Proactive Risk Management Methods for Software Engineering Excellence

by
Elaine Marie Hall

Bachelor of Science, Management Science, Florida Institute of Technology 1979
Master of Business Administration, Florida Institute of Technology 1983
Master of Science, Computer Science, Florida Institute of Technology 1983

A dissertation submitted to
the Graduate School of Florida Institute of Technology
in partial fulfillment of the requirements for
the degree of

Doctor of Philosophy in Computer Science

Melbourne, Florida
April, 1995

We the undersigned committee hereby approve the attached dissertation

Proactive Risk Management Methods for Software Engineering Excellence

by

Elaine Marie Hall

Charles B. Engle, Jr., Ph.D., Committee Chair
Associate Professor and Chair, Computer Science

Frederick B. Buoni, Ph.D.
Professor, Operations Research and Computer Science

J. Richard Newman, Ph.D.
Associate Professor, Computer Science

Walter P. Bond, Jr., Ph.D.
Adjunct Associate Professor, Computer Science

Wade H. Shaw, Ph.D., P.E.
Professor, Technology Management

T. J. Sanders, Ph.D.
Harris Professor and Director,
Division of Electrical and Computer Science and Engineering

Abstract

Title:	**Proactive Risk Management Methods for Software Engineering Excellence**
Author:	**Elaine Marie Hall, MBA, MSCS.**
Major Advisor:	**Charles B. Engle, Jr., Ph.D.**

Risk management applied to software intensive programs is an emerging technology of strategic importance. Risk management is informed decision-making under uncertainty that deals with future consequences of present decisions. This dissertation establishes the role of risk management in software engineering. Barriers to adopting risk management technology are predicted through use of a general two-dimensional framework based on theories of organizational and community-wide technology transfer. Critical success factors for transitioning risk management technology into an organization are identified. A maturity model for risk management practices is developed to understand the essential elements of risk management, and how these elements evolve to increasing levels of effectiveness and efficiency. Proactive risk management methods are tested and evaluated to provide a recommended approach to effectively using risk management methods for increased software engineering quality.

Table of Contents

CHAPTER 2

Risk Management in Software Engineering 11

CHAPTER 3

Risk Management Technology Transfer..... 47

CHAPTER 4

Risk Management Capability 68

CHAPTER 5

Proactive Risk Management 104

CHAPTER 6

List of Keywords

Risk Management

Software Engineering

Maturity Model

Technology Transfer

Process Improvement

List of Figures

List of Tables

List of Charts

List of Abbreviations

ACM	Association for Computing Machinery
A&D	Aerospace and Defense
AFB	Air Force Base
AFLC	Air Force Logistics Command
AFMC	Air Force Material Command
AFSC	Air Force Systems Command
AI	Artificial Intelligence
ANSI	American National Standards Institute
ASQC	American Society for Quality Control
CASE	Computer-Aided Software Engineering
CBA	Capability Maturity Model Based Appraisal
CM	Configuration Management
CMM	Capability Maturity Model
CMU	Carnegie Mellon University
COTS	Commercially Off The Shelf
CSEE	Conference on Software Engineering Education
DID	Data Item Description
DoD	Department of Defense
DOI	Diffusion of Innovations
DPMA	Data Processing Management Association
DSMC	Defense Systems Management College
EFDPMA	Education Foundation of the Data Processing Management Association
FA	Focus Area
FAA	Federal Aviation Administration
FBI	Federal Bureau of Investigation
FFRD	Federally Funded Research and Development
FFRDC	Federally Funded Research and Development Center
GAO	General Accounting Office

GUI	Graphical User Interface
IDA	Institute for Defense Analysis
IEEE	Institute of Electrical and Electronics Engineers
IMP	Implementation
INF	Infrastructure
ISO	International Standards Organization
KP	Key Practice
KPA	Key Process Area
MIL	Military
MQ	Maturity Questionnaire
NASA	National Aeronautics and Space Administration
NATO	National Alliance Treatise Organization
NCOSE	National Council on Systems Engineering
OOP	Object-Oriented Programming
PM	Program Manager
PMO	Program Management Office
PRO	Process
QA	Quality Assurance
QPI	Quantitative Process Improvement
RE	Risk Exposure
RM-CBA	Risk Management Capability Maturity Model Based Appraisal
RM-CMM	Risk Management Capability Maturity Model
RMEF	Risk Management Evolution Framework
RMP	Risk Management Plan
RMS	Risk Management Survey
RMT	Risk Management Technology
ROI	Return on Investment
RRL	Risk Reduction Leverage
SCE	Software Capability Evaluation
SDCCR	Software Development Capability/Capacity Review

SDCE	Software Development Capability Evaluation
SDIP	Software Development Integrity Program
SDP	Software Development Plan
SE	Software Engineering
SEE	Software Engineering Environment
SEI	Software Engineering Institute
SEMA	Software Engineering Management Associates
SEN	Software Engineering Notes
SEPG	Software Engineering Process Group
SIGSOFT	Special Interest Group on Software Engineering
SPA	Software Process Assessment
SPC	Software Productivity Consortium
SPI	Software Process Improvement
SPICE	Software Process Improvement Capability Evaluation
SPIN	Software Process Improvement Network
SPMN	Software Program Managers Network
SPMP	Software Project Management Plan
SPT	Software Process Team
STC	Software Technology Conference
STD	Standard
TBQ	Taxonomy Based Questionnaire
TCA	Technical Collaboration Agreement
TPM	Technical Performance Measurement
TQM	Total Quality Management
TQSE	Total Quality Software Engineering
TR	Technical Report
TRM	Team Risk Management
USA	United States of America
WBS	Work Breakdown Structure

Acknowledgment

I would like to acknowledge the pioneers in the field of software risk management for their foundational material. In order to add to the body of knowledge in this area, I stood on the shoulders of Dr. Barry Boehm, Dr. Robert Charette, and the Software Engineering Institute. Their ideas have inspired me and made this dissertation possible.

I also want to acknowledge my committee chair, Dr. Chuck Engle, and committee members Dr. Pat Bond, Dr. Fred Buoni, Dr. Dick Newman, and Dr. Wade Shaw for their guidance. Dr. Bond extended the rigor of my data analysis to ensure statistical correctness. Dr. Buoni's knowledge of theoretical decision analysis taught me to quantify risk and solve hard problems. Dr. Engle's enthusiasm for the field of software engineering inspired me to make a significant contribution by writing and presenting from a software engineering perspective. Dr. Newman's encouragement through six years of postgraduate study kept me on the "fast track." Dr. Shaw helped me in the survey design and analysis to provide meaningful results.

Dedication

To my loving husband, Thomas Eric Gorsuch.

CHAPTER 1

Introduction

A risk is a potential problem usually caused by lack of information, control, or time. The majority of potential problems on software intensive projects can be managed proactively to reduce rework and other obstacles to successful software delivery. Proactive risk management is the opposite of reactive crisis management. Proactive risk management is taking the action required to identify, assess, and manage risks to prevent problems on software projects. Proactive risk management helps programs succeed by providing them with tools for more informed decision-making and improved communication.

Risk management applied to software intensive projects is an emerging technology of strategic importance. Risk management is not the next silver bullet candidate of software engineering, but it is a powerful weapon against slim profit margins, strong competition, and increasing system complexity. Risk management is informed decision-making under uncertainty that deals with future consequences of present decisions. Risk management focuses on critical success factors and provides methods to apply scarce resources more effectively. Risk management prepares us to adapt to changing circumstances by providing a disciplined approach to comparing

alternatives and managing the uncertainty of the future. Since risk increases as system complexity increases, there is a growing need to develop risk management technology in organizations that develop, maintain, and use computing systems.

The purpose of this dissertation is to develop a risk management maturity model and a proactive approach to risk management to support the need and trend toward the use of risk management in software engineering. Chapter 2, *Risk Management in Software Engineering*, describes how risk management is applied in software engineering, and establishes the current state of the practice. Chapter 3, *Risk Management Technology Transfer*, determines the barriers that must be overcome for adoption of risk management technology in an organization, and in the software engineering community. Chapter 4, *Risk Management Capability*, provides an understanding of essential elements of risk management as applied to software engineering, and how these elements evolve to increasing levels of effectiveness and efficiency. Chapter 5, *A Proactive Approach*, details the use of proactive risk management methods on software projects within one organization. Chapter 6, *Conclusion*, summarizes the results and lessons learned in improving and extending the existing body of knowledge in risk management applied to software engineering.

1.1 Problem

Software systems are developed in an environment full of uncertainty and the software community does not currently have the tools to cope with the uncertainty. The final outcome of a software project is the result of all decisions made on the project over time. The fact that we manage and develop software systems under uncertainty creates a problem of how to improve the quality of our decision-making to control budget, schedule, and technical requirements. The many causes of uncertainty, such as lack of information, advances in technology, and system complexity will always exist.

Who is responsible for managing risks on a software project? Software managers are responsible for budget and schedule, as that is their primary focus and perspective. Software engineers are responsible for the technical software products, and that is their primary focus and perspective. If the schedule slips, it slips one day at a time [Brooks75]. Software engineers have notoriously been the ones who slip schedule, often doubling the original software estimate. Because programmatic and technical risks exist on all software projects, I believe that both managers and engineers should be responsible for risk at their respective levels. Software engineers and software managers must make decisions under uncertainty using proactive risk management techniques for more informed decision-making.

1.2 Hypothesis

My hypothesis is that the Software Engineering Institute (SEI) Capability Maturity Model (CMM), the current model of the software development process, is incomplete regarding management of risks in software development, and that an evolutionary risk management model can be developed for software engineering. The CMM has helped in the area of process risk, but that is only a fraction of the risks that a software project will face. The basis of my assertion is the Software Risk Taxonomy, developed and field-tested by the SEI Risk Program [Carr93]. The taxonomy is organized into three major classes: *Product Engineering*, *Development Environment*, and *Program Constraints*. A relatively small percentage of the taxonomy concerns software process risks. The associated Taxonomy Based Questionnaire (TBQ), is a disciplined and systematic method for identifying risk on a software intensive project.

The Capability Maturity Model for Software (CMM) is a de facto model of the software development process. It was developed by the SEI at Carnegie Mellon University (CMU), under the leadership of Watts Humphrey [Humphrey87a]. The inspiration for the CMM's five levels of software process maturity was Phil Crosby's Quality Management Maturity Grid, a five stage evolutionary framework for adopting quality practices [Crosby80]. Today, the CMM is used by the software community as a software process maturity framework to evolve toward software engineering and management excellence. It is used by the United States government as the basis for an evaluation method for selecting contractors who will develop software with the lowest

overall risk [SCE94]. The foundation of the CMM is continuous process improvement for productivity and quality gains. The CMM is not a silver bullet and does not address all the issues important for successful projects [Paulk93c].

A basic assumption of the CMM is that by increasing maturity levels, quality increases and risk decreases [Humphrey89]. By contradiction, the CMM cannot completely manage risks on software projects, since it only considers process risks. The CMM does not address software technology or human resource issues, which are known sources of risk on a software project [Paulk93a]. The current CMM addresses aspects of risk management as Key Practices of several Key Process Areas. This is neither a complete maturity model for risk management, nor does it provide the necessary focus for risk management. Under the existing CMM architecture, a focus for risk management would require a Key Process Area (KPA) to be defined at a single maturity level. Working groups at the SEI have discussed the pros and cons of a Risk Management KPA at levels 2, 3, and 4. The following quote is from the CMM v2.0 Risk Management working group meeting in February, 1995:

> The CMM itself is a risk management plan. Its stated purpose is to reduce the risk to a program that a project fails to build the correct software, on schedule and budget. It is highly appropriate that Risk Management be incorporated as a fundamental aspect of good overall software management. Level 2 Organizations would stand to benefit greatly from basic risk management practices. The group did acknowledge that Risk Management at Level 2 has potentially the greatest impact to the community. The Risk Management working group concluded that Risk Management needed the additional emphasis that only a KPA would provide.

In my opinion, forcing risk management to live at a single maturity level would reduce the full impact of the technology and stunt its growth. I assert that risk management has an evolutionary nature, just as software process and quality maturity are evolutionary in nature. This implies that risk management practices improve in efficiency and effectiveness as they mature. For this reason, an evolutionary risk management maturity model for software engineering is the preferred model.

What lies beyond risk management? On the cutting edge of my dissertation hypothesis is an innovative mindset that may extend from the highest levels of my risk management maturity model. I use the phrase *Possibility Thinking* to describe the paradigm shift to positive belief under uncertainty. This innovative way of thinking may not result from continuous process improvement of the risk management maturity model, but from the use of a radically creative approach.

1.3 Approach

This doctoral dissertation presents significant original research to determine risk management practices that can be applied to software engineering to produce quality results. I developed the *Risk Management - Capability Maturity Model* (RM-CMM) as an elaboration of my *Risk Management Evolution Framework* (RMEF). To establish the role of risk management in software engineering, and to support the need and trend toward the use of risk management, I developed a proactive approach to risk management in the following phases:

1. *Research existing risk management technology.*

I defined risk management for software engineering and determined the methods and tools that currently exist to support risk management on software programs. The purpose of establishing the role of risk management in software engineering through advanced study is to provide a clear understanding of the origins of risk management.

2. *Determine barriers to adopting risk management technology.*

I determined the barriers that must be overcome for adoption of risk management technology in an organization and in the software engineering community. I also identified the critical success factors for transitioning risk management technology.

3. *Define a maturity model for risk management.*

My goal was to understand essential elements of risk management as applied to software engineering, and how these elements evolve to increasing levels of effectiveness and efficiency. I defined a maturity model for risk management that provides an improvement framework for practical application of risk management.

4. *Develop proactive risk management methods.*

I developed proactive risk management methods that would yield software engineering quality results. I tested and evaluated these methods on software projects to provide real-world results. The methods were improved based on this pilot study.

In my efforts to champion risk management as a viable alternative to the "Software Crisis," I have written articles, papers, trained, and facilitated risk management. The following is a chronology of activities that I have performed which provides the knowledge and experience required for this dissertation (see Figure 1).

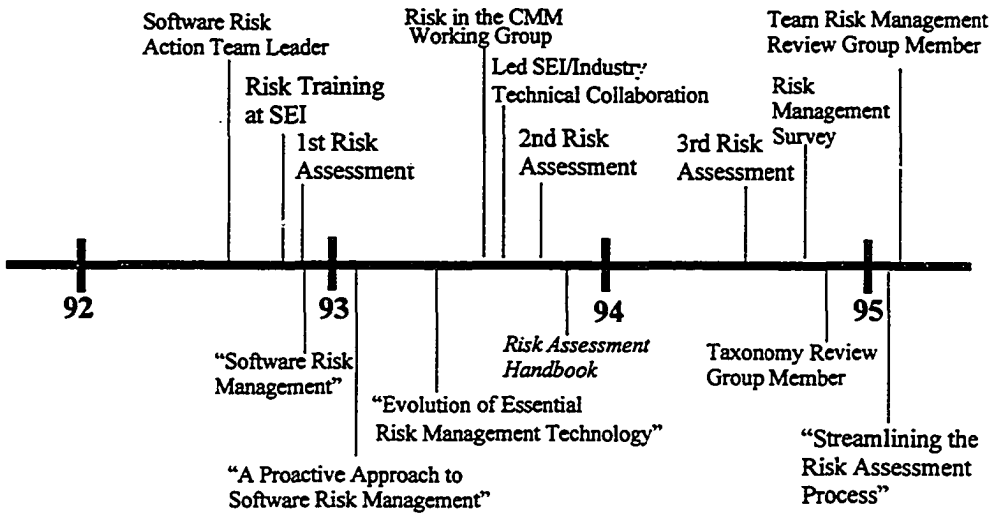


Figure 1. Chronology of a Risk Champion

1.3.1 Vision

My vision is to make a significant contribution to the discipline of software engineering by establishing foundational concepts of software engineering excellence. These foundational concepts are the application of risk management principles to increase software quality and customer satisfaction.

1.3.2 Mission

My mission is to determine the risk management practices that define the maturity levels for risk management. I developed proactive risk management methods to increase the communication of software risks between producers and consumers of software systems and to provide the means to mitigate identified risks. To ensure these methods would be acceptable to the software community, the barriers to adoption and critical success factors of risk management technology transfer were determined. I developed and tested the *Risk Management Evolutionary Framework* (RMEF) and *Risk Management Survey* (RMS) on three software projects to evaluate the methods and incorporate improvements.

1.3.3 Research Objective

The objective of this research is to extend the foundations of software engineering by developing an evolutionary risk management maturity model. The power of a maturity model lies in the ability to create a vision and a common frame of reference that enables comparisons [Koltun92]. The major distinguishing feature of the spiral model is that it creates a risk-driven approach to the software process. The spiral model places a great deal of reliance on the ability of software developers to identify and manage sources of project risk [Boehm88a]. Because risk management is requisite for

the spiral model, developing a maturity model for risk management will contribute to the foundations of software engineering.

The *Risk Management Evolution Framework* (RMEF), and *Risk Management Capability Maturity Model* (RM-CMM) have been developed through this research to illustrate risk management technology as multidimensional and evolutionary. It is hoped that the model will be used to understand risk management technology, and to establish a realistic plan for successfully implementing risk management on software projects within an organization. A proactive approach to risk management has also been defined which includes the *Risk Management Capability Maturity Model Based Appraisal* (RM-CBA), a method for quantitatively assessing an organization's risk management capability using the *Risk Management Survey* (RMS).

CHAPTER 2

Risk Management in Software Engineering

In *The Mythical Man-Month,* Fred Brooks described the tar pit of software engineering as user's perceptions changing through development, advancing technology, and a dependence upon others for resources [Brooks75]. Thirty years later,[1] these conditions remain to be dealt with. There is an increasing dependence on software systems. Due to the nature of the problems being addressed by computers today, software is increasing in complexity. Unlike hardware, software doesn't have the foundational building block components that would cause an order of magnitude decrease in cost in the future. Are software engineers doomed to the tar pits because these conditions are all fundamental truths of software engineering?

We can discuss the above situation in risk management terms: *change, uncertainty* and a *lack of control.* The traditional *risk mitigation strategy* on a late project was to add manpower, and this action plan made the project later. Brooks listed alternative *action plans:* trim the task, reschedule, remove people, increase communication, pilot to refine ideas [Brooks75]. Can disciplined risk management practices be applied on

---

1. Brooks wrote about his experiences at IBM in the 1960's.

software projects to raise the standards of software engineering? To determine the answer to this question, the role of risk management in software engineering must be established. The following objectives have been established to accomplish this goal:

- Define software engineering excellence.
- Define software risk management.
- Determine what drives the use of risk management on software programs.
- Establish how the management of risk will improve software quality.
- Establish the role of risk management in software engineering.

2.1 Software Engineering Excellence

What is software engineering excellence? Is there a connection between excellence in software engineering and the use of risk management techniques? Software engineering excellence is a set of best practices that exist in an organization that develops high-quality software. Fundamental to the notion of software engineering excellence is continuous improvement. Software engineering as a discipline and field of study must continually raise the standards that define its best practices.

2.1.1 Software Engineering

Software engineering is defined as the establishment and use of sound engineering principles in order to obtain economically software that is reliable and works efficiently

on real machines [Naur69]. Software engineering encompasses methods, tools, and procedures that enable the manager to control the process and provide the practitioner with a foundation for building high-quality software in a productive manner [Pressman92]. To define software engineering as a field of study, it is important to review the evolution of software development. When and why were risk management techniques introduced to software development? Do we need software risk management in the future?

2.1.1.1 Past

In the early days, programming was viewed as an "art form" [Pressman92]. Solutions for computing systems focused on hardware cost and performance issues. Software was usually custom made and constrained by the hardware.

Software practitioners were challenged to make the transition from programmers who write programs in an ad hoc fashion to software engineers who develop quality software in a disciplined way (see Figure 2).

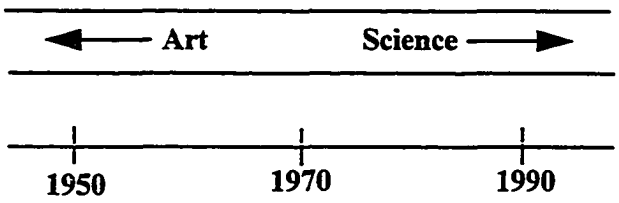


Figure 2. Forty Years of Software Development

2.1.1.2 Present

A software industry valued at billions of dollars per year has evolved from those early days. Software development in 1995 is characterized by standards organizations, engineering environments, maturing languages, and tools (see Figure 3).

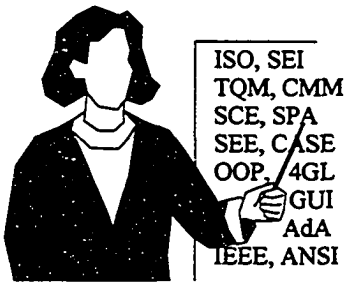


Figure 3. Software Development Today

Today, there is an industry movement to mature the software development process. The Software Engineering Institute (SEI) Capability Maturity Model (CMM) for Software is used to focus on the process aspects of a Total Quality Management (TQM) effort. To improve the quality of software processes, CMM user groups, called Software Process Improvement Networks (SPINs), meet to discuss improvement efforts and results. The CMM is also used by a Software Engineering Process Group (SEPG) to establish the standard software process for an organization. These standard processes are tailored by software engineers for implementation on a specific program. Government contracts use the Software Capability Evaluation (SCE) method to assess

a contractor's ability to develop software [SCE94]. The contractors use a Software Process Assessment (SPA) to determine their own strengths and weaknesses. Both SCE and SPA methods are based on the CMM. Another evaluation method for the software process is the Software Development Capability/Capacity Review (SDCCR) [AFMC93]. All three evaluation methods use a Maturity Questionnaire (MQ) that is based on the maturity model.

Standard development environments, known as a Software Engineering Environment (SEE) and integrated tools such as Computer-Aided Software Engineering (CASE), are used to increase productivity. Programming languages such as Ada promise increased productivity through standardization and reuse. Software methodologies based on data abstraction such as Object-Oriented Programming (OOP) are emerging to support component-based software development.

Growing pains of the software industry continue as demand for software and application complexity increase. The industry's commitment to mature and move away from the "software crisis" is evident. Current state of the practice in software engineering can best be determined from standards organizations, literature, conferences and research.

Standards Organizations.

Software engineering standards have been defined by the Institute of Electrical and Electronics Engineers (IEEE), the Department of Defense (DoD), and the Software Engineering Institute (SEI). International Standards Organization (ISO) 9000-3 has defined international standards for quality management and quality assurance of software. Data Processing Management Association (DPMA) has also written standards for software. Government standards for software engineering have been established by the National Aeronautics and Space Administration (NASA) and the Federal Aviation Administration (FAA).

Literature.

Publications, such as IEEE Software and IEEE Transactions on Software Engineering cover a broad range of topics monthly. Entire issues have been dedicated to software process and object-oriented technologies. The Association for Computing Machinery (ACM) publishes Software Engineering Notes (SEN) for their Special Interest Group on Software Engineering (SIGSOFT).

Conferences.

Every year the IEEE and the Association for Computing Machinery (ACM) sponsor an International Conference on Software Engineering. The ACM SIGSOFT also sponsors an annual Symposium on the Foundations of Software Engineering. Academic issues are discussed at the annual Conference on Software Engineering Education (CSEE). The SEI hosts a Software Engineering Symposium, as well as a

Software Engineering Process Group (SEPG) conference annually. The DoD sponsors a Software Technology Conference annually. Software Engineering is a major track at the annual Pacific Northwest Software Quality Conference.

Research.

Research in Software Engineering is ongoing in government, industry and academic institutions. For example, the industry sponsored Software Productivity Consortium (SPC) conducts research in software engineering. The SEI is a Federally Funded Research and Development (FFRD) organization.

2.1.1.3 Future

Future directions of the software industry are focused on increasing the productivity of software engineers and the quality of software products. These trends include the following:

1. Use of methods to increase productivity
    - Component based code
        - Reuse repositories
        - Commercially Off The Shelf (COTS) software
    - Automated environments
        - Software Engineering Environment (SEE)
        - Computer-Aided Software Engineering (CASE)

2. Means to increase quality
    - Increased professionalism
        - Official standards and ethics
        - Professional certification for software engineers
    - Changes to computer science education curricula
        - Establish software engineering curricula
        - System engineering concepts
        - Business and management concepts.

2.1.2 Total Quality Software Engineering

Experts differ on precisely what quality means. The most widely accepted criteria of Total Quality Management (TQM) is in the Malcolm Baldrige National Quality Award (see Table 1).

Table 1. A National Standard for Total Quality Management

Baldrige Award Criteria	Points
Customer Focus and Satisfaction	300
Quality and Operational Results	180
Human Resource Development	150
Management of Process Quality	140
Leadership	95
Information and Analysis	75
Strategic Quality Planning	60

The application of TQM concepts to software engineering may be described as Total Quality Software Engineering (TQSE). TQSE could be measured by the collective set of standards, benchmarks, and best practices of software engineering. Problems with software quality are well known: it is typically not delivered on time, the cost is significantly greater than predicted, and the user requirements are often not met. If any of these problems can be reduced or prevented, software engineering quality will increase. Sections 2.1.2.1 and 2.1.2.2 identify how risk management fits into existing software engineering standards and benchmarking.

2.1.2.1 Standards

There are various organizations that are active in defining the standards of excellence in software engineering. DoD, SEI, ISO, ANSI/ASQC and IEEE have defined standards for software engineering. What standards have already been established by these organizations for risk management?

Department of Defense (DoD).

The Department of Defense (DoD) has recently approved MIL-STD-498, a new standard for Software Development and Documentation that replaces document-driven DoD-STD-2167A. The DoD-STD-2167A standard for software development includes risk management as a keyword. Specifically, it states the following:

> The contractor shall document and implement procedures for risk management. The contractor shall identify, analyze, prioritize, and monitor the areas of the software development project that involve potential technical, cost, or schedule risks [DoD88].

Risk management continues as a requirement under MIL-STD-498. A draft copy of the Software Development Plan (SDP) Data Item Description (DID) states:

> Software development management. Included shall be the approach for: Risk management, including a discussion of the technical, cost, and schedule risks identified for the project and plans for dealing with them.

Software Engineering Institute (SEI).

The SEI is a Federally Funded Research and Development Center (FFRDC) sponsored by the DoD. Its charter is to advance the practice of software engineering. The SEI Risk Program was established in 1990 to provide a focus for software risk management. The risk program goal is to develop and institutionalize a systematic approach for identifying and managing the uncertainty in developing software-intensive systems.

> The SEI has fostered the development of a software risk management community, which did not exist prior to 1990. The SEI is exploring existing techniques and developing methods for managing risk, assessing practice, preparing organizations to manage risk, and conducting prototype risk assessment methods. The SEI expects to provide the mechanisms for managing risk, as well as providing a process that can be implemented within a project and organization to facilitate the communication of risk issues [SEI93].

International Standards Organization (ISO).

ISO is a worldwide federation of national standards bodies. The ISO 9000 series of standards is a set of documents that specify quality system requirements. The specific standard in the ISO 9000 series that applies to software is ISO 9001. ISO 9000-3 is a guideline for the application of ISO 9001 to the development, supply, and maintenance of software. ISO 9000-3 is an international standard for quality management and quality assurance of software [ISO91]. The guidelines are intended to describe the suggested controls and methods for producing software which meet requirements primarily by preventing nonconformity at all stages of development. Several guidelines specifically address risk:

1. **Corrective action.** The supplier shall establish, document and maintain procedures for investigating the cause of nonconforming product and the corrective action needed to prevent recurrence and initiating preventive actions to deal with problems to a level corresponding to the risks encountered.

2. **Contract review.** Each contract should be reviewed by the supplier to ensure that possible contingencies or risks are identified.

3. **Development plan.** May involve dividing the work into phases, and the identification and analysis of the potential problems associated with the development phases and with the achievement of the specified requirements.

4. **Design reviews.** The design or implementation process should not proceed until the consequences of all known deficiencies are satisfactorily resolved or the risk of proceeding otherwise is known [ISO91].

American Society for Quality Control (ASQC).

The American Society for Quality Control (ASQC) produced standards for quality management and quality assurance that was approved by the American National Standards Institute (ANSI). These standards are technically equivalent to the ISO 9000-9004 series. The guidance for quality management emphasizes the importance of assessing risk.

> Risk, cost, and benefit considerations have great importance for both company and customer. These considerations are inherent aspects of most products and services [ASQC87].

Institute of Electrical and Electronics Engineers (IEEE).

The IEEE produced a standard for software project management plans that was approved by the American National Standards Institute (ANSI). Risk management is a managerial process identified in the Software Project Management Plan (SPMP).

> This subsection of the SPMP shall identify and assess the risk factors associated with the project.This subsection shall also prescribe mechanisms for tracking the various risk factors and implementing contingency plans. Risk factors that should be considered include contractual risks, technological risks, risks due to size and complexity of the product, risks in personnel acquisition and retention, and risks in achieving customer acceptance of the product [IEEE88].

2.1.2.2 Benchmarking

A benchmark is a standard by which software engineering practices or performance may be compared. Benchmarking is the process of comparing and measuring to gain information which will help an organization take action to improve its performance. Several studies in the software industry have captured software engineering best practices, which include risk management. A database of benchmark data in now available as a service to the software industry.

Software Measures and Practices Benchmark.

A major benchmark study of software practices and measures was performed by Software Quality Engineering and Xerox Corporation. The purpose of the survey was to benchmark the software development practices and measures of world-class companies.

> The intent was to seek out and select projects representative of the "best" software engineering work being performed and analyze these in detail to better understand what "best" really means and establish an industry performance benchmark [Hetzel90].

The study approach was to identify a collection of "best" projects and then study them to identify and classify their characteristics. Hetzel found varying perspectives on what "best" means, but noted that all perspectives are important. "Best" may mean ease of use from a customer perspective, market share to a product manager, or ahead of schedule to a software manager. Criteria used for selecting "best" projects was the

perception of use of better practices and measures with the perception of high quality results.

The survey asked 10 "best" companies to measure the degree of usage of various software practices recommended by industry literature. One practice under analysis and design was "Software risks (potential failures) are systematically analyzed." The average usage of all ten projects surveyed for this practice was 1.57. This score is interpreted to mean "Scattered ad hoc usage."

Risk management practices were found of special value in the management survey. One manager noted an effective practice was "Detailed program planning - clearly identify milestones, interdependencies, risks and contingencies." One ineffective practice found was "a tendency to treat everything the same -- low risk items get the same focus as high risk."

TQM/100 Alliance.

The TQM/100 Alliance is a strategic partnership of aerospace and defense (A&D) contractors formed to determine performance benchmarks and best practices. Results of a benchmarking study of business processes performance of fourteen A&D companies by Price Waterhouse answered the question, "What do the best performers have in common?" The factors which clearly differentiated the best performers were compared with characteristics of lesser performers.

> Results indicated that the route to improving profitability was through a continuous improvement program. Additionally, program management should be more formal, with formal risk management. The companies with formal risk management procedures outperformed the other companies in terms of on-time delivery of hardware as well as schedule and budget performance.

2.2 Software Risk Management

What is software risk management? What is the role of risk management in software engineering? A clear understanding of the origins of risk management applied to software will help to define its intended use in the software community.

2.2.1 Risk Management

The dictionary defines "risk" as "the possibility of loss [American85]." This definition can be translated into the fundamental concept of risk management: *risk exposure*, also called "risk impact." Risk exposure is defined by the relationship

$$RE = P(O_u) \times L(O_u)$$

where RE is the risk exposure, $P(O_u)$ is the probability of an unsatisfactory outcome and $L(O_u)$ is the loss if the outcome is unsatisfactory [Boehm91].

Another fundamental concept of risk management is *risk reduction leverage*. Risk Reduction Leverage (RRL) is defined as follows:

$$RRL = \left(RE_{before} - RE_{after}\right) / \left(RiskReductionCost\right)$$

where RE_{before} is the RE before initiating the risk reduction effort and RE_{after} is the RE afterwards. Thus, RRL is a measure of the relative cost-benefit of performing various candidate risk reduction activities. Basic risk management philosophy is to assess, then control risk. These principles of risk management are applied in practice by an iterative sequence of steps involving risk identification, analysis, prioritization, planning, resolution, and monitoring [Boehm89].

Risk management differs from traditional problem-solving, for the simple reason that a risk is not a problem. By analogy, risk management is to a risk what an algorithm is to a problem. An instance of a risk on a software project may be *reduced* by applying risk management procedures. Problems may be *solved* by applying algorithms:

> An algorithm is a general, step-by-step procedure for solving problems. A problem is a general question to be answered, and is described by its parameters, and a statement of the solution. An instance of a problem is obtained by specifying particular values for all the problem parameters. An algorithm is said to solve a problem if it can be applied to any instance and is guaranteed a solution [Garey79].

There are many reasons why risk management should be used on software projects to improve the chance of success. The following establishes the rationale for applying risk management in software engineering:

1. *Software Engineering professionals no longer treat software as an art.*
   - No more isolated disciplines: Integrate systems/software/hardware
   - Methods for risk management promote team involvement
2. *Reduced profit margins require engineers to be cost-conscious.*
   - Need to apply scarce resources more efficiently
   - Reduce waste and avoid expensive rework
3. *To provide a focus on the important issues.*
   - Determine the critical success factors
   - Compare between alternative actions
4. *Increasing system complexity increases project risk.*
   - Need to understand why decisions are made
   - Be prepared to adapt to changing circumstances.

2.2.1.1 Theory

Risk management is based on decision theory, uncertainty theory, utility theory, game theory, and creativity theory. These theories provide different strategies for decision-making under probabilistic conditions. All theories attempt to improve the

quality of decisions. All decision analyses involve the evaluation of two or more alternative courses of action [Clemen91].

Decision theory provides techniques to solve hard problems. Problems may be difficult because they are complex, have uncertain aspects, have multiple objectives or different perspectives. Decision theory uses probabilities to determine outcomes. Techniques to model hard problems include influence diagrams, decision trees, and Monte Carlo simulation.

Uncertainty theory uses probability to model unknown, uncertain or subjective decision problems. Uncertainty can be considered as the lack of adequate information to make a decision [Giarratano89]. An uncertain event has a probability distribution that defines the set of probabilities associated with all possible outcomes. Characteristics of probability distributions include expected value, variance, and standard deviation.

Utility theory attempts to model preferences and risk attitudes. Utility theory is used to select the alternative that maximizes the expected utility function. Utility functions can reveal whether the individual is risk-averse or risk-seeking.

Game theory is used in Artificial Intelligence (AI) research, because it uses heuristics for determining what alternatives to explore in large search spaces. Much of what we call intelligence resides in the heuristics used by humans to solve problems. The presence of an opponent in game-playing adds an element of unpredictability, and

the need to consider psychological as well as tactical factors in game strategy [Luger89].

Creativity theory suggests that our brain processes information at a level that is not accessible to our conscious thought. Creativity theory attempts to understand individual needs and motivations which are critical to the design of organizations that foster creative solutions [Clemen91].

The Society for Risk Analysis has published an international journal quarterly for over a decade called *Risk Analysis*. The journal provides a focal point for new developments in risk analysis for scientists from a wide range of disciplines. It deals with theoretical, social and psychological aspects of risks in the health and engineering disciplines [Plenum81].

2.2.1.2 Origins

The origins of risk management can be traced to the Babylonians use of risk analysis in 3200 B.C. Bernoulli's classic paper, "Exposition of a New Theory on the Measurement of Risk," was published in 1738. Pascal analyzed many risk situations that occur in games of chance. The insurance business is founded on the ability to assess and deal with risk. A classic book on risk was Allan Willett's *The Economic Theory of Risk and Insurance* [Willett51], first published in 1901.

2.2.2 Software Risk

The application of risk management concepts and principles to software has required tailoring. In software development there is often no significant amount of accurate data upon which to base important decisions. Decision-making in software development uses risk management in a manner different from the insurance business. In software development there is a large variation in human productivity [Brooks75], unlike the predictability of a manufacturing process.

Due to problems in the development of computer software, the government has driven the use of risk management to protect its investment in computer technology. Government managers found that as the complexity of systems increases, the software does not achieve the capabilities contracted for, that it is not delivered at the time specified, and that the cost is significantly greater than anticipated [Roe89].

Industry has begun to respond to the increased demand for software risk management. Development of the spiral model, the use of prototyping, and documenting a risk management plan are several ways industry has incorporated useful risk management techniques. Risk management is not the next silver bullet candidate of software engineering [Hall94]. Although risk management can minimize risk, it is always possible to have a bad outcome resulting from a decision made under uncertainty.

As leader of a multifunctional software risk management action team, I investigated foundational concepts for software risk. My dissertation builds on the

foundational concepts developed by government and industry that are described below in sections 2.2.2.1 and 2.2.2.2.

2.2.2.1 Government

The government has been a major driver in defining software risk management to reduce the acquisition risk for software-intensive systems. The Department of Defense (DoD) has been funding research in risk management as one response to its critical deficiencies in software development. Important contributions of the DoD include guidance from the Defense Systems Management College (DSMC), the Air Force Systems Command (AFSC), and funding of the Software Engineering Institute (SEI) Risk Program.

Defense Systems Management College (DSMC).

A memorandum from the Deputy Secretary of Defense in 1981 required DoD action to improve the acquisition process. One initiative was to increase the visibility of technical risk in budgets of weapon systems acquisition programs and incorporate the use of budget funds for technological risk. In response, the DSMC wrote a handbook to familiarize program management personnel with the concepts and techniques of quantitative risk assessment to assist them in internal management

decision-making [DSMC83]. The DSMC model for risk management is shown below (see Figure 4).

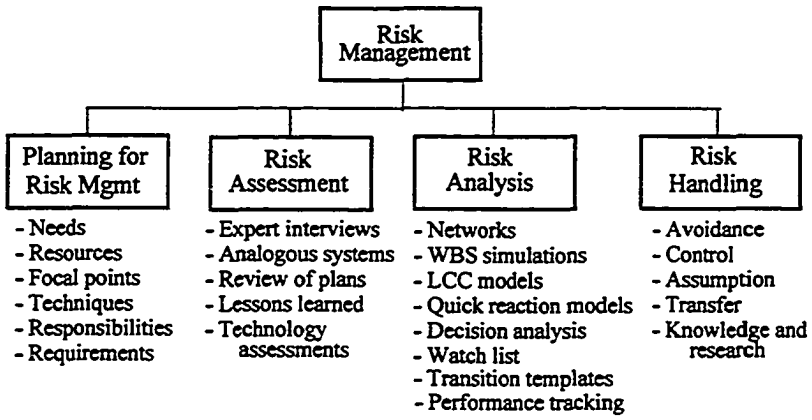


Figure 4. DSMC Risk Management Structure

Air Force Systems Command (AFSC).

The Air Force has several publications on risk, including the landmark AFSC/AFLC Pamphlet 800-45 on Software Risk Abatement written in 1988 [AFSC88]. Since 1983, the Software Development Integrity Program (SDIP) has used the Software Development Capability/Capacity Review (SDCCR) question set to lower the risk of weapon systems acquisition by determining contractor's software capability [Babel90]. The Air Force has developed the Software Development Capability Evaluation (SDCE) model as a basis for the state of the practice in software development [AFMC93]. The primary purpose of the SDCE is to reduce the acquisition risk for software-intensive systems.

Software Engineering Institute (SEI).

In 1984, the DoD awarded the Carnegie Mellon University (CMU) a contract to establish the Software Engineering Institute (SEI). The SEI's Risk Program is exploring existing techniques and developing methods for managing risk, assessing practice, preparing organizations to manage risk, and conducting prototype risk assessment methods. The SEI expects to provide the mechanisms for managing risk, as well as providing a process that can be implemented within a project and organization to facilitate the communication of risk issues. Communicating risk underlies the strategy of addressing risk throughout the acquisition process and strengthening the relationship between government and industry. The Risk Program at the SEI is chartered to develop risk methods and transfer the technology to industry.

Two important contributions of the SEI Risk Program are the Risk Management Paradigm and the Taxonomy Based Questionnaire (TBQ). The Risk Management Paradigm [VanScoy92] (see Figure 5) is a model of how the different elements of a risk management process interact. The Software Development Risk Taxonomy Structure

[Carr93] (see Figure 6) is a repeatable method for identifying risk in software projects using a risk taxonomy and associated questions.

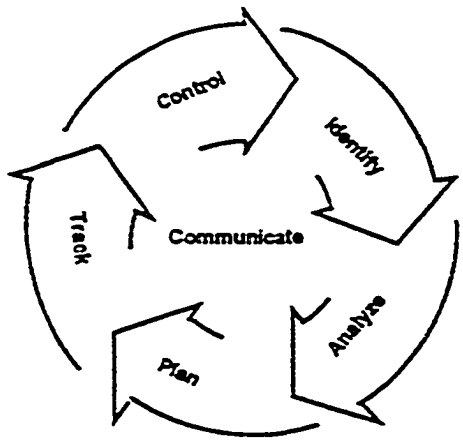


Figure 5. SEI's Risk Management Paradigm

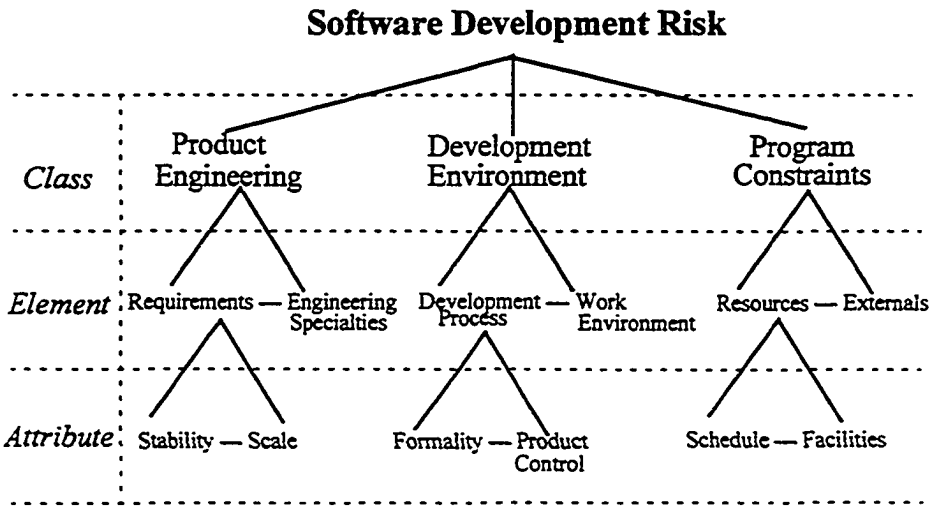


Figure 6. SEI's Risk Taxonomy Structure

2.2.2.2 Industry

Industry has helped to define software risk management to improve software productivity. The industry perspective is reducing the risk of financial loss of developing or maintaining software-intensive systems. Industry must respond to proposal requirements for risk management, as well as the standards for software risk management practices. Two industry leaders that have pioneered efforts in defining software risk management are Dr. Barry Boehm and Dr. Robert Charette.

Dr. Barry Boehm.

Boehm's observation that "successful project managers were good risk managers" led him to develop software risk management concepts that would be integrated into the practice of all developers. Dr. Boehm is the editor of the landmark *IEEE Tutorial: Software Risk Management*, published by the IEEE Computer Society [Boehm89].

Boehm described risk management as a practice with two primary steps, risk assessment and risk control (see Figure 7).

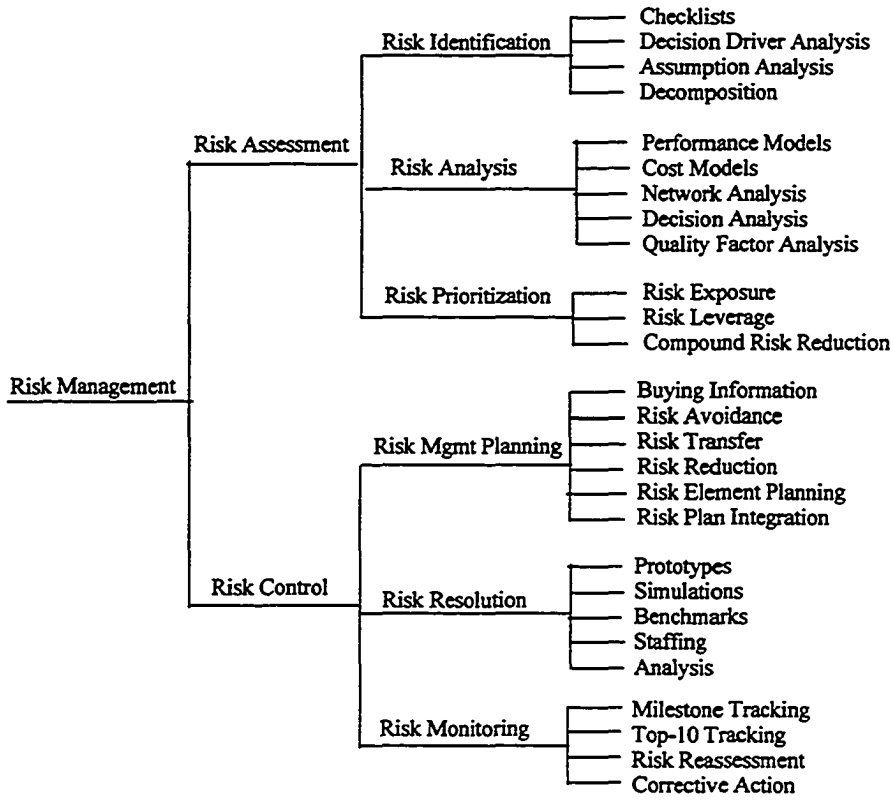


Figure 7. Boehm's Software Risk Management Steps

Boehm published a lifecycle development model that was iterative and risk-driven [Boehm88a] (see Figure 8).

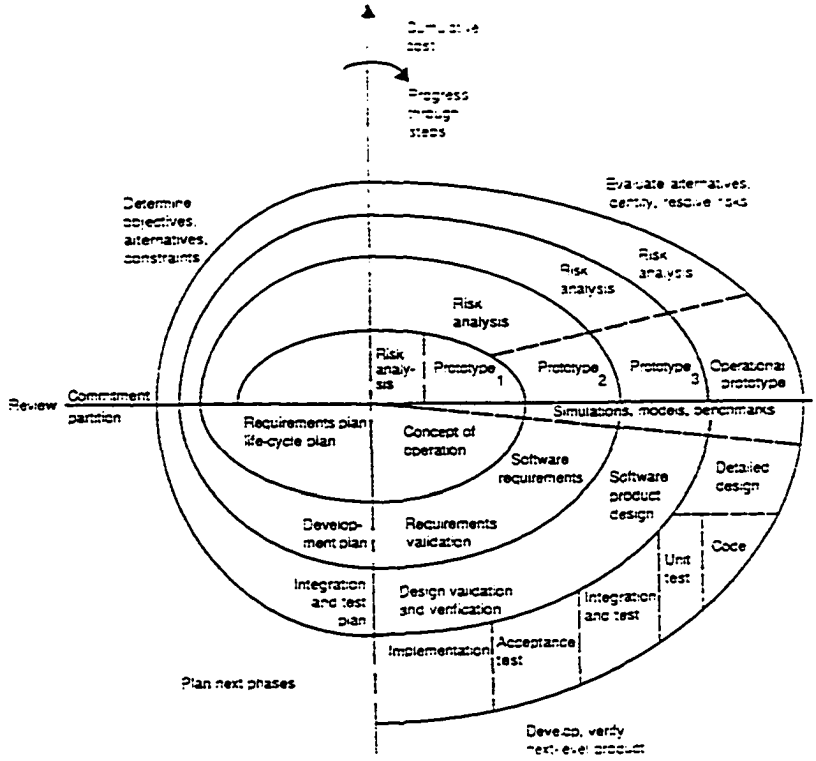


Figure 8. Spiral Model of the Software Process

Dr. Robert Charette.

In 1988, Dr. Charette published his first book on risk assessment and management, for use by software engineers [Charette88]. Charette incorporates Japanese quality concepts into his concepts for risk management. Charette's Risk

Engineering Taxonomy is another model of risk management that shows the steps in using risk management principles for engineering systems [Charette91a] (see Figure 9).

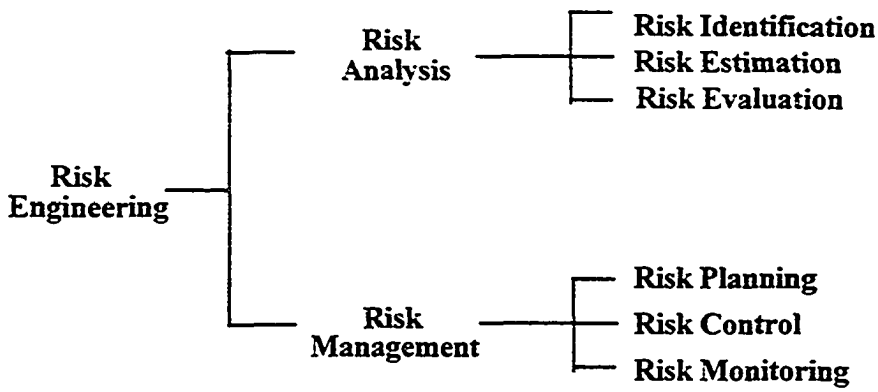


Figure 9. Charette's Risk Engineering Taxonomy

Charette was the first to conceptualize risk management as evolutionary and dynamic in nature in his Management of Risk Helix [Charette90] (see Figure 10).

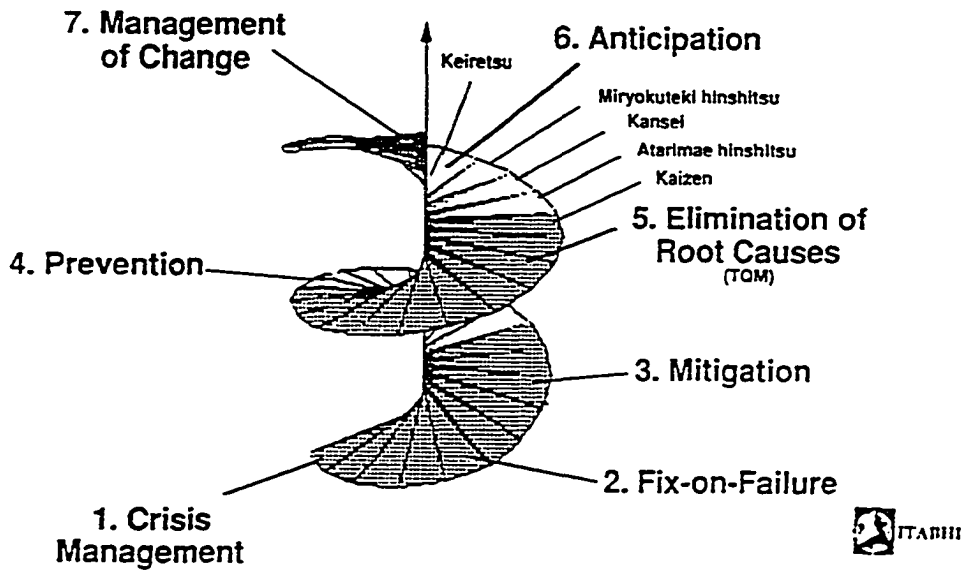


Figure 10. Management of Risk Helix

2.3 Current Practice

What is the state of the practice for risk management on software programs? Risk management process models, methods and tools currently exist to support software programs. A risk management state of the practice survey was taken at the Third SEI Conference on Software Risk from April 5-7, 1994.

2.3.1 Risk Management Technology

Risk management technology (RMT) applied to software incorporates risk management principles and practices into an approach to address risk on software projects. The state of the practice in software risk management is improving. The scope of involvement in risk management is expanding. Once a technique for government acquisition managers, now risk management involves the software managers, engineers, customers and subcontractors [Hall95]. The growth of risk management in software engineering is evident in the increased number of process models, methods, and tools published and in use on software projects.

2.3.1.1 Process

The models for risk management discussed in sections 2.2.2.1 and 2.2.2.2 have been tailored by organizations and defined as a risk management process. The process

specifically describes the set of activities that will be used to perform risk management. A process description should describe roles and responsibilities for who is involved in the process, and when risk management activities will occur. The process description should include how risk management fits into the overall software process.

2.3.1.2 Methods

From checklists to corrective action, traditional methods for software risk management as outlined in Boehm's Software Risk Management Steps are currently in use [Boehm89] (see Figure 7). New methods for risk identification have recently been published after extensive field testing [Carr93]. Risk assessment has evolved into a rigorous interview process with trained, experienced and independent assessment teams facilitating the assessment for a software project. Consensus techniques have been successfully used to prioritize risks in group situations. The concept of Team Risk Management (TRM) developed at the SEI brings government and industry contractors together to share risks for joint resolution [Higuera94]. A new method for software risk evaluation has been described in a recent SEI technical report [Sisti94].

2.3.1.3 Tools

Vendors exhibited risk analysis tools at the Third SEI Conference on Software Risk, which included a risk management expert system. RiskPro™ is a software tool

that automatically produces a risk management plan and performs cost-benefit analysis. @Risk is a software program that uses Monte Carlo simulation and is advertised as the "World's Best Risk Analysis Software." Spreadsheet software is currently used to automate risk tracking and maintain a history of risk information.

2.3.2 Technology Development

Risk management technology continues to be developed through technical collaboration and working groups that address risk in systems development. Working groups share information and address smaller tasks that further define risk management for the industry.

2.3.2.1 Technical Collaboration

The SEI Risk Program has several Technical Collaboration Agreements (TCA) with industry to develop and improve risk management methods. I led a TCA with the SEI for industry to improve risk management methods and develop an understanding of how risk management methods are being applied on software projects. The results of this TCA are documented in a paper that was presented at The Seventh Annual Software Technology Conference [Hall95].

2.3.2.2 Working Groups

Both SEI and NCOSE have risk management working groups. Working groups exchange information by mail or preferably electronic mail. Meetings are scheduled after the yearly conference. The SEI risk working group is chartered to determine the place for risk management in the CMM. The NCOSE risk working group is chartered to promote the definition, understanding and practice of risk management. They believe that world-class systems engineering must include world-class risk management. One of their objectives is to establish a more rigorous framework for the practice of risk management. My participation on these two working groups allows me information from both a systems and software perspective:

1. National Council on Systems Engineering (NCOSE) risk management working group.
2. SEI Risk in the Capability Maturity Model (CMM) working group.

2.3.3 Technical Exchange

Risk management awareness is increased through conferences and training. Training provides technical exchange for instructor and student, while conference presentations are another mechanism for technology transfer.

2.3.3.1 Conferences

The SEI hosts an annual conference on software risk. The annual Software Technology Conference (STC), hosted by the DoD, includes presentations on risk management. The Software Acquisition Conference sponsored by the Education Foundation of the Data Processing Management Association (EFDPMA) in 1994 discussed risk management as the process foundation for future acquisitions. The Software Engineering Process Group (SEPG) National Conference addresses software risk issues. The National Council on Systems Engineering (NCOSE) annual conference also addresses risk within the context of managing technical risks.

2.3.3.2 Training

Training in risk management techniques is now available. The SEI provides training in risk management concepts. Academic institutions such as the University of California at Berkeley provide risk management training through seminars directed at managers of software projects. Video based training is available for risk management by Dr. Richard Fairley of SEMA, Inc.

2.4 Future Directions

What are the future directions for risk management on software programs? There has been an increasing trend toward use of risk management in the development of computing systems. These trends can be seen in a timeline that shows the growth pattern for risk management (see Figure 11).

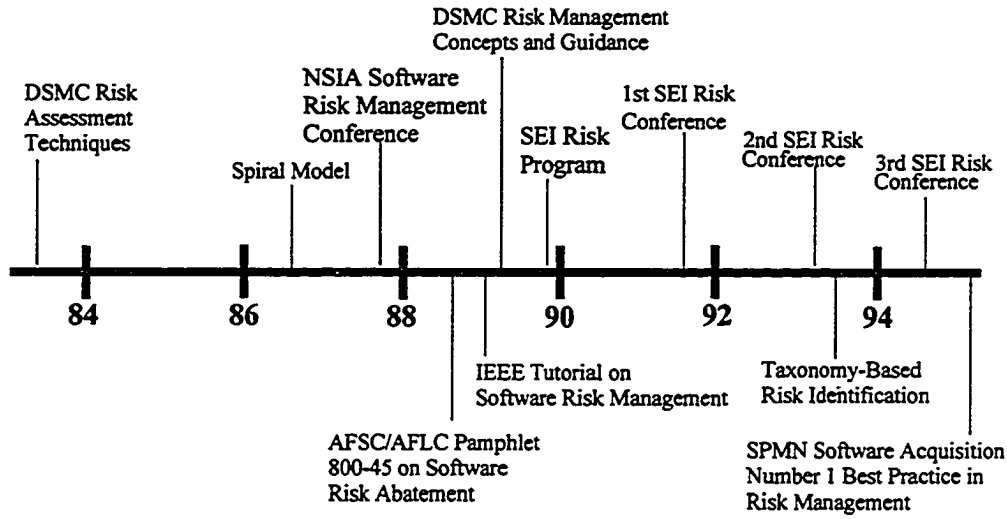


Figure 11. Software Risk Management Timeline

2.4.1 Risk Management Needs

The national problem of cost and schedule overruns in the DoD systems is due, at least in part, to unforeseen, misunderstood, or ignored software technical risks [DSMC91]. What are the needs for risk management in software systems?

The government has identified needs in managing the risks in acquisition and development of software within the DoD [DSMC91] (see Table 2).

Table 2. Prioritized DoD Needs for Software Acquisition Risk Management

Total Score	Government Software Risk Needs
46	Requirements risk assessment tool
42	Risk identification methods
32	Teach and train risk management
24	Procedures for risk assessment
21	Metrics to manage risk
20	Source selection of competent contractors
19	Risk analysis techniques
19	Risk-driven acquisition strategies
16	Best practices
16	Common problems and remedies
14	Software as part of systems engineering
13	Risk abatement guidelines
9	Criteria for prioritizing risks
9	Software sizing techniques
7	Risk concept definition
7	Risk communication
7	Lifecycle risk management
7	Program manager's risk handbook
5	Terminology
5	Templates for lifecycle risk management
3	Cultural change (i.e. shoot the messenger)
1	Contract provisions to manage software
0	Database of program data for PMO's
0	Publish a state-of-the-practice report

2.4.2 Risk Management Trends

What are the trends of risk management in software engineering? The government estimates it will take approximately 10 years to mature risk management methods [SWTS92]. Their assessment includes the following predictions for use of automated risk management tools and proactive risk management methods:

1. Risk assessment techniques (emerging 1992, mature 1997).
2. Risk analysis tools (science 1992, emerging 1997, mature 2002).
3. Automated risk management tools (mature 2002).
4. Proactive management methods and techniques (science 1992, emerging 1997, mature 2002).
5. Core proactive management methods and techniques (mature 2002).
6. Process and product metrics (science 1992, emerging 1997, mature 2002).

The future direction of risk management applied to software engineering is to assess and then control project risk by implementing a cost-effective risk management process. There appears to be a need for awareness and understanding of basic risk concepts and facilitation of the risk management process. Information for software risk management is available, but it is not in the mainstream or in routine use. I believe that risk management can be applied on software projects to raise the standards of software engineering. To encourage the disciplined use of risk management, Chapter 3, *Risk Management Technology Transfer*, determines barriers to adoption and critical success factors for institutionalization of software risk management.

CHAPTER 3

Risk Management Technology Transfer

Risk management involves methods for assessing and managing risk that improve our capability to control risks by making decisions under uncertainty. Risk management technology applied to software systems development is in the early stages of the technology development process [Hudson92] (see Figure 12). The three phases in the development of risk management technology can be described as follows:

1. **Invention** - Generating ideas and testing the feasibility of risk management methods. The SEI invented the TBQ in this manner.
2. **Innovation** - As a full time member of a SEPG, I helped develop a documented media set, standard process notation, training, and tested a beta version of risk management methods by facilitating on projects.
3. **Deployment** - The prototype risk management methods were used on several large software systems in industry. Projects have tailored the standard process to suit their unique organization.

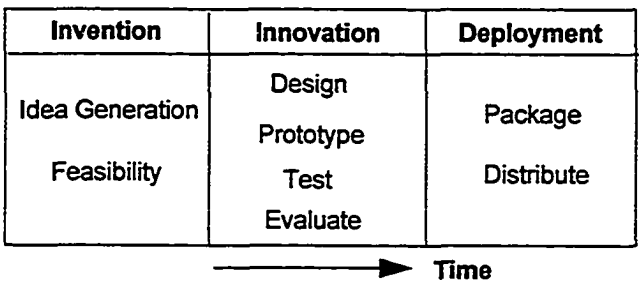


Figure 12. Technology Development Process

Technology transfer refers to those activities necessary to enable a corporation to apply a new technology [Korson92]. Technology transition is the activity of transferring a technology into an organization. It is a long-term process that requires motivation for change, organizational commitment, and resources. Technology adoption is a process that requires increasing levels of commitment over time [Hudson92] (see Figure 13).

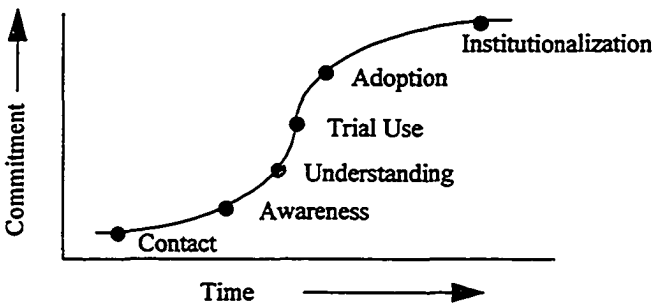


Figure 13. Commitment is a Phased Process

The adoption process may begin when a technology has been developed to the point that it is ready for deployment. Adoption and institutionalization occur when the organization routinely uses the technology to serve the organizational business purpose. Technology transfer is a process that can be systematic and predictable. It is based on understanding the need for change and satisfying that need while minimizing the obstacles inherent in technology transfer. I believe the technology transfer process can be used to successfully transition risk management technology (RMT).

What is the current motivation for the use of risk management in the software community? Motivations may be viewed as a *push* from external sources, such as customers, government, or industry. Motivations may also be viewed as a *pull* from internal sources, such as individuals within organizations who recognize a need for risk management. The motivation for change to the use of risk management must be sufficient to endure the long process of technology transition. Compelling reasons why change is needed provide motivation for the use of risk management as shown below (see Table 3). These dynamic forces are shaping the future of risk management in the software community today.

Table 3. Motivation for Risk Management

PUSH	PULL
Customers intolerant of software crisis	Increased competition and survival
Government sponsored Risk Program	Attendees at SEI Risk Conference
Industry increased professional standards	Industry TQM and customer focus
Software CMM Risk Management KPA	SEPG's comply with SEI CMM
Request for Proposals risk requirements	Proposal teams satisfying requirements

Even if an organization is full of receptive people, change still takes time. Changes in skills or procedures may take only weeks, while changes in structure can take months, and changes in strategy and culture take years. The longer it takes to change the organization, the harder it is to sustain the change. The key groups to target for adoption of risk management are those in leadership positions with authority.

3.1 Technology Transfer Models

For any software process technology to become dominant, it must first overcome a series of obstacles to adoptability. To encourage widespread use of risk management, the barriers for adoption of risk management technology must be determined. I used a two-dimensional framework based on theories about organizational and community wide technology adoption that was used to accurately describe adoption trajectories for other software process technologies [Fichman93].

The models of technology transfer evaluate adoption from two perspectives: diffusion of innovations and economics of technology standards. The first model, diffusion of innovations (DOI), examines attributes of individual adopters. The second model, economics of technology standards, examines attributes of community adoption. Together these two models of technology transfer were used to predict risk management technology's ultimate disposition. Model attributes were rated for risk management adoption using the scale displayed in Table 4. Because the evaluation was

Table 4. Rating Scale for Risk Management Technology Adoption

Evaluation	Criteria	Rating
Advantage	Generally positive characteristic or apparent benefit.	+
Neutral	Either no strong advantage or disadvantage was apparent, or the combined positive and negative characteristics cancelled each other out.	0
Disadvantage	Generally negative characteristic or associated cost.	-

based on a single data point, that being my own knowledge and experience in software risk management, the rating scheme is a rudimentary 3 point scale. The exercise of identifying the risks of risk management technology transfer, however crude or biased, is significant as a strawman for future research.

This chapter describes the adoption framework with respect to prediction of risk management technology. Using the rating scale for risk management technology adoption (see Table 4), I determined the *diffusion of innovations* model applied to risk management technology to be slightly positive, indicating that individuals and their organizations will make slow progress toward adoption. I found the *economics of technology standards* model was predominantly positive, indicating that benefits of risk management adoption will depend on an increasing size of the adopter community. As risk management adopters achieve a critical mass, risk management would be an expected standard for doing business and then become institutionalized in organizations. According to the two-dimensional technology adoption framework, I have categorized risk management technology as a "slow mover" (see Figure 14). This means that risk management technology will diffuse steadily but slowly because of the difficulty of individual organization adoption.

The four quadrants of the adoption framework are described in the paper by Fichman, and have been transcribed below:

- **Niche** - Adoption will start out fast among adopters who are relatively insensitive to standards issues or who have optimistic expectations about future levels of adoption. But adoption will plateau at a position short of dominance because of a failure to achieve critical mass.
- **Dominant Technology** - The technology will be rapidly adopted as a dominant process technology. It will face relatively low barriers to individual or community adoption.
- **Slow Mover** - The technology will diffuse steadily but slowly because of the difficulty of individual organization adoption.
- **Experimental** - The technology will need to evolve before it is widely adopted by mainstream organizations as a dominant technology [Fichman93].

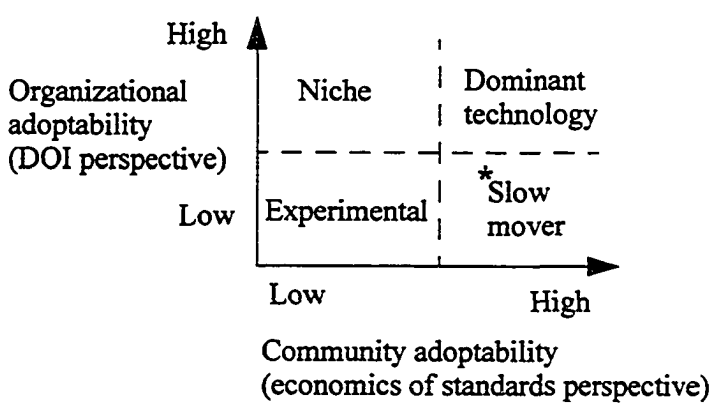


Figure 14. Risk Management Technology Adoption Prediction.

3.1.1 Diffusion of Innovations

Diffusion of innovations (DOI) is the study of technology adoption decisions of individuals or organizations. DOI does not take into account community issues that strongly affect the innovation's inherent economic value [Fichman93]. From this perspective, adoption is largely a communication process. Five generic attributes that influence rates of adoption according to the diffusion of innovations perspective are:

1. **Relative Advantage** - the innovation is technically superior to the technology it supersedes.
2. **Compatibility** - the innovation is compatible with existing values, skills, and work practices of potential adopters.
3. **Complexity** - the innovation is relatively difficult to understand and use.
4. **Trialability** - the innovation can be experimented with on a trial basis without undue effort and expense, and can be implemented incrementally.
5. **Observability** - the results and benefits of the innovation's use can be easily observed and communicated to others.

3.1.1.1 Relative Advantage

Is risk management technically superior to the technology it supersedes? For many organizations, SEI methods for structured and repeatable risk identification using a software risk taxonomy [Carr93] are superior to previous techniques used to identify risks. However, the SEI Risk Management Paradigm is not completely developed with proven methods and tools for risk analysis, planning, tracking, and control. Due to the

immaturity of RMT, there is a disadvantage to early adopters who must pioneer the field to help define it. Early adopters must believe the opportunity for gaining a competitive edge will be worth the price of pioneering the technology. The rating is **0**, since the advantage and disadvantage cancel each other at this time.

3.1.1.2 Compatibility

Is risk management compatible with existing values, skills, and work practices of the potential adopters? Quality principles such as Total Quality Management (TQM) are widely accepted in the software organization where I work, with 100% of the organization trained in TQM concepts. In this organization, TQM techniques for brainstorming, consensus and root cause analysis are reused in the risk management process. Continuous improvement, a philosophy for excellence that emerged in the software industry as Software Process Improvement (SPI) has been used to further refine the risk management process. Risk management is an extension of the corrective action process, which structures the activities and mechanisms for resolving problems.

Risk management uses decision analysis and technical performance measurement (TPM), however these quantitative methods are not routinely used in my organization. Decision analysis uses quantitative techniques to characterize various options by their possible outcomes in terms of risk exposure. Technical performance measurement tracks estimated and actual performance and calculates the difference. When the

difference reaches a threshold, corrective action is taken. If these skills were put into routine practice, quantitative risk management would be easier to adopt. The rating is +, since quality and measurement practices are currently in vogue in many engineering organizations.

3.1.1.3 Complexity

Is risk management relatively difficult to understand and use? In my experience as a member of a risk assessment team, risks are easily identified by individuals or peer groups. Risk analysis is relatively difficult due to the uncertainty that exists in the likelihood and consequences of risks. If a relative risk ranking is all that is desired, the task is much simpler. It is difficult to measure the benefits of risk reduction activities. Did the risk occur despite our efforts? Did the risk not occur because of our efforts? Risk tracking and reporting are largely administrative activities. The difficulty is in the time to perform the tasks. Automated tools in this area would reduce this difficulty. Risk control is difficult due to the lack of available data to evaluate for lessons learned. This feedback is essential for process improvement, but difficult because of the length of time required to measure improvement. The rating is -, due to the difficulty in controlling risks, which is caused primarily by the following factors:

1. A lack of follow-through after risk identification.
2. The resistance to quantify the risk and measure the results.

3.1.1.4 Trialability

Can risk management be experimented with on a trial basis without undue effort and expense, and can it be implemented incrementally? Risks can be identified on a trial basis, but the entire risk management process would need to be used to see risk reduction results. Incremental use of risk management would mean the use of the entire process at a low maturity level. The lower levels of risk management capability would not be as rigorous, quantitative, or proactive as higher maturity levels. Results may be sacrificed, which may be seen as a problem with risk management itself. The rating is **0**, since there is no penalty for incremental implementation, but the use of only a part of the process would not produce satisfactory results.

3.1.1.5 Observability

Can the results and benefits of risk management be easily observed and communicated to others? Yes, identified risks are easily communicated to the entire project team using a Top-10 Risk List. Benefits should be perceived from the customer and the program manager in terms of increased visibility into the project. The rating is +, because honest attempts at performing risk management will produce visible results.

3.1.2 Economics of Technology Standards

Economics of technology standards is the study of technologies that have significant increasing returns for adoption. The benefits of adoption largely depend on the size (past, present, and future) of the community of other adopters [Fichman93]. Four attributes of this model of technology transfer are:

1. **Prior Technology Drag** - a prior technology provides significant network benefits because of a large and mature installed base.
2. **Irreversibility of Investments** - adoption of the technology requires irreversible investments in areas such as products, training, and accumulated project experience.
3. **Sponsorship** - a single entity (person, organization, consortium) exists to define the technology, set standards, subsidize early adopters, and otherwise promote adoption of the new technology.
4. **Expectations** - the technology benefits from an extended period of widespread expectations that it will be pervasively adopted in the future.

3.1.2.1 Prior Technology Drag

A prior technology provides significant network benefits because of a large and mature installed base. Since software risk management is an emerging technology, it does not have a large and mature user base. The drag appears not to be from a prior technology, but from no prior experience identifying and communicating risks of software development. However, several government and industry organizations are working to define the needs and requirements for this emerging discipline. The rating

is **0**, since there is no prior drag, but the fact that RMT itself is immature itself cancels a positive rating.

3.1.2.2 Irreversibility of Investments

Adoption of risk management technology doesn't require irreversible investments in areas such as products, training, and accumulated project experience. Any investment in training should be viewed as learning fundamental skills required to handle complex problems. Tools that support risk management such as spreadsheet software are general purpose tools that have a high reuse potential. The rating is + for the inexpensive tool set used for risk management. One word of caution is that the time spent assessing risk with no follow-through may waste resources.

3.1.2.3 Sponsorship

The SEI Risk Program exists to define software risk management technology, set standards, work with early adopters, and otherwise promote adoption of software risk management through conferences and technical reports. Program Management and System Engineering organizations have established working groups in risk management as well. Other organizations, such as the Software Program Managers Network (SPMN) and the National Council on System Engineering (NCOSE) are also

defining standards for risk management. The SPMN reported Risk Management as the number one best practice in their *Software Acquisition Best Practices Initiative*. The rating is +, since there is a focus for definition of risk management.

3.1.2.4 Expectations

Risk management technology benefits from the expectations that it will be adopted in the future. This expectation has been established by DoD and other government procurements that require planning and implementation of risk management on large software development contracts. As the number of adopters increases, risk management practices will not be viewed as a discriminator, but a minimum standard in doing business with the government. The business climate of the 1990's is such that performing risk management is perceived as a method for corporate survival [Zweig94]. The rating is +, because of the expectation that using risk management is a way to handle the increasing risks we face in a global economy.

3.2 Barriers to Adoption

Technologies have a greater likelihood of success to the degree that barriers to adoption are lowered [Fichman93]. Barriers for adoption of Risk Management Technology (RMT) must be overcome in an organization and in the software engineering community. The barriers for adoption of RMT must be overcome to realize the benefits of this technology. Based on the identified barriers to adoption of RMT, I have summarized the obstacles and suggested a corresponding solution below (see Table 5).

Table 5. Barriers for Adoption of Risk Management Technology

Problem	Solution
Immature technology	Use risk management maturity model
Low customer expectations	Raise expectations and standards
Organizational inhibitors	Communicate benefits of risk management

3.2.1 Immature Technology

The first barrier to adoption is the low maturity level of RMT. Because it is still being defined, the lack of standards and automated tools slows the adoption process. The success of the SEI CMM for Software as a standard software process improvement model shows the benefit of using a standard maturity model to define and communicate activities to successfully performing any technology. The first step toward solving the

problem of low maturity level of risk management technology is to provide a risk management maturity model to the software community.

The complexity of risk management is rated low, due to the difficulty in controlling risks, which is often caused by the inability to follow-through after risk identification. Decision analysis is usually performed for problems with large potential losses or gains. If time were not a factor, quantitative risk management methods would be more likely to be used. If decision analysis tools, a risk management expert system, and reusable mitigation strategies were available, quantitative risk management would be less time consuming. The lack of automation is indicative of the low maturity of the technology itself.

3.2.2 Low Expectations

The attempt by government procurements to require the use of risk management to develop software systems lacks the follow-through required for a rigorous risk management program. I believe this is due in part to the customers' inability to use risk management effectively in their own organization. The government has established the expectation that they would award contracts only to those software contractors at SEI Level 3 or above. Because of this, contractors responded in a positive way to improve their software processes. If customers were more sophisticated, they would have higher expectations that would raise the standards for doing business.

3.2.3 Organizational Inhibitors

Several organizational factors may prevent the adoption of risk management. Among these inhibitors is a lack of management commitment and a resistance to change. Adoption of any technology takes time, but these two factors will slow the rate of adoption dramatically. The best method for overcoming these inhibitors is to increase the awareness of the benefits of risk management, and relate that to the business strategy.

3.3 Critical Success Factors

Overcoming the barriers to adoption is a necessary, but not sufficient condition for adoption of risk management technology. Removing the barriers to adoption provides a climate for risk management to exist, but will not make risk management successful within a specific organization on a specific project. What are the most important factors in implementing risk management successfully? I have outlined the critical success factors required to transition and use risk management technology effectively using a Cause & Effect Diagram (see Figure 15). The Cause & Effect Diagram was developed to represent the relationship between some "effect" and all the possible "causes" influencing it [Brassard88]. The effect or problem is stated on the right side of the chart and the major influences or "causes" are listed to the left. This diagram shows that the ability to perform risk management is caused by four major factors:

1. **Project** - establish the infrastructure for risk management.
2. **People** - execute the risk management process and procedures.
3. **Process** - identify the activities to perform risk management.
4. **Procedures** - develop the approach to performing risk management.

These major factors contain major, minor and possible causes of effecting risk management effectiveness. The major factors are described below in Figure 15.

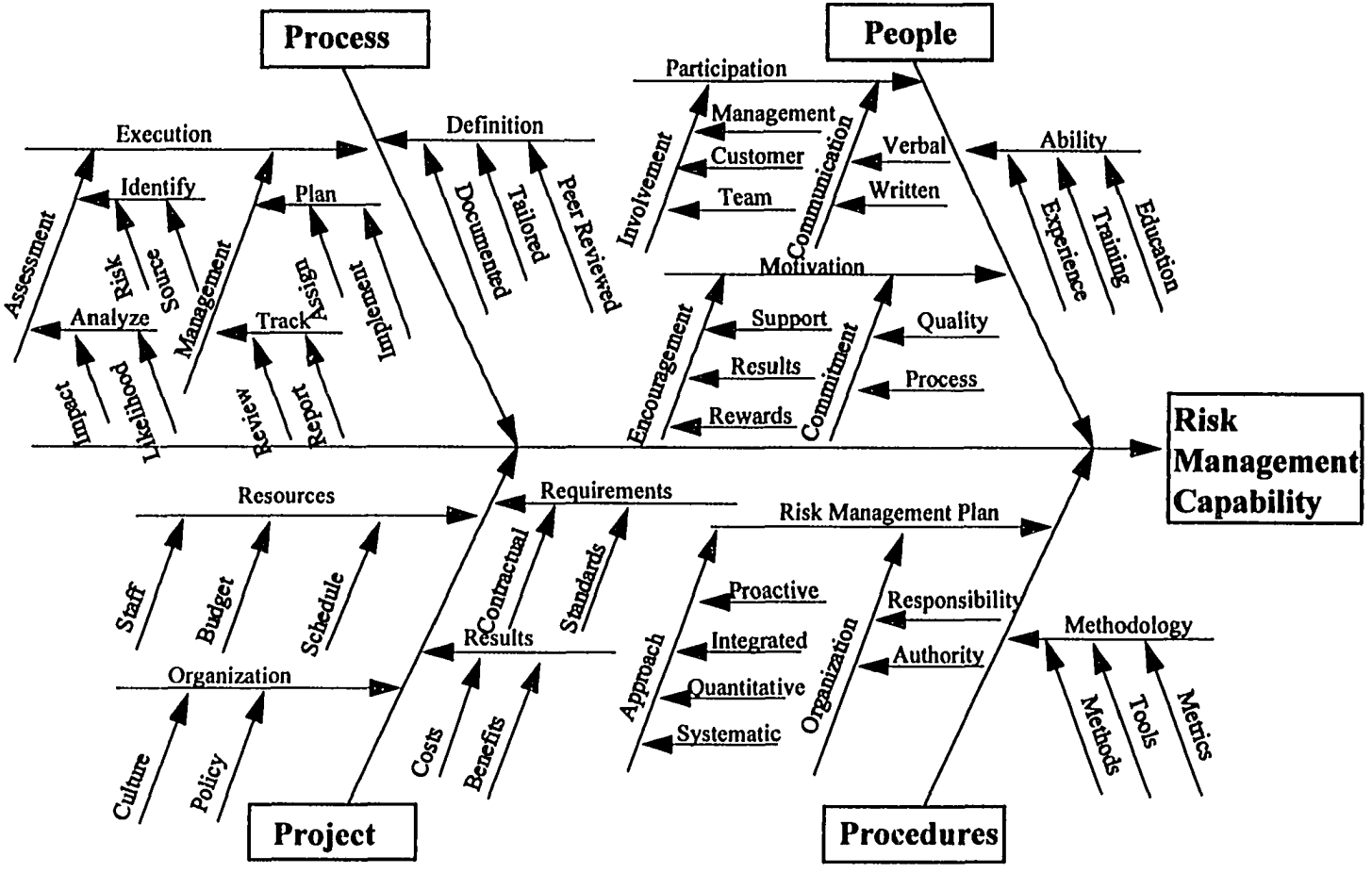


Figure 15. Risk Management Capability Cause/Effect Diagram

3.3.1 Project

Risk management is instantiated for a specific project, that lives within its own organizational context with unique requirements and lifecycle. Some projects are inherently more complex or safety critical. These projects may require more rigorous risk management methods. The project provides the resources in support of risk management, such as staff, budget, and schedule. Positive results perpetuate the use of risk management. The major causes of risk management effectiveness on a project are:

1. Requirements
2. Resources
3. Results
4. Organization.

3.3.2 People

Technology is primarily transferred by people [Cutler 89]. The participation of the people is a key factor to the success of communication regarding project risk. Their ability and motivation to manage risks are essential. For example, one of the success factors for ability is training in risk management. We must realize that without properly preparing those who will use a new technology, we can't expect it to be accepted or

successfully introduced [Debou93]. The major causes of people's effective use of risk management are:

1. Participation
2. Ability
3. Motivation.

3.3.3 Process

The process must be defined for the project. The definition should tailor an existing organizational standard process that is leveraged for all projects. The execution of the defined process should produce predictable results. The major causes of risk management process effectiveness are:

1. Definition
2. Execution.

3.3.4 Procedures

The procedures for performing risk management on a project may be embodied in a Risk Management Plan (RMP). It is important to define the organization structure and approach for using risk management. The proper approach for the project is best determined by understanding the current risk management capability. The

methodology is the means for executing the process. Methodology includes specific methods and tools, which also depend on current risk management capability. The major causes of effective risk management procedures are:

1. Risk Management Plan
2. Methodology.

3.4 Risk Management Prediction

In this chapter, I have predicted that risk management technology will be a "slow-mover" in organizations, and in the software engineering community. Removing the barriers to adoption is a necessary, but not sufficient condition to ensure institutionalization of risk management. Critical success factors were identified and described to provide an understanding of the important factors in implementing risk management successfully. Barriers to adoption include low customer expectations for use of risk management and organizational inhibitors, such as resistance to change. Another barrier to adoption is the low maturity of the technology itself. To take a step toward solving the problem of low maturity, Chapter 4, *Risk Management Capability*, presents an evolutionary framework and maturity model for risk management.

CHAPTER 4

Risk Management Capability

The need for risk management capability is evident in my research on the state of the practice in software engineering and results of a *Risk Management Survey* (RMS) on three projects within a specific organization [Hall95]:

- Risk is increasing due to the increase in software system complexity.
- Gap between risk management state of the practice and state of the art.
- Gap between risk management practice performance and importance.

To understand the path to increasing risk management capability, I have developed an evolutionary framework and a maturity model based on fundamental principles of quality, maturity and technology transfer (see Table 6).

Table 6. Principles of Quality, Maturity and Technology Transfer

Level	Quality	Maturity	Technology Transfer
1	Caveat emptor	Initial	Awareness
2	Commitment	Plan the work	Understanding
3	Empowerment	Work the plan	Trial use
4	Measurement	Measure the work	Adoption
5	Continuous improvement	Work the measures	Institutionalization

The *Risk Management Evolution Framework* (RMEF) is a framework for a risk management maturity model that I presented at the Third SEI Conference on Software Risk. The RMEF is a practical guide to understanding the evolution of essential elements of risk management technology. The *Risk Management Evolution Framework* groups the essential elements of risk management technology by dimensions of process, infrastructure, and implementation [Hall94]. The framework maps the elements to an evolutionary scale that describes five stages of maturity, through increasing levels of knowledge, commitment, communication, efficiency, and effectiveness.

The *Risk Management Capability Maturity Model* (RM-CMM) provides the detail required for a risk management maturity model. The RM-CMM is a five level roadmap for incorporating risk management technology into an organization and achieving the capability to control risk. The RM-CMM may be used as a basis for appraising current risk management practice, establishing an improvement plan, measuring improvement and monitoring progress over time. The RM-CMM brings the discipline of risk management to the development of software intensive systems to advance the state of the practice in software systems engineering and management.

4.1 Risk Management Evolution Framework

The *Risk Management Evolution Framework* (RMEF) provides a description or characterization of the states or "stages" of a risk management maturity model (see Figure 16).

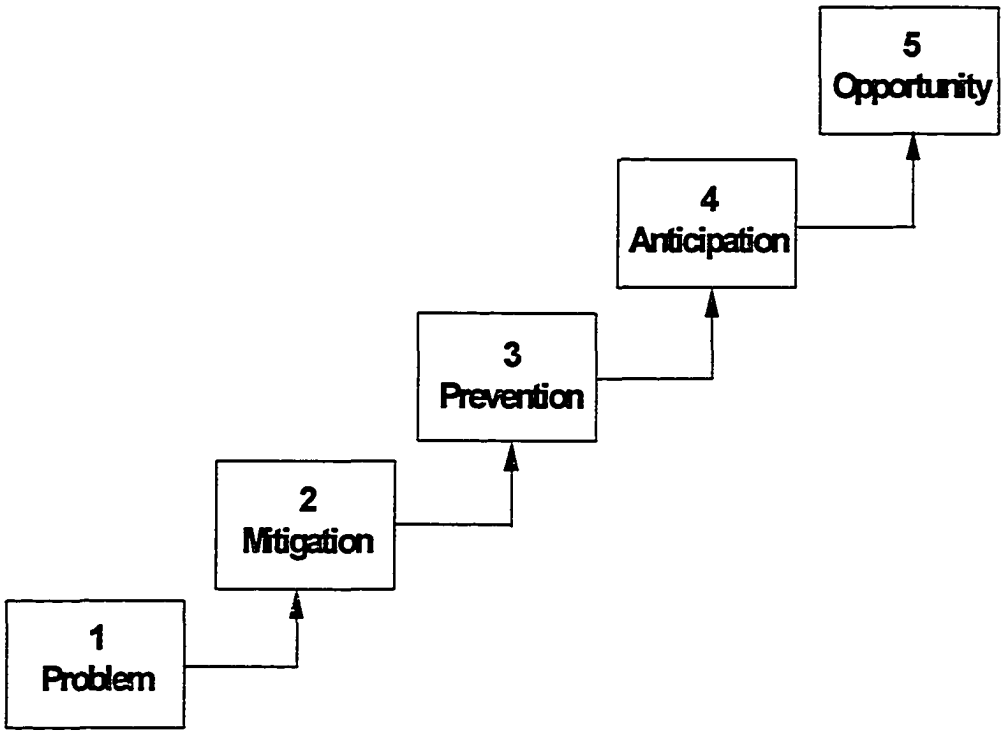


Figure 16. Evolutionary Stages of Risk Management Technology

I intended the *Risk Management Evolution Framework* as a conceptual model that provides a practical guide to understanding the evolution of essential elements of risk management technology. The RMEF builds on foundational concepts developed by Dr. Barry Boehm, Dr. Robert Charette, and the Software Engineering Institute (SEI).

Boehm's observation that "successful project managers were good risk managers" led to his development of the risk-driven Spiral Model for software development. Boehm's software risk management concept of operations was a discipline integrated into the practice of all developers [Boehm89]. Charette showed the dynamic "Management of Risk" as a helix that incorporates Japanese quality concepts of Kaizen (continuous improvement), Kansei (expanding environments of interest), and Keiretsu (controlling suppliers) [Charette93]. The SEI Risk Program developed a Risk Management Paradigm, a model of how the different elements of a risk management process interact [VanScoy92]. My practical experience facilitating risk management in industry [Hall93], and working with the SEI to develop and transfer risk management technology, has contributed to the development of my *Risk Management Evolution Framework*. Risk management data was gathered from a *Risk Management Survey* that substantiates the *Risk Management Evolution Framework* [Hall95]. These results are presented in Chapter 5 of this dissertation. The *Risk Management Evolution Framework* may be used as a model of a strategic plan for the transition of risk management technology into an organization.

The *Risk Management Evolution Framework* provides understanding and motivation for evolving risk management technology. The framework groups the essential elements of risk management technology by dimensions of process, infrastructure, and implementation. *Process* defines the tasks in performing risk management, *Infrastructure* is the organizational foundation that supports risk

management, and *Implementation* is the execution of the risk management process. The framework maps the elements to an evolutionary scale that describes the transformation from problems to opportunities in five stages, through increasing levels of knowledge, commitment, communication, efficiency, and effectiveness. The *Risk Management Evolution Framework* illustrates risk management technology as multidimensional and evolutionary, with major paradigm shifts occurring from one stage to the next (see Table 7).

Table 7. Risk Management Evolution - the Journey from Problem to Opportunity

Risk Management Evolution	Paradigm Shift	
Problem to Mitigation	Crisis management	Risk management
Mitigation to Prevention	Management activity	Team activity
Prevention to Anticipation	Risks are subjective	Risks are quantitative
Anticipation to Opportunity	Risks with negative impacts	Chances with opportunity costs

The evolution of essential elements of risk management technology must be understood to establish a realistic plan for incorporating risk management technology and for successfully implementing a risk management process within an organization [Hall94] (see Figure 17).

Dimensions		Risk Management Evolution Framework				
		1 – Problem	2 – Mitigation	3 – Prevention	4 – Anticipation	5 – Opportunity
PROCESS	Identify	Not seen as positive	Risks are assessed	Risks are volunteered	Risks are sought out	Chances to do better
	Analyze	None	Prioritize risks	Analyze source of risk	Quantitative values used	ROI is calculated
	Plan	None	Action plan is discussed	Action plan is documented	Action plan is executed	Action plan is revised
	Track	None	Monitor critical risks	Monitor all risks	Monitor triggering events	Correct for deviations
	Control	None	Discussions increase awareness of what could be improved	Written evaluations document what could be improved	Written evaluations are analyzed and documented as lessons learned	All feedback is used to improve the process
INFRASTRUCTURE	Policy	No written standards	Report risks at reviews	Commitment to process	Commitment to metrics	Reward for innovation
	Communicate	Lack of communication regarding risks	Risks gathered from lower levels and not communicated to higher levels	Communicate risks within the program team	Between the program team and the customer	Betweeen the program team, customer and the end-user
	Commitment	Upper management	Quality assurance	Management	Employees	Customer and end-user
	Resources	None	Minimal schedule allocated	Minimal schedule and budget are allocated	Sufficient schedule, budget and some resources	Optimal schedule, budget and resources allocated
	Training	No training	Basic risk concepts	Risk management process	How to quantify risks	How to manage risks
IMPLEMENTATION	Participants	Program manager	Program manager and key technical staff	Program team with a single risk champion	Program team and customer, and a few risk champions	Program team, customer, and end-user, with many risk champions
	Procedures	Ad hoc	Verbally stated	Documented	Updated at milestones	Living document
	Methods	Ad hoc	Risk surveys	Risk taxonomy	Risk management form	Risk metrics graphs
	Tools	Ad hoc	Top 10 risk list	Risk database	Technical Performance Measures	Automated risk analysis
	Metrics	None	Defined	Collected	Analyzed	Reported

Figure 17. Risk Management Evolution Framework

4.1.1 Dimensions and Essential Elements

Risk management can be viewed from different perspectives. Three perspectives described by the *Risk Management Evolution Framework* are process, infrastructure, and implementation. These three dimensions provide a separation of responsibility and focus. Parallel efforts in each dimension may speed the transition and evolution of risk management in an organization. These efforts must be synchronized for maximum effectiveness.

4.1.1.1 Process

Process defines the tasks in performing risk management. The process shown below and in the *Risk Management Evolution Framework* is the SEI Risk Management Paradigm [VanScoy92]; however, any risk management process may be substituted. The risk management process may be defined by a process group in an industry, academic, government, or standards organization. The risk management process should be tailored to meet the needs of a project depending on factors such as size and budget. Tailoring suggestions within each process element will allow for flexibility of the process and applicability to a variety of software projects. The risk management process relies upon risk identification to provide data to continue the process. The essential elements of the process dimension of risk management are:

- **Identify** - communicate perceived risk or source of risk.
- **Analyze** - evaluate risk based on established criteria.
- **Plan** - plan to mitigate, reduce or resolve risk.
- **Track** - monitor plan implementation to completion.
- **Control** - use results and feedback for improvement.

4.1.1.2 Infrastructure

Infrastructure is the organizational foundation that supports risk management. Infrastructure is developed top-down by management, by providing leadership and commitment to risk management. Infrastructure is developed bottom-up by empowerment of the workforce through training and encouraging risk communication. Empowerment may be developed through recognition of risk reduction efforts and reward for positive results. Risk management infrastructure relies upon a documented policy requiring risk management on projects. The essential elements of the infrastructure dimension of risk management are:

- **Policy** - the written standards for addressing program risk.
- **Communication** - the mechanisms for risk communication.
- **Commitment** - the people who advocate the use of risk management.
- **Resources** - the budget and schedule allocated to risk management.
- **Training** - the education that formalizes risk management knowledge.

4.1.1.3 Implementation

Implementation is the execution of the risk management process. Implementation is performed by a team according to a documented plan, which describes the procedures, methods and tools used on the project. Risk management implementation depends on the participants to execute the plan. Besides projects, the risk management process may be implemented by steering committees, senior executives, business area teams, proposal teams, middle management, line management, or individuals. The essential elements of the implementation dimension of risk management are:

- **Participants** - the people who perform risk management activities.
- **Procedures** - the documented plan for performing risk management.
- **Methods** - the techniques that implement the process.
- **Tools** - the instruments used to execute risk management.
- **Metrics** - the measures used to determine effectiveness and efficiency.

4.1.2 Evolutionary Stages

The *Risk Management Evolution Framework* depicts risk management technology as multidimensional and evolutionary, with paradigm shifts occurring from one stage to the next. The framework maps the elements to an evolutionary scale that describes the transformation from problems to opportunities in five stages, through increasing levels of knowledge, commitment, communication, efficiency, and

effectiveness. The evolution of these essential elements of risk management technology must be understood to establish a realistic plan for incorporating risk management technology, and successfully implementing a risk management process within an organization.

Problem describes the characteristics of the process, infrastructure, and implementation when risk identification is not seen as positive and people are too busy solving existing problems to think about risks that may occur in the future. ***Mitigation*** details the shift from crisis management to risk management. Management incorporates risk management technology, which asks the questions "What can go wrong?" and "What are the impacts?" ***Prevention*** discusses the shift from risk management viewed as a manager's activity to a team activity. This is a transitional stage where the approach changes from reactive avoidance of risk symptoms to proactive elimination of the root cause of risk. ***Anticipation*** describes the shift from subjective to quantitative risk management through the use of metrics to anticipate predictable risks. ***Opportunity*** is a positive vision of risk management that is used to innovate and shape the future. Potentially the most powerful paradigm shift is perceiving risks as chances to save money and do better than planned.

4.1.2.1 Stage 1: Problem

The *Problem* stage of risk management evolution is characterized by a lack of communication which causes a subsequent lack of coordination. *Problem* describes the characteristics of the process, infrastructure, and implementation when risk identification is not seen as positive and people are too busy solving problems to think about the future. Risks are not addressed until they become problems, because either management was not aware of the risk, or inaccurately estimated the risk's probability of occurrence. Since management reaction to hearing risks is typically "shoot the messenger," most people won't deliver bad news. Crisis management is used to address problems, and people learn that fire-fighting can be exciting, but it causes "burnout."

Stage 1: Problem - Process.

The first element in the process dimension of risk management is *Identify*. In the *Problem* stage of risk management evolution, risk identification may be viewed as a waste of time. There is a belief that risk assessment is too subjective or imprecise to be of value [Kirkpatrick92]. The process is not used because the infrastructure will not support communication regarding risks, or is too busy solving today's problems that tomorrow's potential problems are not discussed.

Stage 1: Problem - Infrastructure.

There are no written standards for addressing program risk. Since there is no training for risk management concepts, the majority of people may be unaware that

methods exist for addressing problems before they occur. The predominant philosophy about uncertainty is "what you don't know won't hurt you." Problems typically appear as significant cost overruns and schedule slips. When these occur, upper management and the customer are made aware of the problems. The problems are often handled by a tiger team of fire-fighters who apply resources to address the problem. Use of scarce resources is likely to cause additional stress to the program budget and schedule.

Stage 1: Problem - Implementation.

The only participant in a nonexistent risk management process is the program manager, who is responsible for planning the program budget, schedule, and allocating resources. Any procedures, methods, or tools used would typically be incomplete and considered ad hoc. Metrics would not be defined.

4.1.2.2 Stage 2: Mitigation

The *Mitigation* stage of risk management evolution is characterized by an introduction to risk concepts. *Mitigation* details the shift from crisis management to risk management. People may be aware of risks but do not systematically confront them. Since their knowledge and experience using risk management is limited, people may be unsure of how to communicate risks. Managers use risk management by attempting to reduce the probability and impact of critical risks by implementing a contingency plan

if the bad outcome occurs. Primary emphasis is in the early phases of a software product definition, since the major risk reduction leverage is in the early phases [Boehm89].

Stage 2: Mitigation - Process.

In the Mitigation stage of risk management evolution, identification attempts to find the major risks before they may adversely affect a program. The process element *Identify* provides methods to communicate perceived risk and sources of risk. One method is a risk assessment, which is performed at the beginning of a program. The risk assessment uses an independent and trained assessment team to enable communication of risks among peer groups in interview sessions. Assessment results are briefed to the project team to provide a baseline of risks for the project to manage. Thereafter, risks may be identified by individuals when asked directly, but are not typically volunteered. *Analyze* is the next element in the risk management process, which is the evaluation of risk data based on established criteria. Risks are analyzed, but not on a regular basis and typically just prior to milestone reviews. Program management usually prioritizes the set of risks to identify the top risks. The process element *Plan* develops the approach to mitigating a risk that has been identified and analyzed. After an action plan approach is discussed, a responsible person is assigned to execute the plan. The risk action plans are informal and not usually documented. The process element *Track* is required to ensure effective action plan implementation. Risk metrics and triggering events are not used in the *Mitigation* stage. Monitoring informal action plans is difficult. Only critical risks

are monitored. The process element *Control* uses results of the process as feedback. Informal discussions increase awareness of what could be improved.

Stage 2: Mitigation - Infrastructure.

An acceptable *Policy* requires that risks be reported at program reviews. Typically, the Top-10 Risk List is shown that satisfies this requirement. The culture represents a belief that "what you don't know may hurt you." *Communication* regarding risk is usually one way, from the bottom-up. Risks are gathered from lower levels without passing them up the management chain. The motivation for using risk management is to make more informed decisions to avoid big "career threatening" mistakes. Minimal schedule resources are allocated, since risk management is viewed as a planning activity performed by management as part of their current job description. *Training* provides the knowledge of basic risk concepts.

Stage 2: Mitigation - Implementation.

The *Participants* who perform risk management activities include the program manager and key technical staff. *Procedures* are verbally stated but not documented in a RMP. *Methods* used include risk surveys, to directly request input of perceived risk by individuals. The survey requires no prior preparation, and provides the top risks with their rating of estimated impact. Reports of top risks are provided to management at program reviews. A Top-10 Risk List is typically in a softcopy format of an automated tool so it can be easily updated. *Metrics* may be defined, but are not collected.

4.1.2.3 Stage 3: Prevention

The *Prevention* stage of risk management evolution is characterized by team and occasional customer involvement. *Prevention* discusses the shift from risk management viewed as a manager's activity to a project team activity. This is a transitional stage where the approach changes from reactive avoidance of risk symptoms to proactive elimination of the root cause of risk. Managers understand that risk management is a dynamic process that cannot be performed in isolation. For risk management to succeed, it must occur at each level within an organization [Charette93]. Instead of focusing on cost and schedule risk (a management perspective, usually a symptom of technical risk), a focus on technical risks leads to discovery of the source of risk. *Prevention* is a transitional stage and turning point from a reactive to proactive approach to risk management. *Prevention* evolves from avoidance of risk symptoms to attempts at eliminating root causes. Most people are experienced in risk identification, but are unsure of how to quantify risks.

Stage 3: Prevention - Process.

In the *Prevention* stage of risk management evolution, identification attempts to find any risk that would adversely affect a program budget, schedule or system functionality. Sources of risk are also identified. A risk assessment is performed at program milestones by an independent assessment team or facilitator. Risks are identified throughout the program by individuals who volunteer the information. After

they are identified, risks are assessed according to subjective evaluation criteria. Prioritization of all analyzed risks occurs by group consensus. This team approach provides buy-in for decisions the team must live with. The action plan documents the approach to reducing a risk. Responsibility is made clear by documenting the assigned person on the plan. Authority to achieve the plan is delegated. To ensure action plan implementation, a risk database is used to track the action plans and monitor all open plans. Written evaluations document ideas of what could be improved in the risk management process.

Stage 3: Prevention - Infrastructure.

A *Policy* requires that a risk management process be used, and a Risk Management Plan is written to tailor the process. The culture represents a belief that "what you don't know will hurt you." Information regarding risks is gathered from lower levels and passed on to management. The motivation for improved risk *Communication* is to prevent problems and surprises. Often discovery at one level requires action at a higher level [Kirkpatrick92]. Minimal schedule and budget *Resources* are allocated to risk management, since risk assessments are only performed at program milestones. *Training* provides the knowledge for understanding the risk management process. Training in quality management provides a working knowledge of root cause analysis that is reused in risk management.

Stage 3: Prevention - Implementation.

The project team performs risk management activities, which includes the program manager and key technical staff. A risk champion advocates risk management and acts as a catalyst to promote its use. A RMP documents the *Procedures* for performing risk management. *Methods* used include risk surveys and a risk taxonomy-based questionnaire [Carr93], which provides a structured and repeatable way to identify and classify risks. A risk database captures risk data in a softcopy format so that it can be easily updated. Reports of top risks are provided to both management and customers at program reviews. *Metrics* are collected, but not analyzed. Collected metrics support the belief that risk prevention is more cost-effective than risk detection. Industry data shows that software development costs are reduced by early detection of risks [Hall93].

4.1.2.4 Stage 4: Anticipation

The *Anticipation* stage of risk management evolution is characterized by the use of metrics to anticipate failures and predict future events. Predictability involves the ability to learn from, adapt to, and anticipate change [Charette92b]. *Anticipation* describes the shift from subjective to quantitative risk management through the use of metrics to anticipate predictable risks. Risk management is used by the program team and customer to quantify risks with reasonable accuracy to focus on the right priorities.

A proactive approach to actively attacking risks and assessing alternatives is used. Alternatives are easier to compare using a quantitative approach.

Stage 4: Anticipation - Process.

In the *Anticipation* stage of risk management evolution, risks and sources of risk are proactively sought out. Mechanisms are in place to input risks by anyone at any time. Self risk assessments are performed by the project team after a risk baseline has been established by an independent assessment team or facilitator. After they are identified, risks are assessed according to quantitative evaluation criteria. Prioritization of all analyzed risks occurs by ordering the quantitative evaluation results. A risk database with triggering events is used to ensure action plans are executed and tracked to closure. Status of action plans is discussed at program reviews to identify progress and/or potential problems and correct for variations. Written evaluations document ideas of how the process could be improved. Evaluations are analyzed and documented as lessons learned.

Stage 4: Anticipation - Infrastructure.

A *Policy* requires that risk management metrics be used on a program. The culture represents a belief that "you can't manage what you can't measure." The attitude is that risks can be quantified. Information regarding risks is gathered from lower levels and passed on to management and the customer. The motivation for including the customer in risk *Communication* is to provide buy-in for decisions and cooperative team risk management. Since risks are shared and acted upon cooperatively by both management

and customer, common *Commitments* are made and trust begins to develop. The team approach tends to avoid people conflicts. Sufficient schedule, budget and some *Resources* are allocated to risk management. Action plans may require funds for resolution approaches such as prototyping or benchmarking. *Training* is available for understanding how to use metrics to measure the risk management process results. Employees are educated in how to manage their own set of risks. Having the knowledge to perform their jobs better causes the people to be empowered.

Stage 4: Anticipation - Implementation.

The program team and customer perform team risk management activities. A few risk champions exist that advocate the use of risk management and act as a catalyst to enable the process. The RMP is updated at program milestones. *Methods* used include checklists or lists of the possible sources of risk and risk indicators. A risk management form is available to all program team members and may be submitted to identify perceived risk at any time. Reports of top risks are provided to the project team and customer throughout the program. *Metrics* are analyzed, but not used to improve the process. Technical Performance Measures (TPM) are used to trigger risk mitigation plans.

4.1.2.5 Stage 5: Opportunity

The *Opportunity* stage of risk management evolution is characterized by the use of innovation and chances to save money and do better than planned. *Opportunity* is a positive vision of risk management that is used to innovate and shape the future. Risk, like quality, is everyone's responsibility. The risk ethic involves every one (program team, customer, end-user) and is a continuous process of identifying, communicating and resolving risks in an open and nonthreatening environment [Kirkpatrick92]. Professional attitudes of engineering excellence allow for open communication and individual contribution. We admit that there are things we do not know and allow for their existence using a best-case, worst-case scenario. People understand there is an opportunity cost associated with every choice, and knowing these trade-offs improves their decision making ability. Risk does not have to be negative [VanScoy92]. Wherever there is a risk there also exists opportunity [Charette91b].

Stage 5: Opportunity - Process.

In the *Opportunity* stage of risk management evolution, chances to exceed expectations and innovative ideas to bring a program in under budget or schedule are identified. Opportunities for cost savings are identified and analyzed according to quantitative evaluation criteria. Return on investment is estimated until actuals can be determined. Action plans are revised as needed to take advantage of current information. Techniques include cost-benefit analysis and decision analysis of

alternative approaches. Action plans are reviewed and corrected as required. Progress is monitored and corrective action is taken as appropriate. Written evaluations document ideas of how the process could be improved. Evaluations are analyzed and documented as lessons learned. This feedback is used to improve the process.

Stage 5: Opportunity - Infrastructure.

A *Policy* establishes rewards for innovation and ideas that save money. The culture represents a belief that "you don't know what you don't know." The attitude is that unknown risks do exist. Information regarding risks is gathered from lower levels and passed on to management, the customer, and end-user. The motivation for enhanced risk *Communication* is to provide for customer delightedness (better than satisfaction) and the highest standards of engineering excellence. Most employees have integrated risk management into their daily activities. Optimal schedule, budget and *Resources* are allocated to risk management, since metrics have been used to tune the process. *Training* provides the knowledge for understanding how to use the results of metrics to improve the process. Course evaluations are written at the end of training and used as feedback to improve training.

Stage 5: Opportunity - Implementation.

The project team, customer and end-user perform risk management activities routinely throughout the program. All participants are empowered to communicate risks. The opportunity to discuss issues, concerns, and fears with peers, managers, and customers provides an understanding and empathy for all team members. Risk

management gives permission to fail without later pointing fingers at an individual. Many risk champions exist that advocate the use of risk management. Risk management is institutionalized and a way of doing business. The Risk Management Plan is a living, on-line document. *Methods* use graphs of risk metrics that are linked to a risk database. Reports of all risks are provided to the project team, customer and end-user throughout the program. *Tools* are used to improve quality and increase productivity. Risk analysis may be automated to hide the complexities of uncertainty theory. A knowledge base of identified risks and resolution strategies exists. *Metrics* are analyzed, reported and used to improve the process.

4.2 Risk Management Capability Maturity Model

The *Risk Management Capability Maturity Model* (RM-CMM) describes an evolutionary path from an inability to manage risk to systematically controlling risk and maximizing opportunities for an organization. I developed the *Risk Management Capability Maturity Model* (RM-CMM) to provide the practices which satisfy goals that achieve the vision for the RMEF stages.

The RM-CMM provides a five stage improvement model to transform an organization's capability to assess and manage risk on software projects. Each successive stage is associated with an increase in risk management capability. The RM-CMM is organized to provide a strategy to transfer risk management technology into

an organization and institutionalize its use. The RM-CMM provides three synergistic perspectives of risk management technology that each contain five focus areas (see Figure 18).

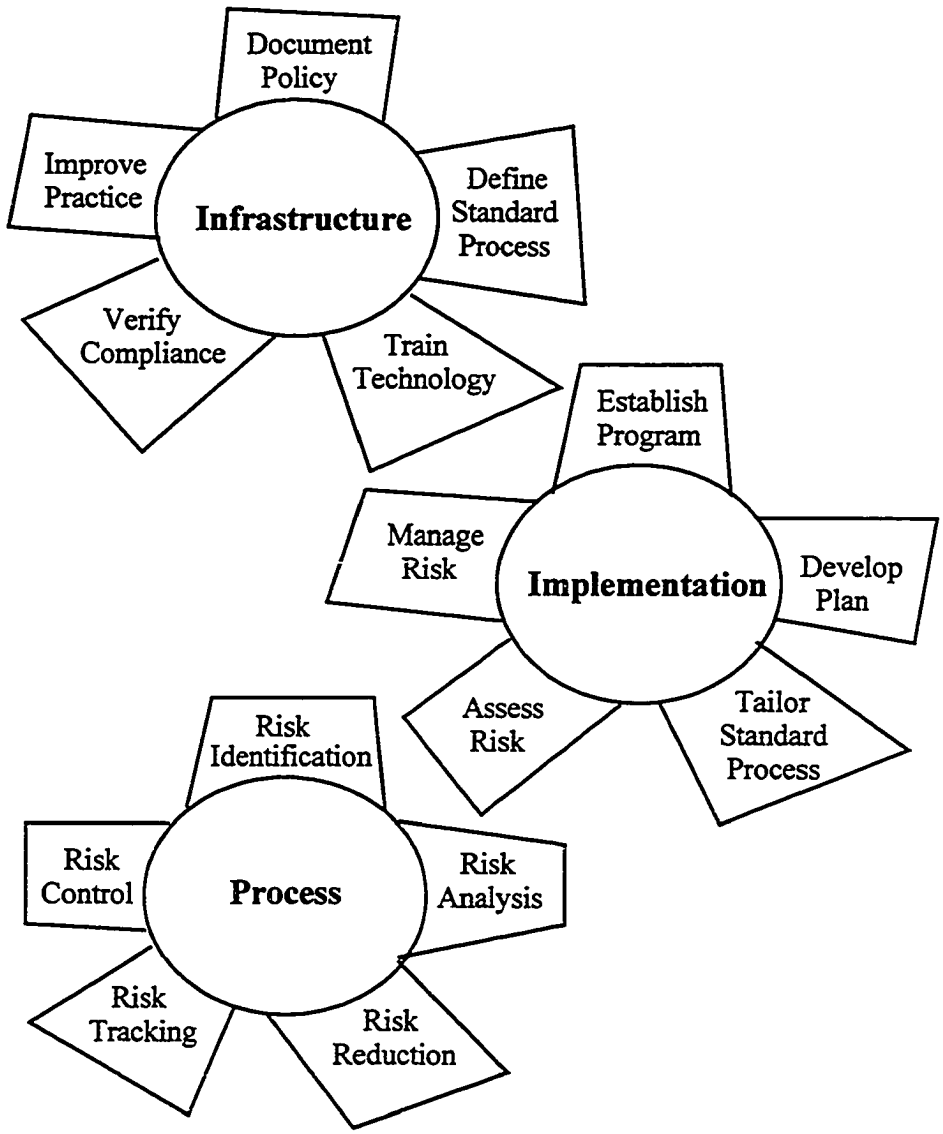


Figure 18. Risk Management Capability Maturity Model (RM-CMM)

4.2.1 Model Architecture

My inspiration for the RM-CMM was the CMM for Software [Paulk93a], which provides risk management practices in several key process areas. The need for a focus on risk management and clarification of process, infrastructure, and implementation practices led to the development of the RM-CMM. The architecture of the RM-CMM maps easily to the CMM, to leverage existing knowledge of the CMM structure. The model architecture (see Figure 19) was developed by comparing and using structures from four maturity models: the Quality Management Maturity Grid [Crosby80], the Capability Maturity Model (CMM) [Paulk93b], the Reuse Maturity Model [Koltun91] and [Stivers93], and the draft SE-CMM [Garcia94]. The draft ISO Software Process Improvement Capability Evaluation (SPICE) architecture was also reviewed.

The Risk Management Capability Maturity Model architecture contains 5 stages, 3 dimensions, 15 focus areas, and 5 - 10 key practices within each focus area. The major architectural structures of the RM-CMM are described below:

- **Stage** - a standard level of capability to measure against.
- **Dimension** - a logical abstraction of task activity and responsibility.
- **Focus Area** - a grouping of practices that perform a task to satisfy a goal.
- **Key Practice** - an observable work activity associated with a maturity stage.

Stages in the evolution of risk management provide incremental enhancements in the capability to control risk. The summation of Key Practice (KP) performance determines the degree to which each of the five stages is satisfied (see Appendix C). When a stage is fully satisfied, the maturity level is achieved. An organizational profile over time shows the progress made in each stage.

Dimensions are clusters of logically related activities categorized by process, infrastructure, or implementation. Process incorporates the definition of activities in performing risk management. Infrastructure incorporates both organizational commitment and ability to perform. Implementation incorporates the planning and procedures required to execute the defined risk management process.

Focus Areas (FA) are categorized by dimensions of process, infrastructure, and implementation, which provide a separation of responsibility and a focus for improvement. Focus Areas are ordered in a logical sequence, but may be parallel and/or iterative in practice. Within each Dimension, Focus Areas are numbered with a unique identification tag (e.g., INF.FA.01). Each Focus Area is a logical grouping of Key Practices, which are observable work activities.

Key Practices (KP) are tasks that describe WHAT must be accomplished to satisfy the Focus Area. Key Practices are ordered in a logical sequence, but may be parallel and/or iterative in practice. Within each Focus Area, Key Practices are numbered with a unique identification tag (e.g., INF.FA.01.KP.01). Each Key Practice has a brief title

and description in verb-object format which summarizes the intended result of the activity. Key Practices are separately classified according to the stage of maturity (see Appendix C) to synchronize parallel improvement efforts between Dimensions and/or Focus Areas. This maturity classification is also used in the RM-CBA, the RM-CMM based appraisal method, which is described in section 4.3.2.

Examples within the Key Practices suggest WHEN, WHO, and HOW. Examples should be provided to increase the understanding of the intent of a Key Practice. Examples are beyond the scope of this dissertation.

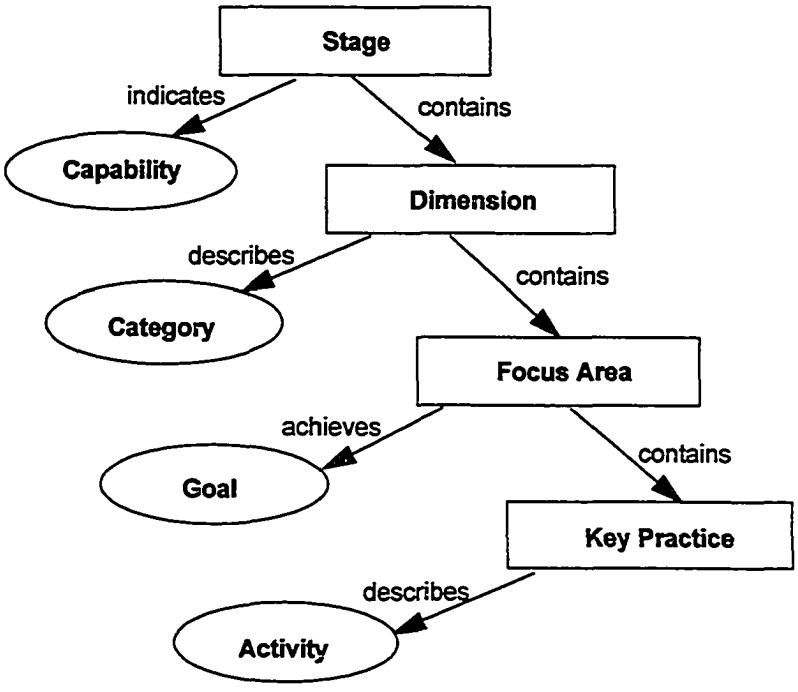


Figure 19. Risk Management Capability Maturity Model (RM-CMM) Architecture

The RMEF stages are similar to the "states" of a state transition diagram. Each stage describes the characteristics of an organization at one of five maturity stages. At the *Mitigation* stage, the RMEF describes the characteristics of an organization that has already achieved the vision established for maturity level 2. Maturity is a quantifiable level of capability achieved through risk management practice. Maturity levels establish a path for improvement and technology transfer. Dimensions of *Process*, *Infrastructure*, and *Implementation* exist at all maturity stages. Focus Areas (FA) are ordered by logical sequence, and not ordered by stage of maturity. Each Key Practice (KP) is associated with a maturity stage (see Appendix C).

The Model is the "arrows" of a state transition diagram, which describe the transitions required at a given maturity state to reach the next higher state. At level 2 (L2), the RM-CMM describes what must be accomplished to achieve the Stage 2 *Mitigation* capability. This includes activities for technology transfer, process definition, assessing, planning, implementing, measuring and controlling.

Vision guides the way to the next maturity stage. Specific goals must be accomplished to achieve the vision. Goal achievement results in realization of the

vision. Strategy provides an approach that supports, but does not guarantee goal attainment (see Figure 20).

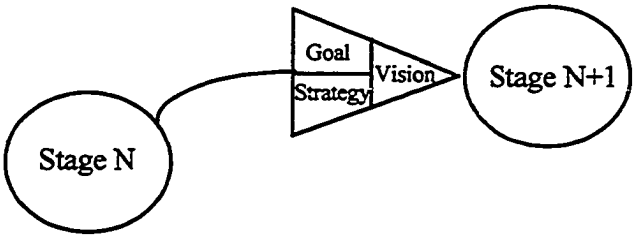


Figure 20. RM-CMM Structure for Evolving Risk Management Technology.

4.2.1.1 Vision, Goals and Strategy

Vision is an ideal state of the practice after the journey through an evolutionary stage. Vision provides direction throughout the journey. Vision is achieved indirectly by accomplishing the goals. Goals must be accomplished to achieve the vision. Strategy specifies an approach to accomplish the goals. Different strategies may be used to satisfy a goal. For each stage of risk management evolution, the goals and the corresponding strategy for their achievement are described below.

Mitigation is a vision for Stage 2. The organization's management makes a commitment to support risk management activities to increase the capability to reduce project risk. By this commitment, a competitive advantage is gained for the company stakeholders (see Table 8).

Table 8. Stage 2 - Mitigation Goals and Strategy

Goal	Strategy
Senior level management commitment is obtained to support risk management activities.	Educate senior level management on the benefits of risk management.
Risk management activities are planned.	Provide sufficient resources to develop an action plan for incorporating risk management.
A standard risk management process for the organization is developed.	Establish a group to define and document the standard risk management process.
Projects use the risk management process to assess project risk.	Provide trained and independent facilitators to help projects assess risks.

Prevention is a vision for Stage 3. Management empowers the project team to communicate technical risks. This increases the visibility into project risks, thereby increasing the capability to prevent problems and surprises which result in customer satisfaction (see Table 9).

Table 9. Stage 3 - Prevention Goals and Strategy

Goal	Strategy
Project management involves the project team in communicating technical risk.	Educate project management on the benefits of team risk management.
Risk management activities are worked according to plan.	Provide sufficient resources to perform risk management activities.
Projects define a tailored risk management process.	Establish a group to define and document the tailored risk management process.
Projects use the risk management process to analyze project risk.	Train the entire project team in the risk management process.

Anticipation is a vision for Stage 4. The project team proactively quantifies project risks to focus on critical success factors and uses metrics to systematically control risks, thereby increasing the predictability of project performance for product success (see Table 10).

Table 10. Stage 4 - Anticipation Goals and Strategy

[illegible]	Strategy
Project management commitment is obtained to support project metrics.	Educate project management on the benefits of project metrics.
Project performance is quantitatively measured.	Provide sufficient resources to measure project performance.
A standard way to measure and analyze project performance is developed.	Establish a group to define and document the measurement process.
Projects use metrics to systematically control project risk.	Train the project team in decision analysis.

Opportunity is a vision for Stage 5. The organization routinely uses optimal risk management methods and rewards for creative ideas that identify cost savings, thereby maximizing opportunities for the organization (see Table 11).

Table 11. Stage 5 - Opportunity Goals and Strategy

Goal	Strategy
The organization management empowers the workforce to practice risk management.	Establish a reward system for efforts in risk management.
Risk management is improved according to feedback, measurement and analysis.	Provide resources to evaluate metrics and improve risk management.
Projects capture effective and efficient risk management results.	Establish a repository of risk management lessons learned.
Projects use risk management to improve productivity and product quality.	Train the project team in automated risk management methods.

4.2.1.2 Process Focus Area

The *Process* dimension focuses on activities related to the tasks in performing risk management effectively and efficiently. The first step is to define the process of identifying risk and the source of risk. The next step is to define the process of analyzing the impact and likelihood of the risk to determine risk ranking. The process of developing and executing a risk reduction plan is then defined. The process of capturing, reviewing and reporting risk status is also defined, along with mechanisms for responding to triggering events, correcting for variations from plans, and process improvement.

Focus Areas (FA) for the *Process* dimension are:

- (PRO.FA.01) Risk Identification
- (PRO.FA.02) Risk Analysis
- (PRO.FA.03) Risk Reduction
- (PRO.FA.04) Risk Tracking
- (PRO.FA.05) Risk Control

4.2.1.3 Infrastructure Focus Area

The *Infrastructure* dimension focuses on activities related to establishing an environment in which the risk management process can be defined, communicated, monitored, and improved. The first step is to obtain commitment and define a policy for performing risk management that is communicated to the entire organization. The next step is to define a standard risk management process that provides a consistent process that is shared across the organization. Training is essential to raise the awareness and understanding of risk management, which provides the motivation and ability to perform the risk process. An independent audit of the risk management activities, training, and project performance is required to verify risk management compliance. Risk management practice is then systematically improved by assessing the risk management capability and developing and implementing improvement action plans to ensure continuous improvement.

The Focus Areas (FA) for the *Infrastructure* dimension are:

- (INF.FA.01) Document Policy
- (INF.FA.02) Define Standard Process
- (INF.FA.03) Train the Technology
- (INF.FA.04) Verify Compliance
- (INF.FA.05) Improve Practice

4.2.1.4 Implementation Focus Area

The *Implementation* dimension focuses on activities related to the execution of the risk management process to cost-effectively control risks. The first step is to establish a risk management program by reviewing requirements from the customer and organization. Planning for risk management activities requires allocating schedule, budget, and staff. The next step is to develop a Risk Management Plan which details the approach, structure, process, methods, tools, and metrics used to implement risk management on the project. The risk management process may be tailored from the organization's standard process by addressing the unique aspects of the project, such as size, budget, and structure to custom fit a cost-effective process to the project. Risks are iteratively assessed and managed to control the project risk.

The Focus Areas (FA) for the *Implementation* dimension are:

- (IMP.FA.01) Establish Program
- (IMP.FA.02) Develop Plan
- (IMP.FA.03) Tailor Standard Process
- (IMP.FA.04) Assess Risk
- (IMP.FA.05) Manage Risk

4.3 Risk Management Capability Appraisal Method

There are various reasons for assessing an organization's risk management capability. The rationale for using a structured capability appraisal method is:

- To establish a baseline for improvement.
- To develop a plan for improvement.
- To measure progress against an improvement plan.
- To select a contractor or subcontractor.
- To monitor project performance.

4.3.1 Risk Management Survey

The Risk Management Survey (RMS) determines the perceptions of risk management practices to understand organization/project strengths and weaknesses. The RMS uses a Likert normed response on a five point scale to capture perceptions of risk practices with respect to their performance and importance on a specific project. The result of administering the RMS is a quantitative measure of the state of the practice in risk management for an organization/project that is based on the RMEF. Over time, RMS results characterize progress and trends in performing risk management. The Risk Management Survey (RMS) is provided in Appendix A.

4.3.2 Risk Management Model Based Appraisal

The Risk Management Capability Maturity Model Based Appraisal (RM-CBA) is a method that is inexpensive, unbiased, and easy to use. RM-CBA uses the RMS, a survey based on the RMEF, to evaluate perceived performance and importance of risk management practices. An interview with each project's program manager is used to obtain data to characterize the project in terms of size, structure, and application domain. To develop an improvement plan, the RM-CBA uses the RMS results and the RM-CMM. Some features of the RM-CBA include:

1. ***Shared vision.*** The RM-CBA involves all risk management participants and organization management to promote commitment, buy-in, and empowerment of the workforce.
2. ***Quantitative.*** Responses to the RMS are entered into a spreadsheet that automatically graphs strengths, weaknesses, and areas for improvement. Progress is easily tracked by applying the RMS at regular time intervals.
3. ***Unbiased.*** Results of the RMS are the collective knowledge and experience of the project teams and the organization management. Responses are categorized by project and organization role, so that they may be compared which serves as a check and balance system to ensure integrity and helps to avoid "gaming."
4. ***Model based.*** The RM-CBA is based on the RM-CMM, a known maturity model, which serves as a standard that enables comparison between organization/project RM-CBA results.

This chapter describes a risk management maturity model and appraisal method for improving the capability to manage risks in software development. Chapter 5, *Proactive Risk Management*, applies the RM-CBA method for test and evaluation.

CHAPTER 5

Proactive Risk Management

Cost-effective risk management methods that increase software project quality are described by my proactive approach to risk management. In this chapter, proactive risk management methods are characterized by stages of risk management maturity. An evolutionary migration strategy for transitioning risk management technology into an organization is developed by assessing an organization's risk management maturity using my *Risk Management Capability Maturity Model Based Appraisal (RM-CBA)* method. The results of applying the RM-CBA method were used to improve the *Risk Management Evolutionary Framework* (RMEF), and refine the *Risk Management Capability Maturity Model* (RM-CMM).

5.1 Proactive Risk Management Methods

If you don't actively attack the risks, they will actively attack you [Gilb88]. The majority of potential problems on software intensive projects can be managed proactively to reduce rework and other obstacles to successful software delivery. Proactive risk management is the opposite of reactive crisis management. Proactive risk

management is taking the action required to identify, assess, and manage risks to prevent problems on software projects. Proactive risk management helps projects succeed by providing them with tools for more informed decision-making and improved communication. Proactive risk management corresponds to the *Risk Management Evolution Framework* (RMEF) stages of *Prevention*, *Anticipation*, and *Opportunity*. Evolutionary methods are used for risk assessment and risk management depending on the current level of risk management maturity. Preventive (stage 3) methods use peer review, nominal group technique, and cause/effect diagrams. Quantitative (stage 4) methods use cost/benefit analysis, decision theory, and technical performance measures (TPM). Opportunistic (stage 5) methods incorporate a knowledge base of mitigation approaches, graphs of metrics, and rewards for innovation and cost savings.

A proactive approach to risk management begins by assessing the risk management capability using the RM-CBA and following the RM-CMM to develop an improvement plan. The following sections detail the proactive approach to risk management for each major dimension of risk management technology at the *Prevention* stage of maturity. The *Prevention* stage of risk management evolution is characterized by team and customer involvement. A focus on technical risks leads to discovery of the source of risk. *Prevention* is a transitional stage and turning point from a reactive to proactive approach to risk management. *Prevention* evolves from avoidance of risk symptoms to attempts at eliminating root causes.

5.1.1 Process Methods

In the *Prevention* stage of risk management, the process supports identification of the sources of risk throughout the project. Risks are assessed using subjective evaluation criteria. Prioritization of all analyzed risks occurs by group consensus. The proactive risk management methods that support the process dimension at this stage are described below (see Table 12).

Table 12. Proactive Process Methods for Stage 3 Prevention

Process	Proactive Risk Management Method
Identify	Risk Assessment, Risk Management Form, Risk Taxonomy, Cause/Effect Diagram
Analyze	Subjective Evaluation Criteria, Nominal Group Technique
Plan	Risk Reduction Template
Track	Risk Database, Top-10 Risk List
Control	Process Improvement Form, Risk Management Survey

5.1.2 Infrastructure Methods

In the *Prevention* stage of risk management, the infrastructure's policy requires that a risk management process be used on software intensive projects. Training provides an understanding of the risk management process. Risks are gathered from lower levels and passed on to management. The proactive risk management methods that support the infrastructure dimension at this stage are described below (see Table 13).

Table 13. Proactive Infrastructure Methods for Stage 3 Prevention

Infrastructure	Proactive Risk Management Methods
Policy	Peer Review, On-line documents
Communication	Newsletter article, Cascaded communication
Commitment	Memo from top management
Resources	Master schedule, Management reserve
Training	Learning Lunch for the project team

5.1.3 Implementation Methods

In the *Prevention* stage of risk management, the implementation is performed by the project team with a single risk champion. A risk management plan documents the procedures for performing risk management. Metrics are defined and collected. The proactive risk management methods that support the implementation dimension at this stage are described below (see Table 14).

Table 14. Proactive Implementation Methods for Stage 3 Prevention

Implementation	Proactive Risk Management Methods
Participants	Risk Champion
Procedures	Peer Review, On-line Risk Management Plan
Methods	Risk Appraisal
Tools	Spreadsheet software
Metrics	Risk Management Index

5.2 Evolutionary Migration Strategy

I prepared an evolutionary migration strategy as a proof of concept for determining an organization's risk management capability and improvement plan using the RM-CBA method. This section describes the strategy for assessing risk management maturity to provide a baseline for measuring progress. The capability assessment also factors into the improvement plan for evolving risk management

technology within an organization. An improvement plan was developed for a specific organization[2] as part of the strategy to evolve their risk management capability.

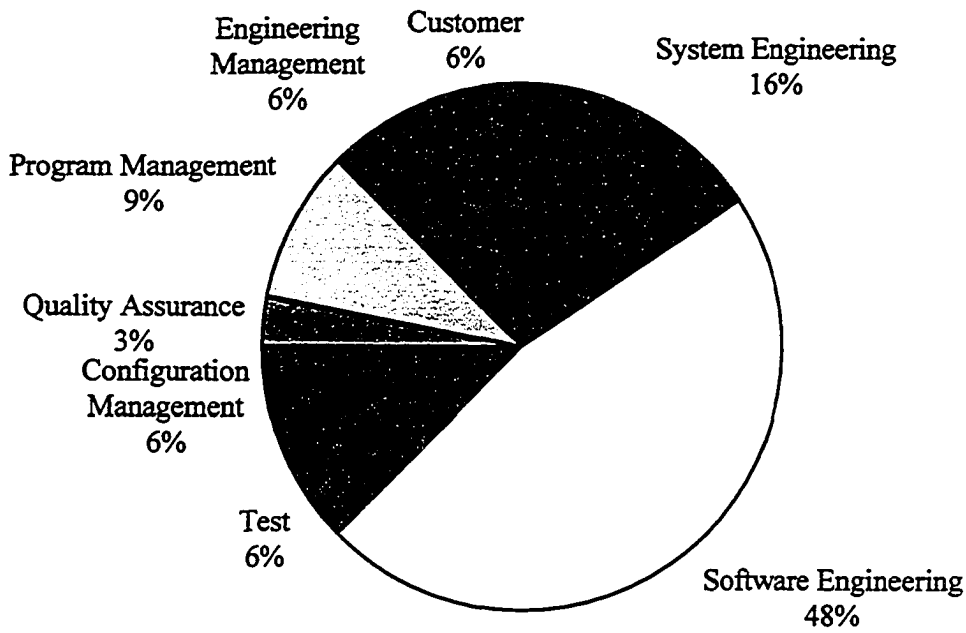
5.2.1 Capability Assessment

Three large software intensive projects were selected to represent the selected organization's state of the practice in risk management. DoD and other government customers, a prime contractor, and subcontractors, the contract monitoring group, and the Software Engineering Institute (SEI) participated through a series of independent risk assessments and a subsequent Risk Management Survey (see Appendix A). The risk assessments provided a baseline of assessed risks that were managed by the projects. Several months later, I interviewed the three Program Managers. I asked them to describe their use of risk management and approve the survey for distribution and collection. To maintain anonymity, distribution and collection of the surveys was performed by the program manager's secretary. I designed the Risk Management Survey (RMS) to assess the state of the practice in risk assessment and risk management. This was achieved by asking survey participants to identify their perceptions of the performance and importance of risk practices on the project. The survey participant's roles are shown as a percentage of the total surveyed (see Chart 1).

---

2. The organization was self-assessed at SEI Level 3, and is registered ISO-9001.

Chart 1. Risk Management Survey Participants

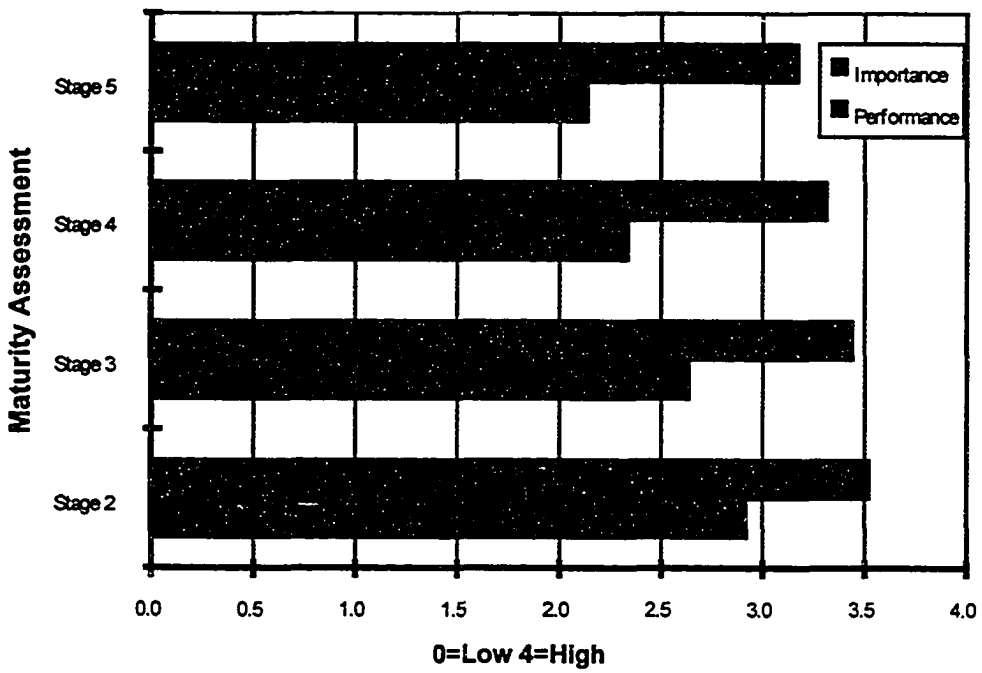


5.2.1.1 Results

All risk management practices were graphed according to the RMEF five stage improvement model to show perceived performance and importance. RMS section II, *Risk Management Practices* (questions 21 - 91), correspond to a maturity stage and essential element within the RMEF. A lookup table is provided for mapping the RMS question number to the RMEF (see Appendix D). Survey responses were entered into an Excel spreadsheet that was developed to graph the survey data. One view of the survey data combines all risk management practices at each maturity stage. For the

three project's surveyed, the risk management maturity falls short of stage 2, although progress is shown at all levels. Risk management is perceived as important in the organization (see Chart 2).

Chart 2. Risk Management Maturity



5.2.1.2 Analysis

Analysis of survey data included scaling the quantitative data to fit a normal distribution. The open-ended questions in RMS section III, *Observations*, were grouped for ease of project comparison. A gap analysis was performed by grouping practices

according to the essential elements of the RMEF, and plotting relative importance versus performance.

Scaling Data.

Survey responses from 0 to 4 provide an ordinal ranking, but do not measure the distance between the data points. To enable metrics and statistical comparisons, the data was scaled to fit a normal distribution. Scaling was performed by mapping the ordinal scale 0 to 4 to fit a standard normal curve with a mean of 0 and a standard deviation of 1. The percentage for each slice was determined by counting the total number of responses for each score and dividing by the total number of responses. The scaled value corresponding to each score from 0 to 4 was found by determining the mean for each slice. After the data was transformed, the scaled values were determined (see Table 15). The adjusted or normalized scores were substituted for the raw scores for subsequent analysis.

Table 15. Data Transformation for Metrics Analysis

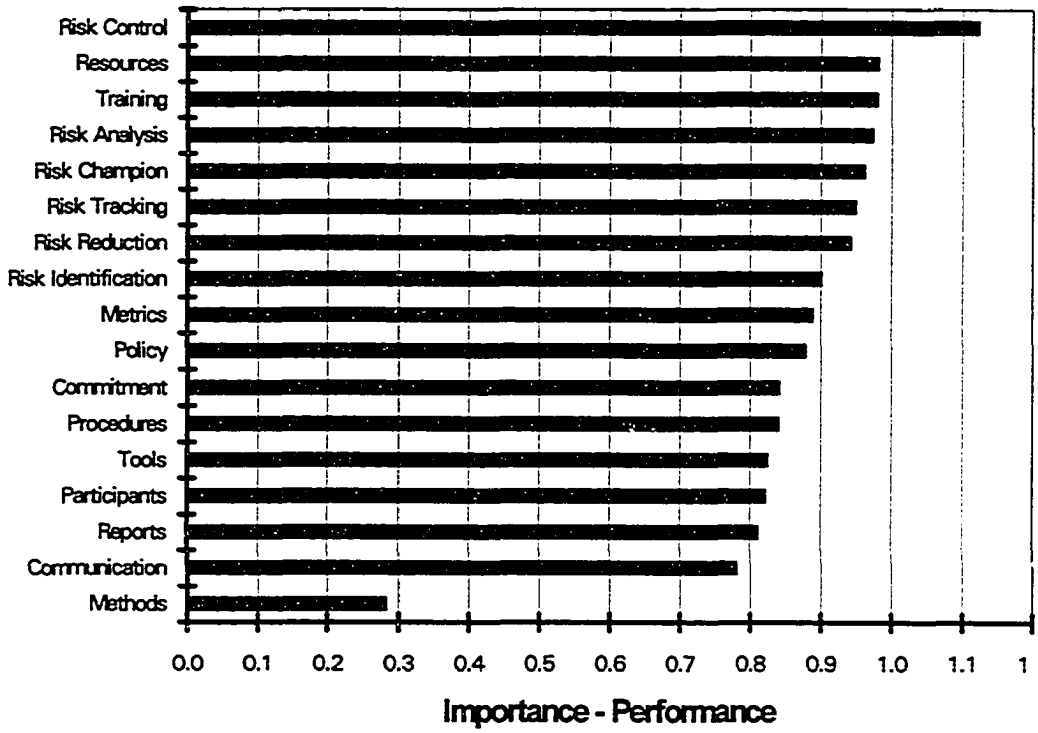
Ordinal Scale	0	1	2	3	4
Scaled Values	0.2	1.2	2.0	3.0	4.3

Gap Analysis.

Gap analysis was performed by normalizing the scores by scaling the raw data and subtracting performance from importance for each respondent. This subtracts out the bias for each person, since presumably the same bias exists in their performance and

importance score. This provides a better standard of deviation, without bias. The difference for each person is summed and divided by the total number of respondents. The essential elements were sorted by magnitude of the difference. Gap analysis shows that the organization is not performing at the level of importance in all areas except *Methods*. The small gap for *Methods* is caused by the use of the Risk Taxonomy, a structured method for risk identification, transitioned to the organization by the SEI. The largest gap shown is in *Risk Control* (see Chart 3).

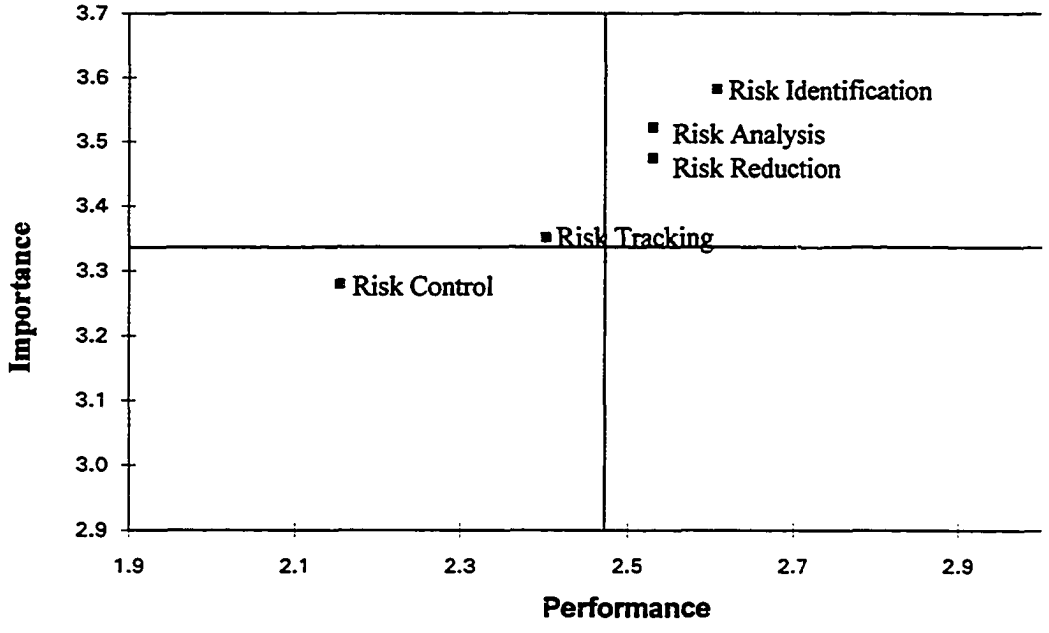
Chart 3. Risk Management Gap Analysis



Importance vs. Performance.

Plotting relative importance versus performance by essential element of the RMEF and the mean importance and performance provides four quadrants that categorize risk management practices. The quadrants show relative strengths and weaknesses and may be used to identify areas for improvement. For a more detailed comparison, essential elements were grouped by dimension and plotted using the mean for that data set. In the graph below, *Risk Identification* falls in the quadrant describing high importance and high performance, while *Risk Control* falls in the quadrant describing low importance and low performance. Note that all elements of the *Process* dimension rated over 3.0 in importance, while none rated over 3.0 in performance (see Chart 4).

Chart 4. Risk Management Process Elements Importance vs. Performance



5.2.1.3 Findings

In order to use survey results and analysis to improve risk management practices, meaningful conclusions must be derived and translated into recommendations. Data from Program Manager and Risk Champion interviews, together with open-ended survey questions, were used to substantiate organization strengths, weaknesses, project needs, and lessons learned regarding risk management.

Observations.

The following quotes summarize major issues found in the organization's risk management practices:

> "I believe the process is in place to make risk management effective, but I don't feel the benefit has been communicated."
>
> "Since the people don't see direct benefit and they are pressed with other duties, the risk management is seen as a hoop to jump through. This has resulted in risks not being presented because the process levies what is seen as extra work."
>
> "I think we were able to avoid some major problems by using risk management."
>
> "Don't do software risk management; do project-wide risk management. Segmenting software does not make sense unless all you want is an SEI check mark."

Strengths.

Organizational strengths were observed in the following areas:

- Reports, Commitment, and Risk Identification
- Structured process in place
- Tracking top risks
- Risk Database
- Risk Management Form.

Weaknesses.

Organizational weaknesses were observed in the following areas:

- Risk Champion, Training, and Risk Tracking
- Process levies "extra work"
- Isolated activity
  - Management-only practice
  - Lack of communication to team
  - Lack of buy-in by Program Management
- Upper management doesn't track/mitigate.

Project Needs.

Project's described needs in the following areas:

- Improved communication/feedback
- Actions to decrease risk
- Distributed risk management
- Administrative support
- Independent evaluation
- Automated tools and tracking.

Lessons Learned.

Project's described the following lessons learned:

- Use of risk management avoids major problems
- Focuses on important problems earlier in lifecycle
- Must be proactive
- Identifies more than software risks
- Identifying is not enough
- Some risks don't go away
- Proceeding at risk is risky.

5.2.2 Improvement Plan

Recommendations were developed for the organization based on the assessment results. An improvement plan for evolving the organization's risk management capability was developed based on the assessed risk management capability, the set of recommendations, and the RM-CMM. The RM-CBA was also improved through modifications that were made to the RMEF, RMS, and RM-CMM based on the evaluation of the pilot study.

5.2.2.1 Recommendations for the Organization

Several recommendations were addressed through a debrief of the survey results to the SEPG, Software Process Team (SPT), and the organization Steering Committee. Benefits of risk management were also communicated through these briefings. Based on my evaluation on the survey results, I made the following recommendations to the organization's SEPG and Steering Committee:

- Feedback survey results to the projects
- Address the project's identified needs
- Communicate the benefit of risk management
- Provide rewards for risk management
- Do program-wide risk management.

Assessment recommendations and the RM-CMM were then reviewed to determine the next steps to evolving risk management in the organization. The organization risk management capability and the vision, goals, and strategy for achieving the next stage of evolution were also considered. Available budget, time, and staffing were factored into the development of the improvement plan.

The RM-CMM was reviewed for the *Implementation* focus area, since the improvement plan being developed is targeted for a pilot project within the organization. The project has an established program and plan for risk management. Their process was tailored from the organizational risk management process. The

improvement plan addresses identified weaknesses, project needs, and transitions proactive methods for iteratively assessing and managing risks.

Since the organization risk management capability that I evaluated is close to stage 2, the vision, goals, and strategy for achieving stage 3 were factored into the improvement plan. The proactive risk management methods that support the implementation dimension at stage 3, *Prevention*, were reviewed and factored into the improvement plan. To achieve the vision of empowerment of the project team to identify risk and the source of risk, the first goal of stage 3, *Prevention*, was reviewed:

- Project management involves the team in communicating technical risk.

The strategy for accomplishing this goal is determined by reviewing the RM-CMM strategies for stage 3 and the Risk Management Capability Cause/Effect Diagram. The strategy should consider:

- Educating project management on the benefits of team risk management.
- Ability of project team to perform risk assessment.
- Motivation of project team to perform risk management.
- Presenting the improvement plan to the organization's management to obtain support for its implementation.

5.2.2.2 Modifications to the Method

Several improvements to the RM-CBA were made as a result of the pilot assessment. Two essential elements of the RMEF were deleted after development of the RMS and RM-CMM. One essential element, *Culture*, was removed from the *Infrastructure* dimension. I realized that a statement regarding culture was inappropriate for the RMS because it was not a tangible risk management practice. Another essential element, *Reports*, was removed from the *Implementation* dimension. In my opinion, *Reports* could be subsumed under the generic category that *Methods* provides. An Excel spreadsheet was developed to easily tabulate and report the data by project, organization, and by the organization's management.

5.2.2.3 Pilot Test Evaluation

The pilot test involved many software engineers and related engineering disciplines to determine the risk management capability of the organization. Each of the three projects surveyed had different implementations for their risk management practices. Despite these differences, there was a common theme to the effective and ineffective practices used on the projects (see Appendix B, *Risk Management Survey Results*). The RMS was found to be a cost-effective method for measuring risk management capability.

CHAPTER 6

Conclusion

6.1 Summary of Results

The origins of risk management on software projects have been described to establish the role of risk management in software engineering. The barriers to adoption determined for risk management technology are the immaturity of the technology itself, low customer expectations, and organizational inhibitors such as resistance to change and lack of management commitment. Critical success factors for risk management capability are the project, people, process, and procedures. The software *project* establishes the requirements and resources for the use of risk management. The participation of the *people* is key to identifying the risks that must be managed for success. The risk management *process* definition and execution must be tailored to a project for maximum effectiveness. The *procedures* of performing risk management are important because they document the organizational structure and approach for using risk management.

To understand how risk management capability evolves, I developed and tested the *Risk Management Evolution Framework* (RMEF), a five stage improvement model

for risk management technology. I developed the *Risk Management Capability Maturity Model* (RM-CMM) based on fundamental principles of quality, maturity, and technology transfer to outline the path to increasing risk management capability. I defined and evaluated proactive risk management methods through a pilot test on a Software Engineering Institute (SEI) Level 3 and ISO-9001 registered organization. The organization's risk management capability was assessed using the *Risk Management Capability Maturity Model Based Appraisal* (RM-CBA), a method I developed based on the RM-CMM. The *Risk Management Survey* (RMS) I designed as a component of the RM-CBA, was used to provide a quantitative baseline that measured strengths, weaknesses, and areas for improvement. An evolutionary migration strategy for transitioning risk management technology was developed for the organization.

6.2 Principles of Risk Management

In this section, I summarize the basic concepts that I believe provide a foundation for software risk management. Fundamental to the notion of risk are concepts of time, uncertainty, choice, and loss. These concepts are embodied in the principles of risk management, which distill the essence of software engineering excellence. These basic truths of risk management applied to software engineering are:

- **Diversify** - There are no silver bullets, so don't put all your eggs in one basket. Don't rely heavily on one customer, vendor, method, tool, or person to fulfill your project needs. Instead, build a balanced approach that stresses mastery of software project fundamentals.
- **Leverage** - Major risk reduction leverage exists in the early phases of software development. Since time is money, early detection of risks reduces rework costs. Use the concept of leverage to focus on project critical success factors. Use time management and the Pareto principle to focus on the important risk consequences that impact success.
- **Synergy** - Together Everyone Achieves More! Use team building and cascading communication to develop an understanding of the project risk from the top-down, and bottom-up. Communication increases understanding, which creates a whole greater than the sum of the parts.
- **Proactivity** - Take the initiative, don't wait for someone else to do it. The rule "no over the wall" eliminates the risk of wasting time waiting for someone else to finish work. Risk management is a proactive approach to preventing problems on software projects.
- **Creativity** - Use structured brainstorming to freely express ideas. Think about the possibilities and ask the tough questions. Use paradigms of continuous improvement and innovation. Changing paradigms may seem risky, but therein lies opportunity.

6.3 Potential Solutions

What can the software community do to help software projects manage their risks? The solution involves academic foundations in software engineering, government expectations for professionalism in the field, and industry awareness with commitment to use risk management on software projects.

Solutions within the academic community include teaching quantitative risk management techniques and addressing software risks in the software engineering curriculum. Basic concepts and tools should be used to provide the student with a working knowledge of managing risk to an acceptable level.

Solutions within the government include training acquisition agencies on what to require in performing risk management on software programs. Expectations that government contracts will use risk management methods should be clearly communicated to the software industry. Incentives for award fees could be used as a motivator.

Solutions within the software industry include defining risk management practices as standards of the software engineering discipline. Establishing a professional certification for software engineers that incorporates risk management principles would also increase professionalism of software engineers.

6.4 Software Engineering Challenge

Knowing our risks provides opportunities to manage and improve our chances of success [VanScoy92]. The genesis for risk management in software engineering is history. The journey in risk management evolution on software projects has begun. The challenge is to use the *Risk Management Evolution Framework* (RMEF) to understand the essential elements of risk management, to ascertain your present position in risk management maturity, and see where your direction is taking you. The *Risk Management Capability Maturity Model* (RM-CMM) should be used to evolve risk management technology in software organizations to succeed in a global economy. Positive expectations when thinking about the future exercises the power of vision. The ultimate in software risk management combines leadership and an empowered workforce to maximize opportunities for a project, an organization, a nation, and the world.

6.5 Future Risk Management Research

Future research in risk management can use the survey instruments presented in this dissertation and contrast data collected from various roles, projects, and organizations. Mitigation strategies for well-known software risks could be described and tested. Examples are needed for each Key Practice (KP) of the RM-CMM to describe WHEN, WHO, and HOW.

6.6 Beyond Risk Management

What lies beyond recognizing opportunity and managing risk? Within our conscious mind, I call this *Possibility Thinking*. Possibility thinking stretches our minds and is used to think of anything that may exist, depending on the circumstances. Possibility thinking is based on probability theory, creativity theory, risk management, and philosophy. Knowledge and wisdom are used to make decisions. We meet difficulty with calmness and composure. At the highest levels of possibility thinking, we create opportunity by managing our subconscious. *Possibility Thinking* has already been described in psychology [Dyer91] and religion [Schuller93]. I believe that concepts for possibility thinking can be applied in an engineering context, and that they extend from the highest levels of the *Risk Management Evolution Framework.*

Works Cited

[AFMC93] Air Force Material Command. *Software Development Capability/Capacity Review.* June 1993.

[AFSC88] Air Force Systems Command. *Software Risk Abatement.* AFSC/AFLC Pamphlet 800-45, 1988.

[American85] *The American Heritage Dictionary.* Second College Edition, Boston: Houghton Miflin Company, 1985.

[ASQC87] American Society for Quality Control, 1987.

[Babel90] Babel, Philip S. *Software Development Integrity Program* (video). Software Productivity Consortium, March, 1990.

[Boehm88a] Boehm, B. "A Spiral Model of Software Development and Enhancement." *IEEE Computer*, Vol. 21, No. 5, May 1988, pp. 61-72.

[Boehm89] Boehm, B. *IEEE Tutorial on Software Risk Management.* New York: IEEE Computer Society Press, 1989.

[Boehm91] Boehm, Barry W. "Software Risk Management: Principles and Practices." *lEEE Software*, 8,1 (January 1991):32-41.

[Brassard88] Brassard, Michael. *The Memory Jogger*™. GOAL/QPC, Methuen, MA., 1988.

[Brooks75] Brooks, F.P. *The Mythical Man-Month.* Addison-Wesley, 1975.

[Brooks87] Brooks, F.P. "No Silver Bullet: Essence and Accidents of Software Engineering." *IEEE Computer, 20,4* (April 1987), 10-19.

[Carr93] Carr, Marvin, Suresh Konda, Ira Monarch, Carol Ulrich, and Clay Walker. *Taxonomy Based Risk Identification.* (CMU/SEI-93-TR-6). Pittsburgh, Pa.: Software Engineering Institute, Carnegie Mellon University, 1993.

[Charette88] Charette, Robert N. *Software Engineering Risk Analysis and Management.* New York: McGraw-Hill, 1988.

[Charette90] Charette, R N. *Application Strategies for Risk Analysis.* New York: Multiscience Press, 1990.

[Charette91a] Charette, RN. "Information Technology Risk Engineering." SEI/NSIA workshop on Software Risk, Carnegie Mellon University, Software Engineering Institute, Pittsburgh, PA, February 1991.

[Charette91b] Charette, Robert N. "Engineering Risk to Maximize Commercial Opportunities." Itabhi Corporation, Software Tools Conference, Wembley, England, June 1991.

[Charette92b] Charette, Robert N. "Building Bridges over Intelligent Rivers." Itabhi Corporation, *American Programmer*, September 1992, Vol.5 no. 7.

[Charette93] Charette, Robert N. "Essential Risk Management: Notes From the Front." Itabhi Corp., 2nd SEI Conference on Risk Management, Pittsburgh, Pa., 1993.

[Clemen91] Clemen, Robert T. *Making Hard Decisions an Introduction to Decision Analysis.* PWS-KENT Publishing Company, 1991.

[Crosby80] Crosby, Phil. *Quality is Free.* McGraw-Hill, New York, N.Y., 1980.

[Cutler 89] Cutler, R. "A comparison of Japanese and U.S. high-technology transfer practices." *IEEE Transactions on Engineering Management*, 36, 1 (Feb. 1989), 17-24.

[Debou93] Debou, Christophe, Norbert Fuchs, and Heinz Saria. "Selling Believable Technology." *IEEE Software*, November, 1993, pp. 22-27.

[DoD88] Department of Defense Military Standard. *Defense System Software Development.* February 29,1988, DOD-STD-2167A.

[DSMC83] Defense Systems Management College. *Risk Assessment Techniques*. Fort Belvoir, VA., July 1983.

[DSMC86a] Defense Systems Management College. "Risk Management: Concepts and Guidance." Fort Belvoir, Va: DSMC, 1986.

[DSMC91] Defense Systems Management College. "Executive Forum For Systems Transition, Managing Software Technical Risk." *Program Manager,* 19, May/June 1991.

[Dyer91] Dyer, Dr. Wayne. *The Universe Within You.* Nightingale-Conant Corporation, Chicago, Illinois, 1991.

[Fichman93] Fichman, Robert G., and Chris F. Kemerer. "Adoption of Software Engineering Process Innovations: The Case of Object Orientation." *Sloan Management Review*, Winter, 1993. pp. 7-22.

[Garcia94] Garcia, Suzanne M. "Process Areas of the SE-CMM Workshop Release." Systems Engineering Capability Maturity Model Project, Carnegie Mellon University, June, 1994.

[Garey79] Garey, Michael R., and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Bell Laboratories, Murray Hill, N.J., W.H. Freeman and Company, N.Y., 1979.

[Giarratano89] Giarratano, Joseph, and Gary Riley. *Expert Systems Principles and Programming*. PWS-KENT Publishing Company, Boston, MA., 1989.

[Gilb88] Gilb, T. *Principles of Software Engineering Management*. Addison-Wesley, 1988.

[Hall92] Hall, Elaine. "Software Risk Management." *ISD Images*, Information Systems Division, Harris Corporation, November, 1992.

[Hall93] Hall, Elaine M. "A Pro-Active Approach to Software Risk Management." *Proceedings of the 1993 Harris Engineering/Manufacturing Seminar*, Harris Corporation, Melbourne, Florida, January, 1993.

[Hall94] Hall, Elaine M. "Evolution of Essential Risk Management Technology." *Proceedings of the Third SEI Conference on Software Risk*, April, 1994.

[Hall95] Hall, Elaine M., Gary P. Natwick, F. Carol Ulrich, and Charles B. Engle, Jr. "Streamlining the Risk Assessment Process." *Proceedings of the Seventh Software Technology Conference*, Hill AFB, Salt Lake City, Utah, April, 1995.

[Hetzel90] Hetzel, Bill, and Rick Craig. *Software Measures and Practices Benchmark*, Software Quality Engineering, TR - 900, December 1990.

[Higuera94] Higuera, Ronald P., Audrey J. Dorofee, Julie A. Walker, and Ray C. William. *Team Risk Management: A New Model for Customer-Supplier Relationships*. (CMU/SEI-94-SR-5). Pittsburgh, PA.: SEI, CMU, July 1994.

[Hudson92] Hudson, Anita. "Technology Transition & Adoption." *Excel: Harris Engineering Productivity Group Newsletter*, Vol. 3, Issue 2, November, 1992.

[Humphrey87a] Humphrey, W. S. *Characterizing the Software Process: A Maturity Framework*. CMU/SEI-87-TR-11, ADA 182895, Software Engineering Institute, June 1987.

[Humphrey89] Humphrey, W.S. *Managing the Software Process*. Reading, MA: Addison-Wesley, 1989.

[IEEE88] IEEE Standard for Software Project Management Plans, ANSI/IEEE STD 1058.1-1987, October 1988.

[IEEE90] IEEE Standard Glossary of Software Engineering, IEEE-STD-610.12, 1990.

[ISO91] ISO 9000-3, American National Standards Institute(ANSI) International Standards Organizations(ISO). *Quality Management and Quality Assurance Standards*. First edition, 1991-06-01, Reference number ISO 9000-3:1991(E).

[Kirkpatrick92] Kirkpatrick, R.J., J.A. Walker, and R. Firth. "Software Development Risk Management: An SEI Appraisal." *1992 SEI Technical Review*, R.L. Van Scoy, ed. SEI: Carnegie Mellon University, Pittsburgh, PA, 1992.

[Koltun91] Koltun, Phil. "Lighting the Way to Reuse." *Harris Excel*. Harris Corporation, Melbourne, FL., October, 1991.

[Koltun92] Koltun, Philip, and Pedro Martinez. "Software Process Assessment Activities at Harris." Harris Corporation, Melbourne, FL., 1992.

[Korson92] Korson, Timothy D., and Vijay K. Vaishnavi, "Managing Emerging Software Technologies: A Technology Transfer Framework." *Communications of the ACM*, September, 1992, Vol.35, No. 9, pp. 101-111.

[Luger89] Luger, George F., and William A. Stubblefield. *Artificial Intelligence and the Design of Expert Systems*. Benjamin/Cummings Publishing Co., Inc., 1989.

[Naur69] Naur, P., and B. Randell (eds.). *Software Engineering: A Report on a Conference sponsored by the NATO Science Committee*. NATO, 1969.

[Paulk93a] Paulk, Mark C., Bill Curtis, Mary Beth Chrissis, and Charles V. Weber. *Capability Maturity Model for Software, Version 1.1*. Technical Report CMU/SEI-93-TR-24, Software Engineering Institute, February, 1993.

[Paulk93b] Paulk, Mark C., Charles V. Weber, Suzanne M. Garcia, Mary Beth Chrissis, and Marilyn W. Bush. *Key Practices of the Capability Maturity Model, Version 1.1*. Technical Report CMU/SEI-93-TR-25, SEI, February, 1993.

[Paulk93c] Paulk, Mark C., Bill Curtis, Mary Beth Chrissis, and Charles V. Weber. "Capability Maturity Model, Version 1.1." *IEEE Software*, vol. 10, no. 4, (July 1993): 18-27.

[Plenum81] *Risk Analysis*, Society for Risk Analysis, 1981.

[Pressman92] Pressman, Roger S. *Software Engineering A Practitioner's Approach*. McGraw-Hill, Inc., 1992.

[Roe89] Roe, Robert A. et al. "Bugs in the Program: Problems in Federal Government Computer Software Development and Regulation." U.S. Government Printing Office, Washington, September 1989.

[SCE94] Members of the CMM-Based Appraisal Project. *Software Capability Evaluation (SCE) Version 2.0 Implementation Guide*. SEI, CMU, Technical Report CMU/SEI-94-TR-05, Pittsburgh, PA. February 1994.

[Schuller93] Scheuller, Robert H. *Possibility Thinking*. Thomas Nelson Publishers, Nashville, Tennessee, 1993.

[SEI91] Proceedings of the Annual SEI Software Risk Conferences, Software Engineering Institute, Carnegie Mellon University, 1991-1994.

[SEI93] Software Engineering Institute, Carnegie Mellon University, 1993.

[Sisti94] Sisti, Francis J., and Joseph Sujoe. *Software Risk Evaluation Method.* (CMU/SEI-94-TR-19). Pittsburgh, PA.: Software Engineering Institute, Carnegie Mellon University, October 1994.

[Stivers93] Stivers, Ben. *Reuse Maturity Model.* Harris Data Services Corporation, Montgomery, AL., June, 1993.

[SWTS92] Software Technology Process Management Support, 1992.

[Travalent94] Travalent, John J., and Elizabeth A. Hubbard. "Implementing Team Risk Management: A Case Study and Lessons Learned." *Proceedings of the Sixth Software Technology Conference*, April 1994.

[VanScoy92] VanScoy, R. *Software Development Risk: Problem or Opportunity.* Technical Report CMU/SEI-92-TR-30, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pa, 1992.

[Willett51] Willett, Alan H. *The Economic Theory of Risk and Insurance*. University of Pennsylvania Press, Pittsburgh, PA., 1951.

[Yourdon92] Yourdon, Edward. *Decline & Fall of the American Programmer.* Yourdon Press, Prentice-Hall, Inc., 1992.

[Zweig94] Zweig, Phillip L., Kelley Holland, Leah Nathans Spiro, Peter Burrows, Greg Burns, and Zachary Schiller. "Managing Risk." *Business Week*, November 1994, 86-96.

Works Consulted

[Abdel-Malek94] Abdel-Malek, Nabil. *Quantitative Process Analysis - An Aid to Process Institutionalization.* The Sixth Annual Software Technology Conference, April, 1994.

[AFMC93] Air Force Material Command. *Software Development Capability/Capacity Review.* June 1993.

[AFSC87] Air Force Systems Command. *Software Risk Management.* AFSCP/AFLCP 800-5,1987.

[AFSC88] Air Force Systems Command. *Software Risk Abatement.* AFSC/AFLC Pamphlet 800-45, 1988.

[American85] *The American Heritage Dictionary.* Second College Edition, Boston: Houghton Miflin Company, 1985.

[Aoyama93] Aoyama, Mikio. "Concurrent Development Process Model." *IEEE Software,* (July 1993): 46-55.

[ASQC87] American Society for Quality Control, 1987.

[Barker87] Barker, Joel. *Discovering the Future: The Power of Vision* (video). Chart House Int., (1987).

[Babel90] Babel, Philip S. *Software Development Integrity Program* (video). Software Productivity Consortium, March, 1990.

[Bell89] Bell, Trudy E. "Managing Murphy's law: engineering a minimum-risk system." *IEEE Spectrum,* 26, 6 (June 1989): 24-27.

[Blum92] Blum, Bruce I. *Software Engineering: A Holistic View.* Oxford University Press, London, 1992.

[Boebert80] Boebert, Earl. "Managing Software Projects in Government and Industry." State of the Art Seminars, October, 1980.

[Boehm81] Boehm, B. *Software Engineering Economics*. Prentice-Hall, Englewood Cliffs, NJ, 1981.

[Boehm88a] Boehm, B. "A Spiral Model of Software Development and Enhancement." *IEEE Computer*, Vol. 21, No. 5, May 1988, pp. 61-72.

[Boehm88b] Boehm, B. "Rapid Prototyping, Risk Management, 2167, and the Ada Process Model." *Proceedings of the U.S. Army AWIS Ada Symposium.* TRW, Fairfax, Virginia, September 1988.

[Boehm89] Boehm, B. *IEEE Tutorial on Software Risk Management*. New York: IEEE Computer Society Press, 1989.

[Boehm90] Boehm, B. "Software Risk Management: Principles and Practices." *IEEE Software* (January 1990): 32-41.

[Boehm91] Boehm, Barry W. "Software Risk Management: Principles and Practices." *IEEE Software*, 8,1 (January 1991):32-41.

[Bowles91] Bowles, Jerry, and Joshua Hammond. *Beyond Quality*. The Berkley Publishing Group, New York, N.Y., 1991.

[Brassard88] Brassard, Michael. *The Memory Jogger*™. GOAL/QPC, Methuen, MA., 1988.

[Brooks75] Brooks, F.P. *The Mythical Man-Month*. Addison-Wesley, 1975.

[Brooks87] Brooks, F.P. "No Silver Bullet: Essence and Accidents of Software Engineering." *IEEE Computer, 20,4* (April 1987), 10-19.

[Carr93] Carr, Marvin, Suresh Konda, Ira Monarch, Carol Ulrich, and Clay Walker. *Taxonomy Based Risk Identification.* (CMU/SEI-93-TR-6). Pittsburgh, Pa.: Software Engineering Institute, Carnegie Mellon University, 1993.

[Charette88] Charette, Robert N. *Software Engineering Risk Analysis and Management*. New York: McGraw-Hill, 1988.

[Charette90] Charette, R N. *Application Strategies for Risk Analysis.* New York: Multiscience Press, 1990.

[Charette91a] Charette, RN. "Information Technology Risk Engineering." SEI/NSIA workshop on Software Risk, Carnegie Mellon University, Software Engineering Institute, Pittsburgh, PA, February 1991.

[Charette91b] Charette, Robert N. "Engineering Risk to Maximize Commercial Opportunities." Itabhi Corporation, Software Tools Conference, Wembley, England, June 1991.

[Charette91c] Charette, R N. "Inside RISKS: The Risks with Risk AnaIysis." *Communications of the ACM,* 34,6,106, June 1991.

[Charette92a] Charette, R N. "Charette Contributes Risk Aversion Techniques to Spiral Process." Software Productivity Consortium, *SPC Quarterly*, Summer, 1992.

[Charette92b] Charette, Robert N. "Building Bridges over Intelligent Rivers." Itabhi Corporation, *American Programmer*, September 1992, Vol.5 no. 7.

[Charette93] Charette, Robert N. "Essential Risk Management: Notes From the Front." Itabhi Corp., 2nd SEI Conference on Risk Management, Pittsburgh, Pa., March 1993.

[Chittister93] Chittister, Clyde and Yacov Y. Haimes. "Risk Associated with Software Development: A Holistic Framework for Assessment and Management." *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 23, No. 3, May/June 1993.

[Clemen91] Clemen, Robert T. *Making Hard Decisions an Introduction to Decision Analysis.* PWS-KENT Publishing Company, 1991.

[Comptroller79] Comptroller General. *Contracting for Computer Software Development. FGMSD-80-4*, Government Accounting Office, Washington, D.C., September 1979.

[Crosby80] Crosby, Phil. *Quality is Free.* McGraw-Hill, New York, N.Y., 1980.

[Cutler 89] Cutler, R. "A comparison of Japanese and U.S. high-technology transfer practices." *IEEE Transactions on Engineering Management*, 36, 1 (Feb. 1989), 17-24.

[Debou93] Debou, Christophe, Norbert Fuchs, and Heinz Saria. "Selling Believable Technology." *IEEE Software*, November, 1993, pp. 22-27.

[Deming82] Deming, W. Edwards. *Out of the Crisis.* Cambridge, MA: Massachusetts Institute of Technology, Center for Advanced Engineering Study, 1982.

[Desjardins94] Desjardins, Richard J. "Applying SEI's CMM and ISO 9001 Requirements: Lessons Learned", Software Technology Conference, April, 1994.

[Dion93] Dion, Raymond. "Process Improvement and the Corporate Balance Sheet." *IEEE Software*, (July 1993): 28-35.

[DoD88] Department of Defense Military Standard. *Defense System Software Development.* February 29,1988, DOD-STD-2167A.

[Drucker92] Drucker, Peter F. *Managing for the Future.* Truman Talley Books, 1992.

[DSMC83] Defense Systems Management College. *Risk Assessment Techniques.* Fort Belvoir, VA., July 1983.

[DSMC86a] Defense Systems Management College. "Risk Management: Concepts and Guidance." Fort Belvoir, Va: DSMC, 1986.

[DSMC86b] Defense Systems Management College. "Systems Engineering Management Guide." Fort Belvoir, Va: DSMC, 1986.

[DSMC88] Defense Systems Management College. *Mission Critical Computer Resources Management Guide.* Fort Belvoir, Virginia: DSMC, 1988.

[DSMC91] Defense Systems Management College. "Executive Forum For Systems Transition, Managing Software Technical Risk." *Program Manager,* 19, May/June 1991.

[Dorofee93] Dorofee, Audrey. "Risk Process Model." Proceedings of the Software Engineering Symposium, Pittsburgh, Pa., August 23-26,1993.

[Dutton93] Dutton, James E. "Commonsense Approach to Process Modeling." *IEEE Software*, (July 1993): 56-64.

[Dyer91] Dyer, Dr. Wayne. *The Universe Within You.* Nightingale-Conant Corporation, Chicago, Illinois, 1991.

[Fairley94] Fairley, R. "Risk Management for Software Projects." *IEEE Software*, May, 1994.

[Fichman93] Fichman, Robert G., and Chris F. Kemerer. "Adoption of Software Engineering Process Innovations: The Case of Object Orientation." *Sloan Management Review*, Winter, 1993. pp. 7-22.

[Fischhoff86] Fischhoff, B., S. Watson, and C. Rope. "Managing Risk." *Policy Sciences,* 17,123- 139, 1986.

[FitzGerald90] FitzGerald, Jerry, and Ardra F. FitzGerald. "A Methodology for Conducting a Risk Assessment." *Designing Controls into Computerized Systems,* Second Edition, Redwood City, CA: Jerry FitzGerald & Associates, 1990.

[Folkes92] Folkes, Susan, and Sue Stubenvoll. *Accelerated Systems Development.* Prentice-Hall International, Hemel-Hempstead, UK, 1992.

[Fowler90] Fowler, P., and S. Rifkin. *Software Engineering Process Group Guide.* Technical Report CMU/SEI-90-TR-24, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pa., Sept. 1990.

[Fowler92] Fowler, P., and J. Maher. *Foundations for Systematic Software Technology Transition.* 1992 SEI Technical Review, R.L. Van Scoy, ed. Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pa., 1992.

[GAO86] Government Accounting Office. *Technical Risk Assessment: The Current Status of DoD Efforts.* GAO/PEMD-86-5, Government Accounting Office, Washington. D.C., 1986.

[Garcia94] Garcia, Suzanne M. "Process Areas of the SE-CMM Workshop Release." Systems Engineering Capability Maturity Model Project, Carnegie Mellon University, June, 1994.

[Garey79] Garey, Michael R., and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Bell Laboratories, Murray Hill, N.J., W.H. Freeman and Company, N.Y., 1979.

[Giarratano89] Giarratano, Joseph, and Gary Riley. *Expert Systems Principles and Programming*. PWS-KENT Publishing Company, Boston, MA., 1989.

[Gilb88] Gilb, T. *Principles of Software Engineering Management*. Addison-Wesley, 1988.

[Gilmore93] Gilmore, Steve, Ray Odom, and Ed Rose. "The Competitive Edge: Building a Collaborative Workforce." *Brevard Technical Journal*, (December 1993): 33-36.

[Gorsuch92] Gorsuch, Tom, and Paul Johannessen. "Empowering Teams: A Post Fact Study." School of Business, Florida Institute of Technology, 1992.

[Hall92] Hall, Elaine. "Software Risk Management." *ISD Images*, Information Systems Division, Harris Corporation, November, 1992.

[Hall93] Hall, Elaine M. "A Pro-Active Approach to Software Risk Management." *Proceedings of the 1993 Harris Engineering/Manufacturing Seminar*, Harris Corporation, Melbourne, Florida, January, 1993.

[Hall94] Hall, Elaine M. "Evolution of Essential Risk Management Technology." *Proceedings of the Third SEI Conference on Software Risk*, April, 1994.

[Hall95] Hall, Elaine M., Gary P. Natwick, F. Carol Ulrich, and Charles B. Engle, Jr. "Streamlining the Risk Assessment Process." *Proceedings of the Seventh Software Technology Conference*, Hill AFB, Salt Lake City, Utah, April, 1995.

[Hansson89] Hansson, S. O. "Dimensions of Risk." *Risk Analysis,* 9,1,107-112,1989.

[Hauser88] Hauser, J. R., and Clausing, D. "The House of Quality." *Harvard Business Review,* 66,3,1988 May June.

[Hefner94] Hefner, Rick, and Karen Flink. "Software Process Modeling: Organizational and Project Perspectives." *Proceedings of the 6th SEPG National Meeting*, April 1994.

[Henley92] Henley, Ernest J., and Hiromitsu Kumamoto. *Probabilistic Risk Assessment.* New York, NY: IEEE Press, 1992.

[Hersh93] Hersh, Art. "Where's the Return on Process Improvement?" *IEEE Software*, (July 1993): p.12.

[Hersh77] Hersh, M. H. "Risk Aversion vs. Technology Implementation." Defense Systems Management College, Fort Belvoir, Virginia, 1977.

[Hetzel90] Hetzel, Bill, and Rick Craig. *Software Measures and Practices Benchmark*, Software Quality Engineering, TR - 900, December 1990.

[Higuera93] Higuera, Ronald, P., and David P. Gluch. "Risk Management and Quality in Software Development." *Proceedings of the Eleventh Annual Pacific Northwest Software Quality Conference*, October 18-20,1993.

[Higuera94] Higuera, Ronald P., Audrey J. Dorofee, Julie A. Walker, and Ray C. William. *Team Risk Management: A New Model for Customer-Supplier Relationships.* (CMU/SEI-94-SR-5). Pittsburgh, PA.: Software Engineering Institute, Carnegie Mellon University, July 1994.

[Hudson92] Hudson, Anita. "Technology Transition & Adoption." *Excel: Harris Engineering Productivity Group Newsletter*, Vol. 3, Issue 2, November, 1992.

[Humphrey87a] Humphrey, W. S. *Characterizing the Software Process: A Maturity Framework.* CMU/SEI-87-TR-11, ADA 182895, Software Engineering Institute, June 1987.

[Humphrey87b] Humphrey, W., and W. Sweet, *A Method for Assessing the Software Engineering Capability of Contractors.* CMU/SEI-87-TR-23, Software Engineering Institute, 1987.

[Humphrey89] Humphrey, W.S. *Managing the Software Process.* Reading, MA: Addison-Wesley, 1989.

[IEEE88] IEEE Standard for Software Project Management Plans, ANSI/IEEE STD 1058.1-1987, October 1988.

[IEEE90] IEEE Standard Glossary of Software Engineering, IEEE-STD-610.12, 1990.

[IEEE91] IEEE Software Engineering Standards collection, Spring 1991.

[Imai86] Imai, M. *KAIZEN: The Key to Japan's Competitive Success.* New York: McGraw-Hill, 1986.

[ISO91] ISO 9000-3, American National Standards Institute(ANSI) International Standards Organizations(ISO). *Quality Management and Quality Assurance Standards.* First edition, 1991-06-01, Reference number ISO 9000-3:1991(E).

[Jones92] Jones, Capers. "Process Assessments and Software Risks." *CrossTalk,* (November 1992): 17-20.

[Juran88] Juran, J. M. *Juran's Quality Control Handbook: Fourth Edition.* New York, NY: McGraw-Hill Book Company, 1952, 1979,1988.

[Juran89] Juran, J. M. *Juran on Leadership for Quality.* New York, NY: The Free Press, A Division of Macmillan, Inc., 1989.

[Kaplan81] Kaplan, Stanley, and John B. Garrick. "On The Quantitative Definition of Risk." *Risk Analysis,* 1,1, (1981): 11-27.

[Katzenbach93] Katzenbach, Jon R., and Douglas K. Smith. "The Discipline of Teams." *Harvard Business Review* 72,2 (March-April, 1993):111-120.

[Kezsbom89] Kezsbom, Deborah S., Donald L. Schilling, and Katherine A. Edward. *Dynamic Project Management.* New York, N.Y.: John Wiley & Sons, 1989.

[Kirkpatrick92] Kirkpatrick, R.J., J.A. Walker, and R. Firth. "Software Development Risk Management: An SEI Appraisal." *1992 SEI Technical Review,* R.L. Van Scoy, ed. Software Engineering Institute: Carnegie Mellon University, Pittsburgh, PA, 1992.

[Koltun91] Koltun, Phil. "Lighting the Way to Reuse." *Harris Excel.* Harris Corporation, Melbourne, FL., October, 1991.

[Koltun92] Koltun, Philip, and Pedro Martinez. "Software Process Assessment Activities at Harris." Harris Corporation, Melbourne, FL., 1992.

[Korson92] Korson, Timothy D., and Vijay K. Vaishnavi, "Managing Emerging Software Technologies: A Technology Transfer Framework." *Communications of the ACM*, September, 1992, Vol.35, No. 9, pp. 101-111.

[Lai93] Lai, Robert. "The Move to Mature Processes." *IEEE Software*, (July 1993): 14-17.

[Leveson88] Leveson, Nancy G., "Software Safety: What, Why and How." *ACM Computing Surveys*, Vol. 18(2), June 1988.

[Lipke92] Lipke, Walter H., and Kelley L. Butler. "Software Process Improvement: A Success Story." *CrossTalk*, (November 1992): 29-39.

[Luger89] Luger, George F., and William A. Stubblefield. *Artificial Intelligence and the Design of Expert Systems*. Benjamin/Cummings Publishing Company, Inc., 1989.

[Matson94] Matson, Jack. "Risking Your Way to the Top." *Graduating Engineer*, (March 1993): 34-36.

[McCall77] McCall, J., P. Richards, and G. Walters, "Factors in Software Quality." three volumes, NTIS AD-A049-014, 015, 055, November 1977.

[Musa89] Musa, John D. "Tools for Measuring Software Reliability." *IEEE Spectrum*, February 1989.

[Nakahara94] Nakahara, Roy H. "Transitioning to a Team Approach to Software Development." *Proceedings of the Sixth Software Technology Conference*, April 1994.

[Naur69] Naur, P., and B. Randell (eds.). *Software Engineering: A Report on a Conference sponsored by the NATO Science Committee*. NATO, 1969.

[Paul89] Paul, J.H., and G.C. Simon. "Bugs in the Program: Problems in Federal Government Computer Software Development and Regulation." Staff Study for the House Committee on Science, Space, and Technology, September, 1989.

[Paulk93a] Paulk, Mark C., Bill Curtis, Mary Beth Chrissis, and Charles V. Weber. *Capability Maturity Model for Software, Version 1.1.* Technical Report CMU/SEI-93-TR-24, Software Engineering Institute, February, 1993.

[Paulk93b] Paulk, Mark C., Charles V. Weber, Suzanne M. Garcia, Mary Beth Chrissis, and Marilyn W. Bush. *Key Practices of the Capability Maturity Model, Version 1.1.* Technical Report CMU/SEI-93-TR-25, Software Engineering Institute, February, 1993.

[Paulk93c] Paulk, Mark C., Bill Curtis, Mary Beth Chrissis, and Charles V. Weber. "Capability Maturity Model, Version 1.1." *IEEE Software*, vol. 10, no. 4, (July 1993): 18-27.

[Petroski82] Petroski, H. *To Engineer is Human: The Role of Failure in Successful Design.* New York: St. Martin's Press, 1982.

[Plenum81] *Risk Analysis*, Society for Risk Analysis, 1981.

[Pressman92] Pressman, Roger S. *Software Engineering A Practitioner's Approach.* McGraw-Hill, Inc., 1992.

[Qoman90] Qoman, H. F. "Risk Management." *Risk Analysis,* 10, 2, pp. 201-205,1990.

[Redwine84] Redwine, S.T., Jr., L.G. Becker, A.B. Marmor-Squires, R.J. Martin, S.H. Nash, and W.E. Riddle. *DoD Related Software Technology Requirements, Practices, and Prospects for the Future.* IDA Paper P-1788, Institute for Defense Analyses, Alexandria, Virginia, June 1984.

[Redwine85] Redwine, S. and W. Riddel. "Software Technology Maturation." IEEE Conference on Software Engineering, Aug. 1985. pp. 189-200.

[Rifkin91] Rifkin, S. and C. Cox. *Measurement in Practice.* Technical Report CMU/SEI-91 -TR- 16, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, July 1991.

[Roe89] Roe, Robert A. et al. "Bugs in the Program: Problems in Federal Government Computer Software Development and Regulation." U.S. Government Printing Office, Washington, September 1989.

[Rogers83] Rogers, E.M. *Diffusion of Innovations.* (New York: Free Press, 1983)

[Rose94] Rose, Susan. "Applying the Software Process Assessment Method to Systems Acquisition." *Proceedings of the Sixth Software Technology Conference*, April 1994.

[Ross90] Ross, N. "Using metrics in quality management." *IEEE Software*, Vol 7. No. 4, July, 1990, pp. 80-82.

[Row88] Row, W. D. *An Anatomy of Risk.* Malabar, Florida: Robert E. Krieger Publishing Company, 1988.

[Royce89] Royce, W.W. "Managing the Development of Large Software Systems: Concepts and Techniques." *Proceedings of the Ninth International Conference on Software Engineering* (reprinted from 1970 original). USA: ACM Press, 1989.

[Rugg93] Rugg, David. "Using a Capability Evaluation to Select a Contractor." *IEEE Software*, (July 1993): 36-45.

[Saaty80] Saaty, T.L. *The Analytic Hierarchy Process.* McGraw-Hill, Inc., 1980.

[SCE94] Members of the CMM-Based Appraisal Project. *Software Capability Evaluation (SCE) Version 2.0 Implementation Guide.* Software Engineering Institute, Carnegie Mellon University, Technical Report CMU/SEI-94-TR-05, Pittsburgh, PA. February 1994.

[Schuller93] Scheuller, Robert H. *Possibility Thinking.* Thomas Nelson Publishers, Nashville, Tennessee, 1993.

[Schulles88] Schulles, Peter R. *The Team Handbook: How to Use Teams to Improve Quality.* Joiner Associates, Inc., 1988.

[SEI91] Proceedings of the Annual SEI Software Risk Conferences, Software Engineering Institute, Carnegie Mellon University, 1991-1994.

[SEI93] Software Engineering Institute, Carnegie Mellon University, 1993.

[Sherer94] Sherer, S. Wayne and Jack Cooper. "Software Acquisition Maturity Model." *Proceedings of the Sixth Software Technology Conference*, April 1994.

[Sisti94] Sisti, Francis J., and Joseph Sujoe. *Software Risk Evaluation Method.* (CMU/SEI-94-TR-19). Pittsburgh, PA.: Software Engineering Institute, Carnegie Mellon University, October 1994.

[Statz94] Statz, Dr. Joyce. "Tutorial: Software Project Management for a Level 3 Organization." *Proceedings of the Sixth Software Technology Conference*, April 1994.

[Stivers93] Stivers, Ben. *Reuse Maturity Model.* Harris Data Services Corporation, Montgomery, AL., June, 1993.

[SWTS92] Software Technology Process Management Support, 1992.

[Travalent94] Travalent, John J., and Elizabeth A. Hubbard. "Implementing Team Risk Management: A Case Study and Lessons Learned." *Proceedings of the Sixth Software Technology Conference*, April 1994.

[VanScoy92] VanScoy, R. *Software Development Risk: Problem or Opportunity.* Technical Report CMU/SEI-92-TR-30, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pa, 1992.

[Weber91] Weber, C.V., M.C. Paulk, C.J.Wise, and J.V. Withey. *Key Practices of the Capability Maturity Model.* Technical Report CMU/SEI-91-TR-25, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pa, August 1991.

[Willett51] Willett, Alan H. *The Economic Theory of Risk and Insurance.* University of Pennsylvania Press, Pittsburgh, PA., 1951.

[Yourdon92] Yourdon, Edward. *Decline & Fall of the American Programmer.* Yourdon Press, Prentice-Hall, Inc., 1992.

[Zweig94] Zweig, Phillip L., Kelley Holland, Leah Nathans Spiro, Peter Burrows, Greg Burns, and Zachary Schiller. "Managing Risk." *Business Week*, November 1994, 86-96.

Appendix A - Risk Management Survey

RISK MANAGEMENT SURVEY

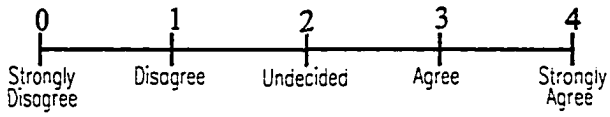
Your title_____ Years experience on software projects_____

Your primary role on the project is (check one):

Program Management_____ Systems_____ Software_____ Test_____

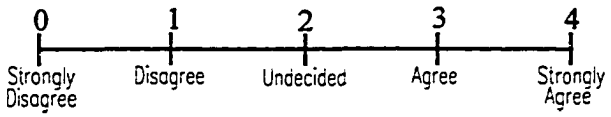
CM_____ QA_____ Customer_____ Other_____

This survey will be used to improve risk management practices on software-intensive projects. Because of your involvement in a software risk assessment, you have been selected to participate in the survey. For each practice, give your perception of both performance (how well we do it) and importance (how important it is) to the project. Characterize your level of agreement to each statement using the following scale:



I. Risk Assessment Practices:	**Performance**	**Importance**
0. *Example*: There is a policy in our organization to identify risks.		
1. A software risk assessment was performed early in the project.		
2. Risk was identified in a constructive manner.		
3. Critical risks to the project were identified and evaluated.		
4. Non-software risks were identified.		
5. The risk assessment team was trained and experienced.		
6. I participated in a software risk assessment for the project.		
7. I identified risks to the project.		
8. I openly communicated in the interview discussion.		
9. I contributed to the risk assessment interview discussion.		
10. My time was used efficiently in the risk assessment.		
11. My issues and concerns were captured and reported.		
12. The risk assessment provided a baseline of assessed risks.		
13. I learned techniques for risk identification from the risk assessment.		
14. I apply techniques for risk identification at work.		
15. The results of the risk assessment activities are visible.		
16. The results of the risk assessment activities are confidential.		
17. Risk assessment is performed at major project milestones.		
18. There is support for practicing risk assessment on the project.		
19. Identified and assessed risks are managed on the project.		
20. The customer has a positive perception of the project risk program.		

(Please turn page over)



II. Risk Management Practices:	Performance	Importance
21. There is a policy to report risks at program management reviews.		
22. The risk management policy is documented.		
23. There is a commitment to the risk management process.		
24. There is a commitment to risk management metrics.		
25. There is a recognition system that rewards risk identification.		
26. Risks are gathered from lower levels and communicated higher.		
27. Risks are communicated within the program team.		
28. Risks are communicated between the program team and customer.		
29. Risks are communicated between the team, customer and end-user.		
30. Upper management is committed to solving problems.		
31. Quality assurance is committed to auditing compliance to standards.		
32. Engineers are committed to use of risk management.		
33. Customers and end-users are committed to use of risk management.		
34. Risk management meetings are held periodically.		
35. Budget is allocated to risk management on the project.		
36. Resources (e.g. people) are allocated to risk management.		
37. Time, budget and people are allocated to risk management.		
38. I have received basic risk concepts training.		
39. I have received risk management process training.		
40. I have been trained in how to measure and quantify risks.		
41. I have been trained in decision techniques (e.g. decision trees).		
42. Risks are identified by individuals when asked directly.		
43. Risks are voluntarily identified by individuals on the project.		
44. Risks are actively sought out on the project.		
45. Opportunities for cost savings are identified on the project.		
46. Risks are analyzed on the project.		
47. Analyzed risks are prioritized on the project.		
48. Risks are analyzed for their root cause on the project.		
49. Risks are analyzed quantitatively on the project.		
50. The cost-benefit of risks are calculated on the project.		
51. Risk reduction is discussed on the project.		
52. Risk reduction plan is documented on the project.		
53. Risk reduction plan is executed to completion on the project.		
54. Risk reduction plan is revised as required on the project.		

(Please continue on next page)

II. Risk Management Practices (cont.):	**Performance**	**Importance**
55. Critical risks are tracked on the project.		
56. All risks are tracked on the project.		
57. Triggering events are monitored on the project.		
58. Deviations from risk reduction plans are corrected.		
59. Discussions increase awareness of what could be improved.		
60. Written evaluations document what could be improved.		
61. Written evaluations are analyzed to provide lessons learned.		
62. The lessons learned are documented on the project.		
63. Feedback is used to improve the process on the project.		
64. The program manager performs risk management.		
65. The program manager and key staff perform risk management.		
66. The program team performs risk management.		
67. The program team and customer perform risk management.		
68. The program team, customer, and users perform risk management.		
69. A risk champion (someone who advocates risk management) exists.		
70. A few risk champions exist on the project.		
71. Many risk champions exist on the project.		
72. Procedures for risk management are verbally stated.		
73. Procedures for risk management are documented.		
74. Procedures for risk management are updated at milestones.		
75. Procedures for risk management exist as a living document.		
76. A risk survey is used to identify risks.		
77. A risk taxonomy structures the process of identifying risks.		
78. A risk management form is used to identify risks at any time.		
79. Risk metrics are graphed to identify risk trends.		
80. A Top 10 Risk List is used to report risks.		
81. A Risk Database is used to track all identified risks.		
82. Risk metrics are used to monitor triggering events.		
83. Risk analysis is automated on the project.		
84. Risk metrics are *defined* on the project.		
85. Risk metrics are *collected* on the project.		
86. Risk metrics are *analyzed* on the project.		
87. Risk metrics are *reported* on the project.		
88. Top risks are reported to management at program reviews.		
89. Top risks are reported to the customer at program reviews.		
90. Top risks are reported to the program team throughout the project.		
91. Top risks are reported to the customer throughout the project.		

(Please turn page over)

III. Observations:

Some effective risk management practices include

_____

_____

_____

Some ineffective risk management practices include

_____

_____

_____

One thing needed in my work group is

_____

_____

_____

The risk management lessons learned on the project include

_____

_____

_____

Other comments regarding software risk management

_____

_____

_____

Appendix B - Risk Management Survey Results

Observations from the risk management survey were responses to the following open-ended statements:

1. Some effective risk management practices include:
2. Some ineffective risk management practices include:
3. One thing needed in my work group is:
4. The risk management lessons learned on the project include:
5. Other comments regarding software risk management:

These observations are direct quotes from survey respondents. Each quote is identified by a letter and number. Projects are designated A, B, or C, and D indicates a management response.

Some effective risk management practices include:

- A-1 Periodically reviewing and prioritizing risks; identifying risks that have been resolved.
- A-6 Including program team to identify risks, prioritizing risk, tracking critical risks.
- A-7 Identifying risks early on for a new project.
- B-1 Using risk process to gather risks at major milestones, prioritizing risk and having mitigation plans are people assigned.
- B-2 Weekly meetings and getting customer involved.
- B-3 Survey, Database, Consolidation, Action/Mitigation Plans, Management Support, Team/Customer involvement.
- B-5 Isolating root cause of risk and working at resolving the problem instead of symptoms.
- B-6 Risk survey, risk management team, customer involvement.
- B-13 Risk Assessment survey (including this survey), risk management forms, risk management board.
- C-11 Risk Tracking, Risk prioritization, Risk Analysis.
- C-14 RRRB
- C-15 Identification and documentation of well-known risks.
- C-19 Identifying risks.
- C-22 Regular activities planned to evaluate risks and identify new ones. Includes risk management in program plans and applicable CURLS (i.e. test plans, transition plans).
- C-25 Identify risks consistently and constantly. Just having a risk program in place is a major step in the correct direction.

- C-26 Identify, quantify and mitigate project risks as an ongoing part of the program process.
- C-38 I believe the process is in place to make the risk management effective but I don't feel the benefit to the Pits has been communicated.
- C-37 Tracking top 10/20 program risks.
- C-41 Identification
- D-1 Risk Assessment techniques - Taxonomy and techniques
- D-2 Brainstorming and tracking results, automated search of risk abatement techniques used by other projects.
- D-6 Top 10 list, risk management forms posted in areas and submittable at any time, risk review board.

Some ineffective risk management practices include:

- A-1 Ignoring risks that are outside program team control. Risks "kicked up" to management are not tracked or mitigated.
- A-6 Statusing all risk/concerns identified by team. (It takes a lot of time, and some are insignificant or uncontrollable).
- A-7 Fitting projects to a management's negotiated schedule and not taking into consideration the real extent of the engineering effort. The customer often forces this issue by setting an end date but not awarding the contract for the original start date.
- B-1 Too much data over 300 risks to filter through - mitigation plans come in slow when customer is involved - need to train customer and subcontractors to understand process, even at management levels in these organizations.
- B-2 Management only practice.
- B-3 Using Joint Risk Working Group session and forum to "best up" the contractor.
- B-5 Performing risk management as a "spare time" activity. Treating risk management as an isolated activity and not as an integral part of the program. Limiting risk management to a "select" few without direct program management involvement.
- B-6 Proceeding at risk, schedule driven development, customer uses risk management to help drive own agenda, uses RM as a Harris bashing tool.
- B-11 Although I participated in the first risk sessions and heard them presented badly at PDR - I know nothing more about them except for one I was asked about.
- B-13 People's attitudes - a brilliant idea of one person may be perceived as (and may actually be) a risk. If identified as a risk, the person with the idea often takes offense. Maybe this is not a practice, but it happens.
- B-18 Need more communication to all program team members.

- C-11 Risks for risks sake.
- C-15 Id and documentation of new risk - some haven't been brought to program level.
- C-19 Not linking requirements clarification with cost during requirements analysis process allowed uncontained cost growth to program.
- C-25 Risks tend to be identified bottoms up. Systems level architectural risks are not visible.
- C-26 Leaving it up to only a few assigned "experts" on the program.
- C-38 Since the PITs don't see direct benefit and they are pressed with other duties the risk management is seen as a hoop to jump through. This has resulted in risks not being presented because the process levies what is seen as extra work.
- C-37 Focus on development almost exclusively for nominal flight conditions. Worst case scenarios are important to the development of ROBUST software and are effective at revealing faulty and/or inadequate designs.
- C-41 Communication, quantification, analysis
- D-1 Buy in by PMO (they don't have the time and $)
- D-2 Directives to "manage" risks without consideration of workload, cost, or schedule commitments.

One thing needed in my work group is:

- A-1 Distributed risk management vs. top-level group grope.
- A-4 Development of automated tools to address risk management (i.e. forms, data base, tracking, reporting).
- A-7 Good tools and an on going training program.
- B-1 Process to communication to risk mitigation to the program team members.
- B-3 Broader communication to entire team.
- B-5 Administrative support for the risk working groups.
- B-13 "Outside Evaluation." Although surveys and the like are a distraction from "the real work," they often have good results. NCIC needs an outside evaluation that is more than a one time event.
- B-14 More/better communication of identified risks. Actions taken to correct or lessen the
- risk impact. Identification of risk is ok, but actions speak louder than words.
- C-11 For the PITs to address and restructure risks.
- C-14 The risks in my work group were derived externally. For reasons above no risks have been formalized in the process internally.
- C-21 Not providing feedback to the program team of risk status.
- C-26 Some better ways of quantifying the real risks identified.
- C-37 More people and plotter access.
- C-41 1. More visibility into program risks. 2. Move input into risk identification/quantization, especially software.
- D-2 More effective use of shared data, on-line access, common tools, and automation.

The risk management lessons learned on the project include:

- A-1 The risks below the "top ten" do not get adequate attention.
- A-4 Upper management needs to provide greater assistance in solving problems beyond program control.
- A-6 It is helpful to identify and track risks. I think we were able to avoid some major problems by using risk management.
- A-7 Some risk just don't go away.
- B-1 Must have prioritization list within workable mitigation plans.
- B-3 You must have continuity in the Risk Management Program. Identifying is not enough. Mitigation plans with action are necessary.
- B-5 Major risks should have been addressed early in the program. Direct program management involvement is needed. Significant extra effort is needed to administer risk management activities.
- B-6 Proceeding at risk is risky.
- B-13 Don't know.
- C-11 Just having a program isn't enough. It has to be used and dynamic.
- C-25 Be careful about tightly coupling program contractual dollars to risk assessments. Identification of "potential" overruns makes everyone uneasy.
- C-26 Allocation of risk and then assignment of appropriate risk to the CTC.
- C-36 I have nothing to do with the software for controlling or monitoring the antenna.

- C-37 For rolling wave planning to be successful it must be automated or significantly simplified. Choice of tools has a significant impact on schedule. Identifying the appropriate participants and well-scoped agenda is essential to a task group's success.

- C-41 Must be more proactive in seeking out risks.

- D-1 Identifies a lot more than software risks.

- D-2 You cannot dictate team to manage risk without providing tools and assistance to do so.

Other comments regarding software risk management:

- A-1 Don't do software risk management; do project-wide risk management; segmenting software does not make sense unless all you want is an SEI check mark.
- A-4 On the Heritage program, risks were identified but no plans were generated or executed to address the risks.
- A-7 Small risks can be solved but major risks such as schedule, available time and money to complete the task are addressed but keep getting pushed to the next milestone. Upper management does not want to hear bad news concerning cost and schedule. Thus everyone paints a brighter picture hoping a miracle will happen.
- B-1 Should be program risk management with software elements.
- B-3 Software risk management is only one arm of the program. It (risk management) must include total system, and must include customer involvement.
- B-5 Needs to be incorporated as part of software management and should not be added on top of other activities.
- B-9 Questions appear to be redundant. The 91 questions need to be reduced to around 15 questions.
- B-11 Maybe if people on project could see risks identified and how they are being managed.
- B-13 The survey has pointed out (or reminded me) that we do have a risk identification, assessment, mitigation plan documented and in place. However my perception is that it is a "background" process. It needs more focus in order to gain the awareness and respect that CM and QA have gained in recent years.
- B-15 Too many questions! Some questions duplicate. Some statements not clear as to what is meant. Should "importance" be characterized on a scale of 0-4? Is 4 the highest importance?

- B-18 My involvement with risk management has been minimal, but I completed this survey to the best of my knowledge.

- C-11 It needs to be generated and used below the systems level.

- C-15 Overall good job identifying risks, not so good doing something about them.

- C-25 Risk Management statements apply to program and system level now. Software specific definitions won't start for several months.

- C-35 I don't feel I am a good candidate for this survey. The first month I was here I was involved in a risk management survey. A couple of months later the results were available and I thought some valuable info was gathering. I have not been involved with any risk management since then. As far as software is concerned, we have identified our potential risk in our spirals.

- C-37 It is much more effective than management by objective and a definite step in the right direction. More efficient means are needed to collect and evaluate metrics. Focuses all on the important problems at hand much earlier in the life cycle than other techniques.

- D-2 All project risk management should be handled in a consistent manner. Software risks are not that different than any other risk.

Appendix C - Risk Management Capability Maturity Model

(PRO.FA.01) Risk Identification

A focus area for the process dimension.

The purpose of **Risk Identification** is to define the process of identifying risk. **Risk Identification** involves defining the process tasks, inputs, outputs, driving and supporting mechanisms to identify the risk and source of risk.

(PRO.FA.01.KP.01) Define Risk Identification Process	L2
(PRO.FA.01.KP.02) Define Risk Assessment Method	L2
(PRO.FA.01.KP.03) Develop Risk Checklists	L2
(PRO.FA.01.KP.04) Develop Risk Identification Form	L2
(PRO.FA.01.KP.05) Establish Risk Database Schema	L2

(PRO.FA.01.KP.01) Define Risk Identification Process

Define the risk identification process in terms of the tasks, inputs, outputs, driving and supporting mechanisms to identify risk and source of risk.

(PRO.FA.01.KP.02) Define Risk Assessment Method

Define a risk assessment method in terms of the tasks, inputs, outputs, driving and supporting mechanisms to independently assess risk and brief summarized findings back to the project.

(PRO.FA.01.KP.03) Develop Risk Checklist

Develop a risk checklist in terms of categories to be reviewed for risk identification. Checklists may be developed using a risk taxonomy, project work breakdown structure, or specific phase of development.

(PRO.FA.01.KP.04) Develop Risk Identification Form

Develop a risk identification form to record an identified risk with sufficient detail to permit subsequent risk analysis.

(PRO.FA.01.KP.05) Establish Risk Database Schema

Establish a risk database schema in terms of data fields such as log number, risk description, priority, and risk status.

(PRO.FA.02) Risk Analysis

A focus area for the process dimension.

The purpose of **Risk Analysis** is to define the process of analyzing risk. **Risk Analysis** involves defining the process tasks, inputs, outputs, driving and supporting mechanisms to assess the impact and likelihood of the risk.

(PRO.FA.02.KP.01) Define Risk Analysis Process	L2
(PRO.FA.02.KP.02) Define Risk Analysis Techniques	L3
(PRO.FA.02.KP.03) Define Risk Evaluation Criteria	L2
(PRO.FA.02.KP.04) Develop Risk Analysis Form	L2
(PRO.FA.02.KP.05) Establish Risk Prioritization Scheme	L2

(PRO.FA.02.KP.01) Define Risk Analysis Process

Define the risk analysis process in terms of the tasks, inputs, outputs, driving and supporting mechanisms to analyze and prioritize risk.

(PRO.FA.02.KP.02) Define Risk Analysis Techniques

Define risk analysis methods in terms of the rigor required to analyze risk in a cost-effective manner. Techniques may include Decision Trees, Influence Diagrams, and Tornado Diagrams.

(PRO.FA.02.KP.03) Define Evaluation Criteria

Define evaluation criteria in terms of categories of risk impact and likelihood of risk occurrence as well as the risk time frame. Categories may be high, moderate, or low.

(PRO.FA.02.KP.04) Develop Risk Analysis Form

Develop a risk analysis form to record the results of risk analysis with sufficient detail to permit subsequent prioritization.

(PRO.FA.02.KP.05) Establish Risk Prioritization Scheme

Develop an efficient risk prioritization scheme. Risks should be prioritized based on their potential impact to the project and likelihood of occurrence.

(PRO.FA.03) Risk Reduction

A focus area for the process dimension.

The purpose of **Risk Reduction** is to define the process of developing and executing a risk reduction plan. **Risk Reduction** involves defining the process tasks, inputs, outputs, driving and supporting mechanisms to reduce risk to an acceptable level.

(PRO.FA.03.KP.01) Define Risk Reduction Process	L3
(PRO.FA.03.KP.02) Define Risk Reduction Alternatives	L3
(PRO.FA.03.KP.03) Define Selection Criteria	L2
(PRO.FA.03.KP.04) Develop Risk Reduction Template	L3

(PRO.FA.03.KP.01) Define Risk Reduction Process

Define the risk reduction process in terms of the tasks, inputs, outputs, driving and supporting mechanisms to reduce risk to an acceptable level.

(PRO.FA.03.KP.02) Define Risk Reduction Alternatives

Define risk reduction alternative approaches to reducing risk to an acceptable level. Alternatives may include risk avoidance, risk transfer, contingency planning and buying information.

(PRO.FA.03.KP.03) Define Selection Criteria

Define criteria in terms of selecting from alternative approaches to reducing risk impact and/or likelihood of risk occurrence. Criteria may include minimization of impacts to cost, schedule, performance, or customer satisfaction. Selection decisions may be based on assumptions, constraints, and historical data, which may be revised as information becomes available.

(PRO.FA.03.KP.04) Develop Risk Reduction Template

Develop a template for a risk reduction plan. The plan may include objectives, approach, start date, milestones, due date, responsible person, resources required, authorization signature, actions taken and results achieved.

(PRO.FA.04) Risk Tracking

A focus area for the process dimension.

The purpose of **Risk Tracking** is to define the process of capturing, reviewing and reporting risk status. **Risk Tracking** involves defining the process tasks, inputs, outputs, driving and supporting mechanisms to monitor risk status.

(PRO.FA.04.KP.01) Define Risk Tracking Process	L3
(PRO.FA.04.KP.02) Define Risk Tracking Techniques	L3
(PRO.FA.04.KP.03) Define Risk Tracking Metrics	L4
(PRO.FA.04.KP.04) Define Triggering Events	L4

(PRO.FA.04.KP.01) Define Risk Tracking Process

Define the risk tracking process in terms of the tasks, inputs, outputs, driving and supporting mechanisms to monitor risk status.

(PRO.FA.04.KP.02) Define Risk Tracking Techniques

Define risk tracking techniques to monitoring risk status. Techniques may include risk tracking metrics, Top-10 List, and Technical Performance Measurement (TPM).

(PRO.FA.04.KP.03) Define Risk Tracking Metrics

Define risk tracking metrics in terms of variables to be monitored, such as time, effort, budget, milestones, function points or lines of code. Variables may be programmatic or technical.

(PRO.FA.04.KP.04) Define Triggering Events

Define triggering events in terms of actual threshold values or deltas that show a deviation from planned estimates. Triggering events may be related to time, such as periodic events or elapsed time.

(PRO.FA.05) Risk Control

A focus area for the process dimension.

The purpose of **Risk Control** is to define the process of integrating risk management to involve management, project team and customer on a routine basis to control risk. **Risk Control** involves defining the process tasks, inputs, outputs, driving and supporting mechanisms for responding to triggering events, correcting for variations from plans, and process improvement.

(PRO.FA.05.KP.01) Define Risk Control Process	L5
(PRO.FA.05.KP.02) Define Risk Control Techniques	L5
(PRO.FA.05.KP.03) Define Risk Control Metrics	L5
(PRO.FA.05.KP.04) Develop Corrective Action Procedure	L4
(PRO.FA.05.KP.05) Develop Risk Management Survey	L4

(PRO.FA.05.KP.01) Define Risk Control Process

Define the risk control process in terms of the tasks, inputs, outputs, driving and supporting mechanisms to respond to triggering events, correct for variations from plans, and process improvement.

(PRO.FA.05.KP.02) Define Risk Control Techniques

Define risk control techniques to respond to triggering events, correct for variations from plans, and process improvement. Techniques may include risk database timer, corrective action procedure, and risk management survey.

(PRO.FA.05.KP.03) Define Risk Control Metrics

Define risk control metrics in terms of the cost/benefits of the risk management process results. Measure costs such as effort and resources required. Measure benefits such as risk reduction leverage, customer satisfaction, and cost savings.

(PRO.FA.05.KP.04) Develop Corrective Action Procedure

Develop a corrective action procedure to correct for variations from risk reduction plans.

(PRO.FA.05.KP.05) Develop Risk Management Survey

Develop a risk management survey to obtain feedback to improve risk management practices.

(INF.FA.01) Document Policy

A focus area for the infrastructure dimension.

The purpose of **Document Risk Management Policy** is to provide organizational commitment to evolve risk management capability by establishing a foundational requirement for performing risk management, thereby developing a risk ethic in the organization culture. **Document Risk Management Policy** involves understanding the existing organization practices and obtaining the commitment to define a policy for performing risk management that is communicated to the entire organization. The defined policy should be at the appropriate maturity level of the organization with respect to its existing risk management practices to establish realistic requirements and expectations.

(INF.FA.01.KP.01) Assign Resources	L2
(INF.FA.01.KP.02) Survey Existing Practice	L3
(INF.FA.01.KP.03) Obtain Commitment	L3
(INF.FA.01.KP.04) Define Draft Policy	L2
(INF.FA.01.KP.05) Peer Review Policy	L3
(INF.FA.01.KP.06) Document Policy	L2
(INF.FA.01.KP.07) Approve Policy	L2
(INF.FA.01.KP.08) Communicate Policy	L3

(INF.FA.01.KP.01) Assign Resources

Assign resources (staff, budget, schedule) and responsibility for documenting the organizations risk management policy.

(INF.FA.01.KP.02) Survey Existing Practice

Survey the existing risk management practices performed within the organization to determine state of the practice.

(INF.FA.01.KP.03) Obtain Commitment

Obtain commitment and buy-in for performing risk management practices by communicating the benefits of risk management.

(INF.FA.01.KP.04) Define Draft Policy

Define draft risk management policy by involving those in the organization affected by the policy requirement.

(INF.FA.01.KP.05) Peer Review Policy

Peer review draft risk management policy to promote understanding and identify problems with the draft. Incorporate feedback and action items from the draft risk management policy peer review.

(INF.FA.01.KP.06) Document Policy

Document organizational risk management policy in a format that is easily understood and maintained, with revision history and current version and date.

(INF.FA.01.KP.07) Approve Policy

Approve organizational risk management policy through the appropriate management to promote support and commitment from the top.

(INF.FA.01.KP.08) Communicate Policy

Communicate organizational risk management policy to the workforce to promote awareness and understanding.

(INF.FA.02) Define Standard Process

A focus area for the infrastructure dimension.

The purpose of **Define Standard Risk Management Process** is to leverage for the organization a common and consistent process that is shared across the organization. **Define Standard Risk Management Process** involves establishing a multifunctional team with a charter to document the organization standard risk management process. The product is reviewed and approved to promote buy-in and acceptance for the standard process. The documented process is distributed within the organization.

(INF.FA.02.KP.01) Establish Action Team	L2
(INF.FA.02.KP.02) Develop Draft Standard Process	L2
(INF.FA.02.KP.03) Peer Review Draft Standard Process	L3
(INF.FA.02.KP.04) Document Standard Process	L2
(INF.FA.02.KP.05) Approve Standard Process	L2
(INF.FA.02.KP.06) Distribute Standard Process	L2

(INF.FA.02.KP.01) Establish Action Team

Establish a multifunctional action team and define the scope, schedule and budget of the product to be delivered.

(INF.FA.02.KP.02) Develop Draft Standard Process

Develop draft standard risk management process by following a standard process definition procedure.

(INF.FA.02.KP.03) Peer Review Draft Standard Process

Peer review draft standard risk management process to promote understanding and identify problems with the draft. Incorporate feedback and action items from the draft standard risk management process peer review.

(INF.FA.02.KP.04) Document Standard Process

Document standard risk management process in a format that is easily understood and maintained, with revision history and current version and date.

(INF.FA.02.KP.05) Approve Standard Process

Approve standard risk management process through the appropriate management to promote support and commitment from the top.

(INF.FA.02.KP.06) Distribute Standard Process

Distribute standard risk management process to the workforce to promote awareness, understanding and provide a common reference for the organization to follow.

(INF.FA.03) Train Risk Management

A focus area for the infrastructure dimension.

The purpose of **Train Risk Management Technology** is to raise the awareness and understanding of risk management through training and case study. **Train Risk Management Technology** involves instruction in the basic principles of risk, the process of risk assessment and risk management, as well as the methods, tools and metrics defined in the organization standard process.

(INF.FA.03.KP.01) Risk Management Concepts	L2
(INF.FA.03.KP.02) Risk Assessment Process	L2
(INF.FA.03.KP.03) Risk Management Process	L3
(INF.FA.03.KP.04) Risk Management Methods	L3
(INF.FA.03.KP.05) Risk Management Tools	L3
(INF.FA.03.KP.06) Risk Management Metrics	L4

(INF.FA.03.KP.01) Risk Management Concepts

Instruction in risk management concepts and the basic principles of risk to provide the foundation and motivation for performing risk management.

(INF.FA.03.KP.02) Risk Assessment Process

Instruction in the risk assessment process for individuals or teams performing independent risk assessment.

(INF.FA.03.KP.03) Risk Management Process

Instruction in the risk management process and the rational for tailoring the process for individuals or teams performing risk management.

(INF.FA.03.KP.04) Risk Management Methods

Instruction in risk management methods including risk survey, risk taxonomy, risk management templates, decision analysis and quality management techniques.

(INF.FA.03.KP.05) Risk Management Tools

Instruction in risk management tools including risk database, spreadsheet, tornado diagrams and expert systems.

(INF.FA.03.KP.06) Risk Management Metrics

Instruction in risk management metrics including statistical analysis, quantitative process improvement, and cost/benefit analysis.

(INF.FA.04) Verify Compliance

A focus area for the infrastructure dimension.

The purpose of **Verify Risk Management Compliance** is to ensure the project adherence to its Risk Management Plan. **Verify Risk Management Compliance** involves an independent audit of the risk management activities, training, and process. A report is generated to document findings.

(INF.FA.04.KP.01) Review Risk Management Plan	L3
(INF.FA.04.KP.02) Audit Agents and Artifacts	L3
(INF.FA.04.KP.03) Generate Audit Report	L3
(INF.FA.04.KP.04) Distribute Audit Report	L3
(INF.FA.04.KP.05) Track Action Items	L4

(INF.FA.04.KP.01) Review Risk Management Plan

Review project Risk Management Plan to understand the activities, agents and artifacts of the Risk Management Plan to prepare for a compliance audit.

(INF.FA.04.KP.02) Audit Agents and Artifacts

Audit the agents and artifacts of project risk management activities and record the results. Audits verify whether planned activities are conducted, participants are trained, and adherence to Risk Management Plan.

(INF.FA.04.KP.03) Generate Audit Report

Generate project risk management audit report by noting implementation performance and discrepancies against the Risk Management Plan.

(INF.FA.04.KP.04) Distribute Audit Report

Distribute audit report to the organization and project management to provide visibility into project risk management performance.

(INF.FA.04.KP.05) Track Action Items

Track project risk management audit report action items until closure.

(INF.FA.05) Improve Practice

A focus area for the infrastructure dimension.

The purpose of **Improve Risk Management Practice** is to systematically evolve risk management capability by developing a method for continuous improvement. **Improve Risk Management Practice** involves assessing the risk management capability and defining the vision and roadmap for increasing in capability. Developing and implementing improvement action plans then evaluating feedback provide an iterative method to ensure continuous improvement.

(INF.FA.05.KP.01) Assess Capability	L4
(INF.FA.05.KP.02) Develop Technology Roadmap	L3
(INF.FA.05.KP.03) Develop Improvement Plan	L3
(INF.FA.05.KP.04) Obtain Resources	L4
(INF.FA.05.KP.05) Implement Improvement Plan	L3
(INF.FA.05.KP.06) Evaluate Feedback	L4

(INF.FA.05.KP.01) Assess Capability

Assess organizational risk management capability by planning and applying an appraisal method and reporting the findings to establish an organizational baseline for improvement.

(INF.FA.05.KP.02) Develop Technology Roadmap

Develop risk management technology roadmap by defining the organizational vision, goals, and strategy for evolving risk management capability. The roadmap should project 3-5 years into the future. The roadmap should be realistic and based on the risk management capability of the organization.

(INF.FA.05.KP.03) Develop Improvement Plan

Develop risk management improvement plan by defining the specific areas to be improved, a schedule, budget and goals to be achieved. The improvement plan should provide the detail required for the next year's activity. The improvement plan should be realistic and based on given constraints.

(INF.FA.05.KP.04) Obtain Resources

Obtain resources required to implement the organizational risk management improvement plan, such as funding, staff, and computing resources.

(INF.FA.05.KP.05) Implement Improvement Plan

Implement a risk management improvement plan by involving people on projects as required to promote buy-in from the organization. Improvement plans should focus on the organizational evolution of risk management technology that will be leveraged to satisfy project's risk management needs.

(INF.FA.05.KP.06) Evaluate Feedback

Evaluate feedback obtained from implementing a risk management improvement plan to provide an iterative method to ensure continuous improvement and increasing organizational risk management capability.

(IMP.FA.01) Establish Program

A focus area for the implementation dimension.

The purpose of **Establish Risk Management Program** is to provide the context for performing risk management that is integrated within a project. **Establish Risk Management Program** involves review of requirements from the customer and organization, and planning for risk management activities by allocating schedule, budget, and staff. Training is coordinated for project participants to encourage their involvement in risk management activities.

(IMP.FA.01.KP.01) Review Risk Management Requirements	L2
(IMP.FA.01.KP.02) Plan Risk Management Activities	L2
(IMP.FA.01.KP.03) Schedule Risk Management	L2
(IMP.FA.01.KP.04) Budget Risk Management	L3
(IMP.FA.01.KP.05) Staff Risk Management	L4
(IMP.FA.01.KP.06) Coordinate Risk Management Training	L3

(IMP.FA.01.KP.01) Review Risk Management Requirements

Review the project's risk management requirements from the customer, statement of work, and organization. Determine the need for risk management based on the project size, budget, and complexity.

(IMP.FA.01.KP.02) Plan Risk Management Activities

Plan risk management activities on the project according to the scope of established risk management needs and requirements.

(IMP.FA.01.KP.03) Schedule Risk Management

Schedule risk management activities on the project master schedule.

(IMP.FA.01.KP.04) Budget Risk Management

Budget for risk management activities identified on the project master schedule.

(IMP.FA.01.KP.05) Staff Risk Management

Staff risk management activities by involving the appropriate mix of people and encourage participation from customer, management, and the project team.

(IMP.FA.01.KP.06) Coordinate Risk Management Training

Coordinate risk management training for project participants to increase their ability to perform risk management and encourage their involvement in risk management activities.

(IMP.FA.02) Develop Plan

A focus area for the implementation dimension.

The purpose of **Develop Risk Management Plan** is to determine the approach to performing risk management cost-effectively on the project. The Risk Management Plan provides the implementation procedures that are followed on the project. **Develop Risk Management Plan** involves defining the approach, structure, process, methods, tools, and metrics used to implement risk management on the project. The Risk Management Plan is peer reviewed, documented, distributed, and maintained.

(IMP.FA.02.KP.01) Define Risk Management Approach	L2
(IMP.FA.02.KP.02) Define Risk Management Structure	L2
(IMP.FA.02.KP.03) Define Risk Management Process	L2
(IMP.FA.02.KP.04) Define Risk Management Methods	L2
(IMP.FA.02.KP.05) Define Risk Management Tools	L2
(IMP.FA.02.KP.06) Define Risk Management Metrics	L2
(IMP.FA.02.KP.07) Peer Review Risk Management Plan	L3
(IMP.FA.02.KP.08) Document Risk Management Plan	L3
(IMP.FA.02.KP.09) Approve Risk Management Plan	L3
(IMP.FA.02.KP.10) Distribute Risk Management Plan	L3

(IMP.FA.02.KP.01) Define Risk Management Approach

Define the project's risk management approach such as proactive, integrated, quantitative, and systematic.

(IMP.FA.02.KP.02) Define Risk Management Structure

Define risk management structure in terms of responsibility and authority.

(IMP.FA.02.KP.03) Define Risk Management Process

Define the project risk management process by tailoring the organizational standard risk management process.

(IMP.FA.02.KP.04) Define Risk Management Methods

Define the project risk management methods such as techniques for communication, prioritization, and consensus. Existing methods include Taxonomy Based Questionnaire and Nominal Group Technique.

(IMP.FA.02.KP.05) Define Risk Management Tools

Define the project risk management tools such as a risk database and spreadsheet software.

(IMP.FA.02.KP.06) Define Risk Management Metrics

Define the project risk management metrics such as number of identified risks, risk exposure, risk reduction leverage, risk reduction cost, and overall savings.

(IMP.FA.02.KP.07) Peer Review Risk Management Plan

Peer review the draft project Risk Management Plan to promote understanding and identify problems with the draft. Incorporate feedback and action items from the draft project Risk Management Plan peer review.

(IMP.FA.02.KP.08) Document Risk Management Plan

Document the project Risk Management Plan in a format that is easily understood and maintained, with revision history and current version and date.

(IMP.FA.02.KP.09) Approve Risk Management Plan

Approve the project Risk Management Plan through the appropriate management and key technical staff to promote support and commitment from the top-down and bottom-up.

(IMP.FA.02.KP.10) Distribute Risk Management Plan

Distribute the project Risk Management Plan to the customer and project team to promote awareness, understanding and provide a common reference for the project to follow.

(IMP.FA.03) Tailor Standard Process

A focus area for the implementation dimension.

The purpose of **Tailor Standard Risk Management Process** is to define the risk management process for a specific project. Unique aspects of a project are addressed, such as size, budget, and structure. **Tailor Standard Risk Management Process** involves reviewing the organization standard process and recommending changes to custom fit a cost-effective process for the project. Deviations from the organization standard process are documented as waivers. The defined risk management process is peer reviewed, documented, approved and distributed to the project team.

(IMP.FA.03.KP.01) Review Standard Process	L2
(IMP.FA.03.KP.02) Define Process for Project	L3
(IMP.FA.03.KP.03) Peer Review Draft Defined Process	L3
(IMP.FA.03.KP.04) Document Defined Process	L2
(IMP.FA.03.KP.05) Approve Defined Process	L3
(IMP.FA.03.KP.06) Distribute Defined Process	L3

(IMP.FA.03.KP.01) Review Standard Process

Review the organization standard process and recommend changes to custom fit a cost-effective risk management process for the project.

(IMP.FA.03.KP.02) Define Process for Project

Define the project's risk management process by tailoring the organization standard process based on an understanding of the unique aspects of the project, such as size, budget, and structure.

(IMP.FA.03.KP.03) Peer Review Draft Defined Process

Peer review the project's defined risk management process to promote understanding and identify problems with the draft. Incorporate feedback and action items from the project's defined risk management process peer review.

(IMP.FA.03.KP.04) Document Defined Process

Document the project's defined risk management process in a format that is easily understood and maintained, with revision history and current version and date.

(IMP.FA.03.KP.05) Approve Defined Process

Approve the project's defined risk management process through the appropriate management and key technical staff to promote support and commitment from the top-down and bottom-up. Obtain waivers for deviations from the organizational standard risk management process to promote support and commitment from the organization management.

(IMP.FA.03.KP.06) Distribute Defined Process

Distribute the project's defined risk management process to the project team and customer to promote awareness, understanding and provide a common reference for the project to follow.

(IMP.FA.04) Assess Risk

A focus area for the implementation dimension.

The purpose of **Assess Risk** is to identify and analyze project risk. **Assess Risk** involves identifying programmatic and technical project risk and source of risk. Risk impact and likelihood are estimated, evaluated, and prioritized according to a defined process.

(IMP.FA.04.KP.01) Conduct Risk Assessment	L2
(IMP.FA.04.KP.02) Develop Candidate Risk List	L3
(IMP.FA.04.KP.03) Define Risk Parameters	L2
(IMP.FA.04.KP.04) Document Identified Risk	L3
(IMP.FA.04.KP.05) Communicate Identified Risk	L3
(IMP.FA.04.KP.06) Estimate and Evaluate Risk	L4
(IMP.FA.04.KP.07) Prioritize Risk	L2

(IMP.FA.04.KP.01) Conduct Risk Assessment

Conduct an independent risk assessment to provide a baseline of assessed risks to the project. Involve all levels of the project to train the risk assessment methods that will be used throughout the project.

(IMP.FA.04.KP.02) Develop Candidate Risk List

Develop the candidate list of risks that will be assessed by reviewing a risk taxonomy, work breakdown structure, or a previously developed checklist of risk areas. Brainstorming techniques may be used in addition to structured risk checklists. The focus is on identification of risk and source of risk in areas such as programmatic (cost, schedule, staff) and technical (performance).

(IMP.FA.04.KP.03) Define Risk Parameters

Define risk parameters such as taxonomy classification, statement of risk or source of risk, with an impact and likelihood rating.

(IMP.FA.04.KP.04) Document Identified Risk

Document identified risk by submitting a risk identification form to the appropriate project authorities.

(IMP.FA.04.KP.05) Communicate Identified Risk

Communicate identified risk to appropriate project personnel to increase awareness of project issues in a timely manner. Logging risks in a Risk Database and use of automated mail servers is one mechanism to facilitate communication of identified risks.

(IMP.FA.04.KP.06) Estimate and Evaluate Risk

Estimate risk impact and the likelihood of risk occurrence to establish a category of risk severity. Evaluate risk impact and the likelihood of risk occurrence in relation to other project risks. Include the risk time frame.

(IMP.FA.04.KP.07) Prioritize Risk

Prioritize risk based on the potential impact to the project and likelihood of occurrence according to a documented prioritization scheme. Adjust risk priority as additional information becomes available.

(IMP.FA.05) Manage Risk

A focus area for the implementation dimension.

The purpose of **Manage Risk** is to develop and execute risk reduction plans and track risk status to control project risk. **Manage Risk** involves planning for risk reduction by developing alternative strategies, selecting an approach, and planning for reducing risk impact and/or likelihood of occurrence. Upon approval of the risk reduction plan, resources are assigned and the plan is executed. Progress is monitored by tracking risk status, reporting risk results, correcting for variations, and process improvement.

(IMP.FA.05.KP.01) Develop Risk Reduction Alternatives	L2
(IMP.FA.05.KP.02) Select Risk Reduction Approach	L3
(IMP.FA.05.KP.03) Develop Risk Reduction Plan	L3
(IMP.FA.05.KP.04) Execute Risk Reduction Plan	L4
(IMP.FA.05.KP.05) Monitor Risk Status	L4
(IMP.FA.05.KP.06) Take Corrective Action	L5
(IMP.FA.05.KP.07) Obtain Risk Management Feedback	L5

(IMP.FA.05.KP.01) Develop Risk Reduction Alternatives

Develop risk reduction alternative approaches to reducing risk to an acceptable level. Alternatives may include risk avoidance, risk transfer, contingency planning and buying information.

(IMP.FA.05.KP.02) Select Risk Reduction Approach

Select risk reduction approach based on selection criteria to reduce risk impact and/or likelihood of risk occurrence. Criteria may include minimization of impacts to cost, schedule, performance, or customer

satisfaction. Selection decisions may be based on assumptions, constraints, and historical data, which may be revised as information becomes available.

(IMP.FA.05.KP.03) Develop Risk Reduction Plan

Develop a risk reduction plan. The plan may include objectives, approach, start date, milestones, due date, responsible person, resources required, and authorization signature.

(IMP.FA.05.KP.04) Execute Risk Reduction Plan

Execute the risk reduction plan. Maintain the plan by documenting risk reduction actions taken and results achieved.

(IMP.FA.05.KP.05) Monitor Risk Status

Monitor risk status by tracking metrics and triggering events, reviewing and reporting risk results. Techniques for monitoring risk status may include risk tracking metrics (such as time, effort, budget, milestones, function points or lines of code), Top-10 List, and Technical Performance Measurement (TPM).

(IMP.FA.05.KP.06) Take Corrective Action

Take corrective action according to a documented procedure as required to correct for variations from risk reduction plans.

(IMP.FA.05.KP.07) Obtain Risk Management Feedback

Obtain feedback on the risk management program by using a risk management survey to obtain perceptions of the organization to measure and evaluate performance and improve the risk management process.

Appendix D - RMS to RMEF Mapping

Dimensions		Risk Management Evolution Framework				
	Elements	1 – Problem	2 – Mitigation	3 – Prevention	4 – Anticipation	5 – Opportunity
PROCESS	Identify		42	43	44	45
	Analyze		46, 47	48	49	50
	Plan		51	52	53	54
	Track		55	56	57	58
	Control		59	60	61, 62	63
INFRASTRUCTURE	Policy		21, 22	23	24	25
	Communicate		26	27	28	29
	Commitment	30	31		32	33
	Resources		34	35	36	37
	Training		38	39	40	41
IMPLEMENTATION	Participants	64	65	66, 69	67, 70	68, 71
	Procedures		72	73	74	75
	Methods		76	77	78	79
	Tools		80	81	82	83
	Metrics		84	85	86	87

Glossary

benchmark - A reference point or standard by which products, practices or performance may be judged.

benchmarking - The process of comparing and measuring to gain information which will help an organization take action to improve its performance.

best practice - An enabler for excellent performance in a process.

cost drivers - Requirements complexity, personnel, reusable software, tools, etc.

cost risk - The degree of uncertainty associated with cost estimates creates a risk of overrunning the budget, known as cost risk.

cost risk control - The process of achieving the desired cost outcome by continual application of risk management to the cost drivers. (e.g. combining historic data, cost estimation model, and refining cost model input assumptions by evaluations, avoidance, control, assumption, or transfer).

decision analysis - Use of decision trees, influence diagrams, and other techniques to characterize options by their possible outcomes in terms of risk exposure.

evaluation - Assessment using defined evaluation criteria.

failure - A departure of a computer program's operation from the user's requirements.

impact of risk - The consequences of risk occurrence.

likelihood of occurrence - The probability that the risk will occur.

method - A regular, orderly, definite way of doing something.

migration strategy - An evolutionary approach to adopting a new technology.

mitigate - To reduce the probability and/or consequence of a risk.

mitigation approach - Means of reducing risk such as avoidance, or transfer.

mitigation plan - An action plan to reduce the probability and/or consequence of a risk.

model - A generalized description used in analyzing or explaining something. A standard of excellence to be imitated.

nontechnical risk - The risk of not meeting program goals such as cost, schedule and profit. The risk of not meeting all other program success criteria besides technical goals.

orientation - An overview or introduction to a topic.

panacea - A supposed remedy, or medicine for all diseases or ills. A cure-all.

performance risk - The degree of uncertainty in the development and deployment process that may keep the system from meeting its technical specifications or that may result in the system being unsuitable for its intended use [AFSC88].

proactive - For action, not reaction. Favorably causing action or change.

proactive approach to risk management - A method to empower project teams to identify and mitigate risks characterized by team participation and customer involvement.

proactive risk management - Actively attacking risks. Acting to identify, assess, and manage risks to prevent problems and create opportunities.

procedure - A description of a course of action to be taken to perform a given task.

process - A set of activities that transform inputs to outputs.

program management - The organization responsible for the execution of the project.

quality - Providing customers with products and services that fully satisfy their requirements. American approach to quality is product-oriented (get the defects out). Japanese approach to quality is people-oriented (Kaizen, or continuous improvement).

quality assurance - An organization responsible for ensuring quality standards are met.

quality control - Methods by which quality is measured, reported and improved.

rework - The cost of not doing something right the first time.

risk (1) - A measure of the probability and severity of a bad outcome. A risk is a potential problem usually caused by lack of information, control, or time. To be considered a risk, there must be uncertainty or change, a choice and a potential loss associated with an action or event.

risk (2) - A measure of the uncertainty of attaining a goal, objective, or requirement pertaining to technical performance, cost, and schedule. Risk level is categorized by the probability of occurrence and the consequences of occurrence. Risk is assessed for program, product, and process aspects of the system. This includes the adverse consequences of process variability. The sources of risk include technical (e.g., feasibility, operability, producibility, testability, and system effectiveness); cost (e.g., estimates, goals); schedule (e.g., technology/material availability, technical achievements, milestones); and programmatic (e.g., resources, contractual) [MIL-STD 499B].

risk abatement - The process of reducing the amount of risk to a system [AFSC88].

risk analysis (1) - Evaluation and estimation of risk with respect to its consequence and probability of occurrence.

risk analysis (2) - Examining the change of outcomes with the modification of the risk drivers. This examination is more involved than risk assessment and should result in the identification of the most critical variables, with insights into desired options for risk handling [AFSC88].

risk assessment - A process that identifies risk and evaluates risk based on established criteria, such as likelihood of occurrence, consequences, and time frame for action.

risk control (1) - The process of implementing risk reduction plans and correcting for deviations from the plan.

risk control (2) - The process of achieving the desired outcomes by continual application of management techniques to the risk drivers [AFSC88].

risk database - The repository of identified risks and associated information.

risk drivers - Those variables that cause probabilities of cost, schedule, performance, or support risk to fluctuate significantly [AFSC88].

risk exposure - Risk Exposure (RE) = Probability * Loss.

risk handling - The identification of options available to reduce or control selected risk drivers [AFSC88].

risk identification - The process of communicating known risk and sources of risk.

risk impact - See risk exposure.

risk leverage - See risk reduction leverage.

risk management (1) - Informed decision-making under uncertainty that deals with the future of present decisions.

risk management (2) - An organized, analytic process to identify what can go wrong, to quantify and assess associated risks, and to implement/control the appropriate approach for preventing or handling each risk identified [MIL-STD 499B].

risk management activities - Actions that support the risk management program.

risk management approach - The strategy for implementing the risk management program.

risk management capability - The range of expected results that can be achieved by implementing a risk management process within an organization.

risk management maturity - The extent to which risk management capability is fully developed in an organization.

risk management paradigm - A model that shows the structure and components of risk management.

risk management performance - A measure of the actual results achieved by the application of risk management.

risk management plan - The documented objectives, organization, and methods for performing risk management.

risk management process - A systematic and structured way to manage risks that includes the activities and mechanisms used to transform program knowledge into decision-making information.

risk management program - The strategy, plans, organization, process, and procedures used to implement risk management.

risk mitigation - Reducing risk by decreasing its consequence and/or probability of occurrence.

risk mitigation activity - Action taken to reduce the impact and/or likelihood of a risk.

risk mitigation strategy - The identification of one or more risk mitigation activities so that the important activities are efficiently and effectively performed.

risk planning - The process of determining and evaluating alternative approaches to reducing risk and documenting the selection in a risk reduction plan.

risk reduction - Any gain in relevant knowledge to decrease uncertainty, risk impact and/or likelihood of occurrence.

risk reduction actions - Any act that attempts to mitigate risk.

risk reduction alternatives - The set of options that may reduce risk if implemented.

risk reduction cost - The cost of implementing the risk reduction plan.

risk reduction leverage - RRL = (RE (before) - RE (after)) / Risk Reduction Cost.

risk reduction plan - The objectives, constraints, and alternatives for reduction of a risk. The risk reduction plan documents the selected approach, triggering mechanisms, resources required, approval authority, and reduction results.

risk tracking - The process of monitoring and maintaining risk status.

role - A unit of defined responsibilities that may be assumed by one or more individuals.

schedule drivers - Resources (e.g. personnel, facilities, and budget), need dates, dependencies, and requirements.

schedule risk (1) - The degree of uncertainty associated with estimating schedule dates creates a risk of not meeting the desired end date, known as schedule risk.

schedule risk (2) - The degree of uncertainty associated with the ability of the program to achieve desired milestones (outcomes) on time [AFSC88].

schedule risk control - The process of achieving the desired schedule outcome by continual application of risk management to the schedule drivers, such as assessing the impact of schedule drivers, historical data, assumptions, and comparing baselines with actuals, then determining the probability that the desired schedule will not be achieved.

silver bullet - A panacea for the software crisis.

software - Code that is developed to execute in a computing system.

software crisis - Problems in the software community that have led to late and over-budget software systems that do not satisfy the intended user community.

software development - All activities required to create a software product.

software development lifecycle - The period of time from software product conception through software operations and maintenance.

software engineering - A discipline for software development and maintenance.

software engineering paradigm - The lifecycle model for software development.

software engineering process - The set of software engineering activities needed to transform a user's requirements into operational software.

software maintenance - All activities required to maintain a software product.

software measures - A dimension, attribute or amount of some aspect of software. A measured quantity.

software practices - Repeatable, consistent procedures following in the process of developing or maintaining software.

software process - The set of activities and practices used to develop and maintain software and associated software products.

software project management - The process of planning, organizing, staffing, monitoring, controlling, and leading a software project.

software reliability - The probability of failure-free operation of a software system for a specified time. Reliability uses statistical analysis to determine the likelihood that a software failure will occur.

software risk - A measure of the probability of an unsatisfactory outcome affecting a software project and the consequence of occurrence.

software risk management - Application of risk management technology to managing and developing software systems.

software safety - Identification and assessment of potential hazards that may impact software systems. Avoidance of system safety failures caused by software errors, which may lead to casualties or serious consequences. The probability that conditions that can lead to a mishap do not occur.

spiral model - A software engineering paradigm that is a risk-driven evolutionary approach to development for large software systems.

statement of work - A description of all work required to complete a project.

statistical decision theory - A generalized decision-oriented risk analysis technique that provides the capability to analyze situations that involve the buying of information to reduce risk.

support risk - The degree of uncertainty associated with the ability of the support organization to maintain, change, or enhance software of the fielded system within the planned support concepts and resources [AFSC88].

system - A set of components organized to accomplish specific functions.

tailor - To modify a process, standard, or procedure to better match process or product requirements.

technical drivers - Requirements, constraints, technology, and development approach.

technical risk - Aspects of a system development that may cause an impact on feasibility, operability, producibility, testability, and system effectiveness. Technical risk includes all activities in engineering the product (requirements, design, code, test and integration), as well as the development environment (see also performance risk).

technical risk control - The process of achieving the desired technical outcome by continual application of risk management to the technical drivers.

technology - The application of science to a specific problem.

technology transition - The process of planning and facilitating changes, and understanding the behavior of individuals and groups that results in the routine use of a new technology.

top ten technical risks - The most significant risks to the development effort.

total quality management - The vision, guiding principles and philosophy that form the foundation of continuous improvement in an organization.

train - To make proficient with specialized instruction and practice.

uncertainty - Having outcomes with unknown probabilities of occurrence.

utility theory - Theory of preference under conditions of risk.