

Running head: AGILE METHODS FOR DEVELOPING INTERNET PRODUCTS

Agile Methods for Developing Internet Products, Customer Satisfaction, and Firm Performance

David F. Rico

## ABSTRACT

The purpose of this document is to serve as a research proposal to recommend a study to examine the effects of using agile methods to develop Internet products on customer satisfaction and firm performance. Agile methods are an approach to developing Internet software, which are characterized by early customer involvement, flexible processes, iterative releases, and self organizing teams. Agile methods are posited as the best approach for developing Internet products as opposed to traditional methods characterized by formal project plans, rigid processes, voluminous documentation, and firm, fixed-price requirements. A survey of 400 managers is suggested to test the hypotheses that the use of agile methods for developing Internet products is linked to scholarly models of customer satisfaction (e.g., technology acceptance, web satisfaction, and online trust) and firm performance. The results of this study may help managers better understand the business effects of using agile methods to develop Internet products.

*Keywords.* Agile methods, traditional methods, software development, Internet products, early customer involvement, flexible processes, iterative releases, self organizing teams, customer satisfaction, technology acceptance, web satisfaction, online trust, firm performance.

## INTRODUCTION

The purpose of this document is to serve as a research proposal to recommend a study to examine the effects of using agile methods to develop Internet products on customer satisfaction and firm performance. Agile methods are an approach to developing Internet software, which are characterized by early customer involvement, flexible processes, iterative releases, and self organizing teams. Agile methods are posited as the best approach for developing Internet products as opposed to traditional methods characterized by formal project plans, rigid processes, voluminous documentation, and firm, fixed-price requirements. A survey of 400 managers is suggested to test the hypotheses that the use of agile methods for developing Internet products is linked to scholarly models of customer satisfaction (e.g., technology acceptance, web satisfaction, and online trust) and firm performance. The results of this study may help managers better understand the business effects of using agile methods to develop Internet products.

Similar, but not identical, studies have empirically examined the use of agile methods for developing Internet products. These studies have implied or even shown limited and indirect linkages between the use of agile methods and primitive constructs of Internet product quality, reliability, and robustness. Some have even gone as far as to link the use of agile methods to basic notions of customer satisfaction and firm performance. In doing so, these studies have demonstrated some empirical evidence of the reliability and validity of these hypothesized inferences and knowledge claims. These studies may reduce the risks associated with attempting to link the use of agile methods to customer satisfaction and firm performance. The uniqueness of this study is a holistic focus on all of the major tenets of agile methods. This study proposes to link the use of agile methods to scholarly and robust models of customer satisfaction and firm performance. Thus, this study may be one of the first holistic examinations of agile methods.

*Purpose and Overview*

Agile methods were a reaction to the rise of traditional software development methods, which were too large, expensive, rigid, and fraught with failure (Highsmith, 2002, p. xv-xxii). Downsizing was the norm and traditional methods were being used by large corporations in decline, rather than young, energetic firms on the rise. Millions of websites were created overnight by anyone with a computer and a modicum of curiosity. The advent of agile methods marked the end of traditional methods in the mind of their creators. Agile methods involve “individuals and interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation, and responding to change over following a plan” (p. xvii). Agile methods are best suited for the Internet, which is characterized by constant change, speed, turbulence, high risk, hyper competition, customer centricity, uncertainty, and disruptiveness (p. 13).

Agile methods emerged with a focus on early customer involvement, flexible processes, iterative releases, and self organizing teams (Agile Manifesto, 2001). Internet technologies such as HTML and Java were powerful new prototyping languages, which enabled smaller teams to build bigger software products in record time. Because they could be built faster, customers could see the finished product sooner and provide earlier feedback, and developers could rapidly refine their products. This gave rise to closed-loop, circular, highly recursive, and tightly knit processes for rapidly creating Internet products, leading to increased customer satisfaction and firm performance. At the same time, traditional methods were maturing, which advocated dozens of best practices, hundreds of principles, rigid project plans and processes, and burdensome documentation. Internet developers sharply reacted to the mismatch of traditional methods and created revolutionary new approaches to developing Internet products called agile methods.

### *Scope and Delimitations*

The scope of this study will be limited to an empirical analysis of using agile methods for developing Internet products, customer satisfaction, and firm size (as shown in Table 1). Only the major factors of agile methods will be examined by this study (e.g., early customer involvement, flexible processes, iterative releases, and self organizing teams). Furthermore, the scope of this study will be limited to an empirical analysis of the links between these factors of agile methods and scholarly models of customer satisfaction and firm performance. The scope of this study will also be limited to an empirical analysis of the use of agile methods for developing Internet products versus operating systems and applications. However, as the latter converge with web technologies, it may difficult to distinguish between Internet and non-Internet software.

Table 1

Scope Delimitations for Linking Agile Methods, Customer Satisfaction, and Firm Performance

Scope	Delimitation
Agile methods	S <sub>1</sub> The scope of this study will be limited to an empirical analysis of using agile methods for developing Internet products
Early customer involvement	S <sub>2</sub> The scope of this study will be limited to an empirical analysis of the links between agile methods and early customer involvement
Flexible processes	S <sub>3</sub> The scope of this study will be limited to an empirical analysis of the links between agile methods and flexible processes
Iterative releases	S <sub>4</sub> The scope of this study will be limited to an empirical analysis of the links between agile methods and iterative releases
Self organizing teams	S <sub>5</sub> The scope of this study will be limited to an empirical analysis of the links between agile methods and self organizing teams
Customer satisfaction	S <sub>6</sub> The scope of this study will be limited to an empirical analysis of the links between agile methods and customer satisfaction
Firm performance	S <sub>7</sub> The scope of this study will be limited to an empirical analysis of the links between agile methods and firm performance

*Rationale and Justification*

The rationale and justification for examining the links between the use of agile methods for developing Internet products, customer satisfaction, and firm performance are convincing. The top 500 U.S. firms spend \$140 billion per year on information technology (“Masters of Technology,” 2004) and much of the annual \$400 billion U.S. defense budget is devoted to information technology as well (Fulghum & Wall, 2004). Furthermore, there are more than 250,000 software projects in the U.S. of which more than 72% have failed or are failing (Standish Group, 2000). By some estimates, two-thirds of all Internet projects in the U.S. use agile methods to produce their wares, which amount to nearly 167,000 projects (Sliwa, 2002). Therefore, executives and managers of Fortune 500 firms may benefit from the knowledge that agile methods may be linked to better customer satisfaction and firm performance.

Erdogmus and Favaro (2003) imply that the use of agile methods is associated with greater business value using real options. Erdogmus and Williams (2003) link the use of agile methods to significant increases in productivity when developing Internet products and services. Favaro (2003) indicates the use of agile methods is linked to improved economic value added (EVA), which is a contemporary measure of firm performance. Both Maurer and Martel (2002) and Muller and Padberg (2002 and 2003) link the use of agile methods to significantly higher programming productivity as well. Cusumano and Selby (1995) and Cusumano and Yoffie (1998) linked Microsoft’s and Netscape’s use of agile methods to their stellar firm performance. And, MacCormack (1998) linked the use of agile methods for developing Internet products to increases in product performance, usability, and reliability. These studies indicate that using agile methods may help alleviate the high U.S. project failure rates and contribute to business value with respect to public and private investments in technology.

### *Relevance and Importance*

The relevance and importance of examining the links between the use of agile methods for developing Internet products, customer satisfaction, and firm performance are high as well. This is because hard evidence of the economic benefits of using agile methods for developing Internet products isn't the only reason executives and managers may benefit from this study. Firms may also garner many substantial indirect economic benefits from using agile methods to develop Internet products as well. For instance, it is a well established fact that the stock market rewards firms with higher market valuations for publicly committing to the use of approaches to quality management techniques such as agile methods (Przasnyski & Tai, 1999). Furthermore, the stock market also rewards firms with higher market valuations for renovating information technology infrastructures with greater investments (Davis, Dehning, & Stratopoulos, 2003). What these studies mean is that a firm's investors may simply reward companies with higher market valuations for simply adopting the use of agile methods for developing Internet products. But, the benefits don't merely stop with potentially higher market valuations associated with public announcements of a firm's commitment to agile methods. Firms may simply benefit from exploiting the use of agile methods in marketing activities, business development proposals, and in their advertising. What this means is that understanding the strategic implications of using agile methods for developing Internet products is extremely relevant and important to managers engaged in business development as well. In other words, merely saying a firm is committed to the use of agile methods may result in more business partnerships, more sales leads, and more contracts for Internet products and services, regardless of the actual commitment to the use of agile methods. For example, potential customers of Internet products and services may exclusively favor firms who outwardly commit to the use of agile methods with new business.

*Significance and Interest*

Examining the links between the use of agile methods for developing Internet products, customer satisfaction, and firm performance may be significant and interesting to a number of stakeholders. These stakeholders may include developers of Internet products, computer scientists, software engineers, and management scientists. Developers may want to use software development approaches that are ideally suited for Internet technologies. Developers want to focus on creating the best possible Internet products with the least amount of pain, bureaucracy, or rigid detail associated with traditional methods. Computer scientists, who are responsible for creating Internet technologies, are interested in software development approaches that enable them to exploit their creations easily, more fully, and without the unnecessary burden of mastering bureaucratic traditional methods. Software engineers, on the other hand, are responsible for creating traditional methods for software development. So this study will help software engineers understand the dynamics of creating Internet products, the agile methods that are ideally suited for creating them, and the reasons traditional methods are not well suited for these technologies. Some of the ultimate stakeholders of this study are management scientists. This is because it is their role and responsibility to conduct empirical studies of various management phenomena such as agile methods and identify the factors and conditions to which they do and do not apply. In other words, management scientists need studies to describe agile methods in scholarly terms, identify their important factors, identify the empirical relationships between their factors, tease out scholarly conceptual models, and, of course, test these theories with qualitative and quantitative research methodologies. That is the goal of this proposed study, which is to test the theoretical basis for using agile methods to develop Internet products and examine their link to scholarly models of customer satisfaction and firm performance.

### *Organization and Outline*

This proposal is organized into seven sections: an introduction, research problem, literature review, research theory, research methodology, conclusion, and references. The introduction describes the context for using agile methods versus traditional methods to develop Internet products. The introduction section consists of subsections for a purpose and overview, scope and delimitations, relevance and importance, significance and interest, and organization and outline. The research problem describes the major issues and delineates the boundaries of this study of agile methods. The research problem section consists of subsections for the research issue, background, questions, goals and objectives, terms and definitions, and assumptions and constraints. The literature review describes the history of software development methods, including the emergence and study of using agile methods. It also describes why this study of agile methods is necessary, based on analysis of existing empirical studies of agile methods. The literature review section consists of subsections for a history of software methods, overview of traditional methods, empirical studies of agile methods, gaps and problem areas in the literature, and need for a new study of agile methods. The research theory graphically depicts the conceptual model along with a scholarly exhibit of the major factors of using agile methods for developing Internet products and the associated hypotheses. The research theory section consists of subsections for a conceptual model, research factors, and research theory. The research methodology describes the use of a quantitative research approach for studying the use of agile methods. The research methodology section consists of subsections for the research design, measures, analysis, results, and timeline. The conclusion section describes potential research findings, contributions, implications, limitations, threats to validity, future research, and recommendations, followed by the references section.

## RESEARCH PROBLEM

The research problem identifies and describes the research issue, background, questions, goals and objectives, terms and definitions, and assumptions for this study. The research problem is simply to investigate, examine, and determine whether the use of agile methods for developing Internet products is linked to better customer satisfaction and firm performance. There is some scant literature that investigates the linkages between the use of agile methods and organizational outcomes, such as customer satisfaction and firm performance. These include Cusumano's and Selby's (1995, 1997) studies of the use of agile methods at Microsoft, Cusumano's and Yoffie's (1998, 1999a, 1999b, 1999c) studies of the use agile methods at Netscape, and other scholarly studies of the use of agile methods among Internet firms (MacCormack, 1998, 2001; MacCormack & Verganti, 2003; MacCormack, Verganti, & Iansiti, 2001).

However, the major tenets, principles, and factors of agile methods had yet to fully evolve and emerge at the time of these writings. And, few studies, if any, examine the effects of all four factors associated with agile methods: a) early customer involvement, b) flexible processes, c) iterative releases, and d) self organizing teams. Furthermore, none of the aforementioned studies link any of these factors to outcome measures, such as customer satisfaction and firm performance. Therefore, this study proposes to analyze the effects of all four factors of agile methods and then empirically link these factors to scholarly models of customer satisfaction and firm performance. Customer satisfaction may be operationalized using time tested and empirically reliable and valid models such as technology acceptance, web satisfaction, and online trust. And, of course, the same is true for firm performance, which may be operationalized using outcome measures such as Tobin's Q, economic value added, or other scholarly measures of productivity, profitability, and financial performance.

*Research Issue*

The research issue is to determine whether the use of agile methods for developing Internet products is linked to better customer satisfaction and better firm performance. Agile methods are posited as a revolutionary new approach to developing Internet products. Powerful new and easy to use information age technologies have emerged, which enable less skilled computer programmers to develop Internet products and services at seemingly break neck speeds using rapid prototyping techniques called agile methods. Agile methods result in faster product development, faster time to market, and earlier market feedback, which enable computer programmers to rapidly perfect their Internet wares, thus resulting in greater customer satisfaction and greater firm performance. Internet technologies are so powerful and easy to use that mere novices are able to develop professional Internet products and services using intuitive, informal, seat-of-the-pants, back-of-the-napkin, and shoot-from-the-hip management techniques.

Traditional methods for software development were created when the first commercial computers began emerging in the 1950s. Scientists and engineers began creating increasingly more powerful and complex computers, and subsequently, inordinately complex computer programs, which were beyond the comprehension of a single human. These early computer programs had millions of components to perform the simplest of operations, therefore giving rise to traditional methods for software development. Traditional methods consisted of rigidly formal project management techniques, customer requirements etched in stone, infinitely bureaucratic processes, hundreds of paper documents, and squeaky clean testing processes similar to those used in microbe free semiconductor manufacturing clean rooms. Computer programmers simply came to the realization that the old rules don't apply any more, thus creating a revolution of agile methods for developing Internet products.

### *Research Background*

The research background is characterized by numerous attempts to link the use of software methods to improved customer satisfaction and firm performance. Computer programmers, computer scientists, software engineers, and management scientists have been trying to solve computer programming problems such as productivity, quality, reliability, customer satisfaction, cost effectiveness, and time-to-market for more than five decades. Knuth (1963) and Dijkstra (1968) introduced flowcharts and stepwise refinement to help ease the process of creating complex computer programs in the 1960s. Fagan (1976) introduced software inspections to increase quality and productivity by an order of magnitude in the 1970s. Madden and Rone (1984) used iterative releases to develop the software for NASA's space shuttle in the 1980s. Sulack, Lindner, and Dietz (1989) used early customer involvement to produce 30 billion lines of application software and garner IBM \$14 billion in revenue in the 1980s as well. Hewlett Packard saved \$350 million using a myriad of software methods in the 1990s (Grady, 1997). Motorola successfully produced an error free paging system at 25 times the normal productivity levels (Ferguson, Humphrey, Khajenoori, Macke, & Matvya, 1997). And, Electronic Brokering Services designed a 65,000 line of code Java system using team processes that conducted \$1 billion worth of online trades per day without error in record time (Goth, 2000). General Dynamics has even noted order of magnitude improvements in productivity, quality, cycle time, and cost reductions using software methods (Diaz & Sligo, 1997). Yet, all of these isolated breakthroughs linking software methods to better customer satisfaction and firm performance have left skeptics demanding more empirical evidence (Sassenburg, 2002). This is the background, which establishes the context for seeking empirical evidence that may link the use of agile methods for developing Internet products to customer satisfaction and firm performance.

### *Research Questions*

The research questions relate to the issue of whether the use of agile methods for developing Internet products is linked to customer satisfaction and firm performance. The 1990s and the new millennium have introduced an onslaught of Internet technologies that have transformed industries, the world economy, and computer programming, resulting in highly dynamic and uncertain market conditions (Cao, Gruca, & Klemz, 2003; Carey, 1998; Christensen, Schmidt, & Larsen, 2003; Gandal, 2001; Kauffman & Wang, 2001). And, agile methods are comprised of four basic principles: a) early customer involvement, b) flexible processes, c) iterative releases, and d) self organizing teams (Agile Manifesto, 2001). Customer satisfaction, on the other hand, will later be operationalized as technology acceptance (e.g., willingness to use an Internet product), web satisfaction (e.g., satisfactory experience with using an Internet product), and online trust (e.g., willingness to conduct monetary transactions using an Internet product). Therefore, the basic research questions are whether the use of early customer involvement, flexible processes, iterative releases, and self organizing teams to develop Internet products are linked to customer satisfaction and firm performance (see Table 2).

Table 2

Research Questions for Linking Agile Methods, Customer Satisfaction, and Firm Performance

Factors	Questions
Early customer involvement	Q <sub>1</sub> Is the use of early customer involvement for developing Internet products linked to customer satisfaction and firm performance?
Flexible processes	Q <sub>2</sub> Are the use of flexible processes for developing Internet products linked to customer satisfaction and firm performance?
Iterative releases	Q <sub>3</sub> Are the use of iterative releases for developing Internet products linked to customer satisfaction and firm performance?
Self organizing teams	Q <sub>4</sub> Are the use of self organizing teams for developing Internet products linked to customer satisfaction and firm performance?

### *Research Goals and Objectives*

The research goals and objectives are to objectively gather information that might determine if a link exists between the use of agile methods for developing Internet products and better customer satisfaction and firm performance. The major factors of agile methods are: a) early customer involvement, b) flexible processes, c) iterative releases, and d) self organizing teams. The major factors of customer satisfaction for this study are: a) technology acceptance, b) web satisfaction, and c) online trust. The major factors of firm performance may include: a) Tobin's Q; b) economic value added; c) or other measures of productivity, profitability, and financial performance. Therefore, the research goals and objectives are to examine the empirical links between the theoretical factors of agile methods, scholarly constructs of customer satisfaction, and significant determinants of firm performance (see Table 3). Conversely, the research goals and objectives of this study are to determine if early customer involvement, flexible processes, iterative releases, or self organizing teams are not linked to scholarly models of customer satisfaction and firm performance. Positive correlations between the factors of agile methods, customer satisfaction, and firm performance are just as important as negative ones.

Table 3

Research Objectives for Linking Agile Methods, Customer Satisfaction, and Firm Performance

Factors	Objectives
Early customer involvement	O <sub>1</sub> Determine if early customer involvement for developing Internet products linked to customer satisfaction and firm performance
Flexible processes	O <sub>2</sub> Determine if use of flexible processes for developing Internet products linked to customer satisfaction and firm performance
Iterative releases	O <sub>3</sub> Determine if use of iterative releases for developing Internet products linked to customer satisfaction and firm performance
Self organizing teams	O <sub>4</sub> Determine if use of self organizing teams for developing Internet products linked to customer satisfaction and firm performance

### Research Terms and Definitions

The research terms and definitions in Table 4 came from a variety of scholarly sources and represent the principle phrases and expressions that will be used throughout this study. Software development methods are essentially management techniques for organizing the creation of complex, large scale software based computer systems. Agile methods are new management techniques better suited for Internet technologies such as HTML and Java.

Table 4

Terms and Definitions for Linking Agile Methods, Customer Satisfaction, and Firm Performance

Terms	Definitions	Source
1. Agile methods	Software development method using early customer involvement, flexible processes, iterative releases, and self organizing teams	Agile Manifesto (2001)
2. Traditional methods	A focus on processes over teams, documentation over programming, contracts over collaboration, and project plans	
3. Software development	A rigorous software development process consisting of analysis, design, coding, testing, installation, and checkout	IEEE (1999)
4. Internet products	Software running on Internet or World Wide Web using HTML and Java, such as a website, web portal, or e-commerce system	Chen & Heath (2001)
5. Early customer involvement	The solicitation of early customer feedback by including them in the product development process to achieve higher satisfaction	Kaulio (1998)
6. Flexible processes	A development process that is tolerant to design changes, late product changes, or change altogether due to flexible designs	Thomke & Reinertsen (1998)
7. Iterative releases	The act of creating a skeletal computer program followed by the gradual enhancement of successive software implementations	Basili & Turner (1975)
8. Self organizing teams	Non-hierarchical groups with different and complementary skills, who are responsible and accountable for organizational outcomes	Zarraga & Bonache (2005)
9. Customer satisfaction	Software products that are capable, usable, high performing, reliable, installable, maintainable, documented, and available	Kan et al. (1994)
10. Technology acceptance	Information systems theory that models how users come to accept and use technology based on perceived usefulness and ease of use	Davis (1986)
11. Web satisfaction	Product choice, online payment, vender trust, travel, shipping, convenience, ecology, customer relations, and value satisfaction	Torkzadeh & Dhillon (2002)
12. Online trust	An Internet user's belief in a website vendor's competence, benevolence, integrity, dependability, and predictability	McKnight et al. (2001)
13. Firm performance	Management quality, product quality, innovation, long term value, solvency, popularity, social responsibility, and assets measures	Saeed, Hwang, & Grover (2005)

### *Research Assumptions and Constraints*

The research assumptions and constraints are that early customer involvement, flexible processes, iterative releases, and self organizing teams are both the principal factors of agile methods and significant predictors of customer satisfaction (as shown in Table 5). This study also assumes customer satisfaction and firm size are significant predictors of firm performance. Other basic assumptions and constraints of this study are that: a) this study is necessary because there are no comprehensive studies linking all four factors of agile methods to any measure of performance (e.g., individual, team, or organizational outcome), b) the factors of agile methods should be linked to scholarly models of customer satisfaction (e.g., technology acceptance, web satisfaction, and online trust), and c) customer satisfaction should be linked to scholarly models of firm performance (e.g., Tobin's Q, economic value added, or other scholarly measures of productivity, profitability, and financial performance).

Table 5

Basic Assumptions for Linking Agile Methods, Customer Satisfaction, and Firm Performance

Factors	Assumptions
Agile methods	A <sub>1</sub> Early customer involvement, flexible processes, iterative releases, and self organizing teams are relevant factors of agile methods
Early customer involvement	A <sub>2</sub> Early customer involvement is a significant predictor of customer satisfaction and firm performance
Flexible processes	A <sub>3</sub> Flexible processes are a significant predictor of customer satisfaction and firm performance
Iterative releases	A <sub>4</sub> Iterative releases are a significant predictor of customer satisfaction and firm performance
Self organizing teams	A <sub>5</sub> Self organizing teams are a significant predictor of customer satisfaction and firm performance
Customer satisfaction	A <sub>6</sub> Customer satisfaction is a significant predictor of firm performance
Firm size	A <sub>7</sub> Firm size is a significant predictor of firm performance

## LITERATURE REVIEW

The literature review provides a scholarly basis for studying of the links between the use of agile methods for developing Internet products, customer satisfaction, and firm performance. The literature review describes the history of software methods as a means of identifying the earliest antecedents of agile methods. Traditional and agile methods are also described since agile methods are posited as a replacement for traditional methods. Several empirical studies are examined describing the effects of using agile methods for developing Internet products. Gaps are identified in order to show the need for a new empirical study of the links between agile methods, customer satisfaction, and firm performance.

### *History of Software Methods*

The history of software methods dates back to the emergence of the first commercial mainframe computers in the 1960s (Greenstein & Wade, 1998). Flowcharting was one of the earliest software methods to emerge, which was a set of graphical symbols used to interpret, document, and communicate complex computer programs to humans (Knuth, 1963). The purpose of software methods changed over the decades to meet the evolving needs of computer programmers. Structured design made easy the task of creating computer programs in the early 1970s (Dijkstra, 1968). Computer aided software engineering (CASE) tools emerged in the 1980s as a means of attempting to fully automate the software development process (Hoffnagle & Beregi, 1985). Personal processes emerged during the 1990s as a means of teaching computer programmers how to apply project management practices to computer programming (Humphrey, 1996). Finally, agile methods were created in the 2000s as a means of quickly creating computer programs in the rapidly evolving field of information technology (Highsmith, 2002, p. xxii). Figure 1 shows 32 major classes of software methods that have emerged over the last 50 years.

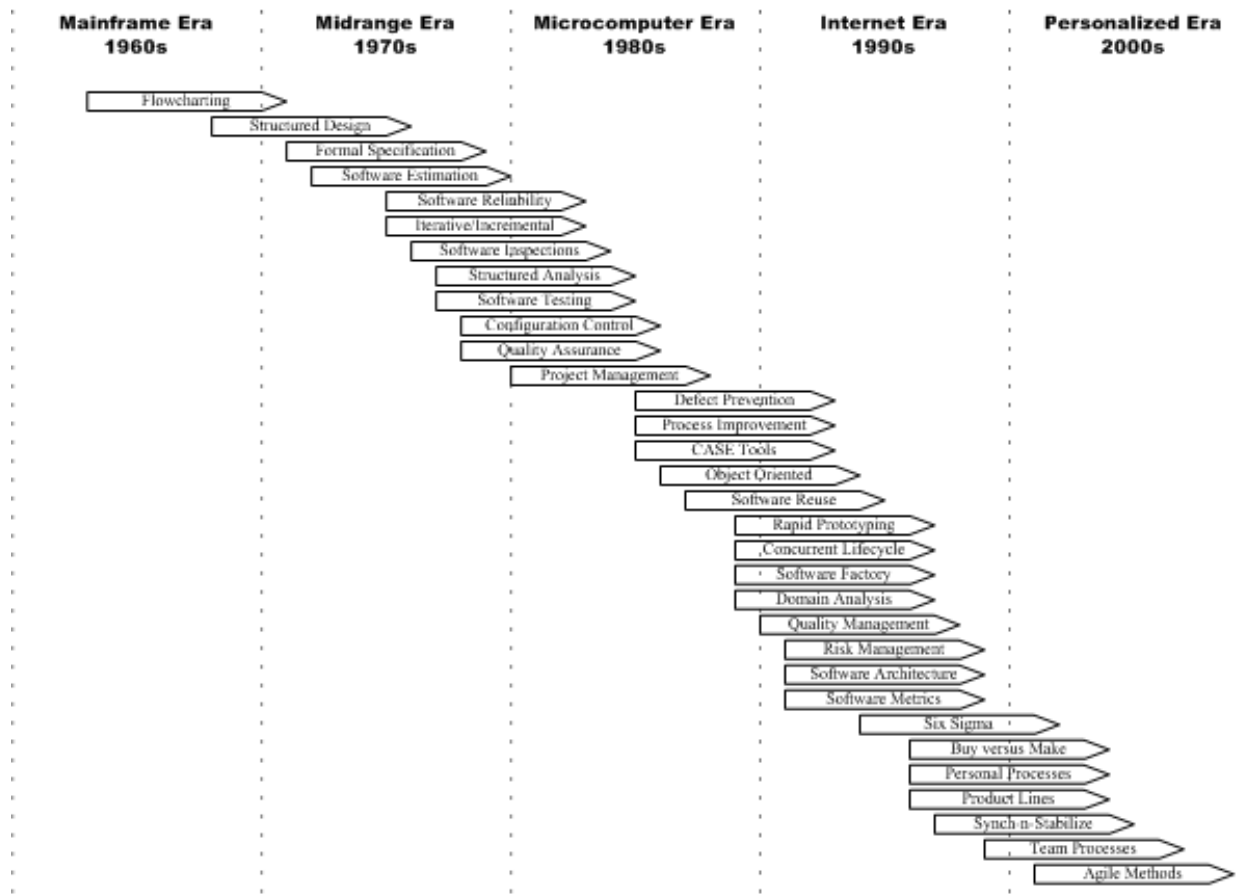


Figure 1. Timeline marking the history of software methods since 1960.

### Overview of Traditional Methods

Traditional methods are software development processes “by which user needs are translated into a software product” (Institute of Electrical and Electronics Engineers, 1990). Furthermore, the traditional method “involves translating user needs into software requirements, transforming the software requirements into design, implementing the design in code, testing the code, and sometimes, installing and checking out the software for operational use” (Institute of Electrical and Electronics Engineers). Modern traditional methods recommend 12 phases, 35 documents, 62 evaluations, 17 records, 4 audits, 9 reviews, and 9 baselines (International Organization for Standardization/International Electrotechnical Commission, 1995). Traditional methods are designed to ensure computer programs are created successfully (see Figure 2).

Phase	System Requirements Analysis	System Architectural Design	Software Requirements Analysis	Software Architectural Design	Software Detailed Design	Software Coding and Testing	Software Integration	Software Qualification Testing	System Integration	System Qualification Testing	Software Installation	Software Acceptance Support
Product	• SRS	• SARAD	• SRD	• SAD • SIDD (t) • DDD (t) • UDD (p) • TVPL (si)	• SDD • SIDD (d) • DDD (d) • UDD (u) • TVPL (su) • TVPL (siu)	• Software • TVPR (su) • TVRR (su) • UDD (u) • UDD (u) • TVPR (sq)	• SOIP • TVRR (si) • UDD (u) • TVPR (sq)	• TVRR (sq) • UDD (u) • SIAR (sfc) • SIAR (spc)	• TVRR (yi) • TVPR (yq)	• TVRR (yq) • SIAR (yfc) • SIAR (ypc)	• SIP	• TVRR (sa) • Training
Evaluation	• Walkthru • Inspection	• Walkthru • Inspection	• Walkthru • Inspection	• Walkthru • Inspection	• Walkthru • Inspection	• Walkthru • Inspection	• Walkthru • Inspection	• Walkthru • Inspection	• Walkthru • Inspection	• Walkthru • Inspection	• Walkthru • Inspection	• Walkthru • Inspection
Record	• SYRER	• SYAER	• SORER	• SOAER	• DDER	• EOCR • SCTRER • SCR	• SIER	• DER • SCR	• SQTER	• SCR • SER • SQTARR	• SIRR	• SCR
Audit									• PCA • FCA	• PCA • FCA		
Review	System Requirements Review	System Design Review	Software Specification Review	Preliminary Design Review	Critical Design Review		Software Test Readiness Review	Software Formal Qualification Review	System Test Readiness Review	System Formal Qualification Review		
Baseline	Functional Baseline		Allocated Baseline	Developmental Configuration			Software Test Baseline	Software Product Baseline	System Test Baseline	System Product Baseline		

PLAN (3)		SYMBOLS (15)	
SIP	Software Installation Plan	(t)	Top-Level
SOIP	Software Integration Plan	(p)	Preliminary
TVPL	Test or Validation Plan	(si)	Software Integration
SPECIFICATION (1)		(d)	Detailed
SRS	System Requirements Specification	(u)	Update
DESCRIPTION (7)		(su)	Software Unit
DDD	Database Design Description	(siu)	Software Integration Update
SAD	Software Architecture Description	(sq)	Software Qualification
SARAD	System Architecture and Requirements Allocation Description	(sfc)	Software Functional Configuration Audit
SDD	Software Design Description	(spc)	Software Physical Configuration Audit
SIDD	Software Interface Design Description	(yi)	System Integration
SRD	Software Requirements Description	(yq)	System Qualification
UDD	User Documentation Description	(yfc)	System Functional Configuration Audit
PROCEDURE (1)		(ypc)	System Physical Configuration Audit
TVPR	Test or Validation Procedures	(sa)	Software Acceptance
REPORT (2)		RECORD (14)	
SIAR	Software Integration Audit Report	DDER	Detailed Design Evaluation Record
TVRR	Test or Validation Results Report	DER	Documentation Evaluation Record
AUDITS (2)		EOCR	Executable Object Code Record
FCA	Functional Configuration Audit	SCR	Source Code Record
PCA	Physical Configuration Audit	SCTRER	Software Code and Test Results Evaluation Record
		SER	System Evaluation Record
		SIER	Software Integration Evaluation Record
		SIRR	Software Installation Results Record
		SOAER	Software Architecture Evaluation Record
		SORER	Software Requirements Evaluation Record
		SQTARR	System Qualification Test Audit Results Record
		SQTER	System Qualification Test Evaluation Record
		SYAER	System Architecture Evaluation Record
		SYRER	System Requirements Evaluation Record

Figure 2. Illustration of a traditional software method (International Organization for Standardization/International Electrotechnical Commission, 1995).

*Waterfall.* The waterfall method was an early attempt at formalizing and professionalizing the development of computer programs (Royce, 1970). It consisted of eight major stages or steps in the development of computer programs: a) system requirements, b) software requirements, c) preliminary program design, d) analysis, e) program design, f) coding, g) testing, and h) operations. The waterfall method was created to ensure large computer programs would be completed successfully and they would meet their time and cost constraints. The waterfall method gave rise to several closely related military and commercial software methods (e.g., International Organization for Standardization/International Electrotechnical Commission, 1995; Space and Naval Warfare Systems Command, 1985, 1994). Agile methods are posited as the antithesis of waterfall methods (Highsmith, 2002, p. 74).

*International Organization for Standardization (ISO) 9001.* ISO 9001 may be applied as a method for developing computer programs in the international community (International Organization for Standardization, 2000). ISO 9001 consists of five clauses or groupings of best practices: a) quality management system; b), management responsibility; c) resource management; d) product realization; and e) measurement, analysis, and improvement. The quality management system clause consists of two best practices in software development: a) general requirements and b) documentation requirements. The management responsibility clause consists of six best practices in software development: a) management commitment; b) customer focus; c) quality policy; d) planning; e) responsibility, authority, and communication; and f) management review. The resource management clause consists of four best practices in software development: a) provision of resources, b) human resources, c) infrastructure, and d) work environment. The product realization clause consists of six best practices in software development: a) planning of product realization, b) customer related processes, c) design and

development, d) purchasing, e) production and service provision, and f) control of monitoring and measuring devices. The measurement, analysis, and improvement clause consists of five best practices in software development: a) general, b) monitoring and measurement, c) control of nonconforming product, d) analysis of data, and e) improvement. There were over 500,000 ISO 9001 registered firms in 2004 (International Organization for Standardization, 2003) and more than 1,100 of those firms were registered specifically for software development (TickIT, 2004). Agile methods are designed to replace ISO 9001 as a software method (Highsmith, 2002, p. xv), whereas ISO 9001 was designed to prevent software developers from using agile methods.

*Software Capability Maturity Model (SW-CMM)*. The SW-CMM was designed as a method for developing computer programs for the U.S. military (Paulk, Curtis, Chrissis, & Weber, 1993). The SW-CMM consists of four levels or groupings of best practices: a) repeatable, b) defined, c) managed, and d) optimizing. The repeatable level consists of six best practices in software development: a) requirements management, b) software project planning, c) software project tracking and oversight, d) software subcontract management, e) software quality assurance, and f) software configuration management. The defined level consists of seven best practices in software development: a) organization process focus, b) organization process definition, c) training program, d) integrated software management, e) software product engineering, f) intergroup coordination, and g) peer reviews. The managed level consists of two best practices in software development: a) quantitative process management and b) software quality management. The optimizing level consists of three best practices in software development: a) defect prevention, b) technology change management, and c) process change management. By 2004, 180 worldwide organizations reported applying all of the best practices associated with the SW-CMM (Software Engineering Institute, 2005). The SW-CMM has now

been replaced by a closely related software method (e.g., Software Engineering Institute, 2002). Agile methods are designed specifically to combat the spread of software methods like the SW-CMM (Highsmith, 2002, p. xv).

*Project Management Body of Knowledge (PMBoK)*. The PMBoK may be applied as a method for developing computer programs (A guide to the project management body of knowledge, 2004, p. 73). The PMBoK consists of nine project management knowledge areas and processes: a) project integration management, b) project scope management, c) project time management, d) project cost management, e) project quality management, f) project human resource management, g) project communication management, h) project risk management, and i) project procurement management. The project integration management process consists of seven best practices in software development: a) develop project charter, b) develop preliminary project scope statement, c) develop project management plan, d) direct and manage project execution, e) monitor and control project work, f) integrate change control, and g) close project. The project scope management process consists of five best practices in software development: a) scope planning, b) scope definition, c) create work breakdown structure, d) scope verification, and e) scope control. The project time management process consists of six best practices in software development: a) activity definition, b) activity sequencing, c) activity resource estimating, d) activity duration estimating, e) schedule development, and f) schedule content. The project cost management process consists of three best practices in software development: a) cost estimating, b) cost budgeting, and c) cost control. The project quality management process consists of three best practices in software development: a) quality planning, b) perform quality assurance, and c) perform quality control. The project human resource management process consists of four best practices in software development: a) human resource planning, b) acquire

project team, c) develop project team, and d) manage project team. The project communication management process consists of four best practices in software development: a) communications planning, b) information distribution, c) performance reporting, and d) manage stakeholders. The project risk management process consists of five best practices in software development: a) risk management planning, b) risk identification, c) quantitative risk analysis, d) risk response planning, and e) risk monitoring and content. The project procurement management process consists of six best practices in software development: a) plan purchase and acquisitions, b) plan contracting, c) request seller responses, d) select sellers, e) contract administration, and f) contract closure. Nearly 3,000 project managers are certified to use the PMBoK every year (PMI, 2004). Unlike the PMBoK, agile methods assume that project scope, schedule, and cost are not predictable (Highsmith, 2002, p. xv).

*Other traditional methods.* Other notable software methods, which represented the culmination of traditional software development, included the Personal Software Process (PSP) and the Team Software Process (TSP). PSP and TSP are project management methods for individual and group software development (Humphrey, 1996, 2000). PSP consists of time records, defect records, coding standards, size estimates, test reports, task plans, schedule plans, quality estimates, reviews, checklists, formal methods, and cyclic planning. TSP consists of 19 best practices to develop and apply: a) goals, b) roles, c) plans, d) prototypes, e) specifications, f) review processes, g) design reviews, h) interfaces, i) verification tests, j) detailed designs, k) computer programs, l) code reviews, m) unit tests, n) product builds, o) integration tests, p) system tests, q) postmortem reviews, r) role evaluations, and s) cycle reports. Like the PMBoK, PSP and TSP assume that project scope, schedule, and cost are predictable, while agile methods assume the exact opposite (Highsmith, 2002, p. xv).

### Overview of Agile Methods

Agile methods are simply small, lightweight, closed-loop software development processes, which are ideal for creating Internet products and services (Highsmith, 2002, p. 13). They consist of soliciting informal customer needs, quickly translating those needs into working computer programs, releasing beta versions of computer programs to customers, soliciting early customer feedback, and repeating the cycle as often as necessary. Agile methods were expressly created for combating the use traditional methods for developing Internet products and are ideal for exploiting rapid prototyping languages such as HTML and Java (Cusumano & Yoffie, 1999a, 1999b, 1999c). Traditional methods are based on formal project plans with long term schedules, rigidly voluminous software processes, and a focus on labor intensive documentation (Highsmith, p. xv). Agile methods, on the other hand, are ideally suited for rapid, closed-loop, iterative prototyping Internet technologies (Cusumano & Yoffie). The principles of agile methods are illustrated in Table 6 and the major agile methods are described in this subsection.

Table 6

#### Agile Manifesto—12 Principles of Agile Methods for Software Development

Factors	Principles
Early customer involvement	1. Business people and developers must work together daily throughout the project
Flexible processes	2. Changing requirements welcome, even late in development, agile processes harness change for the customer's competitive advantage
	3. Agile processes promote sustainable development, because the sponsors, developers, and users can maintain a constant pace
	4. The highest priority is to satisfy the customer through early and continuous delivery of valuable software
Iterative releases	5. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale

- |                       |  |
|-----------------------|--|
| Self organizing teams | <ol style="list-style-type: none"> <li>6. Working software is the primary measure of progress</li> <li>7. Continuous attention to technical excellence and good design enhances agility</li> <li>8. Simplicity is the art of maximizing the amount of work not done and it is essential</li> <li>9. Build projects around motivated individuals, give them the needed environment and support, and trust them to get the job done</li> <li>10. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation</li> <li>11. The best architectures, requirements, and designs emerge from self-organizing teams</li> <li>12. At regular intervals, the team reflects on how to become more effective and then tunes and adjusts its behavior accordingly</li> </ol> |
|-----------------------|--|

---

*Note.* Adapted from Agile Manifesto (2001).

*Extreme programming (XP).* The XP method was specifically designed to thwart the principles of traditional methods based on the waterfall (Beck, 1999). XP programmers solicit informal customer needs, add value by creating working computer programs on a daily basis, and deliver working computer programs in frequent iterations to solicit early customer feedback. XP consists of 13 best practices: a) planning game, b) small releases, c) metaphors, d) simple design, e) tests, f) refactoring, g) pair programming, h) continuous integration, i) collective ownership, j) on-site customers, k) 40-hour weeks, l) open workspaces, and m) just rules. XP programmers solicit early customer needs from on-site customers during the planning game. Then XP programmers translate customer needs into metaphors, simple designs, and test cases. XP programmers team into pairs, creating working computer programs everyday using continuous integration, refactoring (e.g., redesign), and collective ownership. Finally, XP programmers work no more than 40 hours per week, are co-located in open workspaces to maximize communication, and their practices are open to constant interpretation and improvement.

*Scrum*. The Scrum method is based on the notion that software development is not the defined process represented by traditional methods, but an empirical process of learning as you go along (Abrahamsson, Salo, Ronkainen, & Warsta, 2002; Highsmith, 2002, p. 241-250). In other words, Scrum begins with the assumption that software development is a learning process with goals that must be renegotiated in 30-day increments. Scrum also assumes that traditional methods are based on scientific laws, which do not apply to software development. These so-called scientific laws include: a) methodologies overburdened with phases, steps, tasks, and activities; b) the transformation of inputs to outputs; c) repeatability and minimum variation; d) a linear, top-down, hierarchical, command and control focus; and d) the ability to predict and plan for costs, schedules, and quality. Scrum, on the other hand, is a more flexible and agile method. It consists of: a) holding brief planning meetings every 30 days to review customer requirements, b) 30-day computer programming cycles consisting of creating working computer programs every day, and c) demonstrating a working product to customers every 30 days and repeating the cycle. Scrum is devoid of the tenets of traditional methods (e.g., long and rigorous phases and reviews, the production of numerous documents, and large and unpredictable costs and schedules). Scrum is designed to put a working product in the customer's hands at the earliest possible opportunity, whereas traditional methods seem to lack this principle.

*Dynamic systems development method (DSDM)*. Like other agile methods, DSDM emphasizes rapid and frequent iterations focused on the process of computer programming (Abrahamsson, Salo, Ronkainen, & Warsta, 2002; Highsmith, 2002, p. 251-260). DSDM consists of three major phases: a) creating an early prototype or functional model, b) creating a design prototype, and c) developing the final implementation system. Essentially, DSDM creates three working models of a computer program. The goal of the first prototype is to elicit customer

requirements as early as possible. The goal of the second prototype is to gain a better understanding of the customer requirements. And, the goal of the third iteration is to rapidly deliver an increment of a computer program that satisfies customer needs until the next cycle. DSDM has little focus on the tenets of traditional methods: a) formal processes, b) rigorous documentation, and c) the development of late and ineffective computer programs. Instead, DSDM focuses on the act of computer programming, rapid production of prototypes to solicit early customer feedback, and the production of working computer programs in rapid cycles.

*Other agile methods.* Other agile methods include Crystal Methods, Feature Driven Development (FDD), Lean Development (LD), Adaptive Software Development (ASD), Rational Unified Process (RUP), Open Source Software Development (OSSD), Agile Modeling (AM), and Pragmatic Programming (Abrahamsson, Salo, Ronkainen, & Warsta, 2002; Highsmith, 2002, p. 239-319). Crystal methods are characterized by talented computer programmers, frequent communication, a focus on computer programming, and freedom from traditional methods. FDD is a lot like XP, but it follows a simpler 20-80 rule. That is, 20% of the project resources are used creating an overall system design, building a list of product features, and creating a plan to implement the customer requirements. Finally, 80% of the project resources are used to create a simple design for each requirement, rapidly create working computer programs, and deliver them to customers in order to solicit early feedback. LD, ASD, RUP, OSSD, AM, and Pragmatic Programming are all based on the same basic dynamics of: a) soliciting early customer needs, b) creating informal lists of requirements, c) programming on a daily basis, and d) rapidly delivering working computer programs to customers in order to solicit early feedback. Working computer programs are considered final products, not necessarily prototypes or beta versions, and are posited as suitable for mission critical systems (p. 13).

### *Empirical Studies of Agile Methods*

This section provides an overview of several scholarly empirical studies of agile methods, examining firms such as Microsoft, Netscape, Internet firms, and non-Internet firms. The first two major groups of studies are qualitative case analyses of Microsoft and Netscape carried out by management scholars at MIT's Sloan School of Management. The third major group of studies is a quantitative examination of 15 major Internet firms, including Microsoft, Netscape, Yahoo, Intuit, Altavista, and others, by management scholars at Harvard's Business School. The last group of quantitative empirical studies includes two attitudinal studies and one productivity study of agile methods.

The goal of the qualitative scholarly studies was to uncover software development practices at Microsoft and Netscape without any preconceived notion of what the researchers would find. The goal of the quantitative scholarly study was to empirically examine the link between the use of agile methods for developing Internet products and better performing projects. In the last group of quantitative empirical studies, 85% of respondents indicated high experience levels, 50% reported significant cost reductions, 93% reported productivity increases, 88% reported quality increases, 83% reported increased customer satisfaction, and 95% report high levels of developer satisfaction with agile methods

*Microsoft.* Cusumano and Selby (1995, 1997) conducted a scholarly case study of the early antecedents of agile methods by Microsoft called synch-n-stabilize. The purpose of the synch-n-stabilize approach was to get large computer programs to market quickly using many small parallel teams with a maximum amount of entrepreneurial flexibility, freedom, creativity, and autonomy. Synch-n-stabilize was a seven step approach consisting of parallel programming and testing, use of vision statements as requirements, projects consisting of three or four rapid

iterations, daily programming and integration testing, schedules with hard deadlines, early customer feedback, and use of small computer programming teams. The study attributed the use of the synch-n-stabilize method to impressive growths in revenue and high levels of customer satisfaction. While the study set out to empirically identify software development practices at Microsoft, it did not contain an in-depth quantitative analysis of the major tenets of the synch-n-stabilize approach or their empirical relationships to Microsoft's performance.

*Netscape.* Cusumano and Yoffie (1998, 1999a, 1999b, 1999c) conducted an empirical study of the use of early antecedents of agile methods by Netscape. The purpose of the study was to identify the software methods, which were most appropriate for a fast growing Internet firm using technologies such as HTML and Java. Firm performance was a given, since Netscape was one of the fastest growing companies of the 1990s. The study identified four major principles of Netscape's software methods: a) flexible product designs, b) flexible processes, c) iterative releases, and d) rigorous testing. While the study set out to empirically identify software development practices at Netscape, it did not contain an in-depth quantitative analysis of the major tenets of its software methods or their empirical relationships to Netscape's performance

*Internet.* MacCormack (1998) conducted one of the first scholarly studies to examine the empirical links between the use of agile methods for developing Internet products and better performing projects. The study focused on the use of agile methods by 15 Internet firms including Microsoft, Netscape, Yahoo, Intuit, and Altavista. The study examined three major factors associated with agile methods: a) greater investments in product design, b) early market feedback, and c) use of highly experienced personnel. Better performing projects were operationalized as greater breadth of product features, ease-of-use or usability, and reliability (e.g., mean-time-to-failure). In summary, the study did find an empirical link between the use of

agile methods and better performing projects (MacCormack, 2001; MacCormack, Verganti, & Iansiti, 2001; MacCormack, Kemerer, Cusumano, & Crandall, 2003). However, two of the major tenets of agile methods for developing Internet products were not examined by this study: a) flexible processes (vs. flexible products) and b) self organizing teams. And, the study did not examine the links between agile methods and scholarly models of customer satisfaction, namely: a) technology acceptance, b) web satisfaction, and c) online trust. Furthermore, it did not examine the effects of agile methods on firm performance among Internet firms.

*Other empirical studies of agile methods.* Johnson (2002) conducted a survey to measure attitudes with agile methods among international programmers. His purpose was to determine and report the experiences of international software developers with agile methods. He designed a 10-question survey to measure experience levels, business impact, and market direction with respect to agile methods. Based on 131 respondents, 85% indicated high experience levels, 50% reported significant cost reductions, 93% reported productivity increases, 88% reported quality increases, 83% reported increased customer satisfaction, and 95% report high levels of developer satisfaction with agile methods. Maurer and Martel (2002) analyze the productivity increases associated with using extreme programming as an agile methodology. Their purpose is to demonstrate that agile methods dramatically increase productivity over the use of traditional methods. Their results were obtained from an analysis of a small software company consisting of nine developers, which produced an Internet product over a 16-month period. Succi, Pedrycz, Marchesi, and Williams (2002) conducted a survey to measure attitudes among international computer programmers towards the agile method technique of pair programming. Their purpose was to demonstrate that agile methods enhance job satisfaction within the international community. They designed a 21-item survey and received 108 responses over a six-month

period. Half of the respondents used agile methods and half did not. 19% of respondents were very satisfied with agile methods, while only 9% of respondents were very satisfied with traditional methods. 30% of respondents were satisfied with agile methods, while 32% of respondents were satisfied with traditional methods. Only 9% of the total number of respondents using agile and traditional methods was unsatisfied or very unsatisfied.

#### *Gaps and Problem Areas in the Literature*

The gaps and problem areas associated with empirical scholarly studies of agile methods are numerous. The studies of Microsoft and Netscape only examined the use of iterative releases and self organizing teams and even these were problematic. Microsoft and Netscape had far fewer iterative releases than are advocated by agile methods. And, they did not include a scholarly analysis of early customer involvement and flexible processes. Microsoft and Netscape did not practice the agile method principle of having an on-site customer and they both examined flexible product architectures versus flexible processes. So, no one study cited here examined all four principles of agile methods: a) early customer involvement, b) flexible processes, c) iterative releases, and d) self organizing teams.

None of the studies cited here examined the empirical links between the use of agile methods for developing Internet products and scholarly models of customer satisfaction. These include technology acceptance (Koufaris, 2002), web satisfaction (Torkzadeh & Dhillon, 2002), and online trust (McKnight, Choudhury, & Kacmar, 2002). Only the study by MacCormack (1998) focused on the empirical relationships between the use of agile methods for developing Internet products and product quality. This study only examined a narrow and unscholarly set of variables (e.g., variety of website features, usability, and reliability). These primitive measures are not as robust as modern Internet measures advocated by Torkzadeh and Dhillon.

*Need for a New Study of Agile Methods*

The literature review reveals a need for more research to examine the links between the use of agile methods for developing Internet products, customer satisfaction, and firm performance. The first two groups of qualitative studies by management scientists at MIT took nearly a decade to complete and only examined Microsoft and Netscape. The third group of quantitative studies examined a larger population of 15 firms, but did not examine all of the principles of agile methods or scholarly models of customer satisfaction and firm performance.

Therefore, there are five major areas for further examination: a) a larger scale study, b) the application of quantitative survey methods, c) an examination of all of the principles of agile methods, d) an examination of scholarly models of customer satisfaction, and e) a link between these factors and firm performance. The Johnson (2002) and Succi, Pedrycz, Marchesi, and Williams (2002) studies exhibited the feasibility of conducting large scale surveys of agile methods. The Cusumano and Selby (1995, 1997) studies showed the feasibility of examining the principles of agile methods (e.g., iterative releases and self-organizing teams) in industry and some limited measures of firm performance. MacCormack (1998) showed the feasibility of examining the empirical relationships between agile methods and some limited measures of customer satisfaction.

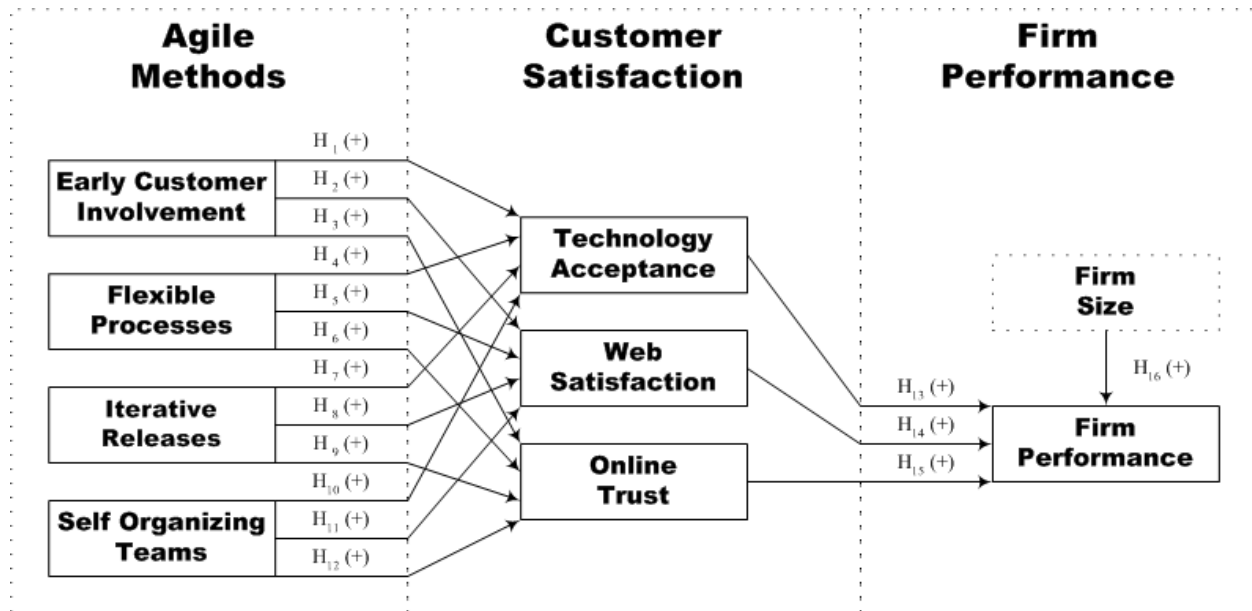
Other scholars are still in the preliminary phases of setting the stage for conducting scholarly empirical studies of the use of agile methods (Lindvall, Basili, Boehm, Costa, Dangle, Tesoriero, Williams, & Zelkowitz, 2002). However, their study isn't aligned with the major principles of agile methods, but rather: a) project size; b) personnel; c) criticality, reliability, and safety; d) training requirements; e) success factors; f) warning signs; and g) refactoring. They will most likely focus on primitive factors such as cost, quality, reliability, and cycle time.

## RESEARCH THEORY

The research theory has been designed to examine the empirical links between the use of agile methods for developing Internet products, customer satisfaction, and firm performance. In particular, it is designed to test the nomological network among the dimensions of agile methods, customer satisfaction, and firm performance. This section consists of a theoretical conceptual model with nine major constructs: a) early customer involvement, b), flexible processes, c) iterative releases, d) self organizing teams, e) technology acceptance, f) web satisfaction, g) online trust, h) firm size, and i) firm performance. This section also consists of a detailed description of each of these factors along with a scholarly examination of their antecedents. Finally, this section identifies 16 hypotheses for examining the relationships between the nine major factors or constructs, along with a scholarly examination of the links between these factors. The research theory establishes the context for the research methodology to follow.

### *Conceptual Model*

The conceptual model exhibited in Figure 3 shows the possible empirical links between the use of agile methods for developing Internet products, customer satisfaction, and firm performance. The major factors of agile methods consisting of early customer involvement, flexible processes, iterative releases, and self organizing teams were adapted from the Agile Manifesto (2001) and its 17 founders. The basic design of the conceptual model is based upon MacCormack, Verganti, and Iansiti (2001), Cusumano and Selby (1995), and Cusumano and Yoffie (1998), lending some support for internal reliability and convergent validity. However, the first objective of this study is to begin providing one of the first complete empirical studies of using agile methods for developing Internet products. Therefore, all four principles of agile methods will be examined, rather than a subset or a superset and treated as major factors.



*Figure 3.* Conceptual model for examining the empirical link between the use of agile methods for developing Internet products, customer satisfaction, and firm performance.

The second objective of the conceptual model exhibited in Figure 3 is to link the use of agile methods for developing Internet products to scholarly models of customer satisfaction. Therefore, the empirical link between the four factors of agile methods and technology acceptance, web satisfaction, and online trust will also be examined. Scholarly models of customer satisfaction will be used to examine these factors. Customer satisfaction will also mediate the effects of using agile methods and firm performance.

Firm size will be used as a control variable to examine its effect on the firm performance. Not a single study in the literature review posited a direct link between the use of agile methods and firm performance. Cusumano and Selby (1995, 1997) and Cusumano and Yoffie (1998, 1999a, 1999b, 1999c) merely assumed there was a link between the use of agile methods and firm performance. Microsoft and Netscape were among the most successful in industry, therefore they assumed agile methods were linked to their performance. However these studies did not empirically test the relationship to firm performance as is proposed here.

Research Factors

As shown in Figure 3, the research factors that will be empirically examined in this study are early customer involvement, flexible processes, iterative releases, self organizing teams, technology acceptance, web satisfaction, online trust, firm size, and firm performance. The earliest antecedents of each of the first seven major factors are shown in Figure 4. The empirical roots of all nine factors relating to the use of agile methods for developing Internet products are examined in the following subsections. Note that the construct of trust predates the formation of online trust and the construct of customer satisfaction predates the formation of web satisfaction.

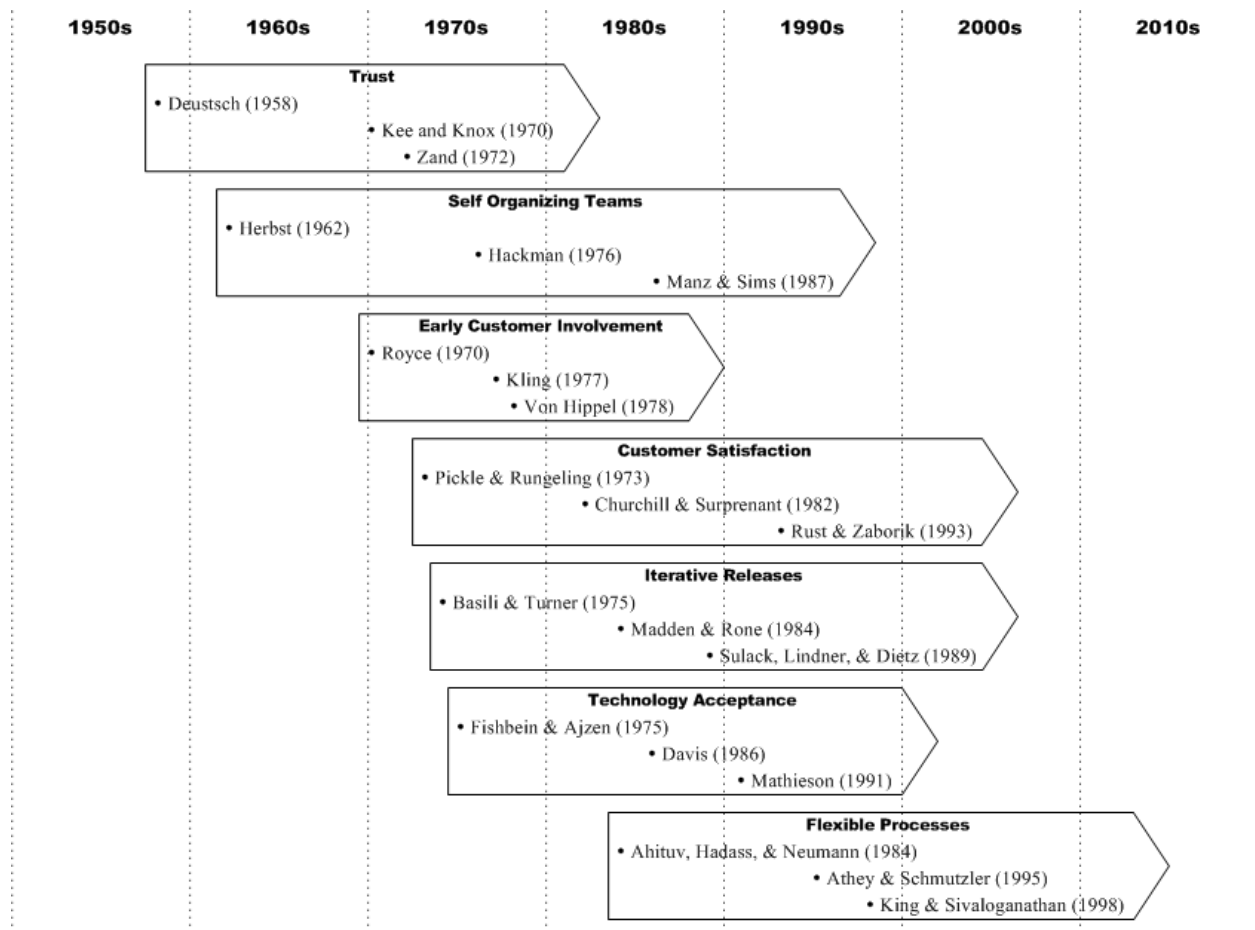


Figure 4. Early antecedents of the factors leading to and from the use of agile methods for developing Internet products.

*Early customer involvement.* Early customer involvement has been recognized as an important goal and objective of using software methods for nearly four decades, as characterized by the inclusion of soliciting requirements from customers prior to computer programming (Royce, 1970). An early empirical study of 473 organizations demonstrated that customers were involved early in the software development process nearly 56% of the time, and ranged as high as 75% to 80% for individual activities (Kling, 1977). Customers were recognized as the source of design innovations in a unique paradigm called Customer Active Participation (Foxall, Johnson, & Murphy, 1987; Von Hippel, 1978). Even more case studies also recognized the central role of early customer involvement in the new product development process (Saren, 1994; Song & Adams, 1993). An empirical study of 110 respondents demonstrated that early customer involvement was an important tool for designing innovative user interfaces for software products and services (Gietzmann & Selby, 1994). The most popular methods of early customer involvement in software development during the 1990s included quality function deployment, user-oriented product development, concept testing, beta testing, consumer idealized design, the lead user method, and participatory ergonomics (Kaulio, 1998). A detailed case study of 11 end users indicated little or no impact to the design and development of software products and services based on early customer involvement (Sugar, 2001). In contrast, an empirical study of 207 firms indicated that the use of quality function deployment, a form of early customer involvement, results in achievement of quality and cost targets for product development (Hoque, Akter, & Monden, 2000). New and holistic methods are now starting to emerge, which advocate early customer involvement in new product development to a wide range of products and services over the Internet (Burnett, Cook, & Rothermel, 2004; Huang & Mak, 2000; Leonard, 2004).

*Flexible processes.* Flexible processes for software development have only recently emerged as an important goal and objective, consisting of iterative or even overlapping product releases, particularly for Internet firms (Aoyama, 1998). An early case study suggested that software methods should vary in rigidity and flexibility based on context specific needs and exhibited a framework for such a purpose (Ahituv, Hadass, & Neumann, 1984). However, flexibility of new product development processes for software as well as other industries has only recently become noticeable, as indicated by a case study of an organization that designed a cost and benefit analysis framework for determining the value of flexibility (Athey & Schmutzler, 1995). In another case study, the use of functional analysis, which is a method of identifying product requirements, prioritizing them, and assigning a monetary value to them, was suggested as means of achieving flexibility in new product development (King & Sivaloganathan, 1998). In an empirical study of 391 projects, the use of technologies that were adaptable to changing customer requirements was attributed to 55% cost reductions in development effort (Thomke & Reinertsen, 1998). In a case study of a \$10 million project at Dell, the use of older inflexible technologies did not prove to be any more economically disadvantageous, because of the risks associated with using newer, innovative, and flexible technologies (Krishnan & Bhattacharya, 2002). A recent empirical study of 40 firms, showed that the use of agile methods, which included early customer involvement, design flexibility, decentralized decision making, and cross functional teams, was correlated to increased competitiveness for Internet products (Kassim & Zain, 2004). Another empirical study of 162 projects from 92 firms, exhibited a correlation between flexible processes (e.g., product, process, and managerial flexibility) and customer satisfaction (e.g., technology acceptance, web satisfaction, and online trust) in high technology settings (Singh & Sushil, 2004).

*Iterative releases.* Iterative releases have been recognized as an important goal and objective of software development for nearly four decades (Basili & Turner, 1975; Royce, 1970). The use of iterative releases by IBM to design and develop spacecraft software was viewed as a key principle of the success of NASA's space shuttle program (Madden & Rone, 1984). IBM once again exploited the use of iterative releases to develop a seven million line of code midrange computer operating system, which helped garner \$14 billion in revenues (Sulack, Lindner, & Dietz, 1989). In one case study, the use of iterative releases for a two year software development project resulted in the delivery of 30% more product functionality with an overall cost reduction of 20% (Woodward, 1999). In another case study of a single firm, 10 of 28 projects that used iterative releases experienced a time-to-market improvement of 28%, cost reductions of 50%, and implemented 125% more product features (Fichman & Moses, 1999). In a study of 161 software projects, the use of iterative releases for small projects increased costs and time-to-market by 15%, while their use on medium and large software projects reduced costs and time-to-market by 15% to 35%, respectively (Benediktsson & Dalcher, 2003, 2004). IBM used iterative releases on four software projects (e.g., a kiosk for the world's fair, an employment management system for the state of Illinois, an automobile loan management system, and a kiosk for New York City's Museum of Modern Art), and attributed improvements in product quality, reductions in training costs, reductions in maintenance costs, and increases in customer satisfaction to their use (Greene, Jones, Matcheri, & Thomas, 2003). MacCormack, Verganti, and Iansiti (2001) attributed the use of iterative releases combined with overlapping stages to improvements in product quality, reliability, and performance, but did not specifically mention customer satisfaction and firm performance. However, Erdogmus and Favaro (2003) showed that the use of iterative releases increased net present value by as much as 300% using real options.

*Self organizing teams.* Self organizing teams, also known as autonomous, composite, self managed, and self directed teams began appearing in literature in the early 1960s (Herbst, 1962). An early Yale study described self organizing teams as groups who have: “a) a relatively whole task; b) members who each possess a variety of skills relevant to the group task; c) worker discretion over such decisions as methods of work, task schedules, and assignment of members to different tasks; and d) compensation and feedback about performance for the group as a whole” (Hackman, 1976). An early empirical study of 276 employees found that in order for self organizing teams to be successful, external leaders, among many other responsibilities, needed to show teams how to: a) perform at a high level, b) discourage poor performance, c) set performance goals, d) monitor their performance, e) establish high expectations, f) and plan their activities (Manz & Sims, 1987). In another early study of 76 self organizing teams, individual participation, role clarity, role acceptance, motivation, job satisfaction, and productivity improved only when independent feedback of group performance was provided (Pearson, 1991). In a study of 126 workers, Thoms, Moore, and Scott (1996) found that high levels of emotional stability, extraversion, openness, agreeableness, and conscientiousness among individuals were significant determinants of the success of self organizing teams. Alper, Tjosvold, and Law (1998) studied 60 self organizing teams and discovered that greater cooperation, rather than competitiveness, led to open mindedness and increased team confidence and effectiveness. In yet another study of 50 self organizing teams, Jordan, Feild, and Armenakis (2002) found that the most significant predictor of team success was what they called, “group potency,” or “a shared belief among team members that they can be effective as a team.” Erdogmus and Williams (2003) show data supporting co-located self organizing teams as a factor in the use of agile methods, though Easley, Devaraj, and Crant (2003) suggest online teams may be more effective.

*Technology acceptance.* The technology acceptance model (TAM) was created in the 1980s to investigate why people do or do not prefer to use new computer systems (Davis, 1986). TAM was based on Fishbein's and Ajzen's (1975) theory of reasoned action (TRA), which posited that "a person's attitude toward a behavior is determined by his or her salient beliefs about consequences of performing the behavior multiplied by the evaluation of those consequences" (Fishbein & Ajzen, 1980). In summary, Davis, Bagozzi, and Warshaw (1989) found from a study of 107 users that people were more likely to use a newly created computer system if they perceived it to be useful, it was easy to use, they had a positive attitude, and they actually intended on using the computer system. Though TAM became a de facto standard throughout the 1990s, Lederer, Maupin, Sena, and Zhuang (2000) were among the first to adapt TAM to the application of Internet products and services. In a study of 163 respondents, they determined that TAM's factors were good predictors of end user acceptance of websites, though they had to translate ease of understanding and ease of finding to ease of use, and information quality to perceived usefulness. Childers, Carr, Peck, and Carson (2001) using a modified TAM found that navigation flexibility, convenience, product information, usefulness, ease of use, and enjoyment were significant predictors of online retail shopping among 274 Internet users. In another adaptation of TAM for the Internet, Koufaris (2002) discovered that product attractiveness, web skills, search mechanisms, website challenges, shopping enjoyment, and good product focus were significant determinants of 280 shopper's intentions to return to an online bookstore. In a recent study of 242 users, Shih (2004) extended TAM to show that web security; perceived quality, ease of use, and usefulness; user satisfaction; and a positive attitude were linked to online shopping. This study may be the first attempt to marry the outcome of using agile methods to the factors associated with an adaptation of TAM for the Internet.

*Web satisfaction.* The use of customer satisfaction measurement dates back nearly four decades, as indicated by an empirical study of 97 firms linking customer satisfaction to profitability (Pickle & Rungeling, 1973). IBM was one of the first firms to begin using factors such as capability, usability, performance, reliability, documentation, and overall system software to measure satisfaction (Kan, 91; Kan, Dull, Amundson, Lindner, & Hedger, 1994; Kan 1995, p. 273-293; Kekre, Krishnan, & Srinivasan, 1995). However, Cho and Park (2001) produced one of the first studies, which identified product information, consumer service, purchase result and delivery, site design, purchasing process, product merchandizing, delivery time and charge, payment options, ease of use, and additional information services as important determinants of customer satisfaction to 485 Internet shoppers. In a study of 568 online shoppers, McKinney, Yoon, and Zahedi (2002) found that information quality (e.g., relevance, understandability, reliability, adequacy, scope, and usefulness) and system quality (e.g., access, usability, entertainment, hyperlinks, navigation, and interactivity) were significant determinants of customer satisfaction. In one of the most sophisticated models to date, Torkzadeh and Dhillon (2002) determined that product choice, online payment method, vendor trust, shopping travel, shopping errors, shopping convenience, ecology, customer relations, and product value were the most important measures of customer satisfaction from a survey of 620 online shoppers. In a rather large survey of nearly 60,000 users, Cao, Gruca, and Klemz (2003) determined that satisfaction with the ordering process (e.g., product information, ease of ordering, product selection, and website performance) led to satisfaction with the price, as well as satisfaction with the fulfillment process (e.g., product representation, on-time delivery, order tracking, and customer support). Again, this study may be one of the first attempts to integrate the outcomes of using agile methods for developing Internet products to scholarly models of web satisfaction.

*Online trust.* The earliest antecedents of online trust began appearing nearly 50 years ago (Deustsch, 1958). Based on these early studies, Kee and Knox (1970) began linking what they termed as previous experience, structural and situational factors, dispositional factors, perception of motives and confidence, and subjective trust or suspicion as the basic determinants of behavioral trust or suspicion. In a landmark study, Zucker (1986) linked modern notions of institutional trust to the influx of immigrants into the U.S. as a result of the industrial revolution. Kicking off the Internet age, Srivastava (1999) developed one of the first models of trust for Internet products and services, consisting of: a) business practice assurance, b) transaction integrity assurance, c) information protection assurance, and d) legal environment assurance. In perhaps the first empirical study of online trust, Gefen (2000) found that familiarity and disposition to trust were linked to product inquiry and purchase among 217 online shoppers. In yet another seminal study of online trust, Lee and Turban (2001) found that perceived integrity, perceived ability, trust propensity, third party recognition, and trust in Internet shopping were significant predictors of consumer trust of Internet shopping among 445 online users. In an extensive study of 288 firms, Ratnasingam and Pavlou (2003) determined that technology trust, trading partner trust, and perception of risk were linked to the perceived benefits and e-commerce performance among business-to-business firms. And, in a study of 274 online shoppers, Pavlou and Gefen (2004) found that perceived effectiveness of feedback mechanisms, perceived effectiveness of escrow services, perceived effectiveness of credit card guarantees, trust in intermediary, trust propensity, positive past experience, and seller's performance were significant predictors of trust, perceived risk, and transaction intentions in the community of sellers. An underlying question may be, "what is the link between the use of agile methods and online trust?" This is one of the unanswered questions that will be examined in this study.

*Firm size.* Firm size as a predictor of firm performance has been studied for over 50 years, when Alexander (1949) tested the hypothesis that smaller firms had similar rates of returns to larger firms, which challenged a deeply held myth of that era. Those results were categorically rejected, when Hall and Weiss (1967) found that larger firms had significantly higher returns, which overturned numerous studies from the mid 20th century. More pertinent to this study, Kowtha and Choon (2001) found that firm size, strategic commitment, prior competencies, competitive intensity, firm age, and website age were related to a firm's commitment to e-commerce in a study of 135 firms. In a study of 400 firms, McDade, Oliva, and Pirsch (2002) found that firm size was a powerful predictor of a firm's adoption of technology, though the technology adoption rate was tempered if it was entirely too radical. In another study of 160 firms, larger firm size was positively correlated to a progressive policy of adopting Internet technologies: a) e-mail adoption, b) web presence, c) prospecting, d) business integration, and e) business transformation (Teo & Pian, 2004). Zhu (2004), in a study of 114 firms, discovered that larger firms have larger investments in information technology, which were significantly correlated to larger investments in e-commerce capabilities and better firm performance. Significantly better e-business, operational, marketing, and adaptability performance was positively correlated to larger firm size in a study of 181 firms by Yang, Yang, and Wu (2005). A study of 1,200 firms revealed little difference between firm size and the ability to generate sales support, trace customer orders, expedite customer service, generate sales leads, generate new customers, enhance company image, assist in supply chain management, develop relationship marketing, improve communications with customers, improve advertising, or build customer loyalty using Internet technologies (Murphy, Poist, Lynagh, & Grazer, 2003). This study will use firm size as a control to mediate the use of agile methods and firm performance.

*Firm performance.* Firm performance has been a topic of discussion within the field of information technology since the advent of the first large scale commercial computers in 1954 (Brabb & Hutchins, 1963). In an early study, the use of information systems had a positive impact on the performance of a 400 person sales force, but not enough to justify the high cost of the computer systems (Lucas, 1975). Bender (1986) identified a positive relationship to firm performance, Cron and Sobol (1983) found inconsistent results, Turner's (1982) study of 58 banks found no correlation, Banker's, Kauffman's, and Morey's (1985) study did find a correlation, and Weill's (1992) study of 33 firms was inconclusive. Mukhopadhyay, Kekre, and Kalathur (1995), on the other hand, reported that Chrysler saved \$220 million from strategic investments in modern Internet technologies. In a national level study, Tam (1998) found that investments in information technology in Southeast Asia had no measurable impact on shareholder return. Evidence is now starting to emerge that the information technology paradox has turned the corner with the advent of modern Internet technologies. Liang, Lin, and Chen (2004) discovered a significant and positive impact between Internet technologies and earnings per share among 60 firms. Furthermore, Lee and Grewal (2004) in a study of 106 firms "show that both the adoption of the Internet as a communications channel and e-alliance formation positively influence firm performance" using Tobin's Q to model firm performance along with cost of capital, return on assets, change in gross domestic product, and the Herfindahl Index as control variables. Finally, in a study of 129 firms, Ravinchandran and Lertwongsatien (2005) identified a correlation between information systems human capital, information technology infrastructure, and information system partnership quality on information systems capabilities, information technology support for core competencies, and firm performance. This study may link the use of agile methods to firm performance through customer satisfaction using Tobin's Q.

### Research Hypotheses

The research hypotheses for examining the relationships between the use of agile methods for developing Internet products, customer satisfaction, and firm performance are shown in Table 7. There are 12 hypotheses linking early customer involvement, flexible processes, iterative releases, and self organizing teams to technology acceptance, web satisfaction, and online trust. And, there are three hypotheses linking technology acceptance, web satisfaction, and online trust to firm performance. Finally, there is only one hypothesis linking firm size to firm performance. The relationships for each of the 16 hypotheses are described below, along with the scholarly underpinnings for each of the relationships.

Table 7

Research Hypotheses for Linking Agile Methods, Customer Satisfaction, and Firm Performance

Factors	Hypotheses
Early customer involvement	H <sub>1</sub> Early customer involvement is linked to technology acceptance
	H <sub>2</sub> Early customer involvement is linked to web satisfaction
	H <sub>3</sub> Early customer involvement is linked to online trust
Flexible processes	H <sub>4</sub> Flexible processes are linked to technology acceptance
	H <sub>5</sub> Flexible processes are linked to web satisfaction
	H <sub>6</sub> Flexible processes are linked to online trust
Iterative releases	H <sub>7</sub> Iterative releases are linked to technology acceptance
	H <sub>8</sub> Iterative releases are linked to web satisfaction
	H <sub>9</sub> Iterative releases are linked to online trust
Self organizing teams	H <sub>10</sub> Self organizing teams are linked to technology acceptance
	H <sub>11</sub> Self organizing teams are linked to web satisfaction
	H <sub>12</sub> Self organizing teams are linked to online trust
Technology acceptance	H <sub>13</sub> Technology acceptance is linked to firm performance
Web satisfaction	H <sub>14</sub> Web satisfaction is linked to firm performance
Online trust	H <sub>15</sub> Online trust is linked to firm performance
Firm size	H <sub>16</sub> Firm size is linked to firm performance

*H<sub>1,2,3</sub>: Early customer involvement is linked to technology acceptance, web satisfaction, and online trust.* Kling (1977) classified early customer involvement as end users who: a) participate as members of a design team, perform systematic analysis of benefits and costs anticipated from a proposed computer application, review designs for new application, provide test data for an application, sign off on accepting an application, and provide informal feedback on problems with the data processing unit. More importantly, the results of such early user involvement included: a) systems are valued that increase personal competence and pride in work, b) people are accepted as non rational and error prone, c) jobs are designed to be personally satisfying, d) automated procedures are designed to fit job needs, e) the burden of precision is placed on the machine, f) systems are forgiving, g) users can fully collaborate in system designs, and designs and assumptions are intelligible to users through clear documentation. Pine (1989) indicated that IBM used early customer involvement as a central strategy in the development of a new seven million line of code midrange computer operating system with 30 billion lines of application software, achieving unusually high levels of customer satisfaction. And, of course, MacCormack (1998) found a positive correlation between early customer involvement and improved product features, technical performance, and reliability in a study of 15 Internet firms. Numerous other studies have demonstrated the link between early customer involvement and customer satisfaction (Burnett, Cook, & Rothermel, 2004; Gietzmann & Selby, 1994; Huang & Mak, 2000; Leonard, 2004, Sugar, 2001). Since one of the principle purposes of using agile methods is to improve customer satisfaction (Highsmith, 2002, chap. 5), this study will test for a stronger link between the agile method principle of early customer involvement and customer satisfaction using scholarly models of technology acceptance, web satisfaction, and online trust (Davis, 1986; Pavlou & Gefen, 2004; Torkezadeh & Dhillon, 2002).

*H<sub>4,5,6</sub>: Flexible processes are linked to technology acceptance, web satisfaction, and online trust.* In economic terms, Thomke and Reinertsen (1998) defined development flexibility as “a function of the incremental economic cost of modifying a product as a response to changes that are external (e.g., a change in customer needs) or internal (e.g., discover a better technical solution) to the development process.” In technical terms it is the ability to: “a) pursue a more efficient development strategy that can tolerate a higher risk of design changes, b) make late product changes that lead to better design solutions with respect to customer needs and technologies, and c) avoid the need for product changes entirely because design commitments can be made very late.” Thomke and Reinertsen postulate development flexibility is an order of magnitude more cost effective than traditional methods and ideally suited for fast moving turbulent environments. Krishnan and Bhattacharya (2002) defined design flexibility as a parallel path (e.g., an approach in which the firm pursues multiple parallel paths to design the product with a specific technology choice in mind for a period) and sufficient design (e.g., a product design that is adequate for a product to function with multiple technologies). In their case, design flexibility results in 10% savings in product development costs, along with greater market penetration and customer satisfaction, again postulating its association with technological uncertainty. Highsmith (2002, chap. 3) defines flexible processes as: a) creating and responding to change, b) nimbleness and improvisation, c) conformance to business value, and d) balancing flexibility and structure. Since one of the principle purposes of using agile methods is to adapt to changing requirements (Beck, 1999), thereby improving customer satisfaction (Highsmith, chap. 5), this study will test for a stronger link between the agile method principle of flexible processes and customer satisfaction using scholarly models of technology acceptance, web satisfaction, and online trust (Davis, 1986; Pavlou & Gefen, 2004; Torkzadeh & Dhillon, 2002).

*H<sub>7,8,9</sub>: Iterative releases are linked to technology acceptance, web satisfaction, and online trust.* Basili and Turner (1975) defined one of the earliest antecedents of iterative releases for software development. Their process called iterative enhancement consisted of developing: a) a skeletal implementation of a software product, b) a project control list of remaining unimplemented features and tasks, c) coding and debugging the remaining features one at a time, d) analyzing the results, and e) updating the project control list. The primary goal of iterative enhancement was for computer programmers to learn as many lessons as possible during software development, thereby reducing costs and risks as well as improving the design, quality, and reliability of software products. However, iterative enhancement did not involve releasing intermediate versions to customers in order to solicit early customer feedback or enhance customer satisfaction. In another early antecedent, Madden and Rone (1984) of IBM described the use of 17 incremental releases to develop the software for NASA's space shuttle. All requirements were not known early in the process, the space shuttle was implemented in pieces, and individual software functions needed to be tested. IBM's approach resulted in fully functioning software for each incremental release, whereas iterative enhancements were purely for the benefit of increasing a computer programmer's knowledge and experience. Sulack, Lindner, and Dietz (1989), Cusumano and Selby (1995), Cusumano and Yoffie (1998), and MacCormack (1998) implicated a link between iterative releases and customer satisfaction for firms such as IBM, Microsoft, Netscape, and numerous Internet firms. Since one of the principle purposes of using agile methods is to improve customer satisfaction (Highsmith, 2002, chap. 5), this study will test for a stronger link between the agile method principle of iterative releases and customer satisfaction using scholarly models of technology acceptance, web satisfaction, and online trust (Davis, 1986; Pavlou & Gefen, 2004; Torkzadeh & Dhillon, 2002).

*H<sub>10,11,12</sub>: Self organizing teams are linked to technology acceptance, web satisfaction, and online trust.* Fagan (1976) was one of the first people to assert that computer programmers working in teams have higher productivity than those working as individuals. Fagan postulated that a group of reviewers could find more defects in computer programs than those working alone and thus satisfy its customers. In a study of 185 software development teams, Sonnentag, Frese, Stolte, Heinbokel, and Brodbeck (1994) found that team leader goal orientation was related to the quality of the software development process, final software product, and interaction within the software team. In a study of 117 software developers, Gobeli, Koenig, and Bechinger (1998) found that lower conflict intensity and improved conflict management styles were positively correlated to increased customer satisfaction, as well as software project success. Conventional team factors (e.g., presence of expertise, professional experience, administration coordination) and software development methods and expertise coordination processes (e.g., recognizing where expertise is needed, knowing where expertise is located, and bringing expertise to bear) are positively correlated to better team performance in a study of 69 software development teams (Faraj & Sproull, 2000). Numerous other studies also link teamwork to improved performance, software quality, and even customer satisfaction (Foote, Weiss, Matson, & Wenger, 2002; Hoegle & Gemuenden, 2001; Hoegl, Praveen-Parboteeah, & Gemuenden, 2003; Humphrey, 2000; Levesque, & Wilson, 2001; Sawyer, 2001; Sonnentag, 2000; Tiwana, & McLean, 2005). Since one of the principle purposes of using agile methods is to improve customer satisfaction (Highsmith, 2002, chap. 5), this study will test for a stronger link between the agile method principle of self organizing teams and customer satisfaction using scholarly models of technology acceptance, web satisfaction, and online trust (Davis, 1986; Pavlou & Gefen, 2004; Torkzadeh & Dhillon, 2002).

*H<sub>13,14,15</sub>: Technology acceptance, web satisfaction, and online trust are linked to firm performance.* In an adaptation of Fishbein's and Ajzen's (1975) theory of reasoned action, Davis (1986) determined that perceived usefulness, perceived ease of use, attitude toward using, and behavioral intention to use were significant determinants of actual system use, which also has implications for web satisfaction (Torkzadeh & Dhillon, 2002) and Online Trust (Pavlou & Gefen, 2004). In a study of 50 websites, Saeed, Hwang, and Grover (2002) found that value adding services, which help online shoppers locate, purchase, take delivery, or dispose of products are positively correlated to firm performance as measured by economic value added, especially when combined with advertising. Zhu and Kraemer (2002) studied 260 firms, and linked e-commerce capability (e.g., information, transaction, customization, and supplier connection) and information technology intensity to improved firm performance (e.g., lower cost, greater supply chain efficiency, and higher firm profitability). Supplier side digitization (e.g., supplier readiness, supplier process alignment, and supplier side online information capabilities), system integration, and customer side digitization (e.g., customer process alignment, customer side online information capabilities, and customer readiness) were significant determinants of firm performance (e.g., revenue per employee, gross profit margin, return on assets, and return on invested capital) in a study of 1,125 Internet firms (Barua, Konana, Whinston, & Yin, 2004). Technology acceptance, web satisfaction, and online trust may be a predictor of web value adding services, e-commerce capability, information technology intensity, and supplier and customer side digitization, which are themselves important indicators of firm performance. This study will test for a stronger link between technology acceptance, web satisfaction, and online trust and scholarly measures of firm performance (e.g., Tobin's Q; economic value added; or other productivity, profitability, and financial performance measures).

*H<sub>16</sub>: Firm size is linked to firm performance.* In a study of 135 firms, Kowtha and Choon (2001) found that firm size, strategic commitment, prior competencies, competitive intensity, firm age, and website age were related to a firm's commitment to e-commerce. In a study of 260 firms, Zhu and Kraemer (2002) linked firm size, e-commerce capability (e.g., information, transaction, customization, and supplier connection), and information technology intensity to improved firm performance (e.g., lower cost, greater supply chain efficiency, and higher firm profitability). Murphy, Poist, Lynagh, and Grazer (2003) studied 1,200 firms, which revealed little difference between firm size and the ability to generate sales support, trace customer orders, expedite customer service, generate sales leads, generate new customers, enhance company image, assist in supply chain management, develop relationship marketing, improve communications with customers, improve advertising, or build customer loyalty using Internet technologies. Teo and Pian (2004) conducted a study of 160 firms, which indicated that larger firm size was positively correlated to a progressive policy of adopting Internet technologies: a) e-mail adoption, b) web presence, c) prospecting, d) business integration, and e) business transformation. In a study of 114 firms, Zhu (2004) discovered that larger firms have larger investments in information technology, which were correlated to larger investments in e-commerce capabilities and better firm performance. Finally, Yang, Yang, and Wu (2005) studied 181 firms and found that firm size was positively correlated to significantly better e-business, operational, marketing, and adaptability performance. In some studies, firm size had no effect on firm performance (Alexander, 1949). Other studies have found that larger firms are significantly more profitable than small firms (Hall and Weiss, 1967). This study will test for a stronger link between firm size and scholarly measures of firm performance (e.g., Tobin's Q; economic value added; or other measures of productivity, profitability, and financial performance).

## RESEARCH METHODOLOGY

The research methodology is designed to examine the empirical links between the use of agile methods for developing Internet products, customer satisfaction, and firm performance. It consists of subsections for a research design, research measures, research analysis, research results, and research timeline (as shown in Table 8). The research design recommends a quantitative survey research design to test the links between agile methods, customer satisfaction, and firm performance. The research measures describe the variables, instrument, and items for implementing this study. The research analysis recommends the use of confirmatory factor analysis (CFA) for examining the internal reliability and convergent validity of the conceptual model, as well as the research factors, instrument, and items. The research results will simply be used to perform the hypothesis testing and construct the final path model (e.g., conceptual model showing which paths are valid or invalid). Finally, the research timeline recommends a one year study to develop the draft proposal, gain committee approval for the final proposal, conduct the six month study and prepare the draft dissertation, and then defend the dissertation. This may be one of the first empirical studies linking the four major factors of agile methods to scholarly models of customer satisfaction and firm performance. The results will be significant, whether or not the hypotheses indicate relationships between the factors.

Table 8

Research Methodology for Linking Agile Methods, Customer Satisfaction, and Firm Performance

Element	Explanation
1. Design	Quantitative survey research of 400 software managers
2. Measures	A nine construct survey instrument with 36 items (using a five point Likert scale)
3. Analysis	Confirmatory factor analysis (CFA) for internal reliability and convergent validity
4. Results	Testing of the 16 hypotheses and construction of the final structural path model
5. Timeline	12 month study to develop final proposal, collect data, and defend dissertation

### *Research Design*

A survey is recommended for examining the links between the use of agile methods for developing Internet products, customer satisfaction, and firm performance (as shown in Table 9). Quantitative survey research is recommended because of short timeframe available for this study, and because it will allow the collection of a large quantity of data from which to generalize from a sample to a population. Johnson (2002); Maurer and Martel (2002); and Succi, Pedrycz, Marchesi, and Williams (2002) have already demonstrated the feasibility of conducting large scale survey research in the domain of agile methods. Structural equation modeling (SEM) will be used to analyze the conceptual model and hypotheses to test the relationships between the factors of agile methods, customer satisfaction, and firm performance. SEM is a general, linear, cross sectional, and confirmatory statistical modeling technique used to determine whether a conceptual model has any semblance of validity, as proposed by this study (Kline, 2005). A sample size of 400 software engineering managers will be sought for this study, in order to administer a measurement instrument with 36 items. The first 28 items will consist of a five point Likert scale (e.g., strongly agree, agree, neutral, disagree, and strongly disagree). The sample may be drawn from one of the previous studies or a new one will be selected.

Table 9

Research Design for Linking Agile Methods, Customer Satisfaction, and Firm Performance

Element	Explanation
1. Method	Quantitative survey research method
2. Rationale	Speed and ability to generalize from a sample to a population
3. Statistics	Structural equation modeling (SEM)
4. Sample size	400 software engineering managers
5. Instrument	Survey questionnaire with nine major constructs and 36 items
6. Scales	Five point Likert scale (e.g., strongly agree to strongly disagree)

### *Research Measures*

The research measures describe the variables, instrument, and items for examining the use of agile methods for developing Internet products as shown in Table 10. The research instrument consists of nine major constructs: early customer involvement, flexible processes, iterative releases, self organizing teams, technology acceptance, web satisfaction, online trust, firm size, and firm performance. The research instrument is composed of nine constructs representing the nine major factors of agile methods and four items for each of them. The research instrument contains 36 items in total, which were derived from the most scholarly sources of empirical studies of agile methods for developing Internet products where possible. A five point Likert scale will be used to measure the 28 items comprising the first seven constructs (e.g., early customer involvement, flexible processes, iterative releases, self organizing teams, technology acceptance, web satisfaction, and online trust).

The scales will consist of a range of choices such as strongly agree, agree, neutral, disagree, and disagree. Firm or business unit revenue will contain a range of choices such as: less than \$1,000,000, \$1,000,000 to \$10,000,000, \$10,000,000 to \$100,000,000, \$100,000,000 to \$1,000,000,000, and greater than \$1,000,000,000. Annual Internet/intranet development budget will contain a range of choices such as: less than \$100,000, \$100,000 to \$1,000,000, \$1,000,000 to \$10,000,000, \$10,000,000 to \$50,000,000, and greater than \$50,000,000. Number Internet/intranet products and services will contain a range of choices such as: 1, 1 to 5, 5 to 10, 10 to 25, and greater than 25. Number of Internet product developers will consist of a range of choices such as: 1 to 10, 10 to 25, 25 to 50, 50 to 100, and greater than 100. Economic value added and Tobin's Q are described by Saeed, Hwang, and Grover (2005). Sharpe's and Treynor's performance indices are described by Davis, Dehning, and Stratopoulos (2003).

Table 10

## Research Measures for Linking Agile Methods, Customer Satisfaction, and Firm Performance

Construct	Variable	Instrument/Item	Source
Early customer involvement	Customer location	Customers reside with developers	Beck (1999)
	User interface designer	Customers design user interface	Gietzmann & Selby (1994)
	Beta releases/prototypes	Customers evaluate beta releases/prototypes	MacCormack (1998)
	Beta testing	Customers participate in beta testing	Cusumano & Selby (1995)
Flexible processes	Feature lists	Feature lists used as project plans	
	Metaphors and stories	Metaphors and stories used as requirements	Beck (1999)
	Informal processes	There are no written processes or documents	Highsmith (2002)
	Design flexibility	Products designed to withstand late changes	MacCormack (1998)
Iterative releases	Daily operational builds	Programmers create daily operational builds	Cusumano & Selby (1995)
	Frequent releases	Programmers make early, frequent releases	Beck (1999)
	Refactoring or redesign	Software is frequently refactored/redesigned	Cusumano & Selby (1995)
	Working release	Each release is a working product	Highsmith (2002)
Self organizing teams	Qualifications	Only best and brightest people are used	Cusumano & Selby (1995)
	Autonomous teams	Autonomous and empowered teams are used	Hackman (1976)
	Team cooperation	Cooperation vs. competition is encouraged	Alper, Tjosvold, & Law (1998)
	Pair programming	Software is created by two or more people	Nosek (1998)
Technology acceptance	Website perception	Website is perceived as useful	Davis (1986)
	Website usability	Website is easy to use	
	User attitude	User has a positive attitude toward website	
	User intention	User intends on using website	
Web satisfaction	Website features	Website has broad range of products/services	Torkzadeh & Dhillon (2002)
	Online shopping	Website alleviates need for shopping trip	
	Shopping convenience	Website makes shopping very convenient	
	Customer service	Website provides good customer service	
Online trust	Internet trust	User trusts Internet for conducting business	McKnight, Choudhury, & Kacmar (2002)
	Website trust	User trusts website for conducting business	
	Vendor/owner trust	User trusts vendor/owner of the website	
	Issue resolution trust	User trusts vendor/owner to resolve issues	
Firm size	Firm revenue	Firm or business unit revenue	Zhu (2004)
	Development budget	Internet/intranet development annual budget	
	Product/service number	Internet/intranet product and service number	
	Developer number	Number of Internet product developers	
Firm performance	EVA	Economic value added	Saeed, Hwang, & Grover (2005)
	TQ	Tobin's Q	
	SPI	Sharpe's performance index	Davis, Dehning, & Stratopoulos (2003)
	TPI	Treynor's performance index	

### Research Analysis

The research analysis will consist of confirmatory factor analysis (CFA) to verify the correlations between the nine major factors associated with using agile methods for developing Internet products. The factors of agile methods are early customer involvement, flexible processes, iterative releases, self organizing teams, technology acceptance, web satisfaction, online trust, firm size, and firm performance. Therefore, the use of CFA will help verify the relationships between the factors, variables, and items described by the conceptual model and operationalized by the research instrument. The use of CFA in this study will help “assess the role of measurement error in the model, validate a multifactorial model, and determine group effects on the factors” (Garson, 2005). In summary, CFA will help test the overall model fit.

Table 11

Research Analysis for Linking Agile Methods, Customer Satisfaction, and Firm Performance

Statistic	Definition
1. Cronbach’s alpha ( $\alpha$ )	A commonly used measure testing the extent to which multiple indicators for a latent variable belong together
2. Chi-square ( $\chi^2$ )	A fit test that should not be significant if there is a good model fit, while a significant chi-square indicates lack of satisfactory model fit
3. Goodness-of-fit index (GFI)	The percent of observed covariances implied by the model that deals with error in reproducing the variance-covariance matrix
4. Normed fit index (NFI)	A reflection of the proportion by which the researcher's model improves fit compared to the null model or random variables
5. Non normed fit index (NNFI)	A negative value indicates that the $\chi^2$ /degrees of freedom ratio for the null model is less than the ratio for the given model
6. Comparative fit index (CFI)	A comparison of the existing model fit with a null model which assumes the latent variables in the model are uncorrelated
7. Root mean square (RMS)	The coefficients which result from taking the square root of the mean of the squared residuals, indicating good model fit if close to zero

*Note.* Adapted from Garson (2005).

## Research Results

The information from the confirmatory factor analysis (CFA) will be used to construct a path model as shown in Table 12 and test each of the 16 hypotheses. The results of these tests cannot be known in advance. It is not the intention of the study to advocate the use of agile methods for developing Internet products. And, this study does not assume that early customer involvement, flexible processes, iterative releases, and self organizing teams are related to customer satisfaction and firm performance. Instead, it is the intention of this study to determine whether these relationships do in fact exist. These results will be relevant, important, significant, and interesting regardless of whether the outcome or whether the conceptual model are valid.

Table 12  
Research Results for Linking Agile Methods, Customer Satisfaction, and Firm Performance

Factors	Hypotheses	$\beta$	<i>t-value</i>	<i>p</i>
Early customer involvement	H <sub>1</sub> Early customer involvement → technology acceptance	+/-	+/-	<i>p</i> < 0.05
	H <sub>2</sub> Early customer involvement → web satisfaction	+/-	+/-	<i>p</i> < 0.05
	H <sub>3</sub> Early customer involvement → online trust	+/-	+/-	<i>p</i> < 0.05
Flexible processes	H <sub>4</sub> Flexible processes → technology acceptance	+/-	+/-	<i>p</i> < 0.05
	H <sub>5</sub> Flexible processes → web satisfaction	+/-	+/-	<i>p</i> < 0.05
	H <sub>6</sub> Flexible processes → online trust	+/-	+/-	<i>p</i> < 0.05
Iterative releases	H <sub>7</sub> Iterative releases → technology acceptance	+/-	+/-	<i>p</i> < 0.05
	H <sub>8</sub> Iterative releases → web satisfaction	+/-	+/-	<i>p</i> < 0.05
	H <sub>9</sub> Iterative releases → online trust	+/-	+/-	<i>p</i> < 0.05
Self organizing teams	H <sub>10</sub> Self organizing teams → technology acceptance	+/-	+/-	<i>p</i> < 0.05
	H <sub>11</sub> Self organizing teams → web satisfaction	+/-	+/-	<i>p</i> < 0.05
	H <sub>12</sub> Self organizing teams → online trust	+/-	+/-	<i>p</i> < 0.05
Technology acceptance	H <sub>13</sub> Technology acceptance → firm performance	+/-	+/-	<i>p</i> < 0.05
Web satisfaction	H <sub>14</sub> Web satisfaction → firm performance	+/-	+/-	<i>p</i> < 0.05
Online trust	H <sub>15</sub> Online trust → firm performance	+/-	+/-	<i>p</i> < 0.05
Firm size	H <sub>16</sub> Firm size → firm performance	+/-	+/-	<i>p</i> < 0.05

*Research Timeline*

The research timeline for examining the links between the use of agile methods for developing Internet products, customer satisfaction, and firm performance is shown in Figure 5. Four major groups of activities over a one year period are proposed: candidate prepares proposal (e.g., 2 months), committee reviews proposal (e.g., 2 months), candidate conducts research (e.g., 6 months), and candidate defends dissertation (e.g., 2 months). The first period (e.g., candidate prepares proposal) will be used to design a survey instrument and organize the draft proposal. The second period (e.g., committee reviews proposal) will be used to solicit committee approval and prepare the final proposal. The third period (e.g., candidate conducts research) will be used to administer the proposed survey to the sample of 400 managers and organize the draft dissertation. The fourth period (e.g., candidate defends dissertation) will be used to defend the dissertation and organize the final dissertation.

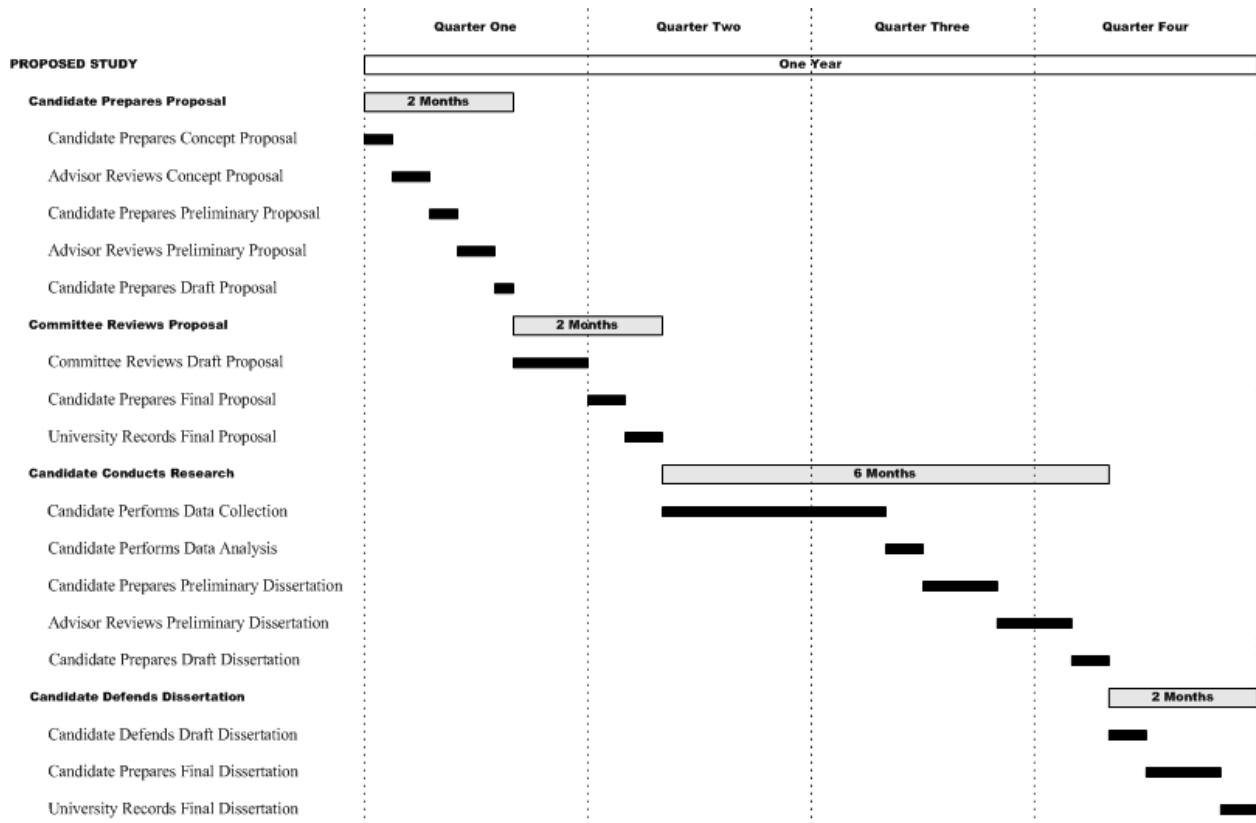


Figure 5. Research timeline to conduct study of agile methods for developing Internet products.

## CONCLUSION

The conclusion of the study of the use of agile methods for developing Internet products, customer satisfaction, and firm performance will consist of the subsections shown in Table 13. The findings will simply summarize the results of the study. That is, were there any empirical links between early customer involvement, flexible processes, iterative releases, self organizing teams, customer satisfaction, and firm performance? The contributions will be results of some of the first tests between the four major factors of agile methods and scholarly models of customer satisfaction and firm performance. The implications will either be a confirmation or disconfirmation of the theoretical tenets of using agile methods for developing Internet products among the 250,000 Internet projects in the U.S. With the myriad of industries, market sectors, firm types, firm sizes, and for-profit versus non-profit organizations, the limitations may consist of the generalizability or non-generalizability of the results of this study. One of the primary threats to validity will be the robustness, reliability, and validity of the design of the research instrument, variables, items, and scales. And, of course there may be many factors related to the success or failure of agile methods that were not within the scope or purview of this study.

Table 13

Conclusion for Linking Agile Methods, Customer Satisfaction, and Firm Performance

Element	Explanation
1. Findings	Links between use of agile methods, customer satisfaction, and firm performance
2. Contributions	One of the first comprehensive empirical studies of the use of agile methods
3. Implications	Empirical confirmation or disconfirmation of validity of using agile methods
4. Limitations	May not be generalizable to all industry, organization, product, and service types
5. Threats to validity	Reliability and validity of research instrument, sample size, and response rate
6. Future research	Effects of other factors such as virtual teams, tools, groupware, culture, etc.
7. Recommendations	Whether or not to use or study agile methods for developing Internet products

## REFERENCES

- A guide to the project management body of knowledge*. (2004). Newtown Square, PA: Project Management Institute.
- Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. (2002). *Agile software development methods: Review and analysis* (478). Oulu, Finland: VTT Publications.
- Agile Manifesto. (2001). *Principles behind the agile manifesto*. Retrieved July 23, 2005, from <http://www.agilemanifesto.org/principles.html>
- Ahituv, N., Hadass, M., & Neumann, S. (1984). A flexible approach to information system development. *MIS Quarterly*, 8(2), 69-78.
- Alexander, S. S. (1949). The effect of size of manufacturing corporations on the distribution of the rate of return. *Review of Economics and Statistics*, 31(3), 229-35.
- Alper, S., Tjosvold, D., & Law, K. S. (1998). Interdependence and controversy in group decision making: Antecedents to effective self managing teams. *Organizational Behavior & Human Decision Processes*, 74(1), 33-52.
- Aoyama, M. (1998). Web based agile software development. *IEEE Software*, 15(6), 56-65.
- Athey, S., & Schmutzler, A. (1995). Product and process flexibility in an innovative environment. *RAND Journal of Economics*, 26(4), 557-574.
- Banker, R. D., Kauffman, R. J., & Morey, R. C. (1990). Measuring gains in operational efficiency from information technology: A study of positran deployment of hardees inc. *Journal of Management Information Systems*, 7(2), 29-54.
- Barua, A., Konana, P., Whinston, A. B., & Yin, F. (2004). An empirical investigation of net enabled business value. *MIS Quarterly*, 28(4), 585-620.
- Basili, V. R., & Turner, J. (1975). Iterative enhancement: A practical technique for software development. *IEEE Transactions on Software Engineering*, 1(4), 390-396.
- Beck, K. (1999). Embracing change with extreme programming. *IEEE Computer*, 32(10), 70-77.
- Bender, D. (1986). Financial impact of information processing. *Journal of Management Information Systems*, 3(2), 232-238.
- Benediktsson, O., & Dalcher, D. (2003). Effort estimation in incremental software development. *Journal of Engineering Manufacture*, 150(6), 351-357.
- Benediktsson, O., & Dalcher, D. (2004). New insights into effort estimation for incremental software development projects. *Project Management Journal*, 35(2), 5-12.

- Brabb, G. J., & Hutchins, E. B. (1963). Electronic computers and management organization. *California Management Review*, 6(1), 33-42.
- Burnett, M., Cook, C., & Rothermel, G. (2004). End user software engineering. *Communications of the ACM*, 47(9), 53-58.
- Cao, Y., Gruca, T. S., & Klemz, B. R. (2003). Internet pricing, price satisfaction, and customer satisfaction. *International Journal of Electronic Commerce*, 8(2), 31-50.
- Carey, J. W. (1998). The Internet and the end of the national communication system: Uncertain predictions of an uncertain future. *Mass Communication Quarterly*, 75(1), 28-34.
- Chen, J. Q., & Heath, R. D. (2001). Building web applications: Challenges, architectures, and methods. *Information Systems Management*, 18(1), 68-79.
- Childers, T. L., Carr, C. L., Peck, J., & Carson, S. (2001). Hedonic and utilitarian motivations for online retail shopping behavior. *Journal of Retailing*, 77(4), 511-535.
- Cho, N. & Park, S. (2001). Development of electronic commerce user-consumer satisfaction index (ECUSI) for Internet shopping. *Industrial Management & Data Systems*, 101(8/9), 400-405.
- Christensen, J. F., Schmidt, M. L., & Larsen, M. R. (2003). Turbulence and competitive dynamics in the Internet services industry. *Industry and Innovation*, 10(2), 117-143.
- Churchill, G. A., & Surprenant, C. (1982). An investigation into the determinants of customer satisfaction. *Journal of Marketing Research*, 19(4), 491-504.
- Cron, W., & Sobol, M. (1983). The relationship between computerization and performance: A strategy for maximizing economic benefits of computerization. *Information and Management*, 6(3), 171-181.
- Cusumano, M. A., & Selby, R. W. (1995). *Microsoft secrets: How the world's most powerful software company creates technology, shapes markets, and manages people*. New York, NY: The Free Press.
- Cusumano, M. A., & Selby, R. W. (1997). How microsoft builds software. *Communications of the ACM*, 40(6), 53-61.
- Cusumano, M. A., & Yoffie, D. B. (1998). *Competing on internet time: Lessons from netscape and its battle with microsoft*. New York, NY: The Free Press.
- Cusumano, M. A., & Yoffie, D. B. (1999a). Building a company on Internet time: Lessons from netscape. *California Management Review*, 41(3), 21-28.
- Cusumano, M. A., & Yoffie, D. B. (1999b). Judo strategy: The competitive dynamics of Internet time. *Harvard Business Review*, 77(1), 70-81.

- Cusumano, M. A., & Yoffie, D. B. (1999c). Software development on Internet time. *IEEE Computer*, 32(1), 60-69.
- Davis, F. D. (1986). *A technology acceptance model for empirically testing new end user information systems: Theory and results*. Unpublished doctoral dissertation, Massachusetts Institute of Technology, Boston, MA, United States.
- Davis, F. D., Bagozzi, R. P., & Warshaw, P. R. (1989). User acceptance of computer technology: A comparison of two theoretical models. *Management Science*, 35(8), 982-1003.
- Davis, L., Dehning, B., & Stratopoulos, T. (2003). Does the market recognize IT-enabled competitive advantage? *Information & Management*, 40(7), 705-716.
- Deutsch, M. (1958). Trust and suspicion. *Journal of Conflict Resolution*, 2(4), 265-319.
- Diaz, M., & Sligo, J. (1997). How software process improvement helped motorola. *IEEE Software*, 14(5), 75-81.
- Dijkstra, E. W. (1968). The structure of THE: Multiprogramming system. *Communications of the ACM*, 11(5), 341-346.
- Easley, R. F., Devaraj, S., & Crant, J. M. (2003). Relating collaborative technology use to teamwork quality and performance: An empirical analysis. *Journal of Management Information Systems*, 19(4), 247-268.
- Erdogmus, H., & Favaro, J. (2003). Keep your options open: Extreme programming and the economics of flexibility. In M. Marchesi, G. Succi, J. D. Wells, & L. Williams (Eds.), *Extreme programming perspectives* (pp. 503-552). New York, NY: Addison Wesley.
- Erdogmus, H., & Williams, L. (2003). The economics of software development by pair programmers. *The Engineering Economist*, 48(4), 283-319.
- Fagan, M. E. (1976). Design and code inspections to reduce errors in program development. *IBM Systems Journal*, 12(7), 744-751.
- Faraj, S., & Sproull, L. (2000). Coordinating expertise in software development teams. *Management Science*, 46(12), 1554-1568.
- Favaro, J. (2003). Value based management and agile methods. *Proceedings of the Fourth International Conference on Extreme Programming and Agile Processes in Software Engineering (XP 2003)*, Genova, Italy, 16-25.
- Ferguson, P., Humphrey, W. S., Khajenoori, S., Macke, S., & Matvya, A. (1997). Results of applying the personal software process. *IEEE Computer*, 30(5), 24-31.

- Fichman, R. G., & Moses, S. A. (1999). An incremental process for software implementation. *Sloan Management Review*, 40(2), 39-52.
- Fishbein, M., & Ajzen, I. (1975). *Belief, attitude, intention, and behavior: An introduction to theory and research*. Reading, MA: Addison Wesley.
- Fishbein, M., & Ajzen, I. (1980). *Understanding attitudes and predicting social behavior*. Englewood Cliffs, NJ: Prentice Hall.
- Foote, N., Weiss, L., Matson, E., & Wenger, E. (2002). Leveraging group knowledge for high performance decision making. *Organizational Dynamics*, 31(3), 280-295.
- Foxall, G. R., Johnson, B., & Murphy, F. S. (1987). The development of control software for flexible manufacturing systems by british aerospace warton: A case history of user initiated product innovation. *Journal of Marketing Management*, 2(3), 259-274.
- Fulghum, D. A., & Wall, R. (2004). Is this war? *Aviation Week & Space Technology*, 160(6), 22-24.
- Gandal, N. (2001). The dynamics of competition in the Internet search engine market. *International Journal of Industrial Organization*, 19(7), 1103-1117.
- Garson, G. D. (2002). *Structural equation modeling*. Retrieved July 30, 2005, from <http://www2.chass.ncsu.edu/garson/pa765/structur.htm>
- Gefen, D. (2000). E-commerce: The role of familiarity and trust. *Omega*, 28(6), 725-737.
- Gietzmann, M. B., & Selby, M. J. P. (1994). Assessment of innovative software technology: Developing an end user initiated interface design strategy. *Technology Analysis & Strategic Management*, 6(4), 473-483.
- Gobeli, D. H., Koenig, H. F., & Bechinger, I. (1998). Managing conflict in software development team multilevel analysis. *Journal of Product Innovation Management*, 15(5), 423-435.
- Goth, G. (2000). The team software process: A quiet quality revolution. *IEEE Software*, 17(6), 125-127.
- Grady, R. B. (1997). *Successful software process improvement*. Upper Saddle River, NJ: Prentice Hall.
- Greene, S. L., Jones, L., Matcheri, P., & Thomas, J. C. (2003). Iterative development in the field. *IBM Systems Journal*, 42(4), 594-612.
- Greenstein, S. M., & Wade, J. B. (1998). The product life cycle in the commercial mainframe computer market, 1968-1982. *RAND Journal of Economics*, 29(4), 772-789.
- Hackman, J. R. (1976). *The design of self-managing work groups* (Technical Report No. 11). New Haven, CT: Yale University.
- Hall, M., & Weiss, L. (1967). Firm size and profitability. *Review of Economics & Statistics*, 49(3), 319-331.
- Herbst, P. G. (1962). *Autonomous group functioning*. London, England: Tavistock.

- Highsmith, J. (2002). *Agile software development ecosystems*. Boston, MA: Addison Wesley.
- Hoegl, M., & Gemuenden, H. G. (2001). Teamwork quality and the success of innovative projects: A theoretical concept and empirical evidence. *Organization Science: A Journal of the Institute of Management Sciences*, 12(4), 435-449.
- Hoegl, M., Praveen-Parboteeah, K., & Gemuenden, H. G. (2003). When teamwork really matters: Task innovativeness as a moderator of the teamwork performance relationship in software development projects. *Journal of Engineering & Technology Management*, 20(4), 281-302.
- Hoffnagle, G. F., & Beregi, W. E. (1985). Automating the software development process. *IBM Systems Journal*, 24(2), 102-120.
- Hoque, M., Akter, M., & Monden, Y. (2000). Effects of designers' participation and their evaluation measures on simultaneous achievement of quality and cost of product development teams. *International Journal of Innovation Management*, 4(1), 77-96.
- Huang, G. Q., & Mak, K. L. (2000). Modelling the customer-supplier interface over the world wide web to facilitate early supplier involvement in the new product development. *Journal of Engineering Manufacture*, 214(9), 759-769.
- Humphrey, W. S. (1996). Using a defined and measured personal software process. *IEEE Software*, 13(3), 77-88.
- Humphrey, W. S. (2000). *The team software process* (CMU/SEI-2000-TR-023). Pittsburg, PA: Software Engineering Institute (SEI).
- Institute of Electrical and Electronics Engineers. (1990). *IEEE standard glossary of software engineering terminology* (IEEE Std 610.12-1990). New York, NY: Author.
- International Organization for Standardization. (2000). *Quality management systems: Requirements* (ISO 9001:2000). Geneva, Switzerland: Author.
- International Organization for Standardization. (2003). *The ISO survey of ISO 9001:2000 and ISO 14001 certificates*. Geneva, Switzerland: Author.
- International Organization for Standardization/International Electrotechnical Commission. (1995). *Standard for information Technology: Software life cycle processes* (ISO/IEC 12207). Geneva, Switzerland: Author.
- Johnson, M. (2002). *Agile methodologies: Survey results*. Victoria, Australia: Shine Technologies.
- Jordan, M. H., Feild, H. S., & Armenakis, A. A. (2002). The relationship of group process variables and team performance: A team level analysis in a field setting. *Small Group Research*, 33(1), 121-150.
- Kan, S. H. (1991). Modeling and software development quality. *IBM Systems Journal*. 30(3), 351-362.

- Kan, S. H. (1995). *Metrics and models in software quality engineering*. Reading, MA: Addison Wesley.
- Kan, S. H., Dull, S. D., Amundson, D. N., Lindner, R. J., & Hedger, R. J. (1994). AS/400 software quality management. *IBM Systems Journal*, 33(1), 62-88.
- Kassim, N. M., & Zain, M. (2004). Assessing the measurement of organizational agility. *Journal of American Academy of Business*, 4(1/2), 174-177.
- Kauffman, R. J., & Wang, B. (2001). New buyer's arrival under dynamic pricing market microstructure: The case of the group buying discounts on the Internet. *Journal of Information Systems*, 18(2), 157-188.
- Kaulio, M. A. (1998). Customer, consumer, and user involvement in product development: A framework and a review of selected methods. *Total Quality Management*, 9(1), 141-149.
- Kee, H. W., & Knox, R. E. (1970). Conceptual and methodological considerations in the study of trust and suspicion. *Journal of Conflict Resolution*, 14(3), 357-366.
- Kekre, S., Krishnan, M. S., & Srinivasan, K. (1995). Drivers of customer satisfaction for software products: Implications for design. *Management Science*, 41(9), 1456-1470.
- King, A. M., & Sivaloganathan, S. (1998). Development of a methodology for using function analysis in flexible design strategies. *Journal of Engineering Manufacture*, 212(3), 215-230.
- Kline, R. B. (2005). *Principles and practice of structural equation modeling*. New York, NY: Guilford.
- Kling, R. (1977). The organizational context of user centered software designs. *MIS Quarterly*, 1(4), 41-52.
- Knuth, D. E. (1963). Computer-drawn flowcharts. *Communications of the ACM*, 6(9), 555-563.
- Koufaris, M. (2002). Applying the technology acceptance model and flow theory to online consumer behavior. *Information Systems Research*, 13(2), 205-223.
- Kowtha, N. R., & Choon, T. W. I. (2001). Determinants of website development: A study of electronic commerce in Singapore. *Information & Management*, 39(3), 227-242.
- Krishnan, V., & Bhattacharya, S. (2002). Technology selection and commitment in new product development: The role of uncertainty and design flexibility. *Management Science*, 48(3), 313-327.
- Lederer, A. L., Maupin, D. J., Sena, M. P., & Zhuang, Y. (2000). The technology acceptance model and the world wide web. *Decision Support Systems*, 29(3), 269-282.
- Lee, M. K. O., & Turban, E. (2001). A trust model for consumer Internet shopping. *International Journal of Electronic Commerce*, 6(1), 75-91.
- Lee, R. P., & Grewal, R. (2004). Strategic responses to new technologies and their impact on firm performance. *Journal of Marketing*, 68(4), 157-171.

- Leonard, A. (2004). Focusing on maturity levels for the effective management of end users involved in software project teams. *South African Journal of Business Management*, 35(1), 17-25.
- Levesque, L. L., & Wilson, J. M. (2001). Cognitive divergence and shared mental models in software development project teams. *Journal of Organizational Behavior*, 22(2), 135-144.
- Liang, T. P., Lin, C. Y., & Chen, D. N. (2004). Effects of electronic commerce models and industrial characteristics on firm performance. *Industrial Management & Data Systems*, 104(7), 538-545.
- Lindvall, M., Basili, V., Boehm, B., Costa, P, Dangle, K, Shull, F, Tesoriero, R., Williams, L., & Zelkowitz, M. (2002). Empirical findings in agile methods. *Proceedings of the Second XP Agile Universe Conference on Extreme Programming and Agile Methods (XPAU 2002), Chicago, IL*, 197-207.
- Lucas, H. C. (1975). Performance and the use of an information system. *Management Science*, 21(8), 908-919.
- MacCormack, A. (1998). *Managing adaptation: An empirical study of product development in rapidly changing environments*. Unpublished doctoral dissertation, Harvard University, Boston, MA, United States.
- MacCormack, A. (2001). Product development practices that work: How Internet companies build software. *MIT Sloan Management Review*, 42(2), 15-24.
- MacCormack, A., & Verganti, R. (2003). Managing the sources of uncertainty: Matching process and context in software development. *Journal of Product Innovation Management*, 20(3), 217-232.
- MacCormack, A., Kemerer, C. F., Cusumano, M. A., & Crandall, B. (2003). Trade-offs between productivity and quality in selecting software development practices. *IEEE Software*, 20(5), 78-85.
- MacCormack, A., Verganti, R., & Iansiti, M. (2001). Developing products on Internet time: The anatomy of a flexible development process. *Management Science*, 47(1), 133-150.
- Madden, W., & Rone, K. (1984). Design, development, integration: Space shuttle flight software system. *Communications of the ACM*, 27(9), 914-925.
- Manz, C. C., & Sims, H. P. (1987). Leading workers to lead themselves: The external leadership of self managing work teams. *Administrative Science Quarterly*, 32(1), 106-128.
- Masters of technology*. (2004, September 20). InformationWeek, 1006, 45-54.
- Mathieson, K. (1991). Predicting user intentions: Comparing the technology acceptance model with the theory of planned behavior. *Information Systems Research*, 2(3), 173-191.
- Maurer, F., & Martel, S. (2002). Extreme programming: Rapid development for web-based applications. *Internet Computing*, 6(1), 86-90.

- McDade, S. R., Oliva, T. A., & Pirsch, J. A. (2002). The organizational adoption of high technology products for use: Effects of size, preferences, and radicalness of impact. *Industrial Marketing Management*, 31(5), 441-456.
- McGuire, J. B., Scheeweis, T., & Branch, B. (1990). Perceptions of firm quality: A cause or result of firm performance. *Journal of Management*, 16(1), 167-180.
- McKinney, V., Yoon, K., & Zahedi, F. M. (2002). The measurement of web customer satisfaction: An expectation and disconfirmation approach. *Information Systems Research*, 13(3), 296-315.
- McKnight, D. H., Choudhury, V., & Kacmar, C. (2002). Developing and validating trust measures for e-commerce: An integrative typology. *Information Systems Research*, 13(3), 334-359.
- Mukhopadhyay, T., Kekre, S., & Kalathur, S. (1995). Business value of information technology: A study of electronic data interchange. *MIS Quarterly*, 19(2), 137-156.
- Muller, M. M., & Padberg, F. (2002). Extreme programming from an engineering economics viewpoint. *International Workshop on Economics-Driven Software Engineering Research (EDSER), Orlando, Florida, USA*.
- Muller, M. M., & Padberg, F. (2003). On the economic evaluation of XP projects. *Proceedings of the Ninth European Software Engineering Conference (ESEC 2003), Helsinki, Finland*, 168-177.
- Murphy, P. R., Poist, R. F., Lynagh, P. M., & Grazer, W. F. (2003). An analysis of select web site practices among supply chain participants. *Industrial Marketing Management*, 32(3), 243-250.
- Nosek, J. T. (1998). The case for collaborative programming. *Communications of the ACM*, 41(3), 105-108.
- Paulk, M. C., Curtis, B., Chrissis, M. B., & Weber, C. V. (1993). *Capability maturity model for software version 1.1* (CMU/SEI-93-TR-024). Pittsburgh, PA: Software Engineering Institute (SEI).
- Pavlou P. A., & Gefen, D. (2004). Building effective online marketplaces with institution based trust. *Information Systems Research*, 15(1), 37-59.
- Pearson, C. A. L. (1991). An assessment of extrinsic feedback on participation, role perceptions, motivation, and job satisfaction in a self managed system for monitoring group achievement. *Human Relations*, 44(5), 517-536.
- Pickle, H. B., & Rungeling, B. S. (1973). Empirical investigation of entrepreneurial goals and customer satisfaction. *Journal of Business*, 46(2), 268-273.
- Pine, B. J. (1989). Design, test, and validation of the application system/400 through early user involvement. *IBM System Journal*, 28(3), 376-385.

- PMI. (2004). *Newly certified project management professionals (PMPs): July 2002*. Retrieved July 23, 2005, from [http://www.pmi.org/prod/groups/public/documents/info/pdc\\_newpmpsJuly02.pdf](http://www.pmi.org/prod/groups/public/documents/info/pdc_newpmpsJuly02.pdf)
- Przasnyski, Z. H., & Tai, L. S. (1999). Stock market reaction to malcolm baldridge national quality award announcements: Does quality pay? *Total Quality Management*, 10(3), 391-400.
- Ratnasingam, P., & Pavlou, P. A. (2003). Technology trust in Internet based interorganizational electronic commerce. *Journal of Electronic Commerce in Organizations*, 1(1), 17-41.
- Ravinchandran, T., & Lertwongsatien, C. (2005). Effect of information systems resources and capabilities on firm performance: A resource based perspective. *Journal of Management Information Systems*, 21(4), 237-276.
- Royce, W. W. (1970). Managing the development of large software systems. *Proceedings of the Western Electronic Show and Convention (WESCON 1970), Los Angeles, California, USA*, 1-9.
- Rust, R. T., & Zaborik, A. J. (1993). Customer satisfaction, customer retention, and market share. *Journal of Retailing*, 69(2), 193-215.
- Saeed, K. A., Hwang, Y., & Grover, V. (2002). Investigating the impact of web site value and advertising on firm performance in electronic commerce. *International Journal of Electronic Commerce*, 7(2), 119-141.
- Saeed, K. A., Hwang, Y., & Grover, V. (2005). The relationship of e-commerce competence to customer value and firm performance: An empirical investigation. *Journal of Management Information Systems*, 22(1), 223-256.
- Saren, M. (1994). Reframing the process of new product development: From stages models to a blocks framework. *Journal of Marketing Management*, 10(7), 633-643.
- Sassenburg, H. (2002). The nonsense of ROI. *SPIder Koerier*, 2(4), 4-5.
- Sawyer, S. (2001). Effects of intra group conflict on packaged software development team performance. *Information Systems Journal*, 11(2), 155-178.
- Shih, H. P. (2004). An empirical study on predicting user acceptance of e-shopping on the web. *Information & Management*, 41(3), 351-368.
- Singh, N., & Sushil. (2004). Flexibility in product development for success in dynamic market environment. *Global Journal of Flexible Systems Management*, 5(1), 23-34.
- Sliwa, C. (2002). Users warm up to agile programming. *Computer World*, 36(12), 8-8.
- Software Engineering Institute. (2002). *CMMI for software engineering: Staged representation version 1.1* (CMU/SEI-2002-TR-029). Pittsburg, PA: Author.

- Software Engineering Institute. (2005). *Process maturity profile: Software CMM 2004 year end update*. Pittsburgh, PA: Author.
- Song, J. H., & Adams, C. R. (1993). Differentiation through customer involvement in production or delivery. *Journal of Consumer Marketing*, 10(2), 4-12.
- Sonnentag, S. (2000). Excellent performance: The role of communication and cooperation processes. *Applied Psychology: An International Review*, 49(3), 483-497.
- Sonnentag, S., Frese, M., Stolte, W., Heinbokel, T., & Brodbeck, F. C. (1994). Goal orientation of team leaders: Its effects on performance and group interaction in software development projects. *European Work & Organizational Psychologist*, 4(2), 153-168.
- Space and Naval Warfare Systems Command. (1985). *Military standard: Defense system software development* (DoD-STD-2167). Washington, DC: Author.
- Space and Naval Warfare Systems Command. (1994). *Military standard: Software development and documentation* (MIL-STD-498). Arlington, VA: Author.
- Srivastava, R. P. (1999). Evidential reasoning for webtrust assurance services. *Journal of Management Information Systems*, 16(3), 11-32.
- Standish Group. (2000). *Extreme chaos*. West Yarmouth, MA: Author.
- Succi, G., Pedrycz, W., Marchesi, M., & Williams, L. (2002). Preliminary analysis of the effects of pair programming on job satisfaction. *Proceedings of the Third International Conference on Extreme Programming and Agile Processes in Software Engineering (XP 2002)*, Alghero, Sardinia, Italy, 212-215.
- Sugar, W. A. (2001). What is so good about user centered Design: Documenting the effect of usability sessions on novice software designers. *Journal of Research on Computing in Education*, 33(3), 235-250.
- Sulack, R. A., Lindner, R. J., & Dietz, D. N. (1989). A new development rhythm for AS/400 software. *IBM Systems Journal*, 28(3), 386-406.
- Tam, K. Y. (1998). The impact of information technology investments on firm performance and evaluation: Evidence from newly industrialized economies. *Information Systems Research*, 9(1), 85-98.
- Teo, T. S. H., & Pian, Y. (2004). A model for web adoption. *Information & Management*, 41(4), 457-468.
- Thomke, S., & Reinertsen, D. (1998). Agile product development: Managing development flexibility in uncertain environments. *California Management Review*, 41(1), 8-30.
- Thoms, T., Moore, K. S., & Scott, K. S. (1996). The relationship between self efficacy for participating in self managed work groups and the big five personality dimensions. *Journal of Organizational Behavior*, 17(4), 349-362.

- TickIT. (2004). *International register of TickIT: Certificated organizations*. Retrieved July 23, 2005, from <http://www.tickit.org/cert-org.htm>
- Tiwana, A., & McLean, E. R. (2005). Expertise integration and creativity in information systems development. *Journal of Management Information Systems*, 22(1), 13-43.
- Torkzadeh, G., & Dhillon, G. (2002). Measuring factors that influence the success of Internet commerce. *Information Systems Research*, 13(2), 187-204.
- Turner, J. (1982). Organizational performance: Size and the use of data processing resources. *Proceedings of the Third International Conference on Information Systems, Ann Arbor, Michigan, USA*.
- Von Hippel, E. (1978). Successful industrial products from customer ideas. *Journal of Marketing*, 42(1), 39-49.
- Weill, P. (1992). The relationship between investment in information technology and firm performance: A study of the value manufacturing sector. *Information Systems Research*, 3(4), 307-333.
- Woodward, S. (1999). Evolutionary project management. *IEEE Computer*, 32(10), 49-57.
- Yang, S. M., Yang, M. H., & Wu, J. T. B. (2005). The impacts of establishing enterprise information portals on e-business performance. *Industrial Management & Data Systems*, 105(3), 349-368.
- Zand, D. E. (1972). Trust and managerial problem solving. *Administrative Science Quarterly*, 17(2), 229-239.
- Zarraga, C., & Bonache, J. (2005). The impact of team atmosphere on knowledge outcomes in self managed teams. *Organization Studies*, 26(5), 661-681.
- Zhu, K. (2004). The complementarity of information technology infrastructure and e-commerce capability: A resource based assessment of their business value. *Journal of Management Information Systems*, 21(1), 167-202.
- Zhu, K., & Kraemer, K. L. (2002). E-commerce metrics for net enhanced organizations: Assessing the value of e-commerce to firm performance in the manufacturing sector. *Information Systems Research*, 13(3), 275-295.
- Zucker, L. G. (1986). Production of trust: Institutional sources of economic structure: 1840-1920. *Research in Organizational Behavior*, 8(1), 53-111.