ANTECEDENTS AND CONSEQUENCES OF CERTIFICATION OF SOFTWARE ENGINEERING PROCESSES

By

GORDON KEITH FULLER

B.A.Sc., The University of Toronto, 1984
M. Eng., The University of Toronto, 1986

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

in

THE FACULTY OF GRADUATE STUDIES

(Individual Interdisciplinary Studies Graduate Program)

THE UNIVERSITY OF BRITISH COLUMBIA

January 2006

Abstract

Software development projects are frequently problematic and often fail. Certification of software engineering processes to quality standards such as the ISO 9000 or SEI CMM family of standards may be a viable means of dealing with these problems. Nevertheless, such certification is infrequently sought. This research consists of two investigations, one on the consequences and the other on the antecedents of certification of software engineering processes.

The first investigation is comprised of four event studies of the market reaction to announcements of certification of software engineering processes. The first event study shows that in America the market interprets the announcement of certification to ISO 9000 as portending increased future revenue flows for companies that focus on the production of products, but not for companies that produce services. In contrast, the second event study shows that the Japanese market anticipates a reduction in future revenue streams associated with the announcement of certification for companies that focus on the production of services but not for companies producing products. Respectively, the third and fourth event studies found no significant American market response to announcements of assessments using the Software Capability Maturity Model or the Capability Maturity Model Integrated standards.

The second investigation used a survey methodology. There was a significant relationship between the likelihood to certify and the competitiveness of the company's marketplace, the size of the company's typical project team, and the anticipated direct costs of certification. No evidence was found to support the hypotheses that risk of failure, size of company, culture of quality, or focus on product versus service were related to likelihood to certify.

The most interesting findings are interpreted as follows. The differences between the consequences of certification in Japanese and American markets possibly reflect a American focus on short-term cost reduction when products are produced versus a Japanese longer-term concern over quality issues when services are provided. Although American companies that produce products are likely to see a financial benefit, they are no more likely than those producing services to seek certification, suggesting that reasons other than financial benefit influence decisions to certify.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Abbreviation	Meaning	Page Introduced
4GL	A Fourth Generation Language is a tool set that allows a programmer to create software by describing what the software should do.	34
AAR	Average Abnormal Returns are the abnormal returns for a point in time relative to an event, that are averaged across a number of companies experiencing the event.	50
AMR	AMR corporation is the parent company of American Airlines and American Eagle.	2
AR	Abnormal Returns are the difference between the returns that are expected based on a market model, all other things being equal, and the actual returns observed.	50
CAAR	Cumulative Average Abnormal Returns are the average abnormal returns accumulated over a number of days in an event window.	50
CASE	Computer Assisted Software Engineering tools are a type of 4GL that allow a software developer to develop software.	35
CEO	The Chief Executive Officer is the most senior manager of a corporation, company, or large organization.	121
CFO	The Chief Financial Officer is the individual responsible for all financial matters in an organization.	121
CIO	The Chief Information Officer is the individual responsible for information technology in an organization.	121
CRM	Customer Relationship Management is packaged software that tracks all interactions of customers with a business.	6
DSDM	Dynamic Systems Development Method is a prototypic methodology used in the development of information systems.	35
EP	The Estimation Period is the period of time leading up to an event window in an event study. The behavior of the stock prices and the index prices in the estimation period is used to calculate a market model to predict the stock price, all other things being equal, in the event window.	51
ERP	Enterprise Resource Planning is packaged software that can support all aspects of a company's business processes.	2

EW	The Event Window in an event study is the days around the event that are examined to determine if the stock prices display a statistically significant cumulative average abnormal returns.	51
GAO	The Government Accountability Office is an agency of the U.S. Congress which provides oversight to federal government programs.	1
ISO	The International Organization for Standardization is a network of national standards bodies which is coordinated by a Central Secretariat based in Geneva, Switzerland (www.iso.org).	12
SEI	The Software Engineering Institute is an American body which is based at Carnegie Mellon University (www.sei.cmu.edu). It works to identify and create best practices in the field of software engineering. It is funded by the U.S. Department of Defense.	4
ISO 9000	The ISO 9000 series of standards deal with quality of products. Early versions focused on identifying defective outputs. The second generation of ISO 9000 standards shifted its focus from discovering defects, to improving processes to prevent defects from occurring. The current generation of ISO 9000 standards is geared towards supporting a broader spectrum of industry, whereas the earlier versions were more closely focused on manufacturing. (Refer to Chapter 2 for a full description of the standards).	7
ISO 15504	The ISO 15504 series of standards is a new offering and is specifically targeted at the software industry. It is similar in design to the SEI CMM standards. (Refer to Chapter 2 for a full description of the standards).	12
SEI CMM	The SEI created a series of standards that assessed the maturity of an organization, and was based on the premise that more mature organizations produce better quality outputs. CMM stands for Capability Maturity Model. Examples of these standards include the Software Capability Maturity Model (SW-CMM), and the Capability Maturity Model Integrated (CMMI) which replaced the SW-CMM. (Refer to Chapter 2 for a full description of the standards).	9
TQM	Total Quality Management is a corporate commitment that company policies and processes will be focused on meeting and exceeding customer expectations.	121

Acknowledgements

I would like to express my sincere appreciation to everyone who made my return to University to do this work possible.

- My research supervisor and committee chair, Ilan Vertinsky, has been instrumental in facilitating my education; he has always made time for me, provided me with good advice, and supported me in every way possible.
- My supervisory committee members Masao Nakamura, David Edgington, and Ted Dodds who all agreed to sit on my committee and have never wavered over the past five years.
- My wife Kathy Pichora-Fuller and son Robert Fuller who both supported my mid-life change from income earning professional to dependent student with all the stress and uncertainties that came with it.

Chapter 1 - Setting the Scene

Background

It has been claimed that the first industrial computerized information system was built in 1951 by the British firm J. Lyons (Bird, 2002; Economist, 1954). The LEO (Lyons Electronic Office) was a custom-built computer with operating software that was assembled to automate the business processes of the company, a large firm operating tea houses, bakeries, hotels, plantations, catering services, and many other food businesses in the United Kingdom. In the more than 50 years since the inaugural operation of LEO the use of computers has spread across most industries, and they have become a commonplace commodity in society.

As the size and cost of computers have shrunk, the complexity of the software that operates them has grown. The Microsoft Windows XP operating system is believed to have approximately 50 million lines of code (Mundie, 2002), the original Star Wars strategic defense initiative was planned to have between 40 and 100 million lines of code (GAO, 1990), and even cellular phones have 2 million lines of code and are projected to increase to having approximately 10 million lines of code in the near future (Hellestrand, 2005; Kim, 2005).

Considering this burgeoning complexity, it should not be surprising that Information Systems (IS) projects are often rife with problems. In this chapter, I explore the magnitude and frequency of IS project failures to establish the importance of my own research. I then introduce the role of certification of software engineering processes, and the nature of quality in software engineering and its relationship to certification. The scope of my research will be defined and prior research in this and closely related fields will be introduced. Finally, an overview of the present research program will be provided in conjunction with a description of the organization of the rest of this dissertation.

Software Project Failures

In this section, several examples of spectacular IS project failures will be described to provide a sense of the magnitude of the costs involved. Two broadly cited studies on the frequency of project failure will then be reviewed to establish the frequency of IS project failure.

There have been many examples of failed IS projects. One example of an IS project that failed and received extensive media attention is the 1992 "Confirm" reservation system. The system, jointly undertaken by AMR, Budget Rent A Car Corporation, Hilton Hotels Corporation, and Marriott International, was cancelled after 4 years and after 125 million dollars (U.S.) had been spent (Oz, 1994). After the collapse of the project, the various parties sued each other for hundreds of millions of dollars, and finally settled out of court. Other spectacular examples include the Denver Airport project, which finally produced a useable system, but was 86 million U.S. dollars over budget and almost three years late (Gibbs, 1994). When FoxMeyer Corporation failed to complete their conversion to a SAP Enterprise Resource Planning (ERP) system in 1996, it was cited as a major factor contributing to driving the company into bankruptcy (Scott, 1999). Another well known example of a failed IS project comes from Nike and its ERP project, which cost approximately 400 million U.S. dollars. This resulted in over 100 million U.S. dollars in lost sales when it failed, and ultimately it was blamed for a drop of approximately 2 billion U.S. dollars in market capitalization (Koch, 2004b). These examples, and others like them (ComputerWorld, 2000), highlight the magnitude of the risks that major IS projects entail. Smaller companies running smaller projects also face significant problems in successfully completing IS projects.

There are two highly cited large-scale studies which have examined the magnitude of the problem in the information systems sector within the United States. The Standish Group conducted one such study in 1994 with a follow-up study in 2001. The Standish Group estimated in 1994 that 250 billion U.S. dollars were spent on information technology application development in the United States (Standish Group, 1995). It was further estimated that approximately 81 billion U.S. dollars were spent on projects in 1995 that would be cancelled before completion due to project problems. This represented over 30% of all IS projects in the United States. Projects that were not cancelled were often delivered late and completed significantly over budget. Depending on how success was quantified, up to 84% of IS projects could be considered to have failed. By 2001, the Standish Group found that there had been minimal improvement in the performance of IS projects. Their survey of the industry found in that year that only 26% of projects were self-assessed by the sponsoring organization as being successes, and failed projects were estimated to have cost approximately 75 billion dollars (Standish Group, 2001).

The second large-scale study was conducted by the American National Institute of Standards and Technology (NIST) Acquisition and Assistance Division which released a report in 2002. This report quantified the cost attributable to a failure to adequately test for software errors in the United States to be in the range of 22 to 59 billion U.S. dollars (NIST, 2002). The costs reported in the NIST study include costs directly attributable to the software engineering processes as well as other costs including those accrued to the consumer. Nonetheless, it may be argued that improved software engineering processes would produce fewer software errors, reducing costs to both parties.

IS project failures are thus frequent and expensive. Projects that do not fail have a high incidence of software errors, which are also expensive. Research into ways to address these issues is therefore both needed and worthwhile.

Certification of Software Engineering Processes

This section explores the nature of software engineering, how software engineering can be broken down into various processes, and the role of certification in managing these processes.

Compared to most engineering disciplines, software engineering is a "relative newcomer," and there are many different definitions of software engineering. The Software Engineering Institute (SEI) at Carnegie Mellon University asserts that "Engineering is the systematic application of scientific knowledge in creating and building cost-effective solutions to practical problems in the service of mankind", and that "software engineering is that form of engineering that applies the principles of computer science and mathematics to achieving cost-effective solutions to software problems" (SEI, 2005d). Ian Sommerville, the author of a widely used text on software engineering (Sommerville, 2001), takes a broader view of software engineering and defines it as "an engineering discipline that is concerned with all aspects of software production" (Sommerville, 2004). A key engineering principle is that a problem should be broken into its component parts for analysis and resolution. When this principle is applied to problematic IS projects, it leads to the identification of key software engineering processes. Sommerville identifies these processes as being specification, design, development, verification, validation and management (Sommerville, 2001). These six processes form the basis of what is often referred to as a classical software development methodology. In other words, a software engineer needs to know the overall aim of the program about to be designed; the program then needs to be designed before it is built; after it is built it should be checked to verify that it satisfies the client's specifications; it then must be tested to verify that it operates correctly; and finally the entire project must be managed. For Sommerville, all software engineering projects are expected to have these six processes, although they may vary in form depending on the corporate organization and situation. For example, the first stage in every software project begins with a definition of what objectives the software is expected to carry out and how the software is expected to perform. In a traditional, or formal, methodology, the specification of functionality is a document which is completed before the software is written. In a prototypic methodology, the specification of functionality may comprise a prototype application, but as with a traditional methodology, the specification is completed before the bulk of the development work is undertaken. In either case, the software development team knows how the definition of requirements will be made, and the form that the specification will take. The steps taken to specify the required functionality provide the requirements definition process.

Regardless of the methodology chosen for a software project, all projects share some common characteristics. Software projects are, by definition, temporary and produce a unique product, service, or result (PMI, 2004). These two characteristics differentiate projects from ongoing operations, and this forms the crux of the difference between software engineering and manufacturing. Whereas manufacturing is usually oriented towards an ongoing process focused on producing a series of identical products, software engineering is oriented towards a series of one-time efforts focused on producing unique products.

Another characteristic of software projects is that they are intrinsically tied to the business processes they support. Projects that have a goal of producing software to support business processes that already exist, and are in use, are less complex than projects that have a goal of creating new business processes and then supporting the new processes. In the later case the scope of the software is much more likely to change as the underlying business processes evolve. Also, user resistance is increased if a project not only introduces new ways of performing known job functions, but also changes those job functions. Implementation of Enterprise Resource Planning (ERP) and Customer Relationship Management (CRM) software systems are examples of projects that almost always involve changes to both the business processes and the software systems. Not surprisingly these sorts of projects always figure prominently on lists of major software project failures (Charette, 2005; ComputerWorld, 2002).

Of course, there is no single "right" way to build software, and the software engineering processes may vary depending on the methodology used, but they should be consistent among projects in the same company following the same methodology. Lack of consistency leads to uncertainty in estimations of budgets and timelines, and reduces the likelihood of having a repeatable outcome.

It was established in the previous section that quality problems are pandemic in the field of software engineering (many projects fail and these failures incur substantial expenses). In most fields of engineering, the typical response to quality problems is the institution of a quality assurance program, frequently combined with independent third-party certification of the program. Sadly, this is not often the case in software engineering; there are several certifications of software engineering processes available but they are seldom adopted. This reluctance to adopt quality standards is often attributed to skepticism about the relevance of the standards to software engineering (Stelzer, Mellis, & Herzwurm, 1997). A recent article in the journal of the IEEE Computer Society found that software engineering was seriously deficient in terms of quality and rigour compared to circuit engineering and genetic engineering (Poore, 2004). Although many organizations have "adopted" formal methodologies, their actual use varies widely. It is common for IS projects not to follow established methods because the projects are all "different" and it is claimed by the project management that this wide difference between projects either renders a standard methodology "inappropriate", or that time constraints prevent compliance with an established methodology. Indeed, the successful adoption of formal methods in software development relies heavily on the development of a corporate culture that fosters their use. The use of a standard methodology needs to be accepted within any corporation as a useful means to produce good quality, rather than only a mandated layer of bureaucracy that must be "endured".

Certification of software engineering processes therefore is often a "dramatic event" that goes beyond the industry norm; it is not only a public statement that the company "believes in quality", but is also an expensive investment. It is difficult to quantify the actual cost of certification since it can vary so widely depending on the extent of the changes to business processes that are required, but Q/P Management Group Inc. has estimated that preparing for an ISO 9000 certification may cost a small to midsized company 25,000 to 50,000 U.S. dollars in fees paid to external bodies for training, auditing, consulting, and registration (Clifford & Martin, 2002). Such an investment in third-party certification also requires an ongoing commitment to quality merely to maintain the certification.

Given the high cost of certification, and the apparent resistance of the software development industry to certification, an interesting research question is whether certification of software engineering processes does in fact lead to more successful projects. Quantitative studies of project success are difficult to undertake, since it is difficult to obtain access to reliable data on project success. It is likely that certified processes do enhance the chances of success for some projects, but it is clear that in other cases the cost of the increased overhead introduced by the certified processes will exceed the benefit obtained (Rost, 2005). Thus, in order to come to grips with questions over the actual value of certification, it is useful to understand more about what sorts of characteristics are shared by companies that are likely to certify and also whether certification of software engineering processes is likely to lead to improved profitability.

Quality and Certification

In the previous section, it was established that software engineering projects have processes that are designed to maximize the likelihood of successful project completion. Certification of the software engineering processes to industry quality standards is a viable tool to establish corporate standards, policies, and procedures with respect to software engineering processes, and to ensure that software engineering processes are followed. In this section, the concept of "quality" as applied to software engineering processes is examined.

Certification of software engineering processes requires the inspection, or audit, of the processes by an independent third party who declares that the processes meet a defined standard. The standard is usually developed with the goal of assuring the uniformity or level of quality of the processes. The question of what is meant by the "quality" of software engineering processes is often ill defined, but is generally accepted by industry practitioners to mean that the processes are clearly specified or documented, that they are followed as specified, and that a record of the processes is kept. Some standards, such as ISO 9000, are fairly generic and deal more with how processes should be documented and managed. Other standards, such as the Software Engineering Institute's Capability Maturity Model (SEI CMM) family of standards, specifically address software engineering processes. Software engineering processes may be specified so as to maximize the likelihood that the software development project will be completed, that it will be roughly on budget and on schedule, and that the software will meet the functional requirements as evidenced by the successful adoption of the software by the customer or client for the intended purpose. How various standards address the performance of software engineering processes tends to vary according to the particular standard adopted, as is explored in Chapter 2.

While relatively little quantitative research has been carried out on quality in software engineering processes, considerable research has been carried out on quality in computer programming activities. Key studies have focused on what quality means and how it can be measured, including proposals of software metrics such as self-documentation, data commonality, error tolerance, and hardware independence, (Blundell, Hines, & Stach, 1997; Jones, 1978; Sherif, Ng, & Steinbacher, 1985). Other research focuses on techniques for quantitative assessment of the metrics once they are gathered (Khoshgoftaar & Seliya, 2004; Thwin & Quah, 2005). Another approach to evaluating software quality is to examine the users' qualitative perception rather than trying to identify and quantitatively measure software metrics (Stavrinoudis, Xenos, Peppas, & Christodoulakis, 2005). As noted earlier in this chapter, however, software engineering and its processes have a much greater scope than simply the end software product. Indeed, programming is only a small part of a software engineering project. A project may succeed in producing a very high quality piece of software, but if the developed software is finished and adopted significantly later than scheduled, or significantly over budget, or perhaps not adopted at all, then the project may be deemed to be a failure.

The Research Topic

If software engineering processes are so broad in scope, and if the quality standards are so generic in specificity, then how can a research question be identified that is both useful and achievable? Thus far I have provided evidence indicating that IS projects are problematic and often fail, that quality standards exist that can be applied to software engineering processes, yet quality standards are infrequently applied. The topic chosen for this dissertation specifically addresses the quantitative identification of the antecedents and consequences of certification of software engineering processes. In the study of antecedents of certification, an economic model for the motivation for certification is built, hypotheses are constructed, and tested. The study of consequences examines if certification is likely to be worthwhile. It is carried out using event studies of market returns. This research will be useful to software engineering practitioners within industry who are trying to decide if certification will be worthwhile, as well as to consulting organizations that would benefit from knowing the types of companies that are most likely to choose to certify their software engineering processes.

Scope Limitations of the Research

I have established the motivation for the research, built the foundation to support the discussion of software engineering and its processes, introduced the role of certification, and discussed how the existing quality standards view the concept of quality as applied to software engineering processes. The specific topic of antecedents and consequences of certification of software engineering processes has been identified. In this section, the topic of the present research is defined further by identifying common types of certifications, including certification of *individuals* and certification of *software products*, and clarifying that although these types of certification are related to software engineering, they are not directly relevant to the topic of certification of *software engineering processes*.

Certification of an individual's competence is commonly provided by vendors (e.g. Microsoft Certified Systems Engineer, Oracle Certified Database Professional) or associations (e.g., the IEEE Computer Society's Certified Software Development Professional, the Canadian Information Processing Society's Information Systems Professional). Having qualified personnel working on an IS project will likely improve the likelihood of project success, but there is no clear linkage between the qualification of individuals and certification of software engineering processes insofar as the credentials of individuals are not considered when processes are certified. In order to focus more clearly on the software engineering processes, certification of individuals and their skills is excluded from the scope of the present research.

Similarly, the topic of the present research does not include certification of the quality or capability of individual software products, such as the certification provided by the Free Software Foundations GPL Software Certification Program, which tests software for compliance with established technical standards and operational requirements. As previously noted, although the quality and characteristics of the product of software engineering processes are related to the processes followed in its production, the current thesis research focuses on the *process* rather than the *product*.

Certifications that are within the scope of this research include the ISO 9000 family of standards, the SEI CMM family of standards, and the ISO 15504 family of standards. (While the ISO 15504 family of standards would be an ideal choice to study because of their specificity to software engineering and their status as ISO standards, their relative newness means that there is an insufficient number of companies that are aware of or have adopted this particular standard. The ISO 15504 family of standards is described in Chapter 2, but is not included in the quantitative studies reported on in Chapters 3 or 4.)

Prior Research

Prior research into certification of software engineering processes includes case studies describing certification efforts and descriptions of the importance and processes of certification (Raman, 2000; Robinson & Simmons, 1996; Work, 2002). Survey studies focusing on the software industry have examined the frequency of ISO 9000 certification and its perceived utility (Griesemer, 1999; Kuilboer & Ashrafi, 2000), as well as the validity of common arguments against the use of ISO 9000 certification in the software field[1] (Stelzer et al., 1997). For example, in a study of draft ISO 15504 certification trials, the maturity of the business units engaged in software development was found to be greater for companies that had previously held ISO 9000 certifications and for companies that had larger IT staff groups (Jung & Hunter, 2001). Nevertheless, I have not found any quantitative research that focuses on why companies choose to certify their software engineering processes. Some work has investigated the reasons for certification in other fields such as forestry (Nakamura, Takahashi, & Vertinsky, 2000), manufacturing (Pan, 2003), or for certification of other processes such as environmental processes (Quazi, Khoo, Tan, & Wong, 2001). Similarly, I have not found quantitative research on how certification of software engineering processes affects companies' profitability, although similar research has been carried out in the field of manufacturing (Beirao & Cabral, 2002; Docking & Dowen, 1999; Martinez-Costa & Martinez-Lorente, 2003; Nicolau & Sellers, 2002; Pan, 2003; Terziovski, Power, & Sohal, 2003; Terziovski, Samson, & Dow, 1997).

An Overview of the Present Research

The remainder of this dissertation documents a line of research into the antecedents and consequences of certification of software engineering processes. The dissertation follows a manuscript-based format, with each of the two major studies having its own chapter. The literature review specific to each study, the hypotheses, the methodology, results, and conclusions of each study are found together in the respective chapters. The sequence of the studies reflects the order in which they were carried out

[1] Stelzer et al.'s work details many of the arguments that a practitioner often hears in practice. These include "ISO 9000 is inappropriate for mature/small/decentralized businesses", "The standard requires excessive documentation", and "The standard is too inflexible for an innovative business".

(event studies examining the consequences first, and survey on antecedents second). This sequence accommodates the use of findings from the event study in the formulation of the model of antecedents.

Chapter 2 examines quality certifications applicable to software engineering processes. It focuses on the history and details of the ISO 9000 family of standards, the Software Engineering Institute's Capability Maturity Model family of standards, and the emerging ISO 15504 family of standards.

Chapter 3 describes a series of event studies focusing on corporate announcements of certification of software engineering processes. Event studies are premised on the principle of an efficient marketplace, which claims that the marketplace will correctly evaluate the effect of events on future revenue streams. Thus, change in market price is taken to accurately reflect the impact of the certification. The study examines the North American market response to both ISO 9000 and SEI CMM certification announcements. . The North American market exhibited an unexpected response to announcements of ISO 9000 certification, leading to the replication of the study in the Japanese market. A literature review specific to event studies is included in this chapter.

Chapter 4 reports on a survey-based research project to investigate which sorts of companies are likely to certify. The research focuses on economic motivations for certification. The survey was administered to senior executives in Canadian companies engaged in software engineering. It focuses on testing a variety of hypotheses based on supposed economic benefits of certification of software engineering processes. A literature review supporting the development of these hypotheses is included in the chapter.

Chapter 5 summarizes the research, reports on an informal review of results with industry practitioners, and provides closing conclusions.

References

Beirao, G., & Cabral, J. A. S. (2002). The reaction of the Portuguese stock market to ISO 9000 certification. *Total Quality Management, 13*(4), 465-474.

Bird, P. (2002). *J. Lyons & Co. - LEO Computers*. Retrieved June 6, 2005, from http://www.kzwp.com/lyons/leo.htm

Blundell, J. K., Hines, M. L., & Stach, J. (1997). The measurement of software design quality. *Annals of Software Engineering, 4*, 235-255.

Charette, R. N. (2005). Why software fails. *IEEE Spectrum, 42*(9), 42-49.

Clifford, H., & Martin, J. (2002). *ISO Certification: The "Good Housekeeping Seal of Approval" for the 2000s*. Retrieved June 6, 2005, from http://www.qpmg.com/iso2.htm

ComputerWorld. (2000). *Top 10 corporate information technology failures*. Retrieved June 6, 2005, from http://www.computerworld.com/computerworld/records/images/pdf/44NfailChart.pdf

ComputerWorld. (2002). *The Best and the Worst*. Retrieved September 15, 2005, from http://www.computerworld.com/news/2002/story/0,11280,74620,00.html

Docking, D. S., & Dowen, R. J. (1999). Market interpretation of ISO 9000 registration. *Journal of Financial Research, 22*(2), 147-160.

Economist, T. (1954). *Electronic Abacus*. Retrieved June 6, 2005, from http://www.economist.com/science/displaystory.cfm?story_id=863244

GAO, U. S. (1990). *Report to the Chairman, Legislation and National Security Subcommittee, Committee on Government Operations, House of Representatives*

- *STRATEGIC DEFENSE SYSTEM.* Retrieved June 6, 2006, from http://www.fas.org/spp/starwars/gao/im90061.htm

Gibbs, W. W. (1994). Software's Chronic Crisis. *Scientific American, 271*(3), 86-95.

Griesemer, J. A. (1999). *A Field Study of the Impact of ISO 9001 on Software Development in the United States.* Unpublished Doctor of Professional Studies, Pace University.

Hellestrand, G. (2005). *Virtual system prototypes speed multiprocessor design.* Retrieved June 6, 2005, from http://www.us.design-reuse.com/articles/article10497.html

Jones, T. C. (1978). Measuring programming quality and productivity. *IBM Systems Journal, 17*(1), 39-63.

Jung, H.-W., & Hunter, R. (2001). The relationship between ISO/IEC 15504 process capability levels, ISO 9001 certification and organization size: An empirical study. *The Journal of Systems and Software, 59*, 43-55.

Khoshgoftaar, T. M., & Seliya, N. (2004). Comparative assessment of software quality classification techniques: An empirical case study. *Empirical Software Engineering, 9*(3), 229-257.

Kim, W. (2005). On Digital Convergence and Challenges. *Journal of Object Technology, 4*(4), 67-71:

Koch, C. (2004). *Nike rebounds: How (and why) Nike recovered from its supply chain disaster*. Retrieved 15, 2005, from http://www.cio.com/archive/061504/nike.html

Kuilboer, J. P., & Ashrafi, N. (2000). Software process and product improvement: An empirical assessment. *Information and Software Technology, 42*, 27-34.

Martinez-Costa, M., & Martinez-Lorente, A. R. (2003). Effects of ISO 9000 certification on firms' performance: a vision from the market. *TQM & Business Excellence, 14*(10), 1179-1191.

Mundie, C. (2002). *Trustworthy computing - Today and in the future*. Retrieved June 6, 2005, from http://www.microsoft.com/presspass/exec/craig/11-13svspeaker.mspx

Nakamura, M., Takahashi, T., & Vertinsky, I. (2000). Why Japanese firms choose to certify: A study of managerial responses to environmental issues. *Journal of Environmental Economics and Management*.

Nicolau, J. L., & Sellers, R. (2002). The stock market's reaction to quality certification: Empirical evidence from Spain. *European Journal of Operational Research, 142*(3), 632-641.

NIST. (2002). *The Economic Impacts of Inadequate Infrastructure for Software Testing*. Gaithersburg, MD: National Institute of Standards and Technology Acquisition and Assistance Division.

Oz, E. (1994). When professional standards are lax: The CONFIRM failure and its lessons. *Communications of the ACM, 37*(10), 29-43.

Pan, J.-N. (2003). A comparative study on motivation for and experience with ISO 9000 and ISO 14000 certification among Far Eastern countries. *Industrial Management and Data Systems, 103*(8-9), 564-578.

PMI. (2004). *A Guide to the Project Management Body of Knowledge* (Third ed.). Newtown Square, Pennsylvania: Project Management Institute Inc.

Poore, J. H. (2004). A tale of three disciplines... and a revolution. *Computer, 37*(1), 30-36.

Quazi, H. A., Khoo, Y.-K., Tan, C.-M., & Wong, P.-S. (2001). Motivation for ISO 14000 certification: Development of a predictive model. *Omega The International Journal of Management Science, 29*, 525-542.

Raman, S. (2000). It is software process, stupid: Next millennium software quality key. *IEEE AES Systems Magazine*(June).

Robinson, K., & Simmons, P. (1996). The value of a certified quality management system: The perception of internal developers. *Software Quality Journal, 5*(2), 61-73.

Rost, J. (2005). Software Engineering theory in practice. *IEEE Software, 22*(2), 96-95.

Scott, J. E. (1999). *The FoxMeyer Drugs' bankruptcy: Was it a failure of ERP?* Retrieved June 6, 2005, from http://www.ndsu.nodak.edu/ndsu/bklamm/BPandTC%20references/BP%20TC%20Scott%201999%20Foxmeyer%20drugs%20bankruptcy%20was%20it%20a%20failure.pdf

SEI. (2005). *What is software engineering?* Retrieved June 6, 2005, from http://www.sei.cmu.edu/about/overview/whatis.html

Sherif, Y. S., Ng, E., & Steinbacher, J. (1985). Computer software quality measurements and metrics. *Microelectronics and Reliability, 25*(6), 1105-1150.

Sommerville, I. (2001). *Software Engineering* (Sixth ed.). Harlow, England: Addison-Wesley.

Sommerville, I. (2004). *What is software engineering?* Retrieved June 6, 2005, from http://www.comp.lancs.ac.uk/computing/resources/IanS/SE7/Presentations/PDF/ch1.pdf

Standish Group. (1995). *Chaos*. Retrieved April 11, 2004, from http://www.standishgroup.com/sample_research/PDFpages/chaos1994.pdf

Standish Group. (2001). *Extreme Chaos*. Retrieved April 11, 2004, from http://www.standishgroup.com/sample_research/PDFpages/extreme_chaos.pdf

Stavrinoudis, D., Xenos, M., Peppas, P., & Christodoulakis, D. (2005). Early estimation of users' perception of software quality. *Software Quality Journal, 13*(2), 155-175.

Stelzer, K., Mellis, W., & Herzwurm, G. (1997). A critical look at ISO 9000 for software quality management. *Software Quality Journal, 6*(2), 65-79.

Terziovski, M., Power, D., & Sohal, A. S. (2003). The longitudinal effects of the ISO 9000 certification process on business performance. *European Journal of Operational Research, 146*(3), 580-595.

Terziovski, M., Samson, D., & Dow, D. (1997). The business value of quality management systems certification: Evidence from Australia and New Zealand. *Journal of Operations Management, 15*(1), 1-18.

Thwin, M. M. T., & Quah, T.-S. (2005). Application of neural networks for software quality prediction using object-oriented metrics. *The Journal of Systems and Software, 76*(2), 147-156.

Work, B. (2002). Patterns of software quality management in TickIT certified firms. *European Journal of Information Systems, 11*(1), 61-73.

Chapter 2 - Applicable Quality Standards

Introduction

Companies that choose to certify the quality of their software engineering processes have several options. Indeed, there is a plethora of process standards in software engineering (Sheard, 2001). Sheard's typology of these standards and their relationship to each other is shown in Figure 2-1.

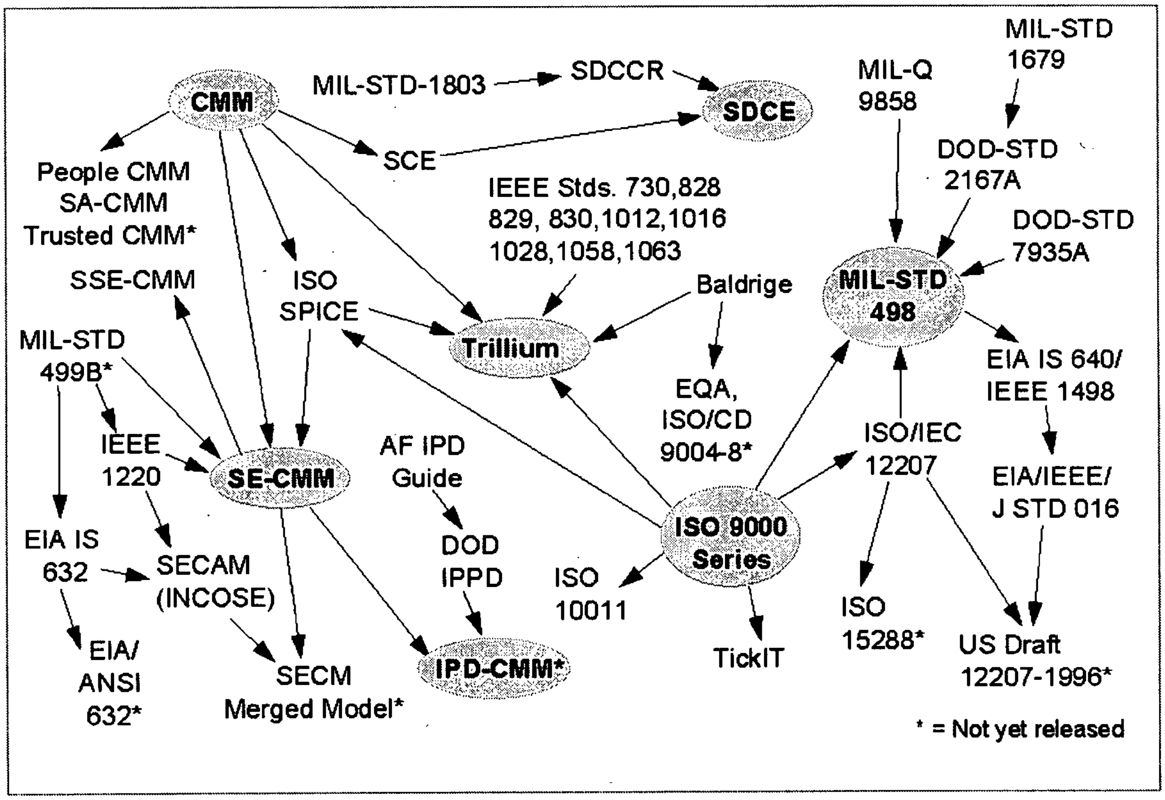

Figure 2-1: Software Engineering Process Certification Standards (Sheard, 2001)

Many of these are local variants of a smaller number of underlying core standards. For example, the ISO 9000 standards have been locally adapted in the United Kingdom as ISO 9001/TickIT, and by the American Society for Quality as Q9000. Three of the underlying core standards are of particular interest for the present work. The ISO 9000 standards and the Software Engineering Institute's (SEI) Capability Maturity Model (CMM) standards are the two most widely adopted software engineering standards in use today. The emerging ISO 15504 standard (ISO SPICE in Sheard's typology) proposes to bridge the differences between the ISO 9000 family of standards and the SEI CMM family of standards. Each of the two main existing standards and the emerging ISO standard will be examined in the following subsections.

ISO 9000

The International Standards Organization initially published the ISO 9000 series of standards in 1987. They were designed with large industrial manufacturing firms in mind. The publication of the ISO 9000 series of standards in 1987 was pivotal in establishing a nascent culture of quality in industry. The primary focus of the 1987 version of the standards was on maintaining quality based on testing and discovering problems after a product was built. The application of the ISO 9000 series of standards to software engineering did not broadly emerge until the next revision of the ISO 9000 standards occurred in 1994.

The ISO 9000 family of standards was revised in 1994 (ISO, 1994a, 1994b, 1994c, 1994d). A major shift in emphasis in the 1994 revisions involved establishing processes to ensure that quality was built into the product, in essence becoming proactive instead of merely reactive. This family of standards was primarily composed of four documents which dealt with the entire life cycle of products and services. The first, ISO 9001:1994, applied to companies that engaged in the full product manufacturing lifecycle, including design and development, production, installation and servicing. The second in the series, ISO 9002:1994, was used by companies that engaged in a more limited life cycle, and did not do their own design and development. ISO 9003:1994 was the most limited of the three and was applied to companies that did not use design control, process control, purchasing or servicing, and relied primarily on quality assurance inspection and testing to manage their quality program. Where the first three standards defined quality assurance models that may have been appropriate to various companies, the final document, ISO 9004:1994, gave guidance on how to implement a quality management system. All of these documents were written from the point of view of a manufacturing business. While these four documents remained stable over the next six years, an ongoing series of supporting documentation that dealt with how to apply the core standards were published in the period between 1994 and 2000.

In 2000, the 1994 standards were re-written and ISO 9001/9002/9003 were merged into ISO 9001:2000 (ISO, 2000b). This consolidation was largely the outcome of a regular review of the standards that found that the ISO9000 standards and supporting documents had grown to over 20 publications. The rationalization of ISO 9001/9002/9003 into ISO 9001:2000 was made possible by allowing companies to explicitly identify and disregard portions of the new standard that were inappropriate to their circumstances. At the same time, the ISO 9004 standard was revised and re-issued as ISO 9004:2000 (ISO, 2000a). (Figure 2-2 shows the various ISO 9000 versions on a timeline, along with the CMM standards and milestones in the evolution of IS methods.)

ISO 9004:2000 is geared to extend beyond ISO 9001:2000 to improve customer satisfaction. The 2000 series of documents were referred to as a "consistent pair" of standards because they were similarly structured and designed to facilitate extension from ISO 9001:2000 to ISO 9004:2000 (ISO, 2001). Companies that were certified to the

1994 series of ISO 9000 standards migrated to the 2000 series when their current certificate expired, or by December 15, 2003. (Certificates renewed to the 1994 version of the standards expired on December 15, 2003.) A major benefit of the 2000 revision was that the new standards were expected to apply to all organizations irrespective of size, sector, product or service. This was an improvement over the 1994 version of the standards, which were often difficult to apply beyond the major manufacturing industries.

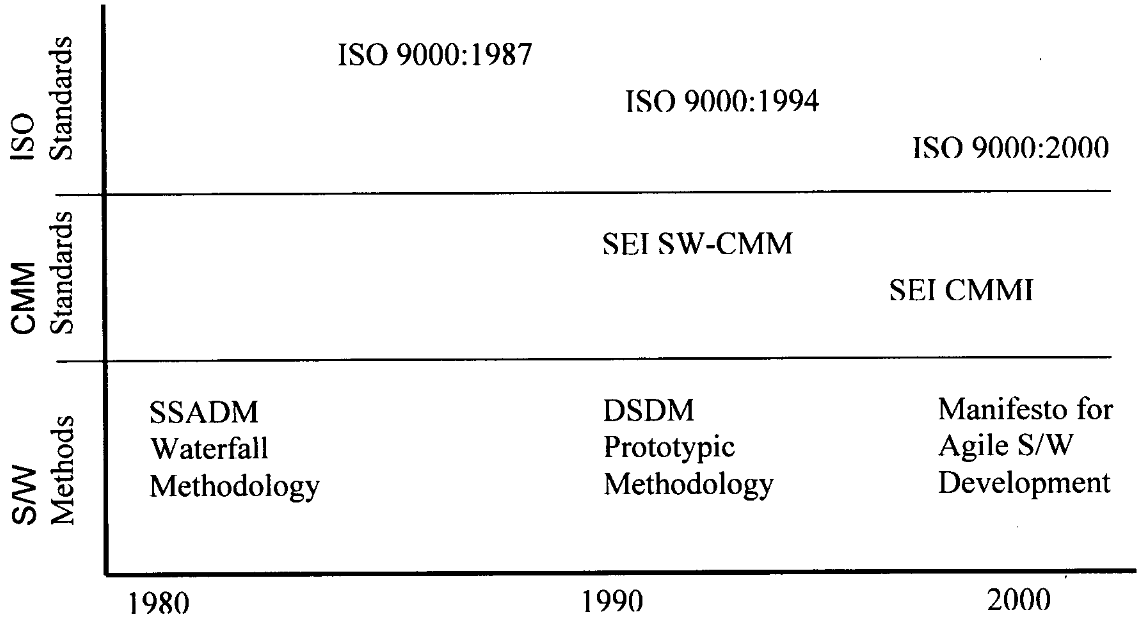

Figure 2-2: History of Standards and Methods

The key requirements identified in ISO 9001:2000 were split into five categories: systemic, management, resource, realization, and remedial. The systemic requirements to be demonstrated were that there was a quality management system, and that it was adequately documented. It was required that the corporate management supported quality management, that the company specifically addressed customer requirements, and that the quality management system was effectively planned, controlled, and run. In order to produce a quality product it was also required that the personnel, infrastructure, and working environment were developed and maintained with quality in mind. The realization of a quality management system starts with developing specifications, developing a product, then managing the purchasing, operational, and test functions to ensure that the product is built according to the specification. In order to take remedial action when the system fails it is necessary to monitor and measure quality to know when the failure occurs, to be able to take corrective action, and then to take measurements of the process so that it can be analyzed and improved.

Guidelines on how the ISO 9000 standards could be applied to software engineering were published in 1991 as ISO 9000-3:1991 (ISO, 1991). This led many software engineering organizations to undertake ISO 9000 certification. Organizations continued to obtain certification to the ISO standards, although software engineering was often only a small part of the overall scope of the processes that were certified. (For example a manufacturer of a telephone switch not only designs and manufactures the hardware, but it also designs and develops the software that runs on the hardware.) ISO has claimed that provision of services, including software engineering were better supported under the new version of the standard, but it is noteworthy that although the standard identified the generic sorts of things that should be done, it was completely free of detailed specification. For example, in ISO 9001:2000 section 7.2.1 specifies that the business must identify the customer's product requirements; the ISO standard does not specify how the requirements are to be identified nor how they are to be documented. This lack of specificity is not surprising given that the same standard is intended to apply to producers of all types of products and services. The interpretation of what was adequate to meet the standard was up to the company and the third-party auditor that it hired. The audit was essentially a "pass/fail" test. Companies were either certified as compliant, or they were not; there was no concept of "degree" of compliance.

Third parties, who have been themselves certified as registrars, may be contracted to perform an audit to certify an organization's compliance to the standards. The practice of providing consulting services to prepare for the certification audit, and the performance of the audit itself, can become a major line of business for some of the registrars. This raises the question of the "independence" of the third-party certification. Clearly there is a risk that if a third-party consultant has a profitable contract helping a client company prepare for the quality audit, then that third party has an interest in seeing that the client passes the audit and may not have the necessary "arms length" relationship to carry out the audit. If the independence of the auditors (and hence the validity of the certification) is questioned, then it becomes necessary to take a closer look at the benefits which are assumed to accrue from certification.

In 1994, the first 20 German software houses to attain ISO 9001 certification were surveyed to assess the degree to which the companies had improved their software engineering processes and the degree to which customers were able to derive benefit from the certification (Stelzer et al., 1997). Surprisingly, the study found that although the software engineering processes appeared not to have generally improved through the certification effort, there was a perception of increased quality by the firm's customers. It is possible that this is an example of the "Hawthorne Effect" (Pennock, 1930; Putman, 1930), whereby improvements in production are seen as a result of any effort to improve the firm's environment, regardless of its efficacy. While several articles have criticized the validity of the Hawthorne Effect, it continues to be widely acknowledged (Carey, 1967; Chant, 1993).

Moreover, some companies that certify take the issue of quality management to heart, while others may only do the bare minimum required to become certified. So, for some firms the goal is the "destination"; for others it is the "voyage" that is important. It is likely that improvement in processes will be maximized if the company actually embraces quality management, but there is no easy way to differentiate the two types of motivation; it follows then that evaluating the outcome of introducing ISO quality standards to in-house software engineering may be difficult.

Software Engineering Institute Capability Maturity Model

In the early 1990's, the Software Engineering Institute (SEI) at Carnegie Mellon established a standard specifically targeted at software engineering called the Software Capability Maturity Model (SW-CMM). The seminal work that led to this standard was published in 1989 (Humphrey, 1989). In this work, Humphrey described the maturity model and proposed an assessment process. The standard itself was published as two technical reports in 1993 (Paulk, Curtis, Chrissis, & Weber, 1993; Paulk, Weber, Garcia, Chrissis, & Bush, 1993). The standard identified five maturity levels that may be attained by organizations involved in software engineering. The SW-CMM was classed as a "staged" model, since the model was described as a series of maturity levels that are necessarily sequential, and the model assesses the entire organization at each level. These maturity levels correspond to software processes and are called *initial*, *repeatable*, *defined*, *managed*, and *optimizing*. It does not make sense to proceed to the *managed* level before the business reaches the *defined* level, and so on.

Each level of maturity raises practical issues for practitioners. The *initial* level is one that is characterized by ad hoc software development, with no formal repeatable processes in place. To be at the *repeatable* level, the organization depends on the folklore of the employees; they are able to repeat prior successes based on their experience, and the degree to which a project is well managed is a function of the people working on it. An organization that has reached the *defined* level has captured the experience of employees as a formal set of standards that define how software engineering is carried out. At this stage, there starts to be a more uniform quality across projects. To be at the *managed* level, the defined processes are measured and the measurements are collected to assess the efficiency of the processes. The choice of what measures are appropriate, and how they should be collected, is a common stumbling block to successful attainment of the *managed* level of maturity, and the barrier that most often leads to dissent and the collapse of the improvement process. In the final stage, called *optimizing*, enough quantified insight to the software engineering processes has been acquired that the organization is able to make calculated changes to the process, and so is able to predict the consequences of changes.

After the SW-CMM was completed, the same approach was taken by the SEI to model the maturity levels of organizations engaged in activities other than writing software. Examples included the Systems Engineering Capability Maturity Model, the Software Acquisition Capability Maturity Model, the People Capability Maturity Model, the Systems Security Engineering Capability Maturity Model, the Integrated Product Development Capability Maturity Model, and others. Different Capability Maturity Models, including the Software CMM, the Software Engineering CMM, and the

Integrated Product and Process Development CMM have been subsequently integrated into the Capability Maturity Model Integration (CMMI) Product Suite (CMMI-SE/SW/IPPD) (Bate & Shrum, 1998). In 2004 it was anticipated that more of the CMM models would be folded into the CMMI Product Suite in the future.

In the CMMI models, there are six levels of capability dimension. These levels are *incomplete*, *performed*, *managed*, *defined*, *quantitatively managed*, and *optimizing*. The business processes - in our case software engineering processes - are divided into a number of process areas (24 in the CMMI-SE/SW/IPPD version 1.0), and the process areas are grouped into four categories which are *process management*, *project management*, *engineering*, and *support*. Each process area is evaluated on each of the capability dimensions. Within each process area the model identifies certain features as being *required*, *expected*, or *informative*.

At this point in time, the only required model components are called *goals*. An example of a goal specific to the *requirements management* process area is "Requirements are managed and inconsistencies with project plans and work products are identified"[2]. The required components, or goals, are understood more easily by examining the expected components of the model. Currently, the only expected component is a statement of a "*practice*". Although certified companies are required to achieve the goals set for a particular capability dimension in a practice area, they are expected to do so by adopting a *practice*. In actuality, they may meet the goal by adopting an alternative practice that can be shown to accomplish the same goal. Each required goal is associated with 2 to 7 expected practices. CMMI-SE/SW/IPPD has 186

[2] (Ahern, Clouse, & Turner, 2001) page 67, Table 4-1: Specific Goals for Four Process Areas.

specific practices, which correspond to 54 specific goals. (A specific goal or practice is one that is unique to a single practice area; generic goals and practices span more than one practice area. For example, the generic goal for all processes at capability level 2 is "The process is institutionalized as a managed process"[3].)

Because each process area may be improved somewhat independently of other process areas, the model may be considered to be a "continuous model", in which case the capability level is represented as a profile of all of the process areas rather than as a single number. Alternatively, the assessment may assign a single maturity level based on the highest level where all goals from all process areas, from that level and all lower levels have been met, in which case it is referred to as a "staged model". In the latter case, the organization is referred to as being "a level *n*" company. The choice of continuous or staged model is up to the company that has the certification or self-assessment done.

Several authors have compared the CMM to the ISO 9000 series of standards (Haase, 1996; van der Pijl, Swinkels, & Verrijdt, 1997). It is suggested that organizations that have become certified to ISO 9000 standards are roughly at the *defined* level of the SEI CMM standard.

The SEI trains and authorizes lead assessors who are qualified to conduct assessments. (Note that the SEI specifically does not refer to the assessment as a "certification", using the term "assessment" instead.) Many of these assessors work for consulting organizations and offer their services to companies wishing to obtain an objective third-party assessment. In keeping with the concept of an assessment, rather

[3] (Ahern et al., 2001) page 98, Table 6-1: Generic Goals.

than a certification, the assessment makes no representation as to the subsequent performance of an organization, and may be repeated as often, or as infrequently, as the organization wishes; there is no “best before date” for a CMM type of assessment. Organizations are also able to conduct self-assessments using the SEI models and tools.

For the purposes of this work I will refer to CMM types of assessments carried out by qualified third-party assessors as certifications.

ISO 15504

A new ISO standard, ISO 15504, has been issued which is a convergence of ISO 9000 and CMM standards. The first documents for this emerging standard were produced in June 1995 (Bicego & Kuvaja, 1996). Initial work on this standard was tested on a project termed SPICE (Software Process Improvement and Capability dEtermination) (Simon, 1996). As of 1996, it was proposed as an ISO standard, and published as a technical report. ISO 15504, however, took significantly longer than expected to pass through the steps required to become an official ISO standard. In the meantime it was used in its provisional form as the basis for a European process assessment methodology for large companies (called BOOTSTRAP), and for another methodology appropriate for small to medium enterprises (called TAPISTRY) (Pasi Kuvaja, 1999; Pai Kuvaja, Palo, & Bicego, 1999). The ISO 15504 Technical Report was finally revised and issued as a series of formal standard documents starting in 2004.

The standard consisted of five documents or parts. These were "Concepts and Vocabulary", "Performing and Assessment", "Guidance on Performing an Assessment", "Guidance on Use for Process Improvement and Process Capability Determination", and "An Assessment Model and Indicator Guidance". All five parts came under the general title "Information Technology - Process Assessment". (Note that the original overall title was "Software Engineering - Process Assessment", but it was changed in March 2002.) Of these parts, only the second, "Performing an Assessment", was considered to be normative, while the remainder were informative[4].

The ISO 15504 standard was quite similar to the CMMI standard. It had a two-dimensional model of processes and process capability. The process model was not specified by the standard, although an exemplar was provided for informational purposes in the fifth part of the proposed standard. The capability was characterized by nine process attributes, which were grouped into five capability levels.

ISO 9000 standards have been widely used in Europe and Japan (Azuma, 1996), while CMM standards have been widely adopted in North America and India. As the world's economies become more global in nature, with more businesses operating in many diverse countries, convergence of the ISO 9000 and the SEI's CMM standards should be beneficial. ISO 15504 has the potential to provide that convergence.

Comparison of Standards

In order to better understand the differences between the ISO 9000 and the SEI CMM standards, it is useful to look at the development of these standards in a historical context. (Refer to Figure 2-2 to see the standards and methods arranged chronologically.) Computers and the programming of information systems to use computing resources are a relatively recent development. Early computers ran programs which consisted of instructions that were very basic, and the sequence of instructions were hard-wired (Museum, 2004). Programmers in that age were also hardware technicians. Over time,

[4] "Normative" documents specify an auditable behavior and define the requirements that must be met for certification. "Informative" documents provide guidance and clarification but are not auditable.

technologies developed that allowed symbolic representation of the instructions that made up the programs and the degree of abstraction of those instructions rapidly increased. It became no longer necessary to explicitly tell the computer which registers (physical memory devices) to read, which basic logical operations to perform, and where to store the result. Programming languages developed constructs that allowed symbolic representation of variables, complex operations on the variables, and conditional operations such as looping, and conditional and unconditional branching in the list of instructions. Programming eventually progressed to the point where the computers became multi-purpose machines capable of running different programs without physical modification of the computing hardware. Unfortunately, the complexity of the software frequently outstripped the sophistication of programming techniques. It was not uncommon to have code that consisted of large monolithic code fragments that made frequent use of unconditional branching directions, which produced so called "spaghetti code" that was difficult to debug and maintain.

One development that helped establish the concept of quality in the field of software development was E. Dijkstra's 1968 letter to the editor of the journal "Communications of the ACM" (Dijkstra, 1968), which launched "structured programming", a revolution in programming methodology. (Some people would view Dijkstra's prior paper (Dijkstra, 1965) as being the true origin of the principles of structured programming, but his letter to the editor three years later received the public attention that led to the ideas being broadly adopted.) Structured programming focuses on program clarity; it eschews the use of program structures, such as the GOTO statement, which obscures code intelligibility; it encourages the practice of breaking code into smaller fragments, functions or subroutines, which allow the intent and purpose of the program code to emerge from the detail (Donaldson, 1973). Structured programming has laid out guidelines that were generally adopted to identify practices and goals that are consistent with quality assurance processes. Adoption of structured programming did not occur overnight. However, by the time that the first ISO 9000 standard came into existence in 1987 there was a clear trend towards standardization of coding practices, with early adoption of software quality assurance methods.

The ISO 9000 standards emphasized documentation and repeatability. Indeed, it is often anecdotally claimed that ISO 9000 certification does not ensure a good quality product, only a uniform quality of product. ISO 9000 certification works well with highly structured or "mature" software engineering methodologies such as Structured Systems Analysis and Design Methodology (SSADM), which is a classic "waterfall" methodology introduced in 1980 whereby progress through the software development life cycle is serial and gated (Ashworth, 1988). That is, specification must be complete before analysis begins, and analysis should be fully documented before programming, and so on. With the introduction of ISO 9000 in 1987, the question of how the standard could be applied to software engineering quickly became contentious, and in 1991 the ISO introduced guidelines for the application of ISO 9000 to software development (ISO, 1991).

A second development in the field of programming that was important in the field of certification of software engineering processes was the adoption of what have been called "fourth generation languages" (4GLs). These occurred from the early 1980's. The term 4GL is generally attributed to James Martin (Martin, 1981), and is a tool that essentially allows a problem to be described, and which then produces the software code that is resolved into machine instructions. A good example is Oracle Corporation's SQL*Forms tool. In early versions of SQL*Forms, introduced in the early 1980s, a programmer would answer a series of questions on the screen regarding where on the screen enterable fields would be placed. These questions also included specifications of data validation rules for the fields, which database elements the fields were associated with, and which database operations were allowed. The answers to these questions would be saved and an executable program could then be generated to allow a user to use the screen form to interact with the database. The same answers could be used to generate forms which could run on a wide variety of different computers and peripheral hardware.

4GLs are closely related to Computer Assisted Software Engineering (CASE) tools. CASE tools are also code generation products, but rather than specifying the desired functionality using specified syntax, the program is typically described using a drawing metaphor. CASE tools are frequently capable of not only producing software code, but also methodological outputs, such as design and analysis diagrams, reports, and data dictionaries. Both 4GLs and CASE tools have aided in the production of standardized products, facilitating improvements in quality, but in addition they are adaptable to methodologies where the program specification is developed as a functioning prototype rather than being fully defined before programming starts. This latter characteristic has promoted the adoption of prototypic methodologies, such as Rapid System Development (Gane, 1987) and Dynamic Systems Development Method (DSDM), which was introduced in 1994 as a reaction to the long timelines and heavy documentation requirements of SSADM (DSDM, 1995). The essence of a prototypic methodology is the replacement of a detailed document specifying what the software should look like and do, with a functional prototype, and the willingness to drop required functionality in order to maintain timelines. This has led to conflicts with the original philosophy of the ISO 9000 standards, which focused on attaining quality through checking whether or not the product was compliant with the specification, and taking corrective action if it was not. In the 1994 version of the ISO 9000 standard, the emphasis shifted to attaining quality through prevention of problems, as evidenced by documented conformance to procedures. This new orientation has reduced the dependency on a formal initial specification, but it still relies heavily on written documentation. The adoption of prototypic methodologies was not prevented with the 1994 revision to the ISO 9000 standards, but neither was it facilitated.

ISO 9000 originated in the manufacturing sector, whereas the CMM originated in software engineering practice. At about the same time that the 1994 revision to the ISO 9000 standard was being formulated, the CMM standard was introduced (Paulk, Curtis et al., 1993; Paulk, Weber et al., 1993). The focus of the CMM standard was the ability of the organization to have repeatable software engineering processes that facilitated continuous improvement of the quality of the software being produced. Key features of the CMM standard include the development of a methodology that works for the company, identification and measurement of metrics for that methodology, and then fine tuning the methodology to improve quality. CMM was similar to the ISO 9000 standard in that it imposed a fairly heavy overhead on the software developers, diverting time from coding into measuring, documenting, and following procedures.

Many people have viewed the ISO 9000 standard as being an "outward facing" standard in that adoption of the standard leads to certification and publication of the certification. Part and parcel of the certification was support of the effort at the highest levels of management. Often CMM standards have been considered to be "inward facing" in that the SEI emphasizes assessment rather than certification. The standard has been seen as a tool to enhance quality of software development processes, and often adoption of it has arisen out of the software development business unit. Both standards had their roots in governmental defense spending; the precursor to the ISO 9000 standard had its roots in British wartime munitions quality improvement efforts (Seddon, 2001), and the Software Engineering Institute at Carnegie Mellon University was funded by the US Department of Defense (SEI, 2005c).

By the middle of the 1990s, many software developers were rebelling against what was viewed as the oppressive overhead of both the ISO 9000 and the CMM standards. This culminated in 2001 in the Manifesto for Agile Software Development (Beck et al., 2001). Agile software development focused on incremental software development to meet customer needs rather than following a formal methodology, producing documentation, or even following a project plan. While there were many reports of success on relatively small projects, a recently published account of a major project cites certain problems such as reliance on all team members having expert systems development skills in order to maintain the quality of the software produced (Stephens & Rosenberg, 2003). It seems likely that smaller, less critical projects would benefit from a "lighter" methodology, and hence from the freedom gained by avoiding standards. Standards such as ISO 9000 and CMM have provided the structure to support a more mature quality assurance process as a project became larger, more complex, and riskier. However, that structure often came at a cost. Not only has certification been expensive, but compliance has also been expensive both in terms of financial and human resources. As ISO 15504 has started to become adopted there is hope that it might bridge the gap between the ISO 9000 standard's certification of process, and the CMM's certification of business capability.

References

Ahern, D. M., Clouse, A., & Turner, R. (2001). *CMMI Distilled - A Practical Introduction to Integrated Process Improvement* (First ed.). Boston: Addison-Wesley.

Ashworth, C. M. (1988). Structured systems analysis and design method (SSADM). *Information and Software Technology, 30*(3), 153-163.

Azuma, M. (1996). Software products evaluation system: Quality models, metrics and processes - International standards and Japanese practice. *Information and Software Technology, 38*(3), 145-154.

Bate, R., & Shrum, S. (1998). CMM Integration (CMMI) Framework. *SEI Interactive, 3*(1).

Beck, K., Beedle, M., Bennekum, A. v., Cockburn, A., Cunningham, W., Fowler, M., et al. (2001). *Manifesto for Agile Software Development.* Retrieved September 28, 2004, from http://agilemanifesto.org/

Bicego, A., & Kuvaja, P. (1996). Software process maturity and certification. *Journal of Systems Architecture, 42*, 611-620.

Carey, A. (1967). The Hawthorne studies: A radical criticism. *American Sociological Review, 32*(3), 403-416.

Chant, G. (1993). The Hawthorne effect. *Journal of the Market Research Society, 35*(3), 279.

Dijkstra, E. (1965). *Programming considered as a human activity.* Paper presented at the Proceedings of the 1965 IFIP Conference, New York City.

Dijkstra, E. (1968). Go To Statement Considered Harmful. *Communications of the ACM, 11*(3), 147-148.

Donaldson, J. R. (1973). Structured programming. *Datamation, 19*(12), 52-54.

DSDM. (1995). *DSDM Manual* (Second ed.). Surrey UK: Tesseract Publishing.

Gane, C. (1987). *Rapid System Development*. New York: Rapid System Development Inc.

Haase, V. H. (1996). Software process assessment concepts. *Journal of Systems Architecture, 42*(8), 621-631.

Humphrey, W. S. (1989). *Managing the software process*. Reading, Mass.: Addison-Wesley Publishing Company Inc.

ISO. (1991). *Quality management and quality assurance standards - Part 3: Guidelines for the application of ISO 9001 to the development, supply, and maintenance of software* (No. ISO 9000-3:1991). Geneve, Switzerland: International Standards Organization.

ISO. (1994a). *Quality management and quality system elements -- Part 4: Guidelines for quality improvement* (No. ISO 9004-4:1993/Cor1:1994). Geneve, Switzerland: International Standards Organization.

ISO. (1994b). *Quality systems -- Model for quality assurance in design, development, production, installation and servicing* (No. ISO 9001:1994). Geneve, Switzerland: International Standards Organization.

ISO. (1994c). *Quality systems -- Model for quality assurance in final inspection and test* (No. ISO 9003:1994). Geneve, Switzerland: International Standards Organization.

ISO. (1994d). *Quality systems -- Model for quality assurance in production, installation and servicing* (No. ISO 9002:1994). Geneve, Switzerland: International Standards Organization.

ISO. (2000a). *Quality management systems -- Guidelines for performance improvements* (No. ISO 9004:2000). Geneve, Switzerland: International Standards Organization.

ISO. (2000b). *Quality management systems -- Requirements* (No. ISO 9001:2000). Geneve, Switzerland: International Standards Organization.

ISO. (2001). *The year 2000 revisions of ISO 9001 and ISO 9004*. Retrieved April 9, 2002, from http://www.iso.ch/iso/en/iso9000-14000/iso9000/2000rev1.html?printable=true

Kuvaja, P. (1999). BOOTSTRAP 3.0 - A SPICE conformant software process assessment methodology. *Software Quality Journal, 8*, 7-19.

Kuvaja, P., Palo, J., & Bicego, A. (1999). TAPISTRY - A software process improvement approach tailored for small enterprises. *Software Quality Journal, 8*, 149-156.

Martin, J. (1981). *Application Development Without Programmers*: Prentice Hall.

Museum, C. H. (2004). *Computer Timeline*. Retrieved August 9, 2005, from http://www.computerhistory.org/timeline/timeline.php?timeline_category=cmptr

Paulk, M. C., Curtis, B., Chrissis, M. B., & Weber, C. V. (1993). *Capability maturity model for software, version 1.1* (No. CMU/SEI-93-TR-24): Software Engineering Institute.

Paulk, M. C., Weber, C. V., Garcia, S. M., Chrissis, M. B., & Bush, M. W. (1993). *Key practices of the capability maturity model, version 1.1* (No. CMU/SEI-93-TR-25): Software Engineering Institute.

Pennock, G. A. (1930). Industrial Research at Hawthorne. *Personnel Journal, 8*, 269-313.

Putman, M. L. (1930). Improving Employee Relations. *Personnel Journal, 8*, 314-325.

Seddon, J. (2001). *The Case Against ISO 9000*: Oak Tree Press.

SEI. (2005). *SEI - Sponsoring and Oversight Organizations*. Retrieved Sept. 15, 2005, from http://www.sei.cmu.edu/about/overview/sei/sponsor.html

Sheard, S. A. (2001). Evolution of the frameworks quagmire. *Computer, 34*(7), 96-98.

Simon, J.-M. (1996). SPICE: Overview for software process improvement. *Journal of Systems Architecture, 42*(8), 633-641.

Stelzer, K., Mellis, W., & Herzwurm, G. (1997). A critical look at ISO 9000 for software quality management. *Software Quality Journal, 6*(2), 65-79.

Stephens, M., & Rosenberg, D. (2003). *Extreme Programming Refactored: The Case Against XP*: Apress.

van der Pijl, G. J., Swinkels, G. J. P., & Verrijdt, J. G. (1997). ISO 9000 versus CMM: Standardization and certification of IS development. *Information and Management, 32*, 267-274.

Chapter 3 - Consequences of Certification[5]

Introduction

In the first chapter it was established that development of software is a risky business. Far too often the software development process fails. One way to reduce the risk of failure and the costs associated with software development may be to modify the software engineering processes of a company so that internal procedures are rationalized and strengthened, leading to an advanced process that then can be certified as meeting standards such as those specified in the ISO 9000 series. A number of studies have argued that the improvements to software engineering processes that are necessary to meet certification levels should reduce the costs associated with software development (Kuilboer & Ashrafi, 2000; Stelzer et al., 1997; Terziovski et al., 1997), and a great deal has been written on how to apply the ISO 9000 and CMM standards to software engineering (Schmauch, 1994; Stelzer et al., 1997; Yang, 2001). A consulting specialization has even developed to help companies to modify their existing procedures to attain certification to the ISO and CMM standards. There remains an open question, however, as to whether or not the time and money spent on certification yields benefits that exceed the costs of certification.

Benefits may take many forms. Certification almost always involves changes to corporate culture reflecting the importance of quality. Intangible benefits to a companies reputation are expected as the public's perception of the company changes. Of course the most obvious consequence of certification is the anticipated change in profitability. This chapter explores the impact of announcements of certification on corporate stock prices

[5] A version of the first event study in this chapter, American ISO certification, has been accepted for publication. Fuller, G. Keith and Vertinsky, Ilan (2006) Market response to ISO 9000 certification of software engineering processes, The International Journal on IT Standards and Standardization Research.

to determine if certification of software engineering processes has beneficial consequences to the company's profitability.

It is reasonable to expect that adoption of quality assurance processes in software development should reduce the rate of project failures. Having more reliable software engineering processes that reduce the rate of project failure should, in turn, reduce software development costs. For projects that would not actually fail without the certification, costs should still be reduced because fewer resources will be expended on correcting errors introduced during the software design, development, and implementation processes. Certification may also act as a means to assure potential customers that a company's software development processes will produce better quality products (Anderson, Daly, & Johnson, 1999), thereby leading to increased sales.

The above arguments support the supposition that certification should increase profits by either reducing costs or by increasing sales. If certification is expected to contribute to profits, obtaining it will increase the value of the certifying firm. Because certifications occur at different points in time, with different market conditions in effect, it is very difficult to directly assess the impact of certification on profitability. One study that tried to do so in the electronics industry found some evidence that the average certified company in that sector was larger and more profitable than the average non-certified industry (Simmons & White, 1999). In this chapter, event study methodology is used to test whether the expected net benefits from ISO 9000 certification of companies in the United States and in Japan that engage in software development are reflected as changes in their market valuation. In addition, North American companies that have chosen to adopt the Software Engineering Institute's Capability Maturity Model certification are also examined. The use of the event study methodology is not unique in the general field of certification and quality assurance, but this marks the first time that the methodology has been applied to certification of software engineering processes.

The goal of durable goods manufacturing processes is to produce a production run of products where every product in the run is exactly the same as every other product. If you are producing ball bearings then every ball bearing must be the same diameter, weight, and have the same durability as every other ball bearing being made (within design limits). Software, on the other hand, is required to support a specific business process. Each piece of software produced will be different from every other piece of software being produced, just as the business processes being supported are different. Software engineers are continuously innovating, bringing software, data, and business processes together in unique ways. Not only will each software product have different specifications, but there is no unique way to satisfy those specifications. No two software engineers are likely to write the same code to accomplish the same task. Thus, software engineering is different from most manufacturing since the processes are inherently creative. The ISO 9000 series of standards has its origins in manufacturing and this leads many software professionals to be wary of a certification which may constrain their creativity. The use of event study methodology allows us to look at the appropriateness of certification for software engineering through the eyes of the marketplace rather than through the eyes of software engineers.

Literature Review

In this section the literature pertaining to event study methodology, and its use in testing the value of announcements of quality certifications in industry is reviewed.

Event study methodology is well established, having been in use for over thirty years. In 1969, an early study sampled stock prices at monthly intervals in order to examine abnormal returns associated with stock splits (Fama, Fisher, Jensen, & Roll, 1969). By 1980, event studies had become well established, and simulation techniques (still based on a monthly sampling interval) were used to validate the methodology and to evaluate different techniques for carrying out the studies (Brown & Warner, 1980). Two important findings were that the use of a market model based on a least squares regression is a valid means of predicting "normal" stock prices, and that the methodology is relatively insensitive to sample size. This work was extended in 1985 to use daily instead of monthly returns. Although the use of daily returns violated normality assumptions, the methodology was shown to be robust with respect to these violations (Brown & Warner, 1985). The same study showed the methodology to be valid with sample sizes as small as five companies, although it was important that the stocks were regularly traded, with stocks from the New York Stock Exchange (NYSE) producing much better results than those from the American Stock Exchange (AMEX) because of the frequency of trading.

Event study methodology has been extensively used in the fields of finance and marketing. Use of the methodology to look at certification announcements is not unique. In a study in which the methodology has been applied to the more general case of ISO 9000 registration across all business sectors in the United States, it was found that while small firms reaped significant abnormal market returns from certification there was no evidence to support the hypothesis for larger companies (Docking & Dowen, 1999). Docking and Dowen noted that the results of their study were more statistically significant when the subject pool was limited to companies that announced their certification after the 1992 Maastricht Treaty which marked the emergence of the European Union (EU). (Once the EU was established, ISO 9000 certification was broadly adopted as a means of quality assurance to facilitate trade into and within the EU.) A subsequent study based on Portuguese companies across all business sectors that had attained ISO 9000 certification also found statistically significant market response to the announcement of certification, independent of the size of the company (Beirao & Cabral, 2002). Two recent event studies examined the consequences of ISO 9000 certification in Spanish firms (Martinez-Costa & Martinez-Lorente, 2003; Nicolau & Sellers, 2002). The first Spanish study found significant abnormal returns on the day of the announcement, whereas the other did not. All of these event studies spanned most of the industrial sectors in the country of interest.

Several event studies have also been carried out on the effect of prestigious awards for quality management on the market's expectations of future profits. These studies have yielded conflicting results. One study that looked at a variety of quality awards found that the market did show statistically significant abnormal returns around the time of the announcement of the awards (Hendricks & Singhal, 1996). This finding was challenged in a follow-up article which criticized the original work based on the sample size and on the effect of a small number of what could be considered outliers in the results (Adams, McQueen, & Seawright, 1999). A similar study that looked at American companies who had won the Malcolm Baldridge National Quality Award failed to find any significant abnormal market returns around the time of the announcement of the award (Przasnyski & Tai, 1999). However, two other similar studies of American firms that had won major quality awards did find significant abnormal returns (Ramasesh, 1998; Soteriou & Zenios, 2000). While these studies looked at companies that had won awards for quality management, rather than looking at companies that had attained certification of the quality of their processes, both the awards and the certifications have in common that they provide external validation of the quality of business processes.

Overall, prior research supports the supposition that certification of business processes can yield abnormal market returns, at least in some cases. It has been suggested that future work should focus on a single sector since the market response to certification may vary across industrial sectors (Soteriou & Zenios, 2000). To my knowledge, there is no research concerning the specific case of software engineering processes. The current study extends prior research by isolating the sector to focus on firms that are primarily engaged in software engineering. Studying the market response to ISO 9000 certification of software engineering is of particular interest because the certification is often resisted by practitioners in the field. The justification is that software engineering is different from general manufacturing because its processes are inherently more creative, and hence more difficult to describe as an invariant procedure. The event study methodology provides insight into the value assigned by the marketplace to certification in this sector. The high failure rate of projects, and the costs incurred, makes the market response a valuable objective measure to complement the practitioner's subjective point of view.

Methodology

This section describes the event study methodology, and reviews appropriate parametric and non-parametric tests.

The first step in the event study methodology was to identify suitable study companies that had announced that they had been certified to a common standard, and the date of that announcement for each company. The daily stock prices for each company being studied were collected for a period before and after the announcement. The period of time running from approximately 8 months before the announcement through approximately 6 days before the announcement was used as a baseline, and is referred to as the "estimation period". The period running from 5 days before the announcement through 5 days after the announcement is here referred to as the "evaluation period". Various subsets of the evaluation period will be examined to determine if the actual stock prices were significantly different from what would be expected if the announcement had no effect, and these will be referred to as "event windows".

For each of the studied companies, an index was constructed by selecting a group of similar companies. The stock prices for each of the certified companies and their corresponding index was collected for the estimation period and the evaluation period. An index price was calculated as the sum of the stock prices for the companies comprising the index for each day in the estimation period and the evaluation period. The performance of the index and the studied company during the estimation period was used to develop a market model based on a least squares regression, which was then used to predict what the studied company's daily stock price would have been during the evaluation period if the announcement had no effect on the market's evaluation of the company's worth, and hence its stock price.

The predicted returns (percent daily change in price) in the evaluation period were then compared to the actual returns. The daily difference between the two is measured for each of the studied companies. Since the null hypothesis is that the announcement of certification will have no effect on stock price, the daily difference between predicted and actual returns for each company is referred to as the "abnormal return". The abnormal returns (AR), are then averaged across all studied companies for each day in the evaluation period (average abnormal returns or AAR), and accumulated over the days in each event window being studied (cumulative average abnormal returns or CAAR). The significance of the CAAR for each event window is then tested.

The most common technique used to test the significance of the CAARs is a parametric test. This test is premised on the assumption that the predictive model, built using the daily stock prices of the index companies and the respective studied companies over the estimation period, will form an accurate prediction of what the stock prices would be in the event window if the announcement of the certification has no effect. The standard deviation of the abnormal returns in the event window is assumed to be the same as the standard deviation of the abnormal returns in the substantially longer estimation period.

Some studies have used an alternative non-parametric test to gauge the significance of the daily average abnormal returns (Corrado, 1989). This procedure combines the estimation period and the evaluation period, and ranks the daily abnormal returns from smallest to largest. It then proceeds to determine the likelihood that the returns in the event window would have their observed rank if the announcement of certification had no effect on stock price. Corrando's test has more power than the more conventional parametric test and is more suitable when the data fails to meet the underlying assumptions of normality of the distribution of abnormal returns, or when the stock is thinly traded (Campbell & Wesley, 1993; Cowan & Sergeant, 1996; Maynes & Rumsey, 1993). When a non-parametric test has been used to evaluate event study results for the effects of certification, it was Corrado's test that was most frequently chosen (Martinez-Costa & Martinez-Lorente, 2003; Nicolau & Sellers, 2002). A review of event study methods ranks this test as the most powerful non-parametric test for event studies (Armitage, 1995).

The present study uses both parametric and nonparametric tests to examine the significance of abnormal returns. The following two subsections describe the parametric and nonparametric tests in detail.

Parametric Test

The parametric test is described in detail below, and an example MathCad worksheet implementing the method is given in Appendix I.

First, two matrices, EP_Closing_Prices and EP_Index_Prices, are constructed which hold the actual closing prices and the market indices for the estimation period for each of the companies. (Throughout the analysis the prefix EP will be used to denote variables containing data from the estimation period leading up to the event, and EW will be used to denote variables containing data from the event window surrounding the event.) These matrices are constructed with the rows corresponding to the days in the estimation period, and the columns corresponding to the companies or the corresponding indices. A third matrix, OLS_Param, is constructed containing the parameters for the market models that can be used to predict the closing stock prices based on the indices. This matrix has rows that correspond to the companies, and two columns. The first column contains the market model parameter that indicates the y-axis intercept, and the second column contains the market model parameter that indicates the slope of the linear regression.

The market model parameters and the indices are used to construct a matrix, EP_Model, which contains the estimated stock prices in the estimation period,

$$\text{EP\_Model}_{\text{day},\text{company}} := \text{OLS\_Param}_{\text{company},1} + \left(\text{OLS\_Param}_{\text{company},2}\right)\cdot\left(\text{EP\_Index\_Prices}_{\text{day},\text{company}}\right)$$

The same process is followed to generate a predicted market model for the event window, EW_Model.

Next the "returns", or daily percent changes are calculated for the actual stock prices and the model prices in the estimation period and the event window. For example, the actual returns in the estimation period are calculated as

$$\text{EP\_Actual\_Return}_{\text{day}-1,\text{company}} := \frac{\left(\text{EP\_Closing\_Prices}_{\text{day},\text{company}} - \text{EP\_Closing\_Prices}_{\text{day}-1,\text{company}}\right)}{\text{EP\_Closing\_Prices}_{\text{day}-1,\text{company}}}$$

Similar calculations are used to calculate EP_Model_Return, EW_Actual_Return, and EW_Model_Return.

Once we start calculating the difference between the predicted and the actual returns in the event window, we will need to be able to evaluate the statistical significance of the difference. In order to do so we will need to have an estimate of the standard deviation of the differences. This estimate is based on the standard deviation of the differences in the estimation period, and this standard deviation is what is calculated next. First the difference between what the model predicts, and what the market exhibits is calculated and called the abnormal returns,

$$\mathrm{EP\_AR} := \mathrm{EP\_Actual\_Return} - \mathrm{EP\_Model\_Return}$$

These abnormal returns are then averaged across all of the companies being observed for each of the days in the estimation period,

$$\mathrm{EP\_AAR}_{\mathrm{day}} := \mathrm{mean}\left[\left(\mathrm{EP\_AR}^{\mathrm{T}}\right)^{\langle \mathrm{day} \rangle}\right]$$

(Note that in this notation the matrix of abnormal returns is transposed so that the days are in columns which can be projected out and averaged.) The average abnormal returns are then averaged across the days in the estimation period to obtain the mean average abnormal return,

$$\mathrm{EP\_MAAR} := \mathrm{mean}(\mathrm{EP\_AAR})$$

EP_MAAR is then used to calculate the standard deviation[6],

$$\mathrm{EP\_s\_AAR} := \sqrt{\frac{\sum(\mathrm{EP\_AAR} - \mathrm{EP\_MAAR})^2}{\mathrm{EP\_Return\_Days} - 1}}$$

Having completed the above steps, the analysis turns to the event window. Again, the abnormal returns are calculated,

$$\mathrm{EW\_AR} := \mathrm{EW\_Actual\_Return} - \mathrm{EW\_Model\_Return}$$

[6] Note that this is the normal way of calculating the standard deviation, and is the way that the seminal paper used in equation A.8 ((Brown & Warner, 1985), while a more recent paper does not have the denominator under the square root radix in equation 4 ((Docking & Dowen, 1999)). In personal correspondence with the authors of the latter paper, they have indicated that this difference is due to a typographic error introduced when their paper was typeset.

The abnormal returns are averaged,

$$EW\_AAR_{day} := mean\left[\left(EW\_AR^{T}\right)^{\langle day\rangle}\right]$$

and the average abnormal returns are summed to calculate the cumulative average abnormal return for the event window,

$$EW\_CAAR := \sum EW\_AAR$$

Finally, the significance of the cumulative average abnormal return is evaluated. The null hypothesis that we are working with is that there is no significant difference between what the stock prices are estimated to be in the event window, all other things being equal, and what they actually are. This implies that the abnormal returns in the event window have a mean value of 0, and a variance that is the same as the variance that we estimated for the estimation period. The t-test for significance is then given as follows.

$$t := \frac{EW\_CAAR}{EP\_s\_AAR \cdot \sqrt{EW\_Return\_Days}}$$

Note that although the test statistic follows a t distribution, the degrees of freedom is derived from the length of the estimation period, and is very large. Hence the distribution may be considered for all intents and purposes to be N(0,1).

Non-Parametric Test

Corrado's test is described in detail in this section, and an example MathCad worksheet showing the implementation of the test is shown in Appendix II.

Corrado's test starts with ranking each firm's abnormal returns across the combined estimation period and the event window. These ranks are assembled into a matrix Ri,j,, where the rows correspond to the day and the columns correspond to the company.

The test statistic replaces the daily excess returns used in the parametric test with the difference between each day's rank and the mean rank. The test statistic for the event day d is N(0,1) and is given as follows.

$$t := \left(\frac{1}{\text{Companies}}\right) \cdot \sum_{j=1}^{\text{Companies}} \frac{\left[R_{d,j} - \left(\frac{\text{Tot\_days}}{2}\right) - 0.5\right]}{S}$$

In the above formula the standard deviation S(K) is calculated using the combined estimation period and event window as follows.

$$S := \sqrt{\left(\frac{1}{\text{Tot\_days}}\right) \cdot \sum_{i=1}^{\text{Tot\_days}} \left[\frac{1}{\text{Companies}} \left[\sum_{j=1}^{\text{Companies}} \left[R_{i,j} - \left(\frac{\text{Tot\_days}}{2}\right) - 0.5\right]\right]\right]^2}$$

The following sections will describe how the event study methodology and the tests have been applied to data from the American and Japanese software industries.

Event Study 1 - American ISO Certification

This section describes an event study carried out on North American companies that announced certification to the ISO 9000 standard in the period 1990 to 1999, and which identified software development as being a specific activity included in the scope of the certification.

Data Collection and Analysis

A key decision in the research design is the choice of the event window. The event window is the period of time around the event during which stock prices are tested to determine if they are significantly different from what would be expected, all other things being equal. This study considers an event window which consists of the day of the announcement and the day after. This choice of event window assumes that the announcement is a "crisp" event, meaning that the certification has not come to the notice of the marketplace prior to the announcement, and that once the announcement is made the information is rapidly disseminated to the marketplace.

In 2001, North American companies that had attained ISO 9000 certification of their software design and development processes between 1990 and 1999 were identified using the database of certification registrations provided at the website http://www.qualitydigest.com/. The search of the database was limited to companies in the United States that had indicated that their business fell primarily into the Standard Industrial Classifications (SICs) 7371, 7372 and 7373. Respectively, these correspond to custom software development, packaged software development, and systems integration. The database search produced 225 records of certification. Elimination of duplicate records and records pertaining to companies that were not publicly traded on either the NYSE or the NASDAQ Exchange yielded 23 companies of interest. (Limiting the choice of companies to those traded on either the NYSE or the NASDAQ is a conventional means of ensuring that the stocks are regularly traded. The daily stock prices were manually reviewed to confirm that they were regularly traded.)

The historical daily closing stock prices for each of these companies were obtained from http://finance.yahoo.com/ for a period ranging from 199 days before the announcement of the certification through 5 days after the announcement.

In order to create an index for each of the certified companies, the Yahoo Finance classification of each company's Sector and Industry was determined, and a random sampling of 10 additional companies from the same classification was made. The historical stock prices for each index's component companies were obtained using the same procedure that was used for the certified company. As discussed above, the indices' component companies' daily closing prices were summed to calculate the daily index price. The historical stock prices of the certified companies and their indices were next split into an estimation period ranging from 199 days before the announcement through 6 days before the announcement, and an evaluation period ranging from 5 days before the announcement through 5 days after the announcement.

	Company	Ticker	alpha	beta	R
1	Abbott Laboratories	ABT	-3.519	0.139	0.981 ***
2	ADP	ADP	17.001	0.0467	0.259
3	Analog Devices Inc	ADI	10.537	0.03738	0.606 ***
4	Beckman Coulter	BEC	16.967	0.03754	0.094
5	CIBER Inc	CBR	-2.404	0.155	0.735 ***
6	Honeywell Inc	HON	18.11	-0.07737	0.270
7	IBM	IBM	15.868	-0.01831	0.272
8	L3	LLL	42.965	-0.0189	0.727 ***
9	Lockhead Martin	LMT	-0.0261	0.296	0.885 ***
10	MSC Software Corp	MNS	13.306	-0.01603	0.203
11	Nortel Networks	NT	-4.552	0.273	0.962 ***
12	Raytheon Training Inc	RTN	14.509	0.103	0.594 ***
13	Schlumberger	SLB	-0.921	0.22	0.964 ***
14	Unisys Corp	UIS	-10.876	0.175	0.822 ***
15	Varian Medical Systems	VAR	-13.092	0.197	0.771 ***
16	Adc Software	ADCT	-1.331	0.06685	0.712 ***
17	Direct Insite Corp	DIRI	-1.233	0.03851	0.853 ***
18	Evans and Sutherland	ESCC	-13.053	0.188	0.732 ***
19	Intergraph Corp	INGR	1.839	0.09086	0.778 ***
20	Mapics Inc	MAPX	26.181	-0.0424	0.060
21	Mentor Graphics	MENT	1.775	0.06934	0.237
22	Novell	NOVL	12.176	-0.02888	0.202
23	OAO Corp	OAOT	-3.4	0.0259	0.804 ***

*** Correlation above .5 threshold.

Table 3-1: ISO American Market Model Parameters

A market model for each of the certified companies was next formed by conducting a linear regression analysis of the daily index prices against the corresponding certified company's daily prices using SPSS 11 software. The results of this analysis are shown in Table 3-1.

The quality of the market models ranges from excellent to poor, as shown by the Pearson's Correlation Coefficients (Rs) ranging from 0.981 through 0.060. Since the purpose of developing the market models is to use them to predict the expected stock price of the certified companies in the absence of the certification, there is no justification to using companies whose stock prices cannot be effectively modeled. For this reason, companies for which the market model has an R of less than the arbitrary value of .5 were eliminated. This yields a set of 15 certified companies for which acceptable models were developed.

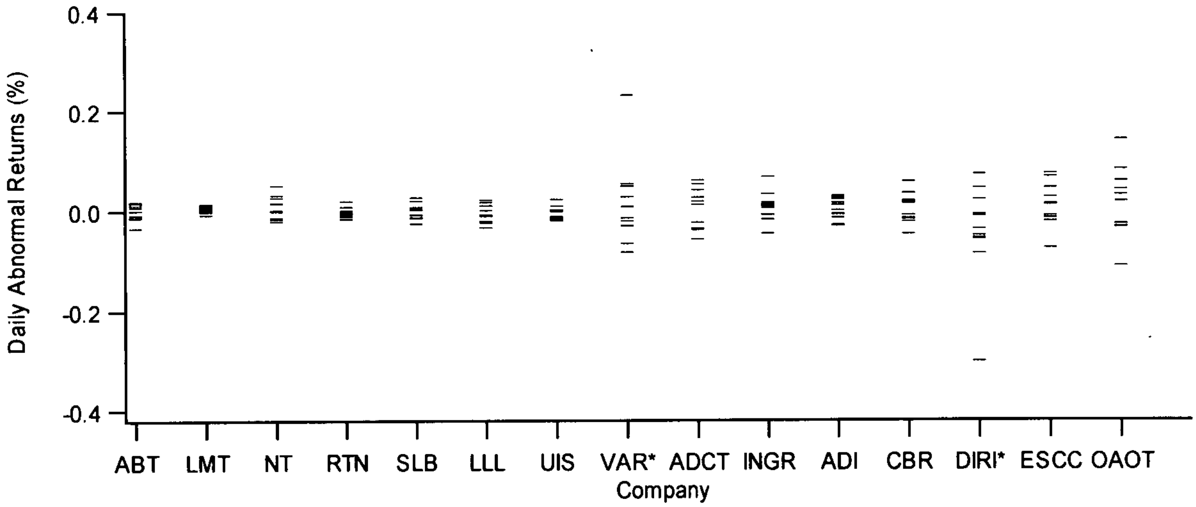

Figure 3-1: ISO American Companies Daily Abnormal Returns

Event studies, particularly with small sample sizes, are likely to be sensitive to the presence of outliers i.e., the inclusion of a company that has returns that are significantly outside the typical range can swing the results of the entire group. To check for outliers, the abnormal returns in the evaluation period were plotted as shown in Figure 3-1. Two companies were identified as outliers (marked with ‘*” in Figure 3-1). These companies were eliminated from the sample, reducing the number of studied companies to 13.

The selected 13 companies were rank ordered by their net revenue (as obtained from Edgar Online, http://www.edgar-online.com/), as shown in Table 3-2. The companies were then divided into two groups, 7 small companies with net revenue below 2 billion dollars, and 6 larger companies with net revenues above 2 billion dollars. (All prices are in U.S. dollars.)

Company	Year of Certification	Net Revenue ($US)
LMT	92	16,030,000,000
NT	96	11,919,000,000
RTN	94	10,098,000,000
ABT	91	6,876,588,000
SLB	93	6,705,000,000
UIS	94	2,877,000,000
ADCT	98	1,547,383,000
INGR	93	1,050,277,000
LLL	98	1,037,000,000
ADI	94	773,474,000
ESCC	92	148,594,000
CBR	95	120,151,000
OAOT	96	57,891,000

Table 3-2: ISO American Company Size

Besides looking at different sizes of companies, it was decided to also look at the companies broken into groups based on whether they were primarily engaged in the provision of services or products. The distinction between provision of services or products is based on their profile on the Yahoo Finance website, and by inspection of the companies' web sites. The 13 companies divided into two groups; 6 companies providing services, and 7 companies providing products. Refer to Table 3-3 for a listing of which companies fall into which group based on size and on whether they focus on provision of products or services.

Company	Small Companies	Producing Products
LMT		✓
NT		✓
RTN		✓
ABT		✓
SLB		
UIS		
ADCT	✓	
INGR	✓	
LLL	✓	✓
ADI	✓	✓
ESCC	✓	✓
CBR	✓	
OAOT	✓	

Table 3-3: ISO American Company Group Membership

The first analysis applied the parametric event study techniques, as described in the earlier section, to test the research hypothesis that announcement of certification of software engineering processes leads the marketplace to increase its estimation of future earnings. This analysis was carried out on all 13 of the companies taken together, on groups of larger and smaller companies, and on the companies grouped by whether they primarily engaged in the production of products or the provision of services. The event window that was tested was formed by the day of the announcement and the day after, [0,1]. The analysis failed to reject the null hypothesis that there is no difference in the abnormal returns before and during the event window for the composite group of all companies, or for either the groups of larger or smaller companies, or for the companies that are primarily engaged in provision of services. However, the companies engaged in provision of products showed a statistically significant positive market response for the event window [0,1] (z = 2.018; p < .05). That is to say, the parametric test was able to reject the null hypothesis for the companies that focus on production of products.

The specific z statistics for each of the parametric tests of the a priori hypothesis described above are given in Table 3-4. Having failed to reject the a priori null hypothesis for the companies taken as one group, or for the companies grouped by size, the study shifts to a search for any event window which exhibits a CARR which would have reached statistical significance if it had been proposed a priori. This phase of the study starts by repeating the parametric analysis for all possible event windows in the evaluation period, for the aggregate group of companies, for the small and large companies and for the companies that provide products and services (61 event windows x 5 groups of companies for a total of 305 tests).

Group	**Event Window [0,1]**
All Companies	1.085
Small Companies	1.027
Large Companies	0.387
Service Companies	0.147
Product Companies	2.018 *
* significant at .05 level of confidence	

Table 3-4: ISO American Companies Parametric z Statistics

As part of the post-hoc phase, Corrado's non-parametric test was also carried out on the data. The test yields a likelihood that the returns for a given day in the event window are sufficiently large to reject the null hypothesis that there is no effect of the announcement of certification. All days in the event window opening five days before the announcement and closing five days after the announcement, [-5,5] were tested for the combined data set, for the large and small company data sets, and also for the data sets comprising the companies that provide products and services (11 days x 5 groups of companies for a total of 55 tests).

The combined data set showed no significant cumulative average abnormal returns for any of the event windows, and also did not yield any significant single day abnormal returns from Corrado's test.

The data set composed of the smaller companies also showed no evidence of positive abnormal returns in any of the event windows from either the parametric or the non-parametric tests.

When the set of larger companies was examined, the parametric tests showed no significant findings for any of the event windows. However, Corrado's non-parametric test did produce a negative test statistic that exceeded the critical value for the .05 level of significance for the fifth day after the announcement (z = -2.164; the probability of rejection at this level is discussed in the discussion and conclusions section).

The companies that are primarily focused on provision of services (e.g. information technology, communications infrastructure, computer graphics, oilfield, and systems integration services) yielded no significant test results from either the parametric test, or from Corrado's non-parametric test.

Companies that focused on provision of products (e.g. integrated circuits, aircraft, networking, defence electronics, surveillance, and health care products) yielded parametric z-test statistics above the .05 level of significance for event windows [-2,1], [-2,2], [-2,3], and [-1,1], in addition to [0,1] which was noted in the initial results section, (z = 2.23, 2.01, 2.04, 2.12 respectively). Corrado's test did not show that the returns of any one of the days in the evaluation period were significant.

The z-statistic values obtained from Corrado's test for each day of the evaluation period, for each group of companies is shown in Table 3-5.

	Companies				
Day	**All**	**Small**	**Large**	**Services**	**Products**
-5	0.427	1.325	-0.728	0.280	0.310
-4	-0.740	-0.329	-0.682	-0.950	-0.095
-3	0.186	0.555	-0.295	1.498	-1.177
-2	0.451	-0.681	1.285	-0.492	1.079
-1	0.961	0.296	1.017	-0.014	1.314
0	0.359	0.900	-0.400	-0.424	0.890
1	1.193	1.219	0.420	0.062	1.555
2	0.639	-0.508	1.371	0.362	0.518
3	1.757	1.631	0.780	1.580	0.871
4	-1.125	-1.790	0.236	0.301	-1.810
5	-0.812	1.073	-2.164 *	0.267	-1.353
* P < .05					

Table 3-5: ISO American Company Non-Parametric z Statistics

Results and Discussion

The significant test result for the companies engaged in producing products is the most important finding in this study. This result indicates that the marketplace believes that the announcement of certification is a harbinger of improved future revenues for these companies. Why should the marketplace make this interpretation for companies which are primarily engaged in production of products, and not companies engaged in provision of services? Recall that improved revenues could stem from reduced costs, or from improved sales. Improved quality will impact both costs and revenues for companies that develop products. Cost reductions would not only include reduced fixed costs to develop the product, but also variable support costs that would increase as volumes of sales increased. Sales would also be expected to benefit from improved quality, which could be a key product differentiator.

On the other hand, companies that are purveyors of services may expect to see increased sales due to third party assurances of quality, but they are often shielded from the effects of costs by the terms of their contracts. Contracts may be either fee for service, fixed price, or some combination of the two. The profit accrued from a fixed price contract is directly affected by incremental costs from problems unforeseen at the time that the contract is formed. A typical example of this sort of incremental cost in software development is the cost of finding and fixing bugs in the software during the test phase. In fee for service contracts the client retains closer control of the project, and also bears the majority of the risk of incremental costs. The closer the contract is to fixed price, the more the profit margin may be expected to be sensitive to costs due to quality issues. Fixed price contracts, however, are seldom delivered as initially specified since the user requirements usually change during the course of the contract, and change requests expand the scope (and cost) of the contract. The increased costs in fixed price contracts accrued to quality problems may then be masked by these changes in scope. This leads to a situation where service providers are more likely to see benefits from increased quality through increased sales, rather than from reduced costs. If the marketplace sees announcements of certification as being significant for product developers, and not for service providers, then it suggests that the marketplace is anticipating a greater effect on reduced costs than for increased sales.

Before summarizing the remaining results, it is necessary to consider how the results may be interpreted. Performing a single test on the data from a group of companies, based on an a priori hypothesis may yield results that are considered statistically significant. Once the test is repeated on varying event windows for a group of companies, however, we have moved to a line of inquiry which is looking for any event window showing results of interest.

Besides the change from testing an a priori hypothesis to looking for any situation with interesting results, there is also a statistical issue that is introduced when we start repeating the analysis for any possible event window in the evaluation period. In this case, the parametric test has been applied to all possible event windows (61) for each group of companies (5) within the evaluation period surrounding the event. Since the significance level of each individual test is expected to yield a false positive in 1 out of 20 cases, (alpha = .05), repeating the test on this number of varying subsets of the same data can be expected to yield approximately 15 false positive results for 300 tests. The risk of false positive results stemming from multiple tests is commonly addressed in statistics by applying a correction to the significance level using a variant of the Bonferroni adjustment (for example, dividing the significance value by the number of tests undertaken). The parametric test results cited in the post-hoc analysis portion of the Results section could be significant if the test that yielded each result had been the only such test performed on the data. However, the approach actually taken was to look at all possible event windows for each group of companies within the evaluation period surrounding the event. Applying a Bonferroni correction, the parametric test results of the post-hoc analyses are not significant, and fail to confirm the research hypothesis. However, the results are still of interest. Without Bonerroni correction, the post-hoc parametric test results appear to support the initial results, in that the companies that produce products are the only group of companies that have any event windows with z-statistics that exceed the critical value for a .05 level of confidence. All of these event windows include the day of the announcement, and there is a suggestion that windows which open one or two days prior to the announcement may also be significant. This may indicate that there is leakage of the certification announcement to the marketplace prior to the official announcement.

This problem with the validity of repeated tests is also often an issue when using Corrado's non-parametric test. Corrado's test examines the likelihood that the day being looked at will attain its rank when the days are ordered by magnitude of average abnormal returns. In the literature this test is commonly applied to all days in the evaluation period, or to all days in an event window which may be a week or more in length. While it is tempting to follow the literature in the use of Corrado's test, it is prudent to take into account an adjustment of the level of confidence using a Bonferroni correction when doing so, or to cautiously interpret the results as being suggestive rather than conclusive. The only group of companies that yielded a non-parametric z-statistic that exceeded the critical value for the .05 level of significance was the group of large companies, which showed such a result for the fifth day after the announcement of certification, although it fails to reach the critical value when a Bonferroni correction is applied. Hence, we cannot take the non-parametric result as being significant; the results are only of exploratory interest. Furthermore, it is noteworthy that the statistic is negative, indicating that the marketplace anticipates a reduction in future earnings following the announcement. This non-parametric result is not consistent with the parametric analysis. When companies certify their business processes, it is often the case that the scope of the certification is limited to a geographic location or a line of business. The larger the business, the smaller the portion of the business that is likely to become certified, and the effect of the certification is likely to be proportionately smaller. Thus, it is unlikely that a very large company would see a significant increase, or drop, in profitability from certification. Possibly the fact that the business has sought certification could be seen by the market as signaling a systemic problem with quality, leading to a revision in the market's assessment of the company's anticipated profitability. Given that the non-parametric result is not evident in the parametric test, and taking into account the statistical issues around repeated tests that were noted above, this result should probably be ignored.

To conclude, the market assessment of ISO 9000 quality certification of software engineering processes supports the hypothesis that the certification does correspond to increased profitability in those companies engaged in developing products. Post-hoc analysis of the data suggests that leakage of the certification announcement takes place.

The findings should be of use to companies engaged in producing products that are looking for a business justification for certification, and also to companies looking to market consulting services geared towards preparing companies engaged in software engineering for a certification audit. It is useful to replicate the study with a different sample, as this may allow confirmation of the post-hoc findings that the announcement of certification leaks out to the marketplace a day or two in advance of the official announcement. It is also interesting to explore the qualitative perceptions of the marketplace to certification of software engineering processes to determine when market observers first become aware of certification efforts and results, and to determine whether the market sees certification as being a glass half full or a glass half empty in the sense that it portends improved productivity or signals hitherto unseen problems in quality.

Event Study 2 - Japanese ISO Certification

This section describes an event study that was carried out on Japanese companies who announced certification to the ISO 9000 standard in the period between July 1994 and October 2000, and who identified software development as being in the scope of the certification.

Data Collection and Analysis

This study was undertaken to see if the results obtained from the American study could be replicated. The data for this study was drawn from the Japanese marketplace. Japan was chosen because the data are readily available, there is a reasonable expectation that the stock markets are efficient (Jacobson & Aaker, 1993; Korkie & Nakamura, 1997; Sadique & Silvapulle, 2001), and I have a personal interest in Japan because I have worked there. Since one of the post-hoc findings of the American study was that the announcement of certification could be leaked, the event window for the Japanese study was broadened to include the day before the announcement of certification, [-1,1].

The Japanese study is similar to the previously described American study. A search of the online registration database at http://www.worldpreferred.com/ for Japanese companies that are certified to the ISO 9000 standard, and who used the word "software" in the description of the scope of the certification, yielded 262 candidate companies. After eliminating duplicates, the remaining companies were checked using Thompson Datastream services to determine which companies were publicly traded at the time of the certification, and had historical stock pricing information available. This narrowed the list of candidate companies down to 20 companies.

The daily stock prices for each of the candidate companies were collected from Thompson Datastream for the range from 261 days before the certification through 5 days after the certification. Additionally, daily index prices for a variety of indices were also collected from Thompson Datastream for each candidate company. The indices that were collected included the TOPIX index, the TSE Services index, the Japanese Information Technology index, and the Japanese Software and Computer Services index for all candidate companies. In addition, for those companies that were traded on the TSE, the exchange's categorization of the companies' line of business was determined and the corresponding TSE index was also collected. These indices include the TSE Construction index, the TSE Machinery index, the TSE Retail index, the TSE Transportation Index, the TSE Wholesale Trade index, and the TSE Electric Machinery index.

The data collected for the candidate companies and the indices was next split into two parts. The estimation period, which was used to build the market model and estimate the standard deviation, included the prices ranging from 261 days before the certification through 6 days before the certification. The evaluation period, which included the a priori event window, included the prices ranging from 5 days before the certification through 5 days after the certification.

Next, a variety of possible market models were built for each candidate company using a linear regression of the indices against the actual stock prices. In each company's case, the regression with the highest Pearson's correlation coefficient was chosen to determine which index was best suited to serve as the market model. The result of this process is shown in Table 3-6, which lists the candidate companies, the market model parameters, and the Pearson's correlation coefficients.

	Company	Index	Alpha	Beta	R	
1	Jastec Co. Ltd	TSE Services	372.693	.653	.638	***
2	Nippon Software Ltd.	Japan IT	-1114.736	3.83	.877	***
3	Computer Engineering & Consulting Ltd.	Japan IT	-1829.494	4.954	.648	***
4	Kajima Corp.	TSE Construction	-162.651	.806	.920	***
5	Meikei Inc.	TSE Services	-367.866	.777	.813	***
6	Goodman Co. Ltd.	TSE Services	1046.929	.9	.622	***
7	Miura Co. Ltd.	Japan IT	1715.71	-.204	.578	***
8	Tsuzuki Tsushin Gijutu Co. Ltd.	TSE Services	-1218.33	1.429	.796	***
9	Mutow Information Center Co. Ltd.	Japan IT	466.688	.226	.702	***
10	Kayaba Industry Co. Ltd.	TSE TOPIX	-586.142	.708	.845	***
11	Japan Business Computer Co. Ltd.	Japan IT	-189.832	2.169	.876	***
12	Japan Information Processing Service Co. Ltd.	TSE Services	142.956	.907	.801	***
13	Sec Corporation	TSE TOPIX	144.003	.105	.601	***
14	Hitachi Information Systems Ltd.	TSE Services	-1343.712	2.026	.953	***
15	OKI Electric Industry Ltd.	TSE TOPIX	-707.189	.813	.957	***
16	Ines Corporation	Japan IT	-545.603	3.395	.888	***
17	Fuso Dentsu Co. Ltd.	Japan IT	1501.002	-.191	.267	
18	Fuji Soft ABC Inc.	Japan S/W & Comp. Services	140.736	5.569	.930	***
19	Sintokogio Ltd.	TSE Services	473.398	-.06882	.891	***
20	Sokkia Co. Ltd.	TSE Services	-1805.75	2.258	.856	***

*** Correlation above .5 threshold.

Table 3-6: ISO Japanese Market Model Parameters

Table 3-6 shows that if we use the same arbitrary threshold of R = 0.5 as was used in the American case, then we loose only one candidate company, Fuso Dentsu Co. Ltd., for having an inadequate market model. A second company, Tsuzuki Tsushin Gijutu Co.

Ltd. was also eliminated as the trading information showed that the company was infrequently traded. Next, the companies were ranked based on their size. (This ranking is shown in Table 3-7). The web site http://japanfinancials.com/ provided some financial data for the top 500 companies in Japan[7]. Using this web site, the candidate companies were ranked based on sales. This resulted in the elimination of 2 more companies, Meikei Inc. and Sec Corp., which were not listed. At this point there were 16 companies remaining in the study, which were grouped into two groups corresponding to small companies which have sales of less than 40,000,000,000 Yen[8], and large companies which had sales above 40,000,000,000 Yen.

Company	Sales (1000 Yen)
Kajima Corp.	2,100,701,000
OKI Electric Industry Ltd.	673,170,000
Kayaba Industry Co. Ltd.	198,602,000
Hitachi Information Systems Ltd.	101,021,000
Mutow Information Center Co. Ltd.	67,737,159
Japan Business Computer Co. Ltd.	67,665,148
Sintokogio Ltd.	67,483,433
Miura Co. Ltd.	46,713,000
Japan Information Processing Service Co. Ltd.	44,024,381
Ines Corporation	33,899,596
Sokkia Co. Ltd.	28,878,185
Fuji Soft ABC Inc.	28,157,010
Computer Engineering & Consulting Ltd.	24,837,002
Nippon Software Ltd.	23,731,376
Goodman Co. Ltd.	9,784,640
Jastec Co. Ltd.	5,635,244

Table 3-7: ISO Japanese Company Size

As was done in the American study, the 16 remaining company's abnormal returns in the evaluation period were plotted to examine them for outliers. This is shown

[7] The website japanfinancials.com obtains the data it provides from the "*yuka shoken hokokusho*", which are the public financial filings which are required by Japanese security laws.

[8] The 40,000,000,000 Yen threshold is equivalent to 364,400,000 US$ at an exchange rate of 0.00911 US$ / Yen. (The average exchange rate in the period from 1994 to 2000 as obtained from OANDA Corporation.)

in Figure 3-2. Four of the companies were identified as being outliers and were eliminated, leaving a final composite pool of 12 companies of interest. Table 3-8 shows how these companies break down into sub-groups based on size and on whether they focus on provision of products or services.

Company	Small Companies	Producing Products
Kajima		✓
Kayaba		✓
Miura		
JIPS		
Ines	✓	
Sintokogio		✓
Sokkia	✓	✓
Jastec	✓	
Nippon SW	✓	✓
CES	✓	
Goodman	✓	✓
Fuji Soft	✓	

Table 3-8: ISO Japanese Company Group Membership

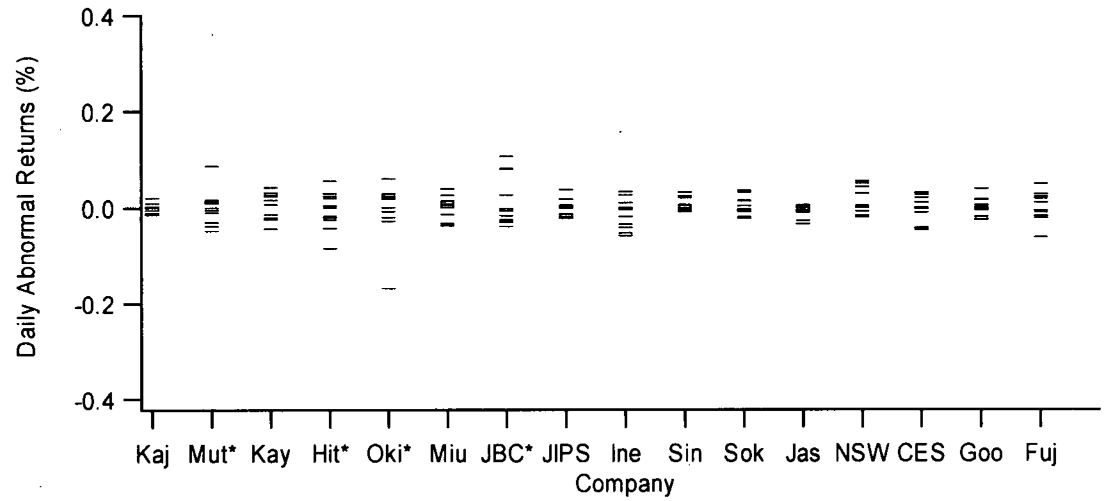

Figure 3-2: ISO Japanese Companies Daily Abnormal Returns

Results and Discussion

The event study analysis was initially carried out using an event window of [-1,1] on the entire pool of Japanese companies, for the groupings of small and large companies, and for the companies that are primarily concerned with the provision of services and products. The only group of companies that had a statistically significant test statistic was the group of companies that focused on provision of services, ($z = -2.298$; $p < .05$).

Following the example set in the North American study, the analysis was repeated for all possible event windows in the period from 5 days before the announcement through 5 days after the announcement. The parametric analysis found one additional event window, [-1,2], that yielded a negative test statistic above the critical value for a .05 level of confidence for the companies primarily focused on the delivery of services ($z = -2.071$). The non-parametric analysis yielded no findings exceeding the .05 critical value. None of the other groups of companies had any event windows where an analysis yielded a test statistic that exceeded the critical value for a .05 level of confidence.

As was explained in the American study, none of these post-hoc results are statistically significant when the level of confidence is adjusted using a Bonferonni correction. However, the findings are of interest, especially where they do or do not support the findings from the American study.

Event Study 3 - North American SEI SW-CMM Certification

This section describes the first of two event studies that were carried out on American companies that announced certification to the Software Engineering Institute's Capability Maturity Model standards. The companies in this study attained certification to the SEI Software Capability Maturity Model standard. These companies were implicitly involved in software development.

Data Collection and Analysis

The first of the SEI event studies was carried out in 2002 and focused on the then current Software Capability Maturity Model, SW-CMM. Companies that had attained some level of SEI SW-CMM certification were identified. The primary source for this information was the SEI's public list of published maturity levels as of June 25, 2001. (http://www.sei.cmu.edu/sema/pub_ml.html) This was augmented with information gleaned from web searches. Specifically, for each company identified, the date of the company's press release or the earliest publication of the information was found, and taken as the date of the event. This yielded a list of 48 companies, 27 of which were in the USA. Each of the American companies was examined to determine if they were publicly traded, which reduced the list to 16 companies. Of these, 4 companies were either listed on the AMEX exchange or on other smaller exchanges, leaving 12 companies which were listed on either the NYSE or the NASDAQ. Historical daily closing prices were obtained for each of these companies using the YAHOO Finance website (http://chart.yahoo.com/d?s=) for a range of days running from 200 days before the event through 5 days after the event. YAHOO provided the data adjusted for dividends, splits, and spin-offs. One of the companies had a significant discontinuity in its stock price during the event window, and was eliminated.

Each of the 11 remaining companies was categorized by sector and industry groupings using the Yahoo Market Guide, and 25 other companies were randomly selected from each of the represented sector and industry groupings to develop the indices required to construct the market models. The daily closing prices for each of the 25 companies in each index were collected using the YAHOO Finance website for range of days running from 200 days before the event through 5 days after the event. The indices were constructed by adding the daily closing prices of each of the component stocks. As in the previous analyses, the market models were developed using SPSS and those companies with a suitable market model were identified. The market models and their correlations are shown in Table 3-9.

Company	Ticker	Level	alpha	beta	R
1 Cognizant Technology Solutions	CTSH	5	28.955	.05997	.57 ***
2 Computer Sciences Corporation	CSC	4	-54.921	.18900	.936 ***
3 Covansys	CVNS	4	.826	.00387	.524 ***
4 Dynamics Research Corporation	DRCO	2	7.855	-.00033	.107
5 Hewlett Packard	HWP	5	-143.989	.28700	.654 ***
6 Mellon Financial Corporation	MEL	2	-31.109	.11100	.258
7 Modis Professional Services	MPS	2	18.522	-.00902	.164
8 Motorola	MOT	5	19.793	.03313	.083
9 Northrop Grumman	NOC	4	-29.149	.14800	.671 ***
10 Rockwell Collins	ROK	3	46.714	-.01272	.034
11 Boeing	BA	5	-19.178	.17100	.852 ***

*** Correlation above .5 threshold.

Table 3-9: SW-CMM American Market Model Parameters

The daily abnormal returns in the evaluation period for the 6 companies with a correlation above 0.5 were then plotted and are seen in Figure 3-3. One of the companies is clearly an outlier. Elimination of this company yields a data set comprised of 5 companies. This is a small data set, but still within the guidelines.

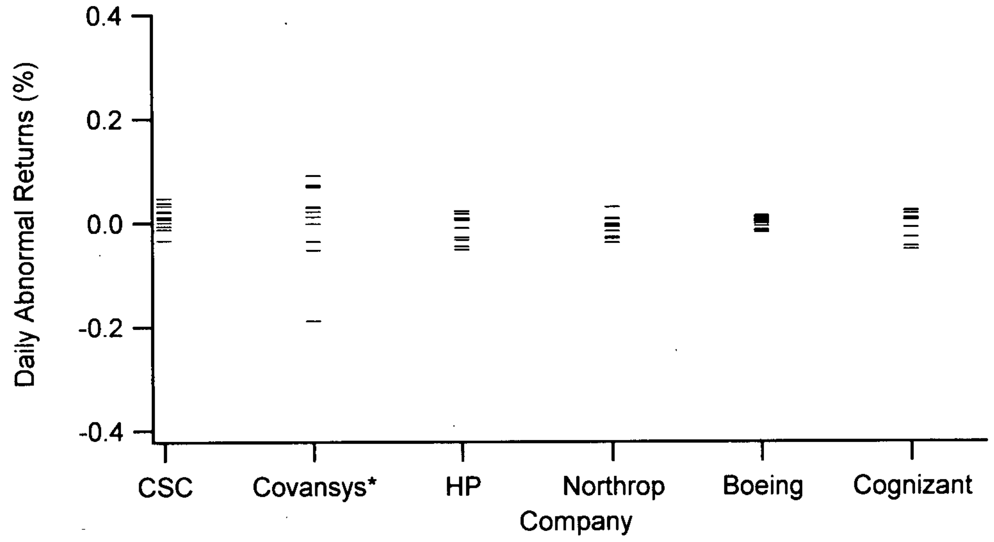

Figure 3-3: SW-CMM American Companies Daily Abnormal Returns

Results

The event study analysis was carried out on the aggregate data set as described previously and no significant abnormal returns were found using either the parametric nor the non-parametric tests.

Event Study 4 - North American SEI CMMI Certification

As described in Chapter 2, the Software Engineering Institute created a successor to the SW-CMM standard called the Capability Maturity Model Integrated, CMMI. A follow-up event study of the companies that had announced certification to the CMMI standard was carried out in 2005.

Data Collection and Analysis

A list of companies that had announced certification to the CMMI were obtained in July 2005 from the SEI website (http://seir.sei.cmu.edu/pars), and augmented by a web search for certified companies. This yielded a list of 350 certifications from 239 companies. Of these, 52 companies were American. Only 16 of these companies were publicly traded on the NYSE or the NASDAQ. Data was collected for each of these companies in the same manner as outlined in the SW-CMM study. The list of these companies, the correlation of their indices, the regression coefficients for their market models, the maturity level certified, and whether the company focuses on provision of products or services is shown in Table 3-10.

	Company	Ticker	Level	alpha	beta	R	Svc or Prod	
1	Advantest America	ATE	2	-2.062	0.616	0.882	P	***
2	Bearing Point	BE	3	1.225	0.274	0.753	S	***
3	Boeing	BA	5	3.31	0.200	0.929	P	***
4	Computer Sciences Corporation	CSC	5	27.53	0.112	0.725	S	***
5	Diebold	DBD	2	22.017	0.493	0.543	P	***
6	General Dynamics	GD	3	-20.099	0.884	0.982	P	***
7	Honda	HMC	2	6.686	0.215	0.789	P	***
8	Honeywell	HON	3	-12.541	0.135	0.930	P	***
9	ADP	ADP	2	20.471	0.134	0.877	S	***
10	JP Morgan Chase	JPM	2	-29.34	0.151	0.970	S	***
11	Lockheed Martin	LMT	5	-16.83	0.291	0.950	P	***
12	NCR	NCR	2	-1.53	0.094	0.969	S	***
13	Northrop Grumman	NOC	5	24.251	0.15	0.656	P	***
14	Raytheon	RTN	3	12.604	0.307	0.803	P	***
15	Reuters	RTRSY	2	-26.579	0.308	0.952	S	***
16	SRA International	SRX	3	-19.154	0.194	0.929	P	***

*** Correlation above .5 threshold.

Table 3-10: CMMI American Market Model Parameters

The daily abnormal returns in the evaluation period were then plotted and are shown in Figure 3-4. None of the companies had significant outliers.

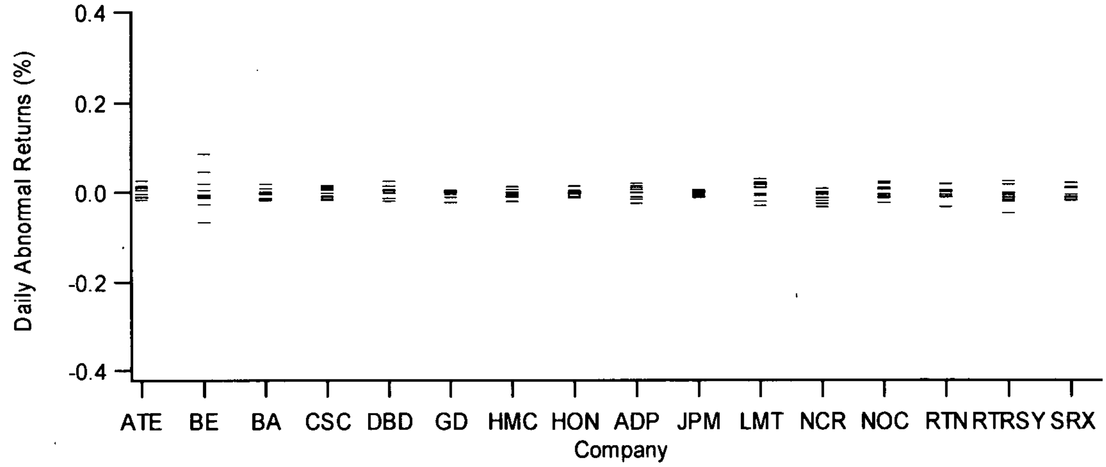

Figure 3-4: CMMI American Companies Daily Abnormal Returns

Results

Three analyses were conducted. First the aggregate dataset was analyzed. Second, the two data sets formed by dividing the companies into two groups concerned with provision of products (10 companies) and services (6 companies) were analyzed. Finally, the two data sets formed by dividing the companies into two groups composed of those companies that attained a maturity level of 3 or higher (9 companies), and those companies that attained a maturity level of less than 3 (7 companies) were analyzed. No significant abnormal returns were found for any of the analyses using either the parametric or the non-parametric tests.

Summary and Conclusions

Out of four studies undertaken, the American and the Japanese ISO 9000 studies both yielded statistically significant results supporting the hypothesis that the market treats the announcement of certification as signaling an increase in future revenues. In the American case, the market response was shown for companies primarily concerned with provision of products and was positive. The Japanese study, however, found a significant negative market response for firms primarily engaged in the provision of services. In both cases, the findings confirm that for some companies certification of software engineering processes has a measurable impact on the marketplaces estimation of future earnings. Under the premise of an efficient marketplace, this is believed to indicate that companies that certify may expect to see a change in future revenue streams. I suggest that in the American case, the positive market response reflects an expectation of reduced costs for companies producing products, while in the Japanese case the negative response reflects a longer term concern with previously unappreciated quality problems in companies primarily concerned in provision of services.

The suggestion that the Japanese marketplace is less concerned than the American marketplace with the short-term cost implications of the certification, and more concerned with longer term issues of quality would be consistent with the findings of a paper that found that motivation for adoption of ISO 9000 varied across countries, and that Japanese companies in general placed a higher value on "quality improvement" than on "corporate image" (Pan, 2003).

Japan was a relatively late adopter of the ISO 9000 standard, and a significant factor in the adoption was exports to Europe (Corbett, 2002). Announcement of certification to ISO 9000 could be interpreted by the Japanese domestic market to imply that the company has such significant quality problems that they have turned to certification to try to improve their situation. Whereas in the American study it was argued that legal contracts shielded services companies from cost implications of quality problems, there is evidence in the literature that the Japanese market relies more on relationships than on legal contracts, and uses sanctions to punish business partners who are seen to fail to live up to their obligations (Hagen & Choe, 1998). Services contracts are more dependent on an ongoing relationship than product sales are, and hence are more susceptible to sanctions over the longer term. The marketplace may then be responding to the financial implications of the anticipated quality problems.

There is a suggestion in the literature that Japanese software engineering is less concerned with driving strategic change in business processes and is more directed towards incremental solutions that support existing business processes, in contrast to the situation in North America (Bensaou & Earl, 1998). Under this hypothesis, the business processes are less malleable in Japan, and hence less likely to adapt to account for quality problems in associated information systems. Steps taken by individual companies to address quality problems could then be viewed in the market as more serious in nature, resulting in a greater impact in Japan. In contrast, in America the marketplace is more likely to view the consequences of certification as being strictly financial in nature.

A search of the literature for related studies revealed an interesting paper that examined manufacturing and non-manufacturing companies across all industrial sectors in the United States (Ho, Keh, & Ong, 2005). This study found that investment in research and development contributed to the value of the manufacturing firms but not to the non-manufacturing firms, while investment in advertising benefited the non-manufacturing firms and not the manufacturing firms. My finding indicated that American firms engaged in software engineering that are primarily focused on development of products showed a positive market response to certification of software engineering processes. This clearly parallels Ho et. al.'s findings regarding investment in research and development. However, I did not find evidence supporting their finding that non-manufacturing firms benefited from investment in advertising. All told, Ho et al.'s study may be interpreted to support my supposition that the American market interprets announcements of certification of software engineering processes as signaling increased future revenues from reduced costs rather than increased sales.

I have suggested that the difference between the American and the Japanese findings may reflect a difference in the Japanese market's attitude towards the relatively short-term cost implications of certification announcements compared to its reaction to the possible implications of the announcement with respect to hitherto unanticipated possible quality problems. In a paper looking at the impact of product development outcomes in the American pharmaceutical industry on market valuation, a strong asymmetry of response was found (Sharma & Lacey, 2004). Specifically, a negative market response to announcements of new product failures was found to be significantly stronger than the positive market response to new product successes. If this asymmetry can be generalized to the Japanese market then it may help to explain why the negative interpretation of the significance of the announcements may overwhelm the positive impact of the announcements.

To summarize the discussion of possible differences between the American and the Japanese reactions to announcements of certification of software engineering processes, consider the effects to be split between marketing effects and quality effects. In the American case an argument can be made that firms focusing on production of products see benefits both in marketing and quality, while firms focusing on services only see benefits in the area of marketing since the impact of improved quality is masked by contractual conditions in the case of services. This results in the maximum benefit being seen in the companies producing products. Japanese companies rely more on relationships than legal contracts, and the impact of changes in quality in the product area would be different. In the Japanese case the effects of the certification are generally negative; the market views the announcement either as an indicator of previously unknown quality problems or as an indicator that the firm intends to move into a new and potentially risky market that may require the certification. Both of these negative reactions may come to bear in both marketing and quality effects of services and in the quality effects of products, but will have relatively little impact on the marketing impact of products since products in Japan are more typically targeting at a domestic market which will mostly disregard a foreign certification. This leaves the Japanese stock market seeing a predominantly negative effect for companies that focus on provision of services. Further research would need to be carried out to determine if this argument is valid.

The major post-hoc finding is that there is very likely a leakage of the announcement of certification, inasmuch as the market appears to start reacting to the announcement the day before it is made. In fact, if this had not been suggested in the first study, then the second study would likely have made an a priori choice of event window that would not have yielded a significant result.

While comparing the American and the Japanese ISO 9000 studies, it is worth noting that although both studies compared larger companies to smaller companies, the manner in which the companies were ranked was not completely equivalent. The metric used was different, the accounting methods used to arrive at the metric were likely different, and the sizes of the companies that fell into the smaller or larger groups in the two countries was not the same. The purpose of creating the groups was to be able to compare the larger companies and the smaller companies within each country, not to make comparisons between the countries.

The event studies of announcements of SEI certification have not yielded any statistically significant results indicating abnormal returns. What do these findings mean? One possibility is that while the data sets are large enough to produce statistically valid results given a strong market response, they may be small enough that the analysis may not uncover a weak market response. This is more likely in the case of the SW-CMM study where the data set was comprised of only 5 companies, which is the smallest size that the literature supports, whereas in the CMMI study the data sets are significantly larger. A stronger possibility is that the market greets the announcement of CMM certification with a lack of interest.

CMM certification is fundamentally different from ISO 9000 certification in that companies who choose to certify often choose not to publicly advertise the fact. The ISO 9000 registrars and the consulting companies that assist with ISO 9000 certifications routinely publish the names of certified companies. This may be viewed as encouraging other companies to follow suit, or to encourage customers to adopt ISO 9000 certification as a vendor requirement which in turn will increase adoption of the standard. The SEI does not release any information on who has become certified to the CMM series of standards without their explicit permission. Indeed the SEI prefers not to refer to the results of the examination of a company by an auditor as a "certification" at all. The 2002 SEI Annual Report claims that over 5,000 organizations world-wide had used the

SW-CMM for appraising and improving software development processes, yet in 2001 I was only able to identify 48 companies who had announced what level of maturity the audit had found.

It is informative to look at the industry press to gauge the industry attitude towards CMM certification. In a comparison of IS quality frameworks one article speaks favorably about the SW-CMM standard characterizing it as being low in level of abstraction and specific in IT relevance, whereas the ISO 9000 standard is characterized as being highly abstract and only vaguely relevant to IT (Anthes, 2004). In another article examining CMM ratings, the standard receives high marks, but readers are cautioned that they may not see much benefit from dealing with highly ranked companies unless they are themselves highly ranked (King, 2003). A third article is frankly dismissive of CMM rankings, criticizing the audit process as being corrupt and the results unreliable (Koch, 2004a). While these publications are not rigorous scientific journals, they are in fact more likely to influence market opinion than will a peer reviewed journal. The range of opinions, and the resistance of the industry to adoption of CMM standards, is likely to lead the marketplace to discount the standard's utility. It is very likely a case of the standard not only needing to be useful, but also needing to be perceived to be useful, in order to influence the market.

References

Adams, G., McQueen, G., & Seawright, K. (1999). Revisiting the stock price impact of quality awards. *Omega, 27*(6), 595-604.

Anderson, S. W., Daly, J. D., & Johnson, M. F. (1999). Why firms seek ISO 9000 certification: Regulatory compliance or competitive advantage. *Production and Operations Management, 8*(1), 28-43.

Anthes, G. H. (2004). *Quality Model Mania.* Retrieved May 3, 2005, from http://www.computerworld.com/managementtopics/management/story/0,10801,90797,00.html

Armitage, S. (1995). Event studies methods and evidence on their performance. *Journal of Economic Surveys, 8*(4), 25-52.

Beirao, G., & Cabral, J. A. S. (2002). The reaction of the Portuguese stock market to ISO 9000 certification. *Total Quality Management, 13*(4), 465-474.

Bensaou, M., & Earl, M. (1998, September-October). The right mind-set for managing information technology. *Harvard Business Review,* 119-128.

Brown, S. J., & Warner, J. B. (1980). Measuring security price performance. *Journal of Financial Economics, 8*, 205-258.

Brown, S. J., & Warner, J. B. (1985). Using daily stock returns - The case of event studies. *Journal of Financial Economics, 14*, 3-31.

Campbell, C. J., & Wesley, C. E. (1993). Measuring security price performance using daily NASDAQ returns. *Journal of Financial Economics, 33*(1), 73-92.

Corbett, C. J. (2002). *Global diffusion of ISO 9000 certification through supply chains* (Working Paper). Los Angeles: UCLA Anderson School of Management.

Corrado, C. J. (1989). A nonparametric test for abnormal security-price performance in event studies. *Journal of Financial Economics, 23*, 385-395.

Cowan, A. R., & Sergeant, A. M. A. (1996). Trading frequency and event study test specification. *Journal of Banking & Finance, 20*(10), 1731-1757.

Docking, D. S., & Dowen, R. J. (1999). Market interpretation of ISO 9000 registration. *Journal of Financial Research, 22*(2), 147-160.

Fama, E. F., Fisher, L., Jensen, M. C., & Roll, R. (1969). The adjustment of stock prices to new information. *International Economic Review, 10*(1), 1-21.

Hagen, J. M., & Choe, S. (1998). Trust in Japanese interfirm relations: Institutional sanctions matter. *Academy of Management Review, 23*(3), 589-600.

Hendricks, K. B., & Singhal, V. R. (1996). Quality Awards and the market value of the firm: An empirical investigation. *Management Science, 42*(3), 415-436.

Ho, Y. K., Keh, H. T., & Ong, J. M. (2005). The effects of R&D and Advertising on firm value: An examination of manufacturing and nonmanufacturing firms. *IEEE Transactions on Engineering Management, 52*(1), 3-14.

Jacobson, R., & Aaker, D. (1993). Myopic management behavior with efficient, but imperfect, financial markets: A comparison of information asymmetries in the U.S. and Japan. *Journal of Accounting and Economics, 16*, 383-405.

King, J. (2003). *The pros and cons of CMM.* Retrieved May 3, 2005, from http://www.computerworld.com/managementtopics/management/project/story/0,10801,87882,00.html

Koch, C. (2004). *Bursting the CMM hype*. Retrieved May 3, 2005, from http://www.cio.com/archive/030104/cmm.html

Korkie, B., & Nakamura, M. (1997). Block holding and keiretsu in Japan: The effects of capital markets liberalization measures on the stock market. *Journal of International Money and Finance, 16*(1), 113-140.

Kuilboer, J. P., & Ashrafi, N. (2000). Software process and product improvement: An empirical assessment. *Information and Software Technology, 42*, 27-34.

Martinez-Costa, M., & Martinez-Lorente, A. R. (2003). Effects of ISO 9000 certification on firms' performance: a vision from the market. *TQM & Business Excellence, 14*(10), 1179-1191.

Maynes, E., & Rumsey, J. (1993). Conducting event studies with thinly traded stocks. *Journal of Banking & Finance, 17*(1), 145-157.

Nicolau, J. L., & Sellers, R. (2002). The stock market's reaction to quality certification: Empirical evidence from Spain. *European Journal of Operational Research, 142*(3), 632-641.

Pan, J.-N. (2003). A comparative study on motivation for and experience with ISO 9000 and ISO 14000 certification among Far Eastern countries. *Industrial Management and Data Systems, 103*(8-9), 564-578.

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.

Ramasesh, R. V. (1998). Baldrige Award announcement and shareholder wealth. *International Journal of Quality Science, 3*(2), 114-125.

Sadique, S., & Silvapulle, P. (2001). Long-term memory in stock market returns: International evidence. *International Journal of Finance and Economics, 6*, 59-67.

Schmauch, C. H. (1994). *ISO 9000 for software developers.* Milwaukee, Wisconsin: ASQC Quality Press.

Sharma, A., & Lacey, N. (2004). Linking product development outcomes to market valuation of the firm: The case of the U.S. pharmaceutical industry. *The Journal of Product Innovation Management, 21*(5), 297-308.

Simmons, B. L., & White, M. A. (1999). The relationship between ISO 9000 and business performance: Does registration really matter. *Journal of Managerial Issues, 11*(3), 330-343.

Soteriou, A. C., & Zenios, S. A. (2000). *Searching for the value of quality in financial services* (Working Paper). Philadelphia, PA: The Wharton Financial Institutions Center, University of Pennsylvania.

Stelzer, K., Mellis, W., & Herzwurm, G. (1997). A critical look at ISO 9000 for software quality management. *Software Quality Journal, 6*(2), 65-79.

Terziovski, M., Samson, D., & Dow, D. (1997). The business value of quality management systems certification: Evidence from Australia and New Zealand. *Journal of Operations Management, 15*(1), 1-18.

Yang, Y. H. (2001). Software quality management and ISO 9000 implementation. *Industrial Management and Data Systems, 101*(7), 329-338.

Chapter 4 - Antecedents to Certification

Introduction

As noted in previous chapters, since both the cost and incidence of failure of information systems projects are so high it seems obvious that efforts to improve the quality of software engineering processes should be a high priority in industry. In spite of this, relatively few companies choose to certify their software engineering processes to the two most common quality standards (ISO 9000 or SEI CMM). In order to try to understand why this is, it is useful to gain a better insight into what prompts a company to choose to certify their software engineering processes.

This chapter describes a study of the antecedents to certification. Initially a model of antecedents was developed, and then a survey instrument was designed to test the model. The survey was conducted in Canada. Based on a factor analysis of the responses, a series of indices was created that were valid on both a discriminant and convergent basis. The results were analyzed to determine which of the model components were statistically significant. The following sections describe the model, the methodology, the analysis, and the results.

Literature Review and Model of Antecedents

This section looks at reasons that may explain why companies choose to certify their software engineering processes, and posits a number of hypotheses.

The model of antecedents is broadly based on an economic model that states that all companies will attempt to maximize their profits subject to unavoidable constraints. The profits may be directly maximized by increasing sales, or by reducing costs (Anderson et al., 1999). Prior work in the general field of ISO 9000 certification has found that improving quality, which may be considered to be likely to decrease costs, is a more important motivation than improvements in reputation, which are linked to increasing sales (Casadesus & Karapetrovic, 2005; Eral & Ghosh, 1997; Gotzamani & Tsiotras, 2002). The proposed antecedents are based on the outcomes of the previous study, and prior work in related fields. All of the antecedents in the model are based on economic arguments. The theoretical constructs used in the hypothesized antecedents are shown in Figure 4-1.

The first hypothesis stems directly from the event study described in the previous chapter. In that study, US companies that produced products, as opposed to services, saw a statistically significant increase in their market value when ISO 9000 certification was announced. This leads to the first hypothesis:

H1: Companies that produce products are more likely to choose to certify than are companies that provide services.

The next reason to certify stems from competitive pressures. Many companies feel that when customers face a sea of vendors, any distinction that differentiates a company from the mass of competitors has the potential to increase sales. This has been discussed at length by Michael Porter in his strategies for competition (Porter, 1998). Specifically, when a potential customer is looking for a supplier, a third party certification that one company has demonstrated that they follow a process designed to ensure a quality product will boost that supplier's perceived qualification. In effect, the certification could serve to differentiate the company, and result in an "early adopter advantage" by increasing the volume of sales and the profit margin by reducing competition.

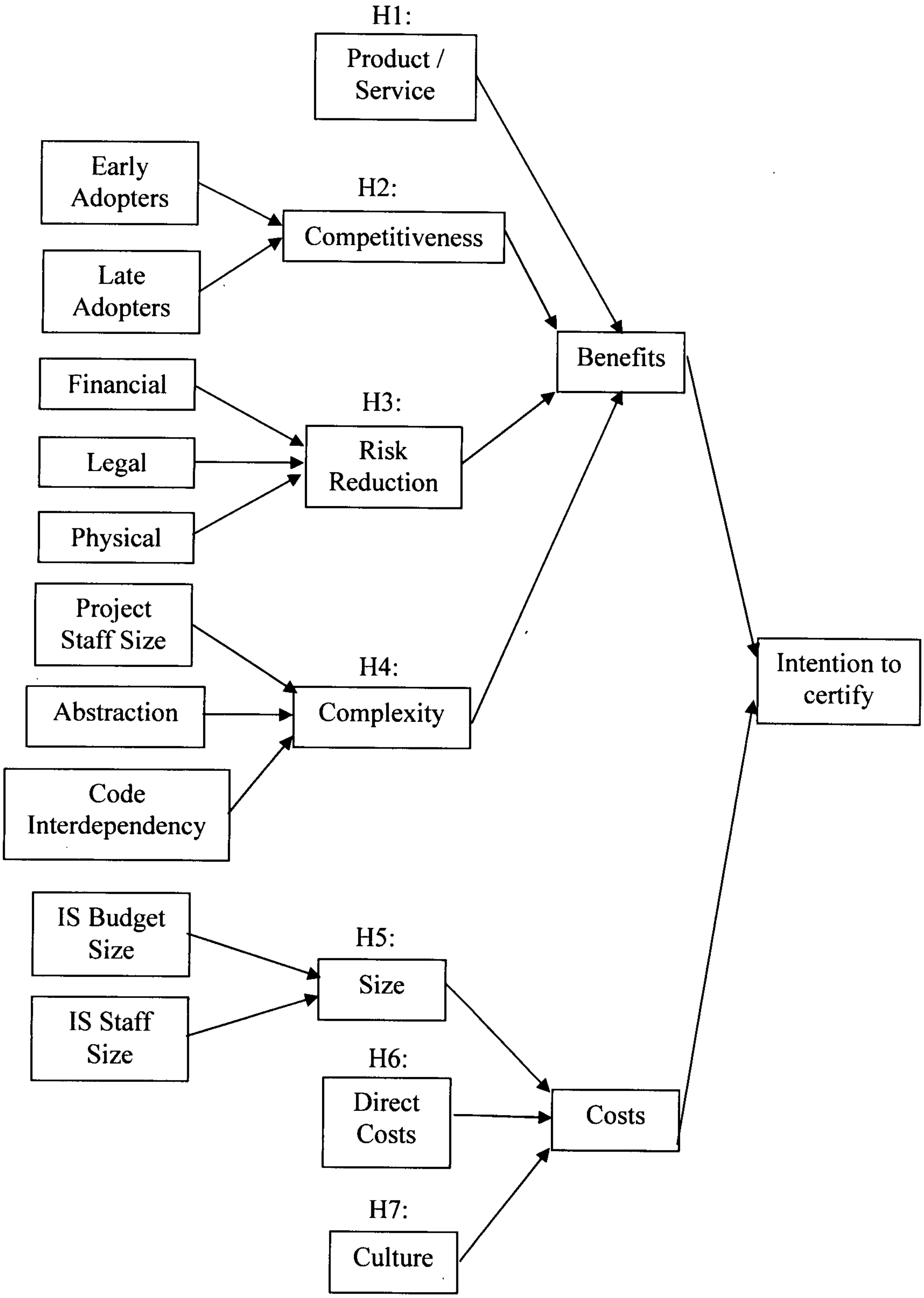

Figure 4-5: Theoretical Constructs Influencing Intention to Certify

Alternatively, if a company's competitors have chosen to certify, then the company may feel a need to "catch-up", and will choose to certify to maintain parity with its competitors. This line of reasoning is consistent with the prediction from institutional theory that the first adopter of a new management practice will be motivated by a desire to resolve a problem, while subsequent adopters will be motivated by external pressures to mimic the early adopter (DiMaggio & Powell, 1983; Oliver, 1991). This has been previously investigated in a study of Canadian and American companies in all industries who have sought ISO 9000 certification (Naveh, Marcus, & Moon, 2004). Although Naveh failed to find a significant difference in performance for first adopters as compared to companies that subsequently decided to certify, it is reasonable to assume that companies in different industries may respond differently. This leads to the second hypothesis:

> *H2: Companies that are in a competitive market will be more likely to choose to certify than those companies that have relatively little competition.*

It has been noted in previous chapters that certification has significant financial and personnel costs. Businesses most often look at new projects and initiatives from a cost/benefit point of view. Companies that perceive that they stand to gain a greater benefit from certification are more likely to choose to incur the costs of certification. One instance of this is that companies that experience a greater degree of risk (financial, legal, or physical) associated with their software development projects will see a greater potential for benefit if they believe that certification will mitigate those risks. The more that a company perceives that a project failure will result in a loss, the more the company will be likely to incur the cost of certification to mitigate the risk. To the best of my knowledge, prior research has not investigated risk as a pre-disposing factor for ISO 9000 certification in the field of software engineering, or in industry in general. This leads to the third hypothesis:

> *H3: Companies that have a higher element of risk in their projects will be more likely to choose to certify than those companies that have relatively little risk.*

While the previous hypothesis looks at the perception of the cost of a project failure, it is also possible to look at the likelihood of failure. It stands to reason that more complex projects have more things that can go wrong than do very simple projects. In the more general product development field, research has focused on measuring complexity as a function of the degree of interdependence of project objectives, the amount of experience that the company has with the technology or type of project, and the "difficulty" of the project (Tatikonda & Rosenthal, 2000). Tatikona and Rosenthal's study also found that only prior experience had a correlation with project success. In my study, complexity was quantified in terms of the number of people on the project, the degree to which the software code has interdependencies between parts (loosely coupled versus tightly coupled design), or the degree of abstraction of the software tools that are used in the project. It makes sense that as more people work on a project, the more activities there will be to coordinate, and the more chances there will be for inter-task dependencies to delay the project. If a project is very simple, then the likelihood that something will go wrong is less than if a project is very complex. Loosely coupled software architectures mitigate the problems stemming from large project teams by minimizing the dependencies between the work of sub-teams. Finally, software that is developed at a higher level of abstraction is easier to understand than is software developed at a lower level of abstraction. This can be seen by comparing software written in machine code, to software written in a third generation language such as C, to software developed using a fourth generation language or environment such as SQL*Forms. The higher the level of abstraction, the closer the code is to the actual business process description. Overall, on a project that has a large number of workers, has a high degree of dependence between components, or has a large number of supported functions, the opportunity for errors and delays in any one part of the project to affect other parts of the project increases. The companies that typically deal with complex projects face a higher likelihood of project failure, and consequently should be more likely to choose to resort to certification to cope with complexity. This leads to the fourth hypothesis:

> *H4: Companies that have more complex projects will be more likely to choose to certify than those companies that have relatively less complex projects.*

Certification costs money, and consumes staff time. Ravichandran argues that IS department size is an important factor in adoption of IS innovation (Ravichandran, 2000). He suggests that larger IS departments have the critical mass required to support innovation. This critical mass is manifested as financial and human resources, which can be allocated to support the innovation. Given the high costs of certification, this argument could also be made in the case of software engineering certification. A New Zealand study of ISO 9000 certified manufacturers found that while almost two thirds of the certified companies were considered small (less than 100 employees), the small companies were more motivated by increasing sales rather than reducing costs (Lee & Palmer, 1999). On the other hand, a very large IS organization could be viewed much like a super-tanker. It is difficult for either the large IS organization or the super-tanker to change course. Hence IS department size is associated with two forces. This leads to the following two alternative fifth hypotheses:

> *H5a: Larger companies are more likely to choose to certify than are smaller companies.*
>
> *H5b: Smaller companies are more likely to choose to certify than are larger companies.*

When a company is considering certification, the anticipated costs will be proportional to the degree of change to be made to the company's business processes and culture. Many companies operate in a relatively undocumented, ad-hoc manner. Many surveys of companies' software development processes find that they are relatively immature from the SEI CMM point of view (Reifer, 2004; SEI, 2005a, 2005b). Companies that have repeatable processes often rely on the experience of their employees rather than having formally documented processes. This is borne out in the literature by the estimation that a company that is ISO 9000 certified is roughly equivalent to a company that is at a CMM level 3 (van der Pijl et al., 1997). If a company is able to become certified with very little change to the way they do business, then the direct costs of certification will be less than if they have to make substantial changes to the status quo. This leads to the sixth hypothesis:

H6: Companies that anticipate lower direct costs of certification are more likely to choose to certify than companies that anticipate relatively higher direct costs.

Finally, a company that has a highly entrenched culture of quality is less likely to resist certification, and the certification effort is more likely to be supported by management. Management support has frequently been cited as a critical success factor in certification initiatives (Ahire & Ravichandran, 2001; Quazi et al., 2001; Rao, Solis, & Raghunathan, 1999). This leads to the seventh hypothesis:

H7: Companies with a corporate culture that values quality are more likely to choose to certify than companies that value quality less.

Methodology

This section describes how the data was collected, and the process used to develop indices[9] that are suitable to test the hypotheses.

The model of antecedents from the previous section was shown to six software industry experts in the Spring of 2003 to determine if there were any further antecedents that needed to be considered. In each case the practitioners dismissed some of the

[9] In this section the terms *measures*, *indices*, *dependent variables*, and *independent variables* are used. Different disciplines follow different conventions in their use of terminology, and the potential exists for confusion to be created. In this work I have postulated a number of hypotheses. Here each hypothesis links one dependent variable, and one independent variable. These variables instantiate constructs such as *competitiveness* or *corporate culture.* The research measures the subject's self assessment of these variables through measures which may be composed of either a single question or an index based on a number of questions from the survey.

In the event studies described in Chapter 3 the term index was used to describe a measure of the price of a group of companies that are thought to share characteristics of the company of interest. (By measuring more than one company, small idiosyncratic fluctuations in stock price were smoothed out.) In this Chapter the term index is used to describe a measure created from a group of questions that all assess a common variable. A summary of the independent variables, the hypotheses that they relate to, and how the measures that assess the variables are formed is provided in Table 4-4 following the description of how they were determined.

hypothesized antecedents as being irrelevant, but none of the practitioners were able to add any new antecedents to the model. None of the hypothesized antecedents were dismissed by all of the practitioners. The model was then instantiated as a survey, which is included in Appendix III.

Participants

In the Fall of 2004, and again in the Spring of 2005, the pool of potential subjects was drawn from the online Strategis database that is maintained by Industry Canada. This database is designed to allow selection of Canadian companies based on their industry focus. Companies voluntarily register in the database. For the purpose of the research, the companies that were identified as candidates for the study were listed under the category "All Software Companies" within the "Information and Communications Technologies (ICT)" portion of the database (http://strategis.ic.gc.ca/epic/internet/inict-tic.nsf/en/h_it06130e.html). This database search produced an initial list of 1913 eligible companies.

Each company's database listing was screened to determine the company size and suitability for the study. To be considered for the study the company needed to appear to be involved in software engineering. Some of the companies that were in the database were actually primarily focused on manufacturing cable, printed circuit boards, or electronic components; these companies were eliminated. Other companies were involved in services such as publicity, legal, marketing, or management consulting; these companies were also eliminated. Since companies that are very small are unlikely to be able to afford the resources to certify, companies that had less than 4 employees were also eliminated from the list of potential participants. To remain on the list of candidate companies the database listing needed to identify a senior executive who could be contacted. In the event that the database did not list the names of executives, then the company's web site was examined to see if the corporate leadership and a suitable contact person was identified. This screening reduced the list of candidates to 235 qualified contacts, representing 235 separate companies.

Each qualified contact was called, although no message was left for those who did not answer the phone. If a candidate could not be reached, then the contact listing was returned to the pool of candidates to be called again when the company name came up again in rotation. (Some of the candidates were called twenty times before the study concluded.) Once a contact was reached on the phone, the study was explained and his/her participation was solicited. If the contact agreed, then the consent information was read to him/her and the survey was then administered. The consent information was subsequently sent by paper mail or email so that the participant would have a record of the study and how to reach me in the future. (The consent form is included in Appendix IV.)

In total, 690 calls were made. Not all candidates who were reached on the phone agreed to participate. Out of the 235 candidates originally identified, 79 were never reached. (It appears that they never answer the phone, perhaps because they use voicemail to screen calls.) Out of 156 executives who were reached, 100 chose to participate, 20 turned out not to be suitable candidates for the study (because their company did not engage in software engineering, or because the individual no longer worked for the company), 17 declined to participate, and 19 said they would consider participation but were not subsequently reachable. The 19 who were subsequently unreachable were construed to have declined to participate without wishing to actually say they would not, so the category of those who declined reached a total of 36. These response rates by category are shown in Table 4-1.

Candidates who were reached	
Agreed to participate	100
Not suitable to participate	20
Declined	36
Candidates who could not be reached	79
Total	235

Table 4-1: Breakdown of Survey Candidates

As previously indicated, the most senior people possible were contacted in each company. Not all companies use the same organizational structure, but the breakdown of the survey participants by job function is given in Table 4-2.

C-level Managers (COO, CEO, CTO)	21
President	14
Vice President	34
Partner	1
Director or Senior Manager	30
Total	100

Table 4-2: Breakdown of Survey Participants by Job Title

Measures

The dependent variable was the likelihood to certify. This was true (1) if the company was currently certified to either the ISO 9000 standard or the SEI CMM standard, or if the company had indicated that they were somewhat likely or very likely to certify to either of the standards in the future, and it was false (0) otherwise. The questions used to assess likelihood to certify were question 1, question 1.1, question 2, and question 2.1. Only 8 companies indicated that they were certified to the ISO 9000 standard, and 3 claimed SEI CMM certification. Of the remaining companies, 31% indicated that they were likely to certify in the future, while 69% of the respondents indicated that they were not likely to certify.

Three independent variables were based on questions in the survey which assessed matters of fact rather than opinion or attitude. Before these questions of fact could be used as measures, the distribution of the responses had to be examined, since questions that lack sufficient variance would add little to the regression analysis, although they could be of interest from a descriptive point of view. A description of these questions and the distribution of responses follow.

The first independent variable, "Product", indicates whether the company is primarily focused on production of products or delivery of services. This measure was assessed by a review of the companies' web sites after the survey was complete.

The second independent variable is a measure of the complexity of the software that is being developed. This variable may be represented as either a bivariate, or a continuous measure. The bivariate form, "ComplexityB", was considered true (1) if the typical project team size was greater than 10 and false (0) otherwise. The distribution of responses to ComplexityB was False:69, True:31. The continuous form, "Complexity", was the response to the open-ended question 6.1. The two questions, question 6.2 and question 6.3 were designed to also assess other aspects of complexity such as functional complexity and the degree to which the software is composed of independent modules as opposed to a single monolithic piece of code. Neither question 6.2 nor question 6.3 had sufficient variance to add any value to the analysis and were not included in the analysis.

The third independent variable, “Size”, was intended to use the responses to the questions 6.4 and 6.5 respectively. A very large percentage of respondents declined to reveal budget figures (question 6.5), so the measure relied on responses to question 6.4 (staff size) only.

Four further independent variables were assessed through indices formed from collections of questions concerning opinions or attitudes. In general, a measure may assess a matter of fact, or a matter of opinion or attitude. The first three independent variables (described above) were assessed by measures of matters of fact. Examples of questions of fact are the annual budget, and the number of employees. In contrast, the next four independent variables were assessed by measures of matters of opinion or attitude. Examples of questions of opinion or attitude were the level of agreement to a statement such as “Our customers are likely to associate certification with a higher quality of product”, or “Quality is important, even if increasing quality will reduce profits in the short term”. Questions assessing attitude or opinion are much less precise than questions of fact, and it is because of this that several such questions are usually used to form an index. Having created questions that are believed to probe an underlying attitude, there is a risk that the group of questions forming an index may actually assess more than one attitude or that several different indices may in fact assess the same attitude.

Survey responses to these questions concerning opinions or attitudes were analyzed to validate the proposed indices, or to modify the composition of the indices to ensure their validity was satisfactory. The independence between the indices is referred to as discriminant validity, and the coherence of each index is referred to as convergent validity.

The data from the surveys were entered into an Excel spreadsheet and imported into the R statistical software package (Dalgaard, 2002; Fox, 2002). R is a public domain version of S or S-Plus statistical software. It is widely used in fields of statistics and social sciences. The version used was 1.9.1 running on a Dell Dimension 4500 PC with the Windows XP Professional operating system version 5.1.

In order to form indices with maximum discriminant validity from the questions that surveyed opinion or attitudes, the 15 survey questions from 5.1 through 5.14 and 6.7 were subjected to an exploratory factor analysis. This was carried out using the "factanal" routine from the "stats" package in R, using maximum-likelihood factor analysis with varimax rotation. This resulted in the questions falling into 4 factors with the component loading matrix shown in Table 4-3. Based on the factor analysis, four potential indices were identified. The questions that have a high loading for the first factor (questions 5.3, 5.4, 5.5 and 5.9) all pertain to the degree of competitiveness of the marketplace. The second factor has high loadings for questions that correspond to the degree of risk should the project fail to succeed (questions 5.7 and 5.8). The third factor corresponds to a culture of quality in the company (questions 5.12 and 5.14). The fourth factor has two questions that have a significant loading, but in this case one question (question 5.10: how familiar the respondent is with the standard) acts as a modifier of the second question (question 5.11: how much would need to be changed to attain certification).

	Loadings			
	Factor 1	Factor 2	Factor 3	Factor 4
Q5.1			-0.174	
Q5.2	0.201	-0.244	-0.258	
Q5.3	0.759		0.145	0.157
Q5.4	0.766			
Q5.5	0.515		-0.416	0.209
Q5.6		0.194	-0.212	
Q5.7		0.511		
Q5.8		0.981	0.172	
Q5.9	0.661	-0.229		-0.311
Q5.10	0.145	0.123		0.633
Q5.11	-0.153			0.585
Q5.12			0.705	
Q5.13			-0.223	-0.196
Q5.14		0.106	0.623	0.275
Q6.7				0.445

Table 4-3: Component Loading Matrix

Note that although the survey was constructed with 15 questions probing opinion or attitude, only 10 questions ended up making a constructive contribution to the indices which formed the measures. In spite of this, sufficient questions were retained to form measures corresponding to all initial hypotheses.

In order to assess the convergent validity of the indices pertaining to measures of competitiveness, risk and culture, Cronbach's Alpha (Cronbach, 1951) for each of the indices was calculated as shown in Table 4-4. Cronbach's Alpha is a commonly used metric to measure the convergent validity of a group of questions, and a value of .70 is a conventional threshold (Nunnaly, 1978), although in practice the threshold chosen is dependent on a number of factors. The way that the coefficient is calculated means that, all other things being equal, it will be higher for larger sets of questions. It is also generally felt that the standard of reliability required for cognitive measures such as intelligence or achievement needs to be higher than for measures of attitudes. For this work, the criterion for validity was chosen to be .60. All three indices have a Cronbach's Alpha value above .60. (The questions making up the index pertaining to direct costs are not probes of a similar construct, insofar as the first question queries the respondent's ability to answer the second question as noted above. As such, the questions would not be expected to have convergent validity and Cronbach's Alpha is not appropriate to assess the index.) The independent variables that the factors represent are noted in Table 4-4. These became the fourth through sixth independent variables as shown in Table 4-5. Table 4-5 shows the independent variables, the hypothesis that each index tests, and how each measure is formed.

Factor	Independent Variable	Survey Questions	Cronbach's Alpha
1	Competitiveness	5.3, 5.4, 5.5, 5.9	0.75
2	Risk	5.7, 5.8	0.65
3	Culture	5.12, 5.14	0.66

Table 4-4: Cronbach's Alpha

Independent Variable	Hypothesis	Composition of Measure
1. Product	H1	Assessed from web after survey completed.
2. Complexity ComplexityB	H4	Q6.1 Q6.1>10
3. Size	H5	Q6.4
4. Competitive	H2	Q5.3+Q5.4+Q5.5+Q5.9
5. Risk RiskB	H3	Q5.7+Q5.8 Q5.7>3 OR Q5.8>3
6. Culture	H7	Q5.12+Q5.14
7. Direct Cost	H6	(0.2*Q5.10)*Q5.11

Table 4-5: Measures

Results

This section examines some descriptive statistics describing the survey results, and details the analysis carried out to test the hypotheses.

Descriptive Statistics

Besides testing the posited hypotheses, a number of descriptive statistics such as the mean and standard deviation of the responses bear examination.

It is noteworthy that relatively few Canadian companies are required to certify due to legislative or customer requirements[10] (4%). Excluding those companies, only 6% of the companies surveyed had certified their processes, including 4% who chose to certify to an ISO 9000 standard, and 2% who chose to certify to an SEI CMM standard. (All descriptive statistics were derived from the data set excluding companies that are required to certify.) In comments received during the administration of the survey, many companies stated that they felt that the ISO 9000 standard was too focused on its roots in manufacturing where the goal was to ensure that every product produced was exactly the same as every other product that comes off of the production line. (These comments were recorded when offered, but were not solicited. The comments are noted as a qualitative note rather than a quantitative measure.) Software engineering was felt to be more inherently creative, with each product reflecting the unique requirements of the business processes that the software would be supporting. Another criticism of ISO 9000 was that the standard was designed to ensure a uniform process, but not necessarily a good process. In particular, the uniform process that the standard supports is documentation heavy, and it was often noted that the cost of maintaining the

[10] All of the companies that were required to certify were in the medical devices, the aerospace, or the defence markets.

documentation was felt to exceed the benefit derived. Relatively few companies surveyed had experience with, or knowledge of, the SEI CMM family of standards.

Another interesting pattern of response dealt with companies that had a software development methodology. When asked if they had a methodology, 83% of the companies indicated that they did. When the companies that indicated that they had a software development methodology were asked what percentage of their projects followed the methodology, the average response was 87%, although the distribution was far from normal (SD = 31%, median = 100%). Only 47 companies claimed to have 100% adherence to their methodology. This clearly indicates that a significant number of companies that reported having a methodology did not follow it consistently. The first two requirements for certification of software engineering processes are that a company has an established process, and that the process is followed. The implication therefore is that about half of the companies surveyed would be unable to attain the basic prerequisites for certification. Of course, it is possible that they do not care to be constrained.

When asked how many employees worked in software development, the mean response was 20, while the median was 14.5. Nearly half of the respondents indicated that their software development group had 10 employees or less. It is possible that the study results are influenced by the predominantly small size of the software development groups in the companies surveyed. Several companies surveyed suggested that they might consider certification if their organization ever became large enough to warrant it, but that they suspected that they would be absorbed by a larger, likely foreign, organization before that happened. The external validity of this study to a population of companies that had larger organizations is undetermined. For example, I would be very cautious about extending the results to another market such as the United States, since it is American corporations that frequently acquire growing Canadian companies and hence the typical size of the software development groups may well be larger in a U.S. sample than in the Canadian sample studied.

Logistic Regression

The dependent variable is the intention to certify software engineering processes to either the ISO 9000 or the SEI CMM series of standards, and is bivariate (true vs false). The independent variables are either bivariate, polytomous, or continuous. An appropriate analysis is logistic regression.

Before the regression is carried out, it is noteworthy that some companies are required to certify due to legal or customer requirements. This is queried in question 4 in the survey. It is deemed that if the answer to that question is greater than 25%, then the company really has no choice in the matter. If no other factors come into play, the requirement overrides all other considerations. Only 4% of the responses to question 4 were true, while 96% were false. This reflects the nature of the Canadian business landscape (Schellinck & Rosson, 2001); the only companies that are required to certify are those that participate in relatively small markets such as the defence or medical marketplaces, or are seeking to enter the European market (Barth, 1994; Yates, 1997). Several respondents indicated that they expected that by the time they would be large enough to participate in these marketplaces they would be acquired by larger American companies, at which time they would no longer be part of the subject pool. Because the sub-group of companies required to certify was so small, and because they could be expected to confound the analysis, they were eliminated from the data set. This results in a sample size of 96 instead of 100.

The logistic regression was carried out using the "glm" routine in the "stats" package in R, fitting a model composed of the indices Competitiveness, Risk, Complexity, Size, DirectCost, Culture, and Product. The results show a high degree of significance for Competitiveness (Pr = .00006), and a less significant result for DirectCost (Pr = .012). None of the other indices made a significant contribution to the model.

Having completed the a priori analysis, the study shifted to a more exploratory, or post-hoc, mode to refine the model. When the survey was administered, several respondents tended to answer the questions regarding risk as "agree" or "disagree" without differentiating between 'strong' and 'somewhat' levels of agreement or disagreement. For this reason, the index Risk was also expressed as a bivariate variable (RiskB) which was true if either the customer or the public were assessed to have any degree of risk. (This is shown in Table 4-4.) The logistic regression was repeated using RiskB instead of Risk. This yielded no significant change in the model.

As previously explained, the questions pertaining to functional complexity failed to yield sufficient variance to be included and the index is formed solely by the typical project team size. Therefore, the index for complexity of a company's typical software engineering project was reviewed. The original independent variable "Complexity" was based on the premise that the larger a team is, the more complex the management of the teams activities becomes. Transforming this index into a bivariate form (ComplexityB) was accomplished by saying that any company that has typical project teams that have more than 10 people in them are engaged in more complex work than companies with teams of 10 or less people. The logistic regression was repeated again with the index ComplexityB replacing Complexity. This model retained the previous significant contribution of the indices Competitiveness and DirectCost, but also showed ComplexityB as making a significant contribution to the model.

In order to obtain the most parsimonious model of antecedents, the regression was repeated once more with only the indices Competitive, ComplexityB and DirectCost included as independent variables. The model is shown in Table 4-6.

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-7.29	1.4562	-5.006	0.0000 ***
Competitiveness	0.38	0.0881	4.275	0.0000 ***
ComplexityBTRUE	1.83	0.6272	2.920	0.0035 **
DirectCost	0.53	0.2236	2.366	0.0180 *
Significance codes:	'***' Pr < 0.001, '**' Pr < 0.01, '*' Pr < 0.05			

Table 4-6: Logistic Regression Model

ComplexityB was a significant antecedent, and since it was created by splitting Complexity into those cases having more or less than an arbitrary number of project team members working on a typical project, a sensitivity analysis on the threshold was conducted. It was determined that using a threshold of 10 through 12 yields a bivariate variable that is significant at the .01 level. A threshold of 10 through 17 yields a bivariate variable that is significant at the .05 level.

Finally, the validity of treating the dependent variable as bivariate was checked by repeating the analysis using the raw survey data for the dependent variable. The original decision to treat it as a bivariate was based on the tendency of the respondents to answer in terms of a simple “yes” or “no” rather than use the Likert scale. When the linear regression was repeated with the raw data the same antecedents are significant, although the level of significance has changed. The results of this regression are shown in Table 4-7.

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.51	0.4016	-1.280	0.2037
Competitiveness	0.20	0.0302	6.731	0.0000 ***
ComplexityBTRUE	0.58	0.2615	2.199	0.0304 *
DirectCost	0.36	0.0904	3.951	0.0002 ***
Significance codes:	`***' Pr < 0.001, `**' Pr < 0.01, `*' Pr < 0.05			

Table 4-7: Linear Regression Model

Conclusions

The initial model of antecedents to certification of software engineering processes had seven hypotheses. Independent variables were formed, and measures instantiated to test each of these hypotheses. Three of the hypotheses were found to be statistically supported through a logistic regression.

The most statistically significant antecedent to certification was the competitiveness of the marketplace in which the company participated. It is assumed that in this situation the certification enhances the company's market position by leading potential customers to believe that the company can do a better job than a competitor can.

The next most statistically significant antecedent to certification is the complexity of the software project being undertaken. It must be remembered, however, that the only measure of complexity used was the typical project team size. This reflects the difficulty of managing large complex projects, and is presumed to represent a belief that certification will either reduce the complexity or facilitate the management of the complexity, leading to an increase in successful projects.

The final, and least statistically significant, antecedent is the anticipated direct costs of the certification. This is reasonable since the lower the direct cost of certification, the easier it is to build a viable cost/benefit justification for the certification.

Given the fact that only three out of seven of the hypotheses were supported by the data, it is necessary to consider if there are any alternative models that might provide a better account for the results.

One alternative to an economic model is suggested by institutional theory (Zucker, 1987). Institutional theory postulates that organizations gradually change to become increasingly similar to, and compatible with, other organizations that they interact with; this is referred to as isomorphism (Meyer & Rowan, 1977). Meyer further delineates isomorphism into two types: competitive and institutional. Of these two, institutional isomorphism is the most general, and describes how institutions relate to other organizations (DiMaggio & Powell, 1983). DiMaggio proposes that institutional isomorphism can be mediated through the three mechanisms related to external motivations, namely mimetic, normative, and coercive pressures.

In this study the companies that are required to certify by legislative or customer requirement could be seen to be driven by coercive pressures. These companies formed a very small portion of the population and were removed before analysis.

Mimetic and normative pressures are closely related in this field of inquiry. Mimetic pressures lead organizations to adopt behavior that is similar to that exhibited by other organizations with which they deal. Uncertainty is a major source of mimetic pressures. Faced with uncertainty, many organizations will opt to follow the example of successful competitors. Normative pressures come about indirectly from external agencies, often stemming from professional standards of practice. As a practice becomes a de-facto standard of practice, its adoption becomes the norm. Both of these pressures could be viewed as stemming from competitive environments, and both could be used to explain the adoption of certification by companies that are responding to the actions of early adopters. Nevertheless, it is not clear how any of these pressures could satisfactorily explain the significance of complexity or direct cost. Whereas competitiveness clearly deals with external pressures, complexity and direct cost are primarily measures of internal factors.

Institutional theory is fundamentally about what leads companies to become more like each other. It would be best applied to explain certification in a temporal dimension, exploring how certification spreads through the industry. This study did not gather temporal information, with the exception of question 5.3 which implied that the company had taken an early adopter position with respect to certification. In the factor analysis, question 5.3 fell into the group of questions that probed competitiveness; however, it did not appear to assess a unique underlying construct.

It is likely that if the study had started with a model based on institutional theory, then different hypotheses would have been formed, different questions asked, and possibly different results found. Institutional theory does not, however, appear to offer a better model to explain this study's findings than did the original economic model. Future research could look at the spread of certification using a model stemming from institutional theory.

The existing research could also be extended in future work to look at motivations for certification such as the desire to create or capture knowledge as a corporate resource. It seems clear that the process of certification should lead to formalization of software engineering processes and the development of metrics to assess those processes. These measurements could support the translation of the individual experience of the employees into a more formal corporate asset.

It would also be interesting to extend the current study to a larger population drawn from a larger economy. As noted above, the Canadian companies tended to be quite small. In particular, the possible effects of company size might be more readily observed in a sample that included larger companies.

Finally, given the difficulties encountered in identifying and recruiting participants for the current study, a better alternative to "cold calling" potential participants should be found.

References

Ahire, S. L., & Ravichandran, T. (2001). An innovation diffusion model of TQM implementation. *IEEE Transactions on Engineering Management, 48*(4), 445-464.

Anderson, S. W., Daly, J. D., & Johnson, M. F. (1999). Why firms seek ISO 9000 certification: Regulatory compliance or competitive advantage. *Production and Operations Management, 8*(1), 28-43.

Barth, C. (1994). ISO 9000 key to international success. *Orlando Business Journal, 43*, 4-6.

Casadesus, M., & Karapetrovic, S. (2005). An empirical study of the benefits and costs of ISO 9001: 2000 compared to ISO 9001//2/3: 1994. *Total Quality Management, 16*(1), 105-120.

Cronbach, L. J. (1951). Coefficient alpha and the internal structure of tests. *Psychometrika, 16*, 297-333.

Dalgaard, P. (2002). *Introductory Statistics with R.* New York, New York: Springer.

DiMaggio, P. J., & Powell, W. W. (1983). The iron cage revisited: Institutional isomorphism and collective rationality in organizational fields. *American Sociological Review, 48*(2), 147-160.

Eral, E., & Ghosh, J. B. (1997). ISO 9000 implementation in Turkish industry. *International Journal of Operations & Production Management, 17*(12), 1233-1246.

Fox, J. (2002). *An R and S-Plus Companion to Applied Regression.* Thousand Oaks, California: Sage Publications Inc.

Gotzamani, K. D., & Tsiotras, G. D. (2002). The true motives behind ISO 9000 certification: Their effect on the overall certification benefits and long term contribution towards TQM. *International Journal of Quality & Reliability Management, 19*(2/3), 151-169.

Lee, K. S., & Palmer, E. (1999). An empirical examination of ISO 9000-registered companies in New Zealand. *Total Quality Management, 10*(6), 887-899.

Meyer, J. W., & Rowan, B. (1977). Institutionalized organizations: Formal structure as myth and ceremony. *American Journal of Sociology, 82*(2), 340-363.

Naveh, E., Marcus, A., & Moon, H. K. (2004). Implementing ISO 9000: Performance improvement by first or second movers. *International Journal of Production Research, 42*(9), 1843-1863.

Nunnaly, J. (1978). *Psychometric theory.* New York: McGraw-Hill.

Oliver, C. (1991). Strategic responses to institutional processes. *Academy of Management Review, 16*(1), 145-179.

Porter, M. (1998). *Competitive strategy: Techniques for analyzing industries and competitors*. New York: Free Press.

Quazi, H. A., Khoo, Y.-K., Tan, C.-M., & Wong, P.-S. (2001). Motivation for ISO 14000 certification: Development of a predictive model. *Omega The International Journal of Management Science, 29*, 525-542.

Rao, S. S., Solis, L. E., & Raghunathan, T. S. (1999). A framework for international quality management research: Development and validation of a measurement instrument. *Total Quality Management, 10*(7), 1047-1075.

Ravichandran, T. (2000). Swiftness and intensity of administrative innovation adoption: An empirical study of TQM in information systems. *Decision Sciences, 31*(3), 691- 724.

Reifer, D. J. (2004). *CMMI in commercial use - A view from the outside*. Retrieved September 30, 2005, from http://www.psmsc.com/UG2004/Presentations/ReiferDonald_CMMI_inCommercial_Use.pdf

Schellinck, T., & Rosson, P. (2001). *Standards and exporting: Canadian companies and ISO 9000* (Dalhousie Discussion Papers in International Business No. 180). Halifax: Dalhousie University Centre for International Business Studies.

SEI. (2005a). *Process Maturity Profile CMMI v1.1 SCAMPI v1.1 Class A Appraisal Results 2004 Year End Update*. Retrieved September 30, 2005, from http://www.sei.cmu.edu/appraisal-program/profile/pdf/CMMI/2005marCMMI.pdf

SEI. (2005b). *Process Maturity Profile Software CMM 2004 Year End Update*. Retrieved September 30, 2005, from http://www.sei.cmu.edu/appraisal-program/profile/pdf/SW-CMM/2005marSwCMM.pdf

Tatikonda, M. V., & Rosenthal, S. R. (2000). Technology novelty, project complexity, and product development project execution success: A deeper look at task uncertainty in product innovation. *IEEE Transactions on Engineering Management, 47*(1), 74-87.

van der Pijl, G. J., Swinkels, G. J. P., & Verrijdt, J. G. (1997). ISO 9000 versus CMM: Standardization and certification of IS development. *Information and Management, 32*, 267-274.

Yates, J. K. (1997). Advantages and disadvantages of E/C international standards. *Project Management Journal, 28*(3), 11-22.

Zucker, L. G. (1987). Institutional theories of organization. *Annual Review of Sociology, 13*, 443-464.

Chapter 5 - Discussion

This chapter reviews the findings on antecedents and consequences of certification of software engineering processes from my quantitative research, and introduces qualitative feedback from discussions with subject matter experts.

Quantitative Research Results

The initial motivation for conducting this research was the observation that far too many software engineering projects fail, and that the failures are very costly. There is a presumption that project failures largely stem from organizational and cultural factors such as corporate politics, and that these factors could be controlled through certification of software engineering processes. In spite of this, certification of software engineering processes is seldom sought. This research investigated whether certification is likely to be worth while, and also what factors are associated with a company's intention to certify.

The only form of certification that was found to prompt a significant market response was certification to the ISO 9000 standard. The literature on announcement of ISO 9000 certification across all industries found support for the supposition that the marketplace expects at least some companies to see a significant change in their profitability due to certification. It is very interesting that the significant market response found in the present study became evident when companies were grouped into those primarily focused on provision of products or services. Furthermore, the nature of the change in profitability (positive or negative), as well as the group expected to see the change, depended on the national origin of the subject pool, with a positive response for product producing companies in America and a negative response for service producing companies in Japan.

Considering the consequences of certification, it was suggested in Chapter 3 that the difference in market response for companies producing products versus those producing services could reflect a difference between the markets' evaluations of the importance of reduced costs and increased sales, with the US market placing a higher premium on reducing costs than on increasing sales. It was suggested that the difference between the direction of the American and Japanese markets responses stems from the Japanese market's negative reaction to what could be perceived as hitherto unknown quality problems.

Considering the antecedents to certification, an examination of the factors that Canadian companies with a high likelihood to certify had in common showed three statistically significant antecedents. First, the companies that are more likely to certify software engineering processes are the companies in a more competitive marketplace. Second, companies that are engaged in developing complex software (with larger project teams) are more likely to certify. And third, companies that anticipate lower direct costs of certification are more likely to certify. In contrast to the findings in the consequences studies, the categorization of companies into those who primarily produce products and those who primarily produce services had no statistically significant relation to likelihood to certify, suggesting a disconnect between which companies are likely to pursue certification and which companies are likely to see improved profitability as a consequence of certification. This suggests that in North America service-oriented companies considering certification are either blissfully ignorant of the likely consequences of certification, or motivations besides economic ones drive their considerations. (This suggestion, of course, assumes homogeneity of North American companies, which has not been verified.)

Closing the Loop

The studies described in Chapters 3 and 4 were quantitative. They yielded some unexpected results, as outlined above. Having completed these studies, a final effort was made to contact industry experts to gauge their reactions to the findings. There was no formal methodology to this inquiry; I simply contacted three people I knew in industry who had experience in the field, and followed up on a reference to an auditor that a colleague provided. The experts included two auditors and two information systems project managers. One of the auditors was an employee of a Toronto company certified to perform ISO9000 audits, while the other was trained in providing SEI CMMI assessments and works in the Chicago area. Both project managers had worked in organizations that had undergone certifications in the Toronto and Vancouver area. I had not previously discussed my research with any of the experts. In all cases, the purpose of the dissertation research and a brief synopsis of its major findings were described, and feedback on the validity of the conclusions was invited.

Interestingly, there was a significant gap between the responses of the auditors and the project managers. All of the experts contacted were supportive of the idea of improving quality, but the two project managers were skeptical about the utility of certification to reach this goal. Further discussion led to the revelation that both project managers had been involved in certification efforts which they described as being motivated by a desire to improve the organization's marketability, rather than actually improving the quality of the processes. One of the project managers described a situation where parallel processes were established, one for the auditors and one for daily use.

The two auditors focused on the benefits that the certified processes could provide, although they acknowledged that companies often have different motivations for seeking certification. They noted that one motivation for certification was purely a marketing strategy; some companies expect to compete more effectively for business if they had been certified. Another typical motivation was thought to be short-term benefits, such as cost reduction or marketing support, and another was longer term quality improvements gained through a culture of Total Quality Management (TQM). The ISO auditor estimated that 4/5 of the firms he worked with were focused on marketing strategy objectives, which he described as "The companies want to be able to check off that they have ISO 9000 certification, but they aren't really interested in the underlying changes". Of the remaining companies he estimated that 4/5 were interested in short-term cost reductions, and only 1/5 (i.e., 4% of the total) were interested in longer term quality improvements. Both auditors emphasized that certification was not by itself sufficient to make improvements in the quality of software engineering processes; real improvements required the adoption of a culture of quality and a commitment to continuous quality improvement. Corporate culture needs to flow from the leadership, and a clear and unambiguous commitment to a culture of quality by all of the C-level management (the CEO, CFO, CIO, and so on) was identified as a common factor in companies that went on to join the 4% that effected long term quality improvements.

Combining the feedback from the project managers and the auditors we obtain a picture of a majority of certifications being focused on increasing sales through marketing. A minority were focused on reducing costs, and a very small number of companies were actually seeking to use certification for the purpose for which it is sold; i.e. achieving a long-term improvement in the quality of software engineering processes.

Practitioners, such as the project managers interviewed in this study, often recognize when certification is motivated solely by marketing. They are likely to know that no actual quality improvements will be attained, and so often develop a rather cynical attitude to the certification effort. This reaction may be counter-productive to quality improvement if morale on the project team drops, or if the increased costs of following heavily documented processes is incurred without offsetting savings through quality improvements.

Conclusions

In my research I have primarily used quantitative methods, augmented by qualitative feedback. These have been interpreted in a variety of ways. The challenge is to try to tie the different aspects of the work into a coherent framework.

The first event study looked at market response as an index of benefit from certification and indicated that the North American market believes that certification leads to profits in the case of firms focusing on the production of products, but not in the case of firms focusing on the provision of services. I suggested in Chapter 3 that this may reflect the marketplace anticipating a greater effect of reduced costs than of increased sales. This is supported by the project manager's reaction to certification efforts that are clearly oriented towards marketing goals. The Japanese event study showed the market not only discounts the value of certification to service oriented companies, but goes further and believes that it portends a decrease in future revenue streams. As I indicated previously, the Japanese market may be less concerned with the short term cost improvements of certification, than it is with the longer term quality implications.

The study of antecedents suggested that there may be other motivations for certification besides the financial ones detected in terms of market response as gauged by event studies, since the division between companies producing products and companies providing services was not related to the likelihood to certify but it was a clear determinant of expected financial consequences to certification. The study of antecedents reported on in this dissertation was based on an economic model of motivations for companies to choose to certify their software engineering processes. It is possible that additional non-economic motives exist such as a desire to create a change in corporate culture, or a desire to foster the creation and capture of knowledge of the companies' business processes.

I believe that the ISO auditor I talked to had the best insight into the problem. He suggested that North American companies that choose to certify their software engineering processes do so with three different motivations: 1. marketing motivations (approximately 80% of certifications) such as desiring the certification to influence potential customers, 2. short-term cost reduction motivations (approximately 16%) such as a desire to standardize processes to minimize costs, and 3. longer term quality motivations (approximately 4%) such as a strategic goal of total quality management (TQM). These motivations are not mutually exclusive, and may be cumulative. The certified processes may be created or implemented in such a manner that they achieve the dominant marketing goal without achieving short-term cost reduction; in other words a company can attain certification by having a documented and repeatable process without it necessarily being a good process. (One of the auditors used the example of a company producing concrete life-jackets. The processes may be certifiable, but not lead to cost reduction or profitability.) However, if a company is motivated by short-term cost reduction, then successful certification should lead to a gain in their marketing advantage by default, in addition to the desired goal of cost reduction. The company seeking cost reduction does not necessarily attain the long-term cultural orientation towards TQM, but it is assumed that a company that is motivated by TQM will gain the advantages of both improved marketability and cost reduction.

One line of future work could be based on the assessment of the extent to which a corporate culture values customer satisfaction (TQM) in addition to financial gain. This could facilitate research into the degree to which certification may foster corporate cultural transformation. It would also help to differentiate those companies that seek a shorter term financial return from their investment in certification from those who have a longer term perspective based on TQM.

Another interesting direction for future work would be the examination of how knowledge creation or knowledge assets could be measured in companies. This could lead to research investigating the correlation of certification of software engineering processes and TQM to the development and utilization of knowledge assets.

Finally, this research found statistically significant support for an unexpected, and startling, difference between Japanese and North American market responses to announcements of certification of software engineering processes. Exploration of this difference is outside the scope of the current research, and will require significant investigation of cultural and business differences between Japan and North America. This topic could form the basis for extensive ongoing research.

As the ISO 15504 standards start to be adopted, a better understanding of the potential benefits may help the industry maximize the return on their investment.

Appendices

Appendix I - Event Study Parametric Mathcad Worksheet

Parametric Analysis of Event Study Data:

Step 1: Create and Populate matrices of stock, index, and OLS model prices for the estimation period and for the event window.

(Note that the prefix EP will be used to denote "estimation period", and the prefix EW will be used to denote "event window".)

Load the Excel files that contain the closing prices and the index prices for the estimation period:

EP_Closing_Prices := Worksheet

EP_Index_Prices := Worksheet

The rows correspond to the days in the estimation period, and the columns correspond to the companies. Verify these values and assign them to appropriate variables:

$\text{EP\_Days} := \text{rows}(\text{EP\_Closing\_Prices})$ $\qquad$ $\text{Companies} := \text{cols}(\text{EP\_Closing\_Prices})$

A least squares regression was carried out separately on the closing and index prices using SPSS. Load the resulting OLS Market Model parameters from an Excel file:

OLS_Param := Worksheet

Use the model parameters and the index prices to calculate the predicted prices in the estimation period. (This will be used to develop the abnormal returns in the estimation period in order to develop an estimated std. deviation.)

$\text{company} := 1 .. \text{Companies}$

$\text{day} := 1 .. \text{EP\_Days}$

$$\text{EP\_Model}_{\text{day},\text{company}} := \text{OLS\_Param}_{\text{company},1} + \left(\text{OLS\_Param}_{\text{company},2}\right)\cdot\left(\text{EP\_Index\_Prices}_{\text{day},\text{company}}\right)$$

Load the Excel files that contain the closing prices and the index prices for the event period:

EW_Closing_Prices := Worksheet

EW_Index_Prices := Worksheet

$\text{EW\_Days} := \text{rows}(\text{EW\_Closing\_Prices})$

Calculate the model closing prices:

$\text{company} := 1 .. \text{Companies}$

$\text{day} := 1 .. \text{EW\_Days}$

$$\text{EW\_Model}_{\text{day},\text{company}} := \text{OLS\_Param}_{\text{company},1} + \left(\text{OLS\_Param}_{\text{company},2}\right)\cdot\left(\text{EW\_Index\_Prices}_{\text{day},\text{company}}\right)$$

Step 2: Calculate the "returns", (daily percent change), for the actual stock prices, and for the OLS model in the Estimation Period and in the Event Window.

Calculate the returns for the actual closing and the index prices during the estimation period:

$$\text{day} := 2..\,\text{EP\_Days}$$

$$\text{EP\_Actual\_Return}_{\text{day}-1,\text{company}} := \frac{\left(\text{EP\_Closing\_Prices}_{\text{day},\text{company}} - \text{EP\_Closing\_Prices}_{\text{day}-1,\text{company}}\right)}{\text{EP\_Closing\_Prices}_{\text{day}-1,\text{company}}}$$

$$\text{EP\_Return\_Days} := \text{rows}(\text{EP\_Actual\_Return})$$

$$\text{EP\_Model\_Return}_{\text{day}-1,\text{company}} := \frac{\left(\text{EP\_Model}_{\text{day},\text{company}} - \text{EP\_Model}_{\text{day}-1,\text{company}}\right)}{\text{EP\_Model}_{\text{day}-1,\text{company}}}$$

Calculate the returns for the actual closing and the index prices during the event window:

$$\text{day} := 2..\,\text{EW\_Days}$$

$$\text{EW\_Actual\_Return}_{\text{day}-1,\text{company}} := \frac{\left(\text{EW\_Closing\_Prices}_{\text{day},\text{company}} - \text{EW\_Closing\_Prices}_{\text{day}-1,\text{company}}\right)}{\text{EW\_Closing\_Prices}_{\text{day}-1,\text{company}}}$$

$$\text{EW\_Return\_Days} := \text{rows}(\text{EW\_Actual\_Return})$$

$$\text{EW\_Model\_Return}_{\text{day}-1,\text{company}} := \frac{\left(\text{EW\_Model}_{\text{day},\text{company}} - \text{EW\_Model}_{\text{day}-1,\text{company}}\right)}{\text{EW\_Model}_{\text{day}-1,\text{company}}}$$

Step 3: Calculate the estimated standard deviation from the estimation period.

Calculate the Abnormal Returns in the estimation period:

$$\text{EP\_AR} := \text{EP\_Actual\_Return} - \text{EP\_Model\_Return}$$

$$\text{day} := 1..\,\text{EP\_Return\_Days}$$

Calculate the Average Abnormal Returns in the estimation period (averaged across companies):

$$\text{EP\_AAR}_{\text{day}} := \text{mean}\left[\left(\text{EP\_AR}^{\text{T}}\right)^{\langle\text{day}\rangle}\right]$$

(Transpose the matrix so we can calculate the mean for each column)

Calculate the Mean Average Abnormal Return for the estimation period (averaged across days):

$$\text{EP\_MAAR} := \text{mean}(\text{EP\_AAR})$$

Estimate the Std. Dev.:

$$\text{EP\_s\_AAR} := \sqrt{\frac{\sum(\text{EP\_AAR} - \text{EP\_MAAR})^2}{\text{EP\_Return\_Days} - 1}}$$

Step 4: Calculate the cumulative average abnormal return for the event window, and test its significance.

$day := 1 .. EW\_Return\_Days$

Calculate the Abnormal Returns in the event window:

$$EW\_AR := EW\_Actual\_Return - EW\_Model\_Return$$

Calculate the Average Abnormal Returns in the event period:

$$EW\_AAR_{day} := mean\left[\left(EW\_AR^{T}\right)^{\langle day \rangle}\right]$$

(Transpose the matrix so we can calculate the mean for each column)

Calculate the Cumulative Average Abnormal Return for the event period:

$$EW\_CAAR := \sum EW\_AAR$$

The statistical t-test is:

$$t := \frac{EW\_CAAR}{EP\_s\_AAR \cdot \sqrt{EW\_Return\_Days}}$$

Repeat the calculation for all possible windows in the estimation window period:

$day1 := 1 .. EW\_Return\_Days$

$day2 := 1 .. EW\_Return\_Days$

$$CARR_{day1,day2} := \left(\sum_{i = day1}^{day2} EW\_AAR_{i}\right)$$

$$t_{day1,day2} := \frac{CARR_{day1,day2}}{EP\_s\_AAR \cdot \sqrt{|(day2 - day1)| + 1}}$$

Appendix II - Event Study Non-Parametric Mathcad Worksheet

Non-Parametric Analysis of Event Study Data:

(Continued from Parametric Analysis Worksheet)

Step 5: Perform Corrado's Test

AR := stack(EP_AR, EW_AR)

Tot_days := rows(AR)   Companies := cols(AR)

```
Corrado(AR, w) :=
  Tot_days ← rows(AR)
  Companies ← cols(AR)
  day ← 1
  rank ← 2
  ab_ret ← 3
  for company ∈ 1 .. Companies
    for index ∈ 1 .. Tot_days
      $Tmp_{index,day}$ ← index
      $Tmp_{index,ab\_ret}$ ← $AR_{index,company}$
    Tmp ← csort(Tmp, ab_ret)
    for index ∈ 1 .. Tot_days
      $Tmp_{index,rank}$ ← index
    Tmp ← csort(Tmp, day)
    for index ∈ 1 .. Tot_days
      $R_{index,company}$ ← $Tmp_{index,rank}$
  S ← $\sqrt{\left(\frac{1}{Tot\_days}\right)\cdot\sum_{i=1}^{Tot\_days}\left[\frac{1}{Companies}\left[\sum_{j=1}^{Companies}\left[R_{i,j}-\left(\frac{Tot\_days}{2}\right)-0.5\right]\right]\right]^{2}}$
  for d ∈ w .. Tot_days
    $t_{(d-w+1)}$ ← $\left(\frac{1}{Companies}\right)\cdot\sum_{j=1}^{Companies}\frac{\left[R_{d,j}-\left(\frac{Tot\_days}{2}\right)-0.5\right]}{S}$
  return t
```

t := Corrado(AR, EP_Days)

Appendix III - Antecedents Survey

Survey on Certification of Software Engineering Processes

The scope of the business unit being certified may vary from company to company. In some companies the entire business's software engineering process may be certified, while in other companies only the software engineering processes within a particular business unit may be certified. Please choose what you feel is the scope of business unit that is most likely to be certified in your company, and complete all questions in the context of that business unit.

1. Has your business obtained ISO 9000 certification of its software engineering processes?
   □ Yes □ No If Yes, then what year was the certification attained? _______

1.1. If your company has not obtained ISO 9000 certification of its software engineering processes, then do you think it is likely in the future?

Very Unlikely	Somewhat Unlikely	Neither Likely Nor Unlikely	Somewhat Likely	Very Likely

If likely, then in approximately how many years? ____

2. Has your business obtained Software Engineering Institute's Capability Maturity Model (SEI-CMM) certification of its software engineering processes?
   □ Yes □ No If Yes, then what year was the certification attained? _______

2.1. If your company has not obtained SEI-CMM certification of its software engineering processes, then do you think it is likely in the future?

Very Unlikely	Somewhat Unlikely	Neither Likely Nor Unlikely	Somewhat Likely	Very Likely

If likely, then in approximately how many years? ____

3. How much of your software development work is oriented towards sale of products or services, (as opposed to developing software for internal use)?

All Sales	More Sales Than Internal	Half and Half	More Internal Than Sales	All Internal

4. What portion of your business is required to certify its software engineering processes due to regulatory or customer requirements?

None	¼	½	¾	All

5. Please indicate the extent to which you agree with the following statements

5.1. "Our customers have a broad choice of products that they could choose instead of ours."	Strongly Disagree	Somewhat Disagree	Neither Agree Nor Disagree	Somewhat Agree	Strongly Agree
5.2. "Our software/services sales volume is extremely sensitive to our product price."	Strongly Disagree	Somewhat Disagree	Neither Agree Nor Disagree	Somewhat Agree	Strongly Agree
5.3. "Certification has/would differentiate us from our competitors, so our customers will be willing to pay more for our products."	Strongly Disagree	Somewhat Disagree	Neither Agree Nor Disagree	Somewhat Agree	Strongly Agree
5.4. "Our customers are likely to associate certification with a higher quality of product."	Strongly Disagree	Somewhat Disagree	Neither Agree Nor Disagree	Somewhat Agree	Strongly Agree
5.5. "Our competitors are likely to certify their software engineering processes, so we should do the same."	Strongly Disagree	Somewhat Disagree	Neither Agree Nor Disagree	Somewhat Agree	Strongly Agree
5.6. "If a typical project of ours fails, then we stand to suffer a significant financial loss"	Strongly Disagree	Somewhat Disagree	Neither Agree Nor Disagree	Somewhat Agree	Strongly Agree
5.7. "If software that we write fails to operate correctly, then our customer may be significantly injured."	Strongly Disagree	Somewhat Disagree	Neither Agree Nor Disagree	Somewhat Agree	Strongly Agree
5.8. "If software that we write fails to operate correctly, then the public may be significantly injured."	Strongly Disagree	Somewhat Disagree	Neither Agree Nor Disagree	Somewhat Agree	Strongly Agree
5.9. "Certification of our software engineering processes will lead to better quality products."	Strongly Disagree	Somewhat Disagree	Neither Agree Nor Disagree	Somewhat Agree	Strongly Agree
5.10. "I am very familiar with the requirements for certification of our software engineering processes."	Strongly Disagree	Somewhat Disagree	Neither Agree Nor Disagree	Somewhat Agree	Strongly Agree
5.11. "Very few of our software engineering processes would need to be changed if we wanted to certify them."	Strongly Disagree	Somewhat Disagree	Neither Agree Nor Disagree	Somewhat Agree	Strongly Agree
5.12. "Quality is important, even if increasing quality will reduce profits in the short term."	Strongly Disagree	Somewhat Disagree	Neither Agree Nor Disagree	Somewhat Agree	Strongly Agree
5.13. "Quality costs money. We should only build in as much quality as our customers are prepared to pay for."	Strongly Disagree	Somewhat Disagree	Neither Agree Nor Disagree	Somewhat Agree	Strongly Agree
5.14. "Our company's corporate culture strongly emphasizes quality."	Strongly Disagree	Somewhat Disagree	Neither Agree Nor Disagree	Somewhat Agree	Strongly Agree

6. Please indicate your answer to the following questions.

6.1. How many people work on a typical software development project in your company? (Include IT staff, user representatives, contractors and vendors.) ____

6.2. How would you characterize the number of functions performed by a typical piece of software your company develops?

Single Function — A Few Functions — — A Lot of Functions — Extremely Many Functions

6.3. How often does your software development make use of reusable code fragments such as functions, procedures, or objects?

Never — Occasionally — — Often — Very Often

6.4. Approximately how many people in your company work in the software development area that would be a candidate for certification? ________

6.5. What is the approximate annual budget for software development in the business area that would be a candidate for certification? ________

6.6. Does your company have a software development methodology? □ Yes □ No

6.7. What proportion of your software projects follow a software development methodology?

None — ¼ — ½ — ¾ — All

6.8. How many other areas of your company are you aware of that have chosen to certify their business processes? (For example ISO 9000) __________

7. Please list the technologies that are most commonly used in your software development projects. (For example: object orientation, fourth generation tools such as code generators, third generation languages such as C++, client server architectures, etc.)

____

____

____

If you would like to receive a copy of the results of this study please send an e-mail to fullerta@interchange.ubc.ca with a subject line of "Certification antecedents study results request". In the body of your message please indicate if you would like to receive the results by e-mail or in hardcopy, and provide the appropriate address. (In order to maintain your anonymity please do not write your name or address on this survey form.)

Thank you for participating in this study.

Appendix IV - Antecedents Consent Form