ACCIDENT VERSUS ESSENCE: INVESTIGATING THE RELATIONSHIP AMONG INFORMATION SYSTEMS DEVELOPMENT AND REQUIREMENTS CAPABILITIES AND PERCEPTIONS OF ENTERPRISE ARCHITECTURE

Brian R. Salmans, B.B.A., M.B.A., M.S.

Dissertation Prepared for the Degree of

DOCTOR OF PHILOSOPHY

UNIVERSITY OF NORTH TEXAS

August 2009

APPROVED:

Leon A. Kappelman, Major Professor
Stephen M. Swartz, Minor Professor
Robert J. Pavur, Committee Member
Anna Sidorova, Committee Member
Mary C. Jones, Chair of the Department of Information Technology and Decision Sciences
O. Finley Graves, Dean of the College of Business Administration
Michael Monticino, Dean of the Robert B. Toulouse School of Graduate Studies

Salmans, Brian R. Accident versus Essence: Investigating the Relationship Among Information Systems Development and Requirements Capabilities and Perceptions of Enterprise Architecture. Doctor of Philosophy (Information Systems), August 2009, 194 pp., 46 tables, 8 figures, 194 titles.

Information systems (IS) are indelibly linked to the global economy and are indispensable to society and organizations. Despite the decisive function of IS in organizations today, IS development problems continue to plague organizations.

The failure to get the system requirements right is considered to be one of the primary, if not the most significant, reasons for this high IS failure rate. Getting requirements right is most notably identified with Frederick Brooks' contention that requirements are the essence of what IT professionals do, all the rest being accidents or risk management. However, enterprise architecture (EA) may also provide the discipline to bridge the gap between effective requirements, organizational objectives, and the actual IS implementations.

The intent of this research is to examine the relationship between IS development capabilities and requirements analysis and design capabilities within the context of enterprise architecture. To accomplish this, a survey of IT professionals within the Society for Information Management (SIM) was conducted.

Results indicate support for the hypothesized relationship between IS development and requirements capabilities. The hypothesized relationships with the organizational demographics were not supported nor was the hypothesized positive relationship between requirements capabilities and EA perceptions. However, the nature of the relationship of requirements and EA provided important insight into the

relationship leading to several explanations as to its meaning and contributions to research and practice.

This research contributes to IS development knowledge by providing evidence of the essential role of requirements in IS development capabilities and in IS development maturity. Furthermore, contributions to the nascent field of EA research and practice include key insight into EA maturity, EA implementation success, and the role of IT professionals in EA teams. Moreover, these results provide a template and research plan of action to pursue further EA research in exploring EA maturity models and critical success factors, and the state of practice of EA in organizations.

ACKNOWLEDGEMENTS

First of all, I would like to thank Dr. Leon Kappelman for his outstanding support as the chair of this dissertation. I appreciate the unique opportunity he offered me in getting involved with enterprise architecture and the SIMEAWG. I owe the successful completion of my PhD program to him and I consider myself most fortunate to be able to work with him and have his significant contributions on my dissertation and academic experience while at the University of North Texas. Additionally, I would like to thank Dr. Robert Pavur, Dr. Anna Sidorova, and Dr. Steve Swartz for their effort and assistance as dissertation committee members. They all provided important guidance not only for this dissertation, but also for other research projects and advice for my academic endeavors, both while in the United States Air Force and for my post-USAF career. Thank you for taking the time for this and for your contributions. I am also thankful for the advice and support of Dr. John Windsor, Dr. Sherry Ryan, and Dr. Victor Prybutok who served as ITDS academic advisors during my “tour” at UNT. My experience has been very rewarding and successful due to their advice.

I also want to express gratitude to my parents, Lee and Karen for their guidance and support for me and my family before and during this PhD program. Finally, I would like to thank my wife, Kristin for her support and love in our marriage that enabled the successful completion of this PhD program. Thank you for all that you do and my academic accomplishments are as much hers as they are mine. I am glad our son Jackson has such a great role model, and hopefully, should he choose to pursue a PhD, will have the same amazing opportunities, individuals, and support in his academic experience as I have had.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The failure rate of information systems (IS) development projects has continued to plague organizations for decades. Researchers cite statistics such as 80% of IS projects being over budget and 25% are cancelled outright and of those that are not cancelled, 75% are operational failures, not operating as specified or simply not being used (Schmidt, Lyytinen, Keil, & Cule, 2001). The often-cited *Chaos Chronicles* report of the Standish Group International reports only a 34% IT project success rate in 13,522 projects at Fortune 500 firms (Nelson, 2005). A discouraging point is the litany of IS development project ills, whether from cost or time overruns, functionality issues, or other such aspects that begins the introduction sections of many research articles on IS development, whether from the 1970s, 1980s, 1990s, or in the twenty-first century (Duggan & Reichgelt, 2006). Many of these failures are the result of one or more of the following situations: total system failure, cost and/or time overruns, and implementation with reduced features or functions not meeting user requirements (Nelson, 2005; Yeo, 2002). A conservative estimate puts the annual cost of failed IS projects in the US from $60 to $70 billion (Charette, 2005). The consequences of these failures can result in decreased revenues, damage to corporate brand or reputation, exposure to legal liabilities, and a decrease in productivity (Baltzan & Phillips, 2007). Moreover, besides impact to financial bottom lines, the failures of IS development can bring significant organizational consequences, even potentially leading to ruin (Goulielmos, 2003; Xia & Lee, 2005).

This high failure rate seems to be at odds with the continually increasing importance of information systems to organizations. Indeed, the survivability of organizations in the extremely competitive business environment of today depends at an increasing rate on the organization’s software (Boehm, 2006). More importantly, software development can be considered to be critically important for society as well (Fitzgerald & O’Kane, 1999). Cattaneo, Fuggetta, and Sciuto (2001) state: “software has become the most critical component in any modern product and service” (p. 3). The organization’s software will be relied upon for competitive differentiation (whether it is unique capabilities, security, or privacy for example) and flexibility to respond to the competitive changes of the business environment. This is of continued importance with the steady increases in the complexity of software and large system architectures, and larger, extended organizations or enterprises. In the academic area as well, IS development is crucial, being considered the core of the information systems discipline (Hirschheim & Klein, 2003; Petkov & Petkova, 2008).

One response to IS failures has been to implement a software process improvement (SPI) program such as the Software Engineering Institute’s (SEI) capability maturity model (CMM) (or the more current capability maturity model integration (CMMI)). In order to improve information systems development processes, practices and quality, the Software Engineering Institute developed the CMM in the late 1980s. The premise of the CMM was to better manage the software process (Paulk, Curtis, Chrissis, & Weber, 1993) and to address increasing quality and productivity issues in IS development (Duggan, 2004). Generally, a capability maturity model is any model which contains essential elements of effective processes and defines an improvement plan evolving to more mature processes, characterized by an improvement in quality and effectiveness (Chrissis, Konrad, & Shrum, 2003). An organization's process maturity is of paramount importance to managing capabilities to meet vital requirements across functional areas within an organization. The SEI CMM is the most widely known maturity model in the IT field (Rogoway, 1998), laying a foundation for the development of maturity models in a variety of business applications. Organizations typically embark on a software process improvement plan either for compliance or to address problems with their IS development programs such as poor software quality, functionality issues, or time and budget overruns (Layman, 2005). However, there is a gap in accounting for the vital role of requirements within the SPI programs.

Another contribution to successful information systems is also one that is regarded as being neglected by the CMM, that of requirements. As Sommerville and Ransom (2005) put it, requirements are mostly outside the scope of process improvement models. This is surprising given the criticality of requirements to IS development. Requirements capabilities have long been recognized as essential aspects of the management of information systems, IS planning, and IS development (Nguyen & Swatman, 2003). Additionally, some research such as Brooks (1995) and Jones (2008) indicate the failure to get requirements right is the most critical problem with IS development.

Of the many different reasons for these IS development problems, the failure to get the system requirements right is considered to be one of the primary if not the most significant, reasons for this high IS failure rate (Duggan & Thachenkary, 2003) as well as the most difficult part of IS development (Brooks, 1995). The failure of IS programs to get the requirements right has an equally lengthy history as that of IS development failures (Brooks, 1995). In his book *The Mythical Man-Month*, Frederick Brooks (1995) discusses the essential importance of getting the requirements right in the software development process and the challenges the IS field has in getting those requirements right. Brooks (1995) hypothesizes the difficulties of IS development can be framed with difficulties of essence and difficulties of accidents. The requirements are the essence and such things addressing software development such as programming languages, tools, development methods, and processes are the accidents (Berry, 2004). Generally, improvements in IS development have come from resolving the accidental difficulties, those serving as barriers making software tasks difficult. But Brooks (1995) maintains resolving the essential difficulties, difficulties inherent in the general nature of software, should be the focus because it would bring about order-of-magnitude improvements in IS development. These inherent difficulties of IS development derive from software's essential properties of complexity, conformity, changeability, and invisibility (Brooks, 1995; McConnell, 1999; Xia & Lee, 2005) and continue to be relevant today (Duggan & Reichgelt, 2006).

Brooks (1995) states: "The hardest single part of building a software system is deciding precisely what to build. No other part of the conceptual work is as difficult as establishing the detailed technical requirements…. No other part of the work so cripples the system if done wrong. No other part is more difficult to rectify later." (p. 199). This difficulty with getting the requirements right was echoed years later by Cheng and Atlee (2007). They maintain the requirements process is difficult because requirements are in the problem space, not in the solution space like other software artifacts.

The critical relationship between IS requirements and IS development has only increased in importance as IS investments continue to account for significant slices of organizational budgets and as software's role in supporting strategic business goals is becoming more important (Harter & Slaughter, 2000). These IS investments are increasingly being reviewed to ensure alignment with and contribution to organizational goals and strategies. Additionally, the trend of more outsourcing, strategic alliances, and demands for security and privacy safeguards within information systems creates an atmosphere demanding more in depth knowledge of not only IS requirements but also an in depth knowledge of the organization itself (McNurlin & Sprague, 2006).

An inability to get the requirements right seems to be a continuing problem for the IS field and is the root cause for most IS project failures (Baltzan & Phillips, 2007). IS cannot get alignment right if we cannot get the requirements right, since the organizational goals we need to align with are part of those requirements. Brooks (1995) contends architectural unity or conceptual integrity — that of a system consisting of one set of design ideas — is the most important consideration in system design. To attain conceptual integrity, Brooks (1995) maintains, the architectural process must be separate from that of implementation.

This leads to another potential critical component of the relationship between IS development and requirements. Enterprise architecture (EA) may provide the discipline to bridge the gap between effective requirements, organizational objectives, and the actual IS implementation. EA facilitates the alignment between the business and IS domains, a possible answer to achieving conceptual integrity and getting the requirements right and dealing with the requirements challenges highlighted by Brooks

(1995). To use Brooks' lexicon, EA may be the effort that can best attack the essential, not the accidental, difficulties in software development regarding IS requirements. EA facilitates the attainment of a comprehensive view of enterprise-wide requirements (Kappelman, McGinnis, Pettit, Salmans, & Sidorova, 2008). EA ensures congruency between organizational strategies, processes, and IS requirements, hence forming an inclusive IS strategy (Young, 2001). Thus a central goal of EA is the alignment of IS requirements to the goals and objectives of an organization.

Enterprise architecture provides a formalized way to capture and document an organization's present and future desired state and thus contributes to the management of change to the desired state. As an emerging discipline there is still not a consensus on a single authoritative definition of EA. According to one of the "founders" of EA, it is a logical construct defining and controlling interfaces and integration of all the parts of an organization or system and it establishes "order and control in the investment of information systems resources" (Zachman, 1987, p. 454). Another source defines EA as the "analysis and documentation of an enterprise in its current and future states from an integrated strategy, business, and technology perspective" (Bernard, 2005, p. 31). Finally, EA is defined by Ross, Weill, and Robertson (2006) as "the organizing logic for core business processes and IT infrastructure reflecting the standardization and integration of a company's operating model" (p. viii).

The discipline of EA has increasingly been applied in organizations in order to facilitate a disciplined approach to policy, planning, decision-making, and resource development. However, even with the increasing strategic importance of EA (Ross, Weill, & Robertson, 2006; Schelp & Stutz, 2007), there has not been enough basic research into EA which adversely affects the maturity of EA (Langenberg & Wegmann, 2004).

Quantifying the state of IS development capabilities and EA processes and benefits is critical to understanding the impact and relationship between the organization and organizational requirements and to properly manage and improve these efforts. An effective tool to accomplish this is the use of maturity models. Maturity models, derived from stage theories, are based on the belief that systems, products, organizations, and most other entities go through distinct stages over time (Nolan, 1973). The maturity of the IS development and EA programs in an organization not only gives leadership an indication of where and how their programs stand, but can also indicate a proper path for where they want their programs and their organization to go. Measuring the maturity of IS development and EA can allow leadership to make needed course corrections and help to ensure the viability and success of their programs.

Theoretical Background

Process maturity is an important element within this research. Humphrey (1988) defines a process "as a sequence of tasks that, when properly performed, produces the desired result" (p. 74). Capability maturity models provide a basis for the control of IT and organizational processes and practices by identifying strengths, areas for improvement, and subsequent activities to effect improvement in the processes and practices. The standards defined by the maturity models establish levels of maturity and can be used in managing the IS or desired organizational improvements (Steghuis, Daneva, & van Eck, 2005). In developing a software process maturity framework, Humphrey (1988) identified the characteristics of an effective process: 1) predictability and statistical control and 2) measurability. From Humphrey's work, the SEI developed their CMM. It is probably the best known maturity model in the IT arena and is used as a foundation in developing many other maturity models in a variety of applications and approaches.

Prior research investigating the relationship between process maturity level and organizational outcomes has shown positive relationships between process maturity and software quality (Jones, 2002, 2008; Harter & Slaughter, 2000), product quality (Bohn, 1995), and other increased quality outputs (Fenton & Neil, 1999; Zahran, 1998). But the relationship between software process maturity level (as measured by the SEI CMM) and EA maturity level has not been researched.

Numerous research studies have formed the theoretical basis for maturity models. The notion of S curves, where the relationship between the effort of projects or processes and the performance gains from those particular projects or processes generally results in a sinusoidal line provided the foundation for maturity models (Kuznets, 1979). Greiner (1972) advocated a historical aspect of maturity models by emphasizing the role of an organization's history in the future growth and success of a company, rather than a singular focus on outside forces. However, Nolan's (1973, 1979) stages theory is one of the most critical contributions to maturity model theory. Nolan's (1973, 1979) stages theory or maturity model is based on the plot of IS budgets over time based on case studies of a number of companies. The resulting S curve identified several specific points or stages of growth patterns within IS. Originally there were four stages which over time evolved to a maturity model of six stages (Nolan, 1979). In a similar manner as Greiner (1972), Nolan's stated purpose for his stage theory is to assist IS managers through periods of crises.

EA theory is simultaneously emerging with the growth of practitioners' knowledge and experience (Luo, 2006), but gaps are evident regarding inconsistencies with EA concepts, lexicon, goals, benefits, and techniques; knowledge gaps in EA literature; and a lack of consensus and collaboration in the EA community (Luo, 2006).

Problem Statement

A gap remains between theory and practice in IS system failures (Yeo, 2002). Even though there has been much research on the role (and deficiencies) of requirements in IS development and management and the outcomes of software process assessment and improvement programs such as the SEI CMM, there is little research regarding the nature of the relationship between IS development and requirements analysis capabilities. There is substantial research describing and indicating positive outcomes from adopting software process improvement methods such as the SEI CMM(I) (Glass, 1999; Harter & Slaughter, 2000; Jones, 2002, 2003, & 2008; Krishnan & Kellner, 1999). But, as described by Damian and Chisan (2006) in research into the relationships between requirements engineering and other IS development process, "Requirements engineering is an important component of effective software engineering, yet more research is needed to demonstrate the benefits…While the existing literature suggests effective requirements engineering can lead to improvements…there is little evidence to support this" (p. 433). Viewing this problem within the context of EA offers a possible framework for this relationship which addresses the need for alignment of the organizational and IT objectives involved with IS development.

Research Questions

The first research question is what is the relationship between information systems development capabilities and requirements capabilities?

Additionally, the size of organization and IT budget are considered, since many prior research studies (Brodman & Johnson, 1994; Lumsden, 2007, Richardson & von Wangenheim, 2007; Strigel, 2007) indicate some level of influence of these variables on the constructs under consideration.

The second research question is what is the relationship between information systems development and requirements capabilities and perceptions of enterprise architecture?

Research Design

From these research questions, the following relationships were hypothesized:

M1H1: Higher IS development capabilities are associated with higher requirements analysis and design capabilities.

M1H2: Larger organizations are associated with higher IS development capabilities when compared to smaller organizations.

M1H3: Larger organizations are associated with higher requirements analysis and design capabilities when compared to smaller organizations.

M1H4: Organizations in industries where the IT budget tends to be higher (e.g. financial and telecom firms) are associated with higher IS development capabilities.

M1H5: Organizations in industries where the IT budget tends to be higher (e.g. financial and telecom firms) are associated with higher requirements analysis and design capabilities.

M2H1: Higher requirements analysis and design capabilities are associated with more positive perceptions of enterprise architecture.

To address these research questions and hypotheses, a survey of IT professionals was conducted. These IT professionals are members of the Society for Information Management, an association of senior IT executives and academics. In developing the survey, the requirements analysis and design items were derived from the key process areas or best practices from previous research studies and practitioner expertise from an experts group as part of the Society for Information Management Enterprise Architecture Working Group (SIMEAWG). Also the research from Beecham, Hall, and Rainer (2005) influenced the scale development. Their research indicated that as organizations matured their technical requirements problems tended to diminish while their organizational requirements problems maintained a constant trend. Therefore a focus on the organizational requirements maturity influenced the process of deciding on the scale items.

The information systems development items were derived from the SEI CMM(I). These had been used in a previous survey by Kappelman (1997) regarding Y2K strategies.

As this research was framed within an EA context, existent EA maturity models were reviewed, with a core of four EA maturity models being used to develop the survey's items. Additionally, the key IT and business alignment enablers and inhibitors posited by Luftman and McLean (2004) were integrated into the survey to reflect the importance of EA to alignment, and to reflect this persistent concern of top IT management.

Results indicate support for M1H1: Higher IS development capabilities are associated with higher requirements analysis and design capabilities. Hypotheses M1H2, M1H3, M1H4, M1H5, regarding the role of the organization size, as measured by head count, and IT budget variables, were unsupported. However, when combining IT budget with organization revenue to form combination variables, organization size did have an influence on IT development and requirements capabilities. Finally, hypothesis M2H1 (Higher requirements analysis and design capabilities are associated with more positive perceptions of enterprise architecture) was unsupported. However an inverse relationship was found between requirements practices and EA perceptions emerged.

This research study contributes to better understanding the important place of requirements to IS development capabilities. It reinforces the role of requirements in IS development capability maturity models and provides insight into critical factors of IS project failures. Finally, this result supports Brooks' (1995) theory that requirements are the essence of IS development and should be a key focal point for software process improvement programs. The contributions of research model number two regarding EA perceptions provides insight into how IT professionals view EA, with indications that they may be internally focused on the IT aspects rather than on broader, organization-wide issues on which EA is also focused. Moreover, this finding may contribute to knowledge about critical success factors of EA implementation by highlighting IT professionals' perspectives of EA, and the subsequent focus of IT professionals. This is important since IT professionals generally play a crucial role in the success of EA initiatives. Moreover, it provides insight into components of EA maturity models and assessments.

The organization of this research study is as follows. First, in chapter 2 a comprehensive literature review is conducted to illuminate the key constructs of the research model (information systems development, requirements analysis and design capabilities and enterprise architecture perceptions) and provide the theoretical foundations for these concepts and the measurement instruments to be used in the survey. Next, in chapter 3 methodology, a description of the research strategy, development, and execution are provided. Moreover, a detailed depiction of the sample and treatment for validity and reliability considerations are included. Chapter 4 follows with a thorough walk-through of the results from the statistical treatment of the survey results. Finally, chapter 5 concludes with the discussion and interpretation of these results along with the implications for the academic and research communities as well as practitioners. Limitations and future research directions conclude this chapter.

CHAPTER 2

LITERATURE REVIEW

Information systems (IS) permeate the global economy and are indispensable in society and organizations. With the decisive function of information systems in organizations today it seems the level of IS development failures is incommensurate with their critical role. This incongruity seems even more out of place when looking at the lengthy history of IS development projects not meeting requirements or exceeding budgetary and/or schedule limits (Krishnan & Kellner, 1999). It appears as if the performance of IS development efforts is not keeping up with the expectations of organizations.

Brooks (1995) hypothesizes that the difficulties of IS development can be framed with difficulties of essence and difficulties of accidents. This provides a conceptual structure in which to consider the characteristics of software and for thinking about IS development. Generally, improvements in IS development have come from resolving the accidental difficulties, those that served as barriers making software tasks difficult (Mays, 1994). But, Brooks (1995) maintains that resolving the essential difficulties, difficulties inherent in the general nature of software, should be the focus because it would bring about order-of-magnitude improvements in IS development. Further progress is based on focusing on the essence of software (Mays, 1994). These inherent difficulties of IS development derive from software's essential properties of complexity, conformity, changeability, and invisibility (Brooks, 1995, McConnell, 1999; Xia & Lee, 2005) and continue to be relevant today (Duggan & Reichgelt, 2006). Brooks (1995) concludes, "Many of the classical problems of developing software products derive from this essential complexity" (p. 183). In a similar tone, Sawyer, Sommerville, and Viller (1998) state: "No software process, whatever its "capability", can keep delivery times, costs and product quality under control if the requirements are poorly formulated or unstable" (p. 1).

Information Systems Development Capabilities and Maturity

According to Brooks, the accidental part of software engineering is the coding and testing, the IS development aspects (Brooks, 1995; McConnell, 1999). Put another way, the language, tools, and methods in use for IS development are the accidents (Berry, 2003). As will be discussed later, the requirements aspects included in IS development encompass Brooks' essence.

The concept of IS development in research has been described as being based on implicit and explicit assumptions (Hirschheim & Klein, 1989; Hirschheim, Klein, & Lyytinen, 1995). In their research, Hirschheim et al. (1995) propose that ontological and epistemological assumptions within IS development bring about different outcomes and outline the information systems being developed in terms of the two perspectives of its intended function and its structure. With this in mind, Hirschheim et al. (1995) define IS development as including the range of activities in the process of building (which includes analysis and design), implementing, and maintaining an IS. The analysis includes collecting, organizing, and analyzing the relevant facts about a notional IS and its environment while design encompasses the conception, generation, and formation of an IS (Hirschheim et al., 1995). Similarly, Iivari and Hirschheim (1996), using an earlier definition posited by Davis and Olson (1985), define IS development as "the analysis, design, technical implementation (construction), organizational implementation (institutionalization) and subsequent evolution (enhancement, maintenance) of information systems" (p. 611).

A concise definition, including the aspects described above, states that IS development involves the "analysis, design, and implementation of applications and systems to support business operations in an organizational context" (Xia & Lee, 2005, p. 46). This is similar to Swanson and Beath's (1989) even more concise definition of IS development as the "analysis, design, and implementation of new applications" (p. 296). Swanson and Beath (1989) highlight a traditional perspective on IS development which differentiates the maintenance function of IS development with that encompassing major enhancements. Thus the definitions are based on the size of the project. In lieu of this definition, they propose the differentiation between IS development and maintenance should be based on whether the work is on future systems or exisiting, installed systems. In this way, Swanson and Beath (1989) maintain, the definitions place a clearer emphasis on the relationship between the information system and business operations, not just technical-centric issues. Therefore work on installed systems, rather than being termed maintenance, is still considered development (whether "installed system development" or "future system development").

Another conceptualization of IS development is to consider the end state or the goal sought. For example, the goal of IS development "is the effective creation of a set of work products, comprising an operational system and its supporting documents" (Wasserman & Freeman, 1983, p. 57). No matter the semantics used in framing IS development, the development of information systems is a complex endeavor in an environment of stiff competition, uncertainty, instability, and with frequent technological and market disruptions (Nidumolu & Subramani, 2003). In conceptualizing IS development, this research paper attempts to be as consistent as possible with existing IS development research literature while at the same time basing the definition on both the domains of information technology and the organization. Therefore, this research paper will define IS development using the aforementioned definition from Xia and Lee (2005). This definition adequately conveys the technical aspect as well as the organization issues and requirements concerns that comprise IS development. Moreover, it recognizes the inherent complexity of IS development in that it must consider technological issues as well as organizational issues that, in many cases, are outside the control of the IS developers.

The use of the term capabilities in this research paper adequately captures the constructs of IS development and requirements as well as their intended purpose. Using capabilities indicates the vital importance of information systems to organizations as capabilities are considered essential to the sustained competitive advantage of organizations, especially when considering the resource-based view of the firm (Brits, Botha, & Herselman, 2007; Grant, 1996). Grant (1996) defines capability "as a firm's ability to perform repeatedly a productive task which relates either directly or indirectly to a firm's capacity for creating value through effecting the transformation of inputs into outputs" (p. 377). In line with Grant's view of capabilities as a source for competitive advantage, Kaner and Karni (2004) define a capability as "a distinctive attribute of a business unit that creates value for its customers. In integrating the concept of capabilities to the discipline of information systems, Peppard and Ward (2004) consider IS capabilities as irrevocably embedded within the organization. They also state that the strength of organizational IS capabilities are determined by their impact on the bottom line performance of the organization. They go further and consider IS capabilities as affecting the IS strategy, business strategy, IT operations, and business operations, which cumulatively impact overall organizational performance (Peppard & Ward, 2004). This conceptualization of IS capabilities emphasizes that capabilities have a relationship with and impact organizational performance by leveraging IT (Peppard & Ward, 2004). In a similar manner, Araya, Chaparo, Orero, and Joglar (2007) define capabilities directly related to IS as: "…an array of available intangible elements or factors directly related to the acquisition, processing, distribution and utilization of information. These elements enable an adequate development, deployment and utilization of resources directly related to IS/IT to achieve the desired results" (p. 632).

As indicated by the authors in the previous paragraph, a key factor in capabilities is that they are measured by the value they generate for the organization. Thus capabilities differentiate an organization from others and directly affect bottom line performance. An earlier definition put forth by King (1995) defines capability as "internally consistent combination of skills, processes, procedures, organizational structures, physical systems, information systems, and incentive systems that can produce superior levels of that which an organization desires" (p. 68). The work of Willcocks, Reynolds, and Feeny (2007) involving information systems capabilities justifies the use of the term capabilities as a composite of various IS development and requirements practices and perceptions. The definition put forth by Willcocks et al. (2007) is: "A capability is a distinctive set of human-based skills, orientations, attitudes, motivations, and behaviors that, when applied, can transform resources into specific business activities" (p. 128). The capturing of combinations of these objects in survey measurement instruments can help establish the capabilities of IS development and requirements. Moreover, Willcocks et al. (2007) found organizations can develop these capabilities through processes, culture, and structure giving an indication of the nature of items to be included in IS development and requirements measurement instruments.

These definitions give an indication of the importance of key process areas of IS development maturity as well as the role of user requirements inherent in the definition and use of capabilities within this research study. Specifically, in this research study, the use of the term IS development capabilities is consistent with and includes important fundamental properties from the aforementioned literature foundation. The term IS development capabilities as used here, is inclusive of both the technical and social aspects of IS development (IT artifact and social interactions) (Willcocks et al., 1997). The IS development term also indicates a level of IS success in creating value for customers (Kaner & Karni, 2004; King, 1995). It is important to remember IS development capabilities involve the transformation of inputs into an output or outputs as in a finished information system (Grant, 1996). Thus IS development capabilities as used in this research study is encompassing of the requisite factors facilitating the potential capacity for organizational resources and processes to be used in creating information systems that contribute to organizational performance. These factors are based on the literature base for IS development and capability maturity models, and expert opinion from practice all of which inform the creation of the survey measurement instrument for IS development.

Similarly, regarding requirements analysis and design capabilities, the definition as used in this research study includes requirements practices/processes and artifacts/outcomes. This is consistent with Willcocks et al. (2007) and King (1995) whose definitions of capabilities include both practices and outcomes. Furthermore, the requirements capability definition is encompassing of the three dimensions of requirements success as put forth by El Emam and Madhavji (1995): 1) quality of requirements processes; 2) user/customer satisfaction; and 3) productivity/cost effectiveness. Thus requirements analysis and design capabilities involve practices and outcomes of requirements processes with the potential capacity for requirements success by impacting organizational performance through its information systems. Again, as with IS development capabilities, the factors making up requirements capabilities are based on the literature base regarding requirements, and expert opinion from practice all of which inform the creation of the survey measurement instrument for requirements.

The IS discipline has engaged many of the accidental aspects or elements of IS development. Examples include increasingly more capable programming languages, development methodologies, and integrated environments (Brooks, 1995; McConnell, 1999). Advocating the tools aspects of IS development can be seen in the research by Beam, Palmer, and Sage (1987) who emphasize the combination of macroproductivity tools and microenhancement approaches to IS development. But one can see even with these improvements addressing many of the accidental elements, the state of IS development has not correspondingly improved, notwithstanding some progress, as far too many projects still come in over budget and over time without getting the requirements right, the essence element of IS development (Jones, 2009). Brooks contends this is largely caused by addressing the accidents rather than the essence of IS development.

In the rapid paced, innovative, global, hyper-competitive information age environment in which organizations compete today, organizations must be able to develop high quality information systems on time and at low cost. Indeed, the effective development and implementation of information systems is one of the most important success factors in business (Harter, Krishnan, & Slaughter, 2000).

In academia as well, IS development has been seen as integral to the "business" or core of the IS discipline (Hirschheim & Klein, 2003; Hirschheim, Klein, & Lyytinen, 1996). For example, a large number of IS research studies investigate the relationship between information technology, individuals, and organizations in the systems development context of creating information systems that are efficient, effective, and increase job satisfaction (Orlikowski & Baroudi, 1991). Additionally, several researchers have maligned the trend of some IS research that is not inclusive of the information technology (IT) artifact (Benbasat & Zmud, 2003; Orlikowski & Iacono, 2001). The "IT artifact" meaning is sometimes blurred. For instance, Orlikowski and Iacono (2001) define it as "those bundles of material and cultural properties packaged in some socially recognizable form such as hardware and/or software" (p. 121) and Benbasat and Zmud (2003) define it as "the application of IT to enable or support some task(s) embedded within a structure(s) that itself is embedded within a context(s)." Here, the hardware/software design of the IT artifact encapsulates the structures, routines, norms, and values implicit in the rich *contexts* within which the artifact is embedded" (p. 186).

Though IS development research is not the only type of research that contains the IT artifact, it is clear IS development research does include the IT artifact. Another example is included in a study of research articles from 1985-2006 in three of the top research journals in IS. In it Sidorova, Evangelopoulos, Valacich, and Ramakrishnan (2008) identify a change in IS research where less attention is focused on the development of the IT artifact. Instead, the authors maintain, the IS development research has shifted towards process-related and managerial issues.

When considering IS development capabilities, it is helpful to consider the factors or considerations that make up an IS development methodology. In a study of software development methodologies, Wasserman and Freeman's (1983) research indicated that a methodology should include 12 considerations: 1) be inclusive of the entire developmental process; 2) facilitate communication among key players or constituents; 3) support problem analysis and promote better understanding of the problem; 4) support both top-down and bottom-up development; 5) validate and verify system requirements; 6) identify design and performance constraints; 7) provide support for the intellectual work of the IS designers and IS organization, including project management; 8) recognize the inevitability of the evolution of the system; 9) be able to utilize automated tools; 10) support software configuration management; 11) be teachable; and 12) be open-ended regarding new tools and managerial methods that might modify the methodology. As can be seen from this list, requirements should be included in IS development decisions.

A complicating factor in any study of IS development is the sheer number of IS development methods or in navigating the methodological jungle, as Iivari, Hirschheim, and Klein (2001) call it. In their IS development study, they state there are over 1000 IS development methodologies. The problem with this plethora of IS development methodologies is that collective knowledge on IS development is elusive and difficult to synthesize into research studies (Iivari et al., 2001). They suggest the discussion should be about IS development approaches (classes of specific development methodologies that share common features) rather than IS development methods (Iivari et al., 2001). To increase research in this area, the authors contribute a four-tier framework to classify and facilitate understanding of the IS development methodologies. The four tiers, in the form of a pyramid, beginning from the top, consist of paradigms, then IS development approaches, followed by the third tier of IS development methodologies, and finally the IS development techniques form the bottom tier. An understanding of IS development methodologies in IS research is important to this paper's research into IS development and requirements because these methodologies are heavily influenced by programming, the accidental problems of the equation versus the essential problems. As Mylopoulos, Castro, and Kolp (2000) put it: "Software development techniques have traditionally been inspired and driven by the programming paradigm of the day" (p. 1). They go further, explaining how the programming paradigms influence the concepts, methods, and tools used throughout all phases of IS development.

Iivari et al. (2001) attempt to shift this focus from the accidents to the essential problems when investigating IS development research phenomena. This adjustment, facilitated by their framework, requires a change in mindset to the perception of IS development methodologies as instantiations of a higher level and broader abstraction of IS development approaches. Furthermore, the researchers believe their hierarchical framework sets the basis for system analysts to concentrate on generic features of IS development methodologies. These generic features encapsulate the essences of entire classes of IS development methodologies. The study of individual IS development methodologies will include the study of several accidental features. However, the discussion should be about IS development approaches (classes of specific development methodologies that share common features) rather than IS development methods (Iivari et al., 2001). This approach hones the research to what is important: the essence not the accidental properties of IS development. The essence involves thinking about the essential aspects of IS development, that of requirements.

Requirements Analysis and Design Practices

Countless IS development projects fail because of an inadequate requirements process (El Emam & Madhavji, 1995). The critical role of requirements in IS development can be understood in the context of Brooks' (1995) accident versus essence hypothesis. Getting requirements right is most notably identified with Brooks' contention that requirements are the essence of what IT professionals do, all the rest being accident or risk management. The essence of IS development difficulties can be viewed as the requirements process, defining the specification, design, and verification of complex information systems (Berry, 2003; McConnell, 1999). Brooks (1995) considers requirements the most important function of software builders.

Requirements analysis consists of the following processes: 1) define the scope; 2) plan the analysis; 3) gather information; 4) describe the enterprise; 5) take inventory of current systems; 6) requirements definition of new system; and 7) plan for transition

(Hay, 2003). Whether or not IS development projects are successful depends on how successfully they meet the needs of the applicable users and the usage environment (Cheng & Atlee, 2007; Nuseibeh & Easterbrook, 2000). These needs are the requirements. The process of determining or eliciting the requirements has been called several terms such as requirements engineering, requirements determination, or requirements analysis. Beecham et al. (2005) define the requirements process as "activities performed in the requirements phase that culminate in producing a document containing the software requirements specification" (p. 248). Generally the process of gathering requirements consists of the following core functions: 1) eliciting requirements; 2) modeling and analyzing requirements; 3) the communication of requirements; 4) agreement on the requirements; and 5) evolving or changing the requirements (Nuseibeh & Easterbrook, 2000). The process of garnering requirements from the stakeholders and users of a prospective information system is of critical importance to IS development. It is one of the most (if not *the* most) difficult steps in IS development, and has effects throughout the system development life cycle (Browne & Rogich, 2001; Halbleib, 2004). A successful requirements process will increase the probability of successful information systems development (Browne & Rogich, 2001; Davis, 1989).

Extensive research work into understanding and improving the requirements process by Sawyer, Sommerville, and Viller (1998) helps to frame the discussion of requirements analysis and the relationship to the IS development process. In their review of requirements literature, Sawyer et al. (1998) emphasize requirements is not a discrete activity, but point out several research studies indicating the requirements process is cyclical. The cyclical nature of the requirements process captures the three main challenges of requirements. These challenges are first, that they are hard to collect from users and stakeholders because many times they do not really know what they need or want. Many times the actual requirements must emerge during the requirements gathering process. Second, the changeability of requirements (as Brooks (1995) also mentions) impedes successful requirements gathering. Third, the reality of the constraints of time and cost on requirements gathering (Sawyer et al., 1998). In synthesizing these studies, Sawyer et al. (1998) create a requirements process spiral consisting of the three activities of: 1) requirements analysis and validation; 2) requirements negotiation; and 3) requirements elicitation. The spiral consists of three phases, 1) requirements problems, 2) requirements document, and 3) the draft statement of requirements. As the organization moves through successive iterations of the spiral, costs and the volume of information generated increase and the quality of the requirements gathered improves.

Not getting the requirements right can lead to costly errors throughout the IS development lifecycle. For instance, discovering errors during the requirements phase costs significantly less to fix than finding the errors during the IS development or implementation phases (Berry, 2003; Hay, 2003; Jones, 2008). One estimate puts an exponential growth pattern with the cost of errors in IS development (Figure 1).

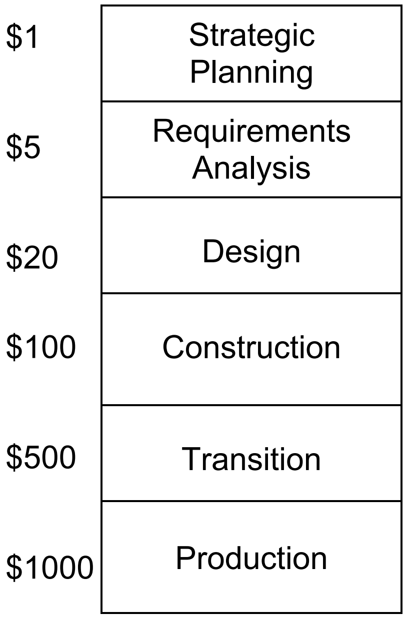


Figure 1. The cost of an error (Hay, 2003).

Another approach to understanding the importance of requirements analysis and design to IS development is research by Jones (2008) into measuring defect potentials. Jones (2008) defines defect potentials as “the probable numbers of defects that will be found during the development of software applications” (p. 2). Based on his study of around 600 organizations and 13,000 IS projects, Jones maintains the most serious IS project defects are not actually initiated in the code but in the requirements and design and that in the United States these defect potentials average 45% of all software defects.

This growth in the costs of errors throughout the system development life cycle, specifically within the requirements phase manifests itself with research indicating an increased and improved effort in the conduct of requirements analysis and design leads to a better development and implementation of information systems (Berry, 2003; Daugulis, 2000).

Challenges to getting the requirements right are myriad. These challenges include the dynamicism of customer requirements, a tendency for IS developers to focus primarily on functional requirements while neglecting political or organizational issues, the difficulty in obtaining domain expertise from the customers and miscommunication between developers and customers (Sawyer, Sommerville, & Viller, 1997). This changeability of requirements is not a new phenomenon. In a study of implicit and explicit assumptions involved in IS development, Hirschheim and Klein (1989) specifically recognized the constantly changing nature of organizational requirements which causes reality to be extremely complex and elusive.

Much of the research into understanding and improving the requirements process aims to identify best practices. For instance, in a case study at Digital Equipment Corporation, Hutchings and Knox (1995) identify cross-functional teams, direct customer involvement, dedicated design areas, and a defined requirements change control and configuration management process as key factors in a successful requirement process.

Brooks (1995) asserts the challenges in IS development center around complexity, but others, while agreeing with Brooks' overall premise, do not believe complexity is the root cause of the crisis in IS development. For instance, Cox (1995) took a human-centric view of the essence and accident model of Brooks. While agreeing with Brooks' overall proposition, Cox believes a human-centric view of Brooks can lead to a silver bullet. However, this silver bullet cannot be technology, rather, it requires a paradigm shift. This paradigm shift, Cox believes, is to change the commercial transaction view of software where the purchase, selling, and ownership of software is based on acquiring the utility inherent in the bits rather than actually owning the bits themselves. In this manner, the complexity of information systems is encapsulated so that it is not a factor in the information system at all (Cox, 1995).

Requirements Outcomes

Getting requirements right sets the foundation for successful IS development projects. Kamata and Tamai (2005) conducted a case study in order to discover relationships between requirements quality (getting the requirements right) and IS project success. They conducted a study of 32 IS projects in a business application development company over a three year period. Their research indicated a relationship between requirements and IS project outcomes. In a closely related study, Sommerville and Ransom (2005) investigated the relationship of a requirements maturity model (which they had previously created) and business benefits. Their findings indicated that higher requirements maturity did lead to business benefits but the authors acknowledge that the fact companies were participating in the process improvement may have led to the increase in performance (the Hawthorne effect where participation in the experiment leads to the changes, not necessarily the treatment).

However, given the importance and emphasis on requirements in IS development, the failure of IS programs to get the requirements right has a lengthy history and getting those requirements right continues to be a problem (Beecham, Hall, & Rainer, 2005; Brooks, 1995; Hutchings & Knox, 1995). In his book *The Mythical Man-Month*, Frederick Brooks (1995) discusses the essential importance of getting the requirements right in the software development process and the challenges the IS field has in getting those requirements right. Brooks (1995) states that “The hardest single part of building a software system is deciding precisely what to build. No other part of the conceptual work is as difficult as establishing the detailed technical requirements…. No other part of the work so cripples the system if done wrong. No other part is more difficult to rectify later” (p. 199). Not getting requirements right seems to be a continuing problem for the IS field. Case study research of 12 software development companies by Hall, Beecham, and Rainer (2002) indicated about 50% of all IS development problems were due to requirements problems.

The Disconnect Between Information Systems Development and Requirements Alignment

The inability to get requirements right is evident in the concept of alignment. Getting the requirements right is an important aspect of successful alignment and meeting the requirements of both users and the usage environment. In other words, achieving alignment between the two is a key determinant of whether or not IS development projects are successful (Cheng & Atlee, 2007).

A useful conceptualization of alignment is the definition posited by Reich and Benbasat (1996) which adequately encompasses the essence of the concept “the degree to which the IT mission, objectives, and plans support and are supported by the business mission, objectives, and plans” (1996, p. 56). A similar definition is where the needs, goals, or objectives of one component or function are consistent with the needs, goals, objectives of another component (Oh & Pinsonneault, 2007). The level of fit and integration within the business strategy, IT strategy, business infrastructure, and IT infrastructure is the definition used by Henderson and Venkatraman (1993) in their research creating the strategic alignment model. IS developers and organizations cannot get alignment right if they cannot get the requirements right, since the organizational goals needed to align with are part of those requirements. An overemphasis or over focusing on the technical points of IS development instead of a focus on the alignment of IS and organizational goals has been cited as a contributing factor for some IS development failures (Kim & Peterson, 2003).

These problems with IS development oftentimes expand to larger threats to the value of IS and IS organizations. These threats are regularly levied in the academic journals (Brown & Hagel, 2003; Carr, 2003; Dearden, 1966; Dearden, 1972). These discussions can create fissures or gaps in the foundation of IS organizations when looking at the future viability of IS to organizations in the overall business environment. These gaps can become larger fault lines when organizations consider the past and future progress and value of IS to the organization. However, Luftman, Lewis, and Oldach (1993) maintain IT can indeed provide strategic value to businesses and alter the nature of organizations and industries by effectively and efficiently being employed. This can be accomplished by the alignment of “business strategy, information technology strategy, organizational infrastructure and processes, and I/T infrastructure and processes” (Luftman et al., 1993, p. 198).

The issue of alignment in organizational and information systems research has been a rich area of study for many years. This has been due, in part, to the promise of increased organizational benefits from alignment, the desire to explain the information technology productivity paradox, and alignment consistently being rated a top concern for managers and executives (Croteau, Solomon, Raymond, & Bergeron, 2001; Chan, Huff, Barclay, & Copeland, 1997; Gregor, Hart, & Martin, 2007; Luftman & McLean, 2004; Niederman, Brancheau, & Wetherbe, 1991).

Henderson and Venkatraman (1993) seminally formalized strategic alignment within the context of the strategic management of information technology with the strategic alignment model (Henderson & Venkatraman, 1993). The strategic alignment model is based on strategic alignment and functional integration. Strategic fit reconciles the external and internal domains. The external domain concerns the business environment and decisions of the organization. The internal domain relates to choices about administrative structure and design of critical business processes. Functional integration, on the other hand, involves the interrelationship between the IT strategy and business strategy. Thus, the strategic alignment model consists of strategic integration between business and IT strategy in the external domain and operational integration between organization and IT infrastructure and processes within the internal domain. For strategic alignment to occur, external and internal alignment must be present. A lack of alignment, these authors posit, is the root cause of the inability of information technology to create value in an organization (a reason for the IT-productivity paradox) (Henderson & Venkatraman, 1993).

Another alignment model that adds to Henderson and Venkatraman's work is Luftman and Brier (1999) and Luftman's (2000) research identifying twelve components of alignment. This research categorized the components into three sections. Under the business strategy section: 1) business scope; 2) distinctive competencies; and 3) business governance. Under organization infrastructure and process section: 1) administrative structure; 2) processes; 3) skills; 4) technology scope; 5) systemic competencies; and 6) IT governance. Finally, under the IT infrastructure and processes section: 1) architecture; 2) processes; and 3) skills. From their multi-year survey of business executives, they identified enablers and inhibitors to the synchronization between business and IT in organizations (Luftman & Brier, 1999).

Information Systems Project Success and Failure

Another symptom of the disconnect between IS development and the failure of getting the requirements right is the issue of IS project failure. Information system project failure is comparatively high when viewed in relation to other high technology projects (Yeo, 2002). Quoting from the *Chaos Chronicles* report of the Standish Group International, Nelson (2005) reports only a 34% IT project success rate in 13,522 projects at Fortune 500 firms. These failures were the result of one or more of the following situations: total failure, cost and/or time overruns, and implementation with reduced features or functions not meeting user requirements (Nelson, 2005; Yeo, 2002). A conservative estimate puts the annual cost of failed IS projects in the US from $60 to $70 billion annually (Charette, 2005). Additionally, that same Standish report listed the failure to clearly state requirements and incomplete requirements were the third and first in rank respectively in a survey of IT managers' reasons for why IT projects fail (Kamata & Tamai, 2007). The consequences of these failures can result in decreased revenues, damage to corporate brand or reputation, exposure to legal liabilities, and a decrease in productivity (Baltzan & Phillips, 2007). Moreover, besides impact to financial bottom lines, the failures of IS development can bring significant organizational consequences, even potentially leading to ruin (Goulielmos, 2003; Xia & Lee, 2005).

A review of IS project failures by Yeo (2002) provides a synthesis of research in better understand the underlying characteristics and causes of this high failure rate. Yeo highlights the work on IS failures of Lyytinen and Hirschheim (1988) who categorized IS failures into four types: 1) correspondence failure where the systems design objectives are not met; 2) process failure where the information system is not developed within budget and/or schedule; 3) interaction failure where the level of end user usage and satisfaction is inadequate; and 4) expectation failure where the IS does not meet stakeholders' requirements or expectations. Another categorization of IS failures Yeo (2002) describes is Flowers' (1996) who takes a situational view of failures. Flowers describes an IS failure as one when 1) the entire system does not operate as expected, leading to performance below expectations; 2) upon implementation is rejected by users or does not perform as intended; 3) the costs exceed the benefits; or 4) the IS project is abandoned before completion. The final research into IS project failure Yeo (2002) synthesizes is Sauer's (1999) "triangle of dependencies" between key stakeholders of individual information systems including the IS, the project's supporter and the IS project's organization. In this model, an information system is considered successful as long as there is sufficient resources and support put towards it.

Looking at IS project failure through the lens of evaluating IS project success, Nelson (2005) reviewed the literature on IT project success and failure (including the aforementioned articles). His research led him to determine that the evaluation of project success should contain process and outcome criteria, specifically process-related criteria: 1) time (whether or not the project was on schedule); 2) cost (on budget); 3) product (the output produced met quality and other specifications such as requirements, usability, ease of use, modifiability, and maintainability); and the outcome-related criteria of: 4) use (whether or not the product is being used); 5) learning (does the product increase constituent knowledge and prepare the organization for the future); and 6) value (the product directly and positively impacts efficiency and/or effectiveness).

Nelson (2005) goes further and proposes that a more effective method to address IS project failure is not only to consider the success criteria above, but to also borrow from the playbook of the US military and attempt to learn from past mistakes through retrospective or post-implementation reviews. This method also incorporates the perspectives of various stakeholder groups into the IS development process and evaluation. Potential benefits from such action include: 1) organizational learning; 2) continuous improvement; 3) improved estimates and schedules; 4) facilitation of team building; and 5) an improvement in recognition and organizational self-reflection before tackling new projects (Nelson, 2005). This research finding was similar to a case study of a NASA software development project conducted by Abdel-Hamid and Madnick (1990). Their study of a failed IS development project developed a postmortem diagnostic tool in order to facilitate learning from past failures in order to improve chances of more successful IS development projects in the future.

Another study and synthesis of the success of IS development projects was conducted by Zmud (1979). In this meta analysis of research studies Zmud specifically investigated the impact of individual differences (cognitive, personality, demographic, and situational variables) on IS success. These individual differences were grouped into cognitive and attitudinal influences. Zmud's conclusion was these individual differences have a significant impact on IS success.

A study of IS development failures by Goulielmos (2003) indicates a failure of organizations to learn from these failures because they are regarded as technical failures rather than being seen through the lens of a socio-technical process within an organizational setting. The study of IS development failures, in other words, concentrates on the accidental problems and not the essential problems of Brooks'.

An important consideration in IS project success and failure is the inclusion of the dimension of risk in IS development projects. The notion of risk in IS development involves IS project risk management where organizations identify and control risk factors to increase the chance of IS success (Barki, Rivard, & Talbot, 1993; Kappelman, McKeeman, & Zhang, 2006; McFarlan, 1981; Wallace, Keil, & Raj, 2004). To achieve beneficial outcomes from IS development projects, Wallace et al. (2004) assert organizations must manage social subsystem risk (the nature of information systems where they exist in a social context) and technical subsystem risk (the nature of information systems where IS development is creating a technical artifact of some complexity based on a set of requirements). Indeed the results of their research clearly indicate the importance of managing requirements risk as the correlation between requirements risk and technical subsystem risk is the highest of any other relationship in their model.

In Kappelman et al.'s (2006) study of failed IS projects, they contend IS project management is very immature regarding risk management. They maintain there are early warning signs or red flags for impending IS project failure. In their findings of early warning signs of the manifestation of risks, requirements issues were ranked as the number two of 53 early warning signs, just below the lack of top management support or commitment to the project (Kappelman et al., 2006). Additionally, not interviewing key stakeholders for project requirements is ranked at number five.

As alluded to by Brooks' hypothesis, complexity is a key factor in the high failure rate of IS projects. In a survey of 541 IS managers, Xia and Lee (2004 and 2005) conceptualized, developed, and validated key measures of IS development complexity. After validating and a confirmatory data analysis, four components of IS development complexity were derived: structural organizational complexity, structural IT complexity, dynamic organizational complexity, and dynamic IT complexity. Structural complexity involves the number of project elements, interrelationships, and integration while dynamic complexity consists of ambiguity, variability, and dynamism (this is where the changing nature of requirements is accounted for) (Xia & Lee, 2004, 2005). Also of note was their finding that the technological aspects of complexity, though more apparent, have less significant effects on IS development performance and outcomes than do the organizational aspects (Xia & Lee, 2004). Again this seems to align with Brooks' accident versus essence hypothesis where the most meaningful effect on IS development will derive from those activities that are not inclusively technical effects or advances. Xia and Lee's (2004) IS development complexity framework is useful as a common language for which to facilitate a discussion of the key dimensions of IS development complexity.

Research on approximately 12,000 IS development projects over a period of 18 years by Jones (2003) provides another viewpoint as to the role of complexity in IS development and requirements. Jones' (2003) research indicated a significant increase in activities and specialization of skills for larger information systems compared to smaller ones as measured by function points (Jones, 2003). Additionally, the type of IS being developed was found to influence the IS development practices with a wide variation of practices being observed (e.g., over 40 methods for gathering requirements, over 700 programming languages). To further analyze this, Jones (2003) categorized the types of software as: 1) military; 2) systems; 3) commercial; 4) outsourced; 5) management information systems; and 6) end-user development. This categorization scheme is helpful in order to identify common characteristics, development methods, and constraints found in various types of IS development projects. These research findings demonstrated that the size and type of IS development projects strongly influenced the IS development activities as well as the need for specific skills of IS persons (Jones, 2003). The research also indicated no specific IS development methodology or programming language ensured or denied successful project outcomes, though the use of a formal IS development method tends to lead to higher quality outcomes.

Characterizing Information Systems Development Capabilities thru Maturity

Maturity Models

In understanding the construct of IS development capabilities, this research relies on the conceptual foundation of maturity models such as operationalized in the capability maturity model. Maturity models provide a basis for the control of IT and organizational processes and practices by identifying strengths, areas for improvement, and subsequent activities to effect improvement in the processes and practices. The standards defined by the maturity models establish levels of maturity and can be used in managing the IS or organizational improvements desired (Steghuis, Daneva, & van Eck, 2005).

A model containing essential elements of successful processes in one or multiple disciplines while describing an evolutionary road map for improvement from unstructured to more mature processes is considered a capability maturity model (Chrissis, Konrad, & Shrum, 2003). Maturity models are grounded in the notion of process improvement which is based on work by W. A. Shewhart, Joseph Juran, and Edwards Deming (Chrissis, Konrad, & Schrum, 2003). Based on work at IBM, Watts Humphrey observed the quality of software produced was directly related to the quality of the process used to produce it (Rao & Jamieson, 2003). Humphrey's work was heavily influenced by the work of Deming on quality and statistical control, such as the Shewhart-Deming improvement cycle of 'Plan-Do-Check-Act' (Rao & Jamieson, 2003). From this work with statistical process control and quantitative measurement, maturity models took form. Humphrey (1988) defines a process "as a sequence of tasks that, when properly performed, produces the desired result" (p. 74).

In research about information systems development and maturity models, Wilson (1996) defined a maturity model as an "abstraction of the changes of form which a class of phenomena typically exhibit in a single pass of their life-cycle…and it is composed of a number of identifiable stages through which an instance of the class will pass in a particular sequence" (p. 423). Maturity models are further rooted in concepts of the quality management field such as Crosby's quality management maturity grid (Kaner &

Karni, 2004). Crosby's maturity grid discussed specific layers in which organizations would evolve as they attempted to achieve a level of excellence in quality management (Kaner & Karni, 2004). When developing process maturity models the process should be: 1) decomposed into its parts or stages; 2) contain task criteria for entry, validation, and exit to the next stage; 3) and be regularly reviewed, analyzed, and used as a basis for process improvement (Radice, Roth, O'Hara, & Ciarfella, 1985).

Moreover, in order to understand and provide insight into growth and development patterns corresponding with information technology capability, stage hypotheses and stage theories have been proposed (Koenig, 2000). The key to stage theory is its predictive value (Koenig, 1992). In turn, maturity models have been derived from stage theory. When assessing organizations' maturity, capabilities, and practices, the use of maturity models are useful. Maturity models are based on the belief that systems, organizations, processes, and other entities go through distinct stages over time (Nolan, 1973). Nolan stresses that the purpose for determining the organizational stage of maturity is to assist in deciding on the optimum deployment of IT resources. An assessment of maturity stages can help an organization understand where it has come from and where it wants to go (and help develop a plan to get there). It can also help to develop proactive and reasonable plans rather than reactive ones (Wilson, 1997). Maturity models include a set of specifically described stages, occurring in a given sequence, a list of aspects or conditions for the changing from one stage to another, and a list of the aspects or conditions that must be present in order for the transition to another stage to have occurred and be identified as having occurred (Wilson, 1997).

Maturity models have been categorized into three groups, depending on their use and contribution. 1) a descriptive tool such as providing an assessment of an organization's IS development current situation, helping to support process improvements (by describing the practices that an organization must perform in order to improve its IS development processes), illustrating projected benefits, and supporting IS development program management efforts by quantifying progress and as a tool for architects to manage their development effort; 2) a prescriptive tool such as setting goals for the future based on desired achievement level of a maturity model or specifying target objectives to identify areas for improvement; and 3) a comparative tool such as benchmarking the effectiveness of an IS development practice by comparing with other IS development programs in other organizations using the same or similar maturity model and providing yardstick against which to periodically measure improvement (Rosemann & DeBruin, 2005; Saha, 2006)

Foundations of Maturity Models

The S Curve

An underlying foundation for maturity models is the concept of S curves. An S curve generally plots the relationship between the time or effort into a project or process and the performance gains from the particular project or process. When the results are graphed, a sinusoidal line (S curve) is usually obtained (Figure 2), with the gradient angle signifying the investment productivity (Nieto, Lopez, & Cruz, 1998). Some authors state an important characteristic of S curves is that progress should not be represented in terms of time but in terms of investment towards its development (Foster 1986 in Nieto, Lopez, & Cruz, 1998 p. 445). However, others justify the use of time and historical analysis/past performance as a legitimate performance characteristic (Greiner, 1978; Nieto, Lopez, & Cruz, 1998). Representing the S curve of a product or process can assist with strategic management decisions and in making business predictions since the S curve can provide information as to the level of effort or output required for a certain level of process productivity increase. It can also reveal the existence of limits in performance (Nieto, Lopez, & Cruz, 1998).

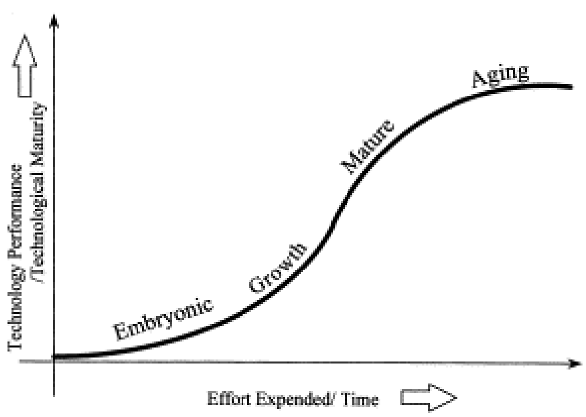


Figure 2 S curve (Nieto, Lopez, & Cruz, 1998).

Greiner (1972) contributed to the knowledge and development of maturity models by emphasizing the role of an organization's history in its future growth and success rather than a singular focus on outside forces. Additionally, an organization's practices such as their management focus and style, and organizational structure also play a role in the evolution of a company. According to Greiner (1972), based on case study analysis, organizations cycle between periods of evolution (periods of growth typically lasting from four to eight years) and periods of revolution (periods of significant disruption to an organization and management practices). Greiner's (1972) model of organization development included the five key dimensions of: 1) age of the organization; 2) size of the organization; 3) stages of evolution; 4) stages of revolution; and 5) growth rate of the industry. The combination of evolutionary and revolutionary stages with their respective management styles and management problems resulted in a framework of five phases of growth or organizational development: 1) growth through creativity/crisis of leadership; 2) growth through direction/ crisis of autonomy; 3) growth through delegation/ crisis of control; 4) growth through coordination/ crisis of red tape; and 5) growth through collaboration/crisis. The model proposed by Greiner is not a checklist of how to transition to the next higher stage, rather it is posited as an inevitable process for organizations that exist in a business environment and its final stage can be considered an end state of ultimate growth (Wilson, 1997). Enterprise architecture and enterprise architecture maturity model questions (and demographic questions such as industry and organization size) dealing with the "As Is" architecture can provide a level of documentation of organizational history advocated by Greiner. These questions can be used for the internal focus that Greiner states is where the basis for future organizational success lies.

Nolan's (1979) stage theory or maturity model is based on the plot of IS budgets over time based on case studies of a number of companies. The resulting S-curve identified several specific points or stages of growth patterns within IS. Originally there were four stages which over time evolved to a maturity model of six stages (Nolan, 1973, 1979). In a similar manner as Greiner (1972), Nolan's stated purpose for his stage theory is to assist IS managers through periods of crises. Even at this early date, Nolan highlighted the fact organizational leadership was not very successful in the use of IS for strategic decision-making. The basis of stage theory is the notion that components within a system move through a predictable pattern of stages over time and these components, or elements, can be described (Nolan, 1973). Generally, there are two classes of stage theory, cyclical/life cycle and progressive/developmental (Koenig, 2000). The cyclical (or life cycle) class assumes a repetitive and predictable life cycle of development while the progressive non-cyclical view attempts to predict more broad or general maturity of stages of the overall growth of information technology (Koenig, 2000). Two underlying characteristics for the stages require 1) the characteristics in each stage must be distinct and testable; and 2) the processes taking place to move from one stage to the next should be identified and described. Nolan's (1973) research of three firms led him to his normative stage theory of managing the computer resource. In it, he hypothesized the planning, organizing, and controlling activities associated with managing IS resources changes and evolves in patterns correlated to the following four stages: 1. computer acquisition (initiation); 2. intense system development (contagion); 3. proliferation of controls (control); and 4. user/service orientation (integration) (Nolan, 1973).

Later, Nolan (1979) revised his model to encompass six stages. These six stages consist of: 1) initiation; 2) contagion; 3) control; 4) integration; 5) data administration; and 6) maturity. The model was expanded to six stages to recognize the trend of managing data versus managing computers and with the experience captured in additional case studies. Integral to this revised theory are the notions of slack and control. An environment of control is characterized by a variety of financial, management, and performance controls to ensure the effectiveness and efficiency of IS activities. A slack environment is one which lacks such controls (Nolan, 1979). Achieving the right balance of control and slack within the various organizational stages is important to maintain an environment of organizational learning and to encourage the progression through the stages. The stages of growth imply different management emphasis at each stage depending on the technology, thereby providing a framework for organizational and IS planning.

The advent of stage theory has been influential in IS, whether as seen in the Software Engineering Institute's capability maturity model, Ross' (2003) EA maturity model or other applications. Indeed, Weber (1987) specifically calls out Nolan's stage hypothesis and theory as one of only a few candidates for an authentic paradigm of IS.

The Software Engineering Institute's capability maturity model (SEI CMM)

In order to improve information systems development processes, practices and quality, the Software Engineering Institute (SEI) developed the capability maturity model (CMM) in the late 1980s. The CMM has a staged architecture, where all key process areas for the maturity stage (and all previous or lower stages) must be satisfied in order to mature or attain that certain stage (Sawyer et al., 1997). The intention of the CMM is to better manage the software process (Paulk, Curtis, Chrissis, & Weber, 1993) and to increase quality and productivity in IS development (Duggan, 2004). For example, Paulk et al. (1993) note more mature processes increase the process capability of organizations. In developing a software process maturity framework, Humphrey (1988) identified the characteristics of an effective process to include: 1) predictability and statistical control; and 2) measurable. Humphrey's work utilized previous work by Radice, Harding, Munnis, & Phillips (1985) and Crosby (1979) containing the concept of a maturity grid. A maturity grid is comprised of a matrix of cells each of which contains attributes or characteristics that define each level of process stage maturity.

From Humphrey's work, Carnegie Mellon's Software Engineering Institute developed a software capability maturity model (SEI CMM). It is probably the best known maturity model in the IT arena and is used as a foundation in developing many other maturity models in a variety of applications and approaches. The work of Humphrey on the CMM is founded on using an operational process for evaluation and assessment rather than only tools or methodologies (Radice, Roth, O'Hara, & Ciarfella, 1985).

Originally, several different maturity models were developed (the software CMM, systems engineering CMM, software acquisition CMM, and people CMM) each with five levels of maturity identifying the various practices required to attain the level of maturity (Humphrey, Snyder, & Willis, 1991). The five levels are initial, repeatable, defined, managed, and optimized. Later, these various CMMs have been consolidated into the capability maturity model integration (CMMI) (Chrissis, Konrad, & Shrum, 2003). Additionally, the CMMI contains six levels and can be represented as a continuous representation of capability levels or as a staged representation of maturity levels. In the continuous representation, the levels are incomplete, performed, managed, defined, quantitatively managed, and optimizing. In the staged representation, the initial level (Level 0-incomplete in the continuous representation) is not used and the next level (Level 1) is called initial. The remaining levels share the same names as the continuous representation (Chrissis et al., 2003). The staged representation consists of established sets of process areas used to define or propose an improvement plan for an organization which is described by the maturity levels. The continuous representation or approach is more flexible and enables an organization to select specific process areas in which to improve relative to these process areas (as represented by capability levels) (Chrissis et al., 2003). Therefore the SEI CMM(I)'s focus is on improving processes for IS development by identifying the key process areas and the evolutionary path through the maturity stages or levels (Chrissis et al., 2003). The SEI CMM(I) is based on the promise of process improvement and process management where the improvement in processes leads to higher quality in the information systems developed (Chrissis et al., 2003). Table 1 gives a sampling of research articles that explore CMM(I) and IS development practices.

Table 1

Articles Exploring Both CMM(I) and IS Development

Author(s) Content	
Ashrafi, 2003	Impact of SPI on s/w quality
Berry, 2003	Requirements are essence, why getting requirements right is hard
Boehm, 2006	Future trends in IS Dev, IS Dev complexity, rapid change
Charette, 2005	Why software fails
Conradi & Fuggetta, 2002	Improvements to SPI programs, refocus towards business/ users
Duggan, 2004	Suggestions to improve IS Dev, Three IS delivery methods paradigms (SDLC, iterative/RAD, s/w reuse),
Glass, 1999	Studies of technology affects s/w quality. CMM results in improvements.
Goldenson, El Emam, Herbsleb, & Deephouse, 1996	Success/failure factors in SPI, SEI CMM, benefits of SPI-SEI CMM
Hall, Beecham, & Rainer, 2002	As CMM increases—req probs decrease, 50% of all probs were with req.
Hutchings & Knox, 1995	Req best practices, CMM not adequately address RAD/Req,
Layman, 2005	Implementing CMM
Jones, 2002	Higher CMM results in lower s/w defects
Jones, 2008	Higher CMM results in lower s/w defects.
Linscomb, 2003	Probs with CMMI, does not adequately support Req
Niazi, Wilson, & Zowghi, 2003	A maturity model for implementation of SPI, why don't orgs implement SPI?
Petkov & Petkova, 2008	Broadening IS Dev productivity research, interdisciplinary research, s/w cost estimation.
Putnam, 1994	Benefits from moving up on CMM
Sawyer, Sommerville, & Viller 1998	Improving the requirements process, requirements not sufficiently addressed by CMM, a Req CMM,
Staples et al., 2007	Why firms don't adopt CMMI
Wilson, 1997	Maturity models in IS Dev

Outcomes of the SEI CMM(I)

Using the CMM(I) to characterize and illuminate the information systems development process capability provides a strong basis for many reasons. One reason is the popularity and heavy usage of it, which demonstrates some level of appropriateness (Beecham, Hall, & Rainer, 2005; El Emam & Madhavji, 1995). Indeed, Rogoway (1998) declares the CMM a *de facto* standard regarding process improvement assessments. Another reason is the adaptability or customizability of the CMM(I) to the specific needs of the organization (Beecham et al., 2005; Paulk et al., 1995). This fact is evident when surveying the number of CMMI adaptations in existence, for instance the strategic planning CMM (Hackos, 1997), the requirements CMM (Beecham et al., 2005), and the people CMM (Curtis, Hefley, & Miller, 2001). Reifer (2000) identifies 34 CMMs that have been developed. A third reason supporting the use of the CMMI is the continued viability of it. The Software Engineering Institute continues to actively support it and update it (Beecham et al., 2005).

Prior research investigating the relationship between process maturity level and organizational outcomes has shown positive relationships between process maturity and software quality (Harter & Slaughter, 2000), product quality (Bohn, 1995), and other increased quality outputs (Fenton & Neil, 1999; Ryan, 2000; Zahran, 1998). In early studies of CMM, Glass (1999) found a consistent trend of improvement when organizations implement the CMM. An evolution in CMM level from 1 to 3 led to better software quality, higher investment returns, and higher productivity at Raytheon according to research conducted by Dion (1993). Interestingly, the improvement in the CMM levels also had secondary, positive effects on the corporate culture of the organization as well. The improved IS development process and corresponding results at Raytheon led to an increase in morale and a lowering of the absenteeism and attrition rates in the software development branch (Dion, 1993). Hutchings and Knox (1995) found similarly positive effects on the corporate culture at Digital Equipment Corporation with the initiation of a requirements-based CMM. In a broader research survey of 138 individuals in 56 organizations results showed positive effects on staff morale from attaining higher CMM maturity (Goldenson, El Emam, Herbsleb, & Deephouse, 1996). Similarly, in a study of the CMM software process improvement methodology on information system development, Clark (2000) found an increase in maturity level of one can reduce the effort required in software development by between 4 to 11%. To strengthen his case, Clark isolated the effects of implementing the CMM from other effects and IS development improvements. Finally, in a research study that combined the results of 12 case studies of 12 software development organizations, Hall, Beecham, and Rainer (2002) found the number of requirements problems tended to trend downward as the CMM level increased in these organizations.

In an opposite approach, Goldenson, El Emam, Herbsleb, and Deephouse (1996) found little evidence software process improvement assessments and efforts (such as the SEI CMM) had negative effects on organizations. Specifically, they found only four percent of respondents labeled their efforts as counterproductive and more than 80% said these efforts had not stifled creativity or placed bureaucratic loads on their organizations (Goldenson et al., 1996). Instead, their research indicated organizations with higher maturity (based on the SEI CMM) are more likely to report higher product quality and staff productivity, meeting customer requests, and meeting budget constraints.

An examination of organizations that moved from CMM Level 1 to Level 3 was conducted by Putnam (1994). The research findings indicated reductions in schedule time, staffing requirements, and effort corresponding to the increase in maturity levels. Similarly, in an extensive study of thirteen companies conducted in order to examine the benefits of initiating a software process improvement program incorporating the CMM, Herbsleb, Carleton, Rozum, Siegel, and Zubrow (1994) documented such benefits as productivity gains, a reduction in IS development time, an overall increase in software quality, and increasing gains in detection of software errors. Most significant was a finding of a return on investment or on business value of five to six times versus cost. Also, the increased detection of software errors provides significant cost savings as identified by Fagan’s (1986) software inspection process, which is based on the works of Deming and Juran. Fagan (1986) found addressing software defects from the beginning of the process will save organizations money.

In a multi-year study of IS development practices, Jones (2002) concludes large applications (larger than 10,000 function points) tend to be more successful when the developing organizations are at or above Level 3 of the CMM. Overall, organizations with a higher level of IS development maturity as measured by the SEI CMM results in an improvement in defect potentials, IS code with fewer high-severity defects, and an increase in defect removal rates (Jones, 2008). The results from Jones’ research are supported by research by Krishnan and Keller (1999). In their study of 45 IS development projects, their research results indicated where CMM was adapted, a significant reduction in the number of defects occurred. Moreover, their research indicated the more consistent implementation of CMM practices could result in product quality improvements (Krishnan & Keller, 1999).

To achieve benefits from the use of maturity models, however, there are various best practices to consider. For instance, Adler, McGarry, Irion-Talbot, and Binney (2005) maintain there are four critical success factors to the implementation of the SEI CMM: 1) creating sufficient strategic momentum to attain the CMM certification; 2) establishing and maintaining the commitment of management to the project; 3) encouraging extensive staff participation in the CMM processes; and 4) establishing a CMM-based culture that is a catalyst for software programmers' acceptance of the CMM-style discipline required to increase the maturity levels.

However, even with the demonstrated benefits of maturity models (Adler et al., 2005; Cameron, 2007; Scott, 2007) it is important to understand why organizations do not adopt maturity models given the research as to the advantages of implementation. Staples, Niazi, Jeffery, Abrahams, Byatt, and Murphy (2007) conducted an exploratory research study into this question. Their research indicated several reasons why organizations do not proceed with the use of maturity models. One important reason was organizational size. Smaller organizations believed their small size impeded the adoption of maturity models. The researchers did not isolate the reasoning: whether from cost, resources, lack of believe in the benefits of CMMI, or inapplicability of CMMI to small organizations. Other reasons for not adopting CMMI include: 1) the services were too costly; 2) the organizations lacked the time for the services; and 3) they used other software process improvement approaches (Staples et al., 2007). The exploratory study confirmed earlier research findings which indicated the resources required for CMMI are thought to be prohibitive for smaller organizations and are not applicable for them either (Brodman & Johnson, 1994; Staples et al., 2007). Smaller organizations have different needs such as a need to concentrate systems development resources on the product itself, not the process (Staples et al., 2007). Moreover, criticisms about onerous burdens on programmers and vast amounts of documentation are other reasons organizations state for not adopting maturity models (Adler et al., 2005).

In a study of software process improvement models including the CMM and CMMI, Cattaneo, Fuggetta, and Sciuto (2001) argue both the CMM and CMMI are too narrowly focused on technical and engineering issues and do not have sufficient organizational context to them. Again, this finding supports Brooks' (1995) essence versus accident hypothesis and the importance of alignment of IT and the business and acknowledging that strictly technical approaches to solving difficulties cannot be the only focus in IS development. When considering the benefits gained by increasing maturity of IS development processes and organizations, it is important to remember a conclusion from research into requirements good practices by Sawyer, Sommerville, and Viller (1997) which stated "even otherwise mature organizations repeatedly experience requirements problems" (p. 19).

Requirements and IS Development Research Gaps

Even with these aforementioned studies indicating benefits to IS development from the initiation of software performance improvement programs such as CMM(I), and the importance of requirements practices and processes there remains a significant, even critical, neglect of research into the relationship of requirements and IS development (Beecham, Hall, & Rainer, 2005; Sawyer, Sommerville, & Viller, 1997; Sawyer, Sommerville, & Viller,1998). This dearth of research can be seen when looking at CMM(I) research. Beecham et al. (2005) state that "both software organizations and the academic community are aware that the requirements phase of software development is in need of further support" (p. 247). Sommerville and Ransom (2005) put it more succinctly by saying requirements is mostly outside the scope of process improvement models. It is imperative to address this gap when looking at the historical context of software development vis-á-vis Brooks' (1995) hypothesis regarding the difficulty in getting the requirements right and the importance of addressing the essential problems of information systems development.

For instance, an extensive quantitative and qualitative research analysis into the software process improvement literature and various factors affecting software process improvement was conducted by Rainer and Hall (2003). Among these factors, the ones identified in all the research were: 1) the importance of senior leadership buy-in and involvement; and 2) the involving of organizational staff in the improvement area. Both of these previous factors are classified as internal process ownership by Rainer and Hall (2003). The other factors consisted of training/mentoring, reviews, standards and procedures, experienced staff, inspections, clear lanes of responsibility for software process improvement, clear goals, respect for those persons involved with software process improvement, the creation of process action teams, management of the software process improvement project, and metrics (Rainer & Hall, 2003). It is of particular note that in Rainer and Hall's research (which included other software process improvement research), the subject of requirements is not discussed or analyzed. However, Rainer and Hall (2003) attribute the fact there are many differences in the factors in the findings of software process improvement research to the focus of researchers on identifying and researching key factors in software process improvement programs.

One approach to bridging this gap is to increase the research about the payoffs from requirements processes and practices. As pointed out by Davis and Hsia (1994) the gap between research and practice may be viewed as the largest gap anywhere within the IS discipline. As discussed by Damian and Chisan (2006) in research into the relationships between requirements engineering and other IS development process, “requirements engineering is an important component of effective software engineering, yet more research is needed to demonstrate the benefits. While the existing literature suggests effective requirements engineering can lead to improvements…there is little evidence to support this” (p. 433). In their 30-month case study of an IS development program, an effective requirements process led to improvements in productivity, quality, and risk management which they could directly link to improved requirements practices. Most significantly, especially when viewed with the results of Jones (2008) research discussed previously, Damian and Chisan’s (2006) data indicated a substantial decline in IS support requests and post deployment defects when compared to earlier software releases.

The gap in research between information systems development practices and maturity (as measured by the SEI CMM(I)) and requirements analysis and design capabilities can, in part, be considered a result of maturity models not sufficiently addressing requirements. Organizations typically embark on a software process improvement program (e.g., SEI CMM) either for compliance or to address problems with their IS development programs such as poor software quality, functionality issues, or over time and budget (Layman, 2005). However, there is a gap in accounting for the vital role of requirements within software process improvement programs. For example, regarding the capability maturity model Integration, Linscomb (2003) acknowledges requirements engineering processes are addressed in the CMMI, but he stresses it is of limited support regarding requirements maturity in the context of industry standards and practices.

While acknowledging the positive impact of the CMM(I) software process improvement methods on software quality and software organizations, several research studies have highlighted an area of improvement for maturity models by increasing their coverage of requirements aspects. Sawyer, Sommerville, and Viller (1997) conducted research with multiple organizations which indicated the CMM and ISO 9000 did not adequately address requirements processes. They go further and maintain one reason is the requirements process is less understood and less homogenous than the IS development process (Sawyer et al., 1997). To address this problem, they developed a three stage requirements process maturity model to fill the gap from IS process maturity models like CMM. Hutchings and Knox (1995) also deemed the attention paid to requirements by the SEI CMM as inadequate. Their research indicated the SEI concentrated more on ensuring the agreed-upon requirements are included in a management process and are included in the information system, but paid inadequate attention to whether the requirements, in fact, were the right requirements.

Another study revealing a need for further support of the requirements phase of information systems development was completed by Beecham, Hall, and Rainer (2005). By studying the type of requirements problems encountered by organizations, they could isolate two specific categories of requirements problems: organizational and technical. Additionally, as organizations matured, the technical requirements problems tended to diminish but the organizational problems maintained a steady level (Beecham et al., 2005). Their findings suggest the SEI CMMI (which consists of many organizational best practices) left a gap in organizational requirements process improvements. Other limitations of the CMMI regarding requirements include: 1) failure to adequately identify and define the technical and organizational aspects of requirements; 2) not recognizing requirements process problems; 3) not adequately assessing and setting requirements improvement priorities; 4) not relating requirements process problems to improvement goals; and 5) not relating the requirements improvement goals to the overall CMMI guidelines and activities (Beecham et al., 2005). But instead of developing a new requirements maturity framework, they exploited the existing SEI CMMI framework to develop a requirements capability maturity model (R-CMM). Their R-CMM came about in response to their research which indicated continuing problems with requirements and in endeavoring to support organizations needing help with their information systems development and requirements practices.

IS Development, Requirements, and Enterprise Architecture

When addressing the problems with IS development by looking at the essential problem of requirements, one method to accomplish this is to consider the requirements analysis and design capabilities within the context of enterprise architecture (EA). The importance of this approach is to obtain knowledge of the architecture of the organization in order to develop information systems supporting (and in alignment with) that architecture (Hay, 2003). Requirements analysis accomplished within the context of enterprise architecture facilitates the role of the enterprise rather than requirements in the context or terms of particular technologies (Hay, 2003). Enterprise architecture significantly contributes to IS development, specifically in its role providing a template with which to guide IS development (Tang, Han, & Chen 2004).

As has been discussed previously, there is a need to turn around the record of IS failures by improving IS development. And this should best be accomplished by attacking the essential difficulties of IS development such as better requirements, which, in turn, may be improved and supported by enterprise architecture, which overall provides a better plan to accomplish IS development. In their research about requirements maturity, Niazi, Cox, and Verner (2008) lament: “Yet despite the regularly documented and recognized importance of [requirements], little work has been done on developing ways to improve the requirements process” (p. 215). By exploring the relationship between IS development, requirements, and enterprise architecture, potential improvements to IS development and requirements may be realized, providing the elixir to Niazi et al.’s concerns. Enterprise architecture could refocus IS development improvements to the essence and away from the accidents.

This section of the paper investigates requirements analysis and design and IS development within the conceptual structure or framework of enterprise architecture. EA is integral to the requirements process and IS development. John Zachman, defined architecture as: “that set of design artifacts, or descriptive representations, that are relevant for describing an object such that it can be produced to requirements (quality) as well as maintained over the period of its useful life (change)” (1997, p. 5). This is different than the steps of the traditional system development life cycle methodology. Enterprise architecture, such as elaborated within the Zachman EA Framework, emphasizes the different perspectives of key constituents within the organization who have a stake in a particular system and its development. Moreover, as described above regarding the difficulty of the requirements process, Berry (2003) points out that significant change in IS development will only come about when the dynamicism of requirements and its effects are dealt with. Enterprise architecture may provide the solution, or at least a significant part of it. The developing discipline of EA involving the integration of software project processes within EA frameworks has been cited as a growing trend in IS development and a possible solution to the challenge of complexity to successful IS development (Boehm, 2006; Finkelstein, 2004, 2007). A number of research articles (Table 2) discuss two or more of these key concepts (i.e., CMM(I), IS development, maturity models, and requirements) within the context of enterprise architecture indicating a definite trend in exploring the symbiotic relationships among these constructs.

Table 2

Articles with EA, IS Development, Maturity Models, and/or Requirements Concepts

Author(s)	Maturity Models	EA	IS Development (ISD)	Requirements	Content	SEI CMM
Boehm, 2006		X	X	X	Future trends in IS Dev, IS Dev complexity, rapid change, others	X
Chen & Pozgay, 2002	X	X	X		EA as a discipline for IS Dev	
Finkelstein, 2004		X	X		Old ways of IS Dev inadequate, do not deal with change and rapid pace, IS Dev strategies/ approaches must use EA	
Finkelstein, 2007		X	X		Old ways of IS Dev inadequate, do not deal with change and rapid pace, ISD strategies/approaches must use EA	
Hirvonen & Pulkkinen, 2004	X X			X	EA mgmt tool, requirements bridge, practice related framework for EA mgmt (EA Grid)	
Tang, Han, & Chen, 2004		X	X	X	EA a major role in IS Dev, compares EA frameworks, uses goals, inputs, outcomes to analyze EA frameworks, level of detail in EA is dependent on level of risk	

Enterprise Architecture

Enterprise architecture is a solution to increasing invisibility, change, and complexity, three of the obstacles to solving the essential difficulties of IS development stated by Brooks (1995). There are many EA definitions in existence, with none being universally accepted (Janssen & Hjort-Madsen, 2007; Rohloff, 2005). However, according to Schekkerman (2004), the Institute of Electrical and Electronics Engineers, many businesses practicing EA, and the US Department of Defense agree “that architecture is about the structure of important things (systems or enterprises), their components, and how the components fit and work together to fulfill some purpose” (p. 21). One definition states enterprise architecture is “the organizing logic for applications, data, and infrastructure technologies, as captured in a set of policies and technical choices, intended to enable the firm’s business strategy” (Ross, 2003, p. 32). In other words, it helps transform IS into an aligned strategic asset, rather than stove piping IS planning into a series of tactical planning exercises centered around specific and separate IT applications or solutions.

The broad range of EA definitions in existence, with no consensus, indicates an early stage of maturity for EA as a whole (Chen & Pozgay, 2002; Schelp & Stutz, 2007; Winter & Fischer, 2006). Similarly, there is no consensus on what EA actually consists of, which frameworks, layers, artifacts, instantiations, or relationships (Winter & Fischer, 2007). Also, the practice of viewing EA as a prescriptive or descriptive approach or a product and a process also results in the diversity of definitions (Van Den Bent, 2006). A leading proponent of EA, The Society for Information Management Enterprise Architecture Working Group (SIMEAWG) (2009), defines architecture as “the set of descriptive representations about an object" (p. 1) while enterprise architecture is "the holistic set of descriptions about the enterprise over time" (p. 1). According to the IEEE 1471-2000 standard for the "Recommended Practice for Architectural Description of Software-Intensive Systems", architecture is defined as: "the fundamental organization of a system embodied in its components, their relationships to each other and to the environment and the principles guiding its design and evolution". An 'architectural description' is defined as "a collection of products to document an architecture". The concept of EA as a managerial tool is another common view (Hirvonen & Pulkkinen, 2004). As stated in their research: "EA serves as a master plan for managing the business, the information, the applications and the ICT infrastructure" (p. 2). Although EA is, by these definitions, about the architecture of the total enterprise, typically the focus is on the business and IT architecture (Goethals, Snoeck, Lemahieu, & Vandenblucke, 2006).

In this research paper, enterprise architecture will be defined using the SIMEAWG's (2009) definition. This definition rightly ensures the holistic, enterprise-wide view of enterprise architecture, ensuring (much like this paper's IS development definition) that there is not an undue techno-centric concentration or view of EA, but instead, enterprise considerations are given adequate exposure.

Enterprise Architecture and Alignment

A central goal of EA is the alignment of IS requirements to the goals and objectives of an organization (Henderson & Venkatraman, 1993; van der Raadt, Hoorn, & van Vliet, 2005; Vaidyanathan, 2005; Wieringa et al., 2003). EA has also been positioned as a business technology alignment tool (Vaidyanathan, 2005). Enterprise architecture can affect alignment by demonstrating how various components of an organization align together and by showing the gaps between the present and desired states of the business and its IS (Gregor, Hart, & Martin, 2007). EA ensures congruency between organizational strategies, process, and IT requirements forming an inclusive IT strategy (Young, 2001).

The role of EA as an enabler of alignment was studied by Gregor et al. (2007) using Strategic Alignment Theory. They identified key enablers of alignment in an organization as organization-wide communication and analytical decision-making. Thus, if an organization's EA can lead to improvements in or create an environment conducive to organization-wide communication and analytical decision-making, the organization as a whole will be better aligned and possibly more effective (Gregor et al., 2007).

Enterprise Architecture and Complexity

As discussed previously, the complexity of information systems along with other challenges such as communication about requirements is a continual impediment to effective requirements analysis and design and information systems development. As aforementioned, Brooks asserts that complexity, along with conformity, changeability, and invisibility, is an inherent, or essential property of information systems (Brooks, 1995). Chen and Pozgay (2002) contend EA can be used in the building of information systems and to facilitate the communication about the information system requirements. In their research on EA and IS development, they advocate the establishment of EA as a discipline and as an essential capability within organizations' IS development programs. They believe EA has the potential to improve IS development capabilities within organizations (Chen & Pozgay, 2002).

Enterprise architecture can further address this trend of increasing complexity (an essential difficulty of information systems) in information systems (and in organizations) by facilitating the abstraction of system complexity. Brooks (1995) says that descriptions of information systems, due to their inherent complexity, often abstract away its essence. And this, he maintains, is a problem unique to software in that simplified models do not help because they ignore complexities that are the essence of the software or information system. However, EA, by using multiple views or perspectives of an information system can prevent the loss of the essence (complexity) of the actual design. In this way EA facilitates system thinking by providing a framework to see interrelationships and change patterns (the stages or milestones of a transition plan). EA can broaden the scope of software process improvement programs such as CMM(I) to better integrate IS and organizational operations and management. Senge (1992) echoes Brooks' concerns when he says "today, systems thinking is needed more than ever because we are becoming overwhelmed by complexity" (p. 69). To interject IS development problems into Senge's (1992) justification for systems thinking in the context of minimizing organizational breakdowns: IS failures, in spite of very capable programmers and advanced IS programming languages and methods, still occur because these individual and diverse capabilities (advanced though they are) are not brought together as a system into a productive whole. Senge (1992) says "systems thinking is a discipline for seeing wholes. It is a framework for seeing interrelationships rather than things, for seeing patterns of change rather than static 'snapshots' "(p. 68).

The discipline of enterprise architecture can facilitate this vision. EA supports and refocuses IS development efforts to a systems view (a more organization-centric rather than technology-centric subsystem view). The research of Cattaneo, Fuggetta, and Sciuto (2001) recommending that software process improvement, in order to succeed, must refocus to include organizational considerations, may have found its answer in EA. Cattaneo et al., (2001) indict software process improvement programs for a lack of business orientation. An EA-influenced software process improvement program (like an EA-enhanced CMM(I)) may be the answer.

The Role of Enterprise Architecture

In an article about IS development strategies for the 21st century, Finkelstein (2004) asserts IS development strategies and approaches used today are the reason for the IS problems presently existing. These approaches lead to redundant data, redundant processes, and stove-piped systems. Much of the blame is due to the requirements process relying on the IT staff to define the requirements based on interviewing users about operational business needs (Finkelstein, 2004 & 2007). This is a problem because the business requirements are difficult to determine, thus the systems resulting from these requirements are not aligned with corporate goals, and strategic directions are ambiguous. Overall, Finkelstein (2004) claims this existing approach is too technology-dependent and does not work well with the complex, dynamic environment characterizing IS development in today’s organizations. Instead, Finkelstein (2004) advocates designing for tomorrow by using enterprise architecture with enterprise engineering in order to develop information systems based on organizational strategic plans of the future. Doing so will reduce IS costs and create a system better aligned and able to adjust with rapid changes.

In establishing EA-informed requirements practices, it is often desirable to make explicit and organize the existing information about the enterprise. This is creating the baseline, or 'as-is' architecture. One general approach advocated by Armour, Kaisler, & Liu (1999) is to characterize the work view (the enterprise's organization's products and services), characterizing the function view (the IT applications supporting the organization's functions), characterizing the information view (the relationships among the organization's information (via ERD, Object Modeling Technology) and the infrastructure view (the specific components of the information systems in use such as hardware, software, network topology, and telecommunications). The conglomeration of these four views comprises the business view. Once the baseline architecture is created, the next step is creating the target or 'to-be' architecture by using the same architectural views described previously except they now describe where the organization desires to be (Armour, Kaisler, & Liu 1999). The 'as-is' or baseline architecture also provides a foundation for use in the retrospective or post-implementation reviews that Nelson (2005) advocates. The use of EA as a living document can provide a historical paper trail for organizations to use in improving their IS projects by learning from the past. This historical record-keeping facet of EA also facilitates process maturity by accounting for an organization's history, as discussed by Greiner (1972).

The symbiotic relationship with requirements that EA offers can best be seen in the EA Management Grid, a framework based on decision making studies in IS, software development, and in EA (Pulkkinen & Hirvonen, 2005). Pulkkinen and Hirvonen (2004) sought to reconcile the gap between management consulting and software development. They began with the premise of EA consulting, planning, and development filling this gap. Their subsequent EA Management Grid, which is a tool to manage requirements among other aspects of work, integrates the various requirements of an organization (from business requirements, to strategic-level IS requirements, to the lower-level operational systems and data management requirements) (Hirvonen & Pulkkinen, 2004; Pulkkinen & Hirvonen, 2005). In their EA management grid, the EA planning and development process is a spiral design where decision about requirements are made at the enterprise/top management, domain/operative management, and system level/IT management (differing levels of abstraction/decision making) considering all four common EA components or dimensions (business architecture, information architecture, systems or applications architecture, and technology architecture). The enterprise-level decisions encompass the entire scope of the organization, while domain-level decisions are concerned with a business function or process and system-level decisions concern a specific system and detailed descriptions of structures (Pulkkinen & Hirvonen, 2005). Thus, all EA-related decisions (including system requirements) go through all levels of an enterprise for all the EA dimensions, with a reduction in the level of abstraction as the decisions go down in level. After conducting case studies on seven EA projects, the authors' research supported the idea that IS development efforts must begin with decisions made at the enterprise level, which includes strategic-level IS requirements within a business architecture component of EA, otherwise, problems ensued (Pulkkinen & Hirvonen, 2005). This is a shift of focus towards comprehensive and strategic management of the IS assets through the EA management grid.

Research Questions, Research Model, and Hypotheses

This literature review has discussed the gap between IS development and requirements analysis and design capabilities. This exploration of IS development and requirements has been informed and is guided by Brooks' (1995) accidents versus essence hypothesis. The continuing relevancy of Brooks' (1995) hypothesis to IS development and requirements is indicated by the number of articles in Table 3.

Table 3

Articles Concerning Brooks' Accident vs Essence Hypothesis

Authors IS	Development (ISD)	Requirements	Content	Brooks	ISD Success/ Failure
Berry, 2003	X	X	Requirements are essence, why getting requirements right is hard	X	
Cox, 1995	X	X	Focus on human-centric view of essence v. accident rather than techno-view. The problem is not complexity but changing the commercial transaction exchange	X	
Damian & Chisan, 2006	X X		Relationship of requirements to ISD		
Duggan, 2004	X	X	Suggestions to improve IS Dev.	X	

(table continues)

Table 3 *(continues)*

Authors IS	Development (ISD)	Requirements	Content	Brooks	ISD Success/ Failure
Duggan & Reichgelt, 2006	X		A meta lit review of s/w quality research, uses IS delivery instead of IS Dev to acct for outsourcing, purchase, rental [cloud?], etc,	X X	
Duggan & Thachenkary, 2003	X	X	JAD and achieving higher quality requirements	X X	
Glass, 1999	X		Studies of technology affects s/w quality. Found CMM results in improvements (based on previous studies).	X	
Iivari, Hirschheim, & Klein, 2001	X		A framework of ISD approaches and methodologies	X	
McConnell, 1999	X	X	Key principles of software engineering.	X	
Xia & Lee, 2005	X		ISD is complex, IS Dev failure rates are high, complexity of IS', IS Dev complexity construct validation, complexity.	X	

Moreover, the potential influential impact of the discipline of EA on requirements and, in turn, on IS development has been considered. The substantial research describing and indicating positive outcomes from adopting software process improvement methods such as SEI CMM(I) was synthesized in order to characterize and quantify IS development capabilities. However, regarding requirements, the research record is not as substantial nor has it had as much impact on practice.

Though research indicates positive outcomes of improved requirements on IS development, the adoption and use of improved requirements practices has not occurred as it has with the CMM(I) (Kamata & Tamai, 2007). Researchers similarly echo the essence of Damian and Chisan's (2006) comments and call for more research into why there is not more acceptance of requirements practices improvements in industry as has occurred with IS development improvement practices such as CMM(I). These findings lead to two research questions:

> Research question 1: What is the relationship between information systems development capabilities and requirements capabilities?
>
> Research question 2: What is the relationship between information systems development and requirements capabilities and perceptions of enterprise architecture?

This research study's Research Model 1 (Figure 3) includes the moderating variables of organization size and IT budget. A summary of research studies discussing organization size and organizational resources in relation to CMM(I) is in Table 4.

Table 4

Organization Size and Resources In Relation to CMM(I) Articles

Authors Organization	Size	Organizational Resources
Brodman & Johnson, 1994	X	X
Ein-Dor & Segev, 1978	X	X
Fitzgerald & O'Kane, 1999		
Goldenson, El Emam, Herbsleb, & Deephouse, 1996	X	
Lumsden, 2007	X	
Pino, Garcia, & Piattini, 2008	X	
Richardson & von Wangenheim, 2007	X	X
Ross, Weil, & Robertson, 2006		
Strigel, 2007	X	

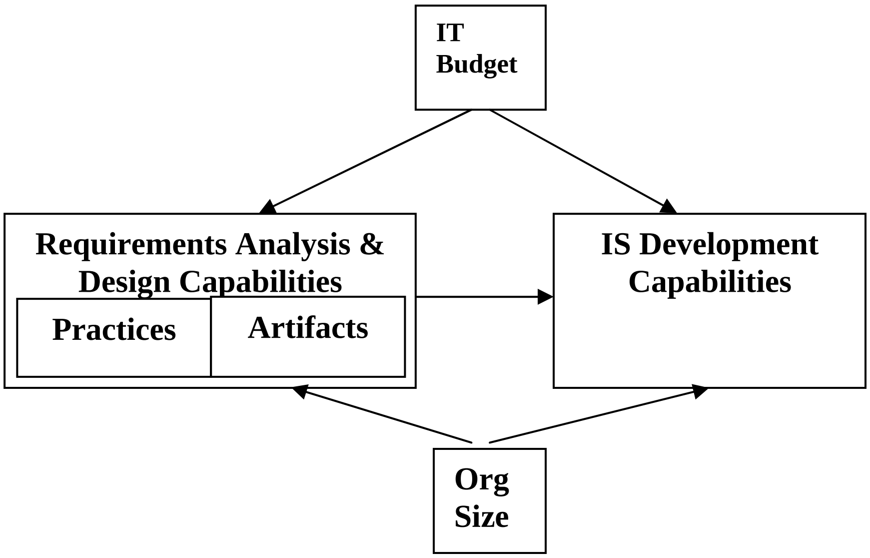


Figure 3 Research Model 1.

Much research on IS requirements describe the symbiotic relationship of requirements and IS development. Although there are varying conceptualizations of successful information systems, many requirements researchers claim IS failures can be directly attributed to poor, inadequate, or low requirements capabilities. For instance, El Emam and Madhavji (1995) state higher requirements capabilities (through an improvement in requirements practices) would be positively associated with economic and software quality outcomes. In an assessment of requirements research, Cheng and Atlee (2007) state the success of an IS is based on meeting individual and organizational requirements. Nuseibeh and Easterbrook (2000) also say IS success hinges on meeting requirements. Similarly, Browne and Rogich (2001) describe how improvements in requirements methods can positively impact the IS development processes and, in turn, the effectiveness of developed information systems. With similar research findings indicating a positive and direct relationship between requirements and IS development success, Kamata and Tamai (2007) investigated 32 IS projects and their research indicated a clear and strong impact on IS development success from quality requirements practices. Vessey and Conger (1994) unequivocally state IS requirements is the most critical contributor to successful IS development projects, a view echoed by Carroll and Swatman (1998). Finally, Halbleib (2004) concisely states that "requirements drive the development process" (p. 8) and requirements can determine project success.

These and other research findings present unambiguous indications of the positive effects on IS development that comes from higher requirement capabilities. These findings also tend to confirm that the focus of IS development is best not on the technology but rather the intended purpose of the IS must be a primary concern when assessing the success of information systems. Finally, the role of requirements in IS development in this research study is informed by the durability over time of Brooks' (1995) hypothesis of requirements being the essence of IS development, thus the following relationship is hypothesized:

> M1H1: Higher IS development capabilities are associated with higher requirements analysis and design capabilities.

When examining the relationship between requirements and IS development capabilities involving a software process improvement plan such as the capability maturity model, it is important to consider the effects of other variables. Conventional wisdom asserts such software process improvement efforts are only applicable to larger organizations because of the resources required making implementation of an software process improvement program too difficult for smaller organizations or that small organizations (which may be working primarily on smaller IS projects) have no need for the structured techniques advocated by software process improvement programs (Johnson & Brodman, 1999; Pino, Garcia, & Piattini, 2008; Richardson & von Wangenheim, 2007). This dichotomy has been recognized by the Software Engineering Institute who has begun a CMMI in small settings initiative to assist small organizations with the CMMI.

While some maintain there are no differences or obstacles for smaller organizations in adopting software process improvement plans (Lumsden, 2007), others feel smaller organizations must apply different approaches for successful implementations (Richardson & von Wangenheim, 2007; Strigel, 2007). Strigel (2007) believes communication within organizations is the fundamental difference as to whether or not larger and smaller organizations apply the same software process improvement practices. Richardson and von Wangenheim (2007) feel small organizations' uniqueness (in terms of specific business models, goals, market segmentation or niches, size, resource availability, and managerial capability, among other characteristics) requires a different approach than larger organizations to applying the CMM. Examples of similar lines of thinking include new assessment methods of software process improvement models specifically tailored to smaller organizations include: ISO/IEC 15504, Rapid Assessment for Process Improvement for Software Development (RAPID), Software Process Improvement Initiation (SPINI), and Software Process Improvement in Regions of Europe (SPIRE) (Richardson & von Wangenheim, 2007). Therefore, based on this discussion the following hypotheses are suggested:

M1H2: Larger organizations are associated with higher IS development capabilities when compared to smaller organizations.

M1H3: Larger organizations are associated with higher requirements analysis and design capabilities when compared to smaller organizations.

Combined with the research of the variable of organization size affecting the success of CMM(I) processes, another organizational resource, IT budget, has been proposed as a key determinant of IS success (Ein-Dor & Segev, 1978) or one affecting the implementation of CMM(I) (Richardson & von Wangenheim, 2007). Therefore, the following hypotheses are posited:

M1H4: Organizations in industries where the IT budget tends to be higher (e.g. financial and telecom firms) are associated with higher IS development capabilities.

M1H5: Organizations in industries where the IT budget tends to be higher (e.g. financial and telecom firms) are associated with higher requirements analysis and design capabilities.

This research study's research model 2 adds the perceptions of enterprise architecture to the relationship of IS development and requirements (Figure 4).

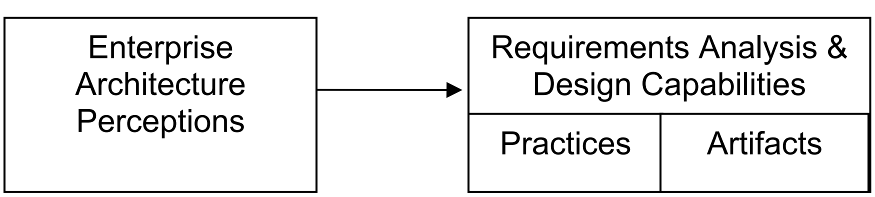


Figure 4 Research Model 2.

The inclusion of EA in the relationship of IS development and requirements capabilities reflects the growing body of research indicating a significant role for EA in the success of IS development projects. This interjection of EA also conveys the impact of EA on improving the requirements process. EA does this by increasing the knowledge of the organization and injecting this knowledge into the requirements process to inform the entire IS development process. As Hay (2003) points out, a requirements analysis process that is accomplished within the context of EA elevates the role of the enterprise in IS development rather than particular technologies.

Enterprise architecture also has been cited as an important contributor to the IS development life cycle (Finkelstein, 2004 & 2007; Tang & Han, 2006). Indeed Chen and Pozgay (2002) claim the practice of EA is a fundamental discipline for information systems. In studying and describing key components of EA practice, Chen and Pozgay (2002) document current EA practices and capabilities in IS development organizations. This research experience supported their premise claiming EA does have a significant impact on IS development. Further, Finkelstein (2004 & 2007) considers IS development within the context of EA as the key to building successful information systems that are more dynamic and able to adequately adjust to the rapid changes required of many information systems.

With the previous support for the role of requirements regarding IS development, it is desirable to consider the contributions of EA to both requirements and IS development. As previously discussed, EA can potentially shift the focus of requirements from a technical-centric view to one encompassing organizational goals and requirements. In a similar manner, EA also shifts the focus of IS development away from technologically-induced tunnel vision to one that considers organizational requirements and enterprise-wide factors. In studying the relationship of IS development and requirements through the lens of EA, it is the premise of this research study that if organizations tend to assign a higher role and value for requirements capabilities then they would value the more encompassing discipline and practices of enterprise architecture (i.e., they have more positive perceptions of EA). This is supported since an important aspect of requirements capabilities leading to successful IS development is including broader, organizational and social issues and not just myopically considering technologic concerns in the requirements process. Additionally, an important factor of getting the requirements right is achieving alignment between the IS and the users' needs (Cheng & Atlee, 2007; Kim & Peterson, 2003).

The role of EA in facilitating organizational alignment has been shown in research (Gregor, Hart, & Martin, 2007; van der Raadt, Hoorn, & van Vliet, 2005; Young, 2001). Moreover, the more holistic approach EA offers, in turn, increases the probability of more adequately achieving alignment and getting the requirements right which increases the potential for higher IS development capabilities and success (Brooks, 1995; Hall, Beecham, & Rainer, 2002; Hay, 2003; Kamata & Tamai, 2005). This is because IS development capabilities must include organizational perspectives via requirements. EA broadens the focus of IS development away from a techno-centric view of the accidental aspects and difficulties of IS development towards a more organizational perspective addressing the essential difficulties of IS development. Moreover, both RAD and EA share many critical success factors (such as a broader, more holistic view of IS projects, importance of senior leadership involvement). Since

EA is a newer phenomena than RAD and in many ways an expansion of RAD, and since higher RAD capabilities are associated with greater IS success, the following hypothesis is presented:

> M2H1: Higher requirements analysis and design capabilities are associated with more positive perceptions of enterprise architecture.

This research study's methodology, as will be described in chapter 3, is organized around these research models that are thought to describe the characteristics in which IS development and requirements capabilities along with organization size, IT budget, and EA perceptions relate to each other.

CHAPTER 3

METHODOLOGY

The intent of this research project is to investigate the relationship between information systems development and requirements analysis and design capabilities as well as the relationship between requirements analysis and design capabilities and enterprise architecture perceptions. To address this goal, individual instruments were developed based on an extensive literature review, input from an experts group, and existing maturity models of information systems development and enterprise architecture. The methodology used in testing the research question and hypotheses is presented in this chapter, including explanations of the development of the survey instrumentation and execution, the nature of the respondents to the survey, the data collection, and data analysis and procedures that were used in this study.

Population and Sample

The population from which the sample was derived consists of senior IT professionals who are members of the Society for Information Management (SIM). Members of SIM consist mostly of IT executives and senior managers in both the public and private sectors plus some academics, consultants, vendors, and other experienced IT professionals. SIM members are located primarily in the United States. The membership list was made available through SIM's Enterprise Architecture Working Group (SIMEAWG) and it consisted of about 2860 usable e-mail addresses of SIM members. This same body of potential respondents has been used in previous IS research. For example the longitudinal studies of Jerry Luftman (e.g., Luftman, Kempaiah, & Nash, 2005) that identify the top critical issues for IT executives, the research by Brancheau, Janz, and Wetherbe (1996) about key issues in IT management, and the Y2K study of Kappelman (1997, 2000).

Research Strategy

A research strategy using an electronic survey was the chosen approach to address the research question. In developing the survey, the SIMEAWG was instrumental in providing expertise with the various items used.

Research Design

Information Systems Development Capabilities Construct

To capture the information systems development capabilities of respondents' organizations, items from the Software Engineering Institute capability maturity model (later the capability maturity model integration (CMMI)) were used (Tables 5 and 6). The final list of items were derived from the literature base for IS development and capability maturity models as well as expert opinion from practice.

Table 5

Information Systems Development Capabilities Instrument

15. For software development and/or maintenance, our IS department specifies and uses a comprehensive set of processes and/or procedures for:
a. Establishing customer agreement on requirements
b. Identifying the training needs of IS professionals
c. Establishing quality goals with customers
d. Estimating all resource needs
e. Tracking progress and resource use
f. Software quality assurance
g. Continuous process improvement
h. Coordination and communication among stakeholders
i. Selecting, contracting, tracking, and reviewing software contractors/outsourcers
j. Analyzing problems and preventing re-occurrence
k. Tailoring the process to project specific needs
l. Continuous productivity improvements

Table 6

Additional Information Systems Development Items

16. This IS department aspires to the software development practices of the Software Engineering Institute's (SEI's) capability maturity model for software development.
17. Whether your IS department aspires to SEI CMM practices or not, at what level would your IS organization be assessed? Initial (Level 1) Repeatable (Level 2) Defined (Level 3) Managed (Level 4) Optimizing (Level 5)

The CMM(I) software process improvement method includes recommended or best practices grouped into several critical process areas that increase or enhance the capability of software processes (Paulk, Curtis, Chrissis, & Weber, 1993). The CMM(I) can be used by organizations to create an improvement strategy based on the CMM(I) maturity level. Many research studies have indicated positive outcomes due to increased maturity levels as measured by the CMM(I) (Bohn, 1995; Dion, 1993; Fenton & Neil, 1999; Glass, 1999; Harter & Slaughter, 2000; Jones 2003 & 2008; Ryan, 2000;

Zahran, 1998). As discussed previously in chapter 2, the CMM(I) is the most popular software process improvement method in the United States and is widely used (Beecham, Hall, & Rainer, 2005; El Emam & Madhavji, 1995). Also, Rogoway (1998) declares the CMM a *de facto* standard regarding process improvement assessments. The 12 items used in this survey were taken from a previous survey (which used an earlier version of the SIM membership listing) by Kappelman (1997). Using these items also provides an opportunity to longitudinally examine the information systems development capabilities and this instrument to assess changes over time.

Requirements Analysis and Design (RAD) Capabilities Construct

The development of the instrumentation to measure the RAD capabilities was adopted from existing enterprise architecture maturity models as well as being derived from previous research studies and practitioner expertise from the SIMEAWG (Tables 7 and 8). As discussed in chapter 2, enterprise architecture provides the context for this study's conceptualization of requirements. Enterprise architecture provides a means for requirements to align with the goals of the entire organization and not be overly focused on the technology and technical solutions (Hay, 2003). Also, with the basis of requirements being the essence regarding information systems development (Brooks, 1995), enterprise architecture's influence on requirements significantly influences IS development (Tang, Han, & Chen, 2004).

Table 7

Requirements Analysis and Design Capabilities -- Practices Instrument

18. Please select the level to which you agree or disagree that each of the following statements are representative of the requirements analysis and design practices in your IT organization. Definition: The purpose of requirements analysis and design (RA&D) is to describe a functional process or a product/service in order to achieve enterprise objectives. My organization's requirements analysis and design (RA&D) efforts and activities: ___
a. are measured.
b. are benchmarked to other organizations.
c. are aligned with the organization's objectives.
d. are highly developed and disciplined.
e. are valued by executive leadership.
f. have executive leadership buy-in and support.
g. are characterized by effective communication between executive leadership and the requirements analysis and design team.
h. describe our present 'as is' environment.
i. describe our "to be" or desired environment.
j. efforts not stifle innovation in our organization. (*reverse coded)
k. are viewed strictly as an IT initiative.
l. improve ability to manage risk.
m. contribute directly to the goals and objectives of our business plan.
n. have IT leadership buy
o. are well prioritized by executive leadership.

Table 8

Requirements Analysis and Design Capabilities-Artifacts Instrument

21. The outcomes or products of my organization's requirements analysis and design (RA&D) activities: ________________________
a. include standards for information systems security.
b. describe our transition from 'as is' to 'to be'.
c. are kept current.
d. are kept in a digital repository or database.
e. are used to standardize our technologies.
f. are used to support strategic business decisions.
g. are approved by the CIO.
h. are approved by the owner of the relevant business processes.
i. are used as the basis for IT procurement.
j. are assessed for their quality.

The use of EA maturity models was integral to the formulation and structure of the RAD instrument in the survey. In developing the survey, all of the available EA maturity models were reviewed, with a core of four EA maturity models used. The survey's RAD questions were mapped to various levels of these four EA maturity models to ensure sufficient coverage. Additionally, the key IT and business alignment enablers and inhibitors posited by Luftman and McLean (2004) were integrated into the survey and the mapping to reflect the importance of alignment to EA, RAD, and IS development, and vice-versa, and to reflect this persistent concern of top IT management (Table 9).

The four maturity models chosen (and shown in Table 9) are widely recognized and used but differentiated enough to provide separate perspectives on aspects of enterprise architecture activities. Each of the EA maturity models is discussed below:

> 1. The Government Accountability Office (GAO) framework for assessing and improving EA management. EA practices have been mandated in the Federal arena since the Information Technology Management Reform Act (also known as the Clinger-Cohen Act) of 1996 required agency CIOs to develop, maintain, and facilitate integrated systems architectures. This was formalized within the Federal EA Framework (FEAF) first published, and later updated, in 1999. The GAO maturity model uses five maturity stages (Creating EA Awareness, Building the EA Management Foundation, Developing the EA, Completing the EA, and Leveraging the EA to Manage Change) each with the same four critical success factors of demonstrates commitment, provides capability to meet commitment, demonstrates satisfaction of commitment, and verifies satisfaction of commitment. These are further categorized into four groups of architecture-related activities, products, events, and structures: architecture governance, content, use, and measurement.
>
> 2. Carnegie Mellon's Software Engineering Institute's capability maturity model (SEI CMM) is probably the best known maturity model in the IT arena and is used as a foundation for developing many other maturity models in a variety of applications and approaches. This capability maturity model is not EA-directed per se, but rather focused on the maturity of system development centering on project management and

software engineering practices. It is based on five increasing levels of maturity (Initial, Repeatable, Defined, Managed, and Optimizing). Various more specific instances of the CMM (such as the Systems Engineering CMM, Software Acquisition CMM, People CMM, and CMM Integration) have been developed by the SEI.

3. The Federal Office of Management and Budget (OMB) is mandated by the Clinger-Cohen Act of 1996 to annually assess the maturity of EA in Federal agencies. In response, the OMB has developed an EA assessment framework. The OMB EA Assessment Framework consists of five levels of maturity (initial, managed, utilized, results-oriented, and optimized) each with three capability assessment areas of completion, use, and results.

4. MIT's Center for Information Systems Research, based on qualitative research and a survey of 456 organizations, developed an EA maturity model comprising four stages (Ross, Weill, & Robertson, 2006). These four stages of enterprise architecture consist of Business Silos Architecture, Standardized Technology Architecture, Optimized Core Architecture, and Business Modularity Architecture. This maturity model provides "a number of lessons to companies attempting to generate more value from IT and implement greater process discipline" (Ross, Weill, & Robertson, 2006, p. 88).

Table 9

Mapping of, Requirements Analysis and Design Capabilities Instrument Into Maturity Models and Alignment Enablers

My organization's requirements analysis and design (RA&D) efforts and activities:	**CMM**	**MIT**	**GAO**	**OMB**	**Luftman**
18a. are measured.	Level 3 Defined	4	3	Results Capability Area Level 5	
18b. are benchmarked to other organizations.	4		3		
18c. are aligned with the organization's objectives.	Level 4 Managed	3	5	Use Capability Area Level 5	
18d. are highly developed and disciplined.	Level 5 Optimizing	3 5		Completion Capability Area Level 5	
18e. are valued by executive leadership.	Level 5 Optimizing	3		Use Capability Area Level 5 (1.4.1.1)	Enabler/Inhibitor
18f. have executive leadership buy-in and support.	Level 4 Managed	3	5	Use Capability Area Level 1 (1.4.1.1)	Enabler/Inhibitor
18g. are characterized by effective communication between executive leadership and the requirements analysis and design team.	Level 3 Defined	3		Completion Capability Area Level 5 (1.3.5)	Enabler/Inhibitor
18h. describe our present 'as is' condition.	Level 2 Under Development	2 3-4		Completion Capability Area Level 2 (1.3.5)	
18i. describe our "to be" or desired condition.	Level 3 Defined	2	3-4	Completion Capability Area Level 3 (1.3.6)	
18j. efforts stifle innovation in our organization.					
18k. are viewed strictly as an IT initiative.	1	-2		Completion Capability Area Level 1 (1.3.5)	Enabler/Inhibitor
18l. improve ability to manage risk.	Level 3 Defined	2	3		
18m. contribute directly to the goals and objectives of our business plan..	Level 5 Optimized	2 4-5		Completion Capability Area Level 4-5 (1.3.5)	Enabler/Inhibitor
18n. have IT leadership buy-in and support.	Level 2 Under Development	2 3		Use Capability Area Level 5 (1.4.1.1)	Enabler/Inhibitor
18o. are well prioritized by executive leadership..	Level 5 Optimized	3 5		Completion Capability Area Level 5	Enabler/Inhibitor

(table continues)

Table 9 *(continues)*

21. The outcomes or products of my organization's requirements analysis and design (RA&D) activities:	**CMM**	**MIT**	**GAO**	**OMB**	**Luftman**
21a. include standards for information systems security.	3		4	Completion Level 2 (1.3.5)	
21b. describe our transition from 'as is' to 'to be'.	Level 3 Defined		3	Completion Level 2 (1.4.2)	
21c. are kept current.	Level 2 Under Development	3		Use Capability Level 4	
21d. are kept in a digital repository or database.	Level 2 Under Development	3		Use Capability Level 2	
21e. are used to standardize our technologies.	Level 3 Defined	2	5	Completion Capability Area Level 4 (1.3.5)	
21f. are used to support strategic business decisions.	Level 5 Optimized	4 5		Results Capability Area Level 4-5 (1.5.1)	Enabler/Inhibitor
21g. are approved by the CIO.	Level 2 Under Development	2 4		Use Capability Area Level 3 (1.4.1)	Enabler/Inhibitor
21h. are approved by the owner of the relevant business processes.	Level 2 Under Development	2 4		Use Capability Area Level 3 (1.4.1)	Enabler/Inhibitor
21i. are used as the basis for IT procurement.	Level 3 Defined	3	5	Use Capability Area Level 4 (1.4.5)	
21j. are assessed for their quality.	Level 3 Defined	3	5		

Other maturity models, many of which are based on the four mentioned above, were used in developing the SIMEAWG survey. Some of these include the National Association for State CIOs (NASCIO) maturity model, which is based on Carnegie-Mellon SEI's CMM. It has 6 levels (no program, informal program, repeatable program, well-defined program, managed program, and continuously improving vital program) which also loosely align with the GAO's maturity model. The Institute for EA

Developments has a six stage maturity model including the following stages: no extended EA, initial, under development, defined, managed, and optimized.

Within the survey, RAD capabilities were divided into two groups of questions, one of which addresses RAD practices (18a-18o) and the other of which addresses RAD artifacts or outcomes (21a-21j).

Enterprise Architecture Perceptions Construct

The key criteria and critical success factors included in the EA maturity models described above were essential in developing the EA perceptions construct (see items in Tables 32 and 33). These factors derived from the various EA maturity models both from academia and practice (discussed above) are appropriate in forming EA perception survey items since they have been developed over many years for use in assessing the stages of EA maturity or progress within different organizations' EA initiatives. With these items as a foundation, the expert opinions of the members of the SIMEAWG were used within the structure of a modified Delphi study to further refine and reduce the set of EA items to the final number used in the survey.

Survey Development

With the influences described in the previous paragraphs, an initial survey was developed with over 120 questions or items. At this stage, a modified Delphi study approach was used with an expert group of EA professionals from industry and academia providing insight and recommendations in order to refine the survey. The use of an expert group in a modified Delphi study provided many benefits to the final survey instrument, the method of survey distribution, and helped to ensure practical relevancy to the outcomes. Beecham, Hall, Britton, Cottee, & Rainer (2005) used a group of software process improvement and requirements experts to develop and validate a requirements process improvement model. Their research established a research approach in which an expert group was used to combine best practices from multiple models and sources to form a cohesive, replicable measurement instrument.

Using a variant or modification of the Delphi method is common in IS research (Kappelman et al., 2006; Okoli & Pawlowski, 2004; Schmidt, 1997). One example includes using a "ranking-type" Delphi method to rank the applicable issues and validate them (Schmidt, Lyytinen, Keil, & Cule, 2001). The Delphi method was developed at the Rand Corporation in the 1950s. Generally, the Delphi method is an approach using iterative loops in order to reach a consensus among a group of experts to a certain problem. Common characteristics include the formation of a group communication process that facilitates the ability of a group of experts to understand and deal with complex problems. Feedback on the contributions of individuals as well as visibility into the group's consensus-forming view are also common characteristics of Delphi studies. Finally, an ability for individuals to revise/review their contributions as well as to provide some level of anonymity are also exhibited in Delphi research (Okoli & Pawlowski, 2004).

In its original form, the experts would go through the repeated questions via an iterative process with questionnaires and they would be anonymous to each other. The Delphi method is thought to be more supportive of independent thought and it avoids disadvantages of using expert groups such as confrontations that lead to narrow-mindedness, unjustified defense of individual's positions, or a predisposition to be swayed in opinion by a persuasive presentation (Dalkey & Helmer, 1963). The ultimate objective of a Delphi study is to achieve the most reliable agreement of opinion from an experts group (Dalkey & Helmer, 1963). It is not imperative for the individuals within a Delphi study to be representative of the population, rather it is the appropriate expertise and depth of knowledge that is important for the individuals participating in the expert group (Okoli & Pawlowski, 2004). The expert group making up the SIMEAWG were drawn from academia and both public and private sector practitioners.

A variation of Schmidt's (1997) three-phase model of a modified Delphi study was used. The three phases are: Phase 1: Brainstorming—consisting of discovering and developing the relevant issues; Phase 2: Narrowing Down—determining the most important issues; and Phase 3: Ranking the issues. This modified approach to using the Delphi method used in this research consisted of repeated e-mailing of variations of the survey to the experts to elicit opinions and then following up with them in onsite working group sessions. The final iterative loop in this modified Delphi method consisted of the experts actually taking the survey in its final, on-line form. Final changes were made to the survey based on this closing feedback loop. Determining the number of rounds to use in a Delphi study is an important function for the researcher (Schmidt, 1997). In this research study, five rounds were conducted. This was deemed adequate in order to arrive at a greater consensus on the content and wording of the items as well as the overall length (determined by number of questions) and structure of the survey.

The requirements analysis and design practices and outcomes questions reflected the expert group's determination that questions regarding requirements-related practices could serve as a surrogate for certain fundamental EA capabilities and practices. The survey was sent out as a pilot test to the SIMEAWG with the intent to develop final recommendations and revisions. At a follow up meeting, the survey instrument was further refined and the final modifications were agreed upon with the changes being implemented to the survey shortly thereafter. One of the goals of the SIMEAWG was a final survey which would take only 10 to 15 minutes to complete. With this in mind, the questions were winnowed to the final total of 80 questions of which 14 are demographic questions. The questions are 5-point Likert-type scales anchored with 1 *strongly disagree* to 5 *strongly agree*, 3 being *Neutral*. "Don't know" type options were provided as well.

Survey Execution/Data Collection

Inquisite survey software was used to create and distribute the final survey. The survey was distributed to the SIM membership mailing list. The SIM membership list was input to the Inquisite survey application which could then electronically distribute the survey. Completed survey results are stored within the Inquisite application located on a server at the Institutional Research Center at the University of North Texas. The results were stored in a comma separated file for later conversion to a Microsoft Excel spreadsheet for analysis by statistical software. The survey was distributed entirely by e-mail with each respondent receiving a personalized e-mail with an embedded individual hyperlink with which to connect to the Web server hosting the survey. Each message with embedded hyperlink was authenticated at the server when a respondent clicked on the link, so each hyperlink cannot be "used" by more than one respondent, nor can a respondent complete more than one survey. Respondent anonymity is assured in the introduction letter preceding the survey. All respondents had to respond affirmatively to an obligatory informed consent notice consisting of the title of the survey, the survey researchers, purpose of the study, procedures for maintaining the confidentiality of the research records, and the approval statement from the University of North Texas Institutional Review Board in order to proceed to the actual survey. A statement of research participants' rights was also provided to each potential respondent of the survey Web site.

In an attempt to gain maximum participation, a survey distribution method advanced by Dillman (2000) was followed. Further, to encourage participation and accurate responses, each potential participant had an option to include an e-mail address to receive a report of the preliminary research findings. They can then compare their organization to the entire sample and industry.

Data Analysis

It is important for the methodology to consider measures to ensure the accuracy or validity of the instruments comprising the survey. To promote content validity, (ensuring the survey instruments address the subject or material in which they are intended to cover) the items were derived from previous research and existing maturity models. Moreover, the experts group within the SIMEAWG was utilized through the modified Delphi study and the pretest to ensure the targeted research subjects and content were indeed covered (Kerlinger & Lee, 2000). As feedback was gathered, the survey in turn was modified to reflect the input of the experts group prior to administration to the full SIM membership list.

Construct validity is concerned with the relationship between a construct and the related measurement instrument. Construct validity consists of convergent and discriminant validity. Convergent validity is where evidence from different sources and methods has similar meanings (Kerlinger & Lee, 2000). Divergent validity is where it is possible to discriminate between different constructs. An initial step in assessing construct validity of a survey instrument is to identify the unidimensionality of the measures of the constructs (Sethi & Carraher, 1993). But it is also important to consider that unidimensionality "is necessary but not sufficient for construct validity" (Gerbing & Anderson, 1988, p. 191). Conducting an exploratory factor analysis at the onset of data analysis is a useful technique to understanding the dimensionality of the constructs used (Churchill, 1979). An exploratory factor analysis is a good preliminary method for this, but should be followed up with a confirmatory factor analysis (Gerbing & Anderson, 1988). Adequate factor loading as well as considering the eigenvalues of the factors will help in determining the nature of construct validity. One rule of thumb regarding factor analysis is that to retain each item it should have a factor loading $\geq$ 0.50 on the applicable factor (Hair, Black, Babin, Anderson, & Tatham, 2006). An alternative view is posited by Tabachnick & Fidell (2007) who use 0.32 as their rule of thumb cut off for interpretable factor loadings.

Further, convergent validity can be assessed from the goodness-of-fit results from the measurement model derived from the results of structural equation modeling (discussed below) while discriminant validity can be assessed by performing a chi-square difference test on the values obtained from the SEM output (Anderson & Gerbing, 1988). Finally, the use of the Delphi method can contribute to construct validity via the iterative process of identifying and defining key constructs and refining survey questions, as was done in this study (Okoli &Pawlowski, 2004).

The external validity or representativeness of findings is an important goal for this research. The use of the membership list of the Society for Information Management provides a basis to the external validity of the findings of this research because of the dispersion of the member demographics in this population in terms of organizational type, size, and budget, as well as individual education, job titles, and experience. Also, the name of the survey was generically titled “SIM Information Management Practices Survey” in part to ensure wide participation and so potential respondents would not proceed or decline to participate based on preconceived notions of the content or aim of the survey. For example, if the survey had IS development or enterprise architecture in the title or subject line this may have encouraged certain substrata of the SIM list while discouraging other segments. Furthermore, this same mailing list has been used in previous research with generalizable findings such as the longitudinal studies of Jerry Luftman (e.g., Luftman, Kempaiah, & Nash, 2005) that identify the top critical issues for IT executives. Also, Kappelman (1997) used the same membership list for his study of the Y2K problem.

Even with valid instruments, the reliability must be assessed. In assessing the reliability or consistency of the instruments, Cronbach’s alpha can be used. Cronbach’s coefficient alpha is the most widely used reliability measure (Churchill, 1979). Generally, a minimum Cronbach’s alpha of 0.70 is desired (Hair et al., 2006).

The research question investigating the relationship between information systems development capabilities and requirements analysis and design practices and the testing of the hypotheses was addressed by analyzing the data from the survey. This analysis began with a factor analysis. Factor analysis is a statistical technique applied to the set of questions in this survey in order to derive the factors that represent underlying constructs from patterns of correlations among the questions (Tabachnick & Fidell, 2007). Factor analysis assists with providing insight into the structure and interrelationship of the questions and constructs in this study (Hair et al., 2006). Additionally, this factor analysis is especially helpful to assess the strength, usability, validity, and applicability of the derived requirements analysis and design instrument which was derived from multiple sources. Moreover, factor analysis can indicate the level of continued strength and applicability of the information systems development capabilities instrument which utilized existing SEI CMM(I) questions. Since the requirements analysis and design instrument and its questions have been influenced by enterprise architecture research and maturity models, a factor analysis can also provide insight into the relationship of the EA-informed questions to the requirements factor and indicate variables to consider deleting. To determine the number of factors to keep, the conceptual foundation of each of the constructs, based on theory in previous literature, was the central guide. Additionally, a scree test in conjunction with identifying factors with latent roots/eigenvalues > 1 was used. An orthogonal rotation of factors (varimax) was also done. Orthogonal rotation is generally preferred when there is a desire to simplify the factors by maximizing each of the loadings' variance across the variables while it is also more conducive to the interpretation of the factors

(Tabachnick & Fidell, 2007). Also, orthogonal rotations are the most commonly employed rotation method (Hair et al., 2006). Generally, when looking at the resulting factor loadings, ranges between 0.30 and 0.40 are considered the minimum acceptable level for interpreting the structure, loadings $\geq$ 0.50 are considered practically significant, and those $\geq$ 0.70 indicate a well-defined structure (Hair et al., 2006).

To further analyze the data from the survey and to statistically test the significance of the hypothesized model (based on the exploratory factory analysis), structural equation modeling (SEM) was performed. SEM was used to depict and test the relationships that were hypothesized in the research models (Figures 3 and 4). SEM tests the structure and relationships of the constructs (i.e., information systems development, requirements analysis and design capabilities, and EA perceptions) that have been hypothesized and developed in this research study. The results of SEM will indicate whether the data from the survey support the research models.

In evaluating and assessing the SEM output such as the validity of the measurement model, various combinations of model fit are used. There are many goodness-of-fit indices which can be used in SEM. Hair et al. (2006) provides a useful framework to classify some of the more popular ones. However, this short description of selected fit indices is not comprehensive. The first category is absolute fit measures. Absolute fit measures provide a direct and basic assessment of how well the model specified by the researcher fits the survey data (Hair et al., 2006). This category consists of such fit indices as the X^2 statistic, the goodness-of-fit index (GFI), the root means square residual (RMSR), the standardized root mean residual (SRMR), and the root mean square error of approximation (RMSEA). The second category is incremental fit indices. Incremental fit indices assess how well a specified model fits an alternative baseline or null model (Hair et al., 2006). Examples of incremental fit indices are the normed fit index (NFI), the comparative fit index (CFI) and the relative noncentrality index (RNI). Finally, the third category of Hair et al. (2006) is the parsimony fit indices. This category of indices is intended to provide an assessment of the best model among a set of competing models. Indices included in this category include the parsimony ratio (PR), the parsimony goodness-of-fit index (PGFI), and the parsimony normed fit index (PNFI).

There is no set standard as to which of the goodness-of-fit indices should be used, it is a matter of personal (or journal editor) preference (Tabachnick & Fidell, 2007). Hair et al. (2006) suggests using multiple indices of differing types, adjusting the index cutoff based on the characteristics of the model, using indices in the comparison of models, and being cautious of trying to achieve better fit at the expense of testing a true model. Alternatively, Tabachnick and Fidell (2007) mention the popularity of reporting the CFI and RMSEA and also highlight the recommendations of Hu and Bentler (1999) to report two types of fit indices, the SRMR and a comparative fit index.

Summary

This chapter presented a description and adequacy of the population used in this research as well as the research strategy, design, and data analysis used to analyze the data and examine and test the research question and hypotheses. The survey execution plan was also provided. Moreover, the information systems development capabilities and requirements analysis and design capabilities instruments were described as well as the aspects of validity and reliability regarding them. The results obtained from the methodological plan presented in this chapter are next presented in Chapter 4.

CHAPTER 4

DATA ANALYSIS AND RESULTS

Introduction

This study investigates the relationship of information systems development (ISD) and requirements analysis and design (RAD) capabilities. Additionally, this relationship is examined within the context of enterprise architecture (EA) perceptions. Two research models were hypothesized to examine these relationships. To accomplish this goal, an electronic survey of information technology (IT) professionals was conducted. This chapter presents the results of statistical analyses of the data garnered from the respondents of this survey. First, this chapter describes the nature of the survey and presents data on the demographic makeup of the survey's population. Secondly, the results of exploratory factor analysis on the observed variables of ISD, RAD, and EA is examined followed by an examination of the results of structural equation modeling analyses on both the primary and the secondary research models. Finally, the results of ANOVA multiple comparison procedures on the variables of organization size and IT budget are presented.

Data Collection

The Society for Information Management (SIM) provided an e-mail list of their membership. The e-mail listing was integrated with the survey software on a Web server in order to form the survey package (consisting of the email notification, cover letter and on-line survey). The survey software then electronically distributed by email the survey invitations as well as the follow-up reminders. The survey software would only send the reminder e-mail messages to those who had not yet completed the survey, thus not sending unnecessary and redundant reminders to those who had already completed the survey. Survey invitation e-mails were sent out with each potential respondent receiving a personalized e-mail with an embedded individual hyperlink with which to connect to the survey Web server. The individual messages with the embedded hyperlink to the survey Web server, were authenticated at the server, so that the hyperlinks could only be used by one respondent. Moreover, a respondent could only complete the survey once. A total of 2863 survey invitations were sent, with 376 quality responses after data purification (responses were removed where over 10% of the questions were not answered, reversed items were used to validate response quality, and consistency checks utilized with duplicative questions). This 13% response rate is consistent with other surveys of the SIM membership including the annual SIM member surveys and the SIM Y2K Working Group conducted in 1996, 1997, 1998, and 1999. A final version of the survey instrument is at the Appendix.

Demographics

The survey consisted of 16 demographic items. Demographically, the responses reveal the average age of the respondents was about 48 years, with each respondent having an average of 8.16 years in their organization, and averaging 4.37 years in their current position (Table 10). The level of responsibility of the survey respondents was generally broad-based, with almost 80% reporting their responsibility at either an enterprise (63.4%) or a business-unit/divisional (15.92%) level (Table 11). The survey respondents were generally well-educated, with close to 90% attaining a bachelor’s or master’s degree (Table 12).

Table 10

Age, Time in Organization, and Time in Present Position

	Number	Minimum	Maximum	Mean	Std. Deviation
3. Age	370	29	73	47.69	8.55
4. Years in Org	376	0	35	8.16	7.50
5. Years in Pos	375	0	35	4.37	4.26

Table 11

Level of Responsibility

7. Responsibility	Frequency	Percent
Department 59		15.65
Division/Business Unit	60	15.92
Enterprise-wide 239		63.40
Team/Workgroup 17		4.51
Not Answered	1	0.27
Total 376		100.00

Table 12

Highest Level of Education Attained

6. Educational Level	Frequency	Percent
Associate Degree	12	3.18
Bachelor's Degree	140	37.14
Master's Degree	198	52.52
High School	1	0.27
PhD, JD, MD, or other terminal degree 16		4.24
Some College	8	2.12
Not Answered	1	0.27
Total 376		100.00

The vast majority of respondents were with profit-oriented organizations (80.64%), with 13.53% not-for-profit, and 4.51% representing the government sector (Table 13). The majority of the respondents' organizations were in the United States (86.21%) as seen in Table 14. A breakdown of the industry type reveals about 13% were in the banking/financial/insurance industry, with close to 12% in the healthcare/biotechnology industry and eight percent in the retail/wholesale business. Only 4.8% were involved in the education industry (Table 15).

Table 13

Organizational Description

9. Organization Type	Frequency	Percent
Governmental organization	17	4.51
Not-for-profit organization	51	13.53
Profit-making corporation	304	80.64
Other 3		0.81
Not Answered	1	0.27
Total 376		100.00

Table 14

Primary Location of Organization

12. Location	Frequency	Percent
Africa 1		0.27
Asia 3		0.80
Europe 8		2.12
North America (other than USA) 29		7.69
South America	1	0.27
United States only	325	86.21
Not answered	9	2.39
Total 376		100.00

Table 15

Type of Industry

8. Industry	Frequency	Percent
Agriculture 1		0.27
Banking/Securities/Investments/Finance/Insurance	48 12.73	
Business Services (Legal/R&D)	8	2.12
Capital Goods Mfg	14	3.71
Chemical 8		2.12
Construction/Engineering 3		0.80
Consumer Goods Mfg	27	7.16
Education 18		4.77
Entertainment 4		1.06
Food Service	6	1.59
Government (Fed, State, Local)	14	3.71
Healthcare/Medical/Pharmaceutical/Biotech 42		11.14
Hotels/Tourism/Travel 3		0.80
IT Services Provider/Consultant	33	8.75
Military 2		0.53
Other 68		18.09
Mining/Energy 8		2.12
Printing/Publishing 6		1.59
Real Estate	5	1.33
Retail/Wholesale 32		8.49
Transportation/Distribution/Logistical 14		3.71
Utilities 10		2.65
Not answered	2	0.53
Total 376		100.00

Gross revenues (Table 16) of responding organizations were under $100 million for about 18%, another 31.9% had revenues up to $1 billion, 35.4% of their revenues ranged from $1 to 9.9 billion, while 18% had revenues of $10 billion or greater. The IT budgets of responding organizations (Table 17) consisted of 12.6% that were under $1 million, 36.3% from $1 to 9.9 million, 31.8% from $10 to 99.9 million, and 16% had IT budgets of $100 million or more. These revenue streams and budgets were associated with organizations that were predominantly smaller than 10,000 employees (about 73%, Table 18) while the number of employees in the IT departments were mostly below

5000 (94%, Table 19). The Job Title Demographic Table (Table 20) shows the job titles of most of the respondents either being director (32%), chief information officer (27%), or some other title than what was listed (21%).

Table 16

Gross Revenue/Income/Budget of Organization

11. Range in Dollars	Frequency	Percent
Less than $50 million	49	13.00
$50 - $100 million	19	5.04
$101 - $500 million	79	20.95
$501 - $999 million	41	10.88
$1billion - $4.9 billion	86	22.81
$5 billion - $9.9 billion	25	6.63
$10 billion - $14.9 billion 13		3.45
$15 billion - $24.9 billion 17		4.05
$25 billion - $50 billion	19	5.04
Greater than $50 billion 10		2.65
Don't Know	12	3.18
Not answered	6	1.59
Total 376		100.00

Table 17

Operating Budget of IT Department

13. Range in Dollars	Frequency	Percent
Less than $100, 000	8	2.12
$100,000 - $249,999	4	1.06
$250,000 - $499,999	17	4.51
$500,000 - $999,999	21	5.57
$1 million - $9.9 million	137	36.34
$10 million - $49.9 million 100		26.53
$50 million - $ 99.9 million	20	5.31
$100 million - $499.9 million 40		10.61
$500 million - $1 billion	10	2.65
Greater than $1 billion	10	2.65
Not answered	9	2.39
Total 376		100.00

Table 18

Number of Employees in Organization

10. Number	Frequency	Percent
Less than 100	35	9.28
100-499 69		18.30
500-999 40		10.61
1000-4999 81		21.49
5000-9999 50		13.26
10,000-19,999 33		8.75
20,000-29,999 14		3.71
30,000-50,000 20		5.31
Greater than 50,000	31	8.22
Don't Know	3	0.80
Total 376		100.00

Table 19

Number of Employees in IT Department

14. Number	Frequency	Percent
Less than 50	147	38.99
50-99 49		13.00
100-499 92		24.40
500-999 33		8.75
1000-4999 35		9.28
5000-9999 8 2.12		
10000-19999 2		0.53
Greater than 30,000	6	1.59
Not answered	4	1.06
Total 376		100.00

Table 20

Respondent Job Title

1. Job Title	Frequency	Percent
Chief Executive Officer (CEO) 8		2.13
Chief Information Officer (CIO) 100		26.60
Chief Technology Officer (CTO) 13		3.46
Director 121		32.18
Enterprise Architect	16	4.26
Member of the Board	1	0.27
Vice President	38	10.12
Other 79		21.01
Total 376		100.00

Nonresponse Bias

To assess nonresponse bias, the responses before and after the date of 8 October 2007 were used. This data was chosen for several reasons. First of all, the initial mailing and first reminder notice were sent out before this date while two additional reminders were sent after this date. Also, there was a definitive gap of eight days before 8 October 2007 from the last completed survey (it was on 30 September 2007). Finally, at the annual meeting of the Society for Information Management (SIMposium), the survey was marketed both at a conference booth and during enterprise architecture presentations. This “marketing blitz” was believed to have been a catalyst for survey responses. The number of usable surveys in the “early responses” group was 204 respondents while the “late responses” group contained about 172 respondents.

A Levene's test was performed to test for the important assumption of equality of variances. A nonsignificant result indicates equal variances. Several demographic variables were chosen as well as the means from the three central constructs of the research (information systems development, requirements, and enterprise architecture) were chosen for the groups. From the results in Table 21, of eight variables tested, years in organization was the only variable indicating both unequal variances as well as a significant difference in means (at the .05 level) between the two groups of early and late respondents (a t-test significance of .040). However, as seen in the Table, several closely related demographic variables (years in organization and years in position) did not have significant differences in means. This fact and that there is no theoretical basis determined for this result and that the t-test is close to the .05 cutoff (i.e., 0.04) leads to the conclusion that this particular result does not provide sufficient evidence for a response bias in the population.

Table 21

Results of t-tests for Nonresponse Bias

Item	Levene's Test for Equality of Variances			T-Test for Equality of Means	
	F statistic	Sig.	df	Sig.	t statistic
Age	1.047	.307	368	.233	-1.196
Years in Org	9.578	.002	372.228	.040*	2.059*
Years in Position	.445	.505	373	.851	-.188
Org income/rev	.066	.798	374	.126	1.534
Org size	.793	.374	374	.153	1.433
#17 CMM Rating	.277	.599	363	.319	.999
RAD #18 & #21	.612	.435	372	.372	.894
EA #19 &#20	.719	.397	372	.246	-1.161

* Unequal variances not assumed

Exploratory Analysis

A principal components factor analysis was performed on each of the constructs in order to enhance the understanding of the underlying order including unidimensionality and data structure as well as for data reduction considerations. Moreover, output from the factor analysis can reveal indications of various types of validity, as will be discussed below. The Kaiser-Meyer-Olkin Measure of Sampling Adequacy (KMO MSA) was run in order to assess the suitability of the data for a factor analysis. As shown in Table 22, the KMO MSA value of 0.885 is well above thresholds recommended to proceed with factor analysis (Hair et al., 2006; Tabachnick & Fidell, 2007). This result provides adequate justification of the suitability of the data for the performance of a principal components factor analysis. For each factor analysis performed, the decision as to whether or not to keep an item is based on theory, practical experience, and expertise from the SIMEAWG. For example, both Hair et al. (2006) and Tabachnick and Fidell (2007) recommend that factor loadings of 0.50 or better are generally considered the minimal threshold for practical significance.

Table 22

KMO (for entire data set)

Kaiser-Meyer-Olkin Measure of Sampling Adequacy.	0.885

In assessing construct validity, its components, convergent and discriminant validity, must be addressed. For convergent validity, all factor loadings should generally be above 0.50 (Hair et al., 2006) in order to demonstrate the items measure an individual latent factor or construct. For discriminant validity, the goal is to minimize the cross loading on items within a factor. Also, by using the SIMEAWG's membership in a modified-Delphi study through several SIMEAWG on-site sessions as well as subjects in a pilot test, face validity concerns were adequately addressed.

The factor analyses for each instrument were principal component factor analyses with a varimax orthogonal rotation. The varimax rotation was chosen because it is appropriate when the goal is to simplify the factors by maximizing the loadings' variance across the variables and it eases the interpretation of the factors (Tabachnick & Fidell, 2007). Orthogonal rotations are the most commonly employed methods and are preferred for data reduction purposes (Hair et al., 2006).

Information Systems Development Instrument

The results of the factor analysis on the information systems development instrument are shown in Table 23. As expected based on theory and the use of an existing and tested instrument, the ISD questions cleanly load on one factor with significant factor loadings. The KMO measure (Table 24) indicates acceptability of the data for factor analysis and the high loadings of the questions indicate construct validity. Reliability analysis (Table 34) indicates adequate reliability, with a Cronbach's alpha score of 0.9220, well above the generally agreed upon lower limit of 0.70 for this measure (Hair et al., 2006). The high factor loadings of the questions (all greater than 0.60) give clear indications of the unidimensionality of this factor. Adequate convergent validity is indicated as all the items possess loadings above 0.50 and the factors have eigenvalues of at least 1.0. The means and other descriptive statistics of each of the ISD items are in Table 25. Total variance explained is 54.232% with no items removed.

Table 23

ISD Factor Matrix

Q15c	0.7918
Q15g	0.7897
Q15d	0.7896
Q15l	0.7777
Q15f	0.7758
Q15e	0.7410
Q15h	0.7408
Q15j	0.7337
Q15b	0.7137
Q15a	0.7056
Q15i	0.6501
Q15k	0.6006

Table 24

KMO Test (ISD Factor)

Kaiser-Meyer-Olkin Measure of Sampling Adequacy.	.929

Table 25

ISD Descriptive Statistics

Survey Item	N Valid	Missing	Mean	Std. Deviation
15a Customer Agreement on Req	374	2	3.83	1.07
15b ID Trng needs	370	6	3.42	1.03
15c Est qual goals	374	2	3.55	1.03
15d Estimate resources	375	1	3.66	1.01
15e Track progress/resources 373		3	3.82	0.93
15f S/W QA	370	6	3.68	0.99
15g CPI	371	5	3.51	1.06
15h Coord/Comm w/ stkholders	375	1	3.90	0.92
15i Selecting/tracking s/w contractors 367		9	3.83	1.01
15j Analyze probs	372	4	3.75	0.95
15k Tailored process	368	8	3.76	0.92
15l Continuous prod improvements 371		5	3.47	0.98
Grand Means			3.68	0.99

Requirements Analysis and Design (RAD) Instrument

The factor analysis of the RAD instrument is at Table 26. Six items were removed due to low factor loadings or cross loadings on the two dimensions—RAD practices and outcomes. The percentage of total variance explained by these two factors is 48.90%. The results in which two factors were interpreted, are well grounded in the theoretical and practical foundation that went into developing this instrument, as explained in Chapters two and three. The RAD 2-factor solution matrix aligns with the overall survey's RAD items, where the items with question 18 were concerned with RAD efforts and activities while question 21 items were concerned with RAD outcomes or products. The first factor, the items within question 18, is labeled "RAD practices" to reflect organizational processes which are consistent with more mature practices and organizations. The second factor, the items within question 21, is labeled "RAD outcomes". These are artifacts which would be the result of the practices or processes within the first factor.

Once again, the KMO measure (0.902) (Table 27) indicates the acceptability of data for factor analysis and the high loadings and no cross loadings between the two RAD factors indicate construct validity. Moreover, unidimensionality is indicated with the low cross loadings and high factor loadings (all but two greater than 0.50) in this final matrix. The inclusion of two items with factor loadings between 0.45 and 0.50 is deemed acceptable since this was the first use of this new and preliminary instrument. Discriminant validity was demonstrated since none of the remaining items had any high cross loadings with any other items (using the criteria of 0.5). As with the ISD instrument, reliability analysis (Table 34) indicates adequate reliability with a Cronbach's alpha score of 0.9130 well above the generally agreed upon lower limit of 0.70 for this measure (Hair et al., 2006). Adequate convergent validity is indicated as all the items possess acceptable loadings and the factors have eigenvalues of at least 1.0. The means and other descriptive statistics of each of the RAD items are in Tables 28 and 29.

Table 26

RAD Practices Factor Matrix

	RAD-Prac	RAD-Out
Q18f	**0.7685**	0.2697
Q18g	**0.7560**	0.3174
Q18e	**0.7430**	0.2919
Q18o	**0.7419**	0.2040
Q18m	**0.7197**	0.2050
Q18c	**0.6999**	0.1809
Q18n	**0.6256**	0.0942
Q18l	**0.5483**	0.1774
Q18i	**0.4710**	0.3021
Q18k	**0.4554**	0.0117
Q21e	0.1529	**0.7659**
Q21c	0.1818	**0.7342**
Q21f	0.1965	**0.7111**
Q21j	0.2196	**0.6887**
Q21b	0.1661	**0.6866**
Q21a	0.1202	**0.6610**
Q21d	0.1548	**0.6453**
Q21i	0.1699	**0.6060**
Q21g	0.2315	**0.5494**
TVE:	24.661%	24.243%

Table 27

KMO Test (RAD Factor)

Kaiser-Meyer-Olkin Measure of Sampling Adequacy.	.902

Table 28

RAD Practices Descriptive Statistics

RAD Practices:	N Valid	Missing	Mean	Std. Deviation
18n RAD IT support	371	6	4.18	0.78
18c RAD aligned w/ org	373	4	3.90	0.84
18m RAD contribute to bus plan	370	7	3.78	0.82
18l RAD improve risk mgmt 367		10	3.61	0.81
18i RAD describe to be	367	10	3.60	0.91
18f RAD exec support	372	5	3.57	0.99
18h RAD describe as is	364	13	3.52	0.82
18e RAD valued by exec	372	5	3.34	1.04
18o RAD well prioritized	370	7	3.34	1.02
18g RAD effective comm	373	4	3.21	1.01
18k RAD IT only	368	9	3.03	1.06
18d RAD dev/disciplined 373		4	3.00	1.05
18a RAD measured	370	7	2.99	1.07
18j RAD stifle innovation 366		11	2.48	0.91
18b RAD benchmarked 368		9	2.36	0.99
Grand Means			3.33	0.94

Table 29

RAD Outputs Descriptive Statistics

RAD Outcomes:	N Valid	Missing	Mean	Std. Deviation
21h approved owner	361	16	3.84	0.91
21b describes transition	366	11	3.76	0.86
21g approved CIO	358	19	3.69	1.05
21f used for strategic	365	12	3.65	0.89
21i used for IT proc	363	14	3.60	0.94
21a includes stds	362	15	3.58	0.93
21e used to standardize	361	16	3.42	0.95
21d kept repository	362	15	3.37	1.02
21j qual assessment	362	15	3.28	0.96
21c kept current	364	13	3.11	0.95
Grand Means			3.53	0.95

Enterprise Architecture (EA) Instrument

The factor analysis conducted on the enterprise architecture items is in Table 30. From the results, three clear factors were derived. The Kaiser-Meyer-Olkin measure of sampling adequacy (0.923) indicates the data is adequate for factor analysis (Table 31). Based on the grouping and wording of the items, the factors were interpreted as: 1) EA and the Organization, 2) EA and IT, and 3) EA Synergies. The first two factors consisted only of EA benefits items from question 20 while the third significant factor consisted of items from questions 20 and 19 (purpose of EA) items. Adequate convergent validity is indicated as all the items possess loadings above 0.50 on the respective factors and the factors have eigenvalues of at least 1.0. Discriminant validity is demonstrated since none of the items had high cross-loadings with any other factors (using the criteria of 0.5). The percentage of total variance explained by these three factors is 54.971%.

Table 30

EA Factor Matrix

	EA-Org	EA-IT	EA-Synergies
Q20s	**0.7944**	0.0296	0.2774
Q20j	**0.7769**	0.0461	0.2668
Q20n	**0.7718**	0.2748	0.0413
Q20g	**0.6960**	0.1271	0.2933
Q20o	**0.6932**	0.3952	0.0370
Q20t	**0.6205**	0.2561	0.1626
Q20m	**0.5938**	0.3605	0.0977
Q20e	**0.5311**	0.3147	0.3986
Q20p	**0.5275**	0.3736	0.1734
Q20l	0.1764	**0.7235**	0.1094
Q20k	0.1793	**0.6915**	0.0588
Q20r	0.0969	**0.6660**	0.0676
Q20q	0.2146	**0.6500**	0.2648
Q20h	0.2060	**0.6334**	0.2156
Q20f	0.2175	**0.6102**	0.4310
Q20i	0.2280	**0.5925**	0.2656
Q19f	0.0150	0.1761	**0.8007**
Q19g	0.1271	0.1752	**0.7559**
Q19b	0.1842	0.1483	**0.6001**
Q20a	0.3682	0.0822	**0.5155**
Q20d	0.3735	0.2988	**0.5047**
TVE:	38.179%	8.832%	7.675%

Table 31

KMO Test (EA Factor)

Kaiser-Meyer-Olkin Measure of Sampling Adequacy.	.923

The EA-Organization factor is comprised of 9 items. The items within this factor reflect the potential of EA to benefit the broader organization domain and address organizational concerns. Examples of the items include: 20e “More effective at meeting *business* goals”, 20g “Improved *organizational* communications and information sharing”, and 20m “Standardizes *organizational* performance measures”.

The EA-IT factor consists of items that specifically highlight the potential benefits of EA to specific goals and objectives of the IT department rather than to the broader organization as a whole. Examples of the items include: 20h “Improved utilization of *information technology*”, 20k “Faster at developing and implementing new *information systems*”, and 20r “Reduced *IT* complexity”.

The final factor, EA-Synergies, reflects a group of items operationalizing the concept of EA as a facilitator of improvements to both the broader organization and IT, facilitating alignment or synergies between the two but also local optimizations. Examples of the items include: “*aligning business objectives with information technology investments*”, “as a tool for *aligning* business objectives with IT initiatives”, “as a tool for planning”, and “as a decision tool”. This factor appears to be relatively independent of a particular respondent’s view as to whether the scope of EA is just IT or whether, as the name suggests, EA is about the whole enterprise.

The factor analysis results are an initial indication of the construct validity of the EA construct. Six items were removed based on lower factor loadings. Additionally, these items are theoretically related as well. These three factors derived from the factor analysis are consistent with prior research such as a study by van der Raadt et al. (2004). Their research indicated three groups of organizations with EA programs: 1) organizations in which EA awareness started with the business, 2) organizations in which EA awareness started with IT, and 3) IT service providers.

As with the other two instruments, reliability analysis (Table 34) indicates adequate reliability, with a Cronbach's alpha score of 0.9180, well above the generally agreed upon lower limit of 0.70 for this measure (Hair et al., 2006).

Table 32

EA Purpose/Function Descriptive Statistics

EA Purpose/Function	N Valid	Missing	Mean	Std. Deviation
19e blueprint	373	4	4.40	0.68
19f planning tool	373	4	4.25	0.68
19g decision tool	374	3	4.13	0.71
19d alignment tool	372	5	4.08	0.86
19b facilitate change	374	3	4.02	0.74
19c comm tool	372	5	3.67	0.90
19a EA purpose/function: snapshot	374	3	2.87	1.10
Grand Means			3.92	0.81

Table 33

EA Benefits Descriptive Statistics

	N		Mean	Std.
Valid		Missing		Deviation
20h IS interoperability	372	5	4.34	0.72
20f Utilization of IT	373	4	4.25	0.70
20a Potential benefits EA: align bus w/IT	373	4	4.14	0.80
20q Effective use of IT	370	7	4.02	0.74
20b Responsive	371	6	3.97	0.80
20i IT ROI	371	6	3.96	0.85
20d Better sit aware	367	10	3.95	0.71
20p Org governance	370	7	3.93	0.77
20g Org comm	371	6	3.92	0.80
20l IS security	369	8	3.92	0.81
20c Less wasted time/effort	373	4	3.92	0.88
20e Effective bus goals	373	4	3.91	0.77
20j Comm between org and IS	369	8	3.82	0.88
20n Collaboration	371	6	3.82	0.79
20k Faster IS dev	369	8	3.80	0.83
20r Reduced IT complexity	367	10	3.78	1.02
20t Reduces stovepipes	365	12	3.78	0.84
20m Standardizes pfc measures 364		13	3.69	0.90
20s Improved org comm	368	9	3.67	0.82
20o Improves trust	368	9	3.49	0.85
Grand Means			3.90	0.81

Table 34

Reliability Analysis

Instrument	Cronbach's Alpha
ISD	0.9220
RAD	0.9130
EA	0.9180

Structural Equation Modeling

Structural equation modeling (SEM) is the confirmatory factor analysis method chosen to assess the research models and examine the relationships between ISD and

RAD capabilities as well as EA perceptions. LISREL is the statistical program used to perform the structural equation modeling.

Research Model 1

In the first research model (Figure 3), a two-factor model of requirements analysis and design capabilities is hypothesized. It is comprised of the latent factors of requirements analysis and design (RAD_Prac[tices] and RAD_Out[comes]). These two latent factors are defined by a set of six (Q18C, Q18E, Q18F, Q18G, Q18M, and Q18O) and nine (Q21A, Q21B, Q21C, Q21D, Q21E, Q21F, Q21G,Q21I, and Q21J) items, respectively. The 15 items were finalized based on the results of the exploratory factor analysis previously discussed and in relating these observed variables to their associated latent factor by analyzing the measurement model. For the latent construct of information systems development capabilities (ISD_Cap), 12 items (Q15A, Q15B, Q15C, Q15D, Q15E, Q15F, Q15G, Q15H, Q15I, Q15J, Q15K, and Q15L) adapted from the SEI CMM were used. The construct variances were set to the value of one in the design of this SEM study in order to set the scale of each of the latent constructs. In analyzing the measurement model, variables with standardized estimates of path loadings less than 0.50 were abandoned (general rule of thumb for path significance per Hair et al., 2006).

In conducting a preliminary evaluation of the appropriateness of the data set for SEM, Bagozzi and Yi (1988) recommend three primary evaluation criteria (preliminary fit criteria) that are useful indications of common anomalies which may indicate model specification errors. These anomalies are negative error variances, correlations greater than one, or extremely large parameter estimates. From examining the output in Figure 5, none of these anomalies are apparent.

A factor analysis was also performed on the final, item-reduced instruments for the constructs of IS development, requirements practices and outcomes, and enterprise architecture as described above (Table 35). While some of the individual items vary as to which factor they load on compared to their loadings previously discussed, this factor analysis performed with no forced factors does provide additional support for the overall structure of the six factors and the structure of the individual constructs (EA with three factors, RAD with two factors) within this study. The variation as to which factors the items loaded on and the number of items per factor is due to individual items being removed due to the strength of their factor loading or cross-loadings as well as due to theoretical concerns and practical expertise about the individual items. These adjustments were initiated as individual factor analyses on each of the constructs was conducted. The results of this factor analysis confirm the overall construct validity of the final factors and instruments used in this research study.

Table 35

Final Overall Research Model Constructs Factor Matrix

	IS Dev	RAD_Out	EA_Org	RAD_Prac	EA_IT	EA_Syn
Q15d	**0.7430**	0.1630	0.0530	0.2160	0.0400	0.0430
Q15g	**0.7380**	0.1780	0.0450	0.1260	0.0400	-0.2510
Q15l	**0.7320**	0.2030	0.0660	0.1270	0.0450	-0.1950
Q15f	**0.7320**	0.1900	0.0820	0.1180	0.0780	-0.0630
Q15b	**0.7270**	0.0850	0.0650	0.0630	0.0270	-0.0970
Q15c	**0.7220**	0.2100	0.1190	0.1750	0.0410	-0.0670
Q15e	**0.6920**	0.0980	0.0200	0.1340	0.0030	0.2050
Q15j	**0.6820**	0.1840	0.0490	0.1890	0.0880	0.0240
Q15a	**0.6490**	0.1130	0.0740	0.2520	0.0120	0.0140
Q15h	**0.6380**	0.1220	0.0420	0.3100	0.0600	0.0210
Q15i	**0.6110**	0.1890	-0.1310	0.1050	0.0780	0.0960
Q15k	**0.5830**	0.1750	0.0550	0.0360	0.0380	0.2050
Q21c	0.2730	**0.7090**	0.1160	0.1120	0.0180	-0.0290
Q21e	0.2280	**0.7000**	0.1090	0.1670	0.1220	-0.1280
Q21f	0.1780	**0.6810**	0.1980	0.1810	0.0390	-0.0220
Q21b	0.2560	**0.6270**	0.0360	0.0520	0.0620	0.2020
Q21d	0.1380	**0.6260**	0.0230	0.0780	0.0460	0.1130
Q21j	0.2840	**0.6030**	0.0610	0.1950	0.0580	0.1070
Q21a	0.2430	**0.5790**	0.0460	0.0800	0.1600	0.0220
Q21i	0.0460	**0.5680**	0.0900	0.1980	0.0850	0.1760
Q21g	0.0860	**0.5380**	0.0760	0.3040	0.0040	-0.1380
Q20j	0.1060	0.0960	**0.8340**	0.0130	0.1060	0.0400
Q20s	0.0600	0.1530	**0.8240**	-0.0010	0.0640	0.1070
Q20g	0.0140	0.0970	**0.7650**	0.0580	0.1770	0.0030
Q20n	0.0780	0.1110	**0.6770**	-0.0390	0.3180	-0.1170
Q20t	0.0170	0.1110	**0.6310**	0.0680	0.2420	0.1360
Q20a	0.0500	0.0630	**0.5620**	0.0170	0.1380	0.3010
Q20c	0.1510	0.0190	**0.4980**	0.0460	0.3450	0.1620
Q18f	0.1920	0.2630	0.0730	**0.8050**	0.0400	-0.0750
Q18e	0.2010	0.2450	0.1320	**0.7720**	0.0630	-0.0780
Q18g	0.3150	0.2680	0.0260	**0.7420**	0.0750	-0.0640
Q18o	0.2630	0.1590	0.0120	**0.7200**	0.0380	0.1010
Q18c	0.3590	0.1790	-0.1170	**0.6170**	0.0610	0.0650
Q18m	0.3460	0.1420	0.0060	**0.5960**	0.0640	0.1010

(table continues)

Table 35 *(continues)*

	IS Dev	RAD_Out	EA_Org	RAD_Prac	EA_IT	EA_Syn
Q20l	0.0030	0.1380	0.1230	-0.0450	**0.7600**	-0.0080
Q20k	0.1180	0.1180	0.0930	-0.1150	**0.6920**	0.0310
Q20h	0.0350	-0.0610	0.2890	0.0240	**0.6770**	-0.0350
Q20q	0.0140	0.0950	0.2760	0.0690	**0.6730**	0.0880
Q20i	0.0430	0.0640	0.2670	0.1270	**0.6230**	0.1250
Q20r	0.0860	0.0670	0.0770	-0.1020	**0.6060**	0.2680
Q19b	0.0980	0.0750	0.2470	0.0180	0.2250	**0.7010**
Q19g	0.0360	0.1840	0.3140	0.0040	0.2130	**0.5630**

After post hoc model modifications (removing lower path loadings and setting covariances based on the modifications indices (as shown in Figure 5), the final full structural model of the first research model (Figure 5) was derived. Reliability analysis of these reduced factors using Cronbach's alpha resulted in 0.922 for ISD, 0.890 for RAD practices, and 0.865 for RAD outcomes (Table 36). Figure 5 shows the final model with the standardized path loadings. By looking at the standardized estimates, it is apparent that the relationship between RAD practices and ISD capabilities is a stronger one than the relationship between RAD outcomes and ISD capabilities. As shown in Figure 6, the t-values between the latent factors in this model are significant, indicating significance of the hypothesized relationships among these factors. Table 36 shows descriptive statistics and reliability analysis of the final configuration of the instruments in this model. Table 37 shows several goodness-of-fit (GOF) indices characterizing this model.

In examining the goodness-of-fit indicators, a myriad of guidance is available. It is important to remember much debate continues to go on about various GOF cutoff values and one should remember that the theoretical foundation and motivation of the research should serve as the overall lodestar as the complex relationships of ISD, RAD, and EA are examined and when looking at whether or not these relationships are reflected in the data. As Marsh, Hau, and Wen (2004) claim, the quest for universal golden rules about GOF cutoff values is unlikely to be realized. Subjective interpretation based on the entire research process, previous research, as well as statistical analysis (and not just the GOF indices) is required. Therefore, the nature of the path loadings and the significance of the standardized estimates of the parameters is an equally, if not more important consideration than GOF. Since GOF indices may contradict each other, many researchers advocate the reporting of several GOF indicators with research findings.

A commonly cited source is Hu and Bentler (1998, 1999) who establish a minimum cutoff of 0.95 for the CFI and a maximum cutoff of 0.06 for the RMSEA. However, they (and others) noted that sample size, complexity of the model, and the level of misspecification have an effect on these guidelines. Ullman (2006) also advocates the use of the CFI and its cutoff of 0.95. The RMSEA, as Ullman (2006) discusses, is an index that estimates the lack of fit in a model when compared to a saturated or perfect model. Ullman (2006) states that RMSEA values of 0.06 or less are an indication of a close fitting model while those over 0.10 are indicative of a poor fitting model. Hair et al. (2006) gives several rules of thumb for GOF cutoffs based on sample size and model complexity. For research models similar in characteristics of the two models in this research, Hair et al. (2006) uses a value greater than 0.90 for CFI and less than 0.07 for RMSEA. Tabachnick and Fidell (2007) cite the CFI and RMSEA as the most popularly reported GOF indices. In looking at the GOF indices in Table 37, it can be concluded that Research Model 1 (Figures 3 and 5) demonstrates good fit. These results indicate support for hypothesis M1H1: Higher IS development capabilities are associated with higher requirements analysis and design capabilities.

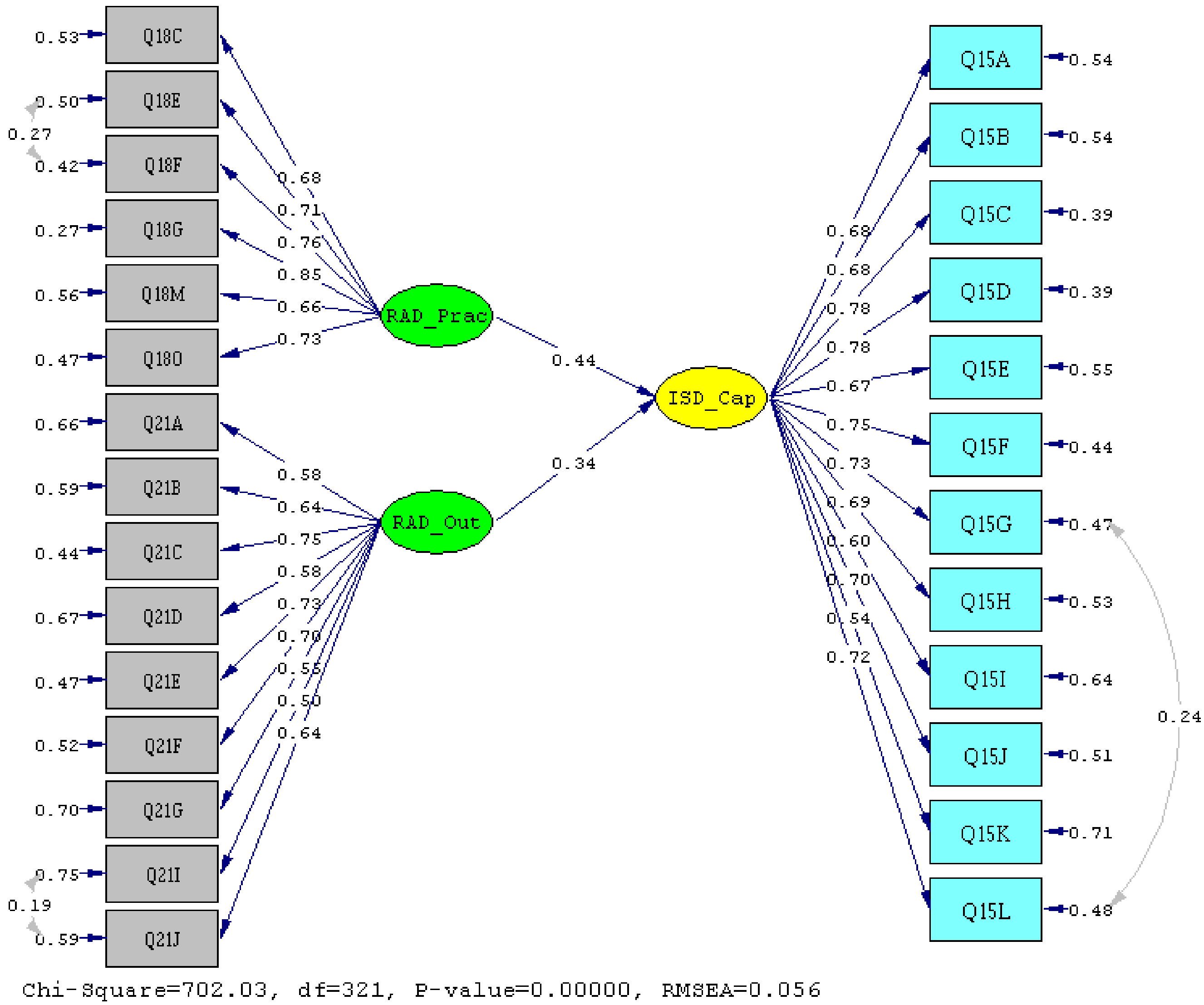


Figure 5 Full structural model for research model 1.

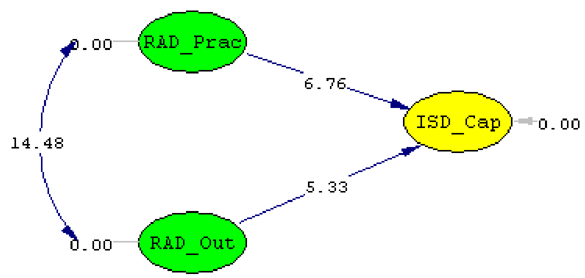


Figure 6. Research model 1 t-values.

Table 36

Descriptives and Reliability Analysis of Final Scales

Scale	Mean	Std Dev	Cronbach's Alpha
IS Development	3.68	0.99	0.922
RAD_Practices 3.52		0.96	0.890
RAD_Outcomes 3.50		0.95	0.865
EA_Org 3.82		0.84	0.852
EA_IT 3.97		0.83	0.806
EA_Synergies 4.10		0.75	0.613
Totals 3.77		0.89	

Table 37

Research Model 1 Goodness of Fit Indicators

Goodness-of-Fit Index	General Criteria or Rule of Thumb	GOF Measure
Root Mean Square Error of Approximation (RMSEA)	< 0.06	0.056
Goodness of Fit Index (GFI)	> 0.90	0.88
Normed Fit Index (NFI)	> 0.90	0.96
Comparative Fit Index (CFI)	> 0.90	0.98
Incremental Fit Index (IFI)	> 0.90	0.98
Parsimony Normed Fit Index (PNFI)	~ 0.76	0.88
Parsimony GFI (PGFI)	~ 0.76	0.75

Research Model 2

The results of the SEM for Research Model 2 (Figure 4) are displayed in Figure 7. Again, in a preliminary evaluation of the model per Bagozzi and Yi (1988), none of the most common anomalies indicating model specification errors (negative error variances, correlations greater than one, or extremely large parameter estimates) are apparent.

The latent factors of requirements analysis and design capabilities (RAD_Prac and RAD_Out) and their associated items remained the same as in the first research model. The latent factors within enterprise architecture consisted of the three latent factors of EA-org[anization], EA-IT, and EA-synergies. EA-org is defined by 6 observed variables (Q20C, Q20G, Q20J, Q20N, Q20S, AND Q20T), EA-IT is defined by 6 observed variables (Q20H, Q20I, Q20K, Q20L, Q20Q, and Q20R), and EA-synergies is defined by 3 observed variables (Q19G, Q19B, and Q20A). As shown in Figure 8, the t-values between the latent factors in this model are significant, indicating significance of the hypothesized relationships among these factors. However, when observing the path loadings between the latent factors RAD_Prac and the EA factors, a negative relationship is indicated.

Table 36 shows descriptive statistics and reliability analysis of the final configuration of the instruments in this model. Though the EA_synergies instrument is below the generally agreed upon lower level of 0.70, it is in the range of acceptability for exploratory analysis (Hair et al., 2006). Further, the lower number of items in this scale also contributes to the lower Cronbach's alpha since Cronbach's alpha is positively related to the number of items in a scale. Table 38 shows several goodness of fit indices characterizing this model. In looking at the GOF indices in Table 38, it can be concluded that Research Model Number Two demonstrates good fit with the data.

Despite the significant positive relationship between RAD outcomes and EA perceptions, the negative t values of the hypothesized relationship between RAD practices and the latent factors comprising EA perceptions, indicate that hypothesis M2H1 (Higher requirements analysis and design capabilities are associated with more positive perceptions of enterprise architecture) is not supported. This will be discussed in more detail in Chapter 5.

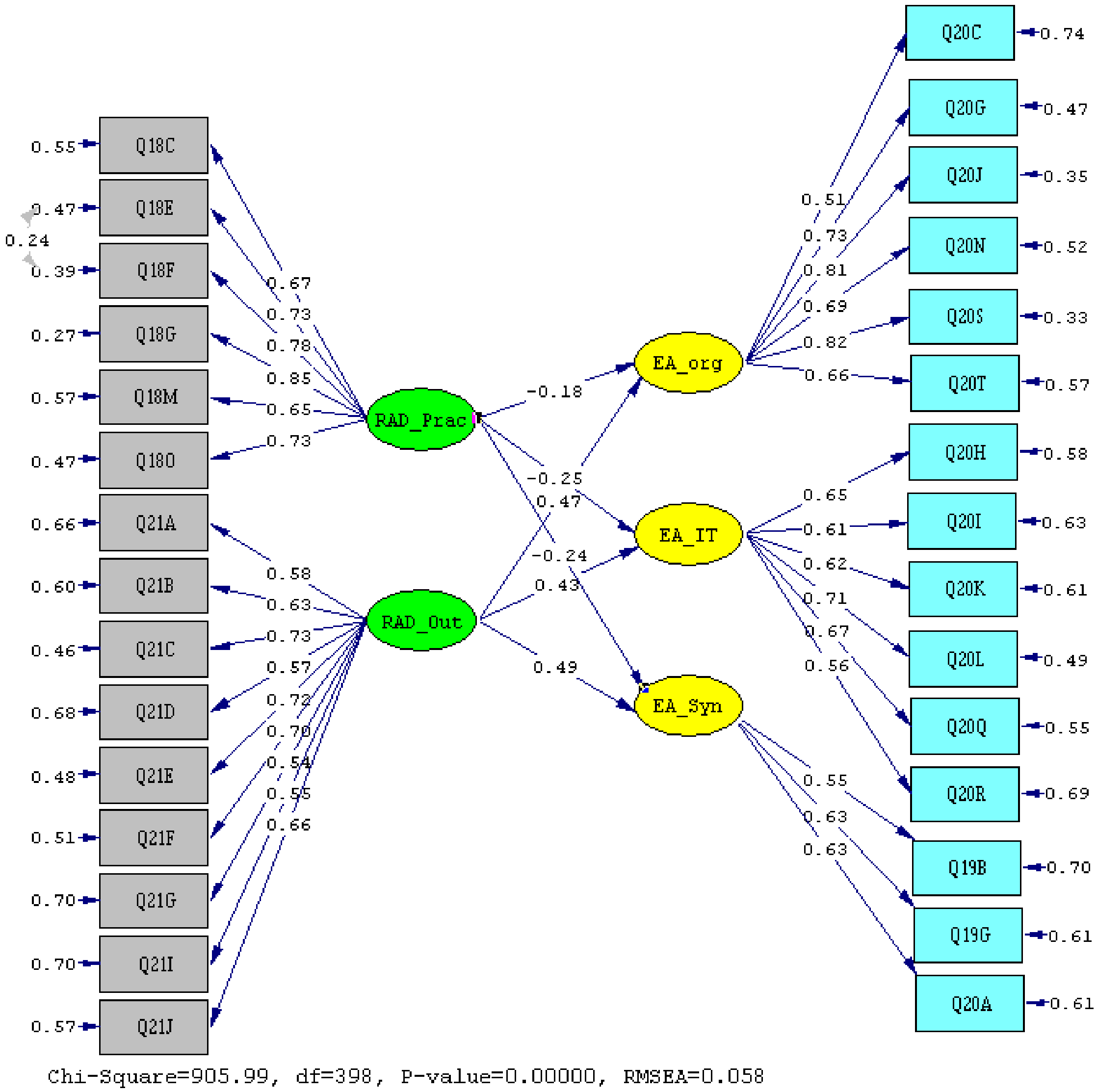


Figure 7. Full structural model research model 2

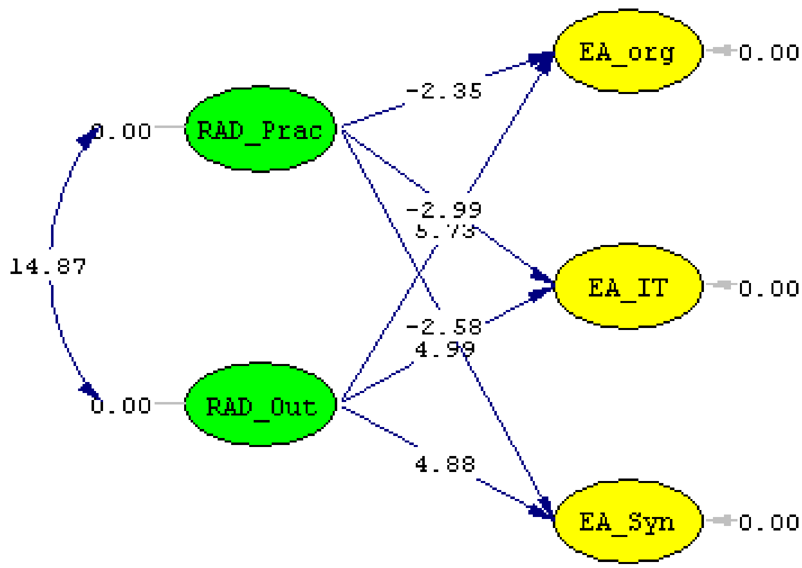


Figure 8. Research model 2 t-values of 2.

Table 38

Research Model 2 Goodness of Fit Indicators

Goodness-of-Fit Index	General Criteria or Rule of Thumb	GOF Measure
Root Mean Square Error of Approximation (RMSEA)	< 0.06	0.058
Goodness of Fit Index (GFI)	> 0.90	0.86
Normed Fit Index (NFI)	> 0.90	0.92
Comparative Fit Index (CFI)	> 0.90	0.95
Incremental Fit Index (IFI)	> 0.90	0.95
Parsimony Normed Fit Index (PNFI)	~ 0.76	0.84
Parsimony GFI (PGFI)	~ 0.76	0.74

Control Variables Analyses

Hypotheses M1H2, M1H3, M1H4, and M1H5 concern the role of the variables of organization size, as measured by number of employees, and IT budget in the relationship between requirements and IS development capabilities. Organization size was categorized into high, middle, and low groups. Organizations of greater than 10,000 employees are in the large group, while the middle group has a range of employees between 1000 and 9999, and the small group has less than 1000 employees. The numbers of respondents in each group were 84, 127, and 147 respectively.

ANOVA multiple comparison procedures were performed on IS development capabilities as the dependent variable and organization size as the independent variable. As indicated in Table 39, no significant results were indicated from various ANOVA multiple comparison techniques. Similar results (Table 40) were evident with the requirements analysis and design capabilities as the dependent variable and organization size as the independent variable. The results from these ANOVA multiple comparison procedures indicate nonsignificant relationships involving organization size in the relationship between ISD and RAD capabilities, thus lacking support for hypotheses M1H2 and M1H3.

Table 39

ANOVA For DV=ISD and IV=Organization Size

	(I) Org Size Grouping	(J) Org Size Grouping	Mean Difference (I-J)	Std. Error	Sig. (p-value)
LSD	0 (n/a)*	1 (1-999)	0.6210	0.4239	0.1438
		2 (1000-9999)	0.6578	0.4243	0.1219
		3 (>=10,000)	0.5017	0.4259	0.2396
	1 (1-999)	0 (n/a)	-0.6210	0.4239	0.1438
		2 (1000-9999)	0.0369	0.0877	0.6747
		3 (>=10,000)	-0.1193	0.0952	0.2109
	2 (1000-9999)	0 (n/a)	-0.6578	0.4243	0.1219
		1 (1-999)	-0.0369	0.0877	0.6747
		3 (>=10,000)	-0.1561	0.0971	0.1086
	3 (>=10,000)	0 (n/a)	-0.5017	0.4259	0.2396
		1 (1-999)	0.1193	0.0952	0.2109
		2 (1000-9999)	0.1561	0.0971	0.1086

*n/a=not answered/don't know

Table 40

ANOVA For DV=RAD and IV=Organization Size

	(I) Org Size Grouping	(J) Org Size Grouping	Mean Difference (I-J)	Std. Error	Sig. (p-value)
LSD	0 (n/a)*	1 (1-999)	0.5771	0.3266	0.0781
		2 (1000-9999)	0.6135	0.3269	0.0613
		3 (>=10,000)	0.5662	0.3282	0.0853
	1 (1-999)	0 (n/a)	-0.5771	0.3266	0.0781
		2 (1000-9999)	0.0365	0.0677	0.5904
		3 (>=10,000)	-0.0109	0.0736	0.8828
	2 (1000-9999)	0 (n/a)	-0.6135	0.3269	0.0613
		1 (1-999)	-0.0365	0.0677	0.5904
		3 (>=10,000)	-0.0473	0.0750	0.5282
	3 (>=10,000)	0 (n/a)	-0.5662	0.3282	0.0853
		1 (1-999)	0.0109	0.0736	0.8828
		2 (1000-9999)	0.0473	0.0750	0.5282

*n/a=not answered/don't know

The IT budget variable was categorized into two categories of high and low. The high category characterized those organizations with an IT budget greater than $10 million, while the lower category had less than a $10 million budget. The numbers of respondents in each group were 180 and 187 respectively.

ANOVA multiple comparison procedures were performed on IS development capabilities as the dependent variable and IT budget as the independent variable. As indicated in Table 41, no significant results were indicated from various ANOVA multiple comparison techniques. Similar results (Table 42) were evident with the requirements analysis and design capabilities as the dependent variable and IT budget as the independent variable. The results from these ANOVA multiple comparison procedures indicate nonsignificant relationships involving IT budget in the relationship between ISD and RAD capabilities, thus lacking support for hypotheses M1H4 and M1H5.

Table 41

ANOVA For DV=ISD and IV=IT Budget

	(I) IT Dept Budget Grouping	(J) IT Dept Budget Grouping	Mean Difference (I-J)	Std. Error	Sig.
LSD	0 (n/a)*	1($0-9.9 million)	0.1444	0.2491	0.5625
		2(>=$10 million)	0.1123	0.2493	0.6527
	1($0-9.9 million)	0 (n/a)	-0.1444	0.2491	0.5625
		2(>=$10 million)	-0.0321	0.0762	0.6739
	2(>=$10 million)	0 (n/a)	-0.1123	0.2493	0.6527
		1($0-9.9 million)	0.0321	0.0762	0.6739

*n/a=not answered/don't know

Table 42

ANOVA For DV= RAD and IV=IT Budget

	(I) IT Dept Budget Grouping	(J) IT Dept Budget Grouping	Mean Difference (I-J)	Std. Error	Sig.
LSD	0 (n/a)*	1($0-9.9 million)	-0.2081	0.2026	0.3049
		2(>=$10 million)	-0.1797	0.2027	0.3760
	1($0-9.9 million)	0 (n/a)	0.2081	0.2026	0.3049
		2(>=$10 million)	0.0284	0.0587	0.6282
	2(>=$10 million)	0 (n/a)	0.1797	0.2027	0.3760
		1($0-9.9 million)	-0.0284	0.0587	0.6282

*n/a=not answered/don't know

In order to more fully explore the dependence and interrelationships between these organizational demographic variables and perceptions of EA, IS development capabilities, and requirements capabilities, two combinations of these demographic variables were examined: 1. IT budget and organization revenue and 2. IT budget and organizational size. By using this particular method, three different characteristics or aspects of organization size were used to provide more insight into the influence of organization size by examining the joint effect of these combination variables. To analyze the role of the organizational or structural variables, ANOVA tests with the combinations of the demographic variables as independent variables and perceptions of EA (in the form of the three factors of EA), IS development capabilities, and requirements capabilities as the dependent variables were performed.

Table 43 shows the relationships among the IS development (ISD) capabilities and organization size variables. The six size categories were determined by combining two categories of IT budget with three categories of organization revenue. The IT budget size categories are indicated by the first digit (either a 1 or a 2) and represent the two categories of 1 small (IT budget under $9.9 million) (n=187) and 2 large (IT budget greater than $10 million) (n=180). The second digit (a 1, 2, or 3) is indicative of the three revenue categories of 1 small (revenue under $500 million) (n=147), 2 medium (revenue between $501 million and $4.9 billion) (n=127), and 3 large (revenue greater than $5 billion) (n=84). In their research into why organizations do not adopt CMMI, Staples et al. (2007) categorized their size variables into the categories of small, medium, and large. When looking at the respondents' organizations' IS development capabilities in Table 43, there are significant differences between the means of those whose IT budget is smaller and those whose IT budget is larger and the size (measured by organization budget or revenue) is in the middle or larger. Also, even when the IT budget is larger, a significant difference in means is indicated between the middle range and larger organization size. Of note is that in cases where a significant difference exists, a large (3) organization revenue is involved.

Table 43

ANOVA For DV=ISD Capabilities and IVs=Combination of IT Budget and Organization Revenue

	IV: Combo of IT Budget/Org Revenue (I)	IV2: Combo of IT Budget/Org Revenue (J)	Mean Difference (I-J)	Std. Error	Sig. (p-value)
LSD	**Small (1)/Small (1)**	Small (1)/Medium (2)	-.11647	.12926	.368
		Small (1)/Large (3)	-.35131	.18089	.053
		Large (2)/Small (1)	-.04841	.15611	.757
		Large (2)/Medium (2)	.04220	.10598	.691
		Large (2)/Large (3)	**-.27615(*)**	.11360	.016
	Small (1)/Large (3)	Small (1)/Small (1)	.35131	.18089	.053
		Small (1)/Medium (2)	.23483	.20088	.243
		Large (2)/Small (1)	.30290	.21913	.168
		Large (2)/Medium (2)	**.39350(*)**	.18676	.036
		Large (2)/Large (3)	.07516	.19118	.694
	Large (2)/Medium (2)	Small (1)/Small (1)	-.04220	.10598	.691
		Small (1)/Medium (2)	-.15867	.13735	.249
		Small (1)/Large (3)	**-.39350(*)**	.18676	.036
		Large (2)/Small (1)	-.09061	.16287	.578
		Large (2)/Large (3)	-.31834(*)	.12272	.010

* The mean difference is significant at the .05 level.

This observation is similar when examining requirements practices (Table 44), (since no significant difference between a small IT budget and large organization revenue was found, these have been left out of Table 44). Again, ANOVA multiple comparison results indicate a significant difference in means of the combination of variables when the IT budget and organization revenue is small and when the IT budget and organization revenue are both large. Even when IT budget is larger and the organization revenue is in the middle range, there is a significant difference with the combination of a larger IT budget and larger organization revenue. Note again that in every case a significant difference is found, a large organization revenue (3) is involved.

Table 44

ANOVA For DV=Requirements Practices and IVs=Combination of IT Budget and Organization Revenue

	IV: Combo of IT Budget/Org Revenue (I)	IV²: Combo of IT Budget/Org Revenue (J)	Mean Difference (I-J)	Std. Error	Sig. (p-value)
LSD	**Small (1)/Small (1)**	Small (1)/Medium (2)	-.12120	.10407	.245
		Small (1)/Large (3)	-.15883	.14909	.287
		Large (2)/Small (1)	.03624	.12569	.773
		Large (2)/Medium (2)	.03603	.08533	.673
		Large (2)/Large (3)	**-.18052(*)**	.09146	.049
	Large (2)/Medium (2)	Small (1)/Small (1)	-.03603	.08533	.673
		Small (1)/Medium (2)	-.15723	.11058	.156
		Small (1)/Large (3)	-.19486	.15370	.206
		Large (2)/Small (1)	.00020	.13113	.999
		Large (2)/Large (3)	**-.21655(*)**	.09881	.029

* The mean difference is significant at the .05 level.

These results indicate that when considering central, fundamentally important IT practices and capabilities in organizations (such as requirements and IS development), the size of an organization (measured by organization revenue or IT budget) may have an influential role. At the least, size does seem to indicate a likelihood of differences in capabilities, although, information intensity may play a role in some cases too.

Summary

This chapter presents the statistical analysis of data resulting from the survey of IT professionals. The results indicate support for the relationship between higher ISD capabilities and higher RAD capabilities, while lacking support for the hypothesis that higher RAD capabilities are associated with higher, more positive perceptions of enterprise architecture. However, as will be discussed in Chapter Five, an interesting inverse relationship between RAD practices and EA is revealed. Also, the ANOVA multiple comparison procedures indicated a lack of support for significant effects from organization size and IT budget. However, a more granular look into the joint effects of IT budget and organization revenue with ISD and RAD capabilities does indicate effects of these variables on the ISD and RAD relationship. An enhanced summary and discussion of these results will next be discussed in Chapter 5.

CHAPTER 5
DISCUSSION

The mission of this research study was to study the relationships between information systems development capabilities, requirements capabilities, and perceptions of enterprise architecture within organizations. Additionally, the impact of organizational revenue and IT budget size within these relationships was also considered. From the results of the statistical analyses, support was found for the hypothesized relationships between IS development and requirements capabilities. However, the hypothesized relationships with the organizational demographics were not supported nor was the hypothesized positive relationship between requirements capabilities and EA perceptions. However, as will be discussed below, some interesting findings from the testing of these unsupported hypotheses were revealed. This chapter will provide a thorough discussion of these findings, proposing explanations for both the supported and unsupported hypotheses. Within this discussion, the research contributions and implications of this study will be offered. Following this, the limitations within this research study are presented and finally, directions for future research are discussed.

Discussion of Research Findings

The results of the structural equation modeling indicate both research models did indeed fit the data well. Regarding the hypothesized relationship where higher IS development capabilities are associated with higher requirements analysis and design capabilities, the SEM results indicate support of this hypothesis whereas the other hypotheses were unsupported.

Research Model 1

The support of the hypothesized relationship associating higher IS development capabilities with higher requirements analysis and design capabilities provides an important linkage between higher requirements capabilities and organizational IS development improvements, with the result being a narrowing of the gap between requirements, requirements payoffs and benefits, and IS development. This finding provides an increase in support for the importance of the requirements activities of IS development called for by Brooks (1995) and other researchers (Beecham et al., 2005; Sommerville & Ransom, 2005). This research finding is also important in that it contributes to the research about the payoffs and benefits from requirements practices, processes, and products (Damian & Chisan, 2006; Davis & Hsia, 1994).

Another important contribution is to highlight the role of requirements within the capability maturity model. As pointed out by many researchers, a gap exists in accounting for the role of requirements within software process improvement programs (Hutchings & Knox, 1995; Layman, 2005; Linscomb, 2003; Sawyer et al., 1997). By providing a focus on the vital role that requirements plays on IS development capabilities, IS professionals have an incentive for the inclusion of requirements capabilities within software process improvement programs such as the CMM(I). Moreover, there can be support for the case for requirements improvement initiatives in not only improving requirements capabilities, but also in improving IS development capabilities.

This aspect of the relationship between IS development and requirements also enhances Kappelman et al.'s (2006) research of red flags of IS project failures and the lack of adequate risk management in IS project management. The role of requirements in enhanced IS development capabilities strengthens the notion that effective IS project management must manage and mitigate the risk of not getting the requirements right.

These findings are strong indications validating Brooks' (1995) notion that requirements are the essence of the IS development process and that greater benefits accrue from a focus on the essential difficulties rather than the accidental ones. Higher levels of maturity in requirements practices contribute to higher levels of IS development maturity which can lead to many of the benefits experienced by organizations with higher levels of IS development maturity, such as lowering defect potential rates and lowering the associated costs of defects in the IS development life cycle (Jones, 2008).

This support for Brooks' (1995) hypothesis is important in order to provide more focus to organizational efforts at improving their IS development capabilities. By providing support for the important role of requirements, this research can facilitate the concentration of efforts of an organization within their software project improvement initiatives. The importance of this concentration of effort is expressed in research by Sawyer, Sommerville, and Viller (1998) who state: "No software process, whatever its 'capability', can keep delivery times, costs and product quality under control if the requirements are poorly formulated or unstable" (p. 1). Overall, whether or not IS development projects are successful depends on how successfully they meet the needs of the applicable users and the usage environment (Cheng & Atlee, 2007; Nuseibeh & Easterbrook, 2000). These needs are the requirements.

Role of the Control Variables: Organization Size and IT Budget

The initial results from the statistical analysis on the demographic variables of organization revenue and IT budget size on the relationship of IS development and requirements indicate these hypotheses were unsupported:

> M1H2: Larger organizations are associated with higher IS development capabilities when compared to smaller organizations.
>
> M1H3: Larger organizations are associated with higher requirements analysis and design capabilities when compared to smaller organizations.
>
> M1H4: Organizations in industries where the IT budget tends to be higher (e.g. financial and telecom firms) are associated with higher IS development capabilities.
>
> M1H5: Organizations in industries where the IT budget tends to be higher (e.g. financial and telecom firms) are associated with higher requirements analysis and design capabilities.

Because of the extent of research indicating the effects of organizational size and resources on these relationships (Johnson & Brodman, 1999; Lumsden, 2007; Pino, Garcia, & Piattini, 2008; Richardson & von Wangenheim, 2007; Strigel, 2007) more granularity within these relationships was sought. Thus combinations of the demographic variables of IT budget and organization revenue were added to the statistical analysis. These results indicate when assessing central, fundamentally important IT practices and capabilities of organizations (such as requirements and IS development), the size of an organization (measured by organization revenue and IT budget) may have a significant role. Size does seem to indicate a likelihood of capability differences.

The notion of organization size having an important relationship within core IT capabilities and practices has been supported, as previously discussed, in prior research (e.g., Johnson & Brodman, 1999; Mabert et al., 2003; Pino, Garcia, and Piattini, 2008; Richardson and von Wangenheim, 2007). This study's finding provides some insight into environmental aspects impacting the success of IS development and requirements improvement initiatives in organizations, nevertheless, considerations like industry and information intensity may also play a role. This finding is important to smaller organizations in order to make them aware of potential difficulties in achieving a more mature level regarding their IS development and requirements capabilities. By recognizing this, smaller organizations planning to improve their IS development capabilities will need to consider applying more resources in order to achieve a higher level and increase the chances of a successful improvement process.

Research Model 2

The results of the SEM on Research Model Number Two indicate hypothesis M2H1 (Higher requirements analysis and design capabilities are associated with more positive perceptions of enterprise architecture) was unsupported. The premise for the hypothesis of a positive relationship between requirements and EA was that if the organization had made the organizational, cultural, and technical investments required for higher requirements capabilities and realized the commensurate benefits, then these same investments would in turn facilitate and provide a basis for EA practices and the RAD benefits result in positive expectations for EA. In other words, the EA and requirements capabilities of an organization would be complimentary or synergistic with each other. Although this notion was supported by the RAD artifacts—EA perceptions findings, of particular interest is the negative relationships between requirements practices and the three factors of EA perceptions.

A plausible explanation for this seeming anomaly is based on the overall outlook of the survey respondents regarding their overall viewpoints towards IT and organizational challenges. The results from the survey reflect the fact IS professionals responding to this survey may be internally focused on the IT aspects of the organization rather than broader, organization-wide issues. (This is similar to a 2006 survey which found that 34% of respondents believe EA is not even involved with the business (van den Berg & van Steenbergen, 2006)). This is evident when looking at the four highest means of the responses to the questions asking respondents to select the level to which they agree or disagree that a series of statements were representative of the potential benefits to an organization from practicing enterprise architecture (using a 5-point Likert scale from *strongly disagree* to *strongly agree*):

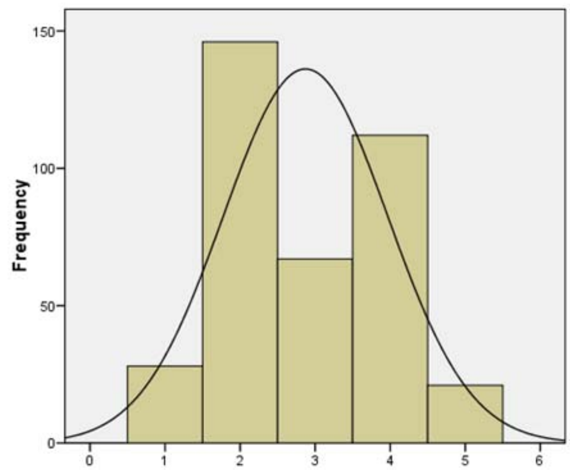
- 20a "Aligning business objectives with information technology investments" with a mean of 4.14 and more that 83% answering agree or strongly agree;
- 20f "Improved utilization of information technology" with a mean of 4.25 and more that 86% answering agree or strongly agree;
- 20h "Improved interoperability among information systems" with a mean of 4.34) and more that 87% answering agree or strongly agree; and
- 20q "More effective use of IT resources" with a mean of 4.02 and more that 78% answering agree or strongly agree.

The common thread of these four items is the potential benefits of EA are all within the context of IT and they all suggest an IT-centric view of EA rather than EA's benefits to the larger organization. Supporting this conclusion, are the results of question 19a "The purpose/function of enterprise architecture is: to provide a snapshot in time of an organization" (Table 45). About 46% of the respondents disagreed or strongly disagreed with this statement (nearly 64% were either neutral or disagreed) while only 35% agreed or strongly agreed. The amount of disagreement with the idea that the architecture of an enterprise can provide "a snapshot in time of an organization" is important in possibly explaining the negative relationship between requirements practices and EA perceptions since responses to question 19a seems to indicate that the majority of respondents do not believe that EA is necessarily about the enterprise at all. Rather, they may believe EA is only about IT: A belief with adverse ramifications on their EA and IT efforts, leading to a persistent shortfall of IT-business alignment, overly complex systems, and enterprise disintegration across stove-piped systems.

Table 45

Question 19a

		Frequency	Percent
	1	28	7.43
	2	146	38.73
	3	67	17.77
	4	112	29.71
	5	21	5.57
Responses		374	99.20
Missing		3	0.80
Mean		2.87	
Median		3.00	
Std Dev		1.10	

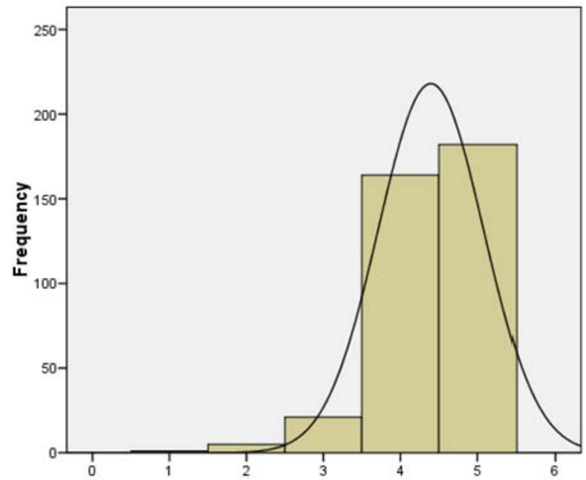


Further support for this conclusion can be seen by looking at responses to question 19e “The purpose/function of enterprise architecture is: to provide a blueprint of an organization’s business, data, applications, and technology”. As indicated in Table 46, nearly 92% of respondents agreed or strongly agreed with the IT-centric view of EA in question 19e.

Table 46

Question 19e

	Frequency	Percent
1	1	0.27
2	5	1.33
3	21	5.57
4	164	43.50
5	182	48.28
Responses	373	98.94
Missing	4	1.06
Mean	4.40	
Median	4.00	
Std Dev	0.68	



Besides characterizing the overall perspective of respondents (organization versus IT), this finding may also reflect difficulties in the early phases of implementing an EA program (Armour, Kaisler, & Liu, 1999) resulting in dissatisfaction with nascent EA initiatives. Many organizations experience difficulties in the beginning with implementing EA or dissatisfaction with the pace and results early on in an EA implementation. Much of the basis for these difficulties is due to the broad nature of EA. As individuals conduct high-level looks at their organizations, there are many important areas that are important to model so organizations find it difficult to know where to begin. Thus there is a tendency to model too much of the organization at the beginning stages of an EA program. This “big-bang” approach or “analysis paralysis” is a major reason why EA efforts fail according to Armour, Kaisler, and Liu (1999).

In examining the finding from Research Model Number Two and these opinions of the survey respondents discussed in the preceding paragraphs, it may be these organizations’ individual requirements processes are more capable or mature, but they are aligned for building locally-optimized, individual information systems, not for the enterprise-wide, integrated system of systems that EA strives to depict, organize, and manage. Thus IS professionals with positive perceptions of their requirements practices may be dissatisfied when looking at their organizations' EA initiatives because they are looking at them from an IT-centric perspective. It goes back to the eternal quest for alignment, organizations realize the need for IT and the business (or technologists and strategists), to combine in a coalition to meet organizational goals, but in many cases these efforts do not succeed (Sauer & WIllcocks, 2002).

In a similar manner, this relationship may exist because the respondents' organizations are very adept (and mature) at locally optimized solutions (as indicated by their responses to requirements practices items) but are lower in their EA maturity. By viewing this relationship through the lens of one of the EA maturity models which was used in the formation of the survey items, the organizations have indications of being within the first stage of EA maturity (Application or Business Silo stage) as documented in Ross' research (2003). Of the firms studied in Ross' (2003) research, most of them were in this initial stage of EA maturity. In this initial stage of EA, according to Ross (2003), organizations' IT resources are focused on individual applications. Moreover, these individual applications, while successful individually, create problems with standardization and integrating with other systems and processes when considering the organization as an enterprise. This environment is one where immediate needs are dealt with while future capabilities are not considered.

Perhaps this explains the finding of the inverse relationship between requirements practices and EA perceptions, where IS professionals appropriately measure their requirements practices higher, but due to a variety of reasons, their perceptions of EA are lower because of the broader, more organization-centric nature of EA. They still maintain a local view, rather than an enterprise view, of their information systems. Some of these reasons for a lower perception of EA may be due to IS professionals' not believing EA is about requirements, or a tendency towards looking at the technical side versus organizational side of systems. They believe the standardization and integration that comes about from more mature EA may impose a more restrictive environment on them and they would lose some aspect of control. This is consistent with Dreyfus and Iyer's (2006) and Iyer and Gottlieb's (2004) research asserting an EA becomes existent due to the implementation of individual projects ("a sequence of IS project implementations").

Another possible explanation is derived from the notion that familiarity breeds realism. The IS professionals responding to this survey that do requirements well realize how hard this is to do and this same belief transfers to their perceptions of EA. They realize how much harder EA is since organizations are accomplishing the requirements practices not just for one system but for the entire enterprise. Thus their perceptions of EA are lower, since they realize EA is no silver bullet. Moreover, the explanation of familiarity breeds realism corresponds to a Gartner report describing the evolution of EA over the last 15 years as an expansion of its scope and focus from a ground-level view of IT to a more encompassing strategic view of the business linking IT to business strategy (Lapkin, 2008). As with IS development failures, academic research on the difficulty of requirements processes is lengthy. For instance Davis (1982) identified three primary reasons that comprehensive requirements are difficult to obtain: 1) the limitations of humans as information processors and problem solvers; 2) the complexity of information requirements; and 3) the complexity of the interactions among users and IS analysts. Browne and Ramesh (2002) highlight the difficultly in understanding individual and organization requirements resulting in the large number of completed information systems not meeting user needs.

The ramifications of this finding are significant to the success of EA initiatives. Primarily, the research suggests that to achieve positive outcomes on the alignment and integration of organization-wide and project-specific requirements, the case will have to be made that EA is about requirements, and specifically enterprise-wide requirements, not vertical, or locally-optimized requirements. The negative relationship between requirements practices and EA perceptions may portend the notion that IS professionals do not believe that EA is about requirements. Ross, Weill, and Robertson (2006) maintain that top performing organizations are distinguished by greater senior management involvement in EA, concerted activities to integrate EA into project methodology, and more mature EA. Possibly, more effort must be exerted to integrate EA into such project methodology components as IS requirements. Perhaps it is, as Cook (1996) maintains, that the emphasis on end-user satisfaction leads to the encouragement of vertically aligned information systems, optimized for a single department or objective, but not necessarily optimized for the enterprise. Therefore good requirements practices may not necessarily convey to a good or more mature EA program. EA is trying to move these good, vertical processes to horizontal, more enterprise-centric processes.

This finding provides additional contributions to EA. First of all, in considering critical success factors for inclusion in EA maturity models (such as the role of leadership, communications, and alignment), the role of IS requirements in an EA context is examined. Although it is tempting to equate the maturity of various practices that contribute to requirements capabilities to that of EA maturity, both of them may not move in lockstep. Though such factors as executive leadership support, effective communication, and alignment are evident in higher maturity requirements and EA processes of organizations, they may not by symbiotic with each other and organizations cannot assume that their mature requirements capabilities automatically lead to or translate to EA maturity and success. Additionally, an effective EA maturity model can be used to provide a measure of return on investment for EA and as a vehicle to report the progress on EA to executive leadership (Sherwood, Clark, & Lynas, 2005). Both of these are necessary in garnering adequate support for an EA program and in order to keep it in place.

Moreover, this research demonstrates that the orientation of the organization's IS professionals (whether IT-centric or a more enterprise-wide perspective) plays a role in how EA is perceived and possibly may affect overall EA implementation initiatives, especially since EA programs are, in many cases, spearheaded from the IT department. IS professionals may be more likely to agree with researchers such as Zarvic and Wieringa (2006) who see EA as IT systems in an enterprise context, not as a holistic, organization-wide construct involving not only information systems but organizational processes and governance as well (Bernard, 2005).

The negative relationship may have ramifications for organizations on how best to staff and structure their EA department or team. If IS professionals tend to remain more technically or tactically focused even in the context of EA, then, as suggested by

Rehkopf and Wybolt (2003), perhaps highly technical people should not be in the EA department or EA staff. As the relationship described in this study between higher requirements capabilities and EA perceptions implies, some IS professionals may be more inclined towards a techno-centric view of the organization and its objectives versus a more enterprise-wide view which is more conducive to effective EA initiatives. The research findings suggest that IT individuals perceive EA in terms of an IT-centric viewpoint. This may, in part, explain why Ross (in Kappelman, 2009) sees EA efforts moving outside of IT control entirely.

This narrower, techno-centric view of EA may have ramifications and increase risks on the successful implementation of EA and the subsequent realization of benefits from EA initiatives. Because of the enterprise-wide purpose of EA, an EA program will necessarily require extensive resources and participation from the organization. This also imposes political, managerial, and organizational challenges (Kaisler et al., 2005; Niemi & Ylimaki, 2007). Therefore organizations putting in place EA programs must be aware of the potential conflict among these views, especially as IS professionals and the IT department are, in many cases, expected to be the lead proponent and department in EA initiatives. This view of EA as not a business-oriented, strategic process may also reflect the oft-cited view of IS professionals being locked-in a technical mindset which contributes to the perceptions of their ignorance toward business and strategy concepts. Many times this view leads to frustrated IS professionals calling for CIOs to ‘get a seat at the table’. A concern is that these research findings indicate that the majority of respondents to this survey do not believe EA is about the enterprise, rather they believe it is only about IT and this view may further erode the ability to attain IT-business alignment and exacerbate the ability of IS professionals to get a ‘seat at the table’. As Jeanne Ross puts it in the Foreword of the forthcoming book, *The SIM Guide to Enterprise Architecture: Creating the Information Age Enterprise*, (Kappelman, 2009), “Although EA was initially a function within the IT organization, we will soon find IT to be a function within EA…Someone needs to provide the leadership to design the processes, implement new systems and processes, change behaviors, and drive value…If the CIO does not take the lead, someone else will emerge, and the CIO will have a new boss”.

Since the preponderance of EA research is on technical rather than human issues (Frampton, 2007), this research provides valuable insight into the perceptions of IS professionals towards EA. Additionally, this research provides an explanation about how perceptions from these individuals potentially could impact EA initiatives in organizations.

Limitations

A limitation of this research is the population from which the sample is drawn: the membership of the Society for Information Management (SIM). SIM is an organization of senior-level IS professionals focusing on IT leadership issues. Therefore the respondents are generally those with broader, more organizational perspectives. This is evident when looking at the job responsibilities of respondents discussed in Chapter Four The level of responsibility of the survey respondents was generally broad-based, with almost 80% reporting their responsibility at either an enterprise (63.4%) or a business-unit/divisional level. Thus respondents with a more tactical or operational perspective may not have been reflected in the survey. However, due to focus of the survey, especially the enterprise architecture context, a strategic, broad, enterprise-level demographic was the desired target for the survey. The reason for this is that it was felt that this type of demographic would be more likely to be able to respond to items involving IS development and requirements capabilities as well as EA when compared to more narrow, technically focused respondents. This was also why the survey was not sent to other available IT outlets for responses.

Another limitation is that the survey focused on perceived potential benefits of EA, which may partially explain the lack of variance among different types of organizations. Therefore, a natural extension of this study would be to measure the perceived organizational benefits of EA and to relate them to specific EA practices and organizational characteristics. Also the lack of an agreed upon and well-understood definition of enterprise architecture is a problem when attempting to garner perceptions of enterprise architecture. Similarly, there are many different types of architecture used when individuals consider EA. Examples include enterprise architecture, data architecture, technical architecture, and software architecture, to name a few. This proliferation of terms and meanings can lead to confusion as to what is being talked about and in interpretation of responses. Other limitations include the use of categorical data which may have hindered the ability of this study to ascertain the implications of size. The requirement (imposed by the SIMEAWG) that the survey be completed in 10-15 minutes may also have limited this study be restricting the number of questions that could be included.

Of course, in developing this survey with the SIMEAWG and using a modified-Delphi technique, unilateral decisions on research design had to yield to group consensus. An example of advice from this expert group was to structure the survey so it would not appear as strictly an EA practices survey. This was decided because the expert group believed that many organizations may be doing EA-related activities but not calling them EA. Thus the name of the survey did not include the words “Enterprise Architecture”, but was called the “SIM Information Management Practices Survey”. Moreover, some practices questions used the term “Requirements Analysis and Design” instead of using the term “enterprise architecture”. This reflected the expert group’s determination that questions regarding requirements-related practices could serve as a surrogate for at least certain fundamental EA capabilities and practices. The results of these decisions did not allow responses that could be used to directly assess the maturity of the organizations’ EA practices and programs nor could survey results be used to specifically determine whether or not the organizations had EA programs or offices. However, the partnership with industry on this survey (and for future research) strengthens the overall research effort. To echo Guide and Wassenhove (2007) who are strong advocates of academic and industry cooperation in research: “…the potential benefits of partnering with industry are enormous” (p. 531). So despite the imposition of some limitations, the overall outcome of this partnership with industry within the SIMEAWG was both essential and rewarding to this research study.

Future Research

As the first survey in what is hoped to be a longitudinal study of enterprise architecture, this survey will provide a foundation to begin further EA surveys. Future research should include items specifically asking about such EA topics as: 1.

Definitions of EA; 2. Whether or not the organizations have an EA program; and 3. more items in which an organization's EA maturity can be assessed.

Another aspect of EA which should be examined is the role of the learning organization (Senge, 1992). The learning organization plays an important role in the evolution of organizations' EA programs and is presumably a foundation in many of the EA maturity models. Items characterizing organizational learning aspects can play an important role in assessing EA maturity through a survey instrument.

Continuing on this research study's contribution to human issues of EA, further research should be conducted as to characterizing and understanding the desired and most effective skill set for enterprise architects. Examining this important issue can provide practical results for organizations wishing to embark on an EA program, as well as to curriculum development efforts of academics.

As part of the longitudinal analyses, future research should also aim to involve a sizeable number of respondents from the US Federal Government. At least since the mandate for EA in all Federal agencies and departments with the Clinger-Cohen Act of 1996, the Federal Government has been actively pursuing EA. There is a broad level of experience and knowledge related to EA from this sector that would be useful in comparison analysis with this and future surveys' private sector respondents. This type of study can provide valuable insight into the compatibility of the public and private sectors' EA initiatives and whether any significant differences exist that would impact the cross-flow of EA-related information, lessons learned, and critical success factors. Such analysis could also target questions as to whether some EA frameworks and maturity models are equally beneficial across sectors.

Finally, this and future EA survey-based research should elicit responses from non-IT, organizational leadership. The research intent should be on establishing perspectives on EA that are characteristics and determinants of the business value and perceptions of EA. This type of research is important in understanding the level of support for EA and perceptions of EA from non-IT individuals.

Conclusion

The mission of this research study was to examine the relationships between information systems development capabilities, requirements capabilities, and perceptions of enterprise architecture within organizations. This research contributed to the body of knowledge regarding the significant role of requirements in IS development and strengthens support for the concept that requirements are the essence of IS development. Furthermore, this research adds to the enterprise architecture body of knowledge by revealing the interaction between EA and IS requirements capabilities, providing insight into how IS leaders perceive EA within their organizations and IT processes.

Information systems research is replete with discussion and analysis on technological topics to improve the nature of information systems and development and face the myriad of challenges of IS practice. Such topics include studies of development methodologies, programming languages, and the integration of new innovations. However, this research study contributes to the body of IS research by examining the roles of requirements and EA. EA, steeped in holistic and strategic values, resonates as a way to address the multiple challenges IS professionals face, from the unremitting trend of IS development failures (e.g. being over budget and over time) to dissatisfaction of IS solutions by end users and executive leadership. EA may not represent the total solution, but it emphasizes a belief that IS processes can and should be improved upon and that they are best conceived as enterprise processes. We should not continue with the status quo but should shift the focus to broader, holistic concepts that attack the essential problems about the nature of information systems. By revealing new aspects of EA, IS development, and requirements, this research study provides a firm starting point to accomplish this shift in focus.

APPENDIX

SIMEAWG SURVEY INSTRUMENT

SIM Information Management Practices Survey
Page 1

Informed Consent Notice

Before agreeing to participate in this research study, it is important that you read and understand the following explanation of the purpose of the study and how it will be conducted.

Title of Study: SIM Information Management Practices Survey

Principal Investigator: Dr. Leon Kappelman, Department of Information Technology and Decision Sciences, College of Business, University of North Texas (UNT).

Purpose of the Study: You are being asked to participate in a research study which involves a survey, which will take about 10 minutes of your time, that will help determine the information management practices and perceptions of corporations and government entities.

Procedures for Maintaining Confidentiality of Research Records: Your responses to this survey will be completely anonymous except to the UNT research team. Only aggregates and summaries will be published in any form. Your participation in this experiment is not completely anonymous since your email address will be used to give you access to the survey. This is to ensure that only qualified professionals participate in the research. There will be no area within the survey asking for personally identifiable data, although some demographic questions will be asked.. The data gathered will only be used for research and publication purposes. The absolute confidentiality of your responses will be maintained in any publications or presentations regarding this study.

If you have any questions about the study, you may contact Dr. Leon Kappelman at telephone number 940 565 4698.

Review for the Protection of Participants: This research study has been reviewed and approved by the UNT Institutional Review Board (IRB). The UNT IRB can be contacted at (940) 565-3940 with any questions regarding the rights of research subjects.

Research Participants’ Rights:

> Continuing with this survey indicates that you have read or have had read to you all of the above and that you confirm all of the following:

1. You understand that you do not have to take part in this study, and your refusal to participate or your decision to withdraw will involve no penalty or loss of rights or benefits. The study personnel may choose to stop your participation at any time.
2. You understand why the study is being conducted and how it will be performed.
3. You understand your rights as a research participant and you voluntarily consent to participate in this study.
4, You have been told you will receive a copy of this form.
{Choose one}
( ) I have read and agree with the Informed Consent; Please start the survey.
( ) I do not wish to participate.

Click here to leave the survey interface

Page 2

Part I. Please answer the following questions about yourself.
1. What is your primary job title? (Select only one)
{Choose one}
( ) Chief Executive Officer (CEO)
( ) Chief Information Officer (CIO)
( ) CFO/Treasurer/Controller
( ) Chief Technology Officer (CTO)
( ) Director
( ) Enterprise Architect
( ) Member of the Board
( ) Vice President
( ) Other (please specify) [ ]

2. What is the primary job title of the person you report to? (Mark only one)
{Choose one}
( ) Agency Deputy Director/Agency Under Secretary/Agency Deputy Administrator
( ) Agency Director/Secretary/Administrator
( ) Chairman of the Board
( ) Chief Executive Officer (CEO)
( ) Chief Enterprise Architect
( ) Chief Financial Officer (CFO)
( ) Chief Information Officer (CIO)
( ) Chief Operating Officer (COO)
( ) Chief Technology Officer (CTO)
( ) Controller
( ) Enterprise Architect
( ) Head of Audit Committee
( ) Member of the Board
( ) Vice President
( ) Other (please specify) [ ]

3. What is your age in years? (fill in the blank)
{Enter text answer}
[ ]

4. How many years have you been with this organization? (fill in the blank)
{Enter text answer}
[ ]

5. How many years have you held your present position? (fill in the blank)
{Enter text answer}
[ ]

6. Education (select your highest degree).
{Choose one}
( ) High School
( ) Some College
( ) Associate Degree
( ) Bachelor's Degree
( ) Master's Degree
( ) PhD, JD, MD, or other terminal degree

7. What is your highest level of responsibility in your organization? (select one)
{Choose one}
( ) Enterprise-wide
( ) Division/Business Unit
( ) Department
( ) Team/Workgroup

9. Please mark the answer that best describes your organization. (Mark only one).
{Choose one}
( ) Profit-making corporation
( ) Governmental organization
( ) Not-for-profit organization
( ) Other [ ]

Profit-making corporation:
{Choose one}
( ) Publicly-traded company
( ) Privately-held company
( ) Other [ ]

Governmental organization:
{Choose one}
( ) Department of Defense
( ) Federal
( ) State
( ) County
( ) Municipal
( ) Other [ ]

Not-for-profit organization:
{Choose one}
( ) Educational
( ) Religious
( ) Philanthropic Foundation
( ) Charitable Institution
( ) Other [ ]

10. How many people work in your organization?
{Choose one}

( ) Less than 100
( ) 100-499
( ) 500-999
( ) 1000-4999
( ) 5000-9999
( ) 10,000-19,999
( ) 20,000-29,999
( ) 30,000-50,000
( ) Greater than 50,000
( ) Don't Know

11. What range best describes the gross revenues/income (or budget if government or not-for-profit) of your organization in the last fiscal year?
{Choose one}

( ) Less than $50 million
( ) $50 - $100 million
( ) $101 - $500 million
( ) $501 - $999 million
( ) $1billion - $4.9 billion
( ) $5 billion - $9.9 billion
( ) $10 billion - $14.9 billion
( ) $15 billion - $24.9 billion
( ) $25 billion - $50 billion
( ) Greater than $50 billion
( ) Don't Know

Part II.
8. Please mark the one answer that best describes the industry of your organization.
{Choose one}

( ) Agriculture
( ) Banking/Securities/Investments/Finance/Insurance
( ) Business Services (Legal/R&D)
( ) Capital Goods Mfg
( ) Chemical
( ) Construction/Engineering
( ) Consumer Goods Mfg
( ) Education
( ) Entertainment
( ) Food Service
( ) Government (Fed, State, Local)
( ) Healthcare/Medical/Pharmaceutical/Biotech
( ) Hotels/Tourism/Travel
( ) IT Services Provider/Consultant
( ) Military

( ) Mining/Energy
( ) Printing/Publishing
( ) Real Estate
( ) Retail/Wholesale
( ) Transportation/Distribution/Logistical
( ) Utilities
( ) Other [ ]

12. Where is your organization primarily located?
{Choose one}
( ) United States only
( ) North America (other than USA)
( ) South America
( ) Africa
( ) Asia
( ) Australia
( ) Europe

In what state is your organization primarily located?
{Choose one}
( ) Alabama
( ) Alaska
( ) Arizona
( ) Arkansas
( ) California
( ) Colorado
( ) Connecticut
( ) Delaware
( ) District of Columbia
( ) Florida
( ) Georgia
( ) Hawaii
( ) Idaho
( ) Illinois
( ) Indiana
( ) Iowa
( ) Kansas
( ) Kentucky
( ) Louisiana
( ) Maine
( ) Maryland
( ) Massachusetts
( ) Michigan
( ) Minnesota
( ) Mississippi
( ) Missouri

( ) Montana
( ) Nebraska
( ) Nevada
( ) New Hampshire
( ) New Jersey
( ) New Mexico
( ) New York
( ) North Carolina
( ) North Dakota
( ) Ohio
( ) Oklahoma
( ) Oregon
( ) Pennsylvania
( ) Rhode Island
( ) South Carolina
( ) South Dakota
( ) Tennessee
( ) Texas
( ) Utah
( ) Vermont
( ) Virginia
( ) Washington
( ) West Virginia
( ) Wisconsin
( ) Wyoming
( ) US Territories

a. As a consultant or service provider, please answer questions 8-19 in terms of a client you are most familiar with, and questions 20 and 21 with your personal point of view.

b. As an academic, please answer questions 8-14 for the organization in which you work and from question 15 on from a theoretical point of view unless you are within the IS/IT functional department of an educational organization. In this case, answer the questions from the point of view of your area of responsibility.

c. Please answer questions 8-19 about the organization or branch in which you work and questions 20 and 21 with your personal point of view.

Please enter your ZIP code:
{Enter text answer}
[ ]

Part III. Please answer the following questions about the Information Technology (IT) "department" in your organization. By "IT department" we mean the functional group (or area) that manages the information assets of some larger organization. By "information assets" we mean the various software, hardware, networks, data, voice, video, and other assets typically managed by IT professionals working together in an organization.

13. What was the total operating budget of this IT department during the last fiscal year? This includes all money spent providing information technologies and services -- people, communications, hardware, software, maintenance, outsourcing contracts, and any other directly-related items.
{Choose one}
( ) Less than $100, 000
( ) $100,000 - $249,999
( ) $250,000 - $499,999
( ) $500,000 - $999,999
( ) $1 million - $9.9 million
( ) $10 million - $49.9 million
( ) $50 million - $ 99.9 million
( ) $100 million - $499.9 million
( ) $500 million - $1 billion
( ) Greater than $1 billion

14. How many people work for this IT department? (Please include all IT professionals and staff, as well as both direct and dotted-line reports. (Include outsourced, consultants, or contract personnel.)
{Choose one}
( ) Less than 50
( ) 50-99
( ) 100-499
( ) 500-999
( ) 1000-4999
( ) 5000-9999
( ) 10,000-19,999
( ) 20,000-30,000
( ) Greater than 30,000

For the following statements choose the ONE answer that best describes your response.
15. For software development and/or maintenance, our IS department specifies and uses a comprehensive set of processes and/or procedures for:

a. Establishing customer agreement on requirements:
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Not Applicable/Don't Know

b. Identifying the training needs of IS professionals
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Not Applicable/Don't Know

c. Establishing quality goals with customers
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Not Applicable/Don't Know

d. Estimating all resource needs
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Not Applicable/Don't Know

e. Tracking progress and resource use
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Not Applicable/Don't Know

f. Software quality assurance
{Choose one}
- ( ) Strongly Disagree
- ( ) Disagree
- ( ) Neutral
- ( ) Agree
- ( ) Strongly Agree
- ( ) Not Applicable/Don't Know

g. Continuous process improvement
{Choose one}
- ( ) Strongly Disagree
- ( ) Disagree
- ( ) Neutral
- ( ) Agree
- ( ) Strongly Agree
- ( ) Not Applicable/Don't Know

h. Coordination and communication among stakeholders
{Choose one}
- ( ) Strongly Disagree
- ( ) Disagree
- ( ) Neutral
- ( ) Agree
- ( ) Strongly Agree
- ( ) Not Applicable/Don't Know

i. Selecting, contracting, tracking, and reviewing software contractors/outsourcers
{Choose one}
- ( ) Strongly Disagree
- ( ) Disagree
- ( ) Neutral
- ( ) Agree
- ( ) Strongly Agree
- ( ) Not Applicable/Don't Know

j. Analyzing problems and preventing re-occurrence
{Choose one}
- ( ) Strongly Disagree
- ( ) Disagree
- ( ) Neutral
- ( ) Agree
- ( ) Strongly Agree
- ( ) Not Applicable/Don't Know

k. Tailoring the process to project specific needs
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Not Applicable/Don't Know

l. Continuous productivity improvements
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Not Applicable/Don't Know

16. This IS department aspires to the software development practices of the Software Engineering Institute's (SEI's) Capability Maturity Model for software development:
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Not Applicable/Don't Know

17. Whether your IS department aspires to SEI CMM practices or not, at what level would your IS organization be assessed?
{Choose one}
( ) Initial (Level 1)
( ) Repeatable (Level 2)
( ) Defined (Level 3)
( ) Managed (Level 4)
( ) Optimizing (Level 5)

18. Please select the level to which you agree or disagree that each of the following statements are representative of the requirements analysis and design practices in your IT organization.
The purpose of requirements analysis and design (RA&D) is to describe a functional process or a product/service in order to achieve enterprise objectives.

My organization's requirements analysis and design (RA&D) efforts and activities: ________________

a. are measured.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

b. are benchmarked to other organizations.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

c. are aligned with the organization's objectives.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

d. are highly developed and disciplined.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

e. are valued by executive leadership.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

f. have executive leadership buy-in and support.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

g. are characterized by effective communication between executive leadership and the requirements analysis and design team.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

My organization's requirements analysis and design (RA&D) efforts and activities: ________________(cont'd)

h. describe our present 'as is' environment.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

i. describe our "to be" or desired environment.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

j. efforts stifle innovation in our organization.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

k. are viewed strictly as an IT initiative.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

l. improve ability to manage risk.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

m. contribute directly to the goals and objectives of our business plan.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

n. have IT leadership buy-in and support.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

o. are well prioritized by executive leadership.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

Page 4

Part IV. Please answer questions 19 and 20 with your personal opinion indicating your degree of agreement or disagreement with the following statements about enterprise architecture. There are no right or wrong answers:

19. The purpose/function of enterprise architecture is:

a. to provide a snapshot in time of an organization.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

b. to facilitate systematic change in an organization.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

c. as a tool for communicating organizational objectives.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

d. as a tool for aligning business objectives with IT initiatives.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

e. to provide a blueprint of an organization's business, data, applications, and technology.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

f. as a tool for planning.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

g. as a tool for decision making.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

20. Please select the level to which you agree or disagree that each of the following statements are representative of the potential benefits to an organization from doing enterprise architecture.

Potential benefits of EA:

a. Aligning business objectives with information technology investments.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

b. More responsive to change.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

c. Less wasted time or money on projects which do not support business goals or objectives.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

d. Better situational awareness.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

e. More effective at meeting business goals.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

f. Improved utilization of information technology.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

g. Improved organizational communications and information sharing.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

h. Improved interoperability among information systems.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

i. Improved ROI from IT spending.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

j. Improved communications between the organization and the information systems department.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

Potential benefits of EA (cont'd):
k. Faster at developing and implementing new information systems.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

l. Improves information systems security across the business.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

m. Standardizes organizational performance measures.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

n. Better collaboration within organization.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

o. Improves trust in the organization.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

p. Assists with organizational governance.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

q. More effective use of IT resources.
{Choose one}
- ( ) Strongly Disagree
- ( ) Disagree
- ( ) Neutral
- ( ) Agree
- ( ) Strongly Agree
- ( ) Don't Know

r. Reduced IT complexity.
{Choose one}
- ( ) Strongly Disagree
- ( ) Disagree
- ( ) Neutral
- ( ) Agree
- ( ) Strongly Agree
- ( ) Don't Know

s. Improved communications within organization.
{Choose one}
- ( ) Strongly Disagree
- ( ) Disagree
- ( ) Neutral
- ( ) Agree
- ( ) Strongly Agree
- ( ) Don't Know

t. Reduces organizational stovepipes.
{Choose one}
- ( ) Strongly Disagree
- ( ) Disagree
- ( ) Neutral
- ( ) Agree
- ( ) Strongly Agree
- ( ) Don't Know

21. The outcomes or products of my organization's requirements analysis and design (RA&D) activities: ______________________

a. include standards for information systems security.

{Choose one}

( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

b. describe our transition from 'as is' to 'to be'.

{Choose one}

( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

c. are kept current.

{Choose one}

( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

d. are kept in a digital repository or database.

{Choose one}

( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

e. are used to standardize our technologies.

{Choose one}

( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

f. are used to support strategic business decisions.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

g. are approved by the CIO.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

h. are approved by the owner of the relevant business processes.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

i. are used as the basis for IT procurement.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

j. are assessed for their quality.
{Choose one}
( ) Strongly Disagree
( ) Disagree
( ) Neutral
( ) Agree
( ) Strongly Agree
( ) Don't Know

Results and analysis from this survey will be presented at SIMPosium in October 2007, followed by a SIM Whitepaper. If you wish to receive a copy of the preliminary results from this survey, please provide your e-mail here:
{Enter text answer}
[ ]

Thank you for taking the EA Survey. Please click the "Finish" button at the bottom of the survey.

REFERENCES

Abdel-Hamid, T. K., & Madnick, S. E. (1990). The elusive silver lining: How we fail to learn from software development failures. *Sloan Management Review, 32*(1), 39-48.

Adler, P. S., McGarry, F. E., Irion-Talbot, W. B., & Binney, D. J. (2008). Enabling process discipline: Lessons from the journey to CMM Level 5. *MIS Quarterly Executive*, *4*(1), 215-227.

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

Araya, S., Chaparro, J., Orero, A., & Joglar, H. (2007) An integrative view of IS/IT and organizational resources and capabilities, *Issues in Informing Science and Information Technology*, *4*, 1-11.

Bagozzi, R. P., & Yi, Y. (1988). On the evaluation of structural equation models. *Journal of the Academy of Marketing Science*, *16*(1), 74-94.

Baltzan, P., & Phillips, A. (2007). *Business-driven information systems* NY: McGraw Hill/Irwin.

Barki, H., Rivard, S., & Talbot, J. (1993). Toward an assessment of software development risk. *Journal of Management Information Systems, 10*(2), 203-225.

Beam, W. R., Palmer, J. D., & Sage, A. P. (1987). Systems engineering for software productivity. *Systems, Man and Cybernetics, IEEE Transactions, 17*(2), 163-186.

Beecham, S., Hall, T., & Rainer, A. (2005). Defining a requirements process improvement model. *Software Quality Journal, 13*(3), 247-279.

Benbasat, I., Dexter, A. S., Drury, D. H., & Goldstein, R. C. (1984). A critque of the stage hypothesis: Theory and empirical evidence. *Communications of the ACM, 27*(5), 476-485.

Benbasat, I., Dexter, A. S., & Mantha, R. W. (1980). Impact of organizational maturity on information system skill needs. *MIS Quarterly, 4*(1), 21-34.

Benbasat, I., & Zmud, R. (2003). The identity crisis within the IS discipline: Defining and communicating the discipline's core properties. *MIS Quarterly*, *27*(2), 183-194.

Bernard, S. A. (2005). *An introduction to enterprise architecture*. Bloomington, IN: AuthorHouse.

Berry, D. M. (2004). The inevitable pain of software development: Why there is no silver bullet. *Lecture Notes in Computer Science*, *2941,* 50-74.

Berry, D. M. (2008). The software engineering silver bullet conundrum. *IEEE SOFTWARE*, 25(2), 18-19.

Boehm, B. (2006). Some future trends and implications for systems and software engineering processes. *Systems Engineering, 9*(1), 1-19.

Bohn, R. E. (1995). The impact of process noise on VLSI process improvement. *, IEEE Transactions on Semiconductor Manufacturing, 8*(3), 228-238.

Brancheau, J. C., Janz, B. D., & Wetherbe, J. C. (1996). Key issues in information systems management: A shift toward technology infrastructure. *MIS Quarterly, 20*(2), 225-242.

Brits, J., & Botha, G. H. K., (2007) Conceptual framework for modeling business capabilities. *Proceedings of the 2007 Informing Science and IT Education Joint Conference,* 151-170.

Brodman, J., & Johnson, D. (1994), What small businesses and small organizations say about the CMM, *IEEE Proceedings of 16th Annual Conference on Software Engineering,* Sorrento, IT, 331-340.

Brooks Jr, F. P. (1995). *The mythical man-month (20th anniversary ed.)*. Boston: Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA.

Browne, G. J. & Rogich, B. (2001). An empirical investigation of user requirements elicitation: Comparing the effectiveness of prompting techniques. *Journal of Management Information Systems, 17*(4), 223-249.

Cameron, B. (2007). Seeing eye to eye on IT -- Until IT and business executives are in sync about priorities, IT 'maturity' will continue to be challenged. *Optimize, 6*(6), 27-42.

Cattaneo, F., Fuggetta, A., & Sciuto, D. (2001). Pursuing coherence in software process assessment and improvement. *Software Process: Improvement and Practice*, 6(1), 3-22.

Chan, Y. E., Huff, S. L., Barclay, D. W., & Copeland, D. G. (1997). Business strategic orientation, information systems strategic orientation, and strategic alignment. *Information Systems Research, 8*(2), 125-141.

Charette, R. N. (2005). Why software fails. *IEEE Spectrum, 42*(9), 36-42.

Chen, P., & Pozgay, A. (2002) Architecture practice: A fundamental discipline for information systems, *Australasian Conference on Information Systems (ACIS)*, *9* 1-12

Cheng, B. H. C., & Atlee, J. M. (2007). Research directions in requirements engineering. *IEEE Future of Software Engineering* (FOSE '07), 1-18.

Churchill, G. A. (1979). A paradigm for developing better measures of marketing constructs. *Journal of Marketing Research, 16*(1), 64-73.

Chrissis, M. B., Konrad, M., & Shrum, S. (2003). *CMMI: Guidelines for process integration and product improvements,* Addison-Wesley Professional.

Clark, B. K. (2000). Quantifying the effects of process improvement on effort. *IEEE Software*, 17(6), 65-70.

Cook, M. A. (1996). *Building enterprise information architectures: reengineering information systems,* Prentice-Hall, Inc. Upper Saddle River, NJ, USA.

Cox, B. (1995). No silver bullet revisited. *American Programmer Journal*, *17,* 85-92.

Crosby, P. B. (1979). *Quality is free: The art of making quality certain*, New York: New American Library.

Croteau, A. M., Solomon, S., Raymond, L., & Bergeron, F. (2001). Organizational and technological infrastructures alignment. *Proceedings of the 34th Annual Hawaii International Conference on System Systems*, 9-18.

Croteau, A.-M., & Bergeron, F. (2001). An information technology trilogy: business strategy, technological deployment and organizational performance. *The Journal of Strategic Information Systems, 10*(2), 77-99.

Curtis, B., Hefley, W. E., & Miller, S. A. (2001). *People capability maturity model (P-CMM) Version 2.0*. Pittsburgh: Software Engineering Institute.

Damian, D., & Chisan, J. (2006). An empirical study of the complex relationships between requirements engineering processes and other processes. *IEEE Transactions on Software Engineering*, *12,* 433-453.

Davis, A. M., & Hsia, P. (1994). Giving voice to requirements engineering. *IEEE Software, 11*(2), 12-16.

Davis, G. (1982). Strategies for information requirements determination. *IBM Systems* Journal, *1*(21), 4-30.

Dion, R., & Raytheon, S. (1993). Process improvement and the corporate balance sheet*, IEEE Software, 10*(4), 28-35.

Dreyfus, D., & Iyer, B. (2006). Enterprise architecture: A social network perspective. Paper presented at the *Proceedings of the 39th Hawaii International Conference on System Sciences*, *8*, 1-10.

Duggan, E. W. (2004). Silver pellets for improving software quality. *Information Resources Management Journal, 17*(2), 1-21.

Duggan, E. W., & Reichgelt, H. (2006). *The panorama of information systems quality.* New York: Idea Group Publishing, 1-27.

Duggan, E. W., & Thachenkary, C. S. (2003). Higher quality requirements: Supporting joint application development with the nominal group technique. *Information Technology and Management, 4*(4), 391-408.

El Emam, K., & Madhavji, N. H. (1995). The reliability of measuring organizational maturity. *Software Process Improvement and Practice, 1*, 3-26.

Fenton, N. E., & Neil, M. (1999). A critique of software defect prediction models. *IEEE Transactions on Software Engineering, 25*(5), 675-689.

Finkelstein, C. (2004, July 1). System development dtrategies for 21st dentury enterprises. *The Data Administration Newsletter, 7,* 1-9.

Finkelstein, C. (2007, April 1). Why is enterprise architecture important? *The Data Administrator Newsletter, 4,* 1-14.

Fitzgerald, B., & O'Kane, T. (1999). A longitudinal study of software process improvement. *IEEE Software, 16*(3), 37-45.

Frampton, K. (2007), Information technology architect capabilities: Which are important and can they be improved? *Unpublished PhD Dissertation*, School of Computer Science and Information Technology, RMIT University, Melbourne, Victoria, Australia.

Glass, R. L. (1999). The realities of software technology payoffs. *Communications of the ACM, 42*(2), 74-79.

Goethals, F. G., Snoeck, M., Lemahieu, W., & Vandenbulcke, J. (2006). Management and enterprise architecture click: The FAD(E)E framework. *Information Systems Frontiers, 8*(2), 67-84.

Goldenson, D. R., El-Emam, K., Herbsleb, J., & Deephouse, C. C. (1996). Empirical studies of software process assessment methods. *Elements of Software Process Assessment and Improvement, 10*(1), 177-218.

Goulielmos, M. (2003). Outlining organisational failure in information systems development. *Disaster Prevention and Management: An International Journal, 12*(4), 319-327.

Grant, R. M. (1999). Prospering in dynamically-competitive environments: organizational capability as knowledge integration. *Knowledge and Strategy, 12*(4), 124-139.

Gregor, S., Hart, D., & Martin, N. (2007). Enterprise architectures: enablers of business strategy and IS/IT alignment in government. *Information Technology & People, 20*(2), 96-120.

Greiner, L. E. (1972). Evolution and revolution äs organizations grow. *Harvard Business Review*, 50(4), 37-46.

Hackos, J. A. T. (1997). From theory to practice: Using the information process-maturity model as a tool for strategic planning. *Technical Communication, 44*(4), 369-380.

Hair, B., Black, W., Babin, B., Anderson, R., & Tatham, R. (2006). *Multivariate data analysis* (6th ed.). Englewood Cliffs, NJ: Prentice-Hall.

Halbleib, H. (2004). Requirements management. *Information Systems Management, 21*(1), 8-21.

Hall, T., Beecham, S., & Rainer, A. (2002). Requirements problems in twelve software companies: an empirical analysis. *IEEE Proceedings on Software,* 149 (5), 153-160.

Harter, D. E., Krishnan, M. S., & Slaughter, S. A. (1998). The life cycle effects of software process improvement: a longitudinal analysis. *Management Science*, *49*(6), 784-800.

Harter, D. E., Krishnan, M. S., & Slaughter, S. A. (2007). Effects of process maturity on quality, cycle time, and effort in software product development. *Management Science, 58*(4), 451-466.

Harter, D. E., & Slaughter, S. A. (2000). Process maturity and software quality: a field study. *Proceedings of the twenty first international conference on Information systems*, 12, 407-411.

Hay, D. C. (2003). *Requirements analysis: From business views to architecture*. Upper Saddle River, NJ: Prentice Hall PTR.

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

Herbsleb, J., Carleton, A., Rozum, J., Siegel, J., & Zubrow, D. (1994). *Benefits of CMM-based software process improvement: Initial results.* Carnegie Mellon University, Software Engineering Institute.

Hirvonen, A., & Pulkkinen, M. (2004). A practical approach to EA planning and development: The EA management grid. *BIS*, 12, 284-302.

Hirschheim, R., & Klein, H. K. (2003). Crisis in the IS field? A critical reflection on the state of the discipline. Journal of the Association for Information Systems, *4(*5), 237-293.

Hu, L., & Bentler, P. M. (1999). Cutoff criteria for fit indexes in covariance structure analysis: Conventional criteria versus new alternatives. *Structural Equation Modeling, 6*(1), 1-55.

Humphrey, W. S. (1988). Characterizing the software process: a maturity framework. *Software, IEEE Software, 5*(2), 73-79.

Humphrey, W. S., Snyder, T. R., & Willis, R. R. (1991). Software process improvement at Hughes Aircraft. *IEEE Software, 8*(4), 11-23.

Hutchings, A. F., & Knox, S. T. (1995). Creating products customers demand. *Communications of the ACM, 38*(5), 72-80.

Iivari, J. (2001). A dynamic framework for classifying information systems development methodologies and approaches. *Journal of Management Information Systems, 17*(3), 179-218.

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

Janssen, M., & Hjort-Madsen, K. (2007), *Analyzing enterprise architecture in national governments*, Proceedings of the 40th Hawaii International Conference on System Science. (*HICSS' 07),* 1-10.

Johnson, D. L., & Brodman, J. G. (1999). Tailoring the CMM for small businesses, small organizations, and small projects. *IEEE Computer & Society*, 15(5), 1-15.

Jones, C. (2002). Defense software development in evolution. *Crosstalk: The Journal of Defense Software Engineering*, 12(9), 26-29.

Jones, C. (2003). Variations in software development practices. *IEEE Software, 20*, 22-27.

Jones, C. (2008). Measuring defect Potentials and defect removal efficiency. *Software Productivity Research, 25,* 289-297.

Jones, C. (2009). Best Practices in Software Engineering; (in preparation) McGraw Hill; 1st edition forthcoming.

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

Kamata, M. I., & Tamai, T. (2007). How does requirements quality relate to project success or failure? *, 8,* 1-24.

Kaner, M. (2004). A capability maturity model for knowledge-based decisionmaking. *Information, Knowledge, & Systems Management, 4*(4), 225-252.

Kappelman, L., (1997), *Year 2000 problem: Strategies and solutions from the fortune 100*, Boston: International Thomson Computer Press.

Kappelman, L., McGinnis, T., Pettit, A., Salmans, B., & Sidorova, A. (2008). *Enterprise architecture: Charting the territory for academic research.* Paper presented at the Proceedings of the Fourteenth Americas Conference on Information Systems Toronto, ON, CA.

Kappelman, L. A., McKeeman, R., & Zhang, L. (2006). Early warning signs of it project failure: The dominant dozen. *Information Systems Management, 23*(4), 31-36.

Kappelman, L. (2009). *The SIM guide to enterprise architecture: Creating the information age enterprise*, Leon Kappelman, Editor, Auerbach: New York, forthcoming.

Koenig, M. (1992). Entering state III: The convergence of the stage hypotheses, *Journal of the American Society for Information Science*, 43 (3), 204-210.

Koenig, M. (2000). Information driven management: The new, but little perceived business zeitgeist*, Libri*, 50, 174-190.

Kim, C. S., & Peterson, D. K. (2003). A Comparison of the Perceived Importance of Information Systems Development Strategies by Developers from the United States and Korea. *Information Resources Management Journal, 16*(2), 1-18.

King, W. R. (1995). Creating a strategic capabilities architecture. *Information Systems Management, 12*(1), 67-69.

Krishnan, M., & Kellner, M. (2003). Measuring process consistency: Implications for reducing software defects. *IEEE Transactions on Software Engineering,* 25 (6), 800-815.

Kuznets, S. S. (1979). Growth, population, and income distribution: selected essays: New York: Norton.

Langenberg, K., & Wegmann, A. (2004). *Enterprise architecture: What aspects is current research targeting*: EPFL Technical Report IC/2004/77, 1-10.

Layman, B. (2005). Implementing an organizational software process improvement program. *IEEE Software Engineering, 2*, 279-288.

Linscomb, D. (2003). Requirements engineering maturity in the CMM. *CROSSTALK The Journal of Defense Software Engineering, 16*(12), 25-28.

Luftman, J. (2000). Assessing business-IT alignment maturity. *Communications of the Association for Information Systems, 4*(14), 1-51.

Luftman, J., & Brier, T. (1999). Achieving and sustaining business-IT alignment. *California Management Review, 42*(1), 109.

Luftman, J., & McLean, E. R. (2004). Key issues for IT executives. *MIS Quarterly Executive, 3*(2), 89-104.

Luftman, J. N., Lewis, P. R., & Oldach, S. H. (1993). Transforming the enterprise: The alignment of business and information technology strategies. *IBM Systems Journal, 32*(1), 198-214.

Lumsden, L. (2007). Business goals count, not organization size. *IEEE Software*, *24,* 54-57.

Luo, H. (2006) Developing the enterprise architecture management guide. *Journal of Enterprise Architecture*, 2(3), 29-31.

Lyytinen, K., & Hirschheim, R. (1988). Information systems failures—a survey and classification of the empirical literature. *Oxford Surveys in Information Technology*, 257-308.

Mabert, V., Soni, and Venkat raramanan, M. (2003). The impact of organization size on ERP implementations in th e US manufacturing sector. *Omega, The International Journal of Management Science*, 31, 235-246.

Marsh, H. W., Hau, K. T., & Wen, Z. (2004). In search of golden rules: Comment on hypothesis-testing approaches to setting cutoff values for fit indexes and dangers in overgeneraliz ing Hu and Bentler's (1999) findings. *Structural Equation Modeling, 11*(3), 320-341.

McConnell, S. (1999). Software engineering principles. *IEEE Software, 16*(2), 6-8.

McDonald, M. P. (Sept 2007). The enterprise capability organization: A future for IT, *MIS Quarterly Executive 6*(3), 179-192.

McFarlan, F. W. (1981), Portfolio approach to information systems, *Harvard Business Review, 48,* 142-150.

McNurlin, B. C., & Sprague, R. H. (2006). *Information systems management in practice,* Pearson: NY.

Mylopoulos, J., & Castro, J. (2000). Tropos: A framework for requirements-driven software development. *Information Systems Engineering: State of the Art and Research Themes, Springer-Verlag*, 261-273.

Nelson, R. (2005). Project retrospectives: Evaluating project success, failure, and everything in between. *MIS Quarterly Executive, 4*(3), 361-372.

Nguyen, L., & Swatman, P. A. (2003). Managing the requirements engineering process. *Requirements Engineering, 8*(1), 55-68.

Niazi, M. (1999). An instrument for measuring the maturity of requirements engineering process. *Product Focused Software Process Improvement. International Conference, PROFES. Proceedings, 12, 1-12*.

Niazi, M., Cox, K., & Verner, J. (2008). A measurement framework for assessing the maturity of requirements engineering process. *Software Quality Journal, 16*(2), 213-235.

Niazi, M., Wilson, D., & Zowghi, D. (2005). A maturity model for the implementation of software process improvement: an empirical study. *Journal of Systems and Software, 74*(2), 155-172.

Nidumolu, S. R., & Subramani, M. R. (2003). The matrix of control: Combining process and structure approaches to managing software development. *Journal of Management Information Systems, 20*(3), 159-196.

Niederman, F., Brancheau, J. C., & Wetherbe, J. C. (1991). Information systems management issues for the 1990s. *MIS Quarterly, 15*(4), 474-499.

Niemi, E., & Ylimäki, T. Defining enterprise architecture risks in business environment. *Software Research Paper*, 8(9), 1-3.

Nieto, M., Lopéz, F., & Cruz, F. (1998). Performance analysis of technology using the S curve model: the case of digital signal processing (DSP) technologies. *Technovation, 18*(6-7), 439-457.

Nolan, R. L. (1973). Managing the computer resource: a stage hypothesis. *Communications of the ACM, 16*(7), 399-405.

Nolan, R. L. (1979). Managing the crises in data processing, *Harvard Business Review*: *49,* Harvard Business School Publication Corp.

Nuseibeh, B., & Easterbrook, S. (2000). Requirements engineering: a roadmap. *Proceedings of the Conference on the future of software engineering, 8,* 35-43*.*

Oh, W., & Pinsonneault, A. (2007). On the assessment of the strategic value of information technologies: Conceptual and analytical approaches. *MIS Quarterly*: *3*(3), 127-145.

Okoli, C., & Pawlowski, S. (2004). The Delphi method as a research tool: An example, design considerations and applications. *Information & Management*, *42*, 15-29.

Orlikowski W., & Iacono, C. (2001). Research commentary: Desperately seeking the IT in IT research-A call to theorizing the IT artifact. *Information Systems Research*, *12* (2), 121-134.

Paulk, M., Weber, C., Curtis, B., & Chrissis, M. (1993). The capability maturity model. *Version,* 1, 18–27.

Petkov, D., & Petkova, O. (2008). On broadening software development productivity research to serve better software engineering management. *International Conference on Information Resource Management (Conf-IRM 2008),* 1-15.

Pfleeger, S. L. (1995). Maturity, models, and goals: How to build a metrics plan. *The Journal of Systems & Software, 31*(2), 143-155.

Pino, F. J., García, F., & Piattini, M. (2008). Software process improvement in small and medium software enterprises: a systematic review. Software Quality Journal, 16(2), 237-261.

Pulkkinen, M., & Hirvonen, A. (2005). *EA planning, development and management process for agile enterprise development*. System Sciences, 2005. HICSS'05. Proceedings of the 38th Annual Hawaii International Conference on Software Systems, 1-10.

Putnam, L. (1994). The economic value of moving up the SEI scale. *Managing System Development, 7*, 98-102.

Radice, R. A., Harding, J. T., Munnis, P. E., & Phillips, R. W. (1985). A programming process study. *IBM Systems Journal, 24*(2), 91-101.

Radice, R. A., Roth, N. K., O'Hara, A., & Ciarfella, W. A. (1985). A programming process architecture. *IBM systems journal, 24*(2), 79-90.

Rehkopf, T. W., & Wybolt, N. (2003). Top 10 architecture land mines [enterprise architecture]. *IT Professional*, *5*(6), 36-43.

Reich, B. H., & Benbasat, I. (1996). Measuring the linkage between business and information technology objectives. *MIS Quarterly, 20*(1), 55-82.

Reifer, D. J. (2000). The CMMI: it's formidable. *The Journal of Systems & Software, 50*(2), 97-98.

Richardson, I. & von Wangenhiem, C. (2007). Guest editors' introduction: Why are small software organizations different? *IEEE Software, 28*, 18-22.

Rogoway, P. (1998). How to reap the business benefit from SPI: Adding SPICE while preserving the CMM. *SPI Newspaper-European Software Process Improvement, 6*(2), 45-49.

Rohloff, M. (2005), *Enterprise architecture-framework and methodology for the design of architectures in the large*, European Conference on Information Systems, 1-15.

Rosemann, M., & de Bruin, T. (2005). *Towards a business process management maturity model, 12*. Proceedings of the 13th European Conference on Information Systems (ECIS 2005). Regensburg, Germany, 1-26.

Ross, J. W., Beath, C. M., & Goodhue, D. L. (1995). *Developing long-term competitiveness through information technology assets. 14,* Center for Information Research Sloan School of Management, Massachusetts Institute of Technology.

Ross, J. W., Peter, W., & Robertson, D. (2006). *Enterprise architecture as strategy*: Boston: Harvard Business School Press.

Ross, J. W., Sloan School of, M., & Center for Information Systems, R. (2003). *Creating a Strategic IT Architecture Competency: Learning in Stage, 9.* Center for Information Research Sloan School of Management, Massachusetts Institute of Technology.

Saha, P. (2006). A real options perspective to enterprise architecture as an investment activity. *Journal of Enterprise Architecture*, *2*(3), 1-18.

Sauer, C. (1999). *Deciding the Future for IS Failures Not the Choice You Might Think*: Oxford: IDEA Group.

Sauer, C., & Willcocks, L. P. (2002). The evolution of the organizational architect. *MIT Sloan Management Review*, 43(3), 41-49.

Sawyer, P., Sommerville, I., & Viller, S. (1997). Requirements process improvement through the phased introduction of good practice. *Software Process Improvement and Practice, 3*, 19-34.

Sawyer, P., Sommerville, I., & Viller, S. (1998*). Improving the requirements process.* Proceedings of the 4th International Workshop on Requirements Engineering: Foundation of Software Quality (REFSQ'98), Pisa, Italy, 71-84.

Sawyer, P., Sommerville, I., & Viller, S. (1999). Capturing the benefits of requirements engineering. *IEEE Software, 16*(2), 78-85.

Schekkerman, J. (2004). *How to survive in the jungle of enterprise architecture.* London*:* Frameworks/Trafford.

Schelp, J., & Stutz, M. (2007). A balanced scorecard approach to measure the value of enterprise architecture. *Via Nova Architecture, 4*, 4-12.

Schmidt, R. (1997). Managing Delphi surveys using nonparametric statistical techniques. *Decision Sciences*, *2*8(3), 763-774.

Schmidt, R., Lyytinen, K., Keil, M., & Cule, P. (2001). Identifying software project risks: An international Delphi study. *Journal of Management Information Systems*, *17*(4), 5-36.

Scott, J. E. (2007). Mobility, business process management, software sourcing, and maturity model trends: Propositions for the IS organization of the future. *Information Systems Security, 24*(2), 139-145.

Senge, P. M. (1992). *The fifth discipline: the art and practice of the learning organization.* Century Business, London.

Sherwood, J., Clark, A., & Lynas, D. (2005). *Enterprise security architecture*: New York: CMP Books.

Sidorova, A, Evangelopoulos, N, Valacich, J., & Ramakrishnan, T. (2008). Uncovering the intellectual core of the information systems domain. *MIS Quarterly*, *32*(3), 467-482.

Sledgianowski, D., & Luftman, J. (2005). IT-business strategic alignment maturity. *Cases on Information Technology: Lessons Learned, 5,* 1-21.

Sommerville, I., & Ransom, J. (2005). An empirical study of industrial requirements engineering process assessment and improvement. *ACM Transactions on Software Engineering and Methodology (TOSEM), 14*(1), 85-117.

Staples, M., & Niazi, M. (2005). Systematic review of organizational motivations for adopting CMM-based SPI. *Information and Software Technology, In Press, Corrected Proof*.

Staples, M., Niazi, M., Jeffery, R., Abrahams, A., Byatt, P., & Murphy, R. (2007). An exploratory study of why organizations do not adopt CMMI. *The Journal of Systems & Software, 80*(6), 883-895.

Steghuis, C., Daneva, M., & van Eck, P. (2005). *Correlating architecture maturity and enterprise systems usage maturity to improve business/IT alignment.* Proceedings of REBNITA 2005. First International Workshop on Requirements Engineering for Business Need and IT Alignment, Paris, France, 64-73.

Strigel, W. (2007). In software processes, organization size matters. *IEEE Software, 24*, 55-58.

Swanson, B., & Beath, C. (1989). Reconstructing the systems development organization. *MIS Quarterly, 13*(3), 293-309.

Tabachnick, B. G., & Fidell, L. S. (2007). *Using multivariate statistics* (5th ed.). Boston, MA: Pearson Allyn and Bacon.

Ullman, J. B. (2006). Reviewing the basics and moving forward. *Journal of Personality Assessment, 87(*1), 35-50.

Vaidyanathan, S. (2005). Enterprise architecture in the context of organizational strategy. *BPTrends, 11*, 1-9.

van den Bent, B. (2006). A quality instrument for the enterprise architecture development process, unpublished masters thesis, Utrecht University, Netherlands.

van Den Berg, M., & Van Steenbergen, M. (2007). *Building an enterprise architecture practice: Tools, tips, best practices, ready-to-use insights*. Netherlands: Springer Verlag.

van der Raadt, B., Hoorn, J. F., & van Vliet, H. (2005). Alignment and maturity are siblings in architecture assessment. *Advanced Information Systems Engineering, 7,* 357-371.

van der Raadt, B., Soetendal, J., Perdeck, M., van Vliet, H., Universiteit, V., & Amsterdam, N. (2004). Polyphony in architecture. *Software Engineering, 2004. ICSE 2004. Proceedings. 26th International Conference on*, 533-542.

Wallace, L., Keil, M., & Rai, A. (2004). How software project risk affects project performance: An investigation of the dimensions of risk and an exploratory model. *Decision Sciences, 35*(2), 289-321.

Wasserman, A. I., Freeman, P., & Porcella, M. (1983). Characteristics of software development methodologies. *Information System Design Methodologies: A Feature Analysis, TW Olle, HG Sol and C. Tully (Eds.), North-Holland*, 37-62.

Weber, R. (1987). Editors Comments. *MIS Quarterly*, *11*(3) p. iii-iv.

Wegmann, A., Balabko, P., Lê, L. S., Regev, G., & Rychkova, I. (2005*). A method and tool for business-IT alignment in enterprise architecture.* 17th CAiSE Forum, Porto, Portugal, June, 1-24.

Wieringa, R. J., Blanken, H. M., Fokkinga, M. M., & Grefen, P. W. P. J. (2003). Aligning application architecture to the business context. *Advanced Information Systems Engineering, 8,* 1028-1029.

Willcocks, L., Reynolds, P., & Feeny, D. (2007) Evolving IS capabilities to leverage the external IT services market. *MIS Quarterly Executive*, *6*(3), 127-145.

Wilson, D. W. (1997). *Maturity models in IS development.* Paper presented at the Proceedings of the 8th IRMA International Conference, Chicago, IL, 1-33.

Winter, R., & Fischer, R. (2006). *Essential layers, artifacts, and dependencies of enterprise architecture.* EDOC Workshop on Trends in Enterprise Architecture Research (TEAR 2006), October 17 2006, Hong Kong, 1-29.

Xia, W., & Lee, G. (2003). Complexity of information systems development projects: conceptualization and measurement development. *Journal of Management Information Systems, 22*(1), 45-83.

Xia, W., & Lee, G. (2004). Grasping the complexity of IS development projects. *Communications of the ACM, 47*(5), 68-74.

Yeo, K. T. (2002). Critical failure factors in information system projects. *International Journal of Project Management, 20*(3), 241-246.

Young, C. (2001). The unexpected case for enterprise IT architectures. *Gartner Research, No. SPA-12-7101*, 44-53.

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

Zachman, J. A. (1998). The framework for enterprise architecture. *DM REVIEW, 8*, 28-33.

Zachman, J. A. (2005). *The Zachman Framework*. Available from the Zachman Institute for Framework Advancement (ZIFA).

Zachman J. A. (2007), Architecture Is architecture is architecture. *EIMInsight Magazine, 1*(1), 1-9.

Zahran, S. (1998). Software process improvement. Boston, MA: Addison-Wesley.

Zarvic, N., & Wieringa, R. (2006). *An integrated enterprise architecture framework for business-IT alignment.* Paper presented at the Proceedings of the Workshop and Doctoral Consortium of the 19th International Conference on Advanced Information Systems Engineering (CAiSE'07). (1), Trondheim, Norway, 262-270.

Zmud, R. (1979), Individual differences and MIS success: A review of the empirical literature, *Management Science*, *25*(10), 966-979.