views a model as merely an abstraction, one that may appear to be very good but may not actually be in practice. Agile, with its highly iterative experience and code-based emphasis, allows developers to simply rely on their coding skills to move efficiently through the decision-making process. This is helpful when selecting a design alternative or a high-impact infrastructure implementation choice. This principle is a reminder that when there is a tough decision to be made, developers can always turn to a rapid evaluation in code (Leffingwell, *et al.,* 2008: 5).

According to Leffingwell, *et al.*(2008: 5), agile is renowned for forcing testing early in the lifecycle of the development process. Many agile thought leaders implemented unit testing and acceptance testing frameworks into the base agile technical practices. Concurrent testing is a cornerstone practice of agile, and is a primary reason why quality is significantly higher in agile, without sacrificing developer productivity. Because testing represents complexity at its highest level, the team that codes the system should be the team that determines how to test the system. With the complexity of today‟s automation frameworks, developers are likely to be directly involved in applying testing automation. It is the responsibility of the development teams to develop, test; and maintain a system-testing framework that continually assesses the system‟s ability to meet its architectural and functional requirements. This responsibility should not be given to any other testing resource or outsourced function.

An agile team‟s ability to meet value delivery commitments is far more reliable when the foundation for the new features is already in place. This is why the need for the continuous build out of system infrastructure (architectural runway) must be in place to deliver features on the product roadmap as a mechanism for decreasing the risk of missed commitments.

Without additional runway, the team won't be able to reliably "land" each release on schedule (Leffingwell, *et al.,* 2008: 6).

It is at this stage where the greatest amount of agility in the EA development process can be introduced. This is signified by the activities at this stage predominantly having a rating of 5, indicating a high agility rating.

Figure 19 below shows a summary of the activities of the EA Approach Stage.

Figure 19: The EA Approach Stage

EA APPROACH STAGE
Elect project sponsor, executive body ***(1)***
EA Team should commit on behalf of themselves ***(5)***
EA Team should be accountable for results ***(5)***
Focus on individuals and interactions over processes and tools ***(5)***
Harness power of collective team rather than an individual ***(5)***
Customer collaboration over contract negotiation ***(5)***
No monopoly on innovation ***(5)***
Define EA scope ***(1)***
Reuse existing architecture artifacts ***(5)***
Develop situational specific strategies, processes, practices ***(5)***
Determine schedule in terms of iterations and releases ***(5)***
Team that codes the system designs the system ***(5)***
Create several architecture models in parallel ***(5)***
Develop architecture in increments ***(5)***
Build simplest architecture that can work ***(5)***
When in doubt, code it out ***(5)***
Whoever builds it should test it ***(5)***
Continuously build out system infrastructure ***(5)***

(Own contribution)

6.2.3 EA Extension and Management Stage

This subsection of the framework guidelines discusses the activities that occur at the EA Extension and Management stage.

Once the initial enterprise architecture is established, the EA team should proceed to extend and maintain the architecture (Harmon, 2003: 12). The ChiefInformationOfficerCouncil (2001) believes that the primary purpose of an EA is to inform, guide, and constrain the decisions for the enterprise, especially those related to IT investments. The true challenge of

enterprise engineering is to maintain the architecture as a primary authoritative resource for enterprise IT planning.
Maintaining the architecture does not provide much opportunity for the introduction of agility. Responding to change over following a plan therefore has a low rating.

Figure 20 below shows a summary of the EA Extension and Management Stage.

Figure 20: The Extension and Management Stage

EA EXTENSION AND MANAGEMENT STAGE
Respond to change over following a plan ***(1)***

(Own contribution)

6.2.4 EA Governance and Management Stage

This subsection of the framework guidelines discusses the activities that occur at the EA Governance and Management stage.

The EA Governance (EAG) and Management Stage focuses directly on establishing and implementing EA in the organization as well as on setting the policy for how EA subsequently should be run. EAG defines the systems, structures and responsibilities within which the EA and affected people must operate (Hansen, 2006: 35). This stage occurs shortly after the election of the project sponsor (s) and executive body and continues throughout the process of EA.

Management should ensure that the teams themselves are empowered to define, develop and deliver software, and that they are held accountable for the results. In order for teams to be held accountable, management should eliminate impediments for the team and they must be allowed to make the decisions required to support that accountability. If not, they will be held accountable for decisions made by others, and that is an ineffective and de-motivating model for team performance (Leffingwell, *et al.,* 2008: 3).

Eliminating impediments for the EA team and empowering them to deliver are activities that directly affect the freedom agile teams have to make decisions and are therefore have a high rating as these factors greatly affect EA agility.

Setting the policy of how EA should be run is an activity that has very little effect on EA agility and is therefore rated low.

Figure 21 below shows a summary of the activities of the EA Governance and Management stage.

Figure 21: The EA Governance and Management Stage

EA GOVERNANCE AND MANAGEMENT STAGE		
Eliminate impediments for the EA team ***(5)***	Set policy for how EA should be run ***(1)***	Management should be accountable for empowering teams to deliver ***(5)***

(Own contribution)

6.2.5 EA Maturity and Measurement Stage

This subsection of the framework guidelines discusses the activities that occur at the EA Maturity and Measurement stage.

Figure 25 below shows the EA Maturity and Measurement Stage. This is a supporting stage for the overall agile EA process and starts in the EA Foundation Stage and continues throughout the agile EA development process. Initially, the organization must evaluate their EA maturity in order to understand their requirements. Thereafter, the value added by EA is measured after the completion of a project iteration for a feature that can be tested and thereby evaluating its justification (Schekkerman, 2005). Good intentions and a good start are not measures of success. What matters in the end is completion that delivers performance and results (Hansen, 2005: 42).

Evaluating EA maturity is a preliminary step that is not only involved in agile EA practices but also in traditional EA practices and does not introduce much agility and therefore is rated low.

Measuring the value added by EA is an important step in any EA program. In the rigid approaches of the past, this step would only occur after the completion of the EA project. The agile approach to EA however favors measuring the value added after each iteration. This step has a medium rating, because although it does not directly introduce agility into the EA process, it is important in providing necessary feedback to the EA team that is critical to

making agile development decisions. In addition, the frequency of the measurements in an agile practice is increased; measurement occurs after each iteration as opposed to only at the end of a project.

Figure 22 below shows a summary of the EA Maturity and Measurement Stage.

Figure 22: The EA Maturity and Measurement Stage

EA MATURITY AND MEASUREMENT STAGE	
Evaluate EA Maturity ***(1)***	Measure value added by each iteration ***(3)***

(Own contribution)

From the framework stages presented, it is evident that the maximum agility score that an organization can achieve is 106. An organization wishing to measure their current level of agility will evaluate their EA effort in order to see which of the factors for the development of agile EA they currently meet. They will then add their score together in proportion to the weighting of each factor to determine their overall score of EA agility.

Figure 23 below shows the scale, which an organization can use to evaluate their level of agility.

Figure 23: Agility Scoring Scale

Agility Score	**LOW**	**MEDIUM**	**HIGH**
Range	**0 - 39**	**40 – 79**	**80 - 106**

(Own contribution)

As indicated above, a score in the range 0-39 corresponds to a low level of agility within the organization‟s EA practices. The range 40 – 79 corresponds to a medium level of agility and the range 80 to the maximum of 106 corresponds to a high level of agility.

Organizations that according to the preliminary framework have low levels of agility are typically still using outdated, rigid approaches to EA as opposed to a more agile approach which is favored for today‟s business environment. Their approach is out of a lack of

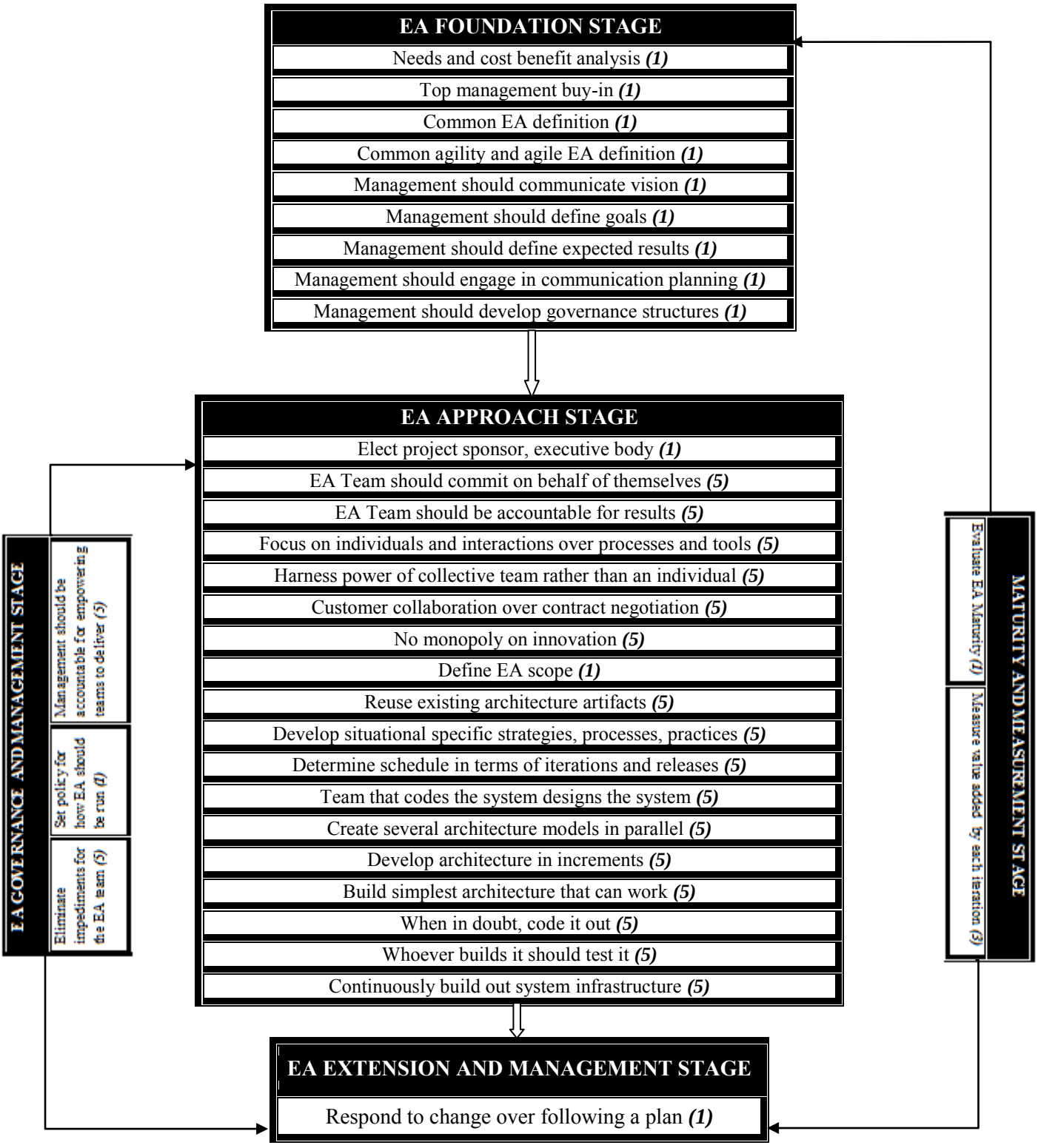
understanding of how agility fits within the context of EA and a belief that EA is done for agility, rather than with agility (Edwards, 2008: 1).

Organizations with a medium score of agility have most likely adopted more agile approaches to developing EA. Since the level of agility at this stage is still relatively low, it is possible however that these agile practices to developing EA were present in the organization without knowledge or focus on ensuring that EA was done in an agile manner. As signified by Kidd (1994), agility can be something that companies achieve without realizing it.

A high score of agility would suggest that the organization sees the value of practicing EA in an agile manner and has therefore focused on ensuring that their EA is done with agility.

Figure 24 on the next page provides an illustration of the complete preliminary framework for the measurement of agility of EA, showing the stages discussed above and how they are connected to one another.

Figure 24: Framework for the Development and Measurement of Agile EA



(Own Contribution)

6.3 Conclusion

The IT industry could benefit from the development of a framework containing a set of guidelines or best practices and an EA agility measurement method that is architecture framework agnostic; allowing it to be used in conjunction with any architecture frameworks that an organizations may be using and that will ensure that EA agility is achieved effectively and efficiently. The preliminary framework guidelines presented in this chapter were developed from the concepts which relate to EA agility uncovered in previous chapters of the literature review.

The measurement technique used to measure EA agility involved an adaptation of the research conducted by Tsourveloudis and Valavanis (2002: 331) into fuzzy logic as well as an intuitive measurement habit as suggested by Hubbard (2007: 11).

The five stages of the framework have a dual function – first as guidelines for creating agile EA, but also as a framework for evaluating and measuring the agility of EA. An organization wishing to engage in agile EA will be able to do so by ensuring that their efforts are in line with the criteria set out in the framework. The measurement method in the framework assigned a weighting to each criterion according to its overall importance in the development of agile EA. To measure how agile an organization‟s EA is, the organization will evaluate their EA effort in order to see which of the criteria they currently meet. They will then use the measurement method within the framework to add their score together to determine their overall score of EA agility.

The framework will be used by an organization wishing to engage in agile EA. It will be used as the initial step to agile EA by measuring the current state of the EA in order to identify the level of agility as well as the gap between the organization‟s current practices and those set out in the framework.

This chapter presented the preliminary framework which serves to inform this research of the fundamental elements necessary to design a comprehensive theoretical framework of factors for the successful development of agile EA in organizations.

The next chapter will present the research methodology adopted to explore the factors for the successful development of agile EA in organizations.

Chapter 7: Research Methodology

Chapter six provided a summation of the theoretical work by defining a framework and guidelines for the implementation of agile EA as well as a measurement method that can be used to assess the level of agility within an organization.

The discussion involved an introduction explaining how the preliminary framework was devised and thereafter a presentation and description of the preliminary framework and its guidelines.

The aim of this chapter is to present the research methodology adopted to explore the factors for the successful development of agile EA in organizations. The research questions, research method, unit of analysis, research instrument, the approach to analysing data and the research design are discussed.

It was concluded that in this research, data analysis follows an interpretive and qualitative approach based on case studies through a number of interviews with systems experts in four South African organizations in one province. The interviews are guided through the exploration of the extent of understanding of EA, agility and agile EA, the methods to measure the agility of EA and the factors affecting the development of agile EA in these organizations. This research follows an inductive approach to capture the interpretive experiences of participants and develop theoretical propositions from them.

7.1 Introduction

The aim of this section is to provide an introduction to the research methodology used in this research.

Yin (2003: 20) states that every type of research should have an implicit, if not explicit research design. This will guide the researcher in the process of collecting, analysing and interpreting observations. The topics that need to be addressed are what research questions need to be studied, what data are relevant, what data to collect, and how to analyse the results (Yin, 2003: 21).

The main problem of the research is to analyse and evaluate the extent of understanding of enterprise architecture, agility and agile EA, the methods that can be used to measure the agility of EA as well as the factors affecting the successful implementation of agile EA.

This was explored initially through a literature review. This chapter describes the research methodology and strategy to explore the extent of understanding of EA, agility and agile EA, the methods to measure the agility of EA and the factors for the successful implementation of agile EA within an organization in South Africa.

The qualitative and interpretive case study research methodology is chosen to enable an understanding of the phenomenon agile enterprise architecture with an exploratory strategy to identify the extent of understanding of EA, agility and agile EA, the methods to measure the agility of EA and the factors that affect the development of agile EA.

This chapter describes the research paradigm used in this study with a thorough investigation into why a qualitative interpretive approach is the most appropriate for the means of this study. The research questions and the research method are highlighted, including the research approach used; the unit of analysis; the research instrument; and the approach to analysing data. The chapter concludes with a summary of the research design in the three phases. The design will guide the collection and analysis of data to present an exploration of the extent of understanding of EA, agility and agile EA, the methods to measure the agility of EA and the factors required for the successful development of agile EA.

7.2 Research Questions

The previous section provided an introduction to the research methodology. This section describes the research questions that are studied within this thesis. The discussion includes a presentation of the research questions, previous research into EA and the preliminary framework as well as the factors influencing the choice of research method.

This research has the intention of expanding knowledge and contributing to the understanding of EA, agility and agile EA, measurement methods which can be applied to measure the agility of EA and which factors should be taken into account if an organization is interested in developing agile EA.

The research questions that are studied will be used to determine the research methodology most appropriate for this research:

- *What is the extent of understanding of enterprise architecture within selected South African Organizations?*
- *What is the extent of understanding of agility and agile EA within selected South African Organizations?*
- *What methods (if any) are used to measure the agility of EA within selected organizations?*
- *What are the factors for the successful implementation of agile EA within an organization?*

It is important for those involved in its implementation to understand what is meant by „enterprise architecture". Varying views lead to a lack of unified knowledge, presence of varying cultures, and policies that prevent efficient management of enterprise processes, innovation and building software (Shuja, 2010: 1). In the same vein; it is important for an organization to have a consistent definition for agility as well as a common understanding of what it means to be agile. This will allow them to be in the best position to sense environmental change and respond effectively to that change (Gartner, 2006: 2; Tsourveloudis and Valavanis, 2002: 330). It is also necessary to be aware of any methods that an organization uses (if any) to determine the level of agility of their EA. A common understanding of both EA and agility is the initial step in understanding how the two relate to each other and the factors affecting the development of agile EA as well as applicable measurement techniques from the perspective of the four selected South African Organizations. This will allow an expansion on the existing theory or development of new theoretical concepts (Hunter, 2004: 296).

7.2.1 Previous Research and Preliminary Framework

This subsection of the research questions describes the problems associated with the previous research on EA and how the preliminary framework and research questions will be investigated.

Many problems are evident with the traditional approaches to developing EA. It is clear that EA can benefit from a more agile approach to development.

The review of the literature showed that not only is there a lack of a consistent definition for both EA and agility, but some enterprise architects still use outdated, rigid approaches to EA

which are incompatible with today‟s business environment. Their approach is out of a lack of understanding of how agility fits within the context of EA and a belief that EA is done for agility, rather than with agility. Identifying the level of agility is the first step in understanding how much agility should be introduced and where; in order for the organization to survive or outperform its competitors (Nair, 2008). Agility within the context of EA is a relatively new area and therefore there is limited research into a method and guidelines describing how the level of agility of EA can be measured. That does not stop an individual or team from using any available methods for determining the level of agility of EA. However, a lack of a comprehensive, standardized process leaves the determination of the current level of agility open to the interpretation of those engaged in the EA effort which could lead to the use of methods that are in some cases ineffective.

The problem of measuring the agility of EA may be conceptualized as a process of socio-technical innovation (Rowlands, 2005: 85). This judgment is qualified through the review and discussion of various definitions for EA, agility, agile EA; approaches to developing and measuring agile architecture as well as a case that is argued for the importance of agility when developing EA. By focusing on the problem of developing and measuring agile EA as a process of socio-technical innovation, an initial framework is developed which is comprised of the individual factors that contribute to the unexamined aspects of developing agile EA within organizations. This literature and framework provide valuable tools for the examination and analysis of the process of developing agile EA. Such a theoretical framework can be used as a base to make some explicit theoretical statements (Rowlands, 2005: 86).

The study is, however, not constrained by this framework and these concepts may also be considered as a purpose of the study. The development of a framework which aggregates the factors that affect the development of agile EA and a measurement technique is seen as a continuous building process (for future research), into appropriating a comprehensive framework for the successful development and measurement of agile EA in an organization.

The current aim on the exploratory research is to investigate the validity of the definitions created in the literature review for EA, agility and agile EA and applicability of the concepts in the framework in a real work environment. Additional concepts for the development of agile EA will also be investigated as well as measurement techniques used by organizations to measure the agility of EA in order to add knowledge to the framework. Future research will

look into testing the framework in an architecture environment in order to establish its effectiveness in developing and measuring agile EA.

It is important to determine the current level of agility in an organization as well as the general process and additional factors that should be taken into account when an organization analyses their EA practices and decides to adopt a more agile approach to developing EA. This research identifies a need to enable the exploration and explanation of this problem to uncover the factors that should be considered for the successful development of agile EA.

7.2.2 Factors Influencing the Choice of Research Method

This subsection of the research questions describes the factors influencing the choice of research method.

Agility within the context of EA is a relatively new area and therefore there is limited research into a method and guidelines describing how the level of agility of EA can be measured. Previous research only helped identify some of the factors influencing the development of agile EA. As mentioned previously, measuring EA agility is a relatively new area. The majority of agility measurement has been researched in the areas of software development and supply chain management. The previous research predominantly followed positivist approaches. Positivist research practices use an empirical-analytic paradigm which presumes that the research proceeds through the objective testing of hypotheses. This involves a process of deductive analysis to discover objective findings through scientific research methods (Gasson, 2004: 85).

Interpretive research rejects the very idea that one can be objective and neutral in research (Willis, 2007: 210). Instead, interpretive research assumes that the researcher participates by describing specific cases through narrative articulation and interpretation (Packer, 1999) offering a perspective that helps the understanding of a particular phenomenon (Willis, 2007: 190). This involves a process of inductive analysis to introduce subjectivity into the research so that findings are not measured, but rather observed.

The aspects of the phenomenon under investigation – the extent of understanding of EA, agility and agile EA, the methods to measure the agility of EA and the factors that affect the development of agile EA – are too complex to define and measure with standard instruments. To gain greater knowledge, interpretive research proposes a method capable of capturing

social meanings of EA, agility and agile EA; agile EA measurement techniques and the factors to take into account when developing agile EA as generated by the selected organizations. These phenomena will be understood by accessing the meanings that participants assign to them (Myers, 1997: 242).

This research does not predefine dependent or independent variables, nor does it set out to test hypotheses. It aims to produce an understanding of the social context of the phenomenon and the process whereby the phenomenon is influenced by the social context (Rowlands, 2005: 81). The possibility of an objective or factual account of events and situations is thus rejected (Rowlands, 2005: 84). This research instead seeks a relativistic, shared and deeper understanding of EA, agility and agile EA; agile EA measurement techniques and the factors involved in developing agile EA.

An interpretive and qualitative case study research strategy is thus chosen as the method best suited to investigating the extent of understanding of EA, agility and agile EA; agile EA measurement techniques and the factors involved in developing agile EA in a real-life context.

7.3 Research Method

The previous section described the research questions that are studied within this thesis. The discussion included a presentation of the research questions, previous research into EA and the preliminary framework as well as the factors influencing the choice of research method.

This section discusses the research method employed in this thesis. The discussion will include a presentation of the paradigm, unit of analysis, research instrument and the approach to analysing data.

Qualitative and interpretive research offers an approach to investigating subjects in their natural surroundings (Hunter, 2004: 292).

7.3.1 Qualitative and Interpretive

This subsection of the research method discusses the qualitative interpretive research paradigm.

Interpretive research sees human behaviour as the outcome of the subjective interpretation of the environment (Rowlands, 2005: 81). The development of agile EA is assumed not to be an

objective phenomena; with known properties or dimensions. Interpretive research is thus consistent and compatible with the epistemological assumptions that the world and reality are interpreted by people in the context of social and historical practices (Rowlands, 2005: 83). Experience of the world is subjective and best understood in terms of individual meanings rather than a researcher‟s objective definitions (Rowlands, 2005: 83).

Since researchers are closely involved with research participants in a subjective manner, a concern about researcher bias arises. Questions in an interview may be posed in a certain way or certain aspects of the discussion may be pursued more or less intensively. This flexibility is beneficial in allowing the researcher to obtain relevant data. In the end, emphasis should be placed on the research method in order to counteract the potential introduction of bias (Hunter, 2004: 292).

Rowlands (2005: 87) recommends that interpretive research should be guided by one or more social theories. This research is guided by definitions and a conceptual framework that is built on previous research. However, given that this study is based on theory building not theory testing, the definitions and framework are used solely as a guide. They help to further make sense of what occurs in the field in order to ensure that important issues are not overlooked, provided a set of provisional constraints to be investigated and guided interpretation are the focus (Rowlands, 2005: 87).

7.3.2 Case Studies

This subsection of the research method discusses the empirical inquiry known as a case study and how it will be applied in this thesis.

As this research involves exploration into a fairly new phenomenon, it is appropriate to analyse a range of cases to ensure that what is being described covers the field, at least in a preliminary way (Kelly, 1999: 381).

According to Yin (2003: 13), a case study is a method of inquiry used to investigate a contemporary phenomenon within its real-life context. The distinctive need for case studies arises out of the desire to understand complex social phenomena (Yin, 2003: 2).

Case studies focus on a particular context based on real people and situations. Data can be gathered using several techniques including observations, interviews, historical sources, journals and tests and provide a means of either confirming existing knowledge or discovering new concepts (Willis: 2007: 239). Case studies are used to gather rich, detailed

data based on human behaviour best understood as lived experiences in a social context where there is no need for predetermined hypotheses and goals (Willis, 2007: 240). These cases begin with an idea of what data will be gathered but the initial and tentative plans for data collection may change over the course of the research process (Willis, 2007:241).

A case study method is used in this research to enable an understanding of the problem, nature and complexity of defining EA, agility and agile EA, measuring the agility of EA and to determine emerging factors that influence the development of agile EA. This research will contribute to the general pool of knowledge by relating the findings from particular cases in selected South African organizations to generalized theory which can be adapted to any organization with agile EA development issues (Rowlands, 2005: 83).

The case study used in this research is characteristically rich, as empirical investigation is conducted at two levels of inquiry:

- An analysis of the extent of understanding of EA, agility and agile EA as well as the methods used to measure the agility of EA in four selected South African Organizations. This investigation forms an essential understanding of the issues related to agile EA development and measurement. A case study is thus appropriate for illustrating certain topics of evaluation in a descriptive mode (Yin, 2003: 15).

- The exploration of factors affecting the development of agile EA in these organizations. Research inquiry is based on an initial framework of factors which are modified and refined as data is collected and analysed throughout the empirical investigation. This second level of inquiry serves as the main objective of the case study, to explore the factors affecting the development of agile EA in selected South African organizations.

Since the aim of this research is to gain an in-depth understanding of the factors affecting the development of agile EA, it is appropriate to use semi-structured interviews in the investigation (Kelly, 1999: 382).

7.3.3 Unit of Analysis

This subsection of the research method discusses the organizations that will be involved in the empirical study.

Selected South African organizations are chosen to represent several case studies. The criterion for selection was that the organisations should possess a formal enterprise architecture practice. In conducting multiple case studies among these selected organizations, the aim is to expand and generalize the theory rather than enumerate frequencies (Yin, 2003: 2007). This offers a situation in which case conclusions are determined based upon similarities and differences among cases involved in the study (Hunter, 2004: 296).

Making contact with respondents and gaining their cooperation requires a process of evaluation of potential respondents to determine whether or not they are suitable (Kelly, 1999:384). As the investigation of agility within the process of EA is a new phenomenon; the only criteria for participation in the empirical study is that an organization have a formal enterprise architecture practice. Further research could investigate whether results differ in different sectors, provinces, etc. Consent should then be negotiated with the respondents. Establishing trust with the participants requires a keen political sensibility and understanding of how power relationships are structured. The parameters, objectives and methods of the research should be agreed on at the start (Kelly, 1999: 385).

Ethical consent is also required for the study. In some way, the material being covered may be of interest to outside parties and disclosure thereof may be threatening to the participants concerned (Kelly, 1999: 385). It is unethical to ask participants to participate fully without informing them of what will be done with the results of the study. Sensitive research includes research into issues where there are strong social alignments and tensions, for example, between the business and the IT people of the organizations (Kelly, 1999: 386). Participants need to, at all times, be comfortable with the level of exploration and discussion and should be approached beforehand and informed of the type of questions that are to be asked (Kelly, 1999: 387). At the request of the participating organization, a confidentially agreement can be signed to ensure the confidentiality of the material discussed.

7.3.4 The Research Instruments

This subsection of the research method discusses the research instruments that will be used to gather the data in the empirical study.

The main instrument that will be used in this research includes semi-structured interviews with systems experts at the four selected organizations guided by qualitative survey questions derived from the important findings from the literature review. Each interviewee will be interviewed once. The initial interview will be viewed as a pilot study and will be used to

refine the research instrument and preliminary framework before the empirical study is conducted with the remaining three organisations.

Interpretive researchers prefer the use of semi-structured or open interviewing methods which address the tensions between the real-life world, interview situation and the analytical framework (Willis, 2007: 245). These interviews allow participants to provide long explanations; deviations from the sequence of questions asked; and answers and opinions from other people. The researcher may suggest, agree or disagree with the answers given and interpret the meanings of the questions. The researcher is at liberty to improvise by adding question categories or making changes to the initial qualitative survey instrument where necessary (Willis, 2007: 246).

The interviews used in the case study are designed as follows:

- The type of interview: A semi-structured approach is adopted in which questions in the interview are structured but not restricted. The questions constitute an interview guide, with prompts to explore/probe for other information.
- The people to be interviewed: Systems experts at the four selected organizations in one South African province are interviewed to determine the extent of understanding of EA, agility and agile EA, the methods to measure the agility of EA and the factors for the successful implementation of agile EA within an organization.
- How the interview is conducted: Individual interviews with participants are conducted, the approximate length of which is likely 45 minutes long.
- The interview equipment:
    - Recorders: audio recorders to record the interview.
    - Writing pad, pens, pencils and highlighters: For taking notes during the interview based on each interview question.
- Consent from the participants to conduct an interview: Initially, an email (see Appendix C) or letter (Appendix D) is sent to the organization contact or head of department to request that the organization participate in the research. The email/letter explains the research, intended outcomes and what would be required of the organization should they choose participate. Thereafter, if the organization agrees to

participate and an individual from the organization has been chosen to be their representative, an email or letter is sent to the participant to inform them of the research and interview. The letter explains the purpose of the interview and the relevance of the participant‟s perspective. Other aspects discussed in the letter include the type of questions to be asked, the type of interview, the approximate length of the interview and the proposed dates of the interview. The participant is also emailed a copy of the questions (see Appendix A) to be used in the interview in order to familiarize themselves with the questions. Prior to the commencement of an interview, the participants are required to sign a consent form (see Appendix E) stating that they have been informed about the purpose and are willing participants of the interview.

In interpretive research, as and when data is gathered from the interviews, it is subsequently analysed. This method ensures that the collection, analysis and write up of data and results are all effectively integrated (Willis, 2007: 241).

7.3.5 Approach to Analyzing Data

This subsection of the research method discusses how the data gathered will be analysed.

The theoretical orientation of the case study that relies on the application of the conceptual framework shapes the data collection plan and hence guides the analysis of data.

This research follows an inductive approach to capture the interpretive experiences of participants and develop theoretical propositions from them. This method has been effectively used in recent Information Systems (IS) research to develop the theory of IS practice (Rowlands, 2005:87).

The data gathered from interviews is used to develop conceptual categories or to illustrate, support or challenge theoretical assumptions held prior to data gathering (Willis, 2007: 243). Elements of the data transcribed from interviews are coded into categories of what is being observed (Gasson, 2004: 81).

Patterns and relationships between categories are then identified, followed by a write up of initial ideas and interpretations concerning cross-category insights (Gasson, 2004: 82). These categories may then be refined as necessary. This entire process is iterative, constantly cycling between coding, synthesis and data collection.

Decisions about what data to collect next and where to find it are made according to the researcher‟s theory development needs (Kelly, 1999: 382). The literature review is used as a tentative theory basis from which to collect more data to test this theory. If the first collection of data extracted fits the previous theory, the researcher moves on to the next data collection. However, if the data extracted does not bare any similarities with previous research, the researcher modifies the theory so that it fits with the original as well as the new data which is checked against the collection of more data, and so on. The goal is to build a theory that fits with every set of data extracted and which can then be generalized (Willis, 2007: 89).

The closure of this theory is guided by the concept of saturation which is reached when no new themes, categories or relationships emerge when collecting more data. Finally, formal theories from the data analyses are developed (Gasson, 2004: 84). The process ensures that results are simple, meaningful, broad, explanatory, generalized and internally consistent (Willis, 2007: 308).

Conclusions can be drawn from multiple sources of confirmation. Member checks are used to check emerging conclusions with participants involved in the case studies. Participatory research allows the active participation of participants in formulating research conclusions. Extended experience in the environments follows the notion that the more the researcher experiences the environment in each of the cases, the better the researcher understands the dominant topics under investigation. Peer reviews are used to attain the opinions of colleagues about emerging conclusions (Willis, 2007: 220). Researcher journaling may be used to analyse the researcher‟s reflective views made during data collection and analysis (Willis, 2007: 221). Finally, audit trails may also be used to document the research process from the gathering of data to the final write up. A record is kept of when ideas emerged along with the supporting data and how these ideas were refined and expanded (Willis, 2007: 221).

This research methodology is designed to guide the collection and analysis of data to present an exploration of the extent of understanding of EA, agility and agile EA, the methods to measure the agility of EA and the various factors that affect an organizations ability to develop agile EA.

7.4 Research Design

The previous section discussed the research method used in this thesis. The discussion included a presentation of the research paradigm, unit of analysis, research instrument and the

approach to analysing data. This section summarizes the research design discussed this chapter.

The research design for theory building which is followed in this study is adapted from Yin (2003) and Rowlands (2005: 88) and uses an inductive approach to multiple case studies.

7.4.1 Phase 1 – Define and Design

- *Define study area, describe questions and create conceptual model* - In this phase an initial definition for EA, agility and agile EA and a start list of factors that affect the development of agile EA is drawn from the preliminary framework. Since this research is non-linear, it does not proceed in a number of predetermined steps so theory and questions may change in unpredictable ways (Willis, 2007: 202).

- *Identify participating organizations and select case* - The participating organizations are gathered from the four selected South African organizations. These organizations represent the four case studies under investigation.

- *Design data collection protocol and instruments to be used* – Questions that prompt respondents for their definition of EA, agility and agile EA; the methods used to measure the agility of EA in their organizations and key concepts extracted from the preliminary framework are transformed into an open qualitative survey which will be used as a guide in semi-structured interviews with participants in each of the four cases.

- *Conduct pilot case* - A pilot study is conducted with a participant from one of the cases, who assists in the preliminary testing of the definitions developed from the literature review as well as the framework, which will then be refined and modified as necessary. A second conceptual framework is developed based on the previous literature as well as the emergent data identified from the pilot study (Rowlands, 2005: 88). A pilot case helps to refine data collection plans with respect to both content of the data and the procedures to be followed (Yin, 2003: 79). This process is based on a relatively unstructured approach in preparation for a more structured approach to be taken in the main study (Kelly, 1999: 394).

7.4.2 Phase 2 – Data Collection and Case Analysis

- *Conduct case studies in the field* – The refined framework and definitions are used to adjust the qualitative survey instrument which is then used in the collection of data through semi-structured interviews with participants in each of the remaining three organizations. The study may involve periodic interviews interspersed with observations so that the researcher can question the subjects and verify perceptions and patterns (Willis, 2007: 208).

- *Write individual case reports and analyse data* – Once data collection in the field is complete, it is analysed in each of the cases and reports are written identifying the emerging categories of data. Working with data during collection allows for emerging insights, hunches and tentative hypotheses which direct the next phase of data collection (Willis, 2007: 202).

7.4.3 Phase 3 – Cross-Case Analysis

- *Analyse and draw cross-case conclusions* – Patterns and similarities across all cases are identified, after which a number of implicit conclusions about emerging results are determined.

- *Shape propositions, confirm and sharpen emerging theory* – The iterative process of extracting broad categories and concepts that describe conditions, events, experiences and consequences is used to provide empirically valid accounts of unique data and generalized patterns. Propositions are then made from the analysis of the emergent categories (Rowlands, 2005: 89).

- *Build theory and transferability to generalized cases* – The propositions and discoveries are discussed in relation to literature to note consistencies with and discrepancies from earlier findings (Rowlands, 2005: 89).

- *Reach closure* – Any conclusions are made with the context fully in mind (Willis, 2007: 222). The extent to which each of the propositions is supported by previous research is indicated as well as the extent to which the research has added some new perspective (Rowlands, 2005: 89).

7.5 Conclusion

In this research, data analysis follows an interpretive and qualitative approach based on case studies through a number of interviews with systems experts in four South African organizations in one province. The interviews are guided through the exploration of the extent of understanding of EA, agility and agile EA, the methods to measure the agility of EA and the factors affecting the development of agile EA in these organizations. This research follows an inductive approach to capture the interpretive experiences of participants and develop theoretical propositions from them.

This chapter presented the research methodology which provides a guideline to effectively collect and analyse data in order to create a comprehensive definition for EA, agility and agile EA as well as identify methods for the measurement of agility of EA and a framework of factors that affect the development of agile EA.

The next chapter will present the exploratory pilot study conducted with a participant from one of the cases who assists in the preliminary testing of the definitions developed from the literature review as well as the framework.

Chapter 8: Exploratory Pilot Study

Chapter seven presented the research methodology adopted to explore the extent of understanding of enterprise architecture, agility and agile EA, the factors for the successful development of agile EA as well as techniques for the measurement of agility of EA within organizations.

The discussion included the research questions, research method, unit of analysis, research instrument, approach to analysing data and the research design.

The aim of this chapter is to present the details of an exploratory pilot study conducted through an interview with a systems expert at organization # 1. The extent of understanding of enterprise architecture, agility and agile EA, the methods used to measure the agility of EA as well as the factors affecting the development of agile EA at the organization are explored.

The results of this exploratory study form the basis of reflections and enhancements to the preliminary framework before a comprehensive empirical investigation is conducted in all four organizations.

8.1 Introduction

This section provides an introduction to the exploratory pilot study.

Comprehensive definitions for EA, agility and agile EA were created in chapters 2, 3 and 4 respectively. The factors affecting the development of agile EA and a measurement method are identified in a preliminary framework in chapter 6. In order to further analyse and explore the extent of understanding of enterprise architecture, agility and agile EA, the methods used to measure the agility of EA as well as the factors for the development of agile EA, an exploratory pilot study is conducted. The study involves a semi-structured interview with an individual who knows most about the systems used at organization # 1.

This chapter presents a brief design of the qualitative survey used in the exploratory study and its results. This is followed by an analysis of the results and a summary of the exploratory pilot study.

8.2 Survey Design

The previous section provided an introduction to the exploratory pilot study. This section describes the design of the qualitative survey used within the exploratory study.

The survey is divided into six sections based on the preliminary framework of the factors affecting the development of agile EA. Section A examines the EA Foundation Stage which involves the preliminary steps the organization takes before attempting to develop EA. Section B examines the EA Approach Stage which includes the steps to develop agile EA and occurs if the organization decides to commence on an EA program, based on conclusions from the foundation stage. Section C examines the Extension and Management Stage in which the organization‟s EA team extends and maintains the architecture. Section D examines the EA Governance and Management Stage which focuses directly on how the organization establishes and implements EA as well as on setting the policy for how EA subsequently should be run. Section E examines the EA Maturity and Measurement Stage in which the organization tests the maturity of the current EA and measures the value added by their efforts. Section F details the demographics of the respondent and the institution. Section F was chosen as the last set of questions to be asked in order to ensure that the respondent is occupied with the most important questions of the survey first, thus supplying enthusiastic opinions and answers. The survey also contains a section where the respondent will be asked to make any further comments relating to the survey questions, research, etc.

8.3 Survey Delivery

The previous section described the design of the survey. This section describes the means by which the survey was employed.

An interview at the system expert‟s office at the organization was conducted. Audio recording equipment and a copy of the questionnaire were used during the interview.

8.4 Results

The previous section described how the survey was delivered. This section describes where the results of the exploratory study can be found.

Should they be of interest to the reader, in-depth responses to each of the interview questions are detailed in Appendix B, under the heading, “Organization #1.”

8.5 Analysis of the Results

The previous section described where the results of the exploratory pilot study could be found. This section provides a description of the response rate and an analysis of the results obtained from the responses from the representative of organization # 1.

8.5.1 Response Rate

This subsection of the analysis of the results describes the type of individual that the information required in the study can be obtained from.

The respondent noted that the information requested in the survey can only be obtained by senior members of an organization‟s IT function involved in the development of the EA. The EA team consists of enterprise architects who are the most senior IT staff, reporting directly to the Chief Information Officer of an organization.

Fortunately the respondent is a senior member of the organization and has the role of Principal consultant on Enterprise Strategy. Therefore, based on the preliminary framework, sufficient information was extracted from the respondent at organization # 1.

8.5.2 Summary of the Exploratory Pilot Study

This subsection provides a summary of the exploratory pilot study.

The exploratory pilot study provided interesting and valuable insight into the issues related to agile EA development. As this research attempts to uncover the answers to specific questions, the analysis of the results can be grouped under the following headings:

- *The extent of understanding of enterprise architecture at organization # 1*
- *The extent of understanding of agility and agile EA at organization # 1*
- *The methods used to measure the agility of EA at organization # 1*
- *The factors affecting the successful implementation of agile EA at organization # 1*

8.5.2.1 The extent of understanding of enterprise architecture at organization # 1

This part of the summary of the exploratory study summarizes and analyses the results from the respondent of organization # 1, with respect to the extent of understanding of enterprise architecture.

From the respondent‟s definition of EA, it is clear that the EA team believe that enterprise architecture is a far reaching, broad description of all of the key elements and relationships that make up an organization. However from the definition as well as the suggestions made by the respondent to change the research questions to focus more on architecture deliverables, it is clear that those involved in the EA effort tend to view EA as noun. It has been found that when people focus on EA as a noun, they focus on the outputs and are more concerned about producing a predefined set of deliverables, rather than about meeting the strategic imperatives of the enterprise (Gartner, 2008a: 2).

8.5.2.2 The extent of understanding of agility and agile EA at organization # 1

This part of the summary of the exploratory study summarizes and analyses the results from the respondent of organization # 1, with respect to the extent of understanding of agility and agile EA.

The respondent of organization # 1 indicated that the term, „agility‟ envisioned something responsive, or swift and within the context of EA; an EA that was adaptable or could be changed easily if a change was required. Like the term „enterprise architecture,‟ the meaning as well as a shared understanding of agility is achieved through the definition workshop that runs prior to any work being done on the EA project. However, since the respondent indicated that their view of EA was that it relates that to the artifacts produced by the process of architecting and not that EA relates not only to the artefacts produced but also to the process of producing them; then this adaptability relates to how easily the architecture created by the architects can be changed.

8.5.2.3The methods used to measure the agility of EA at organization # 1

This part of the summary of the exploratory study summarizes and analyses the results from the respondent of organization # 1, with respect to the methods used to measure the agility of EA.

The respondent indicated that the organization uses various methods to measure the agility of the created architecture. The organization makes use of the methods that have been made

available in the established enterprise architecture frameworks (Zachman, TOGAF, GARTNER, etc.).

As mentioned previously, these methods focus on measuring the agility of the end product created by the process of EA. The organization does not possess any methods for measuring the agility of the process of creating the architecture.

8.5.2.4The factors affecting the successful implementation of agile EA at organization # 1

This part of the summary of the exploratory study summarizes and analyses the results from the respondent of organization # 1, with respect to the factors for the successful development of agile EA. The responses have been grouped according to the five sections (Sections A, B, C, D and E) based on the survey designed to investigate the preliminary framework of the factors affecting the development of agile EA:

- *Section A - EA Foundation Stage*

The following paragraphs provide a summary analysis of the results obtained from the responses about the EA Foundation Stage from the representative of organization#1.

Prior to embarking on an EA project, the organization creates a needs and cost-benefit analysis and statement of work for the architecture which takes into account financial, human and other aspects of the project.

Both business and IT are involved in deciding whether a project is worth undertaking. A need is discovered in most cases from a business point of view, however IT can propose certain project undertakings from their standpoint, for example; a project to reduce redundancy in the IT function therefore making them more efficient.

Top management review the project proposal and thereafter conduct a value realization; signing off on project worthy proposals. The EA team achieve a common understanding of EA, agility and agile EA in a definition workshop which runs prior to any work being done on the project. Management communicates their vision for the project and clearly defines the expected results; initially through a needs and cost-benefit analysis and then later on in a definition workshop which runs prior to any work being done; and serves to inform project members of the project objectives and achieve a common understanding of project concepts. They ensure that sufficient communication structures are established that will be used

throughout the project and encourage weekly status meetings and the use of collaboration software tools where face to face collaboration is not possible. Management defines the way the EA project is to be run and the EA team is required to make regular check-ins of objectives with management in the weekly status meetings.

Those involved in the EA project within the organization have a common understanding of the meaning of enterprise architecture as well as agility and other concepts related to the EA project through the definition workshop conducted prior to any work being done.

- *Section B - EA Approach Stage*

The following paragraphs provide a summary analysis of the results obtained from the responses about the EA Approach Stage from the representative of organization#1.

A project sponsor or executive is elected for the project. He/she will be responsible for driving the success of the project. An architecture review board comprising of senior members of the organization (architects, CIO, etc.) is elected to ensure that top management is involved in high-level decisions of the EA project. This board enforces architectural policies and standards; the decisions on the EA project are communicated to business afterwards.

The EA team members are committed to the success of the project. Their commitment is as a result of the members of the team being senior IT staff within the organization. These staff members have reached that position not only due to their qualifications and experience but also because of their high level of motivation and commitment to the organization and its goals and their ability to motivate and encourage commitment from others.

The EA team is provided with a clear scope in which to work and is required to show the value added by each project iteration/release; throughout the lifecycle of the project. The EA team is held accountable for meeting the deadlines, goals and results of the overall project. The artifacts produced are presented to management and their value realized soon afterwards. A review is conducted following a project iteration to ensure that the project is still in line with its requirements. Thereafter, ways to improve the project are discussed and if any are present, they are added to the project requirements.

The organization places a great emphasis on face to face team collaboration as this fosters a better exchange of ideas and collaboration as opposed to collaboration over software tools.

However, use of these tools is encouraged in instances where face to face meetings are not possible and communication is required. The EA team communicates not only amongst themselves using these methods, but also with the project sponsor/executive responsible for the project throughout its lifecycle.

Within the EA team, the Chief architect is the main person responsible for the architecture and supervises and delegates responsibilities to the other members of the team. Leffingwell, *et al.*(2008: 2), suggest that this is in contradiction with agile principles; as since agile focuses on harnessing the power of the collective team, rather than any one individual. The system architect should no longer dictate the technical direction.

Where possible, the EA team reuses existing architecture artifacts. However, in some cases, the artifacts have become obsolete and part of the project work will be to replace them entirely.

In developing the architecture models the EA team develops a combination of event based strategies that will depend on the occurrence of certain events as well as set strategies for specific areas using well known industry frameworks/models, for example, the Mckinsey Model. Where possible, the EA team creates several architecture models in parallel. This depends on the situation (Technical model can be created in parallel with a reference model for applications). At a logical level this is possible, however at a physical level it is not as the two are interdependent. The EA team also makes an effort to continuously build out the system infrastructure by ensuring that part of the solution design is infrastructure design.

The work to be done on a project is broken down into manageable tasks and a schedule drawn up to indicate the work to be completed in a given period. This allows the architecture to be developed in increments.

Additional features are added to the architecture in addition to the requirements of the EA project to cater for events that the EA team believe can occur, in a hope to make the architecture robust and handle future changes. However, according to Leffingwell, *et al.*(2008: 2), technology should be cut to the minimum needed to solve the problem at hand. The organization is not aided by creating artificial and unneeded layers of complexity. Ambler (2010) adds that architecture must be based on requirements and the details should be identified on a just-in-time (JIT) basis during iterations via initial iteration modelling at the beginning of each iteration, or by modelling storming throughout the iteration. The end result

is that architecture emerges over time in increments, faster at first because of the greater need to set the foundation of a project, but still evolving over time to reflect the greater understanding and knowledge of the development team. This follows the practice to model in small increments and reduces the technical risk of the project. The development team always has a firm and proven foundation from which to work. In other words, the development team should think about the future, but wait to act (Ambler, 2010).

Ambler (2010) views a model as merely an abstraction, one that may appear to be very good but may not actually be in practice. Agile, with its highly iterative experience and code-based emphasis, allows developers to simply rely on their coding skills to move efficiently through the decision-making process. This is helpful when selecting a design alternative or a high-impact infrastructure implementation choice. However, the developers at organization # 1 believe that using code to evaluate design alternatives would most likely result in the EA project not meeting its deadline. As a result, they choose not to do so.

The EA team that designs the system does not code the system. The EA team comprises of architects whose sole responsibility is to derive functional specifications (use case diagrams, etc.). The EA team then delivers the designs to a development team within the organization and in some cases this function is outsourced to an external organization. However, Leffingwell, *et al.*(2008: 2) believe that in an agile approach, the team that designs the system should be responsible for coding it.

In organization # 1, the EA team is given freedom to make decision relating to the EA project as at times the models developed do not work in practice. Teams themselves should be empowered to define, develop and deliver software, and they are held accountable for the results.From a management perspective, in order for teams to be held accountable, they must be allowed to make the decisions required to support that accountability. If not, they will be held accountable for decisions made by others, and that is an ineffective and de-motivating model for team performance (Leffingwell, *et al.,* 2008: 3).

The developers within organization # 1 are not permitted to officially test their own code. They conduct minor tests to ensure that the code delivers the intended results; however official testing is outsourced to an independent testing team. Leffingwell, *et al.* (2008: 5) believe that because testing represents complexity at its highest level, the team that codes the system should be the team that determines how to test the system. It should be the

responsibility of the development teams to develop, test; and maintain a system-testing framework that continually assesses the system‟s ability to meet its architectural and functional requirements. This responsibility should not be given to any other testing resource or outsourced function.

- *Section C - Extension and Management Stage*

The following paragraphs provide a summary analysis of the results obtained from the responses about the Extension and Management Stage from the representative of organization#1.

After the completion of a project the EA team has a set of policies which dictate how to extend and make changes to the current architecture laid out in the EA teams‟ change management policy document.

- *Section D - EA Governance and Management Stage*

The following paragraphs provide a summary analysis of the results obtained from the responses about the EA Governance and Management Stage from the representative of organization#1.

The governance policies that are established at the project‟s inception are carefully derived by management to enable effective governance. These policies are reviewed at certain intervals to ensure that they are still effective, do not hinder the project progress of the EA team and are changed if the need arises. In some instances, the EA team is given freedom to make project decisions as sometimes the theoretical framework does not work in practice. However these changes should follow the processes declared in the change management policy document.

- *Section E - EA Maturity and Measurement Stage*

The following paragraphs provide a summary analysis of the results obtained from the responses about the EA Maturity and Measurement Stage from the representative of organization#1.

At the beginning of the project the EA team evaluates the current IT architecture by doing a 360 assessment. This assessment includes, but is not limited to uncovering what has been done and what needs to be done in order to obtain a baseline. The organization uses various methods to measure the agility of the created architecture. The organization makes use of the

methods that have been made available in the established enterprise architecture frameworks (Zachman, TOGAF, GARTNER, etc.).Throughout the lifecycle of the project, the EA team is required to show measurable results after each project iteration and present these results at the project status meetings.

When the respondent from organization # 1 was prompted for any other additional factors that could affect the development of agile EA, no additional factors could be uncovered.

The organization‟s view of enterprise architecture as a noun might lead those involved in the EA effort to be more focused on the outputs and be more concerned about producing a predefined set of deliverables, rather than about meeting the strategic imperatives of the enterprise. Because of this definition, although agility is defined correctly within organization # 1, the agility of EA within organization # 1 now refers to the agility of the deliverables as opposed to also focusing on the agility of the process of EA which results in those deliverables.

The pilot study uncovered that the EA effort at organization # 1 would benefit from a more agile approach to developing their EA, firstly; in terms of harnessing the power of the collective team as opposed to one individual (The Chief Architect) dictating the direction of the project.

The EA team could also benefit from the addition of developers who were part of the EA process and provided input on design decisions that were made by the architects; as opposed to those designs being handed to them for coding once they had been developed solely by the architects. Those developers should also be responsible for testing, and maintaining a system-testing framework that continually assesses the system‟s ability to meet its architectural and functional requirements.

The developers could also consider using their coding skills to make more informed decisions between design alternatives or infrastructure choices. Ambler (2010) suggests that is helpful when selecting a design alternative or a high-impact infrastructure implementation choice as a model is merely an abstraction, one that may appear to be very good but may not actually be in practice.

Although the addition of features to the architecture designs is an attempt made by the architects to cater for events they believe may occur, Leffingwell, *et al.* (2008: 4) believe that

the addition of features that are not stipulated in the requirements for the system, lead to unnecessary layers of complexity. Ambler (2010) adds that architects should have the courage to focus on solving today"s problem today and trusting that they can solve tomorrow"s problem tomorrow and the humility to recognize that they cannot accurately predict the future and therefore choose not to overbuild their architectures.

When prompted for comments or recommendations on the survey and questions, the respondent suggested that the questions differentiate between the different levels of architects e.g. domain, solution architects, asked questions that addressed architecture activities at a process level and focused more on architecture deliverables.

It is the view of the researcher that it is not necessary to differentiate between the levels of architects, as what is of importance is whether the criteria in the framework are met and not who is actually responsible for doing them. In addition, responsibilities would most likely vary from organization to organization and might in some cases depend on the methodology being used. A differentiation between the levels of architects might therefore lead to a misrepresentation of an organization"s practices.

Addressing architecture activities at a process level would require a great level of detail. As mentioned previously in chapter 6, the framework created intended to be a meta process methodology. Meta process methodologies do not describe specific development approaches, but rather focus on general procedures to support the development team in establishing a project-specific approach. The framework created in this research is intended to be an architecture framework agnostic, high level structure of factors/considerations that an organization would take into account when developing agile EA. Addressing architecture activities at a process level might lead to the framework being dependent on some of the available architecture frameworks as it would be most likely these processes in the frameworks that would need to be addressed. In addition, the organizations that were contacted and agreed to participate in the research did so with the assurance that the research would not investigate their activities at a low level, thereby uncovering their inner workings or trade secrets.

The respondent indicated that the framework should also focus on the architecture deliverables. As the definition for EA used throughout this research focuses on EA as a verb as opposed to a noun, the focus of the study is on the process of EA that will result in a set of deliverables as opposed to focusing on the deliverables.

The respondent also indicated that an executive body and project sponsor served different roles and should therefore be separate within the framework. The survey questions will also be adjusted to show this change.

When delivering the survey it was apparent that some of the questions did not extract enough information from the user. Part of this thesis work is to uncover the extent of understanding of both EA and agility and how agility fits within the context of EA. The questions relating to EA and agility only prompt for whether the organization has a common understanding of EA and agility. A valid response in both cases would be a simple "yes" or "no." However, it is important to not only uncover whether everyone involved with the EA project has a common understanding of EA and agility, but to also define the organization‟s view of both EA and agility, as well as enquire whether the organization understands how agility fits within the context of EA.

Fortunately, although the survey questions were inadequate, the semi-structured nature of the research interview allows a researcher to prompt the respondent(s) for further information where necessary. When the respondent was asked whether the organization has a common understanding of EA and agility, they were also asked to provide a description of what they believe EA and agility to be as well present their view of whether and how they believe agility first within the context of EA. It is however necessary to update the survey questions to prompt future respondents for these explanations.

A redundancy was uncovered in the framework and survey questions during the survey delivery. The EA Governance and Management Stage within the framework includes a step in which management should set the policy for how EA should be run. This step is already conducted in the EA Foundation stage in which management should develop governance structures. The step to "set the policy for how EA should be run," will need to be removed from the EA Governance and Management Stage in the framework and the corresponding question removed from the survey.

The exploratory pilot study provided an understanding of the view of "enterprise architecture" "agility," as well as the view of how agility fits within the context of EA at organization # 1; with an exploration of the factors that affect the development of agile EA and methods used to measure the agility of EA.

No additional factors could be uncovered that would affect the development of agile EA and no comments were made about the lack of applicability of any of the factors investigated. The necessary changes will be made to the questions within the survey based on their ability to extract the required information and as per the suggestions of the respondent of organization # 1, in order to make them clearer and provide more accurate results in later investigations.

8.6 Conclusion

This chapter presented the exploratory pilot study conducted with an interview with the respondent from organization # 1, to investigate the extent of understanding of enterprise architecture, agility and agile EA, the methods used to measure the agility of EA as well as the factors affecting the development of agile EA. The discussion included the survey design, survey delivery, the results and their analysis.

The next chapter will present the changes to the preliminary framework for the development and measurement of agile EA, as a result of the findings from the exploratory pilot study.

Chapter 9: Theoretical Framework

Chapter eight presented the details of an exploratory pilot study conducted through an interview with a systems expert at organization # 1.

The discussion included an exploration of the extent of understanding of enterprise architecture, agility and agile EA, the methods used to measure the agility of EA as well as the factors affecting the development of agile EA at the organization.

The aim of this chapter is to present the adjustments to the preliminary framework based on the results from the exploratory pilot study and proposes a new theoretical framework that identifies the key factors that affect the development of agile EA in an organization.

It was concluded that the revisions made to the preliminary framework emphasize that a project sponsor and executive body serve separate functions of equal importance within the development of agile EA and that the establishment of an efficient project governance structure is conducted prior to any work being done on the EA project. The theoretical framework developed represents a synthesis of the various components required for the successful development of agile EA after an examination of the practices of organization # 1.

9.1 Introduction

The aim of this section is to provide an introduction to the theoretical framework to be developed.

Following the preliminary framework introduced in chapter 6 and the results of the exploratory pilot study presented in Chapter 8, a new theoretical framework is developed.

This chapter provides a discussion of the adjustments made to the preliminary framework based on the observations made in the exploratory pilot study. This will be followed by the proposal of a theoretical framework of the factors that affect the development of agile EA.

9.2 Revisions to the Framework

The previous section provided an introduction to the theoretical framework to be developed. This section will discuss the revisions to the preliminary framework developed in chapter 6 based on the results from the exploratory pilot study. Initially, the changes will be shown as

they apply to the different stages of the framework. Thereafter, the complete theoretical framework of factors for the development of agile EA will be presented.

Initially, the preliminary framework placed the election of a project sponsor and executive body within the same step in the EA Approach stage as illustrated in Figure 25.

Figure 25: The EA Approach Stage

EA APPROACH STAGE
Elect project sponsor, executive body ***(1)***
EA Team should commit on behalf of themselves ***(5)***
EA Team should be accountable for results ***(5)***
Focus on individuals and interactions over processes and tools ***(5)***
Harness power of collective team rather than an individual ***(5)***
Customer collaboration over contract negotiation ***(5)***
No monopoly on innovation ***(5)***
Define EA scope ***(1)***
Reuse existing architecture artifacts***(5)***
Develop situational specific strategies, processes, practices ***(5)***
Determine schedule in terms of iterations and releases ***(5)***
Team that codes the system designs the system ***(5)***
Create several architecture models in parallel ***(5)***
Develop architecture in increments ***(5)***
Build simplest architecture that can work ***(5)***
When in doubt, code it out ***(5)***
Whoever builds it should test it ***(5)***
Continuously build out system infrastructure ***(5)***

(Own contribution)

As described in the previous chapter, the sponsor of a project is the person who provides the budget for and champions the project. They are usually the head of the department who will receive whatever the project is attempting to deliver. The executive body is elected to ensure that the management is involved in the high level decisions of the project. In addition, the project sponsor may or not be part of the executive body elected and therefore, the two should be separate within the framework, but are of equal importance as illustrated in Figure 26 on the next page which shows the amended EA Approach Stage.

Figure 26: The Amended EA Approach Stage

EA APPROACH STAGE
Elect project sponsor ***(1)***
Elect executive body*(1)*
EA Team should commit on behalf of themselves ***(5)***
EA Team should be accountable for results ***(5)***
Focus on individuals and interactions over processes and tools ***(5)***
Harness power of collective team rather than an individual ***(5)***
Customer collaboration over contract negotiation ***(5)***
No monopoly on innovation ***(5)***
Define EA scope ***(1)***
Reuse existing architecture artifacts***(5)***
Develop situational specific strategies, processes, practices ***(5)***
Determine schedule in terms of iterations and releases ***(5)***
Team that codes the system designs the system ***(5)***
Create several architecture models in parallel ***(5)***
Develop architecture in increments ***(5)***
Build simplest architecture that can work ***(5)***
When in doubt, code it out ***(5)***
Whoever builds it should test it ***(5)***
Continuously build out system infrastructure ***(5)***

(Own contribution)

A redundancy was uncovered in the preliminary framework during the survey delivery. The EA Foundation stage contains a step in which management should develop governance structures as illustrated in Figure 27 below.

Figure 27: The EA Foundation Stage

EA FOUNDATION STAGE
Needs and cost benefit analysis ***(1)***
Top management buy-in ***(1)***
Common EA definition ***(1)***
Common agility and agile EA definition ***(1)***
Management should communicate vision ***(1)***
Management should define goals ***(1)***
Management should define expected results ***(1)***
Management should engage in communication planning ***(1)***
Management should develop governance structures ***(1)***

(Own Contribution)

This step is repeated in The EA Governance and Management Stage within the framework in the form of a step in which management should set the policy for how EA should be run as illustrated in Figure 28 below.

Figure 28: The EA Governance and Management Stage

EA GOVERNANCE AND MANAGEMENT STAGE	
Eliminate impediments for the EA team ***(5)***	Management should be accountable for empowering teams to deliver ***(5)***

(Own contribution)

The step to "Set the policy for how EA should be run," will need to be removed from the EA Governance and Management Stage in the framework as illustrated in Figure 29 below which shows the amended EA Governance and Management Stage.

Figure 29: The EA Governance and Management Stage

EA GOVERNANCE AND MANAGEMENT STAGE		
Eliminate impediments for the EA team ***(5)***	Set policy for how EA should be run ***(1)***	Management should be accountable for empowering teams to deliver ***(5)***

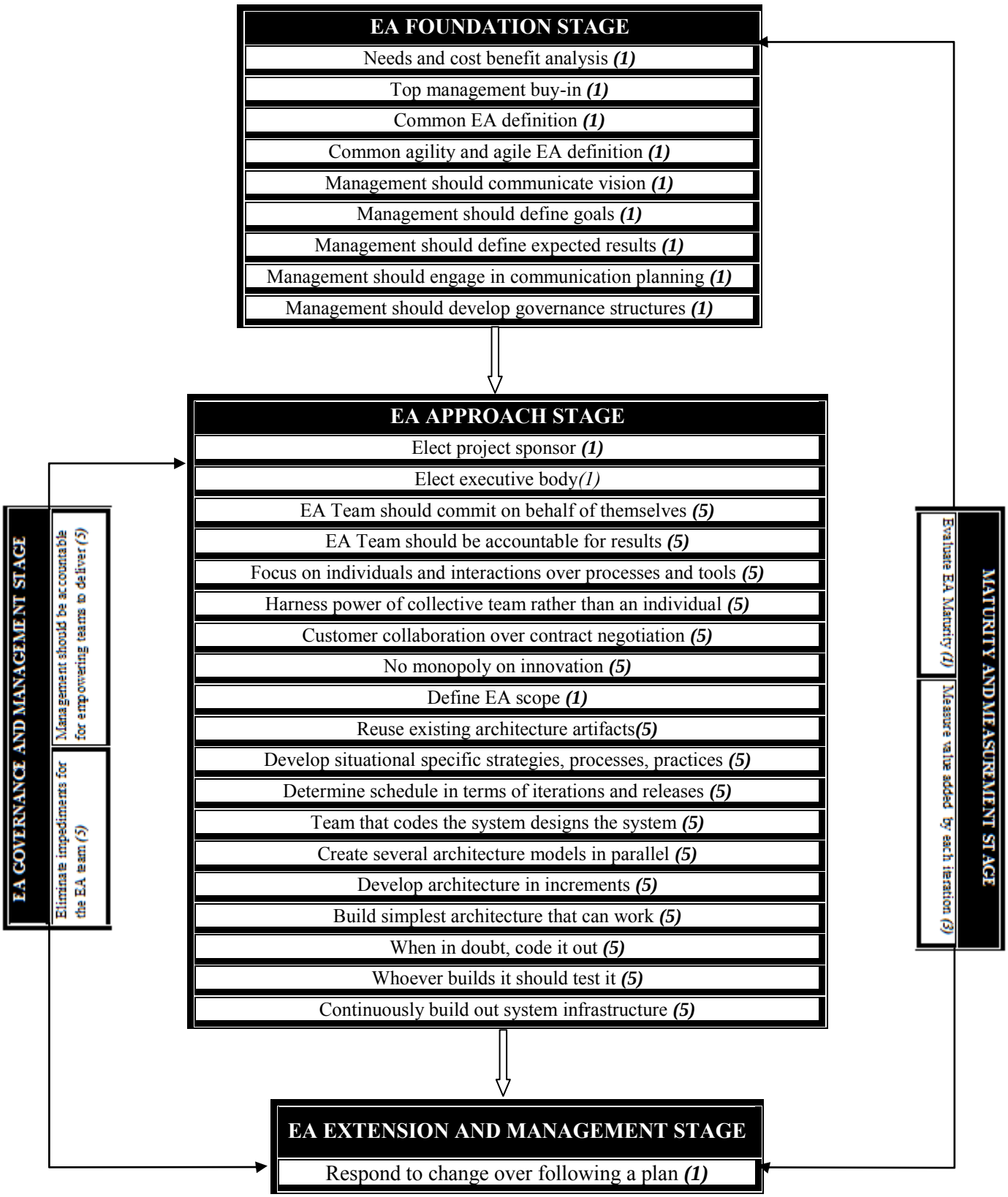
(Own contribution)

The changes made to the preliminary framework have not affected the overall score of agility that an organization can achieve. The removal of a step within the EA Governance and Management Stage of the framework with an agility score of 1 and addition of another within the EA Foundation Stage with an agility score of 1 means that the maximum agility score that an organization can achieve is still 106. There is therefore no need to make any changes to the Agility Scoring Scale presented in chapter 6.

Although they are not shown here, some of the comments from the respondent of organization # 1 prompted a change in the survey instrument. The survey questions have been updated accordingly.

Figure 30 on the next page provides an illustration of the complete theoretical framework for the measurement of agility of EA; showing the changes made to the stages discussed above and how all the stages are connected to one another

Figure 30: Theoretical framework for the Development and Measurement of Agile Enterprise Architecture



(Own contribution)

9.3 Conclusion

The theoretical agile EA development framework proposed in this chapter represents a synthesis of the various components required for the successful development of agile EA.

Revisions made to the preliminary framework emphasize that a project sponsor and executive body serve separate functions of equal importance within the development of agile EA and that the establishment of an efficient project governance structure is conducted prior to any work being done on the EA project.

This chapter presented the adjustments to the preliminary framework based on the results from the exploratory pilot study and proposed a new theoretical framework that identified the key factors that affect the development of agile EA in an organization.

The next chapter will present the design of the design of the empirical study to be conducted through interviews with the systems experts from the remaining three organizations.

Chapter 10: Design of the Empirical Study

Chapter nine presented the adjustments to the preliminary framework based on the results from the exploratory pilot study and proposed a new theoretical framework that identified the key factors that affect the development of agile EA in an organization.

The discussion included a presentation of the changes as they applied to the different stages of the framework. Thereafter, the complete theoretical framework of factors for the development of agile EA was presented.

The aim of this chapter is to detail the design of the empirical study. The empirical study is intended to further explore the extent of understanding of enterprise architecture, agility and agile EA, the methods used to measure the agility of EA as well as the theoretical framework proposed in Chapter 9. The design of the survey instrument is discussed in detail. A full copy of the survey is attached in Appendix A.

It was concluded that following the exploratory pilot study, it became necessary to make changes to the survey instrument created in chapter 7. This led to the key factors for the successful development of agile EA that were identified in preceding chapters being categorized into five subsections in the first part of the survey (A.1 – A.5). The survey also includes a section in which the respondent is allowed to make further comments (B.1.1-B.1.5). The final section of the survey (C1 – C.2) extracts demographical data from the respondent.

10.1 Introduction

The aim of this section is to provide an introduction to the design of the empirical study.

The preceding chapters identify and describe various factors that affect the development of agile EA. These factors are broadly categorized into an EA Foundation Stage, an EA Approach Stage, an EA Extension and Management Stage, an EA Governance and Management Stage and an EA Maturity and Measurement Stage.

In order to further explore the factors for the development of agile EA, the extent of understanding of enterprise architecture and agility; and the methods used to measure the agility of EA at organizations, an empirical study is conducted.

The empirical study comprises semi-structured interviews with systems experts at four selected organizations in one South African province. This chapter describes the final survey design including its structure and context. Each of the sections in this survey is based on previous literature and a review of the exploratory pilot study in Chapter 8.

10.2 Survey Design

The previous section provided an introduction to the design of the empirical study. This section describes the design of the survey used within the empirical study.

The survey is divided into three sections. Section A examines and analyses the factors for the successful development of agile EA. Section B enquires whether the respondent‟s organizations has any methods that they use to measure the agility of EA and allows the respondent to comment on the questions; whether they were easy to understand and relevant to the topic as well as the framework and research in general. The respondent is also encouraged to make suggestions that would make the survey better, as well as suggest other factors that have been omitted that would encourage the development of agile enterprise architecture. Section C details the demographics of the respondent and the institution.

10.2.1 Section A – The Factors for the Successful Development of Agile EA

This subsection discusses the first part of the survey which examines the factors for the successful development of agile EA.

Section A of the survey is divided into 6 stages based on the preliminary framework of the factors affecting the development of agile EA. Stage 1 (A.1.1 – A.1.9) examines the EA Foundation Stage which involves the preliminary steps the organization takes before attempting to develop EA. Some of the survey questions within this stage have been modified from the original survey questions used in the pilot study to allow the respondent to provide definitions for EA, agility and agile EA. Stage 2 (A.2.1 – A.2.18) examines the EA Approach Stage which includes the steps to develop agile EA and occurs if the organization decides to commence on an EA program, based on conclusions from the foundation stage. Stage 3 (A.3.1) examines the Extension and Management Stage in which the organization‟s EA team extends and maintains the architecture. Stage 4 (A.4.1 – 4.2) examines the EA Governance and Management Stage which focuses directly on how the organization establishes and implements EA. Stage 5 (A5.1 – A.5.3) examines the EA Maturity and Measurement Stage in

which the organization tests the maturity of the current EA, measures the value added by their efforts.

The structure and layout of Section A is outlined in Table 4 below.

Table 4: Section A – The Factors for the Successful Development of Agile EA

A.1	EA Foundation Stage	
	A.1.1	Needs and cost benefit analysis
	A.1.2	Top management buy-in
	A.1.3	Common EA definition
	A.1.4	Common agility and agile EA definition
	A.1.5	Management should communicate vision
	A.1.6	Management should define goals
	A.1.7	Management should define expected results
	A.1.8	Management should engage in communication planning
	A.1.9	Management should develop governance structures
A.2	EA Approach Stage	
	A.2.1	Elect project sponsor
	A.2.2	Elect Executive body
	A.2.3	EA should commit on behalf of themselves
	A.2.4	EA team should be accountable for results
	A.2.5	Focus on individuals and interactions over processes and tools
	A.2.6	Harness power of collective team rather than an individual
	A.2.7	Customer collaboration over contract negotiation
	A.2.8	No monopoly on innovation
	A.2.9	Define EA scope
	A.2.10	Reuse existing architecture artifacts
	A.2.11	Develop situational specific strategies, processes, practices
	A.2.12	Determine schedule in terms of iterations and releases
	A.2.13	Team that codes the system designs the system
	A.2.14	Create several architecture models in parallel
	A.2.15	Develop architecture in increments
	A.2.16	Build simplest architecture that can work

	A.2.17	When in doubt code it out
	A.2.18	Whoever builds it should test it
	A.2.19	Continuously build out system infrastructure
A.3	EA Extension and Management Stage	
	A.3.1	Respond to change over following a plan
A.4	EA Governance and Management Stage	
	A.4.1	Eliminate impediments for the EA team
	A.4.2	Management should be accountable for empowering the teams to deliver
A.5	EA Maturity and Measurement Stage	
	A.5.1	Evaluate EA maturity
	A.5.2	Measure value added by each iteration

(Own contribution)

10.2.3 Section B – Further Comments

This subsection discusses the second part of the survey which examines whether the organization has any methods for measuring the agility of EA and any further comments the respondent may make.

Section B enquires about whether the respondent‟s organization has any methods that they use to measure the agility of EA. This question was separated from the first part of the survey in order for the questions of Section A to focus solely on the framework factors for the development of agile EA.

In addition, Section B allows the respondent to comment on the questions; whether they were easy to understand and relevant to the topic. The respondent is also encouraged to make suggestions that would make the survey better, as well as suggest other factors that have been omitted that would encourage the development of agile enterprise architecture. The structure and layout of the section is outlined in Table 5 on the next page.

Table 5: Section B – Further Comments

B.1	Further Comments	
	B.1.1	Methods to measure the agility of EA
	B.1.2	Survey questions
	B.1.3	Survey suggestions
	B.1.4	Additional factors for the development of agile EA
	B.1.5	Additional comments

(Own contribution)

B.1 enquires about any further comments the respondent may have. Question B.1.1 attempts to uncover the methods, if any that the organization currently uses to measure the agility of EA. B.1.2 identifies how the respondent found the survey questions; whether they were easy to understand and relevant to the topic. B.1.3encourages the respondent to make suggestions that would make the survey better. B.1.4 prompts the respondent for any other factors that they believe may have been omitted that would encourage the development of agile enterprise architecture. B.1.5 enquires as to whether the respondent would like to make any additional comments pertaining to the framework, research or survey questions.

10.2.2 Section C – Demographics

This subsection discusses the third part of the survey which examines the respondent demographics.

Section C enquires about the respondent‟s job title and other relevant identifying characteristics, as well as general information about the organization itself. This part is chosen as the last set of questions to be asked in order to ensure that the respondent is occupied with the most important questions of the survey first, thus supplying enthusiastic opinions and answers. The structure and layout of the section is outlined in Table 6 below.

Table 6: Section C – Demographics

C.1	About you	
	C.1.1	What is your job title?
	C.1.2	What is your role in the organization?
	C.1.3	Who do you report to directly?
	C.1.4	How long have you been working for the organization?

C.2	About your organization	
	C.2.1	How would you best describe your organization and its function?
	C.2.2	How long have you been operational?

(Own contribution)

C.1 enquires about the respondent‟s job – related details. Questions C.1.1 – C.1.4 identify the respondent‟s job title; their role in the organization, who they report to and how long the respondent has been working in that position.

C.2 requests basic information about the organization itself. C.2.1 prompts the respondent to describe the organization and its function. C.2.2 enquires how long the organization has been operational.

10.3 Survey Delivery

The previous section described the design of the survey used within the empirical study. This section describes the means by which the survey was employed.

An interview at the system expert‟s office at the organization was conducted. Audio recording equipment and a copy of the questionnaire were used during the interview.

10.4 Conclusion

Changes were made to the survey instrument first developed in chapter 7. The changes were made as a result of the comments and experiences of the respondent and the researcher during the exploratory pilot study and the review thereafter and are deemed necessary in order to better enhance the data collection and analysis process that will follow.

The key factors for the successful development of agile EA were identified in preceding chapters and categorized into five subsections in the first part of the survey (A.1 – A.5). The survey also includes a section in which the respondent is allowed to make further comments (B.1.1- B.1.5). The final section of the survey (C1 – C.2) extracts demographical data from the respondent.

This chapter provided a detailed description of the design of the empirical study to be undertaken. The next chapter will present the analysis of the results of the empirical study

Chapter 11: Analysis of the Results of the Empirical Study

Chapter ten presented the design of the empirical study.

The discussion included the responses from the interviews with each organization with respect to the extent of understanding of enterprise architecture, agility and agile EA; the methods used to measure the agility of EA as well as the factors affecting the development of agile EA.

The aim of this chapter is to present the analysis of the results of the empirical study and to discuss the impact that they have on the research and the theoretical framework. The analysis of the results follows an inductive approach. A comparative factor analysis is made of the extent of understanding of enterprise architecture, agility and agile EA; the methods used to measure the agility of EA as well as each of the factors affecting the development of agile EA. A holistic analysis of the results is also provided.

It was concluded that that the use of semi-structured surveys proved useful in gathering information, opinions and experiences from respondents. Detailed results are included in Appendix B. The four sets of responses were sufficient to provide for interpretive case study analysis. The results provide insight into the challenge of understanding EA, agility and agile EA as well as the factors that affect the development of agile EA in the selected South African organizations.

A better understanding of EA and how agility fits within the context of EA (agile EA) is required. Before an organization can successfully develop agile EA it needs to achieve a common understanding of the terms “enterprise architecture”, “agility” and “agile enterprise architecture.” The definitions created within the literature review for EA, agility and agile EA were comprehensive enough to encompass those provided by the respondents in the empirical study and are therefore deemed sufficient. From the results gathered, it is clear that agility is of growing importance to organizations. Agility within the context of EA is a concept which EA teams are slowly taking into account and realising as an important factor that has to be measured. The organizations investigated in this study generally embrace the principals for the development of agile EA and believe that it is the clear future direction of their EA efforts. However, the principle that promotes developers to use code to evaluate alternatives was

deemed to not work in practice as it will result in budget and project time overruns. Concerns are evident in the lack of team commitment at the start of an EA project and that organizations find it difficult to have a clear cut deliverable and choose an approach to get that deliverable. In addition, the EA team should not only consist of architects, but also developers whose responsibility will be coding the models developed. In the end, while there may be an attempt to introduce elements of agility into the EA process to develop architecture in an agile manner, the success of these additions all add up to effective leadership and managing.

11.1 Introduction

This section provides an introduction to the analysis of the results of the empirical study.

Each of the previous chapters provided insight into the challenge of understanding enterprise architecture, agility and agile EA; the methods used to measure the agility of EA as well as the factors affecting the development of agile EA. This chapter will expand and generalize the available theory by analysing the results of the four interviews conducted at the selected South African organizations. Organization # 1 was revisited after the exploratory study and the results were modified and updated to reflect the requirements of the final survey designed in Chapter 10.

The data gathered from the interviews is used to develop conceptual categories or to illustrate, support or challenge theoretical assumptions held prior to data gathering.

This chapter provides an analysis of participant demographics, followed by a comparative factor analysis of the extent of understanding of enterprise architecture, agility and agile EA; the methods used to measure the agility of EA as well as each of the factors affecting the development of agile EA. Finally, a holistic analysis of the understanding of enterprise architecture, agility and agile EA; the methods used to measure the agility of EA as well as each of the factors affecting the development of agile EA is discussed.

11.2 Respondents

The previous section provided an introduction to the analysis of the empirical results. This section provides a description of the respondents involved in the empirical study.

The survey was completed through semi-structured interviews with each of the four organizations. Requests to conduct interviews were sent to each organization‟s presumed director of information, systems analysis, IT projects, etc. These directors either presented their interest in conducting the interviews themselves, or referred a more suitable respondent. In the interviews, respondents were required to answer open-ended questions about the extent of understanding of enterprise architecture and agility, the methods used to measure the agility of EA, as well as the factors affecting the development of agile EA at their organizations. While respondents were encouraged to answer questions based on the structure of the survey, they were free to elaborate or further explain their organization‟s current EA issues.

An interview at each of the four organizations was conducted to provide a total of four different sets of interpretive results. The identities of the respondents, the organizations and where they are situated in South Africa are protected under a confidentially agreement, but the results of the interviews are suitable for release.

11.3 Participant Demographics

The previous section provided a description of the respondents involved in the empirical study. This section provides an analysis of the participant demographics.

Section C of the survey (detailed in Appendix A) addresses the respondent‟s demographic details. These questions are designed to elicit information regarding the respondent‟s position at the organization and basic information about the organization itself. Table 7below illustrates the demographic data collected on the survey respondents.

Table 7: Respondent Demographics

	Organization # 1	**Organization # 2**	**Organization # 3**	**Organization # 4**
Job Title	Principal Consultant on Enterprise Strategy Consulting	Chief Architect	Senior Manager in Advisory Services	Enterprise Architect
Years in Position	3 years	2 years	3 years	20 years
Report To	Associate Director	Chief Information Officer	Associate Director	Communication Executive

Role	To engage with clients to understand business and IT requirements in order to define, architect and implement the appropriate IT Strategy	Responsible for the organization‟s enterprise architecture and ensures that the business strategy, along with any relevant trends is expressed in the enterprise's processes and systems	IT consulting, including enterprise architecture, IT transformation and program management	Strategy development, business modelling and design within all the architecture domains
Description of Organization	An American public multinational corporation that develops, manufactures, licenses and supports a wide range of products and services predominantly related to computing through its various product divisions	One of the big four banks in South Africa, providing a variety of banking and financial services	A global organization providing services in four main areas: Assurance, Tax, Transactions and Advisory	An American multinational technology and consulting corporation that manufactures and sells computer hardware and software and it offers infrastructure, hosting and consulting services in areas ranging from mainframe computers to nanotechnology
Years organization has been operational	36 years	10 years	22 years	100 years

(Own contribution)

No significant features regarding the demographic data of the participants were evident. Given that the sample chosen was purposive, the organizations were selected from organizations that have a formal enterprise architecture practice in one province in South Africa. Due to a confidentially agreement, the identity of these organizations and their province will remain anonymous.

Each of the respondents is responsible in some way for the administrative and core business systems used at each organization.

All of the respondents held senior positions and reported directly to executives of their respective organizations. This comes as no surprise, as the position of architect, or consultant on architecture is a senior one, which comes with it many years of experience and therefore a higher position in an organization‟s hierarchical structure.

The largest difference that can be noted amongst the respondents is that three out of the four respondents have been in their position for less than five years, whereas the respondent from organization # 4 has been in the same position for 20 years. As all the respondents interviewed are the most experienced in EA practices within their organizations, the likely cause of this is the respondents from organizations 1-3 having recently taken up new positions. The length of time spent in a role uncovered within this research is therefore no reflection of how experienced a respondent is in EA as he/she most likely had many years of experience in different roles/positions before being promoted to their current position of architect, or consultant on architecture. This may also imply the difference in hierarchical structures within organizations 1-3 which seem to lend themselves to an individual moving into more different roles than in organization # 4 based on their level of experience as they move up the hierarchical ladder.

An interesting feature is that while all of the respondents report to senior members within the organization, the respondent from organization # 2 is the only one that reports to a senior member of IT. This again may be attributed to the difference in nature and hierarchical structures of the organizations. Organization # 2 is a financial institution which exclusively provides financial services to the public, whereas the other 3 organizations provide a variety of services ranging from audit, tax, transactions, consulting and advisory as well as other IT related services.

Three out of the four organizations are global, with two being headquartered in the USA. The most significant difference in the data gathered about the organizations is that organization # 4 has been operational for 100 years as opposed to organizations 1-3 having been operational for less than 40 years; namely 36 years, 10 years and 22 years respectively.

While slight differences exist in the gathered respondent and organization demographic data, these variances bear no significance to the findings of the study as the only requirement for an organization to participate in the research study was that they currently have a formal EA practice.

11.3 Comparative Factor Analysis

The previous section provided an analysis of the respondent demographic data and uses an inductive approach. This section presents a comparative factor analysis of the results of the empirical study.

Due to the imperative nature of this study, in-depth responses to each of the interview questions are detailed in Appendix B. This section summarizes these results and presents them by case.

As a sufficient starting point in the exploration of the development of agile EA, the extent of understanding of enterprise architecture, agility and agile EA and the methods used to measure the agility of EA are first investigated first.

11.3.1 The extent of understanding of enterprise architecture

This subsection of the comparative analysis of the results analyses how the respondents of the four organizations view EA.

Organization # 1

The respondent views EA as the design of the entire make up of an organization.

Organization # 2

This respondent views EA as a framework that defines the structure and operations of an organization and is used to manage and align an organization‟s IT assets to its business goals.

Organization # 3

EA is the high-level representation of how business, IT and the people and processes that support them within an organization relate to one another.

Organization # 4

The respondent views EA as a whole set of artifacts that will be delivered against certain business challenges and involves designing solutions that include all areas of the business, risk and governance, etc.

Summary Analysis of the Extent of Understanding of enterprise architecture

It is clear that while there are slight variations in the definitions across all the four organizations; those involved in the EA effort tend to view EA as a broad description of all of the key elements and relationships that make up an organization. From the definitions provided, it can be seen that the respondents assume that implementation is not within the scope of EA. In fact, that is where it should play a central role in order to ensure successful implementation of the business strategy that represents corporate goals. The recommendation by Gartner research is to view EA as a course of action (verb) that results in deliverables rather than the deliverables (noun) produced as a result of the process. This would explain why the respondents from the organizations refer to the deliverables produced as, "the enterprise architecture." It has been found that when people focus on EA as a noun, they focus on the outputs and are more concerned about producing a predefined set of deliverables rather than about meeting the strategic imperatives of the enterprise (Gartner, 2008a: 2).

11.3.2 The extent of understanding of agility and agile EA

This subsection of the comparative analysis of the results analyses how the respondents of the four organizations view agility and agile EA.

Organization # 1

This respondent views agility as a term meaning something responsive or swift and within the context of EA; an EA that was adaptable or could be changed easily if a change was required.

Organization # 2

This respondent views agility as the speed at which something can change. The faster something can change, the more agile it is. Therefore an agile EA is one that can be changed quickly and easily.

Organization # 3

This respondent views agility as the ability to handle changing circumstance and that an agile architecture is one that lends itself to being changed easily, should a need arise.

Organization # 4

This respondent refers to agility as something that is easily changed. However the concept of agility was one that had not been considered within the context of EA.

Summary Analysis of the Extent of Understanding of agility and agile EA

All of the respondents were able to provide a definition for the term "agility." Their definitions differ slightly, however they are all in line with the view of agility as the ability to react to upcoming changes effectively and efficiently.

Perhaps the most noticeable difference was that the respondent from organization # 4 indicated that agility was not a concept that had been considered within the context of EA. This is most likely due to the belief that each architecture effort, if done as a part of a fresh full life-cycle project, is meant to deliver agility, therefore architecture is done *for* agility rather than *with* agility. However Lin, *et al*. (2006) believe that the lack of a systematic approach to agility does not allow companies to develop the necessary proficiency in change, a prerequisite for agility.

The collective view of agile EA from respondents from the other three organizations is an architecture which can rapidly and cost effectively adapt to, or exploit changes in customer/mission demands, or respond to uncertainties and unexpected conditions in the environment. This view of agility involves introducing agility in the architecture deliverables created as a result of the EA project and is known as product agility. This is achieved by embedding agility characteristics in the end product, which means that the end product or system is architected for extensibility and substitutability (Madni, 2008: 50). The respondents‟ view of agile EA is as a result of their definition of EA focusing on EA as a noun, in that it relates that to the artifacts produced by the process of architecting and not that EA relates not only to the artefacts produced but also to the process of producing them; therefore the agility/adaptability which they refer to relates to how easily the architecture created by the architects can be changed.

It was however established in chapter two, that while some authors view EA as a noun (product), this view is outdated and lends those involved in the EA practice to at times use approaches that are incompatible with today‟s business environment (Edwards, 2008: 1). The view taken in this thesis is to view EA as a verb (a process) which results in artifacts (nouns) created and therefore agile EA within the context of this research focuses on process agility, rather than product agility.

11.3.3 The methods used to measure the agility of EA

This subsection of the comparative analysis analyses the methods, if any, that the four organizations use to measure the agility of EA.

Organization # 1

The organization uses methods made available in the established EA frameworks (Zachman, TOGAF, GARTNER, etc.) to measure the agility of the created architecture.

Organization # 2

The organization makes use of the Enterprise Architecture Executive Council (EAEC) scorecard to measure the architecture.

Organization # 3

The organization use the methods provided in the framework(s) chosen for use by the EA team.

Organization # 4

The organization does not have any methods for measuring the agility of EA**.**

Summary Analysis of the methods used to measure the agility of EA

Three out of the four organizations reported to have methods to measure the agility of EA. However, these methods focus on measuring the agility of the end product created by the process of EA. This is not surprising, as the definitions provided for EA by the respondents of the organizations focus on EA as a noun (the deliverables); thus, within these organizations, a measurement of any aspect of EA will result in a measurement of the created architecture. Therefore, like organization # 4, the other three do not possess any methods for measuring the agility of the process of creating the architecture which is the focus of this research.

11.3.4 The factors affecting the successful implementation of agile EA

This subsection provides a comparative factor analysis of the factors affecting the development of agile EA:

- **Needs and cost benefit analysis**

Organization # 1

Prior to embarking on an EA project, the organization creates a needs and cost-benefit analysis and statement of work.

Organization # 2

A needs and cost benefit-analysis is conducted, based on best practices that have been developed within the organization to assess costs and the benefit a project would have on the organization.

Organization # 3

The respondent indicated that this is an important step in justifying the change that the project will bring.

Organization #4

A needs and benefit analysis is conducted prior to embarking on the EA project. The needs analysis for the EA project is tied back to functional business results.

Summary Analysis of Needs and cost benefit analysis

All of the organizations recognize the importance of conducting a needs and cost benefit analysis as not doing so would mean that they may invest a lot of time and money in solving a problem that is not worthy of its effort.

- **Top management buy-in**

Organization # 1

Top management review the project proposal and thereafter conduct a value realization; signing off on project worthy proposals.

Organization # 2

Top management are required to sign off on potential projects as EA is a key component of the organization.

Organization # 3

Management attend project analysis meetings in which they determine the actions which need to be taken on a project and the intended benefit(s) and sign off on projects that are deemed beneficial to the organization.

Organization # 4

Top management are required to sign off on project worthy proposals. The projects to be signed off on by management will usually depend on the size of the project and whether the project budget requirements fall within the allocated IT project budget.

Summary Analysis of Top management buy-in

Approval from top management for an EA project is a requirement within all of the organizations. This is usually the case where an EA project will be undertaken on a group wide basis, hence the need for endorsement from the individuals that control the organization at this scale.

However what is interesting to note is that within organization # 4, IT is allocated a project budget which they are entitled to use on what they deem project worthy activities. In a situation where the budget required for the EA project can be allocated from the IT project budget; top IT management would be required to sign off on the project as opposed to all the executives of the organization. This shows the growing importance of IT within organization # 4 and the ability for IT to be able to make decisions to a certain extent without their every move having to be sanctioned by other non-IT executives. Perhaps this may be attributed the fact that the main function of organization # 4 is to provide IT related services, hence the acknowledgement of the importance of IT by the other non-IT executives who control the organization. In addition, organization # 4 has been operational for a significantly longer period than the other three organizations, has been a global leader in IT innovation and has had time to mature as an organization.

- **Common EA definition**

Organization # 1

It was clear that the EA team within organization # 1 have a common understanding of what enterprise architecture is, which is achieved through the definition workshop prior to any work on the EA project being done.

Organization # 2

Achieving a common understanding of EA within the organization is extremely problematic. The definition of EA within the organization varies from person to person and there is a constant challenge to educate and clarify its meaning, value and application.

Organization # 3

Not every individual understands the term, "enterprise architecture" and what it means within the organization. This creates a great deal of confusion initially and causes further problems when attempting to meet project objectives.

Organization # 4

Not everyone within the organization has a common understanding of what EA is, or why it is needed.

Summary Analysis of Common EA definition

Only organization # 1 has managed to achieve a common understanding of EA for those involved with the EA effort. This understanding is achieved through a definition workshop which runs prior to any work being done on the project. The rest of the organizations experience the confusion that results out of a lack of a common view of EA for the team members.

- **Common Agility and Agile EA definition**

Organization # 1

The respondent of organization # 1 indicated that the meaning, as well as a shared understanding of agility and agile EA is achieved through the definition workshop that runs prior to any work being done on the EA project.

Organization # 2

The respondent indicated that there is no single view of agility; what it means, how it is interpreted and how it fits within the context of EA. The confusion is further compounded by the existence of other practices e.g. the design methodology called Agile Development.

Organization # 3

The respondent indicated that the meaning of agility and agile EA within the organization was very subjective.

Organization # 4

The organization does not have a common understanding of agility and the concept of agility is one that the organization had not considered within the context of EA.

Summary Analysis of Common Agility and Agile EA definition

The definition workshop which runs prior to any work being done on the EA project allows organization # 1 to achieve a common understanding of agility and how it fits within the context of EA. The view of agility and agile EA within the other three organizations is very subjective. The confusion within organization # 2 is further compounded by the existence of other practices e.g. the design methodology called Agile Development. As the concept of agility is one that organization # 4 has not considered within the context of EA, it is inevitable that those involved in the EA effort do not have a common understanding of agile EA. The three organizations inevitably experience the turmoil that a lack of a shared understanding of agility and agile EA brings when attempting to develop agile enterprise architecture within their organizations.

- **Management should communicate vision**

Organization # 1

Management communicates their vision for the project and informs project members of the project objectives.

Organization # 2

Management is effective in communicating the vision for the project allowing the EA team to understand why they are embarking on the selected project.

Organization # 3

Management communicates the vision for the selected EA project in order to provide strategic direction for the EA team.

Organization # 4

Management provides a clear vision in order to enable the EA team to be able to have a clear understanding of what the project is intended to achieve.

Summary Analysis of Management should communicate vision

The organizations ensure that the EA project does not begin without a complete understanding of the problem that the project is intended to solve. In this way, the project vision provides the picturing of the project's deliverable as the solution to the stated need or problem.

- **Management should define goals**

Organization # 1

Management successfully translates their vision for the project into a set of goals that will allow that vision to be achieved.

Organization # 2

Management is ineffective in communicating the goals of the selected project. At the highest level, a vision for the project is communicated, although the translation of that vision into the goals of the selected project(s) is not communicated effectively.

Organization # 3

Management provides a strategic intent (vision) for the selected project and communicates how they hope to achieve that strategic intent (the goals) through the selected project(s).

Organization #4

Management communicates the goals of the selected project, which allows the EA team to understand the work that needs to be done on the project.

Summary Analysis of Management should define goals

Three of the four organizations are effective in communicating the goals of the selected EA project. Within organization # 2 at the highest level, a vision for the project is communicated, although the translation of that vision into the goals of the selected project(s) is not communicated effectively. This leads to a great deal of sensitivity in the organization, in terms of the EA team being uncomfortable with the project and not knowing what to expect.

The confusion within organization # 2 is due to a lack of efficient leadership (developing a vision and translating that vision into goals and objectives) because a project vision requires a mission that translates to goals which give the project general direction and purpose. The project goals, in turn, lead to a set of project objectives. In this way, it is rather like a work breakdown structure where the important detail is at the lowest level (Wideman, 2000).

The lack of clarity on project goals can perhaps be attributed to the fact that organization # 4 is a financial institution with a traditionally rigid management structure which views IT as a supporting role within the organization. The non-IT management of the organization who are at its forefront create a vision for the organization and its business related activities without much consultation with IT. In this way, IT is constantly "playing catch-up" to business in an effort to ensure that it is aligned with business. These business visions in some cases may require a change in the current IT architecture. The business vision is thrust upon the EA function without much communication as to what it looks like from an IT perspective and the EA team is required to adapt and ensure that the EA project achieves the business related objectives set forth by the vision. A better approach would be to systematically create an IT vision and goals during the creation of the business vision and goals.

- **Management should define expected results**

Organization # 1

The expected results of the EA project are defined throughout the project from the preliminary stages until its completion.

Organization # 2

The respondent indicated that management communicates the expected results of the project. The expected results vary from project to project.

Organization # 3

As IT is seen as an enabler of business, all IT projects within the organization should provide business value or results. The expected results of the project are articulated to the EA team by management.

Organization #4

The respondent indicated that although it is an important step for an organization‟s EA to mature, management is not always clear in defining the expected results of the EA project(s).

Summary Analysis of Management should define expected results

Management is effective in communicating the expected results of the project to the EA team within three out of the four organizations. The expected results vary, from project to project. For example in some instances, the expected results of the project may be from a financial perspective or an operating perspective, e.g. cost efficiencies, optimization, streamlining of the various operations, eliminating duplication.

Within organization # 4 however, there appears to be a lack of efficient leadership, as it is at times unclear what value a project is intended to add to the business. This can be detrimental to project success. As mentioned previously by Wideman (2000), a project vision requires a mission that leads to goals which give the project general direction and purpose. The project goals, in turn, lead to a set of project objectives or expected results. It is important that the organization specifically define their goals as well as how they intend to implement EA and the expected results (Bernard, 2004; Dziewulski, *et al*., 2003: 7; Gartner, 2008b: 4; Hansen, 2006: 23).

- **Management should engage in communication planning**

Organization # 1

Management plans the way in which the EA team will communicate and ensures that the project team attends weekly status meetings.

Organization # 2

There are no formal communication structures for the EA project established by management. Various avenues and pockets of communication are present within the organization with no common understanding throughout the various areas.

Organization # 3

Management is effective in setting up communication structures and is flexible in terms of what will work best for the team.

Organization #4

The program manager of the EA project is responsible for drafting the communication policies and standards for the EA project and communicating these to the EA team.

Summary Analysis of Management should engage in communication planning

The development of communication policies and structures that project team members adhere to is vital for project success. This is true in three out of the four organizations in which methods of communication are established and understood by all the members of the EA team. In this way, each member of the EA project team is aware of the means and channels in which they are able to communicate with other team members as well as any meetings etc., which they are required to attend.

However, within organization # 2, there are no formal communication structures for the EA project established by management. Various avenues and pockets of communication are present within the organization with no common understanding throughout the various areas. In some instances this leads to the EA team communicating with the wrong audience or sending out the wrong message.

Bringing a project in on-time and within budget requires the input and cooperation of every member of the team and it is essential that every team member maintain clear lines of communication throughout all stages of the project.

- **Management should develop governance structures**

Organization # 1

Management (sponsor/executive and architecture board) defines the way the EA project is to be run in the definition workshop which runs before any work for the EA project is done.

Organization # 2

Governance within the organization only exists on a high level, in terms of how the organization operates. However, from an EA perspective there is a lack of governance in terms of the projects and how they will affect the organization and in some cases this has an adverse effect on the project‟s success.

Organization # 3

Within the organization management establishes governance structures that guide and protect the interests of the organization in terms of the money spent on the specific initiatives and other important aspects which relate to project success.

Organization #4

Within the organization there is a particular methodology used and that methodology includes governance policies which have to be adhered to.

Summary Analysis of Management should develop governance structures

A robust governance structure is required for projects to ensure that they are delivered on time and under budget. While management are effective in developing governance structures that will govern the EA project, this is a challenge within organization # 2. Governance within the organization only exists on a high level, in terms of how the organization operates. From an EA perspective however, there is a lack of governance, in terms of the projects and how they will affect the organization and this has an adverse effect on the project‟s success.

The governance problems within organization # 2 can be attributed to a problem in "management" which has to do with the details of initiating, planning, executing, controlling (governing) and closing (Wideman, 2000). It is in line with the research conducted by Gartner (2008b:8), who found that in many organizations, governance initiatives and decisions are defined without a link or even knowledge of EA. Meanwhile, enterprise architects may be separately focused on EA processes and practices, and on engaging business leaders to ensure

that the evolving EA reflects and supports the business strategies and goals. The result is that diverse roles and responsibilities are often misunderstood, resulting in significant overlaps, poor investment decisions, wasted resources and miscommunications.

- **Elect project sponsor**

Organization # 1

A project sponsor or executive is elected for the project prior to it being approved.

Organization # 2

A project sponsor is usually the one who will fund the project and is elected prior to the project‟s initiation.

Organization # 3

A project sponsor whose role is the ownership of the project on the organization‟s behalf is elected for the project prior to it being approved.

Organization #4

It is not possible to have a project without a project sponsor within the organization.

Summary Analysis of Elect project sponsor

The importance of a project sponsor is stressed across all the four organizations. As a project sponsor is the one who will be responsible for funding the project, the EA project cannot exist without a project sponsor.

- **Elect executive body**

Organization # 1

An architecture review board comprising of senior members of the organization (Chief Architects, domain architects, CIO, etc.) is elected to ensure that top management is involved in high-level decisions of the EA project.

Organization # 2

The respondent indicated that within the organization, there is a COO (Chief Operating Officer) forum consisting of all the COOs of the various business units as well as a transformation executive committee which has been established to lead and make decisions regarding any projects that lead to a transformation of the organization and the head of enterprise architecture sits on both these committees.

Organization # 3

Within the organization, an executive body comprised of the Chief Architect and executives within the organization would be elected to lead the EA project as these individuals make decisions on a group wide basis. At a lower level, there would also be a committee of architects; solution, information, business architects etc.; who give guidance in terms of the projects and architecture components surrounding the project.

Organization #4

Every project in the organization has a steering committee consisting of executive managers whose responsibility is to drive project success.

Summary Analysis of Elect executive body

All four organizations elect an executive body/ steering committee to ensure that top management is involved in decisions relating to the EA project and whose responsibility it is to drive project success. The executive body is usually comprised of the senior IT and non-IT executives within the organizations. The chief architect is also part of the executive body and amongst other duties, ensures that the concerns of the EA team are put forward to management.

What is interesting to note is that organization # 3 also has a committee at a lower level comprised of architects; solution, information, business architects etc.; who give further guidance in terms of the projects and architecture components surrounding the project.

- **EA teams should commit on behalf of themselves**

Organization # 1

The respondent expressed that the EA team has a high level of commitment at the start of the project.

Organization # 2

The respondent indicated that initially, there is little commitment from team members on a new project as there is a degree of uncertainty about what the project is intended to achieve.

Organization # 3

The nature of enterprise architects is that they are individuals who want change. They get stuck when things remain the same. Therefore, if a project brings change within the organization, the EA team is motivated to participate.

Organization #4

Not all of the EA team members are motivated and committed to a project at its inception. The level of commitment to and motivation on the project depends on the type of the project and how it meets their own requirements.

Summary Analysis of EA teams should commit on behalf of themselves

Two out of the four organizations express a high level of commitment from their EA teams at the start of the project. However, within organization # 2 there is little commitment from team members on a new project as there is a degree of uncertainty about what the project is intended to achieve. The same low level of commitment to a new project is present within organization # 4, as not all team members are committed to the new project. The level of commitment and motivation to the project within organization # 4 largely depends on the type of the project and how it meets the team members‟ own requirements.

Billows (2010), suggests that there is one factor, above all, that influences how well a project works as a team: commitment to shared objectives.

As mentioned previously by Wideman (2000), a vision for a project translates into goals and objectives/expected results (the details). Within organization # 2 there is a lack of clarity on the goals of a selected project. Nevertheless, the respondent of organization # 2 indicated that

management defines the expected results of the EA project. However, Wideman (2000) suggests that if an attempt is made to only pass on the details to the people in a project team, these objectives may well not make sense because they don't ultimately hang together under the project's vision and goals.

Within organization # 4, management is ineffective in defining the expected results of a new project. Billows (2010), in his research on team commitment proclaims that one of the four elements required for a person to commit to finishing an assignment is that the performance expectation is an unambiguous result that can be measured, so that the individual knows what is expected before they start their work.

It is no wonder why these two organization‟s respondents report a low level of commitment at the start of a project.

- **EA team should be accountable for results**

Organization # 1

The EA team is held accountable for meeting the deadlines, goals and results of the overall project.

Organization # 2

Within the organization, the Chief Architect is the person held accountable for the success of the project.

Organization # 3

Every member within the EA team has a main role and a supporting to play on the EA project within the organization. Their roles are interconnected and as a result, a failure of the EA project is a failure of the team as a whole.

Organization #4

The EA team as a whole is held directly accountable for the success of the project.

Summary Analysis of EA team should be accountable for results

Three out of the four organizations share the view suggested by Ambler (2010); that everyone on the team should be responsible for ensuring that the EA project delivers the expected

results. This is not true in organization # 2, in which the Chief Architect is the person held accountable for the success of the project. One explanation is that organization # 2 being a financial institution has a traditional, very rigid and hierarchical management structure which requires one person at each managerial level; with subordinates and reporting to a manager at the next or higher level. Financial institutions are notoriously difficult to change because of the high levels of bureaucracy; as a result it becomes difficult to accommodate recently developed agile methods which advocate team ownership of the created architecture as opposed to ownership by one person. Architecture however is far too important to leave in the hands of a single person no matter how bright they are; therefore architecture should be a team effort (Ambler, 2010).

- **Focus on individuals and interactions over processes and tools**

Organization # 1

The organization places a great emphasis on face to face team collaboration as this fosters a better exchange of ideas as opposed to collaboration over software tools.

Organization # 2

The EA team meets face to face where possible and also make use of software collaboration tools. However, there is a tendency to not share intellectual capital due to the direct link to job security.

Organization # 3

The EA team prefer face to face collaboration as this builds a bond between the team members, creating a better team dynamic. The frequency and other aspects of these face to face collaborations is controlled by the project management methodology which the EA team is using. The knowledge manager of the EA team runs a knowledge sharing and collaboration process throughout the life of the project.

Organization #4

Where possible, the EA team favours face to face collaboration. However, in some instances this is not possible as the organization also makes use of virtual teams where members of the team may be situated far away from others.

Summary Analysis of Focus on individuals and interactions over processes and tools

Face to face communication is a strongly advocated within all four organizations and where not possible, the members of the EA teams use collaboration software tools. Within organization # 3, the frequency and other aspects of these face to face collaborations is controlled by the project management methodology which the EA team is using and the knowledge manager of the EA team runs a knowledge sharing and collaboration process throughout the life of the project.

In this way, team members obtain a significantly deeper sense of who their team mates are, how they operate, their personalities as well as each team member‟s responsibilities. All of which will make them productive in working together in the future.

A curious feature about organization # 2 is the tendency to not share intellectual capital due to the direct link to job security. Transparency within project teams allows greater agility for managing project changes, and allows for better strategic planning, management, efficiencies and costs savings. The lack of transparency and visibility prevents team members and stakeholders from making factual, metrics-based project decisions and therefore negatively affects project success (Oracle, 2009).

- **Harness power of collective team rather than an individual**

Organization # 1

Within the EA team, the Chief Architect is the main person responsible for the architecture and supervises and delegates responsibilities to the other members of the team.

Organization # 2

The respondent pointed out that while the Chief Architect is held accountable for the success of the EA project, the EA team as a whole was responsible for the architecture.

Organization # 3

Each member of the team has domain specific responsibilities which together form the complete architecture.

Organization #4

The EA team as a whole is responsible for the architecture. The overall responsibility is with the Chief Architect. He/she in turn holds the different domain architects responsible for their different areas of the architecture.

Summary Analysis of Harness power of collective team rather than an individual

The EA team as a whole within all of the organizations is responsible for the created architecture. This is accomplished by each domain specific architect focusing and being responsible for their domain specific architecture, ensuring that it is in line with the project requirements and works together well with the other parts of the architecture. The Chief Architect supervises the creation of the architecture and ensures that all of the different parts fit together well to achieve the project objectives. This is in contrast to historical practices in which agile architecture was a primary function of the system architect. But the most common agile methods don‟t define or even support such a role. Since agile focuses on harnessing the power of the collective team, rather than any one individual, the system architect no longer dictates technical direction (Leffingwell, *et al.,* 2008: 2).

- **Customer collaboration over contract negotiation**

Organization # 1

The EA team communicates and collaborates with the project sponsor/executive responsible for the project throughout its lifecycle.

Organization # 2

The EA team constantly engages with the client(s)/stakeholder(s) throughout the life of the project as this is an integral part of the EA process and ensures service and value is delivered.

Organization # 3

Engaging with the stakeholder(s) throughout the project lifecycle ensures the EA team that they are meeting the demands of the customer(s) and that their efforts are in line with the project requirements.

Organization #4

Communication with the customer(s) throughout the project life is important for project success in order to ensure that the requirements of the project are being met, value of the project to the customer is shown and any changing requirements of the project are discussed.

Summary Analysis of Customer collaboration over contract negotiation

The four organizations collaborate with the customer(s) throughout the life of a project. They understand that communication with the customer(s) throughout the project life is important for project success in order to ensure that the requirements of the project are being met, value of the project to the customer is shown and any changing requirements of the project are discussed.

- **No monopoly on innovation**

Organization # 1

Following a project iteration a review is done to ensure that project is still in line with its requirements. Thereafter, ways to improve the project are discussed and if any are present, they are added to the project requirements.

Organization # 2

It is not part of the organization culture to have strong innovation processes/practices.

Organization # 3

During the life of a project, the requirements of the project may change, or more added. The need for changes is identified not only by the EA team but also by the stakeholder(s). Any changes to the requirements have to be justified by the EA team and sanctioned by the stakeholder(s) as these changes will inevitably affect the project timeline and budget.

Organization #4

No processes are included to introduce innovation on a project. The EA team fear that the inclusion of such processes would lead to scope creep.

Summary Analysis of No monopoly on innovation

Organization # 1 and organization # 3 have review processes in place in which ways to improve the project are discussed and if any are present, they are added to the project requirements. These additions or changes are subject to approval by the project customer(s) as they inevitably impact on the project timeline and budget.

The fear of the negative impact on the project timeline and budget is the reason why organization # 4 has no processes included to introduce innovation on a project. The EA team fear that the inclusion of such processes would lead to scope creep. This can be avoided if the innovative processes are added as part of the original project timeline. Having these processes in place will allow the EA team to sense environmental change and react effectively to that change.

The rigid structure of organization # 2 is such that it is not part of the organization culture to have strong innovation processes/practices, as once an EA project is underway, the organization does not look into changing the project requirements. Ambler (2009) states that such a practice could lead to outdated architecture because even in the unlikely event that a development team managed to create perfect artifacts, they"d be out of date the day after they were published because something within the business, or technical environment would change.

- **Define EA scope**

Organization # 1

The EA team is presented with a clear scope of the project in which to work from the initial definition workshop and the statement of work to be done for the project.

Organization # 2

Initially, the EA team is not provided with a clear scope in which to work; the scope of the project only becomes clearer once the project is underway. There would be a strategic intent for the organization, however what that strategic intent looks like from an architecture perspective is still very unclear.

Organization # 3

The high-level strategic view of the project is broken down into more detail in order to understand the impact of the project and provide the scope of the work to be done.

Organization #4

The EA team creates a clear scope of the work to be done on a project.

Summary Analysis of Define EA scope

The EA teams of organizations 1, 3 and 4 enjoy the benefits of a clearly defined scope in which to work. This allows the members of the team to understand how much and what work needs to be done on a project and gives clarity and direction.

This is not true within organization # 2 in which the EA team is not provided with a clear scope in which to work; the scope of the project only becomes clearer once the project is underway.

In his research on the importance of a scope for a project; Morrison (2006) highlights that a project‟s scope and vision define the broad parameters of the project and provide the foundation for all subsequent steps in the project or programme cycle. A clear scope sets the rough boundaries for what the project will attempt to do. As discussed previously by Wideman (2000), what a project attempts to do can be viewed broadly as the goals of the project and these give the project general direction and purpose. Once an EA team understand what a project intends to achieve (goals), they can formalize the amount of work to be done on a project and thereby obtain a scope of work to be done. As mentioned above, management within organization # 2 is ineffective in defining the goals of the selected project and would explain why at the start of a project, the EA team does not have a clear scope in which to work and this requires the team to initially create a scope for the project which in itself is an iterative process over a number of months. Within organization # 2, there would be a strategic intent (vision) for the organization; however what that strategic intent looks like from an architecture perspective is still very unclear.

- **Reuse existing architecture artifacts**

Organization # 1

Where possible, the EA team reuses existing artifacts. In some cases, the artifacts have become obsolete and part of the project work will be to replace them entirely.

Organization # 2

The respondent indicated that the EA team reuses existing artifacts if possible as this reduces the amount of work to be done on the project.

Organization # 3

Where possible, the EA team will avoid "re-inventing the wheel." Reuse of existing artifacts is a large part of the EA team development framework and brings with it many benefits.

Organization #4

The EA team will reuse existing artifacts where possible. However, the existence of artifacts which the EA team can reuse will also depend on how mature the organization is and the objective(s) of the selected project(s).

Summary Analysis of Reuse existing architecture artifacts

Reusing existing artifacts is a practice conducted in all of the four organizations to speed up the development process. The EA teams see no point in recreating artifacts that are already in place within the architecture. The existence of artifacts which the EA team can reuse will also depend on how mature the organization is and the objective(s) of the selected project(s). At times, these artifacts may need to be modified slightly in order for them to fit in the design of the new architecture. Even these small modifications are less costly than re-development. However, in some cases the artifacts have become obsolete and part of the project work will be to replace them entirely.

- **Develop situational specific strategies, processes, practices**

Organization # 1

The EA team develops a combination of event based strategies that will depend on the occurrence of certain events as well as set strategies.

Organization # 2

As projects have differing requirements and operate in different environments, the EA team develops strategies and practices that will provide the best solution to the problem.

Organization # 3

The EA team develop a project strategy and practices that will best suit the objectives of the project.

Organization #4

The EA team develops situationally specific strategies and practices using a strict method which enables the definition of the strategies and practices and what products need to be worked through.

Summary Analysis of Develop situational specific strategies, processes, practices

The EA teams within the four organizations understand that during the development of EA, each situation is different as projects have differing requirements and operate in different environments. The EA teams use strict methods which help the definition of the strategies and practices for each situation and what products need to be worked through.

- **Determine schedule in terms of iterations and releases**

Organization # 1

The respondent indicated that the project lifecycle is organized in terms of iterations and releases, with project reviews conducted after each iteration.

Organization # 2

The EA team see the EA project as an iterative process and use each iteration to review the work done on the project to ensure it is in line with the project requirements. This allows any deviations from the objectives of the project to be caught early, ensuring that the project will achieve its intended result(s).

Organization # 3

The respondent indicated that the project is organized in terms of iterations and releases in order to show the stakeholder(s) a return on their investment as soon as possible. There are different time frames for different value concepts; some are realized sooner than others.

Organization #4

The Project is broken up into phases and sub-phases which are linked to iterations.

Summary Analysis of Determine schedule in terms of iterations and releases

The EA teams of the four organizations organize the work to be done on a project into iterations and releases. This allows any deviations from the objectives of the project to be caught early, ensuring that the project will achieve its intended result(s) and for the stakeholder(s) to be shown a return on their investment as soon as possible. There are different time frames for different value concepts; some are realized sooner than others.

- **Team that codes the system designs the system**

Organization # 1

The EA team that designs the system does not code the system.

Organization # 2

The team that designed the system is not responsible for coding it. The EA team that designed the system merely provides coding guidance models.

Organization # 3

The EA team that designed the system gives the completed designs to the development team to be coded. The EA team provides clarity where it is needed and supervises the development of the system, ensuring that the developed system is in line with the designs, but does not code the system themselves.

Organization #4

The organization makes use of a design team on projects and a separate coding team.

Summary Analysis of Team that codes the system designs the system

The EA teams within the organizations consist of architects whose sole responsibility is to derive functional specifications (use case diagrams, etc.). In organization # 2, the EA team provides coding guidance models for the development team. The four organizations‟ EA teams provide clarity where it is needed and supervise the development of the system; ensuring that the developed system is in line with the designs, but do not code the system themselves. In some cases within organization # 1, the development function is outsourced to an external organization.

- **Create several architecture models in parallel**

Organization # 1

Where possible, the EA team creates several architecture models in parallel. This depends on the situation (Technical model can be created in parallel with a reference model for applications). At a logical level this is possible, however at a physical level it is not as the two are interdependent.

Organization # 2

The time frames for the completion of a project require parallel development. The organization does not have the luxury of following a sequential approach.

Organization # 3

Creating several models in parallel is a necessary practice of the EA team as creating the models one after the other will result in greater costs and a longer time frame for the completion of the project.

Organization #4

The EA team creates architecture models in parallel where there are different streams of development that can be followed in parallel. The EA team also cross references the models to ensure that they make sense in relation to the overall project.

Summary Analysis of Create several architecture models in parallel

Where possible, creating several architecture models in parallel is necessary for all four organizations to deliver the EA project on time and under budget. In some cases however, this is not possible as the architecture models may be interdependent. During concurrent development of the models it becomes necessary to cross reference the models to ensure that they make sense in relation to the overall project.

- **Develop architecture in increments**

Organization # 1

The work to be done on a project is broken down into manageable tasks and a schedule drawn up to indicate the work to be completed in a given period. This allows the architecture to be developed in increments.

Organization # 2

The methodology the EA team uses follows an incremental approach to developing the architecture.

Organization # 3

The EA project is broken into manageable parts and each part of the project into tasks for the members of the EA team. This allows the architecture to be developed incrementally and the EA team to show the value added by the project after each iteration.

Organization #4

The EA project has a macro design which is then translated to a micro design which will represent the increments for the whole project.

Summary Analysis of Develop architecture in increments

The organization of the work to be done on a project into iterations and releases allows the EA teams of the four organizations to follow an incremental approach to developing the architecture.

- **Build simplest architecture that can work**

Organization # 1

Additional features are added to the architecture in addition to the requirements of the EA project to cater for events that the EA team believe can occur in a hope to make the architecture robust and handle future changes.

Organization # 2

The respondent indicated that the EA team find it difficult to have a clear cut deliverable and choose an approach to get that deliverable; therefore, where they believe it is appropriate, additional features are added to the architecture in an attempt to holistically improve it.

Organization # 3

Architects as thought leaders will suggest and motivate the need for additional features. This is in an attempt to improve the architecture after the various iterations, by bringing in features that will aid future development and assist with acceleration.

Organization #4

Technology is kept to the minimum needed to solve the problem at hand as it is important to not over engineer a solution. The dangers of over engineering a solution are under delivering the solution at a cost that is unacceptable and the Return on Investment (ROI) becomes more difficult to reach.

Summary Analysis of Build simplest architecture that can work

Three out of the four organizations find it difficult to have a clear cut deliverable and choose an approach to get that deliverable; therefore, where they believe it is appropriate, additional features are added to the architecture in an attempt to holistically improve it. This is in an attempt to improve the architecture after the various iterations, by bringing in features that will make the architecture robust, assist future development and assist with acceleration. This is especially not surprising in organization # 2, as once an EA project is underway the organization does not look into changing the project requirements. The EA team therefore spend a great deal of time trying to cater for every possible event that could occur.

This is in contrast with literature on agile development practices which advocate keeping things as simple as possible. Technology should be cut to the minimum needed to solve the

problem at hand. The organization is not aided by creating artificial and unneeded layers of complexity (Leffingwell, *et al.,* 2008: 4).

This is true with organization # 4, in which the EA team believe that it is important to not over engineer a solution as additions come at a cost that is unacceptable and the Return on Investment (ROI) becomes more difficult to reach. The EA team within organization # 4 ensure that the simplest architecture that will solve the problem is built.

- **When in doubt code it out**

Organization # 1

The respondent explained that developers do not use code to evaluate the architecture designs. This will most likely result in project overruns.

Organization # 2

The respondent indicated that the developers do not use code to evaluate different possible solutions as this would in most cases increase the time spent on a project or iteration, meaning that the project will run over time and over budget.

Organization # 3

The developers do not use code to evaluate their designs. Instead, they are required to make decisions on project alternatives based on the requirements of the project, their previous experience, research and best practice in the organization; and choose what they deem the best solution for the task at hand.

Organization #4

Developers do not use code to decide on project alternatives as this will delay the release of the artefact to be developed.

Summary Analysis of When in doubt code it out

Choices between technology alternatives are made based on the requirements of the project, their previous experience, business modelling, research and best practice in the organizations. The developers choose what they deem the best solution for the task at hand as opposed to

using code to evaluate the designs which will most likely result in the project running beyond the scheduled time.

- **Whoever builds it should test it**

Organization # 1

The respondent pointed out that the developers do not officially test their own code. They do minor tests on the code themselves to ensure that it delivers the intended results. However, the official testing is conducted by an independent testing team.

Organization # 2

The team or individual who built a component is responsible for testing it. Unit and component testing are conducted by the developer(s) and the architect responsible for designing the artefact.

Organization # 3

Various testing; e.g. unit, component, system, user, acceptance is conducted by the team or individual responsible for building the component. The different architects may be involved in different types of testing.

Organization #4

The building and testing of the architecture components is conducted by two separate teams. The organization views this as a good approach. If there is a fix, it will have to go back to the build team to build it and then go back to the testing team for testing.

Summary Analysis of Whoever builds it should test it

Two out of the four organizations believe that the team or individual responsible for building a component should be responsible for testing it. Testing is usually conducted by the developer and architect(s) responsible for designing and developing it.

Leffingwell, *et al.* (2008: 5) believe that concurrent testing is a cornerstone practice of agile, and is a primary reason why quality is significantly higher in agile, without sacrificing developer productivity. Because testing represents complexity at its highest level, the team that codes the system should be the team that determines how to test the system.

This is not true within organization # 4, where the building and testing of the architecture components is conducted by two separate teams. The organization views this as a good approach. If there is a fix, it will have to go back to the build team to build it and then go back to the testing team for testing. In organization # 1, the developers do minor tests on their code themselves to ensure that it delivers the intended results, however official testing is conducted by an independent testing team.

- **Continuously build out system infrastructure**

Organization # 1

The EA team makes an effort to continuously build out the system infrastructure by ensuring that part of the solution design is infrastructure design.

Organization # 2

The respondent indicated that Knowledge managers are part of the EA team and work closely with the Information technology departments to ensure that the required infrastructure is in place to support the newly developed architecture artefacts.

Organization # 3

The infrastructure architects are responsible for developing an infrastructure design that will support the new architecture features. This infrastructure constantly evolves throughout the life of the project and is updated where needed to support the features that are added to the architecture.

Organization #4

The IT infrastructure is grown as EA project activities are underway to support the solution that is being designed to be put down.

Summary Analysis of Reuse existing architecture artifacts

The EA teams within the four organizations ensure that the IT infrastructure grows as EA project activities are underway in order to cater for the newly developed components and this is primarily the job of the infrastructure architect(s). Within organization #1, this is achieved by ensuring that part of the solution design is infrastructure design. Knowledge managers are

part of the EA team and work closely with the Information technology departments within organization # 2 to ensure that the required infrastructure is in place.

- **Respond to change over following a plan**

Organization # 1

After the completion of a project the EA team has a set of policies which dictate how to extend and make changes to the current architecture laid out in the EA team‟s change management policy document.

Organization # 2

Change control and governance processes dictate how changes can be made to the current architecture after the completion of a project.

Organization # 3

Changes to the developed architecture are made in accordance with policies developed within the organization based on research and best practices.

Organization #4

The organization has a set methodology in place for change management; and within change management, the EA team must ensure that the artifacts are adapted according to these policies.

Summary Analysis of Respond to change over following a plan

Each of the four organizations has a set of processes and practices which dictate how the EA team will make changes to the newly created architecture should a need arise. In organization # 1, these practices are laid out in the EA team‟s change management policy document. In organization # 2, change control and governance processes dictate how changes can be made to the current architecture after the completion of a project. Organization # 3 developed their change policies using research and best practices. Organization # 4 has a set methodology in place for change management.

- **Eliminate impediments for the EA team**

Organization # 1

The governance policies established by management are reviewed carefully and changed accordingly to ensure that they do not hinder the project progress of the EA team.

Organization # 2

The governance structure within the organization is extremely bureaucratic and as a result, in most cases, hinders project progress.

Organization # 3

The inclusion of the Chief Architect along with business leaders in the executive body responsible for governing the EA project ensures that the EA team has a means to voice their project concerns. These concerns are taken into account and changes made to the current governance structure or policies in order to enable the EA team to better achieve their project goals.

Organization #4

The respondent indicated that the problems encountered by the EA team during the project lifecycle are addressed promptly and effectively as the organization elects a program manager whose responsibilities include eliminating impediments for the EA team.

Summary Analysis of Eliminate impediments for the EA team

Management within three out of the four organizations is effective in eliminating impediments for the EA team when they arise. Within organization # 4, this is the responsibility of the program manager. In organization # 3, the inclusion of the Chief Architect along with business leaders in the executive body responsible for governing the EA project ensures that the EA team has a means to voice their project concerns. These concerns are taken into account and like in organization # 1 and if need be, changes are made to the current governance structure or policies in order to enable the EA team to better develop their project goals.

However, within organization # 2, the governance structure is extremely bureaucratic and as a result, in most cases, hinders project progress. This is most likely to the fact that organization

2 is a financial institution and the governance policies of financial institutions are traditionally strict and bureaucratic.

- **Management should be accountable for empowering the teams to deliver**

Organization # 1

The EA team is given freedom to make project decisions as sometimes the theoretical framework does not work in practice.

Organization # 2

The EA team is able to make recommendations that can influence project decisions. They however do not have the freedom to make project decisions.

Organization # 3

The EA team is able to make minor project decisions. However major project decisions have to be sanctioned by the stakeholder(s).

Organization #4

The EA team is given some creative freedom to make project decisions.

Summary Analysis of Management should be accountable for empowering the teams to deliver

The EA teams within organization #1, organization # 3 and organization # 4 are given some freedom to make project decisions as sometimes the theoretical framework does no work in practice. These decisions generally relate to minor project concerns; major decisions have to be sanctioned by the stakeholder(s).

However, this is not the case in organization # 2, where the EA team is only able to make recommendations that can influence project decisions. They however do not have the freedom to make the project decisions. This again can be attributed to the bureaucratic nature of organization # 2.

- **Evaluate EA maturity**

Organization # 1

At the beginning of the project, the EA team evaluates the current IT architecture by doing a 360 assessment. This assessment includes, but is not limited to uncovering what has been done and what needs to be done in order to obtain a baseline.

Organization # 2

The EA project begins with a basic evaluation of the current architecture.

Organization # 3

The initial phases of the EA project involve evaluating the current architecture in order to be aware of what is currently in place, can be reused, needs to be removed; and what needs to be done to achieve the goals of the project.

Organization #4

In instances where the organization has formal EA and that EA is mature, the EA team evaluate the current EA maturity. However, initially where there was no formal EA, measurement only occurs at the end of the project to assess what results had been achieved.

Summary Analysis of Evaluate EA maturity

Within three out of the four organizations, the initial phases of the EA project involve evaluating the maturity of the organization‟s current architecture. This allows the EA team to understand what the state of the current architecture is and develop a road map with deliverables that allow the architecture to evolve from the current to the desired state.

This is in contrast to the practice within organization # 4, where evaluation of the current EA maturity occurs only in instances where the organization is deemed to have formal EA and that EA is mature. Initially, where there is no formal EA or the current architecture is not mature, measurement will only occur at the end of the project to assess what results had been achieved. This is ineffective practice. The results achieved by the EA project can only be measured by measuring the previous state of the architecture and thereafter measuring the state of the architecture on completion of the project, in order to view the changes that have been made. Neglecting the step of doing an initial maturity assessment of the architecture, even if an organization is not deemed to have formal enterprise architecture, will not provide

a fair assessment of the benefits that an EA project has brought. In this way, every EA project will be deemed to be a success, when in fact it may not have been. In addition, whether an organization has a formal EA practice, every organization has architecture, even if it may not refer to it as that and its maturity should be measured, in order to understand what needs to be done and to avoid re-creating components or processes and practices that may already be part of the organization.

- **Measure value added by each iteration**

Organization # 1

The respondent declared that the EA team is required to show measurable results after each iteration and present these results at the project status meetings.

Organization # 2

The value added by the EA project is measured after each iteration.

Organization # 3

Following each project iteration the EA team assesses the value added and presents the new features, if any, to the stakeholder(s). In some cases, value is realised after two or more iterations.

Organization #4

The value added by the project is not measured after each iteration as depending on the nature of the project, each individual iteration may or may not necessarily produce value. The value added by the project is shown as it arises after any number of iterations.

Summary Analysis of Measure value added by each iteration

The practice within three out of the four organizations is to measure the value added by the EA project after each iteration. This allows the EA team to show measurable results should there be any after each iteration and present these results at the project status meetings, enabling the project team members and the stakeholder(s) to have confidence in the benefits of the project and the stakeholder(s) to have confidence in their investment. Dziewulski, *et al.*(2003: 9)emphasise promoting and publishing success stories throughout the organization.

This is particularly important during the start-up phase of an EA program when participants struggle to find direction, make progress, and above all, to see results.

On the other hand, the practice within organization # 4 is not to measure after each iteration as depending on the nature of the project, each individual iteration may or may not necessarily produce value. The value added by the project is shown as it arises after any number of iterations. Such a practice could result in some benefits while present, not being shown by the EA team. The benefits that the EA team will most likely be aware of are the completion of a new component whose deployment generates a benefit of some sort. However, there are different time frames for different value concepts; some are realized sooner than others. For example, there is value added in terms of having a defined architecture, thereby accelerating change by allowing management to have a view of the organization‟s current structure and capabilities. There could also be value added from a cost of ownership perspective where the reuse of existing artifacts and reduction in duplication in the organization leads to a reduction in cost of ownership. It is therefore a mistake not to measure and attempt to show the value added after each project iteration.

11.4 Holistic Analysis of the Results

The previous section presented a comparative factor analysis of the extent of understanding of enterprise architecture, agility and agile EA; the methods used to measure the agility of EA as well as each of the factors affecting the development of agile EA at the four organizations participating in the empirical study.

This section provides a holistic analysis of the extent of understanding of enterprise architecture, agility and agile EA; the methods used to measure the agility of EA as well as each of the factors affecting the development of agile EA at the four organizations participating in the empirical study.

11.4.1 The extent of understanding of enterprise architecture

This subsection of the holistic analysis of the results analyses the extent of understanding of EA at the four South African organizations.

The results based on investigating the extent of understanding of enterprise architecture support the assumption in previous research in which it has often been seen that the majority of researchers and those involved in the EA effort tend to refer to the collection of artifacts that are produced as the “enterprise architecture” (Gartner, 2008a: 2). It cannot be emphasized

strongly enough that this view is flawed. The process by which those artifacts are derived and applied is far more important than the artifacts themselves. By focusing the definition of EA on the process that creates, applies and maintains the artifacts, the proper emphasis is maintained. This single-minded focus on deliverables is a mistake because it can lead to mountains of artifacts (requirements, models, principles, guidelines, standards) that are not necessarily connected to the strategic imperatives of the enterprise and are therefore not leveraged across the organization. EA is both a management program consisting of phases that overlap and a documentation method supported by an EA framework that together provides an actionable, coordinated view of an enterprise‟s strategic direction, business processes, information flows, and resource utilization (Bernard, 2004: 33).

11.4.2 The extent of understanding of agility and agile EA

This subsection of the holistic analysis of the results analyses the extent of understanding of agility and agile EA at the four South African organizations.

The concept of “agility” is one that was well understood by the respondents which confirms previous literatures‟ assumptions that agility is a concept that is becoming of increasing importance to organizations. Their definitions differed slightly, however they are all in line with the view of agility as the ability to react to upcoming changes effectively and efficiently.

However, when asked what they understood by the term, “Agile Enterprise Architecture,” one of the respondents indicated that they hadn‟t considered agility within the context of EA. This is most likely due to the belief that each architecture effort, if done as a part of a fresh full life-cycle project, is meant to deliver agility, therefore architecture is done *for* agility rather than *with* agility. However, Lin *et al*. (2006) believe that the lack of a systematic approach to agility does not allow companies to develop the necessary proficiency in change, a prerequisite for agility.

The other three respondents indicated that the term related to an architecture which could rapidly and cost effectively adapt to, or exploit changes in customer/mission demands, or respond to uncertainties and unexpected conditions in the environment. This view of agile EA is flawed as it focuses on the agility of the end product(s) created by the EA project, as opposed to the agility of the process of EA which is the view taken in this thesis. The respondents‟ view of agile EA is as a result of their definition of EA focusing on EA as a noun, in that it relates that to the artifacts produced by the process of architecting and not that EA relates not only to the artefacts produced but also to the process of producing them;

therefore the agility/adaptability relates to how easily the architecture created by the architects can be changed. It is in line with assumptions of previous literature which highlight that most EA practitioners and researchers believe that architecture is done *for* agility rather than *with* agility (Stevenson, 1995).

11.4.3 The methods used to measure the agility of EA

This subsection of the holistic analysis of the results analyses the methods used to measure the agility of EA at the four South African organizations.

The fact that three of the four organizations were reported to use methods available within their chosen EA frameworks for measuring the agility of EA has further compounded the assumption provided in the literature that the need for agility is becoming more prevalent (Jain, *et al*., 2008: 367). An organization that can change and adapt quickly to changing circumstances is increasingly being viewed as a winning strategy (Lin, *et al*., 2005: 353). As such, while agility is deemed intangible, organizations have seen that it is frequently part of some critical business decision and have therefore made an effort to measure it. The current practice has therefore evolved from the assumptions in the literature that agility can‟t be measured (Hubbard, 2007: 2). However, the focus of the organizations‟ measurements of agile EA is on the created architecture, which is as a result of their definitions of EA focusing on EA as a noun. There were no organizations in the study that reported any methods for the measurement of agility of the process of EA.

11.4.4 The factors affecting the successful implementation of agile EA

This subsection presents the holistic analysis of the factors for the successful development of agile EA at the four South African organizations.

- **Needs and cost benefit analysis**

It is necessary to review the need for a project and weigh the benefits against the costs. Only then will an organization be able to decide whether a solution is worth implementing, as they may invest a lot of time and money in solving a problem that is not worthy of this effort.

- **Top management buy-in**

Top management buy-in is a very important element in the development of agile EA since the level of clarity on the project, as well as enthusiasm and encouragement for the corporate transformation is in its leadership team.

- **Common EA definition**

It is important for everyone involved with the EA effort to have a common understanding of EA. Varying views lead to a lack of unified knowledge, presence of varying cultures, and policies that prevent efficient management of enterprise processes, innovation and building software and create a great deal of confusion that affects the EA project.

- **Common Agility and Agile EA definition**

As with EA, it is important for an organization to have a consistent definition for agility and agile EA. This will allow them to be in the best position to sense environmental change and respond effectively to that change.

- **Management should communicate vision**

For a project to be successful today, creating a vision is a necessity and not a luxury. This is due to the fact that no project can make serious progress without a clear understanding of where it is going and what it is trying to achieve. An IT vision for the EA project must be communicated to the EA team.

- **Management should define goals**

It is important to define the goals of the project. This step defines the project‟s broad outcome and the steps required to achieve that outcome. Failure to define goals creates sensitivity within the organizations whereby the EA team don‟t know what to expect. This lack of comfort can in turn push a project into overruns, territory battles, personality clashes, missed milestones, and inevitably, unhappy clients. IT goals must be defined for the EA project.

- **Management should define expected results**

The expected results (objectives) of a project must be clear statements. Each objective should have its own purpose that drives the end result of the project. It is important that management spends sufficient time on this step, or completes it correctly, as a failure to do so will ensure an unsuccessful project completion. The expected results of the EA project must be articulated to the EA team from an IT perspective.

- **Management should engage in communication planning**

The development of communication policies and structures that project team members understand and adhere to is vital for project success. In this way, each member of the EA project team is aware of the means and channels in which they are able to communicate with other team members as well as any meetings etc., which they are required to attend.

Bringing a project in on-time and within budget requires the input and cooperation of every member of the team, and it is essential that every team member maintain clear lines of communication throughout all stages of the project.

- **Management should develop governance structures**

Completing a project within the predetermined time and costs constraints and with the end product or service performing to expectations requires a robust governance structure which defines accountabilities and responsibilities for strategic decision-making across the project.

- **Elect project sponsor**

As a project sponsor is the person who sees a need for change, has the authority to make something happen and will be responsible for funding, the EA project cannot exist without a project sponsor.

- **Elect executive body**

Every EA project should have an executive body that drives the success of the project and which is usually comprised of the senior IT and no-IT executives within the organizations. The chief architect should also be part of the executive body and amongst other duties, will ensure that the concerns of the EA team are put forward to management.

- **EA teams should commit on behalf of themselves**

It is important for members of the EA team to be committed to the success of the project. Commitment and compliance are two very different states for project teams. A compliant team is one that shows up because they are required to do so. They are working on the project only because it is the purpose for which they have been hired and are paid for. A committed team treats the project like their own garden or pet – they obsess over it, they care for it, they own it. They are thinking ahead of how to do it better, already solving the next three

problems that haven't been discovered. In addition, it is much easier to go from a committed team to a compliant one. A couple of poorly managed challenges can easily break the chain, and it is much harder to go from a compliant team to a committed one, therefore there should be an effort to sufficiently motivate team members to want to be part of the EA project at its inception.

- **EA team should be accountable for results**

Everyone on the team should be accountable for the project delivering the intended results. This increases understanding and acceptance of the architecture because they worked on it together as a team.

- **Focus on individuals and interactions over processes and tools**

The most effective communication is person-to-person, face-to-face. In some instances where person-to-person communication is not possible, the EA should make use of collaboration software tools. Ambler (2011) suggests that EA teams should make an effort to meet and interact in person as their communication will be further enhanced by a shared medium, for example a plain old whiteboard (POW). As they move away from this situation, perhaps by removing the shared medium or by no longer being face-to-face they are likely to experience a drop in communication effectiveness. When the EA team lose physical proximity, they lose the conscious and subconscious clues that such proximity provides. They also lose the benefit of multiple modalities and the ability to communicate through techniques other than words such as gestures and facial expressions. The ability to change vocal inflection and timing is also lost, people not only communicate via the words they say but how they say those words.

- **Harness power of collective team rather than an individual**

The EA team as a whole should be responsible for the created architecture. Architecture is far too important to leave in the hands of a single person no matter how bright they are; therefore architecture should be a team effort.

- **Customer collaboration over contract negotiation**

Communication with the customer(s) throughout the project life is important for project success in order to ensure that the requirements of the project are being made, value of the project to the customer is shown and any changing requirements of the project are discussed.

- **No monopoly on innovation**

The EA team should put processes in place to assure that innovation is not just incremental and near term. These processes could come in the form of e.g. a review process after each iteration, where the EA team members can explore any technology area in any way they want, as long as there is some correlation to the organization‟s goals. Ambler (2010) believes that his gives the team some mental down time to reflect, think and experiment outside of the everyday rigor and pressures of the iteration and release cycle. These innovative cycles may only be included if they will not negatively affect the budget and timeline of the project and as long as they have been sanctioned by the stakeholders. In order to ensure that they do not affect the timeline of the budget, they should be included in the initial timeline drawn up.

- **Define EA scope**

A project‟s scope and vision define the broad parameters of the project and provide the foundation for all subsequent steps in the project or program cycle. A clear scope sets the rough boundaries for what the project will attempt to do. This allows the members of the team to understand how much and what work needs to be done on a project and gives clarity and direction.

- **Reuse existing architecture artifacts**

Where possible, EA teams should reuse existing architecture artifacts to speed up the development process. The existence of artifacts which the EA team can reuse will also depend on how mature the organization is and the objective(s) of the selected project(s).

- **Develop situational specific strategies, processes, practices**

In the development of EA, each situation is different as projects have differing requirements and operate in different environments. The EA teams should use strict methods which help the definition of the strategies and practices for each situation and what products need to be worked through.

- **Determine schedule in terms of iterations and releases**

The EA project should be organized in terms of iterations and releases. This allows any deviations from the objectives of the project to be caught early, ensuring that the project will achieve its intended result(s) and for the stakeholder(s) to be shown a return on their

investment as soon as possible. There are different time frames for different value concepts; some are realized sooner than others.

- **Team that codes the system designs the system**

The team that designs the system should be responsible for coding it. The current practice is that the team that designs the system is not responsible for coding it. This is because EA teams within organizations consist of architects whose sole responsibility is to derive functional specifications (use case diagrams, etc). The majority of these architects have little or no programming experience and would not be able to handle the workload that designing and developing a system would bring. In addition, those that do have programming experience have long since been part of a development team and would therefore not be as efficient in coding the system as a team of developers. The EA team should therefore not only consist of EA architects, but also developers whose responsibility will be coding the models developed, instead of having architecture models thrust upon them on their completion by the architects. In this, way developers will always be aware of the EA project. Shirazi, *et al.* (2009: 183) have discovered that in some instances development teams do not know that the EA exists and therefore do not structure their development efforts in a way that follows the enterprise architecture. This arises in organizations where developers don‟t work with the enterprise architects. It is good practice to include the developers as this allows everyone involved to have their say in the architecture. This increases everyone‟s understanding and acceptance of the architecture because they worked on it together as a team. It also increases the chance that developers are willing to change aspects of the architecture when the architecture proves insufficient, or perhaps it doesn‟t scale as well as initially thought, because it is the group‟s architecture. In this way, the team that designs the system will effectively be able to code the system.

- **Create several architecture models in parallel**

Where possible, creating several architecture models in parallel is necessary for EA teams to deliver the EA project on time and under budget. During concurrent development, it becomes necessary to cross reference the models to ensure that they make sense in relation to the overall project.

- **Develop architecture in increments**

The architecture should be developed in increments, allowing it to evolve gradually throughout the life of the project, with project reviews after each iteration.

- **Build simplest architecture that can work**

Technology should be cut to the minimum needed to solve the problem at hand. The organization is not aided by creating artificial and unneeded layers of complexity. It is important to not over engineer a solution as additions come at a cost that is unacceptable and the Return on Investment (ROI) becomes more difficult to reach.

- **When in doubt code it out**

Agile practice advocates developers using code to evaluate a high design alternative, or high impact infrastructure choice (Ambler, 2010). In practice however, this is becomes difficult as if developers were to use code to make a decision when confronted with different possibilities it would require additional time and will most likely result in project overruns and additional costs. Approval would have to be obtained from the stakeholders, as it is their money being spent in this effort. From the answers gathered from the respondents, it is evident that EA projects have a set budget, one in which the organizations make all attempts to adhere to and would not make changes to lightly. As a result, it is unlikely that an organization would allow the time and additional budget that would be required. Therefore, in practice, using code to evaluate every design alternative is deemed not possible.

- **Whoever builds it should test it**

Concurrent testing is a cornerstone practice of agile, and is a primary reason why quality is significantly higher in agile, without sacrificing developer productivity. Because testing represents complexity at its highest level, the team that codes the system should be the team that determines how to test the system. Testing should be conducted by the developer and architect(s) responsible for designing and developing a component.

- **Continuously build out system infrastructure**

An effort should be made to ensure that the IT infrastructure grows as EA project activities are underway in order to cater for the newly developed components and this is primarily the job of the infrastructure architect(s).

- **Respond to change over following a plan**

The EA team should have a set of processes and practices which dictate how changes will be made to the newly created architecture.

- **Eliminate impediments for the EA team**

Management should be effective in eliminating any impediments for the EA team in order for the project to have the best chance of meeting its objective(s).

- **Management should be accountable for empowering the teams to deliver**

EA teams should be given some freedom to make project decisions as sometimes the theoretical framework does no work in practice. These decisions generally relate to minor project concerns; major decisions should be sanctioned by the stakeholder(s).

- **Evaluate EA maturity**

The initial phases of the EA project should involve evaluating the maturity of the organization‟s current architecture. This allows the EA team to understand what the state of the current architecture is and develop a road map with deliverables that allow the architecture to evolve from the current to the desired state.

- **Measure value added by each iteration**

The value added by the EA project should be measured after each iteration. This will allow the EA team to show measurable results should there be any after each iteration and present these results at the project status meetings; enabling the project team members and the stakeholder(s) to have confidence in the benefits of the project and the stakeholder(s) to have confidence in their investment.

It can be concluded from the analysis of the results of the empirical study that a better understanding of EA and how agility fits within the context of EA (agile EA) is required.

Understanding agile EA starts with a comprehensive understanding of both agility and EA. The respondents understand agility; therefore a comprehensive understanding of EA will inevitably bring with it the proper understanding of how agility fits within the context of EA.

The definitions created within the literature review for EA, agility and agile EA were comprehensive enough to encompass those provided by the respondents in the empirical study and are therefore deemed sufficient.

From the results of the empirical research, where all the respondents understood the concept of agility and three of them had methods for measuring agility of the created architecture; it is clear that agility is of growing importance to organizations. Agility within the context of EA is a concept which EA teams are slowly taking into account and realising as an important factor that has to be measured.

The organizations investigated in this study generally embrace the principals for the development of agile EA and believe that it is the clear future direction of their EA efforts.

Before an organization can successfully develop agile EA it needs to achieve a common understanding of the terms “enterprise architecture”, “agility” and “agile enterprise architecture.”

However, concerns are evident in the lack of team commitment at the start of an EA project. It is important for members of the EA team to be committed to the success of the project. Commitment and compliance are two very different states for project teams. A compliant team is one that shows up because they are required to do so. To understand the notion of "commitment," the old parable about the argument between the chicken and the pig as to whose contribution to a bacon-and-eggs breakfast was most important can be recalled. "I work incredibly hard to produce those eggs each morning," the chicken says. "And it's the centrepiece of the breakfast meal." "Well, there's no question that you're involved," replies the pig. "But I provide the bacon. I'm committed."

Another alarming factor is that organizations find it difficult to have a clear cut deliverable and choose an approach to get that deliverable. They tend to add additional features to the architecture to cater for events that they believe may occur. It is important to not over engineer a solution as additions come at a cost that is unacceptable and the Return on Investment (ROI) becomes more difficult to reach.

In addition, the EA team should not only consist of architects, but also developers whose responsibility will be coding the models developed, instead of having architecture models being thrust upon them on their completion by the architects. In this, way developers will

always be aware of the EA project and the EA team that designs the system will be responsible for coding it; a practice that is advocated by agile development methods.

Agile practice also promotes developers using code to evaluate a high design alternative, or high impact infrastructure choice (Ambler, 2010). In practice however, using code to evaluate every design alternative is not possible as it will result in budget and project time overruns.

In the end, while there may be an attempt to introduce elements of agility into the EA process to develop architecture in an agile manner, the success of these additions all add up to leadership and managing. The impression from current leadership literature (Wideman, 2000)seems to suggest that many projects are *over-managed and under-led*, where "management" has to do with the details of initiating, planning, executing, controlling and closing. Yet the opposite can also be true - where the person in charge *over-leads and under-manages*. Such projects fail not because of a lack of vision, mission and goals but simply because these leaders are not capable of managing the vision into existence. Although the overall goal of every EA project is to provide business results, the vision, goals and expected results of the EA project should be articulated to the EA team from an IT perspective, allowing them to understand what the EA project is intended to achieve and not only what is to be achieved from a business perspective.

11.5 Conclusion

This chapter provided an analysis of the results of the empirical study. The discussion involved a comparative factor analysis of the extent of understanding of enterprise architecture, agility and agile EA; the methods used to measure the agility of EA as well as each of the factors affecting the development of agile EA. A holistic analysis of the results was also provided.

The results of the empirical study and their analysis form the basis of reflections and enhancements in the structure of the theoretical framework which will be discussed in the next chapter.

Chapter 12: Revisions to the Theoretical Framework

Chapter eleven presented the analysis of the results of the empirical study.

The discussion included the impact that the results have on the research and the theoretical framework. A comparative factor analysis was made of the extent of understanding of enterprise architecture, agility and agile EA; the methods used to measure the agility of EA as well as each the factors affecting the development of agile EA. A holistic analysis of the results was also provided.

The aim of this chapter is to present the revisions to the theoretical framework based on the results of the empirical study and their analysis in Chapter 11.

It was concluded that the areas that required modification in the framework were in relation to goal definition, articulation of the expected project results, communication planning and the development of governance structures. These were modified to reflect the importance of IT management defining these factors to the EA team in relation to the EA project.

In addition, team transparency and the inclusion of developers in the EA team to support the enterprise architects were seen as important factors that should be added to the framework. The inclusion of developers in the EA team also affects the definition for agile EA which was modified accordingly.

As the analysis of the empirical results showed, developers using code to evaluate project alternatives does not work in practice. The corresponding factor was therefore removed from the framework.

12.1 Introduction

This section provides an introduction to the revisions to the theoretical framework.

An empirical investigation of the factors that affect the development of agile EA in selected South African organizations explores the applicability of the theoretical framework presented in Chapter 9. The empirical study was designed to further explore and verify the relevance and validity of this proposed framework. The study reveals the suitability and shortcomings of the theoretical framework which contributes to reflections and enhancements in the structure of the framework.

This chapter aims to revise the existing theoretical framework based on the results of the empirical study and their analysis in Chapter 11. A summary of the revisions to the framework is presented for each factor of the development of agile EA. A resulting framework is then illustrated and discussed.

12.2 Framework Revisions

The previous section provided an introduction to the revisions to the theoretical framework. This section details the revisions to the theoretical framework.

The empirical study explores a framework for the development of agile EA in selected South African organizations. The results of the empirical study are used to make revisions and adaptations to the theoretical framework.

The revisions relate to the EA Foundation Stage and the EA Approach Stage. The Revisions are initially presented according to the different stages in which they occur and thereafter they are summarized and the complete framework presented.

12.2.1 Revisions to the EA Foundation Stage

This subsection details the revisions to the EA foundation Stage in the theoretical framework.

The discussion will include a presentation of the EA Foundation Stage and thereafter a discussion of changes to the stage will be discussed. Finally, the revised EA Foundation Stage will be presented.

Figure 31 below shows the EA Foundation Stage of the theoretical framework.

Figure 31: Theoretical EA Foundation Stage

EA FOUNDATION STAGE
Needs and cost benefit analysis ***(1)***
Top management buy-in ***(1)***
Common EA definition ***(1)***
Common agility and agile EA definition ***(1)***
Management should communicate vision ***(1)***
Management should define goals ***(1)***
Management should define expected results ***(1)***
Management should engage in communication planning ***(1)***
Management should develop governance structures ***(1)***

(Own Contribution)

The empirical research uncovered that in some cases management is ineffective in communicating the goals of the selected project. At the highest level, a vision for the project is communicated, although the translation of that vision into the goals for the selected project is not communicated effectively. The communicated goals for the project are usually from a business perspective; however, how those business goals translate into IT goals for the selected EA project is very unclear. One of the factors to be taken into account during the EA Foundation Stage is that management is required to define goals. There is a need for IT goals to be defined for the EA project which are in line with the business goals. As EA is an IT function and IT is a specialized area, it falls to IT management to define these goals. This stage has been revised to specify that IT management should be required to define EA project goals. This will prompt IT management to develop IT goals from those articulated by business that EA teams can understand.

In addition, another factor to be taken into account during the EA Foundation Stage is that management is required to define expected results. When the goals are articulated from an IT perspective, an equal effort should be made by IT management to define the expected IT results for the EA project. This stage will be revised to specify that IT management should be required to define the expected results of the EA project.

The empirical study discovered that within all organizations there are various avenues for communication that have been available for employees by management. However, in some cases there are no formal communication structures for the EA project established by management. Various avenues and pockets of communication are present within the organization with no common understanding throughout the various areas. The development of effective communication structures for the EA team is usually a function of the project or program manager of the EA project and is therefore a function of IT. The factor in the EA Foundation Stage which relates to management being required to engage in communication planning will be revised to IT management being required to engage in communication planning. This will encourage communication structures to be developed specifically for the EA project as opposed to EA team members using any of the available methods within the organization to communicate. All of the members of the EA team must be aware of and use the methods of communication that have been agreed upon within the EA team.

A robust governance structure is required for projects to ensure that they are delivered on time and under budget. As a result, one of the factors in the EA Foundation stage articulates that

management should develop governance structures. However, the empirical research uncovered that in some instances, governance within the organization only exists on a high level, in terms of how the organization operates. From an EA perspective however, there is a lack of governance, in terms of the projects and how they will affect the organization and this has an adverse effect on the project‟s success. The factor in the EA Foundation Stage which relates to management being required to develop governance structures will be revised to IT management being required to develop EA governance structures.

Figure 32 below shows the revised EA Foundation Stage of the theoretical framework.

Figure 32: Revised Theoretical EA Foundation Stage

EA FOUNDATION STAGE
Needs and cost benefit analysis ***(1)***
Top management buy-in ***(1)***
Common EA definition ***(1)***
Common agility and agile EA definition ***(1)***
Management should communicate vision ***(1)***
IT Management should define EA project goals ***(1)***
IT Management should define EA project expected results ***(1)***
IT Management should engage in communication planning ***(1)***
IT Management should develop EA governance structures ***(1)***

(Own Contribution)

12.2.2 Revisions to the EA Approach Stage

This subsection details the revisions to the EA Approach Stage in the theoretical framework.

The discussion will include a presentation of the EA Approach Stage and thereafter a discussion of the changes to the stage will be discussed. Finally, the revised EA Approach Stage will be presented.

Figure 33 on the next page shows the EA Approach Stage of the theoretical framework.

Figure 33: Theoretical EA Approach Stage

EA APPROACH STAGE
Elect project sponsor ***(1)***
Elect executive body***(1)***
EA Team should commit on behalf of themselves ***(5)***
EA Team should be accountable for results ***(5)***
Focus on individuals and interactions over processes and tools ***(5)***
Harness power of collective team rather than an individual ***(5)***
Customer collaboration over contract negotiation ***(5)***
No monopoly on innovation ***(5)***
Define EA scope ***(1)***
Reuse existing architecture artifacts***(5)***
Develop situational specific strategies, processes, practices ***(5)***
Determine schedule in terms of iterations and releases ***(5)***
Team that codes the system designs the system ***(5)***
Create several architecture models in parallel ***(5)***
Develop architecture in increments ***(5)***
Build simplest architecture that can work ***(5)***
When in doubt, code it out ***(5)***
Whoever builds it should test it ***(5)***
Continuously build out system infrastructure ***(5)***

(Own contribution)

Agile practice advocates developers using code to evaluate a high design alternative, or high impact infrastructure choice (Ambler, 2010). The results of the empirical study show that in practice however, this is becomes difficult, as if developers were to use code to make a decision when confronted with different possibilities it would require additional time and will most likely result in project overruns and additional costs. Approval would have to be obtained from the stakeholders, as it is their money being spent in this effort. From the answers gathered from the respondents in the empirical study, it is evident that EA projects have a set budget, one in which the organizations make all attempts to adhere to and would not make changes to lightly. As a result, it is unlikely that an organization would allow the time and additional budget that would be required. Therefore, in practice, using code to evaluate every design alternative is deemed not possible. The factor in the EA Approach Stage entitled, "When in doubt code it out," which relates to developers to using their coding skills to evaluate high design alternatives will be removed from the framework as it does not work in practice.

There are many benefits of the team that designed the system being responsible for coding it. The empirical study showed that the current practice is that the team that designs the system is not responsible for coding it. This is because EA teams within organizations consist of architects whose sole responsibility is to derive functional specifications (use case diagrams, etc.). EA teams should therefore not only consist of architects, but also developers who will be responsible for developing the models created by the architects. This comes with it many benefits and is in line with agile practice. The factor entitled, "EA team should consist of developers and architects," will be added to the EA Approach Stage and will have an agility score of 5, indicating the high level of agility implementing such a practice would introduce.

The addition of developers to the EA team has also affected the definition developed in chapter 4 for agile EA. Previously the definition stated that, "…. Enterprise architects compose holistic solutions that address the business challenges of the enterprise and support the governance needed to implement them (Gartner, 2008a: 2; Lin *et al*., 2006; Madni, 2008: 50; Tsourveloudis and Valavanis; 2002: 330). " As the theoretical framework has been modified to advocate the inclusion of developers in the EA team, it follows that the definition for agile EA should be altered as well to show this change. The definition will be changed to, "…. Enterprise architects *and developers* compose holistic solutions that address the business challenges of the enterprise and support the governance needed to implement them (Gartner, 2008a: 2; Lin *et al*., 2006; Madni, 2008: 50; Tsourveloudis and Valavanis; 2002: 330)."

The results of the empirical research also uncovered that within one of the organizations there is a tendency to not share intellectual capital due to the direct link to job security. Transparency within project teams allows greater agility for managing project changes and allows for better strategic planning, management, efficiencies and costs savings. The lack of transparency and visibility prevents team members and stakeholders from making factual, metrics-based project decisions and therefore negatively affects project success (Oracle, 2009). Transparency is therefore an important factor that must be taken into account when attempting to develop agile EA. As such, the factor, "Team transparency" will be added to the EA Approach Stage. Team transparency is something which should be present in all EA teams, even those choosing to implement more rigid, traditional development approaches, in order to better achieve the goals of the EA project. Therefore while having team transparency will aid the EA team to better achieve their goals, the level of agility introduced by such a practice will be low as it is more a supporting factor to developing agile EA. Team transparency is therefore given an agility score of 1.

Figure 34 below shows the revised EA Approach Stage.

Figure 34: Revised Theoretical EA Approach Stage

EA APPROACH STAGE
Elect project sponsor ***(1)***
Elect executive body***(1)***
EA Team should commit on behalf of themselves ***(5)***
EA Team should be accountable for results ***(5)***
EA team should consist of developers and architects***(5)***
Focus on individuals and interactions over processes and tools ***(5)***
Harness power of collective team rather than an individual ***(5)***
Team transparency ***(1)***
Customer collaboration over contract negotiation ***(5)***
No monopoly on innovation ***(5)***
Define EA scope ***(1)***
Reuse existing architecture artifacts***(5)***
Develop situational specific strategies, processes, practices ***(5)***
Determine schedule in terms of iterations and releases ***(5)***
Team that codes the system designs the system ***(5)***
Create several architecture models in parallel ***(5)***
Develop architecture in increments ***(5)***
Build simplest architecture that can work ***(5)***
Whoever builds it should test it ***(5)***
Continuously build out system infrastructure ***(5)***

(Own contribution)

12.2.3 Summary of Revisions to Theoretical Framework

This subsection summarizes the revisions to the theoretical framework presented in chapter 9, as a result of the findings of the empirical study and their analysis in chapter 11.

Table 8on the next page details the revisions made to the framework. The table consists of a presentation of the theoretical framework factors from chapter 9, a summary of the empirical results analysis for each factor from chapter 11 which prompted a change and the resulting change, if any to the factor. In some cases, factors have been modified, removed and others added to the framework. Where changes have been made, these have been placed in italics in the third column, which shows the proposed framework of factors for the development and measurement of agile EA. The section ends with a presentation of the amended agility scoring scale and the amended Framework for the Development and Measurement of Agile Enterprise Architecture.

Table 8: Revisions to Theoretical Framework

Proposed Framework of Factors for the development and measurement of agile EA	**Empirical Study Results**	**Revised Framework of Factors for the development and measurement of agile EA**
Needs and cost benefit analysis	It is necessary to review the need for a project and weigh the benefits against the costs.	Needs and cost benefit analysis
Top management buy-in	Top management buy-in is an important element, since the level of enthusiasm and encouragement for the corporate transformation is in its leadership team.	Top management buy-in
Common EA definition	It is important for everyone involved with the EA effort to have a common understanding of EA	Common EA definition
Common agility and agile EA definition	As with EA, is important for an organization to have a consistent definition for agility and agile EA	Common agility and agile EA definition
Management should communicate vision	For a project to be successful today, creating a vision is a necessity and not a luxury.	Management should communicate vision
Management should define goals	IT management should define IT goals for the EA project.	*IT management should define EA project goals*
Management should define expected results	IT management must define the expected results of the EA project from an IT perspective	*IT management should define EA project expected results*

Management should engage in communication planning	IT Management should engage in communication planning	*IT Management should engage in communication planning*
Management should develop governance structures	IT Management should develop EA governance structures	*IT Management should develop EA governance structures*
Elect project sponsor	The EA project cannot exist without a project sponsor	Elect project sponsor
Elect Executive body	Every EA project should have an executive body that drives the success of the project and which is usually comprised of the senior IT and no-IT executives within the organizations	Elect Executive body
EA should commit on behalf of themselves	Team members should want to be part of the EA project at its inception	EA should commit on behalf of themselves
EA team should be accountable for results	Everyone on the team should be accountable for the project delivering the intended results	EA team should be accountable for results
Focus on individuals and interactions over processes and tools	The most effective communication is person-to-person, face-to-face	Focus on individuals and interactions over processes and tools
Harness power of collective team rather than an individual	The EA team as a whole should be responsible for the created architecture	Harness power of collective team rather than an individual
No monopoly on innovation	The EA team should put processes in place to assure that innovation is not just incremental and near term	No monopoly on innovation

Define EA scope	A clear scope is required of the work to be done on the EA project	Define EA scope
Reuse existing architecture artifacts	Where possible, EA teams should reuse existing architecture artifacts	Reuse existing architecture artifacts
Develop situational specific strategies, processes, practices	EA teams should use methods which help the definition of the strategies and practices for each situation	Develop situational specific strategies, processes, practices
Determine schedule in terms of iterations and releases	The EA project should be organized in terms of iterations and releases	Determine schedule in terms of iterations and releases
Team that codes the system designs the system	The team that designs the system should be responsible for coding it	Team that codes the system designs the system
Create several architecture models in parallel	Where possible, creating several architecture models in parallel is necessary for EA teams to deliver the EA project on time and under budget	Create several architecture models in parallel
Develop architecture in increments	The architecture should be developed in increments	Develop architecture in increments
Build simplest architecture that can work	Technology should be cut to the minimum needed to solve the problem at hand	Build simplest architecture that can work

When in doubt code it out	If developers were to use code to make a decision when confronted with different possibilities it would require additional time and will most likely result in project overruns and additional costs	*Removed from framework – does not work in practice*
Whoever builds it should test it	The team that codes the system should be the team that determines how to test the system	Whoever builds it should test it
Continuously build out system infrastructure	An effort should be made to ensure that the IT infrastructure grows as EA project activities are underway	Continuously build out system infrastructure
Respond to change over following a plan	The EA team should have a set of processes and practices which dictate how changes will be made to the newly created architecture	Respond to change over following a plan
Eliminate impediments for the EA team	Management should be effective in eliminating any impediments for the EA team	Eliminate impediments for the EA team
Management should be accountable for empowering the teams to deliver	EA teams should be given some freedom to make project decisions as sometimes the theoretical framework does no work in practice	Management should be accountable for empowering the teams to deliver

Evaluate EA maturity	The initial phases of the EA project should involve evaluating the maturity of the organization‟s current architecture	Evaluate EA maturity
Measure value added by each iteration	The value added by the EA project should be measured after each iteration	Measure value added by each iteration
N/A	All members of the teams should share and have access to information pertaining to the EA project.	*Team transparency*
N/A	EA team should consist of developers as well as architects	*EA team should consist of developers as well as architects*

(Own contribution)

The changes made to the preliminary framework have affected the overall score of agility that an organization can achieve. The removal of a step within the EA Approach Stage of the framework with an agility score of 5 and the addition of two other factors with agility scores of 5 and 1 means that the maximum agility score that an organization can achieve is now107. An organization wishing to measure their current level of agility will evaluate their EA effort in order to see which of the factors for the development of agile EA they currently meet. They will then add their score together in proportion to the weighting of each factor to determine their overall score of EA agility.

Figure 35 below shows the amended scale, which an organization can use to evaluate their level of agility.

Figure 35: Amended Agility Scoring Scale

Agility Score	**LOW**	**MEDIUM**	**HIGH**
Range	**0 - 39**	**40 – 79**	**80 - 107**

(Own contribution)

As indicated above, a score in the range 0-39 corresponds to a low level of agility within the organization‟s EA practices. The range 40 – 79 corresponds to a medium level of agility and the range 80 to the maximum of 107 corresponds to a high level of agility.

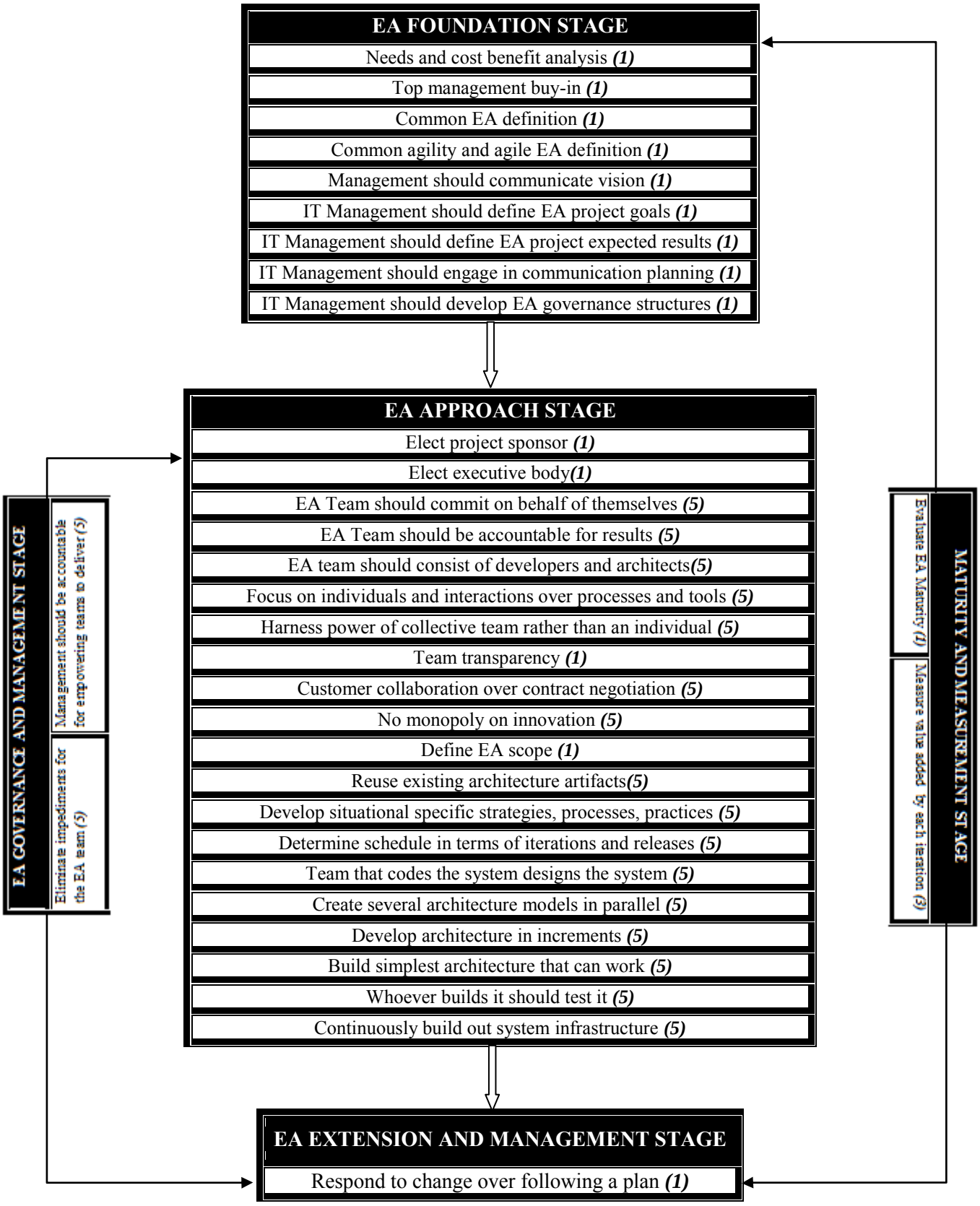
Organizations that according to the preliminary framework have low levels of agility are typically still using outdated, rigid approaches to EA, as opposed to a more agile approach which is favoured for today‟s business environment. Their approach is out of a lack of understanding of how agility fits within the context of EA and a belief that EA is done for agility, rather than with agility (Edwards, 2008: 1).

Organizations with a medium score of agility have most likely adopted more agile approaches to developing EA. Since the level of agility at this stage is still relatively low, it is possible however that these agile practices to developing EA were present in the organization without knowledge or focus on ensuring that EA was done in an agile manner.

A high score of agility would suggest that the organization sees the value of practicing EA in an agile fashion and has therefore focused on ensuring that their EA is done with agility.

Figure 36 on the next page provides an illustration of the complete, revised framework for the development and measurement of agile EA; showing the changes made to the stages discussed above and how all the stages are connected to one another.

Figure 36: Revised Framework for the Development and Measurement of agile Enterprise Architecture



(Own contribution)

12.3 Conclusion

The areas that required modification in the framework were in relation to goal definition, articulation of the expected project results, communication planning and the development of governance structures. These were modified to reflect the importance of IT management defining these factors to the EA team in relation to the EA project.

In addition, team transparency and the inclusion of developers in the EA team to support the enterprise architects were seen as important factors that should be added to the framework. The inclusion of developers in the EA team also affects the definition for agile EA which was modified accordingly.

As the analysis of the empirical results showed, developers using code to evaluate project alternatives does not work in practice. The corresponding factor was therefore removed from the framework.

A comprehensive evaluation of factors for the development of agile EA is essential before an organization can successfully develop agile EA. The literature survey, the preliminary framework and the results of the exploratory study led to the development of the theoretical framework for the development and measurement of agile EA in chapter 9. The empirical study was developed to test the important aspects of this framework. The interviews conducted with each of the organizations confirmed the 34 factors for agile EA development.

A general review of the framework indicates that the factors for the development of agile EA were well explored and developed in relation to the previous literature. The resulting framework provides organizations with the facility to assess the level of agility of their EA practice.

This chapter presented the revisions to the theoretical framework based on the results of the empirical study and their analysis in Chapter 11. The next chapter will conclude the research by identifying the most significant contributions of the research and by suggesting areas of future work.

Chapter 13: Conclusion

Chapter twelve presented the revisions to the theoretical framework based on the results of the empirical study and their analysis in Chapter 11.

The discussion included the revisions to the theoretical framework and a presentation of the complete, amended framework.

The aim of this chapter is to conclude the research by identifying the most significant contributions of the research and by suggesting areas of future work.

13.1 Introduction

The need for a more agile approach to developing enterprise architecture and a way to measure the level of agility of EA has become more and more significant in organizations. This begins with a better understanding of enterprise architecture, agility, how agility fits within the context of enterprise architecture as well as appropriate methods to measure agility.

The analysis in chapter 11 provided an indication of the organizations‟ outlook on the factors for developing agile EA as well as how they view EA, agility and agile EA. This analysis emphasized the importance of understanding enterprise architecture, agility and how agility fits within the context of enterprise architecture.

While there may be various factors that an organization can take into account in order to make the best effort in developing agile EA, their efforts will be for naught if they do not have the proper practices in place which create an environment which will allow EA teams to be able to execute their EA practices properly. In this way, management practices may not seem as agile factors, but they are also important in achieving the goal of developing agile EA.

As mentioned in chapter 4, the most important take away point is that it is all about people helping teams of teams build reliable, extensible, enterprise-class systems in an agile manner. Fancy tools based on theoretically sound frameworks, meta-models, or modelling languages are great to have but they will not do anything if the EA team does not use them. It‟s all about people. Sophisticated models and documents are interesting to create, but they offer little value if nobody reads them. It‟s all about people.

13.2 Contributions of the Research

This section discusses the contributions the research has made to the body of knowledge.

This research investigated the extent of understanding of enterprise architecture, agility and agile EA; the methods used to measure the agility of EA, as well as each of the factors affecting the development of agile EA. This was in an attempt to create comprehensive definitions for EA, agility and agile EA; as well as discover agility measurement techniques, which would lead to a framework for the successful development and measurement of agile EA.

By analysing the extent of understanding of enterprise architecture, this research established that organizations‟ current view of EA is limited in focusing on the deliverables derived by the EA process. The organizations don‟t seem to view implementation to be within the scope of EA and yet that is where it should play a central role. The current view uncovered in the research supports the assumption in the literature review in which it has often been seen that the majority of researchers and those involved in the EA effort tend to refer to the collection of artifacts that are produced as, “the enterprise architecture” (Gartner, 2008a: 2). It cannot be emphasized strongly enough that this view is flawed. The process by which those artifacts are derived and applied is far more important than the artifacts themselves. There is no sign that organizations are looking to change their view of EA to focus not only on the artifacts produced, but also on the process of producing them. However, by focusing the definition of EA on the process that creates, applies and maintains the artifacts, the proper emphasis would be maintained.

The definition created in chapter 2 for EA was comprehensive enough to encompass those provided by the respondents, correctly focused on the process of EA and is therefore deemed sufficient. Therefore, after an analysis of the responses provided by the respondents of the organizations with respect to EA, there was no need to change the definition created in the literature. EA is defined within the context of the research as, “the process of translating business vision and strategy into effective enterprise change by creating, communicating and improving the key requirements, principles and models that describe the enterprise's future state and enable its evolution. The scope of the enterprise architecture includes the people, processes, information and technology of the enterprise, and their relationships to one another and to the external environment. Enterprise architects compose holistic solutions that address

the business challenges of the enterprise and support the governance needed to implement them (Gartner, 2008a: 2)."

The analysis of the extent of understanding of agility and agile EA within organizations established that the concept of "agility" is one that was well understood by the respondents which confirms previous literature assumptions that agility is a concept that is becoming of increasing importance to organizations. The definition created in chapter 3 for agility was comprehensive enough to encompass those provided by the respondents and is therefore deemed sufficient. Therefore, after an analysis of the responses provided by the respondents of the organizations with respect to agility, there was no need to change the definition created in the literature. Agility within the context of this research is defined as, "the ability of an enterprise to operate profitably in a rapidly changing and continuously fragmenting global environment by producing high quality, high performance customer-configured goods and/or services and by sensing environmental change and responding efficiently and effectively to that change. It is the outcome of technological achievement, advanced organizational and managerial structure and practice as well as a product of human abilities, skills and motivations (Gartner, 2006: 2; Tsourveloudis and Valavanis, 2002: 330)."

However, when asked what they understood by the term, "Agile Enterprise Architecture," the respondents indicated that the term related to an architecture which could rapidly and cost effectively adapt to, or exploit changes in customer/mission demands, or respond to uncertainties and unexpected conditions in the environment. This view of agile EA is flawed as it focuses on the agility of the end product(s) created by the EA project, as opposed to the agility of the process of EA which is the view taken in this thesis. The respondents" view of agile EA is as a result of their definition of EA focusing on EA as a noun, in that it relates that to the artifacts produced by the process of architecting and not that EA relates not only to the artifacts produced but also to the process of producing them; therefore the agility/adaptability relates to how easily the architecture created by the architects can be changed. It is in line with assumptions of previous literature which highlight that most EA practitioners and researchers believe that architecture is done *for* agility and cannot be done *with* agility (Stevenson, 1995).The definition created in chapter 4 for agile EA was comprehensive enough to encompass those provided by the respondents, correctly focused on the introduction of agility to the process of EA as well as the artifacts produced and was therefore deemed sufficient. Therefore, an analysis of the responses provided by the respondents relating to agile EA did not change the definition created in the literature. However, the

analysis of the factors in the framework for the development and measurement of agile EA revealed the need to change the definition for agile EA to reflect that developers should be part of the EA team. This change was discussed in the previous chapter. Agile enterprise architecture is defined within the context of this research as, "the systematic process of adhering to agile development principles while translating business vision and strategy into effective enterprise change by flexibly creating, communicating and improving key requirements, principles and models. The models should have agile characteristics embedded in them, describe the enterprise‟s future state while keeping options open as late as possible and enable its evolution. The scope of agile enterprise architecture includes the people, processes, information and technology of the enterprise, and their relationships to one another and to the external environment. Enterprise architects and developers compose holistic solutions that address the business challenges of the enterprise and support the governance needed to implement them (Gartner, 2008a: 2; Lin *et al*., 2006; Madni, 2008: 50; Tsourveloudis and Valavanis; 2002: 330). "

An investigation of available methods being used within organizations to measure the agility of EA uncovered that the majority (three out of four) were reported to use methods available within their chosen EA frameworks for measuring the agility of EA. This has further compounded the assumption provided in the literature that the need for agility is becoming more prevalent (Jain, *et al*., 2008: 367). An organization that can change and adapt quickly to changing circumstances is increasingly being viewed as a winning strategy (Lin, *et al*., 2005: 353). As such, while agility was previously deemed intangible, organizations have seen that it is frequently part of some critical business decision and have therefore made an effort to measure it. The current practice has therefore evolved from the assumptions in the literature that agility can‟t be measured. However, the focus of the organizations‟ measurements of agile EA is on the created architecture, which is as a result of their definitions of EA focusing on EA as a noun. There were no organizations in the study that reported any methods for the measurement of agility of the process of EA. As such, future research could look into testing the effectiveness of the measurement method within the framework for measuring the agility of EA.

However, organizations do believe that an agile approach to EA is the clear future direction of their development methods and all are making active attempts to adopt a strategy that supports the development of agile EA. Their efforts are however focused on developing an agile end product due to their focus of EA as a noun. They do so by ensuring that they follow

practices made available in the widely used EA frameworks to embed agility in the end product created by the process of architecting and would explain why when the respondents were asked whether there were any additional factors that they believed would introduce agility into the process of EA, none could be identified.

The definitions created and enhanced through the empirical study allowed the creation of the main contribution to the body of knowledge; the framework for the development and measurement of agile EA which comprises 34 factors. The comments from the respondents confirmed the applicability of these factors to the development of agile EA. The framework represents an efficient starting point for organizations considering developing agile EA.

13.3 Limitations and Future Research

The previous section discussed the contributions the research has made to the body of knowledge. This section discusses the limitations of the research as well as future research considerations.

Limitations of this Research include:

- Since this study‟s results are limited to an exploration of the extent of understanding of enterprise architecture, agility and agile EA; the methods used to measure the agility of EA as well as each the factors affecting the development of agile EA in four selected organizations in one South African province, they cannot be generalized to organizations in South Africa.
- This research does not investigate why organizations may currently be using more rigid approaches to developing EA.
- Developing agile EA is a new concept which organizations are beginning to take into account. The research does not investigate whether the age of the respondents, or the length of time spent in the architecture practice has any bearing on their understanding of enterprise architecture, agility and agile EA or their willingness to adopt more agile practices.

Future research in this area includes:

- The replication of this research in other South African organizations and provinces in order to generalize the extent of understanding of enterprise architecture, agility and

agile EA; the factors affecting the development of agile EA as well as discover additional methods used to measure the agility of EA.

- An investigation into sector specific organizations to uncover whether the development of agile EA is more prevalent in certain sectors than others.
- An investigation that will involve testing the effectiveness of the use of the framework to developing agile EA.
- An investigation that will involve testing the effectiveness of the framework in determining how agile an organization‟s architecture development process is.

13.4 Concluding Remarks

The strength of the framework for the development and measurement of agile EA involves the recognition and comprehensive understanding of enterprise architecture. A criticism of some of the approaches towards defining enterprise architecture was a failure to view it as a course of action that results in the creation of artifacts and not merely as the artifacts that are created by the process of architecting.

Through the literature review and empirical study, this research developed a comprehensive definition for enterprise architecture. The research also discovered how agility fits within the context of EA; uncovering comprehensive definitions for agility and agile EA and the best practices in agile EA development. The methods used to measure the agility of EA were also investigated. This led to a framework which represents an efficient starting point for organizations considering developing and measuring agile EA.

The investigation of the 34 factors for the development of agile EA provides for a comprehensive coverage of the requirements for the successful development of agile EA. Furthermore, this framework provides the organization with the facility to assess their architecture practices in order to determine the gap between the organization‟s current practices and those set out in the framework. At this stage, an organization can use the results of this initial analysis to draw up a set of project deliverables and road map that will take the organization from its current state to the desired state set out in the framework.

References

ALFRED, R. (2008) *How to Measure Software Agility* [Online] (Updated 19 June 2008) Available at: http://www.wired.com/science/discoveries/news/2008/06/dayintech_0619 [Accessed 22 March 2010].

ALLEN, P. (2009) *How to Measure Software Agility* [Online] (Updated 14 December 2009) Available at: http://www.cutter.com/architecture/fulltext/updates/2009/eau0916.html [Accessed 22 March 2010].

AMBLER, S. (2006) *Examining the Agile Manifesto* [Online] (Updated August 2011) Available at: http://www.ambysoft.com/essays/agileManifesto.html [Accessed 9 March 2012].

AMBLER, S. (2009) *Agile Enterprise Architecture* [Online] (Updated 15 August 2010) Available at: http://www.agiledata.org/essays/enterpriseArchitecture.html [Accessed 10 December 2010].

AMBLER, S. (2010) *Agile Architecture: Strategies for Scaling Development* [Online] (Updated 4 September 2010) Available at: http://www.agilemodeling.com/essays/agileArchitecture.htm [Accessed 10 December 2010].

AMBLER, S. (2011) *Communication on Agile Software Projects* [Online] (Updated December 2011) Available at: http://www.agilemodeling.com/essays/communication.htm [Accessed 12 December 2011].

ANDERSON, P., BACKHOUSE, G. (2008) *Enterprise Architecture - An introduction*. September 2008 ed. [In-house publication] UK: JISC.

ARTETA, B., GIACHETTI, R. (2004) A measure of agility as the complexity of the enterprise system. *Robotics and Computer-Integrated Manufacturing,* 20(6), pp. 495-503.

BERNARD, A. (2004). *An Introduction to Enterprise Architecture*. Bloomington, Indiana: AuthorHouse.

BILLOWS, D. (2010) *Getting Commitment from Your Team* [Online] (Updated 4 June 2010) Available at: http://www.4pm.com/articles/TeamCommirtment.pdf [Accessed 10 December 2011].

COCKBURN, A. (2005) *The declaration of interdependence for modern management or DOI* [Online] (Updated 23 March 2011) Available at: http://alistair.cockburn.us/The+declaration+of+interdependence+for+modern+management+or+DOI [Accessed 8 March 2012].

CHIEF INFORMATION OFFICER COUNCIL (2001) *A Practical Guide to Federal Enterprise Architecture.* February 2001 Version 1.0 [In-house publication] Chief Information Officer Council.

CHIEF INFORMATION OFFICER COUNCIL (2007) *FEA Practice Guidance.* November 2007 Version 1.1. [In-house publication] Federal Enterprise Architecture Program Management Office, OMB.

COHEN, L., LAWRENCE, M. (1994) *Research Methods in Education.* London: Routledge.

CHRISTIAN, I., ISMAIL, H., MOONEY, J., SNOWDEN, S., TOWARD, M. & ZHANG, D., 2001. "Agile manufacturing transition strategies." In Holby, H. (Ed.), *Proceedings of the Fourth SMESME International Conference.* Department of Production, Aalborg University. Aalborg, D.C, May 19 - 22, 2001, Pergamon Press, Inc: USA.

DIMARIO, M., CLOUTIER, R. & VERMA, D. (2008) Applying Frameworks to Manage SoS Architecture. *Engineering Management Journal*, 20(4), pp. 18-23.

DZIEWULSKI, A., IANNUZZI, J., MENZEL, R., RAMCHANDI, C., ROTHSCHILD, P. & TIEMANN, M. (2003) *Advancing Enterprise Architecture Maturity*. March 2003 ed. [Concept Level White paper] USA: Industry Advisory Council (IAC).

EDWARDS, C. (2006) *Agile Enterprise Architecture – Part 1*. December 2006 ed. [In-house publication] USA: ProcessWave.

FLOWER, M. (2000)	*The New Methodology.* [Online] (Updated 13 December 2005) Available at: http://martinfowler.com/articles/newMethodology.html [Accessed 01 March 2010].
FOWLER, M., HIGHSMITH, J. (2001)	*The Agile Manifesto.* [Online] (Updated August 2001) Available at: http://www.pmp-projects.org/Agile-Manifesto.pdf [Accessed 8 March 2012].
GARTNER (2006)	*Defining, Cultivating and Measuring Enterprise Agility*. 28 April 2006 ed. ID Number: G00139734 [In-house publication] Gartner
GARTNER (2007)	*Key Issues for Enterprise Architecture, 2007*. 29 March 2007 ed. ID Number: G00146809 [In-house publication] Gartner.
GARTNER (2008a)	*Gartner Clarifies the Definition of the Term „Enterprise Architecture".* 12 August 2008 ed. ID Number: G00156559 [In-house publication] Gartner.
GARTNER (2008b)	*Key Issues for Enterprise Architecture, 2008-2009*. [Online] (Updated 16 November 2009) Available at: http://www.btmg.biz/EA_Kitty3/Documents/Gartner/Gartner6.pdf [Accessed 05 March 2010].
GASSON, S. (2004)	Rigor in Grounded Theory Research: An Interpretive Perspective on Generating Theory from Qualitative Field Studies. In M. Whitman, & A. Woszczynski, *The Handbook of Information Systems Research* (pp. 79-102). Hershey: Idea Group Inc.
GIACHETTI, R. (2009)	Design for the Entire Business. *Industrial Engineer*, 41(6), pp.39-43.
GILB, T. (1988)	*Principles of Software Engineering Management*. London, UK: Addison-Wesley.
HANSEN, V. (2006)	*Enterprise Architecture – how to establish and sustain a successful EA*. MSc. Copenhagen: IT University of Copenhagen.
HARMON, P. (2003)	*Developing an Enterprise Architecture*. January 2003 ed. [In-house publication] USA: Business Process Trends.
HIGHSMITH, J. (2009)	*Agile Project Management: Adapting Over Conforming*. [Online] (Updated 17 August2009) Available at: http://www.informit.com/articles/article.aspx?p=1377271 [Accessed 8 March 2012].
HJORT-MADSEN, K. (2005)	*Managing and Implementing Enterprise Architectures in Government: Understanding Institutional Forces and IS Control*. PHD Status Report. Copenhagen: IT University Copenhagen and National IT and Telecom Agency.

HUBBARD, D. (2007) *How to Measure Anything Finding the Value of „Intangibles"'in Business*. New Jersey, USA: John Wiley & Sons.

HUNTER, M. (2004) Qualitative Research in Information Systems: An Exploration of Methods. In M. Whitman, & A. Woszczynski, *The Handbook of Information Systems Research* (pp. 291-304). Hershey: Idea Group Inc.

JAMES, G., HANDLER, R., LAPKIN, A., GALL, N. (2005) *Gartner Enterprise Architecture Framework: Evolution 2005*. 25 October 2005 ed. ID Number: G00130855 [In-house publication] Gartner.

JAIN, V., BENYOUCEF, L. & DESHMUKH, S. (2008) A New Approach For Evaluating Agility in Supply Chains Using Fuzzy Association Rules Mining. *Engineering Applications of Artificial Intelligence*. 21(3), pp.367-385.

KELLY, K. (1999) From Encounter to Text: Collecting Qualitative Data for Interpretive Research. In M. Terre Blanche, & K. Durrheim, *Research in Practice: Applied Methods for the Social Sciences* (pp. 379-397). Cape Town: UCT Press

KUTNICK, D. (2006) *Aligning and Measuring Agility.* [Online] (Updated 26 January 2006) Available at: http://www.gartner.com/it/products/podcasting/asset_143240_2575.jsp [Accessed 01 March 2010].

KHOSROWPOUR, M. (1996) "Information Technology Management and Organizational Innovations." In Mehdi Khosrowpour, *Proceedings of the 1996 Information Resources Management Association International Conference*. Washington, D.C, May 19 - 22, 1996,Idea Group Publishing: USA.

KIDD, P. (2000) *Two Definitions of Agility.* [Online] (Updated January 2000) Available at: http://www.cheshirehenbury.com/agility/twodefinitions.html [Accessed 08 March 2012].

LAND, R. (2003) "Applying the IEEE 1471-2000 Recommended Practice To a Software Integration Project." In SERP, (Software Engineering Research and Practice) *Proceedings of the International Conference on Software Engineering Research and Practice*. Las Vegas, Nevada, June 23 - 26, 2003,SERP '03: USA.

LAPPO, P., ANDREW, T. (2008)	*Assessing Agility.* November 2008. [In-house publication] UK: Systematic Methods Research Ltd.
LEE, G. XIA, W. (2010)	Toward Agile: An Integrated Analysis of Quantitative and Qualitative Field Data on Software Development Agility. *MIS Quarterly* 34(1), pp.87-114.
LEFFINGWELL , D. (2008)	*Agile Architecture: What is Intentional Architecture?* [Online] (Updated 8 March 2008) Available at: http://scalingsoftwareagility.wordpress.com/2008/05/08/agile-architecture-what-is-intentional-architecture/ [Accessed 05 March 2010].
LEFFINGWELL, D., MARTENS, R. & ZAMORA, M. (2008)	*Principles of Agile Architecture*. 14 August 2008 ed. [In-house publication] Colorado: Rally Software.
LIN, C., CHIU, H. & TSENG, Y. (2005)	Agility evaluation using fuzzy logic. *International Journal of Production Economics*, 101(2), pp.353-368.
MADISON, J. (2010)	Agile Architecture Interactions. *IEEE Software*, 27(2), pp. 41-48.
MADNI, A. (2008)	AgileTecting: A Principled Approach to Introducing Agility in Systems Engineering and Product Development Enterprises. *Journal of Integrated Design & Process Science*, 12(4), pp.49-55.
MORRISON, J. (2006)	*Define Project Scope & Vision.* [Online] (Updated May 2006) Available at: http://docs.google.com/viewer?a=v&q=cache:2GejW7nOssMJ:www.panda.org/standards/1_2_scope_and_vision/+clear+project+scope&hl=en&gl=za&pid=bl&srcid=ADGEESh8SyEQn2-HV3O6I9tsBZAIXT0JbXsrFmCdP-Qb4hVSfCKPej1EDRtvuu9pnT1ljQvkKuIGC-QAcZ9LYZtgTgLKKBsOfUq-cqaIn_kOt9IsPGImq8-0rCkoTu0ZeTP2u2Rq06pv&sig=AHIEtbSc8ZvsVGWPgdxhcZYTTFEy10BJdw [Accessed 10 December 2011].
MYERS, M. (1997)	*Qualitative Research in Information Systems.* [Online] (Updated 5 November 2008) Available at: www.qual.aucland.ac.nz/[Accessed 26 May 2011].
NAIR, S. (2008)	*Agility Requirements: The Groundwork Of Agile Architecture – Part I.* [Online] (Updated 6 July 2008) Available at: http://earchpal.wordpress.com/2008/07/06/agility-requirements-the-groundwork-of-agile-architecture-%E2%80%93-part-i/ [Accessed 05 March 2010].

NOLAN, R., MULRYAN, D. (1987) Undertaking an Architecture Program. *Stage by Stage*. 7(2), pp.63-64.

ORACLE (2009) *Gaining Transparency in Construction & Engineering.* [Online] (Updated September 2009) Available at: http://www.oracle.com/us/products/applications/primavera/gaining-transparency-168400.pdf [Accessed 11 December 2011].

PACKER, M. (1999) *Interpretive Research: The Logic of Inquiry.* [Online] Available at: http://www.mathcs.duq.edu/~packer/IR/IRlogic.html[Accessed 11 July 2011].

PERKS, C., BEVERIDGE, T. (2003) *Guide to Enterprise IT Architecture*. New York, USA: Springer-Verlag.

PRAHALAD, C.K. (2009) In Volatile Times, Agility Rules. *Business Week*. 4147(1), pp.80-80.

ROSENBLATT, T. (2009) *Agile Manifesto Principles.* [Online] (Updated 2 March 2009) Available at: http://www.cloudspace.com/blog/2009/03/02/agile-manifesto-principles/ [Accessed 8 March 2012].

ROWLANDS, B. (2005) Grounded in Practice: Using Interpretive Research to Build Theory. *The Electronic Journal of Business Research Methodology*, 3(1), pp.81-92.

SEITZ, J. (2008) Business Agility. *Communications News*, 45(10), pp.16-19.

SESSIONS, R. (2006) *A Better Path to Enterprise Architectures*. [Online] (Updated 21 March 2006) Available at: http://www.objectwatch.com/white_papers.htm#ABetterPath [Accessed 01 June 2010].

SESSIONS, R. (2007) A Comparison of the Top Four Enterprise-Architecture Methodologies. *Microsoft Developer Network*, 466232 (3), pp.1-28.

SCHEKKERMAN, J. (2005) *The Economic Benefits of Enterprise Architecture*. Victoria, BC, Canada: Trafford.

SCHMIDT, R., LYYTINEN, K., KEIL, M., & CULE, P. (2001)	Identifying Software Project Risks: An International Delphi Study. *Journal of Information Systems*, 17(4), pp.5-36.
SHUJA, A.K. (2010)	*Enterprise Architecture: What is it and why should we care?* [Online] (Updated 10 March 2010) Available at: http://www.shuja.info/_documents/ea_why_should_we_care.pdf [Accessed 17 June 2010].
SPEWAK, H. (1992)	*Enterprise Architecture Planning*. New York, USA: Wiley-QED.
STEVENSON, A. (1995)	*Positioning Enterprise Architecture.* [Online] (Updated 6 November 1998) Available at: http://users.iafrica.com/o/om/omisditd/denniss/text/eapositn.html [Accessed 02 March 2010].
SUTHERLAND, J. (2010)	*Agile Principles and Values* [Online] (Updated December 2010) Available at: http://msdn.microsoft.com/en-us/library/dd997578.aspx [Accessed 8 March 2012].
TANG, A., HAN, J. (2004)	*A Comparative Analysis of Architecture Frameworks* [Online] (Updated 25 August 2004) Available at: http://www.swin.edu.au/ict/research/technicalreports/2004/SUTIT-TR2004.01.pdf [Accessed 7April 2012].
TSOURVELOUDIS, N., VALAVANIS, K. (2002)	On the Measurement of Enterprise Agility. *Journal of Intelligent and Robotic Systems,* 33(3), pp.329-342.
TARABANIS, K., PERISTERAS, V. & FRAGIDIS, G. (2001)	"Building an Enterprise Architecture For Public Administration: A High-Level Data Model For Strategic Planning." In Global Co-operation in the New Millennium. *The 9th European Conference on Information Systems*. Bled, Slovenia, June 27 - 29, 2001,ECIS: Slovenia
THE OPEN GROUP (2003)	*The Open Group Architecture Framework (TOGAF 8.1.1 'The Book').* 2007 ed. San Antonio: Van Haren Publishing.

THE OPEN GROUP (2009)	*TOGAF Version 9 Enterprise Edition.* [Online] (Updated 4 February 2009) Available at: http://www.opengroup.org/architecture/togaf9-doc/arch/ [Accessed 01 March 2010].
U.S DEPARTMENT OF DEFENSE (1986)	*Technical Architecture Framework For Information Management (TAFIM)*. Volumes 1-8, Version 2.0, 1986 ed. [In-house publication] Reston VA: DISA Center for Architecture.
VINODH, S., SUNDARARAJ, G. & DEVADASAN, S. R. (2010)	Measuring organisational agility before and after implementation of TADS. *International Journal of Advanced Manufacturing Technology, 47(5-8), pp.809-818.*
WIDEMAN, M. (2000)	*Project Vision: Yes or No?*[Online] (Updated 6 November 2000) Available at: http://www.maxwideman.com/musings/project_vision.htmhtml [Accessed 02 December 2011].
WILLIS, J. (2007)	*Foundations of Qualitative Research: Interpretive and Critical Approaches.* California: SAGE.
YIN, R. (2003)	*Case Study Research: Design and Methods.* California: SAGE.
YLIMAKI, T., HALTTUNEN, V. (2006)	Method engineering in practice: A case of applying the Zachman framework in the context of small enterprise architecture oriented projects. *Information Knowledge Systems Management*, 5(3), pp. 189-289.
ZACHMAN, J. (1987)	A Framework for Information Systems Architecture. *IBM Systems Journal, 26(3) pp.112 - 116.*

Appendices

Appendix A – Survey Questions

There are numerous methods available in the widely used architecture frameworks that measure the agility of the end product (the IT architecture) created by the architects. This research focuses on measuring the agility of the process of creating enterprise architecture.

It was clear from the literature uncovered in the research on the process of EA that while various authors have slightly different views on the process of EA in terms of when activities are to be done as well as names for different stages, there are some common themes that emerge. This has led to the identification of five main stages in the process of EA. Section A of the framework questions are therefore split into three main stages; an EA Foundation Stage, an EA Approach Stage and an EA Extension and management Stage and also contains two supporting stages; an EA Maturity and Measurement Stage as well as an EA Governance and Management Stage that occur throughout the three main stages. Each stage contains considerations/factors that must be taken into account when attempting to develop agile EA. Section B examines whether the organization has any methods for measuring the agility of EA and any further comments the respondent may make. Section C enquires about the respondent‟s job title and other relevant identifying characteristics, as well as general information about the organization itself.

In all instances of the questions, where possible, the respondent is asked to elaborate on their answers as opposed to a simple yes/no answer.

A.1 EA Foundation Stage

The EA Foundation Stage is the stage which occurs at the beginning of the EA process.

A.1.1 Is a needs and cost-benefit analysis done before embarking on an Enterprise Architecture (EA) Project?

A.1.2 Are top management required to sign off on potential EA projects?

A.1.3 What is your understanding of the term “enterprise architecture” and does everyone involved with the EA project within the organization have a common understanding of what EA is?

A.1.4What is your understanding of the term “agility” and does everyone involved with the EA project within the organization have a common understanding of what agility is and how it relates to the EA project?

A.1.5 Does management clearly communicate their vision and goals of the selected EA project (s)?

A.1.6 Does management clearly define the expected results of the EA project?

A.1.7 Does management set up efficient communication structures to be used throughout the project?

A.1.8 Does management set up robust governance structures that will effectively govern the EA project?

A.2 EA Approach Stage

The EA Approach Stage occurs if an organization decides to commence on an EA program based on conclusions from the foundation stage

A.2.1 Is a project sponsor/ thought leader elected for the project?

A.2.2 Is an executive body elected to ensure that top management is involved in high-level decisions of the EA project?

A.2.3 Are EA team members motivated and committed to a project at its inception?

A.2.4 Is the EA team held directly accountable for the success of the project?

A.2.5 Where possible, is there emphasis placed on face to face team collaboration, as opposed to collaboration software tools?

A.2.6 As opposed to one person, is the EA team as a whole responsible for the architecture?

A.2.7 Does the EA team communicate and collaborate with the customer(s) throughout the life of the EA project?

A.2.8 Are there processes put in place to ensure that innovation is not just once-off or near term?

A.2.9 Does the EA team have a clear scope of the project in which to work?

A.210 Does the EA team reuse existing architecture artifacts?

A.2.11 Does the EA team develop situationally specific strategies and practices?

A.2.12 Is the schedule of the project organized in terms of iterations and releases?

A.2.13 Does the team that designed the system, code the system?

A.2.14 In situations where it is possible, are several architecture models created in parallel?

A.2.15 Is the architecture developed in increments?

A.2.16 Is technology cut to the minimum needed to solve the problem at hand by building the simplest architecture that can work?

A.2.17Do developers use code to evaluate the architecture models when selecting a design alternative or a high-impact infrastructure choice?

A.2.18 Is the team or individual who built a component also responsible for testing it?

A.2.19 While EA activities are underway, is there also an effort to continuously build out system infrastructure, ensuring that foundation for the new features is already in place?

A.2.7 How soon after the project‟s initiation is the value added by EA realized?

A.3 Extension and Management Stage

The Extension and Management Stage occurs once the initial enterprise architecture has been established.

A.3.1 After the completion of a project, does the EA team have a set of policies which dictate how to extend and make changes to the current architecture?

A.4 The EA Governance and Management Stage

The EA Governance (EAG) and Management Stage focuses directly on establishing and implementing EA in the organization as well as on setting the policy for how EA subsequently should be run and occurs throughout the lifecycle of a project.

A.4.1 Is management effective in eliminating impediments for the EA team?

A.4.2 Does management give the EA team some freedom to make project decisions?

A.5 The Maturity and Measurement Stage

A.5.1 Is the current EA maturity evaluated at the beginning if a project?

A.5.2 Is the value added by the EA project measured after each iteration?

B.1 Further Comments

This section enquires about whether the respondent‟s organizations has any methods that they use to measure the agility of EA and allows the respondent to comment on the questions; whether they were easy to understand and relevant to the topic. The respondent is also encouraged to make suggestions that would make the survey better, as well as suggest other factors that have been omitted that would encourage the development of agile enterprise architecture.

B.1.1 Does your organization have any methods for measuring the agility of EA?

B.1.2 How did you find the survey questions?

B.1.3 Do you have any suggestions that would make the survey better?

B.1.4 Do you have any additional factors that you believe would encourage the development of agile EA?

B.1.5 Do you have any additional comments or questions pertaining to the framework, research or survey?

C.1 Demographics of the respondent and the institution

This section enquires about information pertaining to the respondent and their organization. Short answers are permitted.

C.1.1 What is your job title?

C.1.2 What is your role in the organization?

C.1.3 Who do you report to directly?

C.1.4 How long have you been working for the organization?

C.2.1 How would you best describe your organization and its function?

C.2.2 How long have you been operational?

Appendix B – Detailed Empirical Results

The results below follow the chronological order of questions presented in the survey Appendix A.

Organization # 1

A.1 EA Foundation Stage

A.1.1 *Is a needs and cost-benefit analysis done before embarking on an EA Project?*

Prior to embarking on an EA project, the organization creates a needs and cost-benefit analysis and statement of work for the architecture which takes into account financial, human and other aspects of the project.

A.1.2 *Are top management required to sign off on potential EA projects?*

Top management review the project proposal and thereafter conduct a value realization; signing off on project worthy proposals.

A.1.3 *What is your understanding of the term "enterprise architecture" and does everyone involved with the EA project within the organization have a common understanding of what EA is?*

The EA team believe that enterprise architecture is the design of the entire make up of an organization. Everyone involved with the EA project is informed on what the organization"s view of EA is through a definition workshop which is run prior to any work being done to obtain a common understanding and inform people of the objectives of the project.

A.1.4 *What is your understanding of the term "agility" and does everyone involved with the EA project within the organization have a common understanding of what agility is and how it relates to the EA project?*

The respondent of organization # 1 indicated that the term, „agility" envisioned something responsive, or swift and within the context of EA; an EA that was adaptable or could be changed easily if a change was required. Like the term „enterprise architecture," the meaning as well as a shared understanding of agility is achieved through the definition workshop that runs prior to any work being done on the EA project.

A.1.5 *Does management clearly communicate their vision of the selected EA project (s)?*

Management communicates their vision for the project, initially through a needs and cost-benefit analysis and then later on in a definition workshop which runs prior to any work being done; and serves to inform project members of the project objectives and achieve a common understanding of project concepts.

A.1.6 *Does management clearly communicate the goals of the selected EA project (s)?*

Management successfully translates their vision for the project into a set of goals that will allow that vision to be achieved.

A.1.7 *Does management clearly define the expected results of the EA project?*

The expected results of the EA project are defined throughout the project from the preliminary stages until its completion. Initially the expected results are communicated in the needs and cost – benefit analysis and then later on in the definition workshop which also serves to inform people of the project objectives. Thereafter the objectives are highlighted after each project phase.

A.1.8 *Does management set up efficient communication structures to be used throughout the project?*

Management plans the way in which the EA team will communicate and ensures that the project team attends weekly status meetings. The EA project team also makes use of collaboration software tools, for example, SharePoint workspaces. These collaboration tools allow the EA team to have scheduled and unscheduled communication.

A.1.9 *Does management set up robust governance structures that will effectively govern the EA project?*

Management (sponsor/executive and architecture board) defines the way the EA project is to be run in the definition workshop which runs before any work for the EA project is done. The governance policies that are established at the project‟s inception are carefully derived by management to enable effective governance. These policies are reviewed at certain intervals to ensure that they are still affective and are changed if the need arises. The EA team makes regular check-ins of objectives with management in the weekly status meetings and the artifacts and objectives are communicated after each phase.

A.2 EA Approach Stage

A.2.1 *Is a project sponsor elected for the project?*

A project sponsor or executive is elected for the project prior to it being approved.

A.2.2*Is an executive body elected to ensure that top management is involved in high-level decisions of the EA project?*

An architecture review board comprising of senior members of the organization (Chief Architects, domain architects, CIO, etc.) is elected to ensure that top management is involved in high-level decisions of the EA project. This board enforces architectural policies and standards. Decisions on the EA project are communicated to business afterwards.

A.2.3 *Are EA team members motivated and committed to a project at its inception?*

The respondent pointed out that the position of „architect" within an organization is a senior one and therefore the EA team comprises of senior IT staff within the organization, who have reached that position not only due to their qualifications and experience but also because of their high level of motivation and commitment to the organization and its goals and their ability to motivate and encourage commitment from others.

A.2.4 *Is the EA team held directly accountable for the success of the project?*

The EA team is held accountable for meeting the deadlines, goals and results of the overall project.

A.2.5*Where possible, is there emphasis placed on face to face team collaboration, as opposed to collaboration software tools?*

The organization places a great emphasis on face to face team collaboration as this fosters a better exchange of ideas as opposed to collaboration over software tools. However the use of these tools is encouraged in instances where face to face meetings are not possible and communication is required.

A.2.6*As opposed to one person, is the EA team as a whole responsible for the architecture?*

Within the EA team, the Chief Architect is the main person responsible for the architecture and supervises and delegates responsibilities to the other members of the team.

A.2.7 *Does the EA team communicate and collaborate with the customer(s) throughout the life of the EA project?*

The EA team communicates and collaborates with the project sponsor/executive responsible for the project throughout its lifecycle.

A.2.8 *Are there processes put in place to ensure that innovation is not just once-off or near term?*

Following a project iteration a review is done to ensure that project is still in line with its requirements. Thereafter, ways to improve the project are discussed and if any are present, they are added to the project requirements.

A.2.9*Does the EA team have a clear scope of the project in which to work?*

The EA team is presented with a clear scope of the project in which to work from the initial definition workshop and the statement of work to be done for the project.

A.2.10*Does the EA team reuse existing architecture artifacts?*

Where possible, the EA team reuses existing artifacts. In some cases, the artifacts have become obsolete and part of the project work will be to replace them entirely.

A.2.11*Does the EA team develop situationally specific strategies and practices?*

The EA team develops a combination of event based strategies that will depend on the occurrence of certain events as well as set strategies for specific areas using well known industry frameworks/models, for example, the Mckinsey Model.

A.2.12*Is the schedule of the project organized in terms of iterations and releases?*

The respondent indicated that the project lifecycle is organized in terms of iterations and releases, with project reviews conducted after each iteration.

A.2.13 *Does the team that designed the system, code the system?*

The EA team that designs the system does not code the system. The EA team comprises of architects whose sole responsibility is to derive functional specifications (use case diagrams, etc.). The EA team then delivers the designs to a development team within the organization and in some cases this function is outsourced to an external organization.

A.2.14*In situations where it is possible, are several architecture models created in parallel?*

Where possible, the EA team creates several architecture models in parallel. This depends on the situation (Technical model can be created in parallel with a reference model for applications). At a logical level this is possible, however at a physical level it is not as the two are interdependent.

A.2.15 *Is the architecture developed in increments?*

The work to be done on a project is broken down into manageable tasks and a schedule drawn up to indicate the work to be completed in a given period. This allows the architecture to be developed in increments.

A.2.16 *Is technology cut to the minimum needed to solve the problem at hand by building the simplest architecture that can work, or are features added to the architecture models that cater for events that the developers believe could occur?*

Additional features are added to the architecture in addition to the requirements of the EA project to cater for events that the EA team believe can occur in a hope to make the architecture robust and handle future changes.

A.2.17*Do developers use code to evaluate the architecture models when selecting a design alternative or a high-impact infrastructure choice?*

The respondent explained that developers do not use code to evaluate the architecture designs. This will most likely result in project overruns.

A.2.18*Is the team or individual who built a component also responsible for testing it?*

The developers do minor tests on the code themselves to ensure that it delivers the intended results; however the official testing of the code is a function that is outsourced to an independent testing team or company.

A.2.19*While EA activities are underway, is there also an effort to continuously build out system infrastructure, ensuring that foundation for the new features is already in place?*

The EA team makes an effort to continuously build out the system infrastructure by ensuring that part of the solution design is infrastructure design.

A.3 EA Extension and Management Stage

A.3.1 *After the completion of a project, does the EA team have a set of policies which dictate how to extend and make changes to the current architecture?*

After the completion of a project the EA team has a set of policies which dictate how to extend and make changes to the current architecture laid out in the EA team‟s change management policy document.

A.4 EA Governance and Management Stage

A.4.1 *Is management effective in eliminating impediments for the EA team?*

The governance policies established by management are reviewed carefully and changed accordingly to ensure that they do not hinder the project progress of the EA team.

A.4.2*Does management give the EA team some freedom to make project decisions?*

The EA team is given freedom to make minor project decisions as sometimes the theoretical framework does not work in practice. However these changes should follow the processes declared in the change management policy document.

A.5 EA Maturity and Measurement Stage

A.5.1 *Is the current EA maturity evaluated at the beginning if a project?*

At the beginning of the project the EA team evaluates the current IT architecture by doing a 360 assessment. This assessment includes, but is not limited to uncovering what has been done and what needs to be done in order to obtain a baseline.

A.5.2 *Is the value added by the EA project measured after each iteration?*

The respondent declared that the EA team is required to show measurable results after each iteration and present these results at the project status meetings.

B.1Further Comments

B.1.1*Are there any method used by the organization to measure the agility of the EA?*

The respondent‟s organization uses various methods to measure the agility of the created architecture. The organization makes use of the methods that have been made available in the established enterprise architecture frameworks (Zachman, TOGAF, GARTNER, etc.). The

respondent ensures that the developed architecture is in line with the requirements set out in the chosen framework for the EA project.

B.1.2*How did you find the questions?*

The respondent noted that the questions in the survey are easy to understand and relevant to the topic.

B.1.3*Do you have any suggestions that would make the questions better?*

As the questions are high level, and due to the fact that organizations are beginning to adopt more agile development practices, there would most likely be very little difference in the EA practices between the organizations participating in the empirical study. Greater differences could be uncovered in the organizations‟ practices if the questions differentiated between the different levels of architects e.g. domain, solution architects, asked questions that addressed architecture activities at a process level and focused more on architecture deliverables. In addition, the respondent indicated that the project sponsor was the person who provides the budget for and champions the project. They are usually the head of the department who will receive whatever the project is attempting to deliver. The executive body is elected to ensure that the management is involved in the high level decisions of the project. The project sponsor may or not be part of the executive body elected and therefore, the two should be separate within the framework, but are of equal importance.

B.1.4*Are there any other factors you believe should be taken into consideration when developing agile EA?*

The respondent could not think of any other factors that would encourage the development of agile EA.

B.1.5*Are there any comments you would like to make in relation to the framework, research, survey questions?*

The respondent had no additional comments to add.

Organization # 2

A.1 EA Foundation Stage

A.1.1 *Is a needs and cost-benefit analysis done before embarking on an EA Project?*

A needs and cost benefit-analysis is conducted, based on best practices that have been developed within the organization to assess costs and the benefit a project would have on the organization.

A.1.2 *Are top management required to sign off on potential EA projects?*

Top management are required to sign off on potential projects as EA is a key component of the organization. An investment decision has to be made about what the projects are and how they relate to the current architecture in terms of transformation. However, smaller projects can exist in certain environments which do not require signing off, for example a small project to upgrade certain functionality.

A.1.3 *What is your understanding of the term "enterprise architecture" and does everyone involved with the EA project within the organization have a common understanding of what EA is?*

The respondent indicated that EA is a framework that defines the structure and operations of an organization and is used to manage and align an organization‟s IT assets, to its business goals. Achieving a common understanding of EA within the organization is extremely problematic. The definition of EA within the organization varies from person to person and there is a constant challenge to educate and clarify its meaning, value and application.

A.1.4 *What is your understanding of the term "agility" and does everyone involved with the EA project within the organization have a common understanding of what agility is and how it relates to the EA project?*

Agility refers to the speed at which something can change. The faster something can change, the more agile it is. Therefore an agile EA is one that can be changed quickly and easily. The respondent indicated that there is no single view of agility; what it means and how it is interpreted. The confusion is further compounded by the existence of other practices e.g. the design methodology called Agile Development.

A.1.5 *Does management clearly communicate their vision of the selected EA project (s)?*

Management is effective in communicating the vision of the project allowing the EA team to understand why they are embarking on the selected project.

A.1.6 *Does management clearly communicate the goals of the selected EA project (s)?*

Management is ineffective in communicating the goals of the selected project. At the highest level, a vision for the project is communicated, although the translation of that vision into the goals of the selected project(s) is not communicated effectively. This leads to a great deal of sensitivity in the organization, in terms of the EA team being uncomfortable with the project and not knowing what to expect.

A.1.7 *Does management clearly define the expected results of the EA project?*

The respondent indicated that management communicates the expected results of the project. The expected results vary, from project to project. For example in some instances, the expected results of the project may be from a financial perspective or an operating perspective, e.g. cost efficiencies, optimization, streamlining of the various operations, eliminating duplication.

A.1.8 *Does management set up efficient communication structures to be used throughout the project?*

There are no formal communication structures for the EA project established by management. Various avenues and pockets of communication are present within the organization with no common understanding throughout the various areas. In some instances this leads to the EA team communicating with the wrong audience or sending out the wrong message.

A.1.9 *Does management set up robust governance structures that will effectively govern the EA project?*

Governance within the organization only exists on a high level, in terms of how the organization operates. From an EA perspective however, there is a lack of governance, in terms of the projects and how they will affect the organization and in some cases this has an adverse effect on the project‟s success. The respondent indicated that there is a desire to implement a robust governance framework for EA projects. However the mandate for that governance is not yet in place. A mandate is the process in which governance is given within the organization. Without a mandate to exercise governance, people within the organization will question the validity of the decisions being made.

A.2 EA Approach Stage

A.2.1 *Is a project sponsor elected for the project?*

A project sponsor is usually the one who will fund the project and is elected prior to the project‟s initiation.

A.2.2*Is an executive body elected to ensure that top management is involved in high-level decisions of the EA project?*

The respondent indicated that within the organization, there is a COO (Chief Operating Officer) forum consisting of all the COOs of the various business units. The majority of the projects are discussed at this level. There is also a transformation executive committee, which has been established to lead and make decisions regarding any projects that lead to a transformation of the organization and the head of enterprise architecture sits on both these committees.

A.2.3 *Are EA team members motivated and committed to a project at its inception?*

The respondent indicated that in initially, there is usually little commitment from team members on a new project as there is a degree of uncertainty about what the project is intended to achieve. However, the members of the EA team become more motivated and committed to the project in the later stages after they become more comfortable and familiar with the project and are able to see the value they are adding to the organization and thereby take pride in their efforts.

A.2.4 *Is the EA team held directly accountable for the success of the project?*

Within the organization, the Chief Architect is the person held accountable for the success of the project.

A.2.5*Where possible, is there emphasis placed on face to face team collaboration, as opposed to collaboration software tools?*

The EA team meets face to face where possible and also make use of software collaboration tools. However, there is a tendency to not share intellectual capital due to the direct link to job security.

A.2.6*As opposed to one person, is the EA team as a whole responsible for the architecture?*

The respondent pointed out that while the Chief Architect is held accountable for the success of the EA project, the EA team as a whole was responsible for the architecture. The responsibility lies with the domain specific architects to ensure that they deliver what is intended from a project perspective. For example, from a people perspective there would be a People Architect that would play a strong role.

A.2.7 *Does the EA team communicate and collaborate with the customer(s) throughout the life of the EA project?*

The EA team constantly engages with the client(s)/stakeholder(s) throughout the life of the project as this is an integral part of the EA process and ensures service and value is delivered.

A.2.8 *Are there processes put in place to ensure that innovation is not just once-off or near term?*

The chances of innovation happening on a project are very small. Once a project is introduced, very little innovation, if any can be introduced. The organization is not willing to spend money on processes that might introduce innovation once a project is underway. The organization uses traditional methodologies in which the business requirement is articulated and that is then translated into a functional specification. It is not part of the organization culture to have strong innovation processes/practices.

A.2.9*Does the EA team have a clear scope of the project in which to work?*

Initially, the EA team is not provided with a clear scope in which to work; the scope of the project only becomes clearer once the project is underway. There would be a strategic intent for the organization, however what that strategic intent looks like from an architecture perspective is still very unclear. The early stages of the EA project begin with the definition of the operating model. At this stage, it is unclear what this means and what it entails. Only after the EA team goes through the process of articulating their architecture principles and what is to be transformed does the cope become clearer. That in itself is an iterative process over a 12 month period.

A.2.10*Does the EA team reuse existing architecture artifacts?*

The respondent indicated that the EA team reuses existing artifacts, if possible as this reduces the amount of work to be done on the project.

A.2.11*Does the EA team develop situationally specific strategies and practices?*

As projects have differing requirements and operate in different environments, the EA team develops strategies and practices that will provide the best solution to the problem. These strategies are developed within the confines of the methodologies and practices of the organization.

A.2.12*Is the schedule of the project organized in terms of iterations and releases?*

The EA team see the EA as an iterative process and use each iteration to review the work done on the project to ensure it is in line with the project requirements. This allows any deviations from the objectives of the project to be caught early, ensuring that the project will achieve its intended result(s).

A.2.13 *Does the team that designed the system, code the system?*

The team that designed the system is not responsible for coding it. The EA team that designed the system provides coding guidance models. This is mostly done by the solution/software architects. These guidance models describe how to interpret and use the models created, so are not just hard code or system code. So in that respect, they not only define the model, but also define how to write the code for the model.

A.2.14*In situations where it is possible, are several architecture models created in parallel?*

The time frames for the completion of a project require parallel development. The organization does not have the luxury of following a sequential approach. Parallel development however is a problem in itself, as changes in certain places impact other areas of the architecture making things more difficult to manage.

A.2.15 *Is the architecture developed in increments?*

The methodology the EA team uses follows an incremental approach to developing the architecture.

A.2.16 *Is technology cut to the minimum needed to solve the problem at hand by building the simplest architecture that can work, or are features added to the architecture models that cater for events that developers believe could occur?*

The respondent indicated that the EA team find it difficult to have a clear cut deliverable and choose an approach to get that deliverable; therefore, where they believe it is appropriate, additional features are added to the architecture in an attempt to holistically improve it.

A.2.17*Do developers use code to evaluate the architecture models when selecting a design alternative or a high-impact infrastructure choice?*

The respondent indicated that the developers do not use code to evaluate different possible solutions as this would in most cases increase the time spent on a project or iteration, meaning that the project will run over time and over budget.

A.2.18*Is the team or individual who built a component also responsible for testing it?*

The team, or individual who built a component is responsible for testing it. Unit and component testing are conducted by the developer(s) and the architect responsible for designing the artefact. For example, from a business perspective, the business architect will be responsible for testing a business rule/component of the architecture.

A.2.19*While EA activities are underway, is there also an effort to continuously build out system infrastructure, ensuring that foundation for the new features is already in place?*

The respondent indicated that Knowledge managers are part of the EA team and work closely with the Information technology departments to ensure that the required infrastructure is in place to support the newly developed architecture artefact(s).

A.3 EA Extension and Management Stage

A.3.1 *After the completion of a project, does the EA team have a set of policies which dictate how to extend and make changes to the current architecture?*

Change control and governance processes dictate how changes can be made to the current architecture after the completion of a project.

A.4 EA Governance and Management Stage

A.4.1 *Is management effective in eliminating impediments for the EA team?*

The governance structure within the organization is extremely bureaucratic and as a result, in most cases, hinders project progress.

A.4.2*Does management give the EA team some freedom to make project decisions?*

The EA team is able to make recommendations that can influence project decisions. They however do not have the freedom to make project decisions.

A.5 EA Maturity and Measurement Stage

A.5.1 *Is the current EA maturity evaluated at the beginning if a project?*

The EA project begins with a basic evaluation of the current architecture.

A.5.2 *Is the value added by the EA project measured after each iteration?*

The value added by the EA project is measured after each iteration.

B.1Further Comments

B.1.1*Are there any method used by the organization to measure the agility of the EA?*

The organization makes use of the Enterprise Architecture Executive Council (EAEC) scorecard to measure the architecture.

B.1.2*How did you find the questions?*

The respondent didn"t encounter any problems with the line of questioning.

B.1.3*Do you have any suggestions that would make the questions better?*

The respondent was unable to make any suggestions to improve the survey questions.

B.1.4*Are there any other factors you believe should be taken into consideration when developing agile EA?*

No additional factors for the development of agile EA could be uncovered.

B.1.5*Are there any comments you would like to make in relation to the framework, research, survey questions?*

The respondent had no additional comments.

Organization # 3

A.1 EA Foundation Stage

A.1.1 *Is a needs and cost-benefit analysis done before embarking on an EA Project?*

The respondent indicated that this is an important step in justifying the change that the project will bring. It allows the organization to determine how well; or how poorly the planned action will turn out, before embarking on the project as all aspects of a project have financial implications that can be measured either directly or indirectly.

A.1.2 *Are top management required to sign off on potential EA projects?*

Management attend project analysis meetings in which they determine the actions which need to be taken on a project and the intended benefit(s) and sign off on projects that are deemed beneficial to the organization.

A.1.3 *What is your understanding of the term "enterprise architecture" and does everyone involved with the EA project within the organization have a common understanding of what EA is?*

EA is the high-level representation of how business, IT and the people and processes that support them within an organization relate to one another. Not every individual understands the term, "enterprise architecture" and what it means within the organization. This creates a great deal of confusion initially and causes further problems when attempting to meet project objectives.

A.1.4 *What is your understanding of the term "agility" and does everyone involved with the EA project within the organization have a common understanding of what agility is and how it relates to the EA project?*

The respondent indicated that the meaning of agility within the organization was very subjective. To the respondent, agility is a concept that means the ability to handle changing

circumstance and that an agile architecture is one that lends itself to being changed easily, should a need arise.

A.1.5 *Does management clearly communicate their vision of the selected EA project (s)?*

Management communicates the vision of the selected EA project in order to provide strategic direction for the EA team.

A.1.6 *Does management clearly communicate the goals of the selected EA project (s)?*

Management provides a strategic intent (vision) of the selected project and communicates how they hope to achieve that strategic intent (the goals) through the selected project(s).

A.1.7 *Does management clearly define the expected results of the EA project?*

As IT is seen as an enabler of business, all IT projects within the organization should provide business value, or results. The expected results of the project are articulated to the EA team by management.

A.1.8 *Does management set up efficient communication structures to be used throughout the project?*

The communication structures established for the EA project by management also depend on the input of the EA project team. Management is effective in setting up these structures and is flexible in terms of what will work best for the team. There is a project management methodology which is used within the organization which includes status meetings, steering committee meeting, weekly team meetings, etc.

A.1.9 *Does management set up robust governance structures that will effectively govern the EA project?*

Within, the organization, management establishes governance structures that guide and protect the interests of the organization in terms of the money spent on the specific initiatives and other important aspects which relate to project success, in an attempt to ensure that the outputs of the projects are in line with the objectives. From an IT perspective, there are specific governance policies around governing projects and the solutions developed.

A.2 EA Approach Stage

A.2.1 *Is a project sponsor elected for the project?*

A project sponsor whose role is the ownership of the project on the organization‟s behalf is elected for the project prior to it being approved. Within the organization, the project sponsor meets weekly with the Chief Architect.

A.2.2*Is an executive body elected to ensure that top management is involved in high-level decisions of the EA project?*

Within the organization, an executive body comprised of the Chief Architect and executives within the organization would be elected to lead the EA project as these individuals make decisions on a group wide basis. At a lower level, there would also be a committee of architects; solution, information, business architects etc.; who give guidance in terms of the projects and architecture components surrounding the project.

A.2.3 *Are EA team members motivated and committed to a project at its inception?*

The nature of enterprise architects is that they are individuals who want change. They get stuck when things remain the same. Therefore, if a project brings change within the organization, the EA team is motivated to participate.

A.2.4 *Is the EA team held directly accountable for the success of the project?*

Every member within the EA team has a main role and a supporting to play on the EA project within the organization. Their roles are interconnected and as a result, a failure of the EA project is a failure of the team as a whole.

A.2.5*Where possible, is there emphasis placed on face to face team collaboration, as opposed to collaboration software tools?*

The EA team prefer face to face collaboration as this builds a bond between the team members, creating a better team dynamic. The respondent indicated that following a team meeting, members of the EA team are amongst other benefits; more comfortable with their role on the project and are able to better understand the tasks for the next iteration to which they have been allocated. The frequency and other aspects of these face to face collaborations is controlled by the project management methodology which the EA team is using. The knowledge manager of the EA team runs a knowledge sharing and collaboration process

throughout the life of the project. He/she handles the repository of artifacts and turns them into something that can be used for best practices and industry standards and runs communities of practice sessions where people can communicate and share knowledge.

A.2.6*As opposed to one person, is the EA team as a whole responsible for the architecture?*

Each member of the team has domain specific responsibilities which together form the complete architecture.

A.2.7 *Does the EA team communicate and collaborate with the customer(s) throughout the life of the EA project?*

Engaging with the stakeholder(s) throughout the project lifecycle ensures the EA team that they are meeting the demands of the customer(s) and that their efforts are in line with the project requirements. Meeting with the customer(s) at certain intervals (usually after each iteration) also allows the EA team to present any benefits of the project iteration which provides reassurance to the stakeholders that they made a worthy investment.

A.2.8 *Are there processes put in place to ensure that innovation is not just once-off or near term?*

During the life of a project, the requirements of the project may change, or more added. The need for changes is identified not only by the EA team but also by the stakeholder(s). Any changes to the requirements have to be justified by the EA team and sanctioned by the stakeholder(s) as these changes will inevitably affect the project timeline and budget.

A.2.9*Does the EA team have a clear scope of the project in which to work?*

The high-level strategic view of the project is broken down into more detail in order to understand the impact of the project and provide the scope of the work to be done. Scoping of the project is an important preliminary step of the organization, as there needs to be a timeline and budget drawn up for the project which will depend on the work that needs to be done.

A.2.10*Does the EA team reuse existing architecture artifacts?*

Where possible, the EA team will avoid “re-inventing the wheel.” Reuse of existing artifacts is a large part of the EA team development framework and brings with it many benefits.

A.2.11*Does the EA team develop situationally specific strategies and practices?*

The EA team develop a project strategy and practices that will best suit the objectives of the project.

A.2.12*Is the schedule of the project organized in terms of iterations and releases?*

The respondent indicated that the project is organized in terms of iterations and releases in order to show the stakeholder(s) a return on their investment as soon as possible. There are different time frames for different value concepts; some are realized sooner than others. For example, there is value added in terms of having a defined architecture thereby accelerating change by allowing management to have a view of the organization‟s current structure and capabilities. There could also be value added from a cost of ownership perspective where the reuse of existing artifacts and reduction in duplication in the organization, leads to a reduction in cost of ownership.

A.2.13 *Does the team that designed the system, code the system?*

The EA team that designed the system gives the completed designs to the development team to be coded. The EA team provides clarity where it is needed and supervises the development of the system, ensuring that the developed system is in line with the designs, but does not code the system themselves.

A.2.14*In situations where it is possible, are several architecture models created in parallel?*

Creating several models in parallel is a necessary practice of the EA team as creating the models one after the other will resulted in greater costs and a longer time frame for the completion of the project.

A.2.15 *Is the architecture developed in increments?*

The EA project is broken into manageable parts and each part of the project into tasks for the members of the EA team. This allows the architecture to be developed incrementally and the EA team to show the value added by the project after each iteration.

A.2.16 *Is technology cut to the minimum needed to solve the problem at hand by building the simplest architecture that can work, or are features added to the architecture models that cater for events that developers believe could occur?*

Architects as thought leaders will suggest and motivate the need for additional features. This is in an attempt to improve the architecture after the various iterations, by bringing in features that will assist future development and assist with acceleration.

A.2.17*Do developers use code to evaluate the architecture models when selecting a design alternative or a high-impact infrastructure choice?*

The developers do not use code to evaluate their designs. Instead they are required to make decisions on project alternatives based on the requirements of the project, their previous experience, research and best practice in the organization; and choose what they deem the best solution for the task at hand.

A.2.18*Is the team or individual who built a component also responsible for testing it?*

Various testing; e.g. unit, component, system, user, acceptance is conducted by the team or individual responsible for building the component. The different architects may be involved in different types of testing. For example, from a systems perspective where the architect(s) and developers developed the system; the technical architect would be involved in testing. Where there is end user testing, the business architect would be involved to ensure that the business requirements were actually met through user acceptance testing.

A.2.19*While EA activities are underway, is there also an effort to continuously build out system infrastructure, ensuring that foundation for the new features is already in place?*

The infrastructure architects are responsible for developing an infrastructure design that will support the new architecture features. This infrastructure constantly evolves throughout the life of the project and is updated where needed to support the feature(s) are added to the architecture.

A.3 EA Extension and Management Stage

A.3.1 *After the completion of a project, does the EA team have a set of policies which dictate how to extend and make changes to the current architecture?*

Changes to the developed architecture are made in accordance with policies developed within the organization based on research and best practices. These changes are made at the request or suggestion of the architect(s) and stakeholder(s).

A.4 EA Governance and Management Stage

A.4.1 *Is management effective in eliminating impediments for the EA team?*

The inclusion of the Chief Architect along with business leaders in the executive body responsible for governing the EA project ensures that the EA team has a means to voice their project concerns. These concerns are taken into account and changes made to the current governance structure or policies in order to enable the EA team to better achieve their project goals.

A.4.2*Does management give the EA team some freedom to make project decisions?*

The EA team is able to make minor project decisions. However major project decisions have to be sanctioned by the stakeholder(s).

A.5 EA Maturity and Measurement Stage

A.5.1 *Is the current EA maturity evaluated at the beginning if a project?*

The initial phases of the EA project involve evaluating the current architecture, in order to be aware of what is currently in place, can be reused, needs to be removed; and what needs to be done to achieve the goals of the project.

A.5.2 *Is the value added by the EA project measured after each iteration?*

Following each project iteration, the EA team assesses the value added and presents the new features if any to the stakeholder(s). In some cases, value is realised after two or more iterations.

B.1Further Comments

B.1.1*Are there any method used by the organization to measure the agility of the EA?*

The organization does not have any methods for measuring the agility of EA.

B.1.2*How did you find the questions?*

The respondent commented that the questions were clear and detailed.

B.1.3*Do you have any suggestions that would make the questions better?*

The respondent indicated that when asking questions about management, the questions could differentiate as to whether it was business or IT management that was being referred to.

B.1.4*Are there any other factors you believe should be taken into consideration when developing agile EA?*

No additional factors were discovered.

B.1.5*Are there any comments you would like to make in relation to the framework, research, survey questions?*

The respondent had nothing further to add.

Organization # 4

A.1 EA Foundation Stage

A.1.1 *Is a needs and cost-benefit analysis done before embarking on an EA Project?*

A needs and benefit analysis is conducted prior to embarking on the EA project. The need analysis is for the EA project is tied back to functional business results. For example, if there is a need for governance on projects, this need can be tied back to business in terms of better governance leading to better project investment, budget and scope management; and an increase in project success, etc.

A.1.2 *Are top management required to sign off on potential EA projects?*

The projects to be signed off on by management will usually depend on the size of the project and whether the project budget requirements fall within the allocated IT project budget. Larger projects will require additional budget beyond the scope of the allocated budget for IT

projects. This will require approval from top management. Smaller projects, which fall within the budget for IT projects do not require signing off by top management.

A.1.3 *What is your understanding of the term "enterprise architecture" and does everyone involved with the EA project within the organization have a common understanding of what EA is?*

Enterprise architecture is a whole set of artifacts that will be delivered against certain business challenges and involves designing solutions that include all areas of the business, risk and governance, etc. Not everyone within the organization has a common understanding of what EA is, or why it is needed.

A.1.4 *What is your understanding of the term "agility" and does everyone involved with the EA project within the organization have a common understanding of what agility is and how it relates to the EA project?*

The respondent indicated that the term agility referred to something that was easily changed. However, the respondent‟s definition of agility is not shared by everyone within the organization and the concept of agility is one that the organization has not considered within the context of EA.

A.1.5 *Does management clearly communicate their vision of the selected EA project (s)?*

Management provides a clear vision in order to enable the EA team to be able to have a clear understanding of what the project is intended to achieve.

A.1.6 *Does management clearly communicate the goals of the selected EA project (s)?*

Management communicates the goals of the selected project, which allows the EA team to understand the work that needs to be done on the project.

A.1.7 *Does management clearly define the expected results of the EA project?*

The respondent indicated that although it is an important step for an organization‟s EA to mature, management is not always clear in defining the expected results of the EA project(s).

A.1.8 *Does management set up efficient communication structures to be used throughout the project?*

The program manager of the EA project is responsible for drafting the communication policies and standards for the EA project and communicating these to the EA team.

A.1.9 *Does management set up robust governance structures that will effectively govern the EA project?*

Within the organization there is a particular methodology used and that methodology includes governance policies which have to be adhered to.

A.2 EA Approach Stage

A.2.1 *Is a project sponsor elected for the project?*

It is not possible to have a project without a project sponsor within the organization.

A.2.2*Is an executive body elected to ensure that top management is involved in high-level decisions of the EA project?*

Every project in the organization has a steering committee consisting of executive managers whose responsibility is to drive project success.

A.2.3 *Are EA team members motivated and committed to a project at its inception?*

Not all of the EA team members are motivated and committed to a project at its inception. The level of commitment and motivation to the project depends on the type of the project and how it meets their own requirements.

A.2.4 *Is the EA team held directly accountable for the success of the project?*

The EA team is held directly accountable for the success of the project.

A.2.5*Where possible, is there emphasis placed on face to face team collaboration, as opposed to collaboration software tools?*

Where possible, the EA team favours face to face collaboration. However, in some instances this is not possible as the organization makes use of virtual teams where members of the team may be situated far away from others. In these circumstances, the EA team will use software collaboration tools and attend web conferences to communicate with one another.

A.2.6*As opposed to one person, is the EA team as a whole responsible for the architecture?*

The EA team as a whole is responsible for the architecture. The overall responsibility is with the Chief Architect. He/she in turn holds the different domain architects responsible for their different areas of the architecture.

A.2.7 *Does the EA team communicate and collaborate with the customer(s) throughout the life of the EA project?*

Communication with the customer(s) throughout the project life is important for project success in order to ensure that the requirements of the project are being made, value of the project to the customer is shown and any changing requirements of the project are discussed.

A.2.8 *Are there processes put in place to ensure that innovation is not just once-off or near term?*

No processes are included to introduce innovation on a project. The EA team fear that the inclusion of such processes would lead to scope creep.

A.2.9*Does the EA team have a clear scope of the project in which to work?*

The EA team creates a clear scope of the work to be done on a project.

A.2.10*Does the EA team reuse existing architecture artifacts?*

The EA team will reuse existing artifacts where possible. However, the existence of artifacts which the EA team can reuse will also depend on how mature the organization is and the objective(s) of the selected project(s).

A.2.11*Does the EA team develop situationally specific strategies and practices?*

The EA team develops situationally specific strategies and practices using a strict method which enables the definition of the strategies and practices and what products need to be worked through.

A.2.12*Is the schedule of the project organized in terms of iterations and releases?*

The Project is broken up into phases and sub-phases which are linked to iterations. This is especially true in the development of new products. The number of iterations for a project will depend on the nature of the work to be done. For example, developing a new infrastructure

requires less iterations and releases than developing code on top of the infrastructure for risk and governance technologies.

A.2.13 *Does the team that designed the system, code the system?*

The organization makes use of a design team on projects and a separate coding team.

A.2.14*In situations where it is possible, are several architecture models created in parallel?*

The EA team creates architecture models in parallel where there are different streams of development that can be followed in parallel. The EA team also cross references the models to ensure that they make sense in relation to the overall project.

A.2.15 *Is the architecture developed in increments?*

The EA project has a macro design which is then translated to a micro design which will represent the increments for the whole project.

A.2.16 *Is technology cut to the minimum needed to solve the problem at hand by building the simplest architecture that can work, or are features added to the architecture models that cater for events that developers believe could occur?*

Technology is kept to the minimum needed to solve the problem at hand as it is important to not over engineer a solution. The respondent indicated that the dangers of over engineering a solution are under delivering the solution at a cost that is unacceptable and the Return on Investment (ROI) becomes more difficult to reach. Additional features may only be added to the architecture if these additions are still in line with project requirements.

A.2.17*Do developers use code to evaluate the architecture models when selecting a design alternative or a high-impact infrastructure choice?*

Developers do not use code to decide on project alternatives as this will delay the release of the artefact to be developed.

A.2.18*Is the team or individual who built a component also responsible for testing it?*

The building and testing of the architecture components is conducted by two separate teams. The organization views this as a good approach. If there is a fix, it will have to go back to the build team to build it and then go back to the testing team for testing.

A.2.19*While EA activities are underway, is there also an effort to continuously build out system infrastructure, ensuring that foundation for the new features is already in place?*

The IT infrastructure is grown as EA project activities are underway to support the solution that is being designed to be put down.

A.3 EA Extension and Management Stage

A.3.1 *After the completion of a project, does the EA team have a set of policies which dictate how to extend and make changes to the current architecture?*

The organization has a set methodology in place for change management; and within change management, the EA team must ensure that the artifacts are adapted according to these policies.

A.4 EA Governance and Management Stage

A.4.1 *Is management effective in eliminating impediments for the EA team?*

The respondent indicated that the problems encountered by the EA team during the project lifecycle are addressed promptly and effectively as the organization elects a program manager whose responsibilities include eliminating impediments for the EA team.

A.4.2*Does management give the EA team some freedom to make project decisions?*

The EA team is given some creative freedom to make project decisions.

A.5 EA Maturity and Measurement Stage

A.5.1 *Is the current EA maturity evaluated at the beginning if a project?*

In instances where the organization has formal EA and that EA is mature the EA team will evaluate the current EA maturity mature. However, initially where there was no formal EA, measurement only occurs at the end of the project to assess what results had been achieved.

A.5.2 *Is the value added by the EA project measured after each iteration?*

The value added by the project is not measured after each iteration as depending on the nature of the project, each individual iteration may or may not necessarily produce value. The value added by the project is shown as it arises after any number of iterations.

B.1Further Comments

B.1.1*Are there any method used by the organization to measure the agility of the EA?*

The organization does not have any methods for measuring the agility of EA.

B.1.2*How did you find the questions?*

The respondent believed the questions were in line with the topic and easy to understand**.**

B.1.3*Do you have any suggestions that would make the questions better?*

No suggestions were made that would improve the survey questions.

B.1.4*Are there any other factors you believe should be taken into consideration when developing agile EA?*

The respondent had no additional factors to add to the framework.

B.1.5 *Are there any comments you would like to make in relation to the framework, research, survey questions?*

There are a number of architecture methods and standards available, e.g. TOGAF, etc. The respondent indicated that it would be interesting to include which of the EA standards the framework follows. In addition, It is important to ensure that the business benefits of an EA project are communicated to the organization, even if the EA project is being conducted to improve an IT function; for example, eliminating redundancy in IT operations, etc. The respondent believes that it is a mistake when certain projects are driven around IT benefits. At the end of the day, business funds these projects and the aim of the projects should be to get business results. Currently, it is difficult to find an organization driving an EA project for the sole benefit of the improvement of IT functions. The project is always linked to a business solution. The respondent indicated that EA projects should be driven through a business solution going forward. For example, there may be a new business project which the EA team may see as an opportunity for improving the current architecture. The EA team would adhere to their EA principles in relation to the project in order to bring the improvements to the architecture as part of the overall project. Very few organizations drive specific IT improvement projects anymore. They have to be tied to a business challenge.

Appendix C – Permission Email

Dear Sir/Ma‟am

The Department of Information Systems at Rhodes University is investigating Enterprise Architecture as an area of research under the leadership of Mr. John McNeill.

My name is Bokang Mthupha and I am a Masters student in the Information Systems Department. My area of interest is in how the agility of enterprise architecture can be measured.

The aim of my research is to develop, “A Framework for the Development and Measurement of Agile Enterprise Architecture”.

In support of the development of the framework, the research will:

- Develop a comprehensive definition for Enterprise Architecture as well as discover how it is currently practiced.
- Develop a comprehensive definition for agility and discuss why it is emerging as a critical topic.
- Investigate agile architecture in order to develop a comprehensive definition and discuss how agility fits within the context of EA and the best practices in agile architecture development.
- Investigate suitable measurement techniques that can be used to assess the level of agility of EA.

The framework will comprise the criteria for the development of agile EA and the associated measurement technique.

It is intended that the framework can be used by organisations wishing to embark on agile EA development.

I should like your organisation to participate in the research by way of completion of a questionnaire in respect of EA and agility. I should be most grateful if you or a representative of your organisation would be willing to receive the questionnaire and participate in the survey. The identity of your institution will be treated with complete confidentiality. Should participants wish, the results of the research will be provided to them.We look to you for

guidance in identifying [someone] at your institute that would be suitable to interview (at a time and date that suits them).

Thank you for your time and I hope that you will find our request favourable. If you have questions or wish to verify the research, please feel free to contact me.

Yours sincerely,

Bokang Mthupha (MCOM) Cell: 0837656627

Information Systems

Rhodes University www.is.ru.ac.za

Grahamstown, 6140

South Africa

Appendix D – Permission Letter

RHODES UNIVERSITY

Grahamstown•6140 •South Africa

DEPARTMENT OF INFORMATION SYSTEMS

Tel: [+27] 046 603 8473

Fax: [+27] 046 636 1915

E-mail: g05m2106@campus.ru.ac.za

[Date]

[Head of IT/Systems]

[Address of Organization]

To Whom It May Concern

Re: Permission to conduct research at your institution

The Department of Information Systems at Rhodes University is investigating Enterprise Architecture as an area of research under the leadership of Mr. John McNeill.

My name is Bokang Mthupha and I am a Masters student in the Information Systems Department. My area of interest is in how the agility of enterprise architecture can be measured.

The aim of my research is to develop, "A Framework for the Development and Measurement of Agile Enterprise Architecture".

In support of the development of the framework, the research will:

- Develop a comprehensive definition for Enterprise Architecture as well as discover how it is currently practiced.
- Develop a comprehensive definition for agility and discuss why it is emerging as a critical topic.
- Investigate agile architecture in order to develop a comprehensive definition and discuss how agility fits within the context of EA and the best practices in agile architecture development.
- Investigate suitable measurement techniques that can be used to assess the level of agility of EA.

The framework will comprise the criteria for the development of agile EA and the associated measurement technique.

It is intended that the framework can be used by organisations wishing to embark on agile EA development.

I should very much like your organisation to participate in the research by way of completion of a questionnaire in respect of EA and agility. I should be most grateful if you or a representative of your organisation would be willing to receive the questionnaire and participate in the survey. The identity of your institution will be treated with complete confidentiality. Should participants wish, the results of the research will be provided to them. We look to you for guidance in identifying [someone] at your institute that would be suitable to interview (at a time and date that suites them).

Thank you for your time and I hope that you will find our request favourable. If you have questions or wish to verify the research, please feel free to contact me.

Yours sincerely,

Bokang Mthupha (MCOM) Cell: 0837656627

Information Systems

Rhodes University www.is.ru.ac.za

Grahamstown, 6140

South Africa

Appendix E – Consent Form

RHODES UNIVERSITY

CONSENT FORM

Department of Information Systems

Project Title: A Framework for the Development and Measurement of Agile Enterprise Architecture

Researcher"s names: Bokang Mthupha, John McNeill

- I have received information about this research project.
- I understand the purpose of the research project and my involvement in it.
- I understand that I may withdraw from the research project at any stage.
- I understand that participation in this study is done on a voluntary basis.
- I understand that while information gained during the study may be published, I will not be identified and my personal results will remain confidential.
- I understand that I will receive no payment for participating in this study.

Name: .....

Signed ..... **Date** .....

I have provided information about the research and believe that participant understands what is involved.

Researcher"s signature and Date .....