An Interpretive Study of Software Risk Management Perspectives

by

Keshnee Padayachee

submitted in fulfilment of the requirements for the degree of

MASTER OF SCIENCE

in the subject of

INFORMATION TECHNOLOGY

at

UNIVERSITY OF NATAL

SUPERVISOR : PROFESSOR D PETKOV

MAY 2002

PREFACE

This research was conducted on a part-time basis between 1999 and 2002 at the school of Mathematics, Statistics and Information Technology at the University of Natal, Pietermaritzburg, under the supervision of Professor D Petkov. The results are the original work of the author and have not been submitted for any degree at any other tertiary institution.

ABSTRACT

This dissertation addresses risk management in the software development context. The discussion commences with the risks in software development and the necessity for a software risk management process. The emergent discourse is based on the shortfalls in current risk management practices, elaborated in the software risk management literature. This research proposes a framework for a field investigation of risk management in the context of a particular software development organization. It was experimentally tested within several companies. This framework was designed to provide an understanding of the software development risk phenomena from a project manager's perspective and to understand how this perspective affects their perception. This was done with respect to the consideration of the advantages and disadvantages of software risk management as regards its applicability or inapplicability, respectively. This study can be used as a precursor to improving research into the creation of new software risk management frameworks.

CONTENTS

CHAPTER 4

TECHNIQUES USED IN RISK ANALYSIS OF SOFTWARE DEVELOPMENT

CHAPTER 5

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGMENTS

I am grateful to all of those people who contributed their time, effort, and expertise to help produce this dissertation. I would like to thank: Mr F Wahab, Dr Vermeulen, Mr J Van Staden, Mr A Filmalter, Ms R Du Plooy, Mr P Geldenhuys and Mr H Botha who, despite their busy schedules, contributed to the research data by allowing me to interview them.

The following people provided emotional support and advice: my husband, Devern Padayachee, my family, and my colleagues at UNISA, Mrs E Kritzinger, Professor P Kotzé and Professor J P Van Der Poll.

Thanks also go to Ms M M Malan for locating and providing literary sources of relevant information.

And finally a special thanks to Professor D Petkov for intellectual support, encouragement and vision.

CHAPTER 1

INTRODUCTION

'All projects involve risk - the zero-risk project is not worth pursuing' (Chapman and Ward, 1997:8).

1.1 Background to the Research Problem

The increasing reliance on software systems and the increasing amount of software in systems (Eldridge, 1991:820) is an indication that overcoming the chronic problems of software development such as cost overruns, project delays and unmet user requirements (Ropponen and Lyytinen,1997:41) is not only highly desirable but a priority for the economy. According to Klein and Jiang (2001:195), studies continue to indicate that about 85% of all projects end in failure. Furthermore, it is estimated that 31.1% of projects will be cancelled before they are ever completed (Boehm, 2000a:94). Software project failure is a global epidemic with no limitations irrespective of the revolutionary influx of new and improved standards, languages, methods and tools (Hall,1998). The fact is, that as long as humans design software, no technological advance can significantly change the nature of software to such an extent that it becomes a clear and definable process to all stakeholders in a software project.

Risks to software development can be reduced with reasonable expectations from technology. In Brooks'(1986) article, entitled "Essence and Accidents of Software Engineering", he highlights four inherent characteristics of software that no "silver bullet" can overcome. These characteristics still hold true despite all the technological advances made and all attempts to reduce the agents of chaos in software development (Olson,1993), namely its changeable, invisible, conformable and complex properties.

The creation of software is complex as each software system is unique and the structure is non-repetitive (Sherer,1992). A software engineer is expected to build systems for any type of discipline, for example accounting systems, tax systems and flight control systems and is expected to gain specialised knowledge in any field. Jackson (1998) suggests that product-specific knowledge should be advocated in software engineering, as practiced in other disciplines.

Software's changeability characteristic arises from the fact that the engineering of software, unlike other forms of engineering, has no foundation in physical laws. The source of structure for software engineering is in standards and policies which are defined by teams of experts. Consequently software is very flexible and can easily be redefined (Chittister,1993). The software engineer is expected to make any changes decreed, as software is seen as being conducive to changes. Thus 'creeping user requirements' are major challenges faced by software engineers and place 80% of projects at risk (Jones, 1994).

Software's conformable peculiarity arises on account of human perceptions according to which, (unlike other disciplines) software is expected to conform to changes because it is the newest arrival on the scene (Schach,1993). Even if a company has a poor business strategy, the software engineers have to conform to it, rather than formulate a credible business solution together.

Software's invisible nature arises from the difficulty in depicting the entire system in a manner that is clear to both the developer and the client. As software increases in magnitude and complexity, it becomes increasingly difficult to visualise. The nature of software is such that underlying risks remain invisible until it is too late (Brown,1996:95).

Software development, given its diverse and abstract nature, offers unique challenges and risks (Moynihan,1997). Software remains an elusive entity that is difficult to manage, control or change. Therefore software development is an inherently risky process because there are no guarantees that a software product will be delivered on time, within budget, fulfill its requirements and perform faultlessly (Lam and Vickers,1997). A formal risk management programme is a structured way to evaluate risks to the software development process. It involves identifying and analysing the risks to a project and then implementing and monitoring measures to reduce these risks. As a discipline, risk management has existed in many industries for decades, but it was not defined for the software industry until the late 1980s (Kuver,1999).

The first consideration in any risk management framework is the identification of risks. A risk is any variable within a project that may result in project failure. The project can be threatened by political, communication, schedule, legal, and technical risks (Lister, 1997). According to Pressman (1997) there has been considerable debate regarding the proper definition of software risk, but the general agreement is that risk always involves two characteristics:

- *uncertainty* - the event that characterizes the risk may or may not happen, i.e. there are no 100% probable risks.
- *loss* - if risk becomes a reality, unwanted consequences or losses will occur.

By contrast, Kuver (1999) defined risk as an event having three elements associated with it:

- *Chance* - there is a probability that the event will occur.
- *Consequence*- there will be a negative impact of some kind on the project.
- *Choice*- means that there are alternatives to the event.

The two definitions above are similar, except for the third component, choice. The risks in software development are not pure risks where there is a definite downside. The risks are speculative in nature, meaning that if a risk is dealt with correctly there will be a payoff (Voas, et al,1997). Risks in software development can be averted through choosing the least risky alternative.

The second consideration in risk management is: why do risks exist in software development? The risks in the software industry are inherently related to the changeable, invisible, conformable and complex nature of software. According to Maude and Willis (1991), risks which occur in software development may be due to the following factors:

- It is difficult to determine requirements.
- The requirements specification may change during development.
- It is difficult to estimate the costs and resources.
- There is insufficient information at the start of development.
- The development itself may carry technical risks.

The third consideration is related to the necessity of risk management in the software management plan. Failure in projects is attributed to the manifestation of risks, which implies that practising risk management will result in project success. Powell and Klein (1996) state that acceptance of this argument may appear tantamount to an act of "blind faith" but employing risk management techniques in other disciplines has proved useful. In order to ensure project success, it is vital that risk management becomes an intrinsic part of the software development life cycle. An example in that direction, is the spiral model, which is discussed in chapter Two.

Software risk management is more than a decade old. If it is a solution to the software crisis, the question is: why does the software crisis go on unabated? From Gemmer's (1997) supposition, the problem is the repercussions resulting from treating risk management as just another process thereby overlooking the influences of risk perceptions and the risk propensities of individuals. Treating software risk management as if it were in the same category as any other process, does not take cognisance of the fact that individuals interpret the laws governing software differently (Chittister,1993).

Many organizations are unable to manage risks effectively for any of the following three reasons: Firstly, the existence of a risk-averse culture that rewards crisis management and reprimands those who identify risks to the project's success; secondly, the absence of an infrastructure to support risk management effectively; thirdly, the lack of systematic and repeatable methods to identify, analyse and plan risk mitigation (Carr, 1997). Therefore, even if software risk management is applied, it is applied on an ad-hoc basis which defeats the purpose of applying it in the first instance, as it serves only to compound the problem and to perpetuate negative attitudes towards risk management.

1.2 Goal and Subgoals of the Research

It is difficult to ascertain whether risk management is the determining factor in project success or whether a smaller and less complex project would have a higher probability of success anyway. Research conducted on small controlled projects has proved risk management to be effective

(Ropponen and Lyytinen,1997:42). However, within the software development environment, it is difficult to have controlled experiments.

The main goal of this research is to provide an interpretivist framework for determining the effectiveness of risk management in field conditions and to apply it to an exploratory investigation of risk management practices in several software organizations.

Subgoals that contribute to the main goal above can be identified as:

- To provide an overview of software process models and their support for risk assessment in software development.
- To provide an overview of risk management frameworks according to the existing theory and practice of software construction.
- To provide an overview of analytical techniques for risk assessment in software development.
- To formulate an interpretive investigation framework to determine how project managers "perceive" the effectiveness of software risk management.

  The subgoals involve determining:

  - The factors that result in successful risk management practices.
  - The factors that hinder successful risk management.
  - The factors that result in "risk-aversive attitudes".
  - The factors that foster risk-aware attitudes.
- To test the framework in field conditions in particular software organizations and to interpret the results within the context of those organizations.

1.3 Scope and Delimitations

Software risk management is a very broad field and covers two basic areas: software development risks and information systems risks. The former arises out of risk events that negatively impact the development processes and, if neglected, result in massive product changes (Adler et al, 1999). The latter, encompasses the risks that the "owners or users" of an information system are subjected to (Guarro,1987).

According to Rainer et al (1991) the threats to "owners or users" of an information system emanate from three potential sources:

- from the external environment such as fires;
- from unauthorised physical or electronic access;
- from authorised physical or electronic access such as an increase in end-user computing;

In order for risks to the information system to occur there must be some interactions between the users of the system, the hardware and the software, or the software and the external environment (Bennett et al,1996). The scope of this study is limited to the stage before a system is fully operational and fully integrated as part of an entire information system. Thus, only risks occurring during software development are the issue under consideration in this research.

As this research is conducted within an interpretivist epistemology, the rationale for which is discussed below, the outcome is not intended to provide statistically generalizable results from the surveys conducted but rather to formulate an approach for gaining a deep insight into the current practices within particular organizations.

1.4 Research Methods

The software development process is complex and the best way to comprehend it is to draw information from the point of view of those who experience it. No systematic attempts have been made to tap the opinions of those who actually have experience in managing risks in projects with the exception of Moynihan (1997). An attempt to correct this situation has been presented in Keil et al (1998). The same authors report that previously studies have been conducted on anecdotal evidence or have been limited to a narrow portion of the development process (Keil et al,1998).

There are two basic areas of concern when it comes to conducting a "software risk management" study using a positivistic epistemology. Firstly a "software risk management study" cannot be conducted within controlled environments, as there are too many variables both known and unknown. The second issue is that software entities are more complex owing to their uniqueness (Albanna and Osterhaus,1998). Therefore, success in one project using a particular technique does not guarantee success in another project using the same technique. As an interpretive approach acknowledges differing contexts, it would be better suited to an environment in which generalizations are difficult to formulate.

From the abundant literature on positivism in research (Galliers,1987; Walsham,1993; Walsham,1995; Neumann,1997; Fitzgerald and Howcroft, 1998) the following commonalities in the characteristics of positivist research emerge:

- it warrants the use of precise quantitative data and rigorous exact measures;
- it warrants the use of experiments, surveys and statistics;
- it warrants the use of reduction to downscale complexity so that the cause and effects can be validated under laboratory conditions;
- the research must be repeatable in order to be validated;
- the notion that reality exists independently of an individual's construction of it and that the population shares the same meaning system, implies that all individuals experience the world in the same way;
- the formulation of statistical generalizations or cause-and-effect laws.

Most of the characteristics of positivism do not complement software risk management in practice and these beliefs are impediments to uncovering the real nature of risk management and gaining new perspectives on it. For instance, a positivist researcher requires precise quantitative measures, which are often not available in the risk management scenario. In fact, the measures used are usually predicative and indirect and this is contrary to the "precise measures" required by positivistic researchers. The complexity of a software project is not easily reflected by simple measures and most measures are tailored to suit the project at hand (Shepperd,1993) and therefore these measures cannot be easily generalized.

The second problem with positivism is the belief that a complex problem can be "reduced" like a mechanical system that can be broken down into smaller components. It is possible to "reduce" the complexity of an information system but making casual laws from these micro viewpoints cannot be extrapolated to the macro view due to the multivariant nature of the software development environment (Galliers,1987). There are also too many variables, both known and unknown, to arrive at such generalizations using laboratory conditions. 'For three centuries, mathematics and the physical sciences made great strides by constructing simplified models of complex phenomena, deriving properties from the models, and verifying those properties by experiment. This paradigm worked because the complexities ignored in the model were not the essential properties of the phenomena. It does not work when the complexities are the essence' (Brooks, 1986:11).

The third problem is the belief that social reality is not random (Neumann,1997) which is the complete opposite of the chaotic processes involved in software development. Software development's nature is risky and is hard to acknowledge in real-world situations (Boehm and DeMarco,1997:18).

The fourth problem is the replicability of results. As the world of software development is ill structured and fuzzy (Avison et al,1999:95), it is difficult to replicate the same set of circumstances. For example, say a project was successfully completed within ten months with ten people using a particular risk management strategy. Under the normal science paradigm, if such a situation were to be replicated on another project involving ten people working for ten months, using the same risk management strategy should result in a successful outcome too. But replicating this scenario is impossible, since even if the same project was attempted this does not guarantee that the project outcome will be the same. This probable outcome arises due to human behaviour which cannot be "replicated", precisely because it is human behaviour, which makes organizations unique and complex (Avison et al,1999:95). This underestimation of the human factor is the fifth problem. The positivist epistemology assumes that all individuals share the same system of meaning (Neumann,1997) thereby ignoring the human factor in software development.

Some authors argue that positivist research reduces people to numbers and deplore its concern with abstract laws or formulas that are not relevant to the lives of real people (Neumann,1997) leading to results that are usually inconclusive and inapplicable (Galliers,1987). Failure to include human factors may explain some of the dissatisfaction with conventional information system development methodologies, which do not address real organizations (Avison et al,1999:95). Contemporary project management has been heavily influenced by the normal science paradigm over the course of its theoretical and practical development (Charette,1996:113). It therefore follows that a radical shift away from a positivist view, not only in developing new methodologies but also in researching current software development practices, might be a possible solution. This will provide an intuitive way to modify existing methodologies and develop new ones.

An interpretive epistemology was chosen for this research as it takes into account the social factors in software development that positivism does not. Interpretive research can help to understand human thought and action in a socio-organizational context. Therefore, interpretive researchers do not report facts. They report their interpretations of other people's interpretation (Walsham,1995:78). This form of research can uncover how people construct meaning in their own environments. It does not simply reduce people to numbers or oversimplify the nature of humans but it is based instead on the ideology that people have justifiable rationales for their actions (Neumann,1997). An interview-based survey was conducted using a mixture of quantitative and qualitative approaches within an interpretive epistemology. The justification for the use of an interpretive approach manifests itself in expressing the rationale for rejecting the positivist epistemology for this study.

The normal science paradigm restricts the scientist to a specific paradigm, and whatever falls outside of this paradigm is ignored (Banville and Landry,1989:51). For the above reasons this research was conducted within the interpretivist paradigm. Further elaboration will be provided in chapter Five.

1.5 Relevance of Research

All software projects face the problem of quality, schedule and cost being affected by risks that are unexpected, unplanned or simply ignored. Information systems failure has recently gained prominence in the concerns of information systems professionals and the business community, and the pressure to reduce risks associated with development systems, is increasing. If software risk management is seen as a vehicle to immobilise risks, why then are the statistics surrounding project failure so astronomical? There are three possible reasons. Firstly, risk management is not being applied at an adequate level or risk management is not being practiced at all and, finally, risk management in its current practice is not as effective as expected. This research provides a framework for detecting problems with risk management in the context of an organization so that they can be identified and addressed. As a result the preconditions are created for reducing project failure.

1.6 Overview of the Structure of the Thesis

The theoretical and practical issues of risk management in software development are investigated first at macro level - one of the software development process model that may be followed for a given project. That is discussed in chapter Two. Within a particular process model various frameworks for risk management could be applied, as discussed in chapter Three. Within a given framework, techniques for risk assessment could be either quantitative or qualitative and these are investigated in chapter Three. Therefore chapters Two to Four investigate the issues related to software development risk management within the software industry as they were previously researched by others. The conclusion from the analysis in the second, third and fourth chapters shows the need for a framework for the investigation of risk management practices in field conditions that is interpretivist in nature but one that combines qualitative and quantitative techniques as a means of triangulation. The latter is needed for improvement of the quality of the result of the implementation of the framework. The fifth chapter formulates the framework for risk analysis in software development which combines the methods used in two previously published papers. This chapter justifies the epistemological stance taken and the research methodology used to uncover software development risk management perspectives and perceptions. The sixth chapter provides an analysis of the data collected in the exploratory field testing of the framework. The final chapter provides a summary of the results of this research and postulates possible ways for future work.

CHAPTER 2

RISK CONSIDERATION IN SOFTWARE PROCESS MODELS

"Successful project managers are good risk managers"(Boehm, 1991:33).

2.1 Introduction

Usually a software product begins with an idea and if this idea is feasible, the product is then specified, designed and implemented. Once the product has been installed, it will be maintained throughout its use. If the product outgrows its usefulness and can no longer be maintained to meet new demands, it is decommissioned. This series of steps through which a software product advances, is known as a process model or a life-cycle model. Every product has its own peculiar life-cycle. For instance, some products may spend years just in the conceptual stages as the technology does not yet exist for such a product to become viable.

There are six categories of risk that affect a software development project. According to Pressman (1997) these are project risks, technical risks, business risks, known risks, predictable risks and unpredictable risks. The six categories of risk have ramifications in every phase of the project's life- cycle (see Table 2.1). This chapter explores how software process models cater for these risks and how these risks affect the success of a software development project.

Developers use a myriad of process models to create software. A process model gives the overall "shape" to a project, 'a shape that is designed to make a project survive under the pressures it will face'(Ould,1999:13). Among the most popular are the waterfall model, the evolutionary model, the incremental model and the spiral model (Boehm et al, 2000). The spiral model receives considerable attention as it incorporates risk management. Additionally, to be discussed is the highly unsatisfactory build-and-fix "model", which is not a formal process model but a process that is followed by software developers. These models will be reviewed in the subsequent subsections.

Table 2.1: Depicting categories of risk that affect software development (Pressman,1997)

Project risks	Project risks threaten the project plan, e.g. regarding budgets, schedules, personnel, resources, clients, requirement problems and complexity.
Technical risks	Technical risks threaten the quality and time, e.g. maintenance problems, specification ambiguity and technical uncertainty
Business risks	Business risks threaten the viability of the software, e.g. budget risks.
Known risks	Known risks are those risks that can be uncovered after evaluation such as an unrealistic delivery date.
Predictable risks	Predictable risks are those risks that past experience has shown to occur, such as staff turnover.
Unpredictable risks	Unpredictable risks are those risks that are difficult to identify.

2.1.1 The Build-and-Fix Model

Quality and timeliness should be the aim of every software engineer, but practitioners often perceive a disciplined process as an impediment to rapid progress (Pressman,1996:16) which results in projects being built according to the "Build-and-fix model" where the product is constructed without any specifications or any attempt at design. Instead, the developers simply build a product and it is then reworked until the client is satisfied. This poorly defined and inconsistent application of software engineering practices often leads to increasing risk (Borcz,1996).

If the software development is unstructured then it follows that the risk management process will be unstructured too (Chittister,1993). Often a reactive risk management strategy is adopted in accordance with the ad-hoc nature of this model and crises are merely reacted to as they occur. But Chittister(1993) maintains that risk management can be made more systematic even if the software process is not.

2.1.2 The Waterfall Model

By the mid-70's, the software field had found a set of common anchor points, a sequence of milestones around which projects could be planned, organized, monitored and controlled. These milestones lead to the formulation of the waterfall model (Boehm,1996). The legacy of the waterfall model has not been a positive one despite adopting a highly structured approach where an organization will develop specifications, build a design from the specifications, implement, test, integrate and maintain the resultant system during its lifetime (Cardens-Garcia,1991). The build-and-fix approach failed because of a lack of structure, but simply adding structure as in the waterfall model does not guarantee success either.

The rationale behind the waterfall model is that every product must go through all of these phases and altering the order of the phases will produce a less successful product (Jalote,1997). A belief in the aforementioned assumption and other such assumptions (listed below) is termed "cognitive dissonance" (Charette,1996) and these assumptions are in conflict with risk management. They are based on foundations that are not necessarily true:

- A project is defined as a clear-cut investment activity with an explicit purpose and a distinct beginning, duration and end.
- At least one solution exists given the project's purpose, meaning that the project is feasible, suitable and acceptable.
- The time and resources can be accurately predicted.
- The environmental context is well-understood and fixed, and "success" can be defined and measured.
- The risk involved can be contained.
- Failure to meet the project's objective is caused by a lack of proper skills or their employment, rather than the infeasibility, unsuitability or unacceptability of the project.

It has become apparent that the waterfall model's milestones did not fit an increasing number of project situations (Boehm,1996:73). It fails to recognize political and contextual factors (Middleton,1999:174) and it does not accommodate software's special properties. It is inappropriate for solving partially understood issues (Blum,1992), as it assumes that the requirements specification are accurate (Royce,2000) at the outset. The insistence on sequential determination of the system's requirements, design and code (Boehm et al, 2000:120) coupled with conspicuous division of the phases in software development, according to Chapman and Ward (1997), pose the following risks:

- These steps and stages are difficult to distinguish in practice.
- Not all steps may be necessary in practice.
- The level of detail adds to the complexity, when what is required in practice is simplification.
- Usually moving onto the next phase before the previous stage is complete only compounds the risks in latter stages. At this point risk analysis could be very useful in decision making. The risks of going ahead can be weighed against the rewards sacrificed by not moving to the next phase.

In the waterfall model, the contractor is the sole recipient of all risks incurred. In response to this particular problem, the waterfall model has been given a new lease on life with the following changes advocated by Lott (1997) where each phase is allocated a fixed price contract:

Stage one:	Identify Requirements.
Stage two:	Build Design.
Stage three:	If it is possible to implement, then do so.

The advantages here are, that either party can terminate at any stage and the risks are shared. This process of fixed price allocation can be adopted by other models as well to reduce budget risks.

The waterfall model works well for custom-developed software where the requirements are fixed when development begins (Royce,2000:116) and in smaller developments (Cotton,1996). More often than not, it is impossible to determine the exact requirements at the outset of a project, hence the popularity of various evolutionary approaches toward system development such as the spiral model (Keil,1998).

By the early 80's, companies had realized that the waterfall model was ineffective for developing user-interactive systems due to problems in requirements determination. Prototypes rather than exhaustive specifications were found to be more effective in requirements gathering (Boehm,2000c:114). As Brooks (1986:17) asserts, prototyping is 'one of the most promising of the current technological efforts, and one that attacks the essence' of the software problem . Some prototypes are just developed to assist in writing up the requirements specification and then discarded (Sommerville,1985). Other prototypes may become the entire system, as it happens with the evolutionary delivery process model (Ould,1999).

2.1.3 The Evolutionary Delivery Process Model

In this model the user's needs and the system requirements are partially defined up-front and then refined in each successive build (Cotton,1996). Therefore this model should be considered when the final form of the system cannot be decided upon until something has been tried, or the exact relationship between the system and the business is complex or may change (Ould,1999). The evolutionary model reduces risk by breaking the product down into smaller more manageable pieces and thereby increasing the visibility of the management team. In this way, say May and Zimmer (1996), the following problems can be addressed:

- Missing deadlines
- Unusable products
- Wrong feature sets
- Poor quality.

Unlike the waterfall model, this process does not follow the sequential processes of specifying requirements, designing, implementing, integrating and testing. Instead this entire cycle evolves throughout the process (Royce,2000:121) as it divides the development cycle into smaller incremental waterfall models (May and Zimmer,1996). The evolutionary model defers the full definition of future increments in favour of developing an initial core capability but according to Boehm(2000) this ideology poses the following risks:

- The initial release is optimized for initial demonstration which may not be able to scale up to the next transition.
- The initial release may defer major considerations on security, fault tolerance and the client may expect the expediency achieved in the initial release to be maintained thereby leading back to the first problem .
- Sometimes the first release is totally off the mark because of the lack of user activity.

The problem with evolutionary development is succinctly expressed by Boehm (1991:32) who states 'The code-driven, evolutionary development process model tempts people to say, "Here are some neat ideas I'd like to put into this system. I'll code them up, and if they don't fit other people's ideas, we'll just evolve things until they work". This works fine in some well-supported mini domains like spreadsheet applications but in more complex domains, it most often creates or neglects unsalvable*[sic]* high-risk elements and leads the project down a path to disaster'.

From a risk analysis perspective the evolutionary model can identify project-specific situations that have high risks and is used to reduce these high risks by gaining knowledge (Blum,1992). It is important to exploit the main ideas behind evolutionary development, that is to achieve customer involvement early on. This allows ideas to be tested earlier, so that the cost of failure is less (Olson,1993) thus reducing the risks in the maintenance phase. Baskerville and Stage (1996) claim that risk analysis techniques can support the management of prototype development by providing a framework for determining priorities, resources and activities during the course of an evolutionary prototyping project. The experience gained from each prototype then forms the foundation for the next risk analysis cycle (see Figure 2.1).

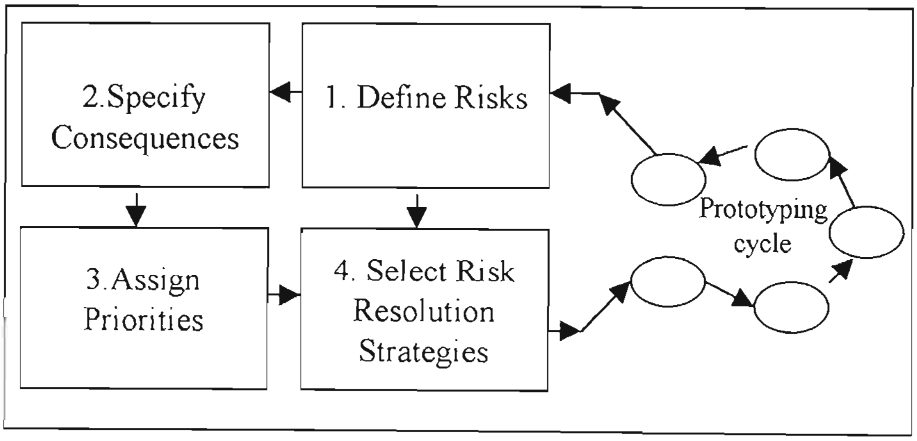

Figure 2.1: The use of risk analysis in evolutionary prototyping (Adapted from Baskerville and Stage (1996))

A combination of "evolutionary prototyping" and risk analysis can be used to reduce risks in software development. The evolutionary model acknowledges that the user's needs are not fully understood, and that not all requirements can be defined up-front. This supports Reifer's (2000) assertion that requirements stay incomplete until the product is delivered. While the evolutionary model defers completing the requirements, the incremental model determines the system requirements first, then performs the rest of the development in a sequence of builds.

2.1.4 The Incremental Model

If prototyping leads to top-down risk reduction then the incremental development may be thought of as being bottom-up (Blum,1992). The incremental model and prototyping reduce risks by attaining knowledge of the system (Powel and Klein,1996), the end result being that users become sensitized to the benefits of computers, and developers become specialists in the field (Blum,1992). The incremental model determines the user needs and defines the system requirements, then performs the rest of the development in a sequence of builds (Cotton,1996) where each build consists of code pieces from various modules that interact together to provide a specific functional capability (McConnell, 2000).

This model involves small deliverables and rapid feedback, and learning about the problem as the design progresses, thereby focusing on the present where knowledge is the greatest. It takes advantage of some of the features unique to software such as conformity and invisibility, thereby making it possible to structure software units to fit arbitrary requirements (Blum,1992).

As the product is designed, implemented, integrated and tested as a series of incremental builds, integration risk is minimized, as components are not brought together for the first time during system integration (McConnell,2000:11). For instance in the waterfall model, the integration happens at the end and therefore inconsistencies tend to show up later. Integration and testing therefore results in schedule delays (Royce,2000) which incurs a reduction in quality as it is too late to make design changes at the integration stage. With the incremental model, requirements and design flaws are detected earlier in the life-cycle, avoiding the "big-bang" integration at the end of the project cycle (Royce, 2000:117).

With this model it might be necessary to modify a previous increment before creating the current one, resulting in software breakages (Hughes and Cotterell,1999). This usually occurs in the earlier stages, where it is easier to fix, since during the later increments the design would have stabilized

(Jalote,1997). Inherent in the incremental model is the danger of regressing to a build-and-fix approach if the increments are too large. Alternatively, increments that are too small can lead to a loss in productivity (Hughes and Cotterel,1999). The incremental model is useful in situations where not all the functionality can be determined or delivered at once, or where the business wants to adjust to the system gradually (Ould,1999).

2.1.5 The Spiral Model

The idea of minimizing risk via the use of prototypes and other means is the concept underlying the spiral model. A somewhat simplistic way of looking at this model is the waterfall model preceded by risk analysis at every phase. The waterfall model emphasizes the end-of-phase certification and the feedback from earlier phases but does not depict the risk-reduction activities, whereas the spiral model focuses on risk reduction by hiding the feedback and certification details (Blum,1992). The spiral model was the first major endeavour to make risk management a formal software engineering activity (Charette,1996:113) with a focus on regular risk reduction (McConnell,2000).

The spiral model uses a basic four-stage cyclic, risk-driven decision stage as a meta-project management mechanism (Charette,1996:113) (see Figure 2.2):

- Determine project objectives, constraints and so on.
- Identify risk, evaluate alternative courses of action and resolve risk in the course chosen.
- Implement the selected course and verify its completion.
- Determine whether or not the risks are at an acceptable level to proceed to the next decision stage.

The spiral model is a constructive attempt to employ risk analysis as a decision-making tool, for instance, weighing the consequences of using a specification document against using a prototype. This type of analysis is conducted regularly since the relevance of performing a particular task or choosing one task over another is assumed to change when the risks of the project change. This flexibility acknowledges the importance of the interpretive, subjective contribution of the designer in estimating the costs and probabilities (Baskerville and Stage,1996:485) in the decision-making process. The spiral model makes no distinction between maintenance and development. Therefore the problem where maintenance is sometimes looked down on by ignorant software professionals does not arise, because maintenance is treated in the same way as development.

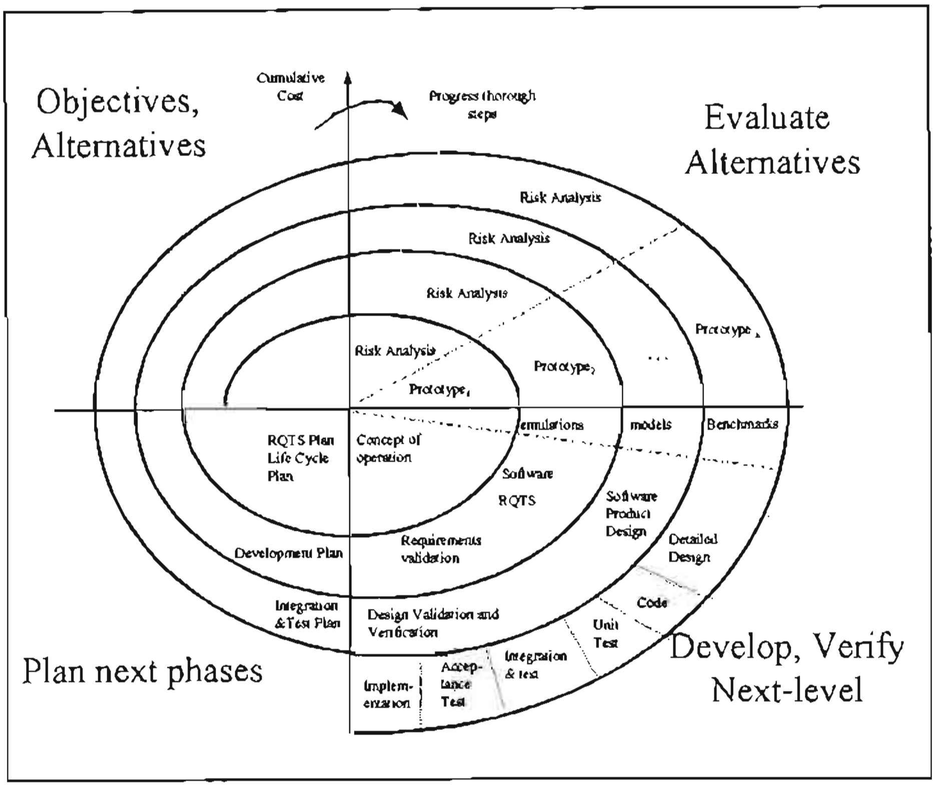

Figure 2.2: Boehm's spiral model of the software development process (Boehm,1989)

There are restrictions to the range of applicability of the spiral model. The spiral model is intended exclusively for the development of large-scale software as it makes no sense to perform risk analysis if the cost of doing risk analysis is comparable to the cost of the project as a whole. Additionally, McConnell (2000:11) asserts that the spiral model is 'so complicated that only experts can use it'. Owing to these issues, Boehm (1991:40) proposes that the incremental process model might be a better alternative than the spiral model since it allows an organization's culture to adjust gradually to risk-oriented management practice and risk-driven process models.

The win-win spiral model is an extension of the spiral model as it acknowledges the role of all the stakeholders roles in reducing risk. It acknowledges the fact that in order to achieve success, every part of the company cooperates in reducing risks to their customers as well as internal risks (Charette,1999). It is based on Theory W, a management theory and approach, which states that making winners of the system's key stakeholders is a necessary and sufficient condition for project success (Boehm et al, 1998:33). The conflicts between all the stakeholders are the root of most software project management difficulties. As seen in Figure 2.3, each of the constituencies has its own desires with regard to the software project. The *users* desire a robust, user-friendly system. The *customers* desire a product delivered reliably, within a short schedule and budget. The *bosses* desire a project with no overruns or surprises. The *maintainers* desire a well-documented, easy to modify system with no bugs. The *development* team desire technical challenges with a preference for design (Boehm and Ross, 1989).

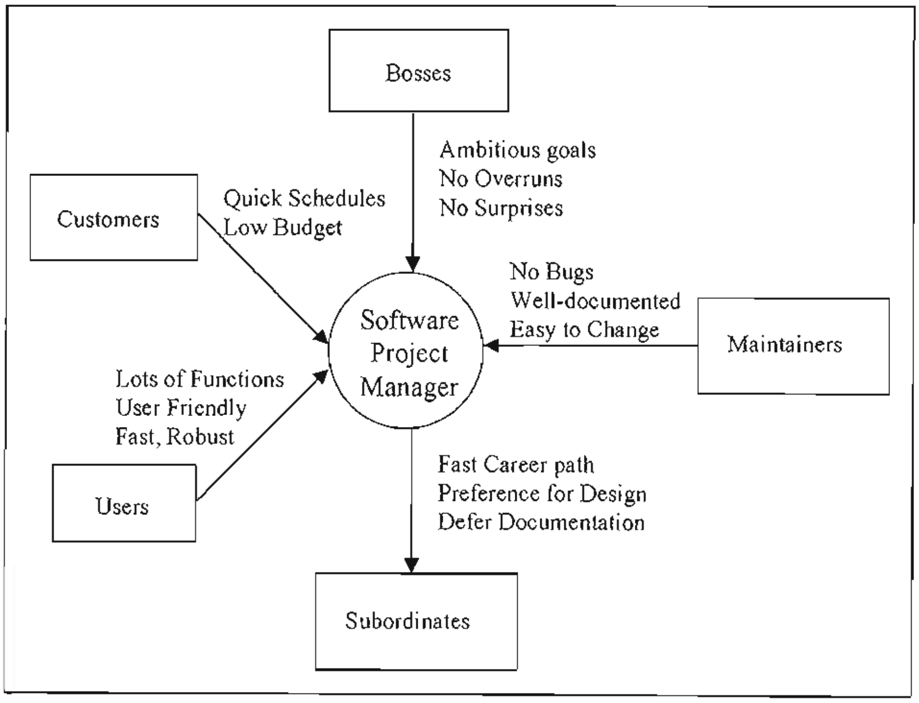

Figure 2.3: Depicting the conflicts arising in the software development due to the desires of each stakeholder (Boehm,1989)

The win-win spiral model extends the spiral model by adding Theory W activities to the front of each cycle (Boehm et al,1998:33). It involves concurrent engineering or joint application development by an integrated team of stakeholders (Boehm, 2000b:124). The win-win spiral model is a good match for development environments where the concept is new to both the users and developers (Boehm et al, 1998:33). The model has three main strengths:

- The model allows teams to adapt to accompanying risks and uncertainties, such as schedule changes and staff changes.

- The model is formal enough to maintain focus.
- It builds trust between the project stakeholders.

According to Boehm et al (1998:34), the differences between the spiral model and the win-win model are the processes of:

- Identifying the system or subsystem key stakeholders.
- Identifying the stakeholders winning conditions for the system.
- Negotiating win-win reconciliations of the stakeholder's winning conditions.

Theory W's fundamental principle is well matched to the problems of software project management. It holds that software project managers will be completely successful if, and only if, they make winners of all the other participants. This principle is exceptionally relevant in people-intensive areas. Risk management focuses the project manager's attention on those portions of the project most likely to cause trouble and compromise the participants winning conditions (Boehm and Ross,1989).

2.2 Comparison of Process Models

The prototype model was developed as a reaction to a specific perceived weakness in the waterfall model, namely that the delivered product may not be what the client really needs and therefore the client may reject the product. The incremental model can also be applied to reduce the client's rejection. This model, notwithstanding its successes, also has some drawbacks as it can regress to the "build and fix approach". The incremental model and the waterfall model are based on the assumption that the requirements are stable, whereas the evolutionary process model acknowledges that the requirements are unstable.

One distinct advantage that the waterfall model has over other models is its document-driven approach. Due to the complexity of software, a process should be visible, so that, it is deliverable-oriented, where each activity must end with the production of some document which makes the process visible. Document-driven approaches like the waterfall model have greater visibility than do code-driven models like the evolutionary model. The waterfall model has good visibility as it produces deliverables, whereas evolutionary development has poor visibility. The spiral model has good visibility as each segment and each ring of the spiral should produce some document (Sommerville,1996). Visibility in software development reduces risk because it is easier to track the process of the development and allows everyone involved in the development to be more aware of the progress and the nature of the system. Visibility can also aid in reducing maintenance risks.

The spiral model incorporates many of the strengths of other models and resolves many of their difficulties (Boehm and Ross, 1989). However, it may not be appropriate in all situations, for instance where the developers are not adequately trained in risk analysis and risk resolution. The spiral model is a meta-model (a model that can generate models (Ould,1999)), as it can accommodate most models as special cases and also provide guidance as to which combination of models best fits a given software situation (Boehm and Ross,1989).

The best way is for each software development organization to decide on a process model that is appropriate for that organization. Its management and its employees should then vary models depending on the features of the specific target software product currently under development. Such a model will incorporate appropriate features from various models thereby maximizing the strengths of the various models whilst minimizing their weaknesses. If a situation arises where none of the process models fits the situation, a risk plan can be used to invent a model that will be appropriate (Ould, 1999).

It is difficult to turn practice into a wholly standard process as every project is different (Lister,1997:22). Prescriptive methodologies tend to decrease productivity due to a morass of paper work, a paucity of methods, the absence of responsibility and a general lack of motivation. It is actually the ability of the developer and the complexity of the project that tend to influence the process (Middleton,1999:174). Using a process model itself can therefore be risky. It is important to be cautious about models that are too complex, too bureaucratic or inflexible (Pressman,1996:18). Therefore software process risk can result from not following a proper software development life cycle or from inherent deficiencies in the software process model itself. Table 2.2 below indicates the risks involved in using each particular model and how each model acknowledges risks. It summarises the preceding argument. The factors that may influence the selection of a particular process model are the probability of the occurrence of risks and the nature of the risks (requirements, schedule or over the entire life-cycle).

Irrespective of the software model followed, there are certain phases that have to be carried out, such as the requirements, implementation and maintenance phases. The next section seeks to highlight the risks that are applicable to these phases and the consequences of these risks for each phase.

Table 2.2: Risk attention given in process models

Models	Risk Attention
Build-and-fix	Owing to its undefined process, it naturally adopts a rather "optimistic approach", assuming that no risks will occur.
Waterfall	Based on the ideology that following a disciplined approach guarantees success, therefore there is no accounting for risk. As a result it does not react well to "unexpected" changes.
Incremental	Direct risks-reducing strategies are not considered but the model itself is a risk-reducing strategy as is stated in Boehm's top ten list (Boehm,1991:35). The incremental model is suggested for reducing requirements risk, schedule and budget risk.
Evolutionary	Direct risk-reducing strategies are not considered but the model can be useful in reducing requirements risk, where the requirements are unclear.
Spiral	Risk attention is fully integrated into the model.

2.3 A Categorisation of Software Development Phases

Here the concern is with the risks related to the software development not proceeding according to plan. At every stage during planning assumptions are made which, if not valid, may put the plan at risk (Huges and Cotterel,1999). It is important to recognize the risk that might occur in the development and maintenance phases. Thereafter plans can be made to avoid these events or minimize their impact if they are unavoidable (Pfleeger, 2000). The development domain spans several phases and therefore it is important to establish what its major components consist of. Blum (1992) proposed that all software process models follow these "transformations":

- from the need in the real world to a problem statement that identifies a software solution to that need. This transformation is represented by the Requirements Specification;
- from the problem statement into a detailed Implementation Statement that can be transferred into an operational system;
- from the Implementation Statement to a system that will satisfy the real-world need. It represents the system in its operational environment.

The first two transformations represent the basic software development life-cycle. If there is any problem with "satisfying the real-world need" and if the supplier is to rectify this problem, then it is termed "Maintenance". In considering the risks in software process development the author, uses Blum's "transformations" of a typical software process model to consider three major areas in the software life-cycle where risk might occur:

- Requirements Specification
- Implementation
- Maintenance

Blum (1992) maintains that the idea in describing what is inherent in all software process implementations is to bring forth a context within which to address the problems that currently limit productivity. Risks occur in every "transformation" and these risks are not characteristic of any phase, except for requirements risk, but they do affect the phases of software development either directly or indirectly. These risks are budget risks, schedule risks, (Chittister and Haimes,1993) and personnel risks(see Table 2.3).

The effect of not reducing risks as soon as possible results in risks compounding. For example, not reducing requirement risks results in design risks. These design risks are carried over into the code and the discovery of these risks during the later phase of testing may be disastrous as it results in schedule delays due to the rework involved. If these risks are not discovered during testing and the product is then handed to the client, it will be sent back for maintenance and this results in maintenance risks (Jalote,1997). In the subsequent section, the four main components of risk and their contribution to the problems experienced in the design, coding, testing and maintenance phases will be addressed. Risk management cannot improve the design, code, testing or maintenance phases as it does not provide any technical solutions or software process improvement strategies. However, it can help one to make informed choices during these phases.

Although there are several sources of risk, factors such as budget, schedule, requirements and personnel are the primary sources of all risk. Technical risks will not be given consideration, as most of the problems are not technical ones but management ones (Brown,1996:95). Therefore the risks in Table 2.3 below are directly related to management issues, rather than technical ones. Other risks such as functionality risks, i.e. not getting the requirements right (Ropponen and Lyytinen,2000), are a result of requirements risks and personnel risks. Resource risks (Steen,1997) such as inadequate staffing are related to personnel, budget and scheduling risks. As these risks are the result of the major risk sources, they will not be dealt with explicitly.

Table 2.3: Depicting types of software risks and their causes[1]

Types of risks	Cause of Risk
Budget risks	Any deviation that results in the project going over budget.
Schedule risks	Any distraction that keeps the project from being on time.
Requirements risks	Not meeting requirements i.e. not building what the client specified.
Personnel Risks	Due to personnel shortfalls such as insufficient expertise.

[1](derived from Chittister and Haimes,1993; Steen,1997; Ropponen and Lyytinen,2000)

2.3.1 Risks in the Requirements Specification Domain

A natural tension exists between the need to get the requirements right and getting the right requirements (Clergy,1994). The implementation of good software requirements management practices, is believed to be one of the first process improvement steps that an organization should take (El Emam and Hoeltje,1997:143). When it comes to determining the requirements there are two basic "schools of thought": one is to perfect the requirements at the beginning and the other is to allow the requirements to grow as the project proceeds (Korac-Borsvet et al,1995). The former approach has a requirements-first emphasis as in the waterfall model while the latter is an approach similar to the evolutionary approach.

According to Boehm et al (2000:120) process models based on requirement-first emphasis make the following incorrect assumptions:

- Participants can determine all requirements in advance of implementation.
- Requirements have no unresolved, high-risk implications.
- Participants fully understand the right architecture for implementing the requirements.
- Requirements match the expectations of all the system's key stakeholders.
- The requirement's nature will change little during development.
- Deadlines allow enough calender time to proceed sequentially.

Requirements are a means, not an end. Requirements, designs and plans should evolve together (Royce,2000:121), therefore using evolutionary approaches such as the spiral model (Keil et al,1998) is more conducive to obtaining requirements from users. According to Nidumolu (1996:79-80), there are three important dimensions of requirements uncertainty:

- Requirements instability, which is the extent to which changes are made in user requirements.
- Requirements diversity, which is the extent to which users differ amongst themselves.
- Requirements analysis, which is the ease with which the process of converting the users' needs to a set of specifications is carried out.

Software projects have volatile requirements that cause the project scope to change frequently. Projects subjected to such volatility are more difficult to control (Keil,2000). The problem with requirements instability comes from users changing the scope and objectives because of a lack of frozen requirements (Keil et al, 1998). The impediments to requirements analysis come from incompleteness, ambiguity, gold plating (excessiveness) and misunderstandings (Schach,1993) in the requirements document. There are therefore two basic problems regarding the requirements: firstly getting the correct requirements from the customer, and secondly making sure that the requirements are implemented correctly. The second problem is sometimes viewed as system functionality risks.

The customer mandate is not the only source of risk to the requirements. Occasionally contractors deliberately hide or underestimate risks even though one would think the possibly disastrous consequences of this behaviour would give them an incentive to detect and disclose risk (Schmidt et al,1999). A requirements document should communicate the essence of what is desired, along with the risks, benefits and importance of each requirement (Bach,1999).

Boehm's risk resolution strategy advocates the use of prototypes to prevent requirements risk which in turn prevents design risk, as it allows designers to identify risks as well (Baskerville and Stage,1996). When prototyping is performed in the requirement analysis, the results are incorporated into the requirements document. The members of the development team use the rapid prototype to construct the specification document and even though the rapid prototype was built in a hurry, the design team can gain insights from it (Blum,1992).

A requirements specification document is one potential source of information regarding future problems (Bach, 1999:113). Ignoring potential areas of risk in the requirements phase can result in errors being found in the testing phase, which can be fourteen times more costly to fix at that juncture (Hammer et al,1996). It is generally accepted that requirements risk is the greatest threat to project success and should be measured throughout the life-cycle. The requirements phase is difficult due to the "conflicts" between the customer and developer (Bennatan,1996). Moreover software customers and users frequently have little feel for what is technologically possible with computers and software and this can result in unrealistic expectations.

2.3.2 Budget Risk and Schedule Risk

The corollary of budget and schedule risks is process risk since formalized procedures are usually abandoned as a result of schedule slippages and budget overruns. When projects are subjected to arbitrary and sometimes irrational schedule and cost constraints, a reduction in quality (Jones,1996:103) ensues. The problem with the budget and schedule of any project is that they have to be estimated (Conrow and Shishido,1997) and wrong estimates expose the contractor to many risks if tasks are underestimated and delivered late (Gilb,1986). Incorrect estimations are the product of being overly optimistic in assessing the limits of performance achievable for any given budget and schedule (Conrow and Shishido,1997:84) and this is exacerbated when developers overlook the tasks (Pfleeger,1998) involved.

Risk analysis could be used to determine the feasibility of the development and the work schedule (Chapman and Ward,1997). Risk management can improve cost estimates as all probable decisions would have been accounted for (Kuver,1999). Keeping the project size as small as possible or decomposing the project into smaller units will improve scheduling and budget estimates (Ropponen and Lyytinen,2000). Therefore, using models such as the incremental and evolutionary process model might be useful in this regard.

2.3.3 Personnel Risks

It is important to consider the effects of personnel risk in a risk management programme because attitudes such as "risk aversion" affect decision-making (Cardenas-Garcia and Zelkowitz,1991; Pfleeger,1998) as people are inclined to report favourable information (Abdel-Hamid,1993:604). For instance, behaviours such as ignoring defects or being tardy in correcting defects can be (Chittister,1993) detrimental to project success. This type of behaviour defies Gilb's (1988) risk-sharing principle, which is based on the axiom that risk knowledge must be shared with clients and colleagues.

The risks that can lead to unsuccessful development from the personnel domain are:

- unrealistic expectations of the personnel's abilities (Ropponen and Lyytinen, 2000);
- poorly defined responsibilities, duties and accountability (Charette,1989);
- ineffective communication between team members (Charette,1989);
- no assistance available for the resolution of conflicts or issues (Charette,1989);
- insufficient staff (Keil et al,1998) due to staff turnover;
- a lack of required knowledge or skills in the project personnel (Keil et al,1998);

Staff turnover can be addressed by using modular software architectures and encapsulation, confining the effects of personnel turnover to small parts of the system (Boehm and DeMarco,1997). Software failures are primarily caused by poor judgement on the part of managers, executives and clients, and not by errors by technical teams (Jones,1996:103). Errors creep into software owing to incorrect assumptions on the part of developers who fail to see a new problem clearly because they tend to inject fragments of past solutions into current solutions (Borcz,1996; Pescio,1997). Since problems in the personnel domain occur due to insufficient knowledge and poor communication, risk management can be a catalyst for opening communication lines regarding problems. A well-defined and disciplined risk management process can increase communication both vertically and horizontally (Conrow,1997) thereby renewing confidence as management is advised of the current status (Kuver,1999).

Taking cognizance of the different work styles can alleviate some of the problems with personnel risks. It can help to determine which people are better suited to a particular activity (Pfleeger,1998). Table 2.4 depicts the different personality traits and the anticipated behaviours in crisis situations. For example, a "Rational Extrovert" is not an appropriate personality type to be involved in identifying risks, as the rational extrovert is only interested in the "bottom line" therefore defying Gilb's (1988) Asking Principle which is based on the axiom that not enquiring about risk information can lead to serious consequences later on.

As there is much uncertainty involved in a project, a more democratic approach should be adopted and good managers should seek out individuals who are flexible enough to interact regardless of their work styles (Pfleeger,1998) or ethics. The success of a project is not only affected by the degree of communication that exists but also by the ability of individuals to communicate their ideas (Pfleeger,1998). As software is a people-intensive endeavour, problems arise out of how people interact and communicate (Phillips,1996). There are aspects of the workers' background that can affect the quality of the project team (Pleeger,1998) since people with good skills and good judgement produce successful projects (Boehm,1991).

Table 2.4: Depicting work styles of different personality types and their reactions to crisis (adapted from Pfleeger,1998)

Personality Type and Behavioural Characteristics	**Reaction in Crisis**
Rational Extrovert: Tells others and decides logically. Examines options and probable effects. Interested in the bottom line and not interested in explanations supporting each option.	Will make decisive decisions in crisis.
Rational Introvert: Asks others and decides logically. Accurate and thorough. Does not make decisions without complete information.	Will ask why a crisis has occurred.
Intuitive Extrovert: Tells others and acknowledges feelings. Prefers professional judgement rather than slow careful analysis. Likes to work where there is interaction among people.	Will offer several solutions to the crisis.
Intuitive Introvert: Asks others and acknowledges feelings .Takes time to make a decision and wants complete information. Analyses not only facts and figures but relational dependencies and emotional involvement.	Will ask what can be done to help in this crisis.

Ineffective communications are usually caused by perceived "class" differences between designers, developers and testers (Hantos and Gisbert,2000). For instance, developers prefer to develop and not gather requirements. Developers have difficulty seeing a problem from the end-user's point of view or a business process view (Abbot, 2001:46). In practice, the software developers are usually not concerned with the correctness of the requirement, as it is assumed that this is the system engineer's job. This "separation of concerns" syndrome has kept the software engineering methods focused on "Abstract Logical Exercises" (Boehm,2000c:115). It is important to realize, as Pressman (1995) contends, that while technologies, businesses and circumstances change, human nature does not. 'The single most important determinant of a project's success is the ability, experience, and motivation of its people' (Brown, 1996:98). Therefore using a win-win spiral will be useful in reducing personnel risks as it tries to understand how people want to win (Boehm and Ross,1989), thereby increasing the motivation of every stakeholder.

2.3.4 The Effect of Risks in the Implementation Domain

Project managers generally perceive the risks in this phase as low as they have control over this area. Risks during design, coding and testing phases occur due to the lack of an effective development process methodology (Keil et al,1998) and the underestimation of project complexity (Conrow and Shishido,1997). This is further exacerbated by the lack of information due to poor communication channels (Charette,1989). It is vital to determine at this stage whether or not the complexities of the system development and implementation can be managed successfully (Keil et al,1998) within the current budget, schedule and personnel resources available.

Requirements Risks

Most of the risks in the design can be traced back to incomplete or ambiguous requirements. Designers find it difficult to work with requirements specification documents because of the lack of "visibility". Software development involves dealing in abstractions and the code is the only output of the project, which makes the design process largely invisible (Pressman,1996). If the code is the only determinant to be used to test if the design works, requirements specification risks become even more apparent in models like the waterfall model, where the design must be done in its entirety before the code. The problem with initial detailed planning is that early, false precision is a 'recurring source of downstream scrap and rework'(Royce,2000:118).

Personnel Risks

Incompetent, ignorant, or uncoordinated personnel create designs that are inconsistent, incoherent, incomplete and inflexible (Charette, 1989). The technological illiteracy of developers can further increase the potential of failure (Boehm, 2000a:95). The lack of collaboration between designers of different preferences and levels of experience (Pfleeger,1998) coupled with following deficient structures (Keil, et al,1998) can increase the risks in the design, coding and testing phases. Additionally the complexity of the design and tight schedules further increases personnel risks (Conrow and Shishido,1997). Using disciplined development processes to break the project into manageable chunks (Keil et al,1998) and allocating tasks to individuals in an appropriate manner can significantly influence the behaviour of the participant and hence the project design (Chapman and Ward,1997). It is therefore important to take into account the ability of an individual as experience, training and preferences may vary (Strassberg,2001:129).

Schedule and Budget Risks

Setting timetables and budgets can have a tremendous effect on the ability of the developer to create user-friendly and fully functional systems (Ropponen et al,2000) because schedule constraints become more and more demanding as software development progresses. For example, if a problem is discovered in the coding phase, it is usually too late to go back and change the design and this will lead to poorer quality and increasing schedule risks (Royce,2000). If inadequate testing takes place during the testing phase then an increase in schedule risks will result.

Incorporating Risk Management in the Implementation Phase

While there are many sources of risk, there is an undeniable relationship between risks and complexity. Therefore, a major part of risk mitigation must be aimed at reducing complexity (Lawson,1998), for example by focussing on getting the design right first, and then introducing risk by trying to optimize the design (Ould,1999). Risk analysis can be useful in determining the risk involved in making a particular design choice or using a new technology (Pfleeger, 2000). Therefore risk analysis can result in the development of contingencies (Keil et al,1998) to deal with the possible risks that may arise from a particular decision.

It is important to include testing as early as possible and software models that demonstrate this attribute, such as the evolutionary model, are more effective in rooting out problems earlier on (Dickey,2000:177). Requirements testing can ensure that the final product meets its requirements. Bach (1999) states that this form of testing can be so much more useful if risk is made a consideration. Perhaps this principle can form a basis for improving other testing techniques. Bach (1999:113) asserts that thinking in terms of risk, beyond just the truisms of requirements-based testing, produces 'a richer set of ideas' (see Table 2.5).

During testing, risk analysis can be useful in determining what the consequences would be of inadequate testing. For example, suppose the schedule risk is growing. As a result, performing regression testing is considered to be a contributing factor to missing the deadline as it is possible that existing functionality works correctly when new functionality is added. Risk analysis can be used to determine what the consequences would be of not performing regression testing (Pfleeger,1998).

Table 2.5: Depicting the principles of requirements based testing and how risk analysis can improve this process (Derived from Bach (1999))

Principles of Requirements- Based Testing	Risks to be Taken Into Consideration
Without stated requirements no testing is possible.	A good tester looks for unintentional gaps in the requirement and works to resolve them, to the degree justified by the risks of the situation. A good tester should go beyond what is stated.
A software product must satisfy all its requirements.	Consider the risk of violating a requirement.
All test cases must be traced back to the requirements and vice-versa.	It is not good enough for a test case to be associated with a particular requirement. In the quest to uncover risks, the better question is: how is this test case associated with this particular requirement?
Requirements must be stated in testable terms.	Simplifying the requirement can be a risk in itself. Instead if left ambiguous, it can lead to meaningful discussions.

Risk analysis cannot directly influence the design, coding or testing but can influence the coordination and control procedures, and identify appropriate participants. Risks occur when key decisions are made without considering the effect they would have on the actual software design. Risk analysis can be used to determine the consequences of each decision. In terms of reducing personnel risks in the implementation, Brooks (1986) avoids offering a technological solution to a problem that is essentially a sociological one. He claims that 'growing great designers' is the answer to software development woes and that this can be achieved through recognition, reward and nurturing.

2.3.5 The Effect of Risks in the Maintenance Domain

Maintenance costs a disproportionate amount of time and money in the software life span (Albanna and Osterhaus,1998) and success is measured by the 'degree of change that the software can absorb'(Charette et al,1997:48). The software development domain and the maintenance domain share the same risks such as the requirements, personnel and schedule and budget risks. However, the risks in the maintenance domain become more critical and more apparent (Charette, et al,1997).

Budget Risks and Schedule Risks

The changes made during maintenance have to be made as soon as possible. There is no schedule to work through as in the development phases (Charette,1997) . Maintenance costs often exceed development costs due to the lack of time, personnel risks and requirement risks.

Personnel Risks

The original developers seldom work on maintaining the product. Therefore, it is difficult for maintainers to work with a system that they were not originally involved in (Charette,1997). The rationale for this is that maintenance is deemed to be less important than the development of a new product. The spiral model process counteracts this notion by not distinguishing between development and maintenance.

Requirements Risks

In the maintenance domain, the demand to make changes becomes even more pronounced because the users have had time to interact with the system (Charette et al,1997). There are three forms of maintenance: corrective, perfective and adaptive. Corrective maintenance is the process of removing faults in the system. The risks can arise owing to the fact that some software cannot be easily corrected and that sometimes correcting one fault can result in regression faults. Adaptive maintenance occurs when the system has to be ported to another environment and risks result when the software cannot be adapted to a new environment (Sherer,1995:370).

The inadequacies of the requirements and implementation phases come back to roost in the maintenance stage. For instance, if the software was poorly designed, the lack of control quickly becomes apparent (Charette,1997). The pressure in the maintenance phase usually emanates from the customer mandate, and this kind of risk is difficult to control and can only be influenced by the project manager (Keil et al,1998). Assuming the user experiences difficulties with the user interface, then the system would have to be changed to deal with this problem. The roots of this problem are the lack of user involvement from the outset of the project (Conrow and Shishido,1997). The following problems lead to system rejection (Keil et al,1998) or changes being requested:

- Failure to gain user commitment.
- A lack of adequate user involvement.
- Failure to manage end user expectations.
- Conflict between user departments.

Waiting until the traditional end (as in the waterfall model of the development cycle) to do the analysis and make decisions to incorporate new enhancements leads to cost growth and schedule slippages (Conrow and Shishido,1997:87). Therefore short iteration cycles encourage the users to use the product early and keep changes to requirements to a minimum (Abbot,2001:46) as in the incremental and evolutionary models.

To prevent system rejection or maintenance requests it is important to maintain good relationships with customers and promote customer commitment to the project. Therefore employing a process that encourages this behaviour would be essential. The process model best suited to this kind of philosophy is the win-win spiral model approach (Keil et al,1998).

2.4 Further Remarks on Risk Issues Related to Software Process Modelling

Building workable systems is not a science, as there are no solutions to real life problems such as unstable requirements (Jackson,1998). When a software development has high complexity and high uncertainty, it is a poor candidate for planning or development according to a "normal science" model of project management (Charette,1996). 'Normal science assumes that large-scale project software are like puzzles to be solved: using reasoned trial and error, based on accepted engineering paradigms, the pieces will fall into place' (Charette,1996:111). The danger inherent in formalised processes is that practitioners become overwhelmed with procedure thereby losing sight of the real aim (Chapman and Ward,1997).

Initially the software community focussed on improving technology; now the emphasis is on improving the process, but 'improving people' is not an issue (Pressman,1995:102). Technological solutions yield benefits that are difficult to sustain (Dutta et al,1998:78) and 'merely improving the efficiency of process' is not adequate (Charette,1999:69). Therefore it is wrong to assume that project management is necessarily the medium for improving all software development projects (Chandrashekar et al,1993:29).

A 'purely mathematical model' can never portray the project manager's ability to make choices among a set of conflicting options. Therefore, the inclusion of risk analysis in the model preserves this important aspect (Cardenas-Garcia and Zelkowitz,1991). Within the realisation of the limitations of process models, lies the certainty that software development is risky and this is where the rationale behind incorporating risk management into the software development process becomes apparent. Risk management focuses on building the right product, project performance, managing change, innovation and uncertainty, while process improvement focuses on building the product right, activity improvement, managing variability, conformance and control (Pleeger,1998).

Once a project is over, performing the postmortem can sustain organizational learning for risk management in the future. Risk management does not have to be a 'whiz-bang' approach but rather an evolution, from evaluating past processes and making a list of issues of concern to finally implementing it (Collier et al,1996). Methods should reflect the 'personality' of the project: simple, complex, team oriented, ad hoc or structured (Hall,1998). Implementing risk management does not have to be a difficult or cumbersome process. Driving forces for including risk management (Hall,1998) are consistent with the ideals of software process improvement:

- focus on goals;
- satisfies customer requirements;
- increases visibility for higher risk areas;
- promotes communication of risks;

- provides for risk-aware decisions;
- helps resolve difficult issues;
- helps avoid surprises;
- helps prevent problems;
- reduces rework.

Much of good project management overlaps with the principles of risk management. For instance project management practice involves the management of process risks such as human errors, omissions and communication failures (Chapman and Ward,1997). The difference is that a software process model describes what is common from project to project, while 'software risk management describes what is different' (Carr,1997:22). However, trying a new project orientation can be risky in itself, as change invites risks (Blum,1992). One thing seems to be certain: that 'successful project managers are good risk managers' (Boehm,1991:33).

2.5 Conclusion on Risk Considerations in Software Process Models

Software process models provide the overall pattern of organizing work in a software development project. Hence this chapter is aimed at investigating these models as they provide a more comprehensive perspective within which risk management is practiced within a given organization and project. It was found that some of them, such as the spiral model, consider risk management more explicitly than others. The provided risk categorisation is helpful for a better understanding of the contents of the following chapters and provides a mechanism for evaluating the risk concerns of software managers in the formulation of the interpretive framework. While the foregoing discussion served the purpose of outlining risk considerations in software process models, the next level of understanding is associated with identifying the activities that typify a risk management strategy. This is investigated in the following chapter.

CHAPTER 3

RISK MANAGEMENT FRAMEWORKS

'Risk management can provide you with some of the skills, an emphasis on getting good people, and a good conceptual framework for sharpening your judgement'(Boehm,1991:41).

3.1 Introduction

The dominance of software in the life cycle of information systems, coupled with increased complexity has escalated the risks encountered by software developers. One promising way to deal with these threats of system failure, which has been utilized since the early Eighties, is software risk management (Lyytinen et al,1998 ;Ropponen et al, 2000). Risk management is a disciplined and systematic process (Chittister,1993) which involves making informed decisions about risks. There are several frameworks for risk management and the classification of activities differs from framework to framework but overall the general structure is characterised by the following phases (Ramamoorthy,1993; Carr,1997; Lister,1997; Ould,1999):

- Risk Identification
- Risk Analysis
- Risk Planning
- Risk Monitoring.

The need for a set of methodologies and processes to facilitate the assessment and management of risk in software development (Chittister,1993) has resulted in risk management frameworks to guide organizations in reducing losses (Bandyopadhyay,1999). A framework is used to "shape the attention" and guide the action of risk managers. Although most approaches consist of similar phases, the methods of conducting these activities differ from framework to framework. For instance, some of the risk management approaches provide similar tactics for different situations or different solutions to the same situations (Lyytinen et al,1998). Approaches range from systematic techniques, to checklists, to quantitative analysis, to qualitative analysis

(Barlas,1996:109). Irrespective of the techniques applied, the primary goal of risk management is to identify and confront risk factors with enough lead time to avoid a crisis (Fairly,1994). Figure 3.1 below is an example of a typical risk management framework.

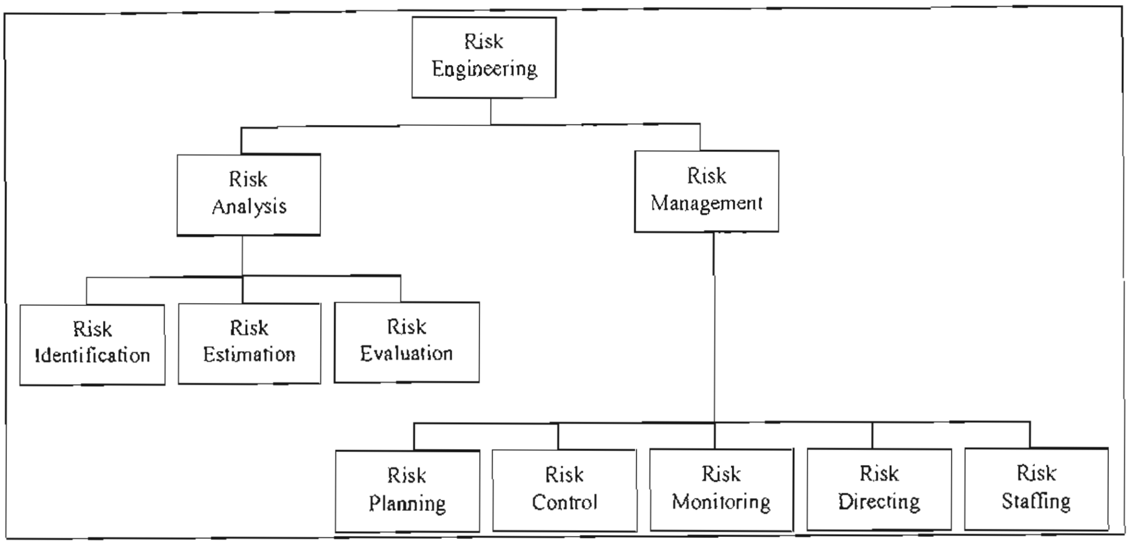

Figure 3.1: Representing a typical risk management framework (Hughes and Cotterell, 1999)

This chapter will provide an overview of the risk management frameworks described in the literature and an in-depth discussion of the four phases involved. Some frameworks consider risk analysis to consist of identification, estimation and evaluation (Lam et al,1997) while others consider risk identification to be a separate phase. But ultimately those steps (listed above) exist irrespective of how the phases are categorised. For purposes of clarification, risk identification and risk analysis will be considered as two separate activities and risk analysis will be treated as the estimation and evaluation process.

The majority of risks tend to emanate from the social, organizational and political realms rather than the technical domain (Powell and Klein,1996) and therefore risks arising out of the personnel domain and requirements should actually be considered as major concerns in risk management (Williams et al,1997). Personnel risks contribute to requirements risk because errors in the requirements definition are attributable to human error (Neumann, 1991b:150). These two risk areas, which are intrinsically intertwined, will be the main focus in this chapter.

3.2 Risk Identification

This phase usually precedes the project life-cycle or alternatively each phase of a project. For example, it might be desirable to determine the risks of the requirements phase before proceeding with the design phase. It is recommended that at the very least, risk identification should occur in the conceptual stages of a project and it should also be a non-static event (Longstaff,2000:45). The techniques applied during risk identification can be broadly categorized as intuitive, inductive, or deductive. The intuitive technique involves unstructured brainstorming. Deductive techniques are based on using hindsight (Frosdick,1997). Inductive techniques revolve around using checklists or adopting a comprehensive categorization approach to cover all possibilities (Powell and Klein,1996). Examples of such checklists include Boehm's top ten risks, Davis's list of requirements risks, Alter and Ginsburg's list of implementation risks and McFarlan's portfolio risks (Lyytinen et al,1998).

Boehm's top ten list, Davis's list of requirements risks, and Alter and Ginsburg's list of implementation risks focus on specific risks, whereas McFarlan's portfolio risks suggests characteristics in projects that contribute to risk. Specific checklists can be problematic as they focus on a list of distinctive risks which can lead to managers focussing solely on analysing these risks and ignoring others, thereby increasing the likelihood of failure (Ropponen and Lyytinen,1997). According to Barki et al (1993) Boehm, in recognizing the problem of making accurate estimates of the probability of a risk or the impact of that risk occurring, developed an approximate method and proposed a prioritized checklist of ten software risk items (see Table 3.1).

Whereas Boehm's top ten list extends over all phases as well as the external environment, Davis' list (Table 3.2) is concerned with selecting procedures that lead to complete and correct requirements (Lyytinen,1998). The rationale for intensifying the focus on requirement risk is based on the critical nature of requirements because requirements drive the entire project (Keil et al,1998). Therefore it is it is vital that the requirements are both complete and accurate, otherwise there is a danger of building a system that no one wants (Keil,1998 ; Deck,2001).

Table 3.1: Boehm's top ten risk items in software development (Lyytinen et al,1998)

Boehm's Top Ten List
Personnel shortfalls
Unrealistic schedules and budgets
Developing the wrong function and properties
Developing the wrong interface
Gold-plating
Continuing stream of requirements changes
Shortfalls in externally furnished components
Shortfalls in externally performed tasks
Real-time performance shortfalls
Straining computer-science capabilities

Table 3.2: Davis' requirements risks in software development (adapted from Lyytinen et al,1998)

Davis' Requirements Risks
Existence and stability of a set of usable requirements
Ability of users to specify requirements
Ability of analysts to elicit and evaluate requirements

Alter and Ginzberg's (1978) list focuses on the implementation factors (Table 3.3), from requirements through to planning. Their rationalizations for considering phases preceding coding, stems from the studies conducted by the same authors, which proved that 'the vast majority of key decisions were made in the early stages of development'.

Boehm's top ten list has two failings. Firstly it is not representative of typical environments as organizations and technological landscapes have evolved, for example, into distributed computing environments (Keil et al,1998). Secondly the checklist represents project-specific risks, such as "Personnel shortfalls" (Powell and Klein,1996).

Table 3.3: Alter and Ginzberg's implementation risks (adapted from Lyytinen et al, 1998)

Alter and Ginzberg's Implementation Risks
Designer lacking experience
Non-existent or unwilling users
Multiple users or designers
Disappearing users, designers or maintainers
Lack or loss of support
Inability to specify the purpose or usage pattern in advance
Unpredictable impact
Technical or cost-effectiveness problems

Discussing the role of uncertainty in determining user information requirements, Davis identifies four sources of project uncertainty: the task to be supported, the application to be developed, the users and the analysts (Barki et al,1993). This inattention to the latter phase risks in the software life cycle and concentration on the requirements risk arises out of the fact that it is more cost effective to identify problems earlier on as the cost of fixing proliferates from one project phase to the next (Brown,1996:98). However, the latter phase cannot be ignored as it is obvious that due to the dynamic nature of software development, initial risk assessment is not adequate as the project evolves (Deck,2001). It is therefore important that software risk management be integrated into every stage of development and not merely be a task at the outset and ignored thereafter (Chatterjee et al,1999).

Alter and Ginzberg's model is based on the problems associated with the organizational acceptance and implementation of the system, and focuses more on the "actors" involved (Lyytinen,1998). As software development is "sensitive to the mistakes of people in development and use" (Neumann,1991b:150) most investments are technologically based and inadequate attention is given to the human and organizational issues which can determine project success (Bronte-Stewart,1998). Unlike Davis's model, it spans most stages of software development and addresses generic risk concerns, unlike Boehm's top ten list (Lyytinen,1998). Alter and Ginzberg (1978) caution against using the eight factors as a comprehensive checklist.

Developing checklists and managing each one as an entity is impractical in complex projects which carry a multitude of risks (Cule,2000). According to Powell and Klein (1996) the problems with checklists are as follows:

- checklists cannot replace experts;
- it is difficult to identify specific risks;
- a checklist is usually not comprehensive enough.

Managers need a means of reducing long checklists to some manageable form without discarding any of the risk items (Cule,2000). McFarlan has focused on three dimensions influencing the risk inherent in a project (Barki et al,1993). The McFarlan model (see Table 3.4) like Davis' model relates only to the very early stages of the software project (Lyytinen et al,1996).

Table 3.4: McFarlans' risk items for risk identification[2]

Risk Item	Effect of risk item	Content of the Risk Item
Project size	The larger the project the greater the complexity.	Cost, time, staffing level and number of parties affected
Experience with technology	Risks increase with lack of experience.	Familiarity of the project team and the software organization with the target technologies
Project structure	Risks increase when projects are poorly structured as reliance on the project manager's judgement increases.	How well structured is the project task

[2](adapted from Boehm and Ross,1989; Lyytinen 1998)

McFarlan's (1974) model categorises its risk items into categories, which gives it an advantage over other models since risks classified into categories based on shared characteristics can be helpful in finding a global solution (Williams,1997:78). However, categorizations must be used with caution. According to Powell and Klein (1996), there are three drawbacks to categorizations:

- categorizations may not cover every single area;
- the gap between general categories of risk and the identifications of risks specific to the project may be hard to bridge;
- they can stifle open debates about risk sources peculiar to any one project.

While the approaches above focus on specific risk items or areas where risks may exist (Powell and Klein,1996), the next type of model focuses on questionnaire-based checklists such as Lyytinen's et al (1998) Risk Management Framework. The framework provides a questionnaire which can be used as a precursor to identifying risk items. The model is based on the socio-technical model, which views organizations as being composed of four components: task, structure, actor and technology.

The actors cover all the stakeholders including users, managers and designers. Structure denotes the project organization. Technology comprises all the development tools and methods. Task signifies all the outcomes in terms of goals and deliverables (Lyytinen,1998). This framework is depicted in Figure 3.2. It considers all four of these components in three environments (Lyytinen et al,1996):

- system environment in which the software is to operate;
- development environment, the environment in which the development takes place;
- management environment, which shapes the software management activities.

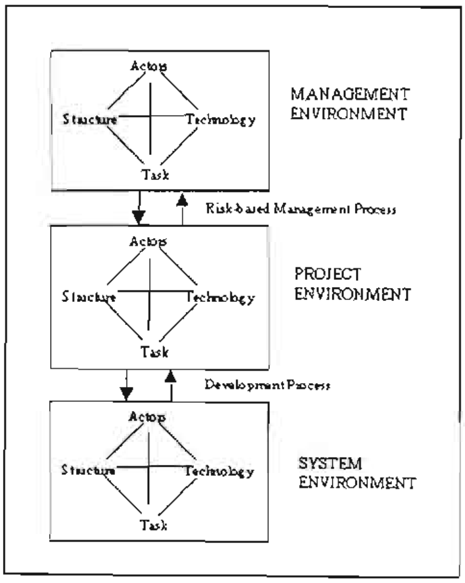

Figure 3.2: Lyytinen et al(1998) risk management framework

This risk management framework considers all four components, task, structure, technology and actor, in all three environments because one component has an impact on the other and this can either reduce or increase risks. The model's mechanism for risk identification is a set of generic risk questions. This is shown in Table 3.5 below.

Table 3.5: Generic questions to support risk-based management (Lyytinen et al,1996)

	System	**Project**	**Management**
Task	What tasks are the software system supporting?	What are the requirements of the system and its environment?	What is an appropriate project environment?
Structure	Which organization is the software system part of?	How is the project organized?	How are the management activities organized?
Technology	Which Technical platform is the software system implemented on?	Which technologies are used to develop the software system?	Which technologies are used to manage the project?
Actor	Who are *[sic]* using the software system or affected by it?	Who are *[sic]* involved in or affected by the project?	Who are *[sic]* involved in managing the project?
Relations	How is the fit and what is the dynamic between components?	How is the fit and what is the dynamics between components?	How is the fit and what is the dynamics between components?

But frameworks such as that of Lyytinen's et al(1996) have been criticized as being one-dimensional and not concentrating on all the dimensions of software development (Chittister,1993). Arising out of this deficiency, is the Holistic Framework for the Assessment and Management of risks which was developed using hierarchical holographic modelling. Fundamentally, hierarchical holographic modelling is based on the premise that large-scale and complex systems, such as software development should be studied and modelled in more than one planar structure or vision (Chittister,1993). Holographic modelling promotes a systemic process that identifies most, if not all, important and critical sources of risk (Longstaff,2000).

Chittister et al (1993) proposed a framework that encompassed several visions. The three decompositions are the Functional, Temporal and Sources of Failure, defined as follows:

- Functional Decomposition represents those aspects that contribute to the specification of the product. These aspects include the requirement, product, process, people, management, environment, system and development attributes.
- Temporal Decomposition refers to the evolution of the software over time, that is requirements, analysis, design coding, testing and operations phases.
- Source Decomposition refers to the sources of system failure, which are hardware, software, human and organizational factors.

From these three decompositions and their attributes, the following perspectives in considering risks arise:

- Consider all the risks that can occur within the Functional Attributes.
- Consider the risks that can arise from the four Source Decompositions within each of the Functional Attributes. For example, determine all the risks that can occur in terms of hardware, software, human beings and organization within the requirement attribute.
- Consider the risks that can arise from the Functional Attributes within each Source Decomposition. For example, in terms of hardware, what risks can occur within each of the seven Functional Attributes.
- Consider the risks associated with each phase of Temporal Decomposition.
- Consider the risks of the Temporal Attributes within each Functional Attribute.
- Consider the risks of the Temporal Attribute for each of the Source Decompositions of failure.

Chittister (1993) admits that there is a certain element of fuzziness involved here and that this merely mirrors the real life fuzziness of developing software, since envisioning large-scale information systems from one limited disciplinary approach is not effective (Longstaff et al,2000). Longstaff et al (2000) has proposed an improvement on the model represented above that considers both endogenous and exogenous events. The model described above focuses solely on endogenous events such as likelihood and consequences of hardware, software, organizational or human failures, and not exogenous events such as regulations, acts of terrorism and other random events.

Other techniques such as assumption analysis, decision driver analysis and decomposition do not increase the decision-maker's understanding of the development domain (Lyytinen et al,1998). Ultimately, irrespective of the methodology used in risk identification, attention must be given to how one perceives a risk. For instance, project managers' identification of risk is based on the following dimensions (Garvey,1997; Keil et al,1998; Ould,1999):

- perceived level of control as managers are very concerned with risks that they have little control over, such as customer risks;
- relative importance of risk;
- past experiences;
- complexity can diminish the ability to identify risks.

The problem with identifying risks is that they come from different sources (Chittister,1993). These sources are the financial, operational, personnel, political and technological (Charette,1999) domains. Creating a complete list is problematic and Powell and Klein (1996) suggest the following approaches to help identify risks:

- use documentation from previous risk analysis in similar areas;
- use people with risk analysis experience in similar areas;
- have diversified skills and viewpoints;
- duplicate analysis by an independent team;
- adopt brainstorming-type approaches to risk identification.

Accounting for every single risk that might possibly occur is also problematic as funds cannot be wasted on a risk that may never occur (Ould,1999). Therefore an all-encompassing risk taxonomy is unrealistic. A different taxonomy for different contexts may be necessary (Moynihan,1997) as every project is new and the construction or introduction of anything new involves risk (Flanagan,1995). Identification of risk is a precursor to placing individuals in an anticipatory mode of thinking about risks (Powell and Klein,1996).

Other checklists exist such as the Software Engineering Taxonomy of risks (Deck,2001) which is used to elicit a range of risks and concerns potentially affecting the software product. It is non-judgmental, semi-structured and as such is not restrictive (Carr et al,1993). A portion of the checklist is shown in Figure 3.3 below.

2.Design

d. [Performance]

Are there stringent response time or throughput requirements?

[22] Are there any problems with performance?

- Throughput
- Scheduling asynchronous real-time events
- Real time response
- Recovery time lines
- Response time
- Database response contention or access?

[23] Has the performance analysis been done?

(Yes) (23.a) What is your level of confidence in the performance analysis?

(Yes) (23.b)Do you have a model to track performance through design and implementation?

Figure 3.3: Sample taxonomy-based questionnaire (Carr et al,1993)

Once risks have been defined they are refined by consensual agreement and reported back to the participants. The identification process places emphasis on the type of people who should be involved, which should be peers (Carr et al,1993). Once identification is complete, the risks are analysed.

3.3 Risk Analysis

The purpose of risk analysis is to analyse the risks identified in such a way that the risks can be ranked in a meaningful manner. The techniques for analysing risk fall into two basic categories: Qualitative or Quantitative techniques. Qualitative techniques use descriptive albeit subjective words to indicate the level of risk and examples include, survey questionnaires and the Delphi Technique. Quantitative techniques, on the other hand, use metrics in order to give a monetary or numerical value to indicate the level of risk. Examples of such techniques include Decision Analysis and Cost Risk Analysis. Further elaboration of the techniques will not be undertaken here but will be provided in the next chapter.

This sets the stage for the definition of further foundation practices, as will the analysis of prior problems (Deck,2001:24). It is harder to be dismissive about risks if they are documented, prioritized and analysed (Williams et al,1997) and if the risk managers consider the messenger to be trustworthy, then they will be more willing to accommodate risk in their decision making (Flanagan,1995).

It is important to note the risks of risk analysis such as making overestimations and underestimations (Charette,1991) and therefore the decision maker's perception must be taken into account. Techniques such as decision analysis give explicit attention to the risk decision maker's attitude toward risk (Covello,1987). A Decision Support System using decision analysis/multi-attribute theory approach would look at the decision maker's options, objectives and uncertainties (Powell and Klein,1996).

The amount of data gathered, the number of people involved and the complexity of the system make it difficult to analyse risk manually (Ramamoorthy,1993). Knowledge-based support tools can be useful in assessing various risks. Support tools need expert knowledge and past experiences need to be documented and encoded (Ramamoorthy,1993). In the absence of knowledge-based tools or quantitative analysis, using "quick and dirty" estimates is just as effective (Williams et al,1997:78). It is important that the estimates made here are fully justified as they have little meaning without details (Gemmer,1997).

3.4 Risk-reducing Measures

Risk resolution techniques are based on using interventions to reduce risky incidents and these resolution techniques usually suggest a schematic plan that will decrease the impact of at least one risk incident or avoid it altogether. Examples include "scenarios" and "user participation" (Lyytinen,1998). At this stage it is important to decide how resources will be utilised (Williams et al,1997) since reducing a risk should not cost more than the risk itself if left untreated. The strategies that can be used for those risks are avoidance, transferral or acceptance (Gemmer,1997).

According to Fairley (1994) risk mitigation involves two types of strategies:

- *Action Planning* occurs when risks are mitigated by an immediate response, for example by training the development team.
- *Contingencyplanning* requires monitoring for some future response, should the need arise, for example late delivery.

Not every risk can be mitigated (Williams, et al,1997) and as discussed above, some risks are not worth mitigating. According to Hall (1998) the following strategies are used in such a case:

- *Risk Avoidance* is a strategy to evade the risk altogether, which is appropriate in a lose-lose situation.
- *Risk Transfer* is a strategy to shift the risk to another person, group or organization.
- *Risk Acceptance* is a strategy to consciously choose to live with the risk consequence.

Boehm, Davis and Alter and Ginzberg attach a set of risk resolution techniques to the risks specified in the checklists, discussed in the risk identification phase (Lyytinen et al,1998) (see Table 3.6, Table 3.7 and Table 3.8).

It is evident from the strategies listed in Tables 3.6 and 3.7 that the burden of ensuring correct requirements is the developer's but it is important to emphasize the user's and the client's involvement. This is because the project manager cannot control user behaviour (Keil et al,1998) or client behaviour but can only influence the clients or users to control requirements risk.

According to Keil et al (1988), the strategies below can help to deal with requirements risk from a customer-centric viewpoint:

- Specify what will not be included in the project.
- Educate the customer about the impact of making changes.

- Draw a line between what is desirable and what is absolutely necessary.
- Promote customer commitment.
- Building trust with customers by meeting commitments.
- Emphasize payoffs.
- Deal with unrealistic user expectations.

Table 3.6: Boehm's list of risk resolution strategies (adapted from Lyytinen et al,1998)

Risk Items	Risk Resolution Strategies
Personal Shortfalls	Staffing with top talent, Job-matching, Team building, Morale building, Cross training, Pre-scheduling
Unrealistic schedules and budgets	Detailed, multi- source cost and schedule estimation, Design to cost, Incremental development, Software re-use, Requirements scrubbing
Developing the wrong function and properties	Organizational analysis, Mission analysis, OPS-concept formulation User surveys, Prototyping, Early user's manuals
Developing the wrong interface	Task Analysis, Prototyping, Scenarios, User characterization
Gold-plating	Requirements Scrubbing, Prototyping, Cost-benefit analysis, Design to cost
Continuing stream of requirements changes	High change threshold, Information hiding, Incremental development
Shortfalls in externally furnished components	Benchmarking, Inspection, Reference checking, Compatibility analysis
Shortfalls in externally performed tasks	Reference Checking, Pre-award audits, Award-fee contracts
Real-time performance shortfalls	Simulation, Benchmarking, Modelling, Prototyping, Instrumentation Tuning, Contracts, Competitive design, Prototyping, Team building
Straining computer-science capabilities	Technical analysis, Cost-benefit analysis, Prototyping, Reference checking

Table 3.7: Davis' resolution strategies to combat risk (adapted from Lyytinen et al,1998)

Requirements Determination Strategy
Asking from users
Deriving from existing systems
Synthesis from characteristics of the utilizing system
Discovering from experiments

Alter and Ginzberg(1978) offer more than a list of resolution strategies. They promote customer involvement as indicated in Table 3.8. They list a set of risk-reduction strategies, which are classified according to compensation (C) and Inhibiting (I). Compensating strategies reduce the impact of the risk while inhibiting strategies avoid the risk. Among Alter and Ginzberg (1978) resolution strategies, "Designers lacking Experience" and "Multiple users and Turnover", have only compensating strategies since these risk items cannot be directly controlled or avoided.

Unlike the prescriptive resolution strategies suggested by Boehm, Davis and Alter and Ginzberg, McFarlan's model for dealing with risks is not prescriptive since it considers the fact that each project is different and requires different managerial approaches (Boehm and Ross,1989). Depending on the size, structure and technology used, different grades of tools are observed (see Table 3.10). In it each tool is an embodiment of resolution strategies (see Table 3.9). These tools are listed below:

- External Integration Tools: Tools to link the project teamwork to the users at both the managerial and the lower levels.
- Internal Integration Tools: Tools to ensure that the team operates as an integrated unit.
- Formal Planning Tools: Tools help to structure the sequence of tasks in advance and estimate the time, money and technical resources the team will need to execute them.
- Formal Control Tools: Tools to help managers evaluate progress and spot potential discrepancies so that corrective active can be taken.

McFarlan (1974) suggests that companies should develop a risk portfolio of all systems built which would not only help managers to assess risks specific to the current project, but would also help them to make more informed decisions in general.

Table 3.8: Alter and Ginzberg's risk resolution strategies (adapted from Lyytinen et al,1998)

Risk	**Risk Resolution Strategies**
Designer lacking experience	Compensating: Use Prototypes, Use evolutionary approach, Use the modular approach, Keep the system simple
Non-existent or unwilling users	Compensating :Hide the complexity, Avoid Change, Obtain management support, Insist on mandatory use, Permit voluntary use , Rely on diffusion and exposure, Obtain User commitment, Obtain user participation, Sell the system
Multiple users or designers	Compensating: Obtain user participation, Obtain user commitment, Obtain management support, Provide training programmes, Permit voluntary use, Rely on diffusion and experience, Tailor system to people's capabilities
Disappearing users, designers or maintainers	Compensating : Obtain management support, Provide training programmes, Provide ongoing assistance
Lack or loss of support	Compensating Permit voluntary use, Rely on diffusion and exposure Inhibiting: Sell the system, Obtain user participation Obtain user commitment, Obtain management support
Inability to specify the purpose or usage pattern in advance	Compensating Use prototypes, Use voluntary approach, Use modular approach Inhibiting: Obtain user participation
Unpredictable impact	Compensating Use prototypes, Use evolutionary approach, Obtain user participation Inhibiting: Obtain management support, Sell the system
Technical or cost-effectiveness	Inhibiting: Use modular approach, Keep the system simple Use Prototypes, Use evolutionary approach

Table 3.9: Risk resolution strategies with specific types of tools(Lyytinen et al,1998)

Tools	Resolution Strategy
External Integration Tools	Selection of a user as project manager Creation of user steering committees Frequency and depth of meeting of this committee User-managed change control process Frequency and detail of distribution of project team minutes to key users Selection of users as team members Formal user specification approval process Progress reports prepared for corporate steering committees Users responsible for education and installation of system Users manage decisions on-key actions dates
Internal Integration Tools	Selection of an experienced DP professional leadership team Selection of a manager lead team Frequent team meeting Regular preparation and distribution of minutes on key design decisions Regular technical status reviews Managed low turnover of team members High percentage of team members with significant previous work relationships Participation of team members in goal setting and deadline Outside technical assistance
Formal Planning Tools	PERT, critical path, etc., networking Milestone phases selection Systems specification standard Feasibility study specification Project approval process Project post audit procedure
Formal Control Tools	Periodic formal status reports versus a plan Change control disciplines Regular milestones presentation meeting Deviation from plan

Table 3.10: Relative contribution of tools to ensure project success (McFarlan,1974)

Project description	External Integration	Internal Integration	Formal Planning	Formal Control
High structure, low technology, large	Low	Medium	High	High
High structure, low technology, small	Low	Low	Medium	High
High structure, high technology, large	Low	High	Medium	Medium
High Structure, High technology, small	Low	High	Low	Low
Low Structure, Low Technology, Large	High	Medium	High	High
Low Structure, Low technology, small	High	Low	Medium	High
Low Structure, High Technology, large	High	High	Low+	Low+
Low structure, high technology, small	High	High	Low	Low

No rationale is provided as to why certain techniques are appropriate for a given risk item. There is a lack of systematic organization where the same techniques are used for the different risk items (Lyytinen,1998). However, reducing one risk may increase others, or at best leave them unaffected (Neumann,1993). Alternatively risks that cannot be mitigated might create a risk that can be mitigated (Williams,1997). Risk mitigation is strongly dependent on the human experts and their past experiences (Ramamoorthy,1997). Generally, contingency plans may be perceived as plans of action that are shelved for possible later use. However, in some cases, the plan is implemented before the anticipated problem occurs (Bennatan,1995).

Ad hoc lists of risk resolution techniques provide a weak understanding of risk management behaviour (Lyytinen et al,1996). The tendency exists for project managers to identify their risks and consider the corresponding mitigation strategies. However, if they do not perceive something as a risk, then the mitigation strategy is ignored even though it might have far-reaching consequences and could not only alleviate a particular risk but might in fact in improve the software process itself, as most mitigation strategies are just good common sense software development practices.

3.5 Risk Monitoring

Risk Monitoring ensures that the risk-reducing methods are implemented effectively and also determines whether or not the risk reducing tactics are in fact reducing risks (Bandyopadhyay,1999). Managers need good metrics in order to make good decisions about risks. Various measures are used to track risks and activate a trigger. A risk activation sheet (see Figure 3.4) can assist in documenting and monitoring risk mitigation plans (Williams et al,1997).

Perception and understanding are not usually considered as issues and it is assumed that all stakeholders have a more or less shared understanding, but there are significant differences between top management and the users (Nottingham,1996:63). It is this "culture gap" which results in wrong or ill-conceived systems (Bronte-Stewart,1998). This problem may be alleviated by the following measures:

- Educating the users on the impact of changes in both project cost and schedule (Keil et al,1998). Specifically in critical software it is important to show the project manager and the customer the consequences of a failure (Borcz,1996).
- Allowing developers to gain practical experience of the user's and client's environment, as it is important that the people involved in software development understand a great deal about the business goals for which they are building an application (Flanagan,1995).
- It is crucial to bring together the different communities, that is, the people who understand the hardware, software, networks and the intended applications, as well as people who understand the human interface (Neumann,1991c:122).

<table>
<tr><td><b>ID</b></td><td>ABC 23</td><td colspan="3"><b>Risk Information Sheet</b>
<b>Identified</b> 2-3-95</td></tr>
<tr><td><b>Priority</b></td><td>6</td><td colspan="3" rowspan="3"><b>Statement</b>
With our lack of experience X Windows software, we may not be able to complete
GUI code on time and it may not be the quality code</td></tr>
<tr><td><b>Probability</b></td><td>High</td></tr>
<tr><td><b>Impact</b></td><td>High</td></tr>
<tr><td><b>Time Frame</b></td><td>Near</td><td><b>Origin</b>: G.Smith</td><td><b>Class</b>: Personnel Experience</td><td><b>Assigned to :</b> S Jones</td></tr>
<tr><td colspan="5"><b>Context</b> : The graphical user interface is an important part of the systems and we do not have anyone trained in the X Windows Systems.</td></tr>
<tr><td colspan="5"><b>Mitigation Strategy</b>
1. Update coding estimates and schedules to reflect the need for increased training and for hiring an expert in
X Windows (changes due 1/5/95)
2. Coordinate with customer and get approval for changing schedule by 1-6-95
3. Identify an available expert from other projects in this division (hired by 15-6-95)
4. Bring in outside training source for current programmers (training complete by 30-7-95)</td></tr>
<tr><td colspan="5"><b>Contingency Plan and Trigger</b>
<i>Plan</i> : Subcontract GUI development to LMN Corp and accept increase in our cost of 25000 dollars
<i>Trigger</i>: If internal expert is not on board and training not completed by 30-7-95</td></tr>
<tr><td colspan="4"><b>Status</b>
GUI code delivered on time, required quality
GUI code has been delivered for testing on schedule
Code 50% complete and 1 week ahead of schedule
Personnel completed 2 weeks training; will monitor progress and quality of work
Brown from project XYZ will be available on 5-6-95 to provide quality assurance, mentoring and critical path programmes
Customer approved revised schedule milestones
Revised estimates and schedule complete; indicates a worst-case and 3-week slip if get an additional expert</td><td><b>Status Date</b>
30-1-96
13-11-95
15-9-95
15-7-95
1-6-95
3-5-95
23-4-95</td></tr>
<tr><td colspan="2"><b>Approval</b>
JQ Jones, ABC Project Manager</td><td><b>Closing Date</b>
15-2-96</td><td colspan="2"><b>Closing Rationale</b>
Code delivered on time. Acceptance test excellent. Risk Gone.</td></tr>
</table>

Figure 3.4: Risk information sheet (Williams et al,1997)

The Software Engineering Institute (SEI) Framework for risk management includes the client in the risk management process itself. In the analysis presented above, none of the frameworks present a model for showing the client's role in reducing risks, but the SEI has developed such a framework which requires customer involvement (Carr,1997) (see Figure 3.5).

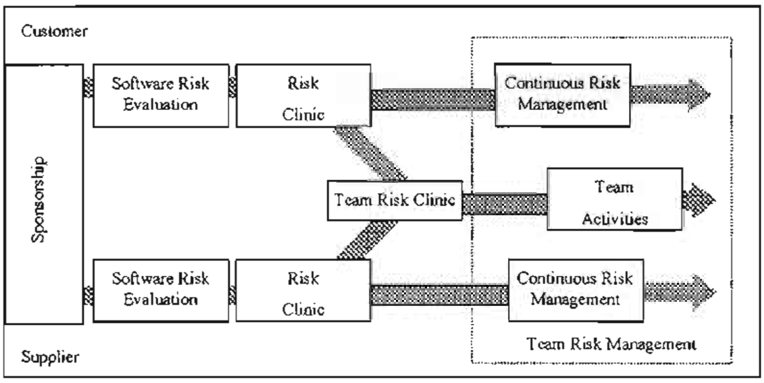

Figure 3.5: The Software Engineering Institute risk management framework (Williams et al,1997)

Software Risk Evaluation (Figure 3.6), which is just a snapshot continuing throughout the life-cycle, involves generating a list of risk statements from the SEI taxonomy, which is then evaluated and prioritized (Williams et al,1997). It is represented as a circle to emphasize a continuous process with communication at the centre because it is a 'conduit through which all information flows'. This is important because a lack of communication is often seen as the obstacle to risk management(Carr et al,1997).

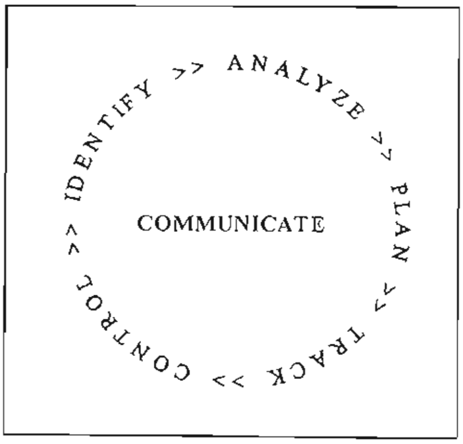

Figure 3.6: Representing SEI Software Risk Evaluation (Carr et al, 1993)

Continuous Risk Management consists of methods and tools that project staff can use to ensure that timely risk identification and analysis are performed and surprises avoided (Carr,1997). The objective of Team Risk Management is that those who are committed to the project, that is, both the customer and supplier, must understand all the relevant perspectives and manage risks accordingly (Williams,et al 1997). The Software Engineering Institute risk management framework encapsulates the 'establishment of a risk baseline through the application of SRE (Software Risk Evaluation)' in both Continuous Risk Management and Team Risk Management (Carr, 1997). This sense of "team" pervades all facets (processes, methods and tools) (Monarch and Gluch,1995).

It is interesting to note that the SEI does not just prescribe techniques to identify and analyse risk, as indicated in Table 3.11, but also modes of communication to facilitate the process and thereby subverting risk aversive attitudes.

Table 3.11: Representing the Software Engineering Institutes techniques in analysis and identification (Monarch and Gluch,1995)

Paradigm	**Methods/Tools**	**Communication Characteristics**
Identify	Group Interview Taxonomy-based questionnaire	non-judgmental non-attribution confidential peer grouping
Analyze	Criteria Filtering Individual Top 5 risks Nominal group technique Comparison risk ranking	individual voice mutual understanding consensus

Theoretically the idea that the client and developer should be involved in risk management is the optimum and should solve the "culture gap" problem. However, Moynihan(1997) concluded that the following factors may affect the process negatively:

- existence, competence, seniority and commitment of client project patron;
- levels of change (to structures, procedures and so on) to be experienced by the client;

- multiplicity and diversity of users to be satisfied;
- level of enthusiasm and support and energy for the project in the client organization;
- logical complexity of the application;
- client's willingness and ability to handle implementation and deployment issues.

As a result of the issues listed above, these types of frameworks are better suited to larger, more formalized and more technical projects for large institutions (Moynihan,1997).

3.6 Further Analysis of Previous Research on Risk Management Frameworks

Ropponen and Lyytinen (1997) concluded that software risk management actually improved system development practices. According to Gemmer (1997), successful risk management consists of three elements:

- process visible to all, which can be repeated and is measurable;
- adequate sources to fuel risk management, sources such as the political, social, financial, environmental and technological realms;
- functional behaviour of the human participants in terms of perceptions, perspectives, communication, consensus, decision making, risk tolerance and management according to a risk management plan.

Gemmer (1997) contends that the last aspect is often neglected but is actually the key. This in effect supports Ropponen and Lyytinen's (1997) conclusion that no specific risk management method is instrumental in defeating risks since the culture in which risk management is practiced is more important than the process itself.

Lyytinen et al (1996) suggest that a risk management framework is a quick way of dealing with risks in comparison to making changes in the management environment such as hiring new people or changing organizational competencies. Risk management frameworks provide structure with which to make better decisions about uncertain future events (Chittister,1993). If risks can be measured then contingency strategies can be provided. However, if risks are unknown or ignored, then surprises occur when least convenient (Chittister,1993).

The more diversified the team is, the more important it is to have systematic, common and agreed-upon risk assessment and management processes (Chittister,1993). A vast majority of software defects are the result of misunderstandings (Neumann,1991a:130). However, a well defined and disciplined risk management process can increase the level of communication both 'vertically and horizontally'(Conrow et al,1997:89) thereby reducing the probability of incorrect assumptions which inevitably lead to errors (Borcz,1996). There is nothing that can be done to reduce risks completely but ignoring them could be disastrous (Lister,1997:20) and as Lister(1997:22) asserts: 'any form of risk management is better than none'. Overall risk management improves predictability (Flanagan,1995) for future projects and is this in a sense means becoming as informed as possible about the software process(Charette,1989).

In spite of the benefits of risk management, there are some drawbacks. Having a systematic framework lulls one into complacency since initial risk assessment will not account for all the actual risks that will appear due to environmental changes (Deck,2001). Alternatively, the cost of accounting for every possible risk is prohibitive (Kitchenham et al,1997) and practising risk management does not guarantee fewer problems (Chittister,1993).

The use of lists directs the attention of the project manager to a narrow set of risks and presents prescriptions to specific sets of risk (Lyytinen,1998) and this is problematic as it is difficult to turn practice into a wholly standard process because every project is different(Lister,1997:22). No single approach can address all pitfalls (Deck,2001) and Lyytinen et al (1998) therefore suggest different approaches in parallel for validation. It is difficult to show the benefits of risk management because it is difficult to measure problem avoidance (Barlas et al,1996:109). This is tantamount to supposing that if a plane is never hijacked then the risk resolution strategy followed to avoid hijacking has worked or else terrorists are no longer interested in hijacking planes anymore. However, the consequences would be disastrous if the plane was hijacked and there were no mitigating measures that could be taken (Charette,1989).

The success of risk management has more to do with the culture in which it is implemented than the tools and techniques used. Therefore it is important to encourage risk-positive attitudes (Brown,1996). The choice of people who guide the risk management process is vital. According to Charette (1999) it is the individual with the best knowledge and experience rather than seniority who should make the critical business decisions. If the executive management is averse to risk then that risk ethic permeates the entire organization (Carr,1997). Risk management is not just the project manager's job or a technical issue (Chittister,1993). Ironically the techniques that seem to work best in high-tech environments are in fact low-tech solutions (Brown,1996:103). Technological solutions are often sought for problems whose solutions require reasonable human behaviour more than sound technology (Neumann,1993:146).

Chittister (1993) asserts that organizational and human failures are endemic to software development failures and that special attention and concern must be paid to them in the risk assessment process since having risk-aversive attitudes inhibits risk management (Carr,1997). Risk aversion or dysfunctional behaviour (Gemmer,1997) manifests itself in the following behavioural characteristics or cultural rules:

- Presenting only positives (Gemmer,1997) by ignoring or concealing mistakes (Chittister,1993).
- Missing signals or valuable data due to inadequate testing (Chittister,1993).
- People not learning from past mistakes (Neumann,1993:146) and even worse, refusing to admit they need to learn or practice the most basic skills of risk or even project management (Charette,1999:72).
- Being overly confident of the skills available or of the formal methods practiced (Neumann,1993:146; Charette,1999).
- Poor communication as information is hoarded for political power (Charette,1993; Gemmer,1997) or opportunism (Lyytinen et al,1998). This unethical professional conduct is debilitating to software risk management (Neumann,1997; Gemmer,1997; Charette,1999). Communication is critical to software development, as a weakness in human communication is analogous to weakness in computer communication (Neumann,1991c:122).
- Concern for the customer is a low priority (Charette,1999).
- Spending resources on process improvement is considered a poor return on investment (Charette,1999).
- Fear of litigation results in risk management being ignored as the existence of a risk plan acknowledges the possibility of failure and can compromise a legal position (Boehm etal,1997).
- Deficient infrastructure to support risk management (Carr,1997).

- Lacking commitment issues (Hall,1998 ; Keil et al,1998; Lyytinen,et al,1998)
- Differences among stakeholders (Lyytinen et al,1998).
- Relying on individual talent and experience.
- Fear of retribution because efforts are made to place blame thereby discouraging people from reporting bad news (Gemmer,1997; Hall,1998).
- People are resistant to change because it implies extra work and time (Hall,1998).

Most investments are technologically based and inadequate attention is given to the human and organizational issues which can determine project success (Bronte-Stewart,1998). It is evident that there is no single cause of problems and therefore no simple solution exists. Risks should be viewed as organizational, social and political rather than technical issues (Powell and Klein,1996). Brown (1996) suggests that a cultural change in software management is imperative because projects fail due to counter-productive management. The measures that can be implemented to encourage risk- seeking attitudes can be summarised as follows:

- Decriminalize risk (Brown,1996) because the culture has evolved in a manner that owning up to risk is confused with defeatism (Boehm et al,1997:18).
- Encourage free flow of knowledge (Charette,1999). Constructive consideration of risks occurs when people talk openly and honestly in a non-judgmental way about potential pitfalls (Ould,1999). Ideally there should be a no-holds-barred atmosphere (Lister,1997:22) and consensus must be achieved (Deck,2001).
- Encourage self-correcting learning (Charette,1999) by not focussing on human-error as this will deteriorate into a blaming exercise (Williams et al,1997).
- Status is irrelevant; experience counts (Charette,1999).
- All stakeholders must be involved in reducing risks (Charette,1999) and be risk-aware at all levels (Carr,1997).
- Provide incentives by empowering performers and providing bonuses (Boehm et al,1997).
- Appoint risk champions in managerial roles (Carr,1997) so that their risk ethic permeates to the lower levels. A risk management champion is someone who is an agent for change (Hall,1998).
- Seek diversity in perspectives on information sources from political, cultural, economic, environmental and technical realms (Gemmer,1997).

- Recognise and minimise biases (Gemmer,1997)

Most risk management approaches deal solely with negative outcomes and how to avoid risks (Lyytinen,1998). However, risk should be seen as either a downside loss or an upside profit (Hall,1998). A risk can be an opportunity, as situations with high potential risk have a high payback. Risk management does not mean removing risks at all costs (Fairley,1994). Therefore, it is important to tackle projects with high risk in order to obtain a competitive edge.

3.7 Conclusion to the Investigation on Risk Management Frameworks

Risk management frameworks must be flexible enough to be easily assimilated into current practices. They should not require following rigid processes. Risk management should actually be about moving beyond the processes of identification, analysis, mitigation and tracking. It should be about fostering a risk-aware culture that facilitates all these processes. All the structures available cannot control human behaviour and it is appropriate human behaviour that is the most important factor. There must, therefore be policies and standards to encourage risk-seeking behaviour and to prevent risk-averse attitudes.

The analysis of the results of this chapter provide the sources for the theoretical foundation for the framework developed in this project and reported later in chapter Five. The quantitative part of the survey draws on the ideas of risk identification, in particular Boehm's top ten list of risk identification. After reviewing risk identification techniques, the author concurs with Ropponen and Lyytinen's (1997:42) motivation for using Boehm's top ten list in their study, that is, it "reflects faithfully a project managers' perspective on software risks by addressing critical concerns and objectives of different stakeholders". This idea matches the objectives of this study as articulated in chapter One. Another contribution to the framework formulation is investigating the commonalities in risk management frameworks. This provided a means to formulate and evaluate questions for the qualitative part of the framework to be presented in chapter Five. The overall analysis of risk management frameworks was useful in identifying their strengths and weaknesses and in deriving the conclusion for the potential usefulness of the above examples of past research results for the purposes of this project.

The next chapter provides a review of the risk analysis techniques because this stage is of particular significance for the risk management frameworks discussed in this part of the dissertation.

CHAPTER 4

TECHNIQUES USED IN RISK ANALYSIS OF SOFTWARE DEVELOPMENT

'The amount of uncertainty is itself a major source of risk, which needs to be reduced as early as possible'

(Boehm,1991:38).

4.1 Introduction

Once risks are identified, there must be some mechanism to assess their importance (Hughes and Cotterell,1999). Risk analysis is used to this end as it allows the ranking of risks to be done in a meaningful way so that managerial attention can be focussed on the areas that constitute the greatest risks (Keil et al,1998). The relative importance of these risks needs to be established, along with some understanding as to why certain risks are perceived as being more important than others. The objective of risk analysis is to convert risk data into decision-making data (Schmidt et al,1999). For the purposes of clarification, the risk analysis process discussed in this chapter consists only of the estimation and evaluation of risks.

According to Hall (1998) the steps in risk analysis are as follows:

- group similar and related risks together to make contingency plans easier;
- determine those variables that can cause the risks to fluctuate;
- determine the sources of risk;
- use risk analysis techniques and tools;
- estimate the risk exposure;
- evaluate the risks against specific criteria;
- rank risks relative to each other.

Steps one, two, three, six and seven do not require any special methodologies and are not difficult to implement, and will therefore not be elaborated upon. The steps that form the crux of the risk analysis phase, are steps four and five.

4.2. Techniques Used in Risk Analysis

As stated earlier the techniques used in risk analysis fall into two basic categories, that is, *Qualitative* or *Quantitative* techniques. *Qualitative* techniques employ descriptive albeit subjective words to indicate the level of risk. Examples include survey questionnaires and the Delphi Technique. *Quantitative* techniques such as cost risk analysis or network analysis, on the other hand use metrics in order to give a monetary value or a numerical value to indicate the level of risk.

4.2.1 Quantitative Techniques for Risks Analysis

Measurement in the field of software engineering has been largely ignored, whereas other disciplines such as electrical engineering would not have evolved were it not for measurements. Hundreds of software metrics have been developed over the last few years, but generally the metrics being employed are the ones that are easy to understand and simple to use (Pighin and Zamolo, 1997). The need for software measurements arises out of the current software crisis. Indeed, it has been suggested that software production is out of control because the lack of measurement controls and metrics means that objectivity cannot be achieved (Fenton,1991).

Measurement has two purposes: Firstly, for assessment, i.e. to keep track of a software project. These metrics are known as *assessment metrics*. The second purpose is for prediction i.e. to determine future characteristics of the software project. These are called *predictive metrics* (Pighin, Zamolo,1997). As risk analysis is predominantly about the future, predictive metrics are used to determine what the probability is of a risk occurring, and what loss would be incurred if such a risk were to occur. Assessment metrics can be also used in the risk monitoring phase, to ensure that the risk action is functioning effectively.

Quantitative techniques in risk analysis translate risk into numbers and are mathematically or computationally based. These techniques provide numerical probabilities or frequencies, or the consequences and likelihood of identified risks. The values used in these techniques are either obtained from historical databases or estimates. Therefore they still contain some degree of uncertainty due to the use of subjectively attained inputs (Baker et al,1998).

Quantitative Risk Assessment is becoming more popular, owing to is its inherent appeal to scientists and it is often mandated by regulatory agencies (Pfleeger,2000) The most frequently applied techniques in software risk analysis are decision tree analysis, network analysis and cost-risk analysis (Bröckers,1995) and these techniques are discussed here.

4.2.1.1 Decision Analysis of Risk

Decision analysis draws theories and methods from several disciplines including statistical decision theory, psychology, systems engineering, systems science, operations research, management science and economics (Covello,1987). Decision analysis is used to structure decisions and to represent real world problems by means of models that can be analyzed to gain insight and understanding. To understand possible loss, possible barriers to achieving goals should be identified, their likelihoods evaluated, and their dependencies in the project or environment ascertained. Dependencies exist where the success of one event relies wholly or in part on the successful completion of another event. Dependencies are particularly important because they have multiplicative risk. Models such as decision trees help one to understand the effect of dependencies on risk evaluation (Flanagan,1995). The basic tenet of decision analysis is that more effective decisions can be made if the decision alternatives and risk preferences are formally expressed and quantified (Covello,1987).

The decision tree is a fundamental risk analysis paradigm (Boehm,1991). As decision analysis is the examination of decisions by breaking them down into the sequences of supporting decisions and the resulting uncertain occurrences, so the decision tree is a representation of such a decision analysis process. The decision tree combines two or more planning alternatives, their respective alternate outcomes and their probabilities into a single tree structure (Bröckers,1995). The risk exposure metric can be used in conjunction with the decision tree to form a composite risk exposure. For instance, for each possible decision there are several possible outcomes. A combined risk exposure can be determined for each possible decision. In Figure 4.1 below the risk exposure A is determined by multiplying the probability (P(UO)) and the associated loss (L(UO)) for outcome one. Then the combined risk exposure is determined by summing up the risk exposure for each choice.

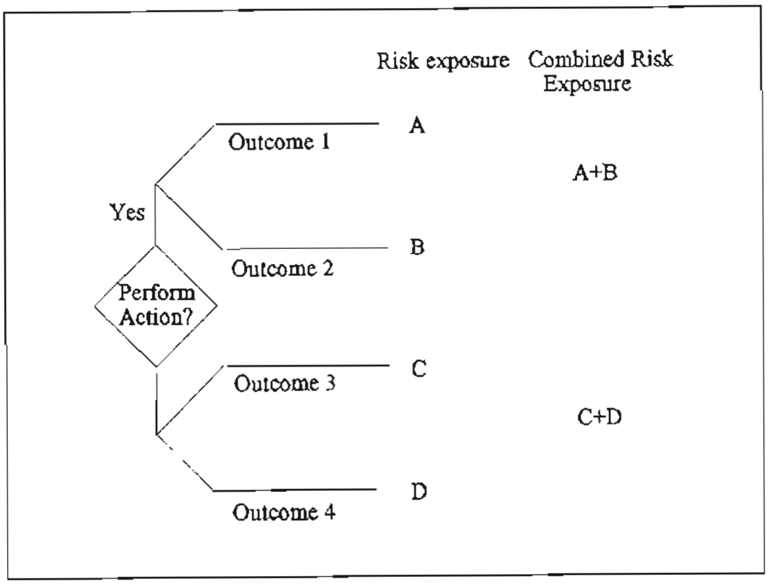

Figure 4.1: Depicting a decision tree (adapted from Pfleeger,1998)

After the impact and probability of each decision alternative have been accessed, the next step is to explicitly depict the value judgements, preferences and tradeoffs (Covello,1987). This step is important as the risk perception and risk propensity of individuals affect their behaviour. Risk perception is the way in which decision makers make assessments of a situation while risk propensity is the tendency of a decision maker to take risky actions. Risk propensity has an impact upon risk perception. For example, a decision maker with a low risk propensity will have a more pessimistic view of a situation (Keil,2000).

Decision analysis of risk acknowledges risk propensities such as risk-seeking, risk- neutral or risk-aversive attitudes (Corvello,1986). Risk-averse people have a conservative risk attitude with a preference for secure payoffs, while risk-seeking people have a liberal risk attitude with a preference for speculative payoffs and risk neutral people have an impartial risk attitude with a preference for future payoffs (Hall,1998). Decision analysis employs utility analysis to determine the risk attitudes. The objective of utility analysis is to construct a utility function that represents a scaling of subjective values assigned by an expert to the consequences of a decision alternative (Corvell,1986). This exponential function uses a parameter known as 'risk tolerance' which determines how risk averse the utility function is. Individuals have different tolerances for risk, which affects the way they make decisions (Hall,1998). The graphs in Figure 4.2 depict the utility functions over monetary outcomes. They show the value(utility) that a risk-averse, risk- neutral and risk-seeking decision maker places on the possible consequences of a decision (Covello,1987).

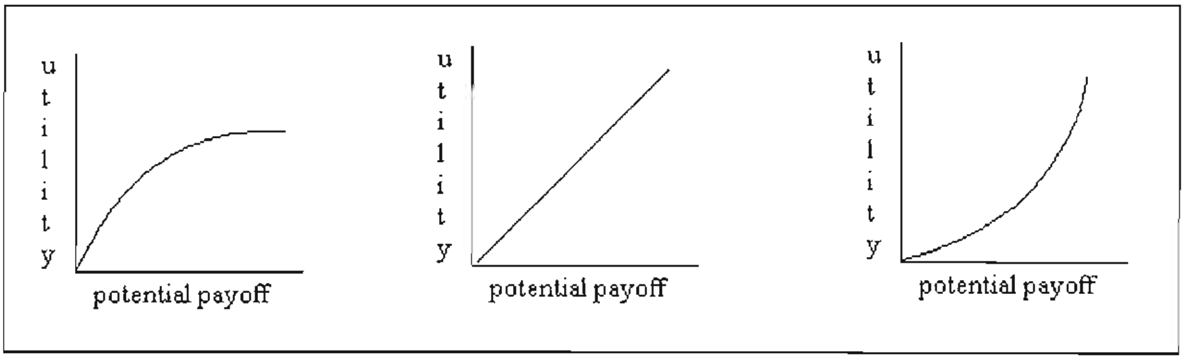

Figure 4.2: Utility function graphs representing: risk-averse, risk-neutral and risk-seeking preferences (Cotterell and Hughes,1999)

As risk is a perceived value, estimates of probability and impact can be biased. It is therefore important to monitor how people perceive risk and act on that perception (Gemmer,1997). In generating points on the utility curve, the decision maker may be asked to choose between two points until the decision maker is indifferent. This information indicates the value or worth (utility) that the decision maker places on possible consequences of decisions (Covello,1987).

In addition to providing unique solutions, the decision tree method provides a framework for analysing the sensitivity of preferred solutions to risk-exposure parameters (Adler et al,1999). The utility functions are also used in sensitivity analysis. The objective of sensitivity analysis is to identify those variables for which better information would be important and those variables whose values are unimportant when doing analysis (Covello,1987). This helps determine the sensitivity of the model to variations in the input by setting each variable to its extreme points. To perform sensitivity analysis is to understand risk, where varying risk tolerance is used to determine at what point the decision changes (Hall,1998).

The difficulty with risk exposure and with decision analysis quantities, is the problem of making accurate estimates of (Boehm,1991) the probability and utility values. Decision analysis operates on the assumption that all decision alternatives and consequences can be enumerated in a meaningful way. The results obtained may be a false illusion of accuracy that can be misleading (Covello,1987).

The next two techniques are based on combining schedule analysis and cost estimation analysis with risk analysis.

4.2.1.2 Cost Risk Analysis

Approaches to identifying risks are usually separate from cost estimation, and therefore a technique that identifies risks in conjunction with cost estimation is an improvement schedule-wise (Madachy,1995). Cost models such as the Constructive Cost Model (COCOMO) and Function Points analysis are often used for project planning and estimation, both to predict person effort and elapsed time. The knowledge used in cost-estimating activities through the use of cost factors can be used to detect the patterns of project risk (Madachy,1997). The estimated effort can be used to assess the impact of risk factors because the effort is the primary cost factor for most software projects (Fairely,1997). The Constructive Cost Model formulation for effort is expressed in person-months is (Madachy,1997):

$$Effort = a \times (Size)^{b} \prod_{i=1}^{number\ of\ cost\ drivers} EM_i \qquad (4.1)$$

where

Size represents the number lines of code

(a,b) are model coefficients

EM_i is the effort multiplier for the *i*th cost driver

Cost estimating is fundamental to project management and therefore, when analysing risk it is important to consider the risk drivers such as the cost drivers found in software estimation models (Hall,1998). The cost drivers account for the attributes of the component that affect the difficulty of its development (Blum,1992). A review of these cost factors reveals sources of risks. Madachy (1997) relates the cost drivers in the Constructive Cost Model to risk by recategorizing the cost factors as attributes of risk, as shown in Table 4.1.

Table 4.1: Depicting the correlation between the cost effort factors of Boehm's Constructive Cost Model with sources of risk (Adapted from Madachy,1997)

Sources of Risk	Cost Factors
Product Risk	Required Software Reliability, Database size, Size, Product complexity, Documentation
Personnel Risk	Analyst capability, Application Experience, Language and tool set experience, Programmer capability, Virtual Machine Experience, Personnel Continuity
Reuse Risk	Required reusability
Process Risk	Use of Software Tools, Multi-site development, Development flexibility, Architecture /risk resolution, Team cohesion, Process maturity
Schedule Risk	Required development schedule
Platform Risk	Execution time constraint, Database size

The cost multiplier data is used to compute the overall risks for each category and for the entire project according to the equation below (Madachy,1997):

$$\textit{Project Risk} = \sum_{j=1}^{\textit{no. of categories}} \; \sum_{i=1}^{\textit{no. of categories of risks}} \textit{risk level}_{ij} \times \textit{effort multiplier product}_{ij} \tag{4.2}$$

where

risk level of one is considered as moderate, two is high and four is very high.

effort multiplier effort = (driver #1 *effort multiplier*) × (driver #2 *effort multiplier*).....(driver #n *effort multiplier*)

Here the risk level corresponds to the probability of the risk occurring and the effort multiplier product represents the cost consequence. As these calculations can be complex, the process is automated by the Expert COCOMO II tool where a knowledge rule base is used to calculate the risk levels depending on the interactions of the cost attributes entered by the user (Madachy,1997).

Alternatively the Constructive Cost Model can be used to calculate effort and schedule and alternate outcomes associated with risk by varying the parameters (Bröckers,1995). For example Känsälä (1997) suggests that the Effort can be varied to account for the risk impact. For example, assume the Effort was calculated to be 50 man-months. Now assume that there is a 50% probability of an analyst with analyst capability of 0.86 being transferred. Suppose two other people were hired each with analyst capability of 1.00 and 1.19 respectively. The Effort will have to be adjusted to take into account the new analyst capability cost factor. Therefore, the project's revised, realistic estimate becomes 57.8 person-months, as:

$$\begin{aligned} RE &= 0.5\times 2pm + 0.5\times(0.5\times 50pm)\times(1 - 0.86)/0.86 + 0.5\times(0.50\times 50pm)\times(1.19 - 0.86)/0.86 \\ &= 1pm + 2pm + 4.8pm \\ &= 7.8pm \end{aligned}$$

The advantage here is that cost estimation and risk analysis are usually considered to be two separate activities, but each process can be used to endorse the other. Cost estimation strategies must consider risk assessment to yield realistic results (Biffl,2000) thereby forming better cost estimates. Considering the cost and risk estimates concurrently is less time consuming and moreover these models are automated.

Känsälä (1997) uses cost drivers to compute indirect risk exposure for risks that are among the cost drivers of the cost model, while Madachy (1997) suggests that categories of risk are generally aligned to the cost attribute categories in the COCOMO. The validity of a cost-risk assessment is highly dependent on the accuracy of parameters used in the COCOMO formulation, while the accuracy of these parameters is dependent on how well it matches the realities of a particular project or organization (Blum,1992). For example Känsälä's (1997) model utilizes the effort metric which is calculated using the size and the exponential term b, which is problematic because of the difficulties in estimating the size and the fact that misclassification of the exponent b will have a major impact on predicted effort. This in turn will result in the risk exposure being inaccurate (Shepard,1995).

4.2.1.3 Network Analysis

This type of analysis is aimed at analysing risks that are caused by deviations from the ideal (Bröckers,1995). The simplest project activity planning tools which explicitly consider risk are the Programme Evaluation and Review Technique (PERT) models, which portray the project activity structure (Chapman and Ward,1997). The objective of PERT is to manage schedule risk, by establishing the shortest development part schedule (Boehm and Ross,1989). The PERT chart can be used to determine the likely effects of schedule risks on the duration of planned activities and this can be used to determine the impact they will have on the activity plan (Hughes and Cotterel,1999).

As the PERT method is well documented, the focus of this discussion will be on how PERT is used to determine the risk of not meeting schedules. The probability of not meeting a target date is calculated as follows:(Hughes and Cotterell,1999)

1. First the expected duration is calculated $t_{e,}$ using the formula:

$$t_e = \frac{a + 4m + b}{6} \tag{4.3}$$

where

a is the shortest time expected to complete the activity, that is the optimistic time

m is the most likely time expected to complete the task

b is the worst possible time allowing for all reasonable eventualities, that is the pessimistic time.

2. Then standard deviation s of an activity time is calculated by the following formula:

$$s = \frac{b-a}{6} \tag{4.4}$$

3. z value is calculated:

$$z = \frac{T - te}{s} \tag{4.5}$$

where T is the target date.

4. The z value is converted into a probability by using the graph of standard normal deviates (see Figure 4.3).

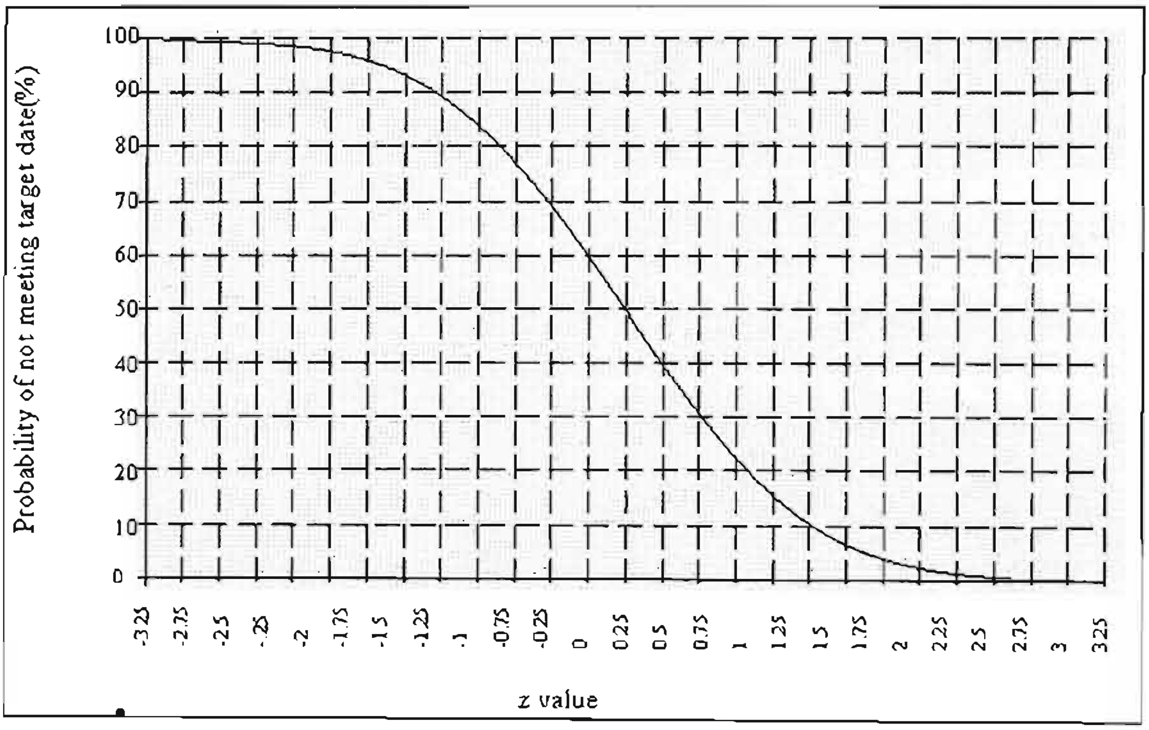

Figure 4.3: Depicting the graph of standard normal deviates (Hughes and Cotterell,1999)

For example, suppose the z value was calculated to be 1.23, then this equates to a probability of 11%, which implies that there is an 11% risk of not meeting a particular target.

The PERT technique is the best technique for analysing schedule risks because the mere process of creating the PERT chart is useful in identifying high-risk situations (Boehm and Ross,1989). The classic PERT technique has been criticised as being inadequate for large projects and weak in modelling and analysing the concurrent, iterative and evolutionary characteristics of software projects (Chang and Christensen,1999:81). The advantage of this approach is that it places an emphasis on the uncertainty in the real world (Hughes and Cotterell,1999). This method enforces viewing schedule estimates and risks concurrently.

4.2.2 Qualitative Techniques

The advantage of qualitative techniques is that, they do not require complex data measurements (Biffl,2000). Qualitative techniques translate risk into descriptive variables and distinguish the possibility of a risk occurring in a linguistic manner. Risk is described as "low" if that risk is unlikely to occur. It is an analysis in relative terms of the outcome and probability of a risk; for example, a high risk compared to low risk. This technique is highly dependent on the experience of the analyst and is therefore highly subjective and prone to inconsistencies (Baker et al,1998). It is nevertheless valuable as an analytical process in the planning and control of a project. Examples include scenario analysis, questionnaires, and the Delphi technique (McGaughey,1994).

4.2.2.1 Delphi Technique

Individuals rarely have both the breadth and depth to act solely on the basis of their own knowledge (Gemmer,1997). The Delphi technique is a collaborative technique for building consensus involving independent analysis and voting by experts given perfect feedback as to how their judgement matches that of the remainder of the group as a whole (McNamee,1999). The Delphi technique uses a series of questionnaires and summarised feedback reports from preceding responses.

This approach is useful for generating and clarifying ideas, reaching consensus, prioritizing, and making decisions on alternative actions. Since face-to-face interaction is not a requirement, the Delphi technique could be used with groups that would not ordinarily meet together. The Delphi technique relies upon expert judgement but attempts to overcome the problems of individual bias (Sheppard,1995). In the Delphi approach a group of people discuss the problems of estimation and finally arrive on a consensus estimate (Jalote,1997). Experts are used to make individual predictions secretly, based on their expertise and using whatever process they choose. Then the average estimate is calculated and presented to the group. Each expert has the opportunity to revise his or her estimate, if desired. The process is repeated until no expert wants to revise. Some users of the Delphi technique discuss the average before new estimates are made; at other times, the users allow no discussion. In another variation, the justifications of each expert are circulated anonymously among the experts (Pfleeger,1998).

If no consensus can be reached, "agreeing to disagree" is not the ideal solution as it can pre-empt the valuable ideas and approaches that come from attempting to reach consensus. An Inference Ladder (see Table 4.2) can be used for bones of contention. If agreement cannot be reached at the highest level (the choices to be made), the discussion moves down to the next level, to focus on the evaluation of the situation. Disagreement at this level moves the discussion to analysis, which leads to the formulation of the risk's characteristics (Gemmer,1997).

Table 4.2: Inference ladder as applied to the discussion of risk (Gemmer,1997)

Abstraction Level	**Can we agree on...**	**using this risk information ?**
Highest ↕ Lowest	What we should do about this situation?	Risk-handling strategy and action plan
	What is our evaluation of the situation? time frame and coupling	Risk statement and its risk probability and impact
	What reasoning are we using to reach this evaluation?	Rationale for risk probability and impact time frame and coupling
	What data are we using to support our reasoning?	Evidence, root causes, and risk tolerance

The inference ladder implies that if people cannot agree on the analysis of the risk, then it is reasonable to assume that the risk planning and strategy will be difficult to achieve. The benefit of the Delphi approach, is the capacity to reach consensus which will aid decisions concerning risk resolutions strategies. However, the Delphi approach relies strongly on expert judgement, and is therefore highly subjective.

4.2.2.2 Scenario Analysis

Typically probability and impact are assessed qualitatively in terms of scenarios (Chapman and Ward,1997). Scenario Analysis will acknowledge uncertainties and highlight critical sources of uncertainty. It will develop a range of possible future scenarios and strategies, and acknowledge the situation when data becomes meaningless. Scenario Analysis will not hide or remove uncertainty, develop one solution, or obtain unavailable market information. This approach helps companies to be responsive to different futures, but does not select a future (Molka-Danielsen,1996).

The simple scenario method advocated by Chapman and Ward (1997) contains the following steps:

1. Estimate a High Impact Scenario.
2. Estimate a Low Impact Scenario.
3. Define an Intermediate Scenario.
4. Estimate the probability of a Low impact.
5. Estimate the probability of a High impact.
6. Estimate the probability of an Intermediate Scenario.
7. Assessing the chance that the risk will occur at all. The rationale behind this step to clarify the overall nature of the risk realisation scenarios before estimating the probability that the risk will be realised.

Table 4.3 below shows an example of how scenarios may be formed where the scenarios are depicted in order of severity.

Table 4.3: Example depicting the simple scenario method (Chapman and Ward,1997)

Probability Risk is Realised	Impact If Realised in Months	Conditional Probability
0.2	8	0.2
	10	0.5
	12	0.3

A set of scenarios can be very useful for the success of risk analysis but scenarios are chosen in an ad hoc fashion, usually guided by the experience of domain experts. Therefore, it is likely that some important scenarios are overlooked, due to the complexity of the system (Cukic etal,1998).

The distinguishing characteristic between quantitative techniques and qualitative techniques in determining the loss and probability is that the former is based on an intermediate objective model. Quantitative methods feed subjective estimates into an objective model, while qualitative techniques merely give subjective estimates by using intuition and experience. The fuzzy metric technique does not fall into either of these categories, because the model is not entirely objective and therefore involves sending a subjective estimate into a subjective model.

4.2.3 Fuzzy Metrics

Fuzzy sets theory can be used to deal with the natural language descriptions of risk (Nidumolu,1996:82). As decision makers find expressing risk by linguistic values easier, fuzzy set theory provides a useful tool to deal with the ambiguity involved in the data evaluation process (Chen, 2001).

The risk assessment model employs a special kind of reasoning known as 'scalable monotonic chaining' which maps the risk specified in individual rules to an intermediate, risk-measuring fuzzy set. The result of this mapping is a scalar value from the domain of the risk metric indicating the degree of risk for a particular model factor. The monotonic reasoning results for each rule summed to produce a final risk value. This value is used to find the actual project risk.

Suppose we had the following rules (see Figure 4.4):

if project duration is long then risk is increased;

if project staffing is large then risk is increased.

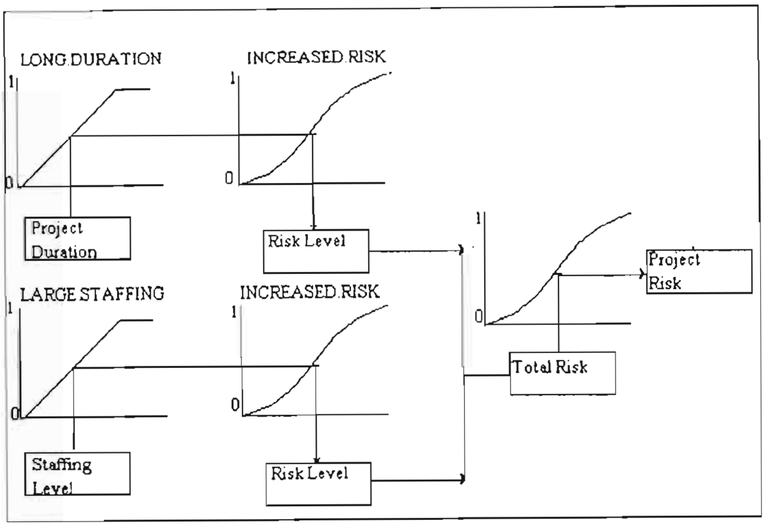

Figure 4.4: The monotonic chaining scheme used in risk assessment(Adapted from Cox,1994)

We then map these risks to an intermediate risk measuring fuzzy set, INCREASED.RISK. The result of this mapping is a scalar value from the domain of the risk metric indicating the degree of risk for this factor. The results for each rule are summed up to produce a final risk value. This value, called TOTAL RISK, is used to find the actual project risk (Cox,1994). As there are many attributes of a project that contribute to risk, formulating rules can be tedious.

4.2.4 Combining Qualitative Methods and Quantitative Methods

There are two basic ways to combine qualitative and quantitative methods. One method is to view the same risk both quantitatively and qualitatively, perhaps in the earlier stages of software development. Where information is lacking, risks can be assessed qualitatively. If the risk is still critical, it can be assessed quantitatively at a later stage. The second method is to use quantitative and qualitative methods concurrently, where the benefits of both are maximised and the shortfalls minimised. Here quantitative data can be used to feed into qualitative models to attain a qualitative estimate or vice versa to produce more insightful estimates.

Qualitative techniques are usually employed at the beginning of the risk management process to identify and rank risks. Those risks with a high or intermediate rank may be further analyzed through quantitative techniques. Methods such as scenario analysis can therefore be used in the initial stages (Baker,1998). During the earlier phase in software development, where information is lacking, qualitative measures might be more appropriate while quantitative measures might be more convenient in the latter stages as the list of risks becomes more specific. The timing is crucial because doing quantitative assessment too early can be tedious. This method of reviewing risks using both methods at different stages is problematic when the quantitative data does not match the qualitative data (Feather et al,2000).

The most problematic aspect of merely quantifying risk data is the possibility of misleading decision makers into thinking they can ignore or give less credence to qualitative data (Pfleeger,2000). A quantitative metric can be made much more informative if it is coupled with qualitative descriptions of the nature of the risk. For example, one can consider the impact and probability separately, look at the time frame in which to deal with the risk, the level of control of the situation, and consider the uncertainty of the situation, such as the familiarity of the risk. These characteristics of risk and their descriptions are summarised in Table 4.4.

Table 4.4: Depicting the characteristics and nature of risks (adapted from Gemmer (1997))

Characteristic	**Description**
Impact	Nature and magnitude of the risk consequences
Probability	Likelihood the risk consequences will become a reality
Time frame	Time during which the team can exercise proactive choices associated with a risk
Coupling	The effect of the risk on other risks
Uncertainty	Lack of understanding about the nature of risks probability distribution function over time

Even a qualitative description can be made more informative by adding on the above characteristics. Using specific characteristics to examine reasoning and evidence changes perception. Therefore instead of just giving a number or saying the risk was "small" it is best to qualify it with by considering all of the above dimensions of risk. The risk analysis process involves understanding the situation better and gaining as much information as possible so that the risk can be dealt with more effectively (Gemmer,1997).

The problem with making risk assessments is the 'inherent limitations of human cognition'. Sometimes when expectations are high, evaluating the risk can be skewed, often minimising the risks and maximising the payback. Here credibility is very important. People evaluate risk based on the risk information that is being relayed to them and therefore it is important that risk analysts be wary of hype. People tend to make decisions according to the source of information they receive. The more they trust the sender, the more willing they are to accommodate risk in their decision making.

The estimation of impact and probabilities is likely to be subjective, time-consuming and costly (Hughes and Cotterel,1996). The ease of administering a risk analysis process is important because in project management the time frames are short. Using simple methods like a probability impact grid to convert a qualitative assessment into a quantitative assessment (Chapman and Ward,1997) might be useful in this regard (see Figure 4.5). Another simple technique such as assessing the risk probabilities and losses on a scale of zero to ten can be easier to administer than, say, a group consensus (Boehm,1991).

		Low	**Medium**	**High**
PROBABILITY	**High**	2	1	1
	Medium	3	2	1
	Low	3	3	2

IMPACT

Figure 4.5: A probability impact grid (Chapman and Ward,1997)

4.3 Estimating the Risk Exposure

As risk implies a potential loss, there are two elements of risk: firstly the probability of an unsatisfactory outcome and secondly the consequences of such an outcome (Fairley,1994). These two measurements must be estimated using risk analysis techniques, the multiplicity of which forms the risk exposure metric which is defined by the formula below:

$$Risk\ Exposure\ =\ Prob(UO)\ \times\ Loss(UO) \tag{4.6}$$

where

Prob(UO) is the probability of an unsatisfactory outcome

Loss(UO) is the loss to the parties affected if the outcome is unsatisfactory.

The risk exposure can be used to determine the degree of attention each risk need be given. Risk exposure has several magnitudes, as risk can impact on the cost, schedule, performance and quality (Chittister,1993). Ideally the risk impact is estimated in monetary terms (Huges and Cotterel,1999). But this can differ depending on the risk being analysed, for example personnel risks can be measured in man-months, while schedule risks can be measured in weeks or months. Table 4.5 shows how a risk exposure metric can be used to assess risks identified.

The table (4.5) presents a list of possible risks to the success of the project such as "Software error kills experiment" and provides an estimation of a probability and loss consequence of that particular risk occurring. The risk exposure metric is then calculated using these two estimates. In some cases the estimates are not specific values, indicative of the difficulty of assigning a single value to risk. For example, the probability of an unsatisfactory outcome for risk A, is given a probability of between three to five, thereby allowing the analyst to view this risk within a range of risk exposures.

Table 4.5: Example depicting risk exposure calculations of risks identified (Boehm,1991)

Unsatisfactory Outcome	**Probability**	**Loss**	**Risk Exposure**
A. Software error kills experiment	3 to 5	10	30 to50
B. Software error loses key data	3 to 5	8	24 to 40
C. Fault-tolerant features cause unacceptable performance	4 to 8	7	28 to 56
D. Monitoring of software reports unsafe condition as safe	5	9	45
E. Monitoring software reports safe condition as unsafe	5	5	15
F. Hardware delay causes schedule overrun	6	4	24
G. Data reduction software errors cause extra work	8	1	8
H. Poor user interface causes inefficient operation	6	5	30
I. Processor memory insufficient	1	7	7
J. Database-management loses derived data	2	2	4

The graph in Figure 4.6 shows the risk exposure contours (RE) and the risks from Table 4.5 mapped on it (Boehm,1991) where the probability is represented by the abscissa and the loss by the ordinate (Blum,1992). The risk exposure contours represent a family of curves for fixed risk exposure values of ten, twenty-five and fifty. Those risks with a specific risk exposure are merely indicated by a point on the graph such as D, E, F, G, H ,I and J while risks A, B, C are indicated by a bidirectional line, extending over the range of risk exposures possible for that particular risk. The analyst can view the risks on this map, noting its position relative to the probability and loss, and relative to the family of risk exposure curves. This allows a risk to be viewed within three possibilities:

- As loss consequence
- As probability
- As an impact.

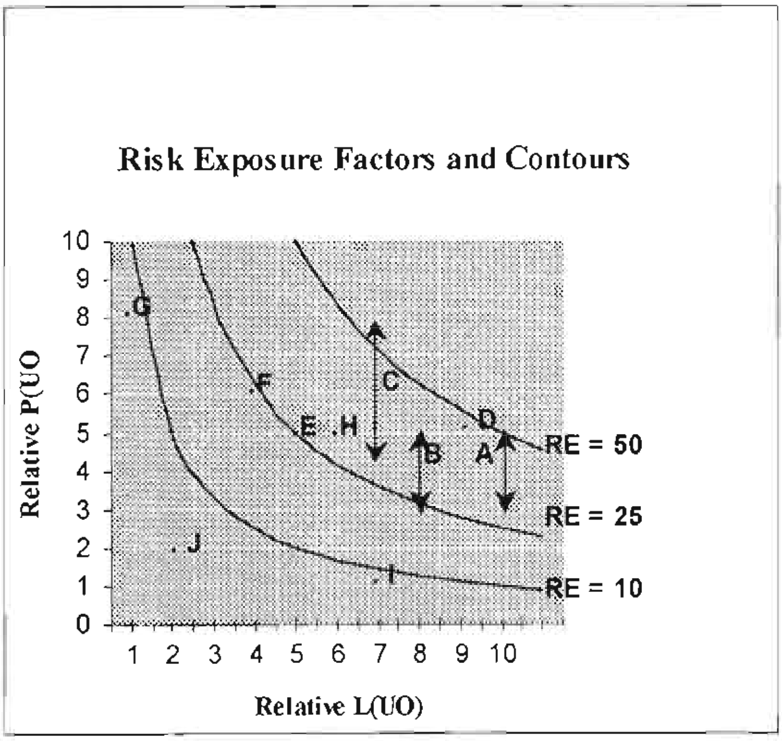

Figure 4.6: Depicting the points plotted from Table 4.5 on the risk exposure graph (Boehm,1991)

It is evident that the risk exposure and probability can change over time (Pfleeger,1998) and therefore the probability and loss must be analysed and tracked over time (Conrow and Shishido,1997:85). The graph in Figure 4.6 can help to assess the risk in this way as it allows the measures of probability and loss to be viewed separately. Therefore risk exposure can be used in managing the risk, which involves the use of two strategies which are either reducing risk's exposure by reducing the probability or by reducing the associated loss (Gemmer,1997). For instance, if staff turnover is one of the risks a project faces, then the probability of such a risk occurring can be reduced by empowering exemplary performers. The loss can also be reduced by good configuration management to make it easier for new replacements to master existing software modules if the team members leave (Boehm and DeMarco,1997).

The perception of the probability of risk occurring depends on several psychological factors such as overconfidence or selectivity and (Gemmer,1997) can be viewed differently depending on the viewpoint. Given that projects involve several classes of participants (customer, developer, user, maintainer), each with different but highly important satisfaction criteria, it is clear that the probability factor is multidimensional. For example, for customers and developers budget overruns are unsatisfactory, for users user-interface shortfalls are unsatisfactory and for maintainers poor quality is unsatisfactory (Boehm,1991). One way of resolving differing viewpoints and perceptions is to use a risk impact model. An impact model is a model that defines thresholds of variances in performance in relation to the programme's expectations (see Figure 4.6). The model provides a common scale for measuring different types of impact (Gemmer,1997). The model can be manipulated to suit any particular circumstances or environment.

Table 4.6:Organization-wide guidelines to assess risk(Impact Model) (Adapted from (Gemmer,1997))

Impact	**Variance in Programme Performance From Expectations**
Catastrophic (impact of 0.75 to 1.00)	Cost increases of more than 20% of the budget. Third miss of a customer delivery schedule even by one day
High (impact of 0.5 to 0.75)	Cost increases of more than 10% of the budget Missing any customer delivery schedule even by one day
Medium (Impact 0.25 to 5.0)	Cost increases of more than 5% of the budget Slip in internal schedule
Low (Impact of 0.00 to 0.25)	Cost increases of 2.5 % of the budget Task slip that reduces margin

The impact model is a two-way communication tool, that allows subordinates to communicate risks in such a way that the managers understand them and also communicates management's desires for feedback (Gemmer,1997). It is a mechanism to find points of commonality since it is difficult to make quantitative estimates. Obtaining an end result of risk exposure should not be the bottom line, as the journey to reach the estimate can be more insightful than the actual figure that is determined at the end (Gemmer,1997).

The difficulties in making estimates can also be alleviated by using a risk magnitude level matrix to make qualitative estimates (see Figure 4.7). The Software Risk Evaluation method developed by the Software Engineering Institute uses the risk magnitude level matrix, which is a quick and easy chart to determine the risk magnitude. Additionally each risk is recorded with its condition, consequence and source, and classified as belonging to a class, element or attribute (Adler et al,1999).

P(U0)
L(U0)
	Very Likely	Probable	Improbable

Catastrophic	High	High	Medium
Marginal	High	Medium	Low
Negligible	Medium	Low	Low

Figure 4.7: Risk management level matrix (Adler, et al,1999)

The next stage would be to determine what steps are needed to take control of the risks. The risk exposure is useful in ranking risk for drawing up contingency plans to deal with the most critical risks (Gemmer,1997). Some risks, once recognised, can be reduced or avoided immediately with little cost and effort but often the cost of taking action needs to be weighed up against the benefits of reducing the risk (Hughes and Cotterell,1999). If the leverage value is not high enough to justify the action, then a less costly or a more effective reduction technique should be considered.

$$\text{Risk Leverage} = \frac{\text{Risk}_{\text{exposure before reduction}} - \text{Risk}_{\text{exposure after reduction}}}{\text{Cost of risk reduction}} \qquad (4.7)$$

The risk exposure corresponds to the contingency needed to protect a project from risk. But if the risk exposure is incorrect, then the allowed contingency cannot protect a project if the original assumption was incorrect (Kitchenham and Linkman,1997). The problem with using just risk exposure to rank the risks is that it leaves out some critical points. For instance, the risk with the biggest risk exposure may not be the most important (Hughes and Cotterel,1999).

4.4 Comparisons of Risk Analysis Techniques

Quantitative risk analysis techniques are historically favoured over Qualitative risk analysis techniques as the former provide a systematic and documentary approach to the management of risk. The quantitative methods are more objective and are based on models and metrics whilst qualitative methods are more subjective. For example, one analyst may report a certain risk as "high"while another may report the same risk as "medium". Therefore the words, "high" and "medium" are relative terms and therefore, do not provide an accurate evaluation of the risk. However quantitative models such as the cost-risk analysis models are usually automated, portraying a final amount. This final amount seduces analysts into overlooking dependency between individual sources of risk (Chapman and Ward,1997).

The validity of the risk exposure metric, whether quantitatively or qualitatively attained, is compromised by two types of uncertainty: Firstly uncertainty in perceiving impact and secondly, uncertainty in perceiving probability (Gemmer,1997). These values that are estimated for the probability and impact are 'manipulated with very positivistic formal and logical mathematical operations' and if the original values are incorrect then the 'probability arithmetic that follows is complete nonsense'(Baskerville and Stage, 1996:484). That implies that the risk exposure is only as good as the estimations in its inputs are. The difficulty with risk exposure is the problem of making accurate estimates (Boehm,1991). If risk exposure is determined quantitatively, the accuracy is compromised by a further consequence. Quantitative methods ignore the intuitive issues and this tends to create an illusion of accuracy (Steen,1997) that is not in fact present. Numbers attained quantitatively tend to obscure reality (Pleeger,2000). At least qualitative estimates improve the quality of decisions since decisions based on intuitive estimates may be worse than making decisions without qualitative estimates (Gemmer,1997).

The first aspect to consider when using a quantitative method is whether or not an imprecise measure is cost effective enough to justify its use. The most desirable metric is a direct measurement of the property of interest. This is not possible with risk analysis as risk is inherently about the future and making predications is unavoidable. Therefore the metric is only as effective as the expert who is employing it. The more experienced or knowledgeable the expert is, the more accurate the prediction is. Guarro (1986) advocates the use of quantitative techniques because feeding subjective estimations into an objectively based model is more desirable than basing decisions on subjective decision schemes. However, the model itself may be subjective because all models are abstractions of reality and no model can include all the catalytic factors. Being cognisant of the subjectivity does not imply that it should be accounted for, since making provision for assumption errors is tantamount to 'double-counting the effects of uncertainty' if uncertainty is accounted for in the model itself (Kitchenham et al,1997).

No amount of empirical information can predict the future with certainty (Strigini,1996). However, the inaccurate estimates do not negate the need for attaining historical data, because forming a historical database of measures could make predictive measures more accurate in future. One method of capturing expert knowledge and making it widely available is through employing knowledge-based tools such as expert systems. Since risk management requires expert knowledge, it is a natural application for knowledge-based approaches (Toth,1994). However, each software system is unique, hence generalizing risk assessment from previous projects will not necessarily help (Voas,1997). One way of reducing the problems of making subjective estimates is to buy information early on to gain insight into the final product. Prototyping is one such method (Adler et al,1999). The second consideration is that the insistence on quantifying a risk can lead to a "paralysis of analysis" and a breakdown in risk communications (Williams, et al,1997).

The fact that quantitative risk analysis techniques are 'Considered as a scientific or a statistical approach' is often tantamount to a 'shallow exercise in simple guesswork'(Baskerville and Stage,1996:484). Under the normal science paradigm, a project is well defined enough, so that not only is there adequate information, but that information is accurate enough to permit the prediction of future events (Charette,1996:113). Paradoxically, the existence of risk management is an acknowledgement that the 'normal science paradigm' does not work. However, quantitative risk analysis is positivistic in nature itself. Therefore risk analysis principles based on normal science principles can lead to problems as happens in software project management. For example, software project management assumes that schedules and budget can be accurately predicted. Analogously the existence of the risk exposure metric promotes the same belief that risks can be accurately predicted. Kitchenham and Linkman (1997) assert that effort estimation with work function points is problematic as it is assumed that this estimation model can include all the factors that affect the effort required to produce a product. The same logic can be applied to the risk exposure metric because it cannot include all the factors that contribute to risk affecting the outcome of a project since, like other estimation models, it is merely an abstraction of reality (Kitchenham and Linkman,1997). Most risks have some combination of political, social, economic, environmental and technical factors, and therefore it is difficult to place a "hard" number on something when there are so many factors coupled with the obscure nature of personal perceptions (Gemmer,1997). Absolute risk does not exist, and it depends to a great extent on the individual's perspective (Barki and Rivard,1993).

Using the latest quantitative methods and tools cannot account for people's perceptions of risk and their attendant behaviours. It is important to recognise and minimise biases in perceiving risk (Gemmer,1997). The normal science model with its practice of using "hard" scientific inputs to make "soft" policy decisions is not prepared to resolve these new complex types of questions where "hard" policy decisions have to be made using "soft" scientific inputs (Charette,1996:112). While one or more formulas and methods for weighting and measuring risk are beneficial, they are not essential to implementing risk management (Kuver,1999).

The time spent identifying, analysing and managing risks pays itself back in many ways. Both quantitative and qualitative methods are useful at different junctures. The latter are more appropriate in the earlier stages. Quantitative techniques can be used to provide a more accurate estimate of the risks identified with qualitative techniques. It is not important if something has a subjective value, but the end result should be a 'list of the most important current risks to the project'(Blum,1992).

Irrespective of the technique applied, risk analysis in itself can be risky. Firstly, one can overestimate the risk and then spend considerable time and effort in eradicating a risk that is unlikely to occur. Secondly, one can underestimate a risk and then be lulled into a false sense of security (Charette,1991). Thirdly, risk analysis is time consuming and there is also the potential for overanalysing risk which results in stagnation in a "paralysis analysis" mode, which in turn results in indecision.

Ultimately risk analysis cannot divorce itself from the nature of software, and it is difficult to quantify and make predictions about an entity that is not entirely comprehensible. The reservations about the merits of conducting risk analysis due to inaccuracy do not outweigh its usefulness during decision making as the choices made are 'more informed and wise rather than isolated, or worse, repetitions of past mistakes'(Bell,1989:51). The problem with conventional risk assessment is that it can never be value-free (Pfleeger,2000). At the very least, risk analysis can facilitate a risk-aware culture rather than taking a reactive approach to risk and assuming the non-existence of risks.

4.5 Conclusion of Risk Analysis Techniques

The need to use both quantitative techniques and qualitative techniques reflects the diversity of problems in risk analysis in specific software development environments. This issue underlines the need for a better understanding of how risk management is conducted in organizations in order to gain an insight into the way it affects organizations. The analysis of the results of this chapter provided the rationale for the epistemological stance taken for the framework developed in this project. It became evident, that conducting a positivistic study, was not going to provide insight into the contextual factors which seem to have an impact on the application of the tools and techniques applied. Hence in the study conducted, the role of the contextual aspects has been awarded special attention, which is the scope of the investigation, described in the next chapter.

CHAPTER 5

PRINCIPLES OF INTERPRETIVE RESEARCH AND THE FRAMEWORK FOR INVESTIGATING RISK MANAGEMENT PRACTICES

"Paradigms should serve as a lens to illuminate research issues, not as blinkers to help achieve closure"
Fitzgerald and Howcroft (1998:320)

5.1 Introduction

The social elements in information technology in terms of design, development and implementation (Davis et al, 1992:294) are omnipresent. The realisation that the positivist research paradigm from natural sciences did not correlate with the social elements in software development led to an investigation of social research methods. Interpretive research is the way to study the intersection between the technical and non-technical in software engineering (Seaman,1999:557). The main goal of this research is to provide an interpretivist framework for determining the effectiveness of risk management in field conditions and to apply it to an exploratory investigation of risk management practices in several software organizations. The framework is not only a mechanism for investigating the perceptions and perspectives of risk management, but it also posits itself as a useful tool for organizations to gain insight into their own risk management strategies. A tool that can be used to determine the shortfalls with a particular risk management strategy or to highlight the significance of applying a risk management strategy. The framework provides a mechanism to pinpoint the problems with risk management in the context of an organization so that they can be identified and addressed.

In an exploratory investigation, the framework, can be used to determine if risk management is practiced successfully, what kind of contextual factors are instrumental in guiding the risk management process and why? It is also important to determine the converse of the previous statement, and the factors and attitudes that cause an organization to ignore risk management practices. The aim of this field investigation is to attain a perspective of risk management as practiced in reality as there is a dearth of studies on real experiences of software processes (Dutta et al,1998:79).

The more traditional epistemology in both information systems and computer science research is a positivist one and this mode of thought is also prevalent in the way methodologies are developed and in the way these models are researched. However, creating a control group to study the effectiveness of software processes is very difficult. For example, few organizations will assign two groups to execute the same project using different methods or processes (Deephouse and Mukhopadhyay,1996:190). The scientific approach works best for evaluating quick tactical improvements and not nearly so well for the evaluation of larger scale strategic changes (Laporte and Papiccio,1997). Although greater credence was given to a positivist approach in the past and most institutions advocate such an approach because it is perceived as being more "scientific" (Galliers,1987:900), a non-positivist epistemology is adopted for this study as there is a significant recent tendency in information systems research (see Lee (1999) and Markus and Lee (1999)) in that direction.

This research falls within the area of software engineering. 'The scope of software engineering is extremely broad. Some aspects of software engineering can be categorized as mathematics or computer science; other aspects fall into the areas of economics, management or psychology' Schach (2002: 4). Hence software engineering can be seen on the border of computer science and information systems. The interpretive approach used in this research reflects rather the closeness of software engineering and information systems to the social sciences because of their involvement of the human element. A substantial body of research has focussed on the need to develop methodologies that support viewing information systems as social constructs (Hirchheim and Klein,1994). The literature survey conducted on the benefits of interpretive research was found to be mostly limited to the field of information systems, but the value of interpretive research can be extrapolated to the field of software engineering as well.

This study will attempt to provide rich, descriptive data about the contexts, activities and beliefs of project managers concerning risk management and simple supporting statistical data. Table 5.1 shows the differences between positivist and interpretive research.

Table 5.1: The differences between interpretive and positivist research (adapted from Stone,1990)

Characteristics	Positivist	Intepretivist
Evaluator's Role	Onlooker	Participant
Evaluator's Relationship to setting	Detached, Neutral	Immersed, Involved
Validation Bias	Measurement and Logic	Experiential
Sources of categories	Predefined	Emergent
Knowledge Acquired	Universal, of or relating to the study or discovery of general scientific laws.	Particular, Relating to or concerned with discrete or unique facts or events
Nature and meaning of data	Factual, Context-free	Interpreted, Contextually Embedded
Evaluation Language	Quantitative (High precision, low variety)	Qualitative (Low Precision, high variety)

5.2 Reasons for Choosing an Interpretive Epistemology

There are three fundamental motivations for adopting an interpretive approach to this study. Firstly interpretive research can help one to understand the sociological aspects in the software development setting as it focuses on human thought and action in social and organizational contexts. This epistemology was complementary to the central theme of this study, which considers the perspectives and perceptions of software project managers concerning risk management. The strong emphasis on the sociological aspects is essential because the exclusion of human factors in past research may explain some of the dissatisfaction with conventional information systems development (Avison et al,1999:95). It is believed that this might explain the repudiation of risk management strategies hitherto.

The second reason for choosing an interpretive epistemology relates to the abstruseness of the research area under enquiry. Interpretive research does not predefine dependent and independent variables, but focusses on the complexity of human sense-making as the research proceeds (Galliers,1987). For instance, this study seeks to discover the rationale behind practitioners not utilising risk management, which by all literary accounts, is a highly positive process and is essential for producing risk-free software. The author does not negate the need for positivist research in software engineering, but supports the belief that the approach should be dependent on the research question concerned. Interpretive research can be the precursor to positivist research, especially in situations where there is a large amount of fuzziness. Interpretive research tries to understand all the nuances of the phenomena, in order to obtain clarification so as to make "sense" of the situation. Positivist research does not work when the phenomenon under scrutiny is ill-defined. Therefore the nature of interpretive research is such that it can demystify the phenomena in question, leading the way for positivist research to be conducted more astutely. Positivist research draws inferences from phenomena which will not work when the phenomenon is ill-defined.

The debate between positivist and interpretivist research paradigms is vacuous as each approach has its own strengths and weaknesses. Neither paradigm is superior to the other, but if interpretive research is viewed through the positivist lens, it will be accorded an inferior status (Fitzgerald and Howcroft, 1998). Softer research approaches are suitable for exploratory research while hard research methods are suitable for confirmatory research (Fitzgerald and Howcroft,1998:322). Monistic models are appropriate for subjects like physics, but not for a fragmented field such as information systems (Banville and Landry,1989:58) development. The strength of this type of research is its ability to represent reality. Significant advances in knowledge and developing theory can be made in this way (Hamilton and Ives,1992). The preceding argument, in terms of the research approach being dependent on the research question, is not tantamount to the argument that qualitative research is 'preliminary to the "real" research of generating hypotheses to be tested using experimental or statistical techniques' (Kaplan and Duchon,1988:574).

Lastly, this study is reality-based. The value of this type of approach is the explaining of 'what goes on in organizations'(Avison et al,1999:94). This is important because as Benbasat and Zmud (1999:4) contend, the relevance of information systems research is being questioned by the business community. The business community, according to the same authors, considers issues covered in current information systems publications as irrelevant owing to three factors. Firstly, the lack of applicability to reality; secondly a lack of contemporaneous issues; and thirdly inaccessibility, as Banville and Landry (1989:57) argue that 'statistical methods, reduce task uncertainty' but 'restrict audiences and give access to prestigious audiences'. Banville and Landry (1989:57) emphasise that fields like business finance, have 'maintained strong connections with the practitioners, whose problems have always been considered worthy research topics and who, being educated by the academics, have always applied to their practical problems the sophisticated methods they have learned', Benbasat and Zmud (1999:7) indicate that what 'tends to be absent is rich, loosely-structured dialogues of the opportunities and problems being experienced in practices and discussions of how these might be examined through academic research'. Owing to the convergent relationship between information systems and software engineering, this argument in favour of interpretive research in information systems holds true for interpretive research in systems development.

One additional benefit of interpretive research is that the focus does not have to be limited by hypotheses testing and tight experimental control. Instead, the external validity of the actual research question and its relevance to practice is emphasized, rather than restricting the focus to what is researchable by rigorous methods (Fitzgerald and Howcroft,1998:320). Banville and Landry (1989:51) also corroborate this view with the assertion that the 'normal scientist observes only what his paradigm tells him to observe and most of the observations that do not fit in this tight schema either go on unnoticed or are put aside as irrelevant, or better, for the sake of "progress" as something that cannot be explained yet'.

The foundations of interpretive methods are to be found in the disciplines of sociology and anthropology (Davis et al,1992:302). A common misconception is that interpretive research and qualitative research are synonyms. Moreover, qualitative research is often seen as being non positivist, anti-positivist or interpretivist (Lacity and Janson,1994:138). Qualitative research can be done with a positivist or interpretivist stance (Klein and Myers,1999:69). As explained in chapter One, the positivist approach was rejected for this study and the interpretive view chosen instead. The following section elaborates on the philosophical stance taken in this research.

The software development process is complex and the best way to comprehend it is to elicit information from those who experience it. Therefore, an interpretive approach is the epistemology best suited to achieve this end. The vehicle for interpretive investigation is usually an in-depth case study although this has been criticised for "observer bias" results. The observer bias results out of what the researcher wants to see and what people want the researcher to see. Case studies usually take a long time and through the process of "immersion", the researcher inevitably influences the people who are being researched (Walsham,1995:77). Therefore interpretive approaches have been criticized for being too subjective and relativist (Neuman,1997). To circumvent this, case study researchers use different sources to add rigour to the research. A primary source is interviews, 'since it is through this method that the researcher can best access the interpretations that the participants have regarding the actions and events that have or are taking place' (Walsham,1995:78). Interviews can open up a richness of information that is hard to obtain quickly in any other way. It helps one to view the expectations of a process as it influences the decision makers and this sort of information cannot be obtained by quantitative studies alone(Laporte and Papiccio,1997).

As this approach is not positivist but interpretivist in nature, it is important to highlight the differences between a survey conducted under the positivist paradigm and a survey conducted under the interpretivist view. These differences are articulated in Table 5.2.

Table 5.2: Highlighting the differences in conducting a survey with a positivist approach and a survey with an interpretivist approach (adapted from (Neuman,1997; Moore 1999))

Positivist Approach to a Survey	Interpretivist Approach to a Survey
Objective Questions - the same questions are asked of everyone.	Placed in context - people's history/place of interview
Aggregate the answers	The answers are fully documented in a qualitative way
Same questions are asked with specific answers	A person's answer may vary depending on the interview context
Large sample	Smaller sample
Samples are random	Samples are chosen purposively
Mostly closed-ended questions	Mostly open-ended questions

5.3 Philosophical Basis for Interpretive Research

In the latter part of the 19th century, dissenters to positivism emerged, owing to dissatisfaction with it. The term "*verstehen*" was particularly prominent in the writings of Edmund Husserl at this time (Hircheim,1992). The term *verstehen* is associated with the connotation of mutual understanding, where one of the parties to the mutual understanding is a scholarly observer. This philosophy of social science is sometimes referred to as the *interpretive* understanding (Lee,1994). However, this notion did not prevail over positivism although, with emergence of the post-positivism era in the 1980s, researchers recognized that orthodox science was inappropriate to social enquiry (Hirschheim,1992:59). The strong ties between social science and information systems in terms of organizational and behavioural research called for interpretive research methodologies to be considered in information systems research (Kaplan and Duchon,1988).

The interpretive approach recognises actions, events and artefacts as occurring within human life and not as an observation of some external reality. Interpretive information systems research might be characterised by an intention to understand the implication of information technology in organizational activity through understanding the context of the information system, and the processes whereby the information system influences and is influenced by the context

(Doolin,1998:302). In this study the experience of the participant in risk management is considered, along with the size and nature of the organization. According to Lacity and Janson (1994) and Fitzgerald and Howcroft (1998) interpretivist research has the following epistemological characteristics:

- reality is socially constructed;
- multiple realities exist;
- research is time and context-dependent;
- interpreter's biases are acknowledged.

Additionally, according to Klein and Myers (1999:72) interpretive research should be based on the following principles:

- The hermeneutic circle: This principle suggests that all human understanding is achieved by iterating between considering the interdependent meaning of parts and the whole that they form. The principle of human understanding is fundamental to all other principles.
- The principle of contextualization: Requires critical reflection on the social and historical background of the research setting, so that the intended audience can see how the current situation under investigation emerged.
- The principle of interaction between the researchers and subjects: requires critical reflection on how the research materials (or data) were socially constructed through the interaction between the researchers and participants.
- The principle of abstraction and generalization requires relating the ideographic details revealed by the data interpretation through the application of the above principles to theoretical, general concepts that describe the nature of human understanding and social action.
- The principle of dialogic reasoning requires sensitivity to the possible contradictions between the theoretical preconceptions guiding the research design and the actual finding with subsequent cycles of revisions.
- The principle of multiple interpretations requires sensitivity to possible differences in interpretations among the participants which are typically expressed in multiple narratives or stories of the same sequence of events under study. These are similar to witness accounts even if all tell it as they saw it.

- The principle of suspicion requires sensitivity to the possible “biases” and systematic “distortions” in the narratives collected from the participants.

The ontological assumption of interpretive research is based on the notion that an individual constructs his own reality. Much of interpretivism is based on ethnomethodology. But there are different philosophical foundations for interpretivism, such as phenomenology and hermeneutics (Walsham,1995:75). Interpretivism holds that reality is a subjective construction of the mind (Hirschheim, 1992).

When a researcher is motivated to understand the experiences of human beings in terms of a particular process, in this case the risk management process, then phenomenological research is the most appropriate methodology to follow. Phenomena, according to phenomenological research, are embedded in a web of meaning related to human experiences, that is, things learned via intuition and imagination (Moreno,2001). In this research the phenomena of software development risks, under the influence of risk management, experienced by individuals in the software development environment were investigated. To distinguish the philosophical underpinnings of phenomenology from other commonly used social ideologies, the next sections provide brief descriptions of ethnomethodology and hermeneutics.

5.3.1 Ethnomethodology

Ethnography and ethnomethodology do not have a one-to-one relationship. Ethnography is based on ethnomethodology, phenomenology and social interactionism (Beynon-Davies,1997:532).‘The term "ethnomethodology" therefore refers to the study of a particular subject matter: the body of common sense knowledge and the range of procedures and considerations by means of which ordinary members of society make sense of, find their way about in, and act on the circumstances in which they find themselves’(Henning,1998). This branch of sociology has particularly emphasized the way in which people continuously have to work at making their own actions make sense to others (Beynon-Davies,1997). ‘It shifts the emphasis away from the production of sociological accounts and theories of social doings to an emphasis upon the description of the accountable practices involved in the production of naturally organised*[sic]* phenomena.’

(Button,2000). Ethnomethodology is concerned with the elaboration of the methods underlying the practical accomplishment of everyday behaviour (Beynon-Davies,1997).

5.3.2 Hermeneutics

Hermeneutics involves the art of reading a text so that the intention and meaning behind appearances are fully understood. The hermeneutic process involves a circle through which scientific understanding occurs when prejudices are set aside, allowing the text to speak for itself (Moustakas,1994). The hermeneutic circle is the foundation of all interpretive work, being based on the notion that understanding is a constant iteration from understanding the whole to the sum of its parts (Klein and Myers,1999:71).

The goal of hermeneutics is twofold: first to ascertain the exact translation of a text, and second to discover the instructions contained in the text, using rules which allow the researchers to identify the intentions of the author and place his or her meaning within its historical and cultural context (Lacity and Janson:149-150). As interpretive research denies the possibility of objective knowledge (Beynon-Davies,1997), these "rules" reflect the background of the interpreter although they are presumed to capture the author's intentions (Lacity and Janson:150). The researcher who admits these biases falls into the hermeneutic tradition as well (Galliers,1987:901).

When interpreting a text, one basic principle applied is that the author of the text knows his or her subject (Davis et al,1992:302). This principle can be used to understand the behaviour of information systems professionals in a crisis situation in the information systems context. Boehm (1997:18) typifies this ideology with the "can-do" mentality displayed by project managers, who, when schedules are slipping, think that the obvious solution is to add on more staff, instead of dealing with the risk itself. That is, people behave in ways they think are rational responses to their situation (Klein and Myers,1999). Language is a transmitter between actual experiences, traditions and the process of understanding (Hirschheim,1992:57) therefore Boland (1984:194) contends that an appreciation of hermeneutics, constitutes the justification for phenomenology in information systems research.

5.3.3 Phenomenology

The relationship between phenomenology and information systems is best described by Boland (1985:200) in his seminal article, "Phenomenology: A Preferred Approach to Research on Information Systems": 'Data becoming information is what information systems are. Data becomes information in the consciousness of a human subject, and that is where we must look if we are to understand information systems. Phenomenology as a social science method holds the best promise for doing so because it is the one method designed with that purpose in mind.'

Interpretive research in the tradition of phenomenology is concerned with the description (Galliers, 1987:901) and analysis of everyday life (Beynon-Davies,1997). It concentrates on the common aspects of individual experiences in order to identify themes and social meanings related to the phenomena of interest (Moreno,2001). Its basis lies in the claim that true knowledge is not the physical but the 'realm of pure thought'(Mingers,2001:106). 'Phenomenology is based on the "intuitive grasping of esssences[*sic*]" of phenomena' where the essences are more concerned with the 'how' and 'why' than the 'which' and 'what' issues (Hirchheim,1992:48). Where the phenomenon is the 'essence of our experience, that is 'which remains after accidents, contingencies and presuppositions we bring to our everyday experience in the life-world are stripped away' (Boland,1985:193). So 'phenomenology is interested in the methodical study of consciousness in order to understand the essence of experience'(Boland,1985:194).

The phenomenological disposition involves giving up the natural science attitude and its assumptions and instead exploring the experiencing subject in as uncommitted a way as possible (Mingers,2001:106). The emphasis on intuition, imagination and universal structures in obtaining a picture of the dynamics that underlie the experience, account for and provide an understanding of how it is that particular perceptions, feelings, thoughts and sensual awareness are evoked in consciousness (Moustakas,1994).

Within phenomenological research, there are two strains of thought, one emanating from Hussel (1859-1938), considered to be the father of the phenomenology (Melville and Goddard,1996), who believed in the need to demonstrate phenomenology as based on a pure subjectivity that yields a pure objectivity (Boland,1985:195). The other thought is that it is impossible to make phenomenology objective because it is impossible to strip away all the assumptions of the interpreter (Boland,1985). The author adopts Husserl's ideology, which is known as 'transcendental phenomenology', in an effort to be as objective as possible. Qualitative research should admit to subjectivity while attempting to be as objective as possible. The researcher in transcendental phenomenological research engages in disciplined and systematic efforts to set aside prejudgements regarding the phenomenon being investigated. This is done in order to 'launch the study as far as possible free of preconceptions, beliefs, and knowledge of the phenomenon from prior experience and professional studies to be completely open, receptive, and naive in listening and hearing research participants describe their experience of the phenomenon being investigated' (Moustakas,1994:22).

5.4 Justification for the Methods Used in Interpretive Research

Human behaviour is one of the few phenomena that is complex enough to require qualitative methods to study it (Seaman,1999:557). There are several research methods associated with qualitative research. These are ethnography, participant observation, interviews, conversation analysis, grounded theory development, case studies and action research.

Ethnography allows for the study of the organizational culture under investigation (Sayer,1998:249) and involves the researcher immersing himself in and recording the life of a social group in a natural setting for a long period of time (Beynon-Davies,1997; Klein and Myers,1999). Action Research involves the formulation of a theory, intervention and action-taking in order to introduce change into the study subject. It also involves the analysis of the ensuing change behaviour of the study subject (Baskerville and Pries-Heje,1999:1). Grounded Theory seeks to develop a theory that is grounded in data systematically gathered and analyzed, where there is a continuous interplay between data collection and analysis and where theory is generated during the research process from the data being collected. Grounded Theory does not begin with the theory and then seeks to prove it. Instead it allows the theory to emerge from the study (Baskerville and Pries-Heje,1999:5).

Irrespective of the methods used, according to Moustakas (1994) all qualitative research has the following commonalities:

- recognising that the human experiences cannot be effectively captured by quantitative techniques;
- focusing on the wholeness of experience;
- obtaining descriptions through first-person accounts acquired during interviews;
- regarding the data of experience as imperative in understanding human behaviour and as evidence for scientific investigations;
- viewing experience and behaviour as an integrated and inseparable relationship of subject and objects, and part and whole.

There are four reasons why face-to-face interviews were chosen as the data collection method. Firstly, unstructured interviews are characteristic of the philosophical stance taken, that is the phenomenological stance. Secondly the study of phenomena requires more than one single site and event that are characteristic of action research, case studies, ethnography and grounded theory (Hamilton and Ives,1992). Thirdly, traditional interpretive methods such as ethnography, case-studies and grounded theory are not feasible when dealing with more than one site for the application of the framework as is the case in this project. These are time consuming, and their basic requirement is the process of immersion into organizations. It is difficult to gain long-term access to organizations. The fourth reason is the high level of subjectivity in other methods. As traditional qualitative methods require observational techniques, the researcher inevitably sees what he or she wants to see and the participant demonstrates characteristics of what he or she wants the researcher to see. Therefore, these methods of inquiry are often criticised as highly subjective. The author contends that the best information is first-hand information; and interview techniques were therefore considered more practical in this study.

At this juncture, it has been established that the interviewing methodology with a Transcendental Phenomenological approach is seen as the best means of conducting this research. The next issue of concern is how this approach differs from other approaches.

According to Moustakas (1994), 'the core processes that facilitate the derivation of knowledge' in transcendental phenomenological research are:

- Epoch, which involves the researcher setting aside understandings and judgements when approaching the participant.
- Reduction, which is a goal to obtain rich, accurate and complete textural description of the experience. One must give equal value to all perceptions. Irrelevant, redundant and overlapping aspects of the narratives are eliminated, leaving only the textural meanings and invariant constituents of the phenomenon. These are clustered into themes, which give the main directions of the final textural description.
- Imagination Variations, the aim of which is to grasp the structural essences of the experience which involves considering how the experience of the phenomenon came to be what it is.
- Intuitive integration of the fundamental characteristics and structural descriptions into a unified statement of the essences of the experience of the phenomenon as a whole.

This study was interview-based, taking a phenomenologically oriented approach in which the respondents were asked about their experiences with software development risk management under the influence of software risk management, and their perceptions and perspectives of it. Fitzgerald and Howcroft (1998:322) assert 'while there may be paradigm incommensurability at the overall ontological and epistemology levels, some plurist ecumenical accommodation is possible at the lower methodological level and indeed even at the axiological level.' Therefore 'epistemological monism can coexist with methodological plurism' (Fitzgerald and Howcroft,1998:322). For instance, this study contains both qualitative and quantitative methods: a quantitative survey and an unstructured interview as alluded to earlier. In the information systems domain, qualitative and quantitative methods are viewed as polar opposites, but an integration forms a richer picture and possibly strengthens findings through triangulation (Fitzgerald and Howcroft,1998:322) which is the cross-validation achieved when different kinds and sources of data converge and are found to be congruent (Kaplan and Duchon,1988).

5.5 Summarizing Remarks on Interpretive Research

This research falls within the area of Software Engineering. It deals with systems development and hence can be seen also on the border of Computer Science and Information Systems. The approaches used in this research reflect rather the closeness of Software Engineering and Information Systems to the social sciences because of their involvement of the human element. Interpretivism asserts that the positivist methodology of natural science is inadequate for understanding human action (Doolin,1998:302). Interpretive information systems research is an acknowledgment of the fact that the study of information systems is not a pure science. Software development is not just an applied science either, as it involves the creation of artefacts which are created by human beings. Information systems is a human artefact which draws meaning by interacting with human participants involved with the technological aspects of the system (Doolin,1998:302).

This research is not intended to prove or disprove any hypotheses and therefore is not restricted by that notion. Moreover, there is a need to bridge the gap between the realities of risk management and what researchers think is appropriate. Understanding work practices is assigned increasing importance because the field of information systems has traditionally been plagued by high implementation costs. The high failure rates are frequently attributed to developers' poor understanding of work practices (Markus and Lee,1999). In order to develop a better understanding of risk management, it is necessary to study how efficiently an organization deals with risks, with or without a systematic risk management framework. Ropponen and Lyytinen's (1997:46) research confirmed that the risk management strategy was not the overriding influence in controlling risk. Gemmer (1997) noted that risk management is more than just a process, it has more to do with perspectives and perceptions surrounding reduction of risk.

This research does not focus on the specifics of risk management such as tools or techniques, because the literature on the subject indicates that risk management is more than just using the appropriate tools and techniques. This research focuses on the cultural aspects that foster positive attitudes towards risk. As these attitudes are perceived as social constructions and not technologically manifested, an interpretive stance is more appropriate for this study.

Using interpretive research in risk management can help to uncover the following aspects:

- Different perspectives on risk management, e.g. those who use it and those who do not.
- Examining the possibility that risk management itself may not be the critical factor in reducing risks, but rather the culture in which risk management fosters, i.e. a risk-awareness culture.
- Understanding how risk management fits into software development in terms of the day-to-day organizational activities of the practitioner.

Interpretive research allows the researcher to look beyond, and gain insights from the people who practice software development in reality. Here the researcher is not bound to prove anything and therefore the aim is to look beyond just the benefits of risk management by exploring the possibilities of gaining insights from those who experience it every day. Interpretive research can answer questions such as: How do practitioners use risk management ? What is their intention in applying risk management? and How do they perceive the usefulness of such a process? It is not merely a process of finding out if risk management is successful and then aggregating the results into a statistic. For example, saying that 80% of practitioners do not use risk management, is a useful statistic that can be uncovered by positivist research, but this statistic does not give any indication of why this is so, and therefore the inference may be drawn that risk management does not work in practice. However, interpretive research can do more than infer; it gives insight into why risk management is sometimes perceived as not working and the answer might be that many practitioners are not familiar with it.

5.6 Derivation of the Framework for Risk Management Analysis

A review of similar studies indicated that research that correlated strongly with the goals of this investigation were Ropponen and Lyytinen''s (1997) paper entitled "Can software risk management improve system development: an exploratory study" and Kontio et al (1998) case study entitled "Experiences in improving Risk Management Processes using the concepts of the Riskit Method." The limitations of each study were compensated for by the strengths of the other. Therefore, it was necessary to include selective questions from both studies. The study conducted by Lyytinen and Ropponen (1997) was not context-specific but it included a quantitative survey which used to objectively assess the effectiveness of a risk management programme. The qualitative interview template provided by Kontio et al (1998) considered the contextual and historical factors but was unable to objectively determine the effectiveness of using a risk management programme. This aspect will be elaborated upon further in the subsequent section. The next issue of consideration is the nature of inquiry. Ropponen and Lyytinen's study involved surveying 83 project managers, while the data collection method used by Kontio et al (1998) was a case study approach which involved observing and interviewing two companies.

This research adopts an interpretivist epistemology but utilises both quantitative and qualitative forms of data collection. The survey's intention is not to aggregate answers and determine any causal laws but to maintain a high level of objectivity in documenting the data and thereby eliminating the "observer bias" so common in case studies. However, the quantitative method has been criticised for ignoring the social aspect and therefore the qualitative survey compensates for this shortfall. Table 5.3 depicts the differences between survey interviews conducted using a quantitative methodology and field interviews conducted using a qualitative methodology.

The proposed framework is illustrated in Figure 5.1. The forward arrow represents the process of determining the risk performance measure from the quantitative survey. It was then compared against the managers' perceptions and perspectives surrounding the usefulness of applying a risk management strategy. The reverse arrow represents how the risk management methodologies affect the risk performance rate. This notion was used for the purposes of triangulation. Findings can be strengthened through triangulation which is the cross validation achieved when different kinds and sources of data converge and are found congruent (Kaplan and Duchon,1988:575).

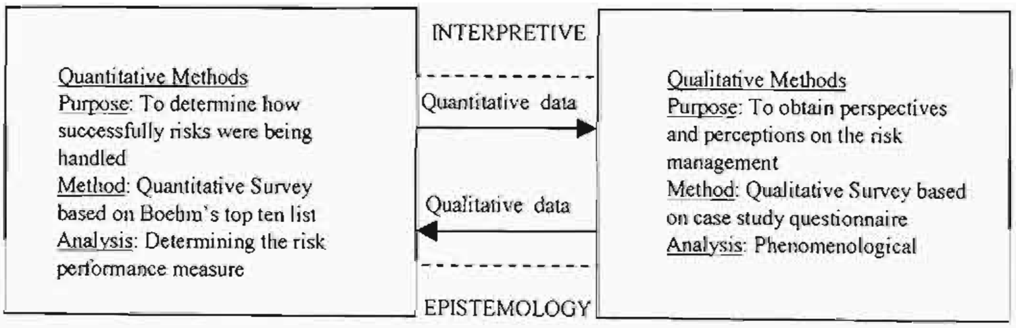
INTERPRETIVE
Quantitative Methods
Purpose: To determine how successfully risks were being handled
Method: Quantitative Survey based on Boehm's top ten list
Analysis: Determining the risk performance measure
Quantitative data
Qualitative data
Qualitative Methods
Purpose: To obtain perspectives and perceptions on the risk management
Method: Qualitative Survey based on case study questionnaire
Analysis: Phenomenological
EPISTEMOLOGY

Figure 5.1: An interpretive framework for investigating risk management practices.

Table 5 .3: Further details on the differences between quantitative surveys and qualitative surveys (adapted from Neuman (1997))

Typical Survey Interview	**Typical Field Interview**
It has a clear beginning and end.	The beginning and end are not clear. The interview can be picked up later.
The same standard questions are asked of all respondents in the same sequence.	The questions and the order in which they are asked are tailored to specific people and situations.
The interviewer appears neutral at all times.	The interviewer shows interest in responses.
The interviewer asks questions and the respondent answers.	A friendly conversation exchange, but with more interviewer questions.
It is almost always one respondent alone.	It can occur in group settings.
It has a professional tone.	It is interspersed with asides.
Diversions are ignored.	Encourages elaboration.
Closed-ended questions are common.	Open-ended questions are common.
Probes are rare.	Probes are frequent.
The interviewer alone controls the pace and direction of the interview.	The interviewer and member jointly control the pace and direction of the interview.
The social context in which the interview occurs is ignored and assumed to make little difference.	The social context of the interview is noted and seen as important for interpreting the meaning of responses.
The interviewer attempts to mould the communication into a standard framework.	The interviewer adjusts to the member's norms and language.

Obtaining qualitative data involved performing phenomenological analysis, which is method of extracting the essences from a narrative. Initially, the aim is to end up with a list of phrases that are relevant to the experience and abstracting it into themes. Thereafter a narrative is created using only themes and essences to form a coherent picture of the actual experience. Then the next step is to look for explanations to account for that particular experience. And finally combining the two narratives to form a complete picture of the situation. This method helps to reconcile and account for the value obtained from the risk performance measure. For example if the risk performance measure was low, the analysis involving accounting for the experience can be used to determine 'why' the value is low.

The risk performance measure elicited by the quantitative survey, cannot account for experience and the contextual factors, therefore the qualitative survey elicited these facts. These contextual factors, in combination with the phenomenological analysis of the experience formed a richer picture of the experience of the phenomena of software development risks. The qualitative data, was hearsay, and therefore had to be measured by some objective mechanism, and the mechanism created by Ropponen and Lyytinen (1997) best served that purpose. It was a means of verifying the account. For instance, if a participant indicated he did not need risk management as his company successfully deals with risks. If the performance rate was low, then that would indicate that his assessment of the situation was not accurate.

The qualitative and quantitative approaches are used in a complementary manner. The instruments of both studies correlated strongly with the goals of this study. The discussion commences with the justifications for the qualitative interview template, followed by the justification for the quantitative survey. The qualitative interview-based survey was based on the case study reported in the paper by Kontio et al (1998). This study was done to determine the effectiveness of the RiskIt method. The framework for the RiskIt method is similar to other risk management frameworks. Therefore the interview template was used for this study as well. This interview covered many issues, but the questions extracted were those that were particularly focussed on the contextual issues, risk management infrastructure and the perspectives of risk management (see Appendix A). This study would have sufficed as it covered the goals of this study but the issue of validity was problematic as is the case in any interpretive study. Owing to the subjectivity of the nature of the inquiry, it was obvious that the project managers would justify the use or non-use of risk management and thus be inclined to endorse their approach as successful in dealing with risks. It was evident that there was a need for an objective measure to determine how successfully managers were dealing with risks under their particular perception of risk management.

Ropponen and Lyytinen (1997) sought to measure the success of managing typical risk items that normally represent major pitfalls in software measurement. The result is the quantitative survey in Appendix A, which is a list of twenty questions based on Boehm's top ten list (Boehm,1991:35). In this survey, project managers were asked how often they experienced a particular event caused by a risk coming to fruition. The purpose of using this survey, was to determine how well the respondent was coping with risks. Ropponen and Lyytinen's (1997) study also correlated with the goals of this study. However, this study was highly positivistic in nature and was not context-specific. The other limitation, which is common with positivistic studies, was those aspects outside the boundaries of the research problem were ignored. For instance, in this study the political risks that were emphasized by some project managers were ignored. It was these particular limitations which provided the rationale for including the interview-based qualitative questionnaire. In this study, the participant was allowed to add any insight deemed necessary to the discussion.

The qualitative survey was used to extract the contextual factors, such as size, nature of the organization and the personal history of the project manager, as well as the perceptions and perspectives surrounding risk management. Thereafter the discourse depended on the level of risk management practiced, ranging from ignoring risks to implicitly dealing with risks, to formalized structured processes to deal with risks. The interview emphasizes the cultural aspects such as attitudes to risk management and what types of attitudes foster positive or negative risk behaviour. The purpose of the qualitative survey is to determine how project managers "perceive" software development risks under the influence of risk management. The subgoals involve determining:

- The factors that result in successful risk management practices.
- The factors that hinder successful risk management.
- The factors that result in "risk-aversive attitudes".
- The factors that foster risk-aware attitudes.

The factors fall into four basic categories, that is, external environments, resources, people's attitudes and technical factors such as measurements and metrics. As a qualitative survey was used to gather this information, the questions had to be recontextualized to coincide with the participant's current practices. For example, if a participant did not practice risk management, the discussion focussed around the factors that lead the participant to negate risk management. The quantitative part of the survey was used to corroborate the qualitative study. For instance, the qualitative survey will ascertain the justification and effectiveness for the use or non-use of risk management, while the quantitative survey will show what effect the use or non-use of risk management has on managing risks. The quantitative survey deliberately does not contain any direct questions about risk management as then participants will structure their answers to justify the use or non-use of risk management. The respondents were asked to complete this questionnaire, without knowing the intentions of the survey. This kind of mixture of quantitative and qualitative methods will serve to broaden the understanding of software development processes.

5.7 Selection of Participants

Purposive sampling was used in the study to attain a global perspective of risk management in the South African information technology industry. Therefore, project managers in information technology were selected as the best possible people to give a broader perspective. Project managers are responsible for defining the structure of the software process, and the use or non-use of software risk management will be directly related to their perceptions of it. In quantitative research a larger sample gives more credence to the research. However, with qualitative research taking larger samples turns results in a qualitative analysis of the surveys into a quantitative analysis as this will be the only way to make "sense" of the data therefore negating the purpose of the exercise (Moore,1999). This study initially opted for ten companies to perform the survey, which is more than the usual number for a qualitative survey but far less than a quantitative survey. Owing to the unavailability of participants, the number of those interviewed was later reduced to seven.

5.8 Instruments for the Study

This study employed two data-gathering techniques, a quantitative survey instrument and an interview survey instead of participant observation, because thoughts, feelings and opinions cannot be observed directly. The interview approach allows for convergence while still allowing individual perspectives to emerge. Each interview was tape recorded and transcribed for analysis.

No particular type of organization was chosen. The only precedent used was that respondents have to be more than just involved in information technology. They must be involved in the creation of new software. The only criterion imposed on the respondents was that they must have experienced the phenomena of software development risks.

5.9 Data Analysis

Inductive data analysis was employed, which involved scanning the qualitative data for categories and relationships among these categories, using transcendental phenomenological analysis. The quantitative data was used to calculate a risk performance measure for each participant which determined how well risks were being dealt with.

5.10 Validity of the Instrument

Both the qualitative survey and quantitative survey had been fully validated as they have been used in previously published research by Ropponen and Lyytinen (1997) and Kontio et al (1998). The validity of interpretive research is largely based on the 'acceptance of the scientific community' (Lacity and Janson,1994:149), where knowledge is validated not by forming arbitrary distinctions between theory and data but by appeal to logical consistency, and agreement with the interpretations of participants (Stone,1990). Validity in an interpretive approach is achieved through the process of triangulation. In this study the quantitative survey serves to corroborate the qualitative data. According to Checkland (1995) the validity of a soft approach can be justified if it serves the purpose of organizational learning. One can conclude that the proposed framework aims exactly at enhancing organizational learning on how to improve software development risk management.

Gathering qualitative data is a difficult task, as the researcher must adapt to changing circumstances. An interpretive approach acknowledges that the researchers have biases and subjectivity is intrinsically linked to the conduction of the study, i.e. the researcher invariably influences the study by focusing on certain key areas. However, the qualitative data is validated by the researcher repeating an "interpretation" of respondents comments, in key areas, so that with immediate feedback, the respondent can validate the researcher's interpretation. Insights are validated with rich descriptions and direct quotes (Lacity and Janson,1994:152).

5.11 Limitations of the Framework

This researcher's collection of quantitative data was as objective as possible but explanations were given by the researcher for questions deemed unclear by the participant. However, the collection of qualitative data proved difficult, even though a set of questions was available, these questions were sometimes not necessary or appropriate to the context. Therefore questions had to be constructed and restructured at the moment of questioning to enable the researcher to ask questions relevant to the context. This interpretation of the context was subjective, as bias cannot be eliminated in qualitative analysis because the qualitative data-gathering process is unstructured, and the participants were allowed to provide as much information as necessary to answer a particular question. As a result the researcher focused on "salient" issues and probed more deeply in that area, to the exclusion of other points of query deemed less "salient" or inappropriate.

5.12 Summary

Using qualitative research to supplement quantitative research is useful to broaden the understanding of information systems (Lacity and Janson,1994:152). A framework was developed in support of this view. The framework was used to investigate project managers' perspectives of the effectiveness or ineffectiveness of risk management methodologies. The auxiliary outcome from this, is to posit the framework as a tool, for project managers to evaluate current methods of dealing with risks. A qualitative methodology has been chosen in order to provide rich, descriptive data regarding the project manager's viewpoints of the phenomena of software risks and a quantitative methodology was used to corroborate the descriptive data. The inquiry was carried out within the natural setting and data collection relied on interviewing. The goal was to capture and understand the project managers' perspective through their eyes and to use this knowledge for feedback into the risk management strategy.

It is important that the researcher does his or her own transcriptions as this facilitates the process of immersion, allowing the researcher to pick up on hesitations in answering questions, as this can indicate a lack of understanding or ignorance. The process of phenomenological research involves many stages, and the process although long, is necessary. It is essential to go through all revisions even though the last step of combining textural and structural texts appears superfluous. The process of conducting revisions is necessary in order to extract the essences, as this process is not as intuitive as one might think.

CHAPTER 6

ANALYSIS OF THE APPLICATION OF THE FRAMEWORK FOR INVESTIGATING RISK MANAGEMENT PRACTICES

'Better understanding of the conceptual lenses with which project managers approach software projects - and the biases that tint those lenses- may help us isolate and avoid behavior that reduces management effectiveness while promoting behavior that increases it'(Moynihan, 1997:41).

6.1 Introduction

As an experimental validation of the framework, it was applied to an exploratory investigation of risk management practices in several software organizations to determine the perspectives and perceptions of risk management. The auxiliary outcome of this research is to posit this framework as a mechanism for project managers for determining the effectiveness of risk management in field conditions, for determining the problems with risk management in the context of an organization so that they can be identified and addressed. This research began with fulfilling the preliminary aim first, as once these interviews were completed, it was deduced in chapter Seven, whether the framework can comply with its auxiliary requirement . This was obtained by qualitative interviews, and the participants shared information on the nature of their organization, in terms of values, culture and methodologies used to facilitate risk management. Whether or not the risks were being successfully dealt with was determined by using a quantitative Likert scale survey. The idea behind the use of two data sources is that the quantitative scale will show how successfully risks are being managed, and the qualitative research will show how the nature of the organization influences this success rate. The comparisons of quantitative data and qualitative data were analyzed in the same context rather than aggregating the data as in the study conducted by Ropponen and Lyytinen (2000).

This chapter focusses on the application of the framework to determine how project managers "perceive" the effectiveness of software risk management. The companies interviewed provided software solutions to all sectors of society be they medical information systems, geographical information systems, telecommunication systems, defence systems or business information systems. The companies ranged in size from eight to fifty people. As stated in chapter Five, no particular criterion was used in the selection except that the managers must have experienced the phenomenon of software development risks.

6.2 Determining the Risk Performance Measure

The analysis used here, involved calculating the risk performance measure for each participant. This was done by "summing" up the responses (see Appendix C). The risk performance rate determines how well the participant manages risk overall. For example, company A successfully manages 90% of all risks encountered. As this is an interpretive approach, which involves considering the context of the situation, the qualitative data is analysed first and then compared against the quantitative data. The quantitative data will only be looked at with respect to the qualitative data.

6.3 Phenomenological Analysis of the Qualitative Data

Even though some companies do not have formal risk management strategies, they can all contribute to understanding dealing with risks. In essence every software project leader deals with risk explicitly or implicitly. The phenomena of software development risks are experienced by anyone in software development. The basis of this research is to gain an understanding of how this experience with software risks was enhanced by explicit or implicit risk management strategies.

The analysis began with the qualitative data. According to Moustakas (1995) in phenomenological research the procedure is as follows:

1. Listing and Preliminary Grouping

   List every expression relevant to the experience ("Horizonalization")
2. Reduction and Elimination: To determine the invariant constituents: Test every expression for two requirements:

a) Does it contain a moment of the experience that is a necessary and sufficient constituent for understanding it?

b) Is it possible to abstract and label it? If so, it is a horizon of experience. Expressions not meeting the above requirements are eliminated. Overlapping repetitive, and vague expressions are also eliminated or presented in more exact descriptive terms. The horizons that remain are the invariant constituents of the experience.

3. Clustering and Thematizing the invariant constituents: Cluster the invariant constituents of the experience that are related into a thematic label. The clustered and labelled constituents are the core themes of the experience.

4. Final Identification of the invariant constituents and themes by application: Check the validity of invariant constituents and their accompanying theme against the complete record of the research participant:

   (a) Are they expressed explicitly in the complete transcription?

   (b) Are they compatible if not explicitly expressed?

   (c) If they are not explicit or compatible, they are not relevant to the experience and should be deleted.

5. Using the relevant, validated invariant constituents and themes, construct for each participant an individual textural description of the experience.

6. Construct for each participant an individual structural description of the experience based on the individual textural description and imaginative variation. The individual structural descriptions provide a vivid account of the underlying dynamics of the experience, the themes and qualities that account for the "how" feelings and thoughts. In this case what conditions necessitate or negate the use of risk management ?

7. Construct for each participant a textural-structural description of the meanings and essences of the experience, incorporating the invariant constituents and themes.

8. From the individual textural-structural descriptions, develop a composite description of the meanings and essences of the experience representing the group as a whole.

6.3.1 Uncovering Themes in Transcriptions

From steps one to four the following 56 invariant constituents were identified in step one and clustered into seven themes:

THEME 1: IMPEDIMENTS TO SUCCESSFUL RISK MANAGEMENT

This theme addresses the problems experienced by participants that either result in them negating the need for risk management or having difficulty in implementing it. The following horizons were identified:

(a) Seen as being implied in the planning phase/specification.

(b) Difficulty in identifying risk.

(c) Difficulty in quantifying risk.

(d) Tertiary level risk management not adequate to apply in reality.

(e) Viewing risk as a challenge.

(f) Cannot visualise the process in reality.

(g) Small companies do not see the need for formal processes.

(h) Formal processes stifle creativity.

(i) Risk management caters specifically for generic risk concerns.

(j) Project managers view themselves as being solely responsible for reducing risk.

(k) Ignorance.

(l) Not enough people trained in academic institutions for risk management.

(m) Poor coordination between official bodies and companies.

(n) Poor platforms for risk communication.

(o) No training in management; more emphasis on technical training.

THEME 2: PROMOTION OF SUCCESS RISK ATTITUDES

This theme identified ways in which cultural attitudes can be fostered to produce behavioural patterns conducive to identifying risks. The horizons identified are:

(a) Non-individualist expectations.

(b) Take all perspectives.

(c) Non-personal management style.

(d) Become client-centric (by promoting the attitude that risk identification is about adding value to the client).

(e) Congregation, with users and managers to facilitate the software development process.

(f) Learn from past mistakes.

(g) Help individuals overcome their shortfalls.

(h) Optimize the cost to the customer.

THEME 3: PERCEPTION

This theme addresses participants' perception of risk management in terms of awareness, appreciation, cognizance and misconceptions. The following horizons were identified:

(a) Considering risk management as inherent in project management.

(b) Equating risk management to risk avoidance.

(c) Equating risk management to crisis management.

(d) Equating risk management to problem solving.

(e) Risk management is a part of standards.

(f) The risk management awareness is growing.

(g) Aware of risk management from a tertiary level perspective.

(h) Risk Management is about looking backward (learning from past mistakes).

THEME 4: PERSPECTIVE

This theme identified the participants' opinion of risk management, in terms of its usefulness in application or the justification of its inapplicability to their situation. The horizons identified were as follows:

(a) Using formal techniques cannot account for experience.

(b) Risk management is a new concept.

(c) It is an old concept under a new name.

(d) Risk Management pushes the process forward.

(e) A lot of risk can be reduced by merely understanding the client's requirement.

(f) Risk management keeps the process on track.

(g) Avoiding any high-risk projects in the first instance.

(h) It is about having control rather than trusting employees to act appropriately.

THEME 5: RISK MANAGEMENT IMPROVEMENTS

This theme identified ways in which risk management frameworks can be readdressed so that the following improvements can be made:

(a) Risk Management should take between two per cent and ten per cent of resources.

(b) The process of adding extra resources to buffer the effect of risks.

(c) Anchor it within the company policy.

(d) Give every employee a job description that includes reactions in a crisis situation.

(e) The risk management must be a balanced structure (where it is not too rigid).

(f) Reduce risks dramatically by understanding requirements.

THEME 6: IDENTIFICATION OF RISKS

This theme looked at how participants identified risks:

(a) Identify risks in planning only.

(b) Brainstorming type of session.

(c) Use questionnaires.

(d) Checklists.

(e) Considering the resources needed will identify areas of risk.

(f) Logically identify risks.

THEME 7: ANALYSIS OF RISKS

This theme looked at how participants analyzed risks:

(a) Rank it according to processes.

(b) Automated.

(c) External evaluation of risks.

(d) Quantify risks logically.

(e) Judge through experience.

Most of the themes identified, confirmed the literary review concerning the misconceptions of risk management and the negative attitude towards it. The only unexpected response was that partial blame was placed on tertiary institutions for not providing adequate training in risk management to alleviate the misconceptions and negative attitudes.

6.3.2 Constructing the Textural, Structural and Textural-Structural Descriptions

For each company an individual textural description of the experience was constructed using validated invariant constituents and themes. A textural description merely gives a paraphrased account of the experience. Thereafter structural description of the experience based on the individual textural description was constructed to account for experience, where the nature of the organization and size, together with the perceptions and perspectives, was taken into account. The final step of this phase was constructing a textural-structural description for each participant of the meanings and essences of the experience, incorporating the invariant constituents and themes.

Textural Description of Company A

This company specialises in developing software for medical professionals. The manager has 30 years experience. They do not address risk explicitly but do make contingency plans for any possible problems, by building 'in some fat into each and every field of projects' . This "fat" is the result of experience and statistics from past projects. Risk identification is tantamount to specification, 'normally when we start with a new project, we evaluate what would be required as far as the resources are concerned'.

Risk-aware attitudes are fostered by following a three-step process, 'the first question I want to know is: How can we repair the damage ? Then the second one is: How can we prevent it from happening in the future ? The third is: Who caused it and if it was you who caused it, how can we help you overcome it so that this thing won't happen in the future?'

This project manager believes that risk management comes out of looking backward, 'Risk Management comes as something that you've learnt in the past.' He does not see the need for learning about formal processes because 'if you go with book learning there are some things that you do forget which you've learnt, if you go with experience then normally one remembers' what they have learnt. He thinks 'risk management takes a new concept'. He does not deny the need for risk management and indicates that he is willing to accommodate it with one proviso,' I can accept one per cent, I can accept two per cent; when it gets to ten per cent that's hitting my pocket.' He also feels that the concept of risk management is actually 'not something new; maybe somewhere in the past it was under another name.'

Structural Description of Company A

This manager believes that he is practising risk management. He explains: 'not something new maybe somewhere in the past it was under another name.' He believes that "book learning", following formal procedure can only go so far, but it is experience that counts. This company specialises in only one type of software, which could account for the fact that he has extensive experience in that area. He is very much aware of what could happen and he probably knows in advance what the client's needs are because all his clients are from the same field.

Textural-Structural Description of Company A

This manager believes that risk management is inherent in project management and that he is probably practising it anyway. He does not subscribe to formal procedures advocated by standards and bodies because he believes that experience is more important. This company specialises in one type of software, which could indicate why his experience can be more useful than actually applying new strategies.

Company B Textural Description

This manager has one year of experience in project management in the commercial sector and has very little experience in risk management. His perception of risk management is correlated with crisis management. He views risks as challenges and therefore 'there must be solutions for every one of those problems'. His perspective of risk management was that it was unnecessary for a small company of eight people. His experience of dealing with risk is that he 'generally kills the fires'. They identify risks through brainstorming and questionnaires sent to the users. Identification of risks done in the planning phase are analyzed and ranked according to "processes". In terms of learning from past mistakes, he does 'analyze but not capture' data. He believes that risk management can be improved by bringing users and managers together.

Company B Structural Description

This manager obviously does not understand the nature of risk management as he considers it to be crisis management. He says 'risk management comes in where you have a problem where the system has been already implemented'. He believes that risks occur due to inadequate planning. He has no contingency plans for the risks that are analysed and identified as he says that he is the one who is generally putting out the fires. Functional type of risks are viewed as most important, since he indicted that risks were ranked according to processes. His lack of enthusiasm for risk management is explained by his perception that his company is too small.

Company B Textural-Structural Description

This is a small company and the manager does not apply risk management explicitly. The rationale for this is that the company is too small and whatever happens, they can deal with it collectively or he has the ability to put out the "fires". He equates risk management to crisis management. The initial claim that risks are being identified, and analysed is actually a misconception because he equates risks to problems.

Company C Textural Description

This manager has four years of experience in risk management. Risk management was considered from the inception. This company focuses on a complete business solution, looking at what the client needs, rather than what the client thinks he needs, and developing the software and training users. This company analyses the situation, and identifies "the critical areas in the company". Software risk management is considered to be a standard and is fully automated. It begins with using checklists, ranking elements qualitatively, then putting plans into place, while external observers make sure that the risk levels are being improved. A positive cultural attitude is fostered by making sure that no one feels as if they are being "attacked" and the ultimate aim should be adding value to the client. They have a living document, which enables them to learn from past mistakes. Overall the experience is that risk management is "actually pushing" the process "forward".

Company C Structural Description

This company has considered risk management from its inception, and considers it a standard. They have outside observers tracking the risk levels. Their software development strategy involves looking at what the client needs, not what he thinks he needs. Therefore, the focus is on a total business solution, from developing to training. This kind of holistic approach reduces requirements risk. This risk management strategy considers risks from a different angle, in that risks are considered to threaten the formulation of a complete business solution. For example, she feels that the client's own requirements can threaten this process, ('you analyse the given risk document') and telling the client that his processes are not working is a real problem.

Company C Textural-Structural Description

Risk management is seen as something vital, but the process is viewed from the point of view of formulating a true business process solution. Here the manager indicates that risk management is a positive process that drives the process forward.

Company D Textural Description

This company has a large staff complement of about fifty and is involved in major projects such as geographical information systems. This manager has thirteen years of experience in project management and five years in risk management. He perceives risk management as important: 'I think it is probably necessary' but he is not sure how to go about implementing it. He cites three problems: 'better awareness' and 'better formalised structure' and 'what we need is to find a structure keeping balance'. He says that he also has problems identifying and quantifying risk: ' it is case of finding the risk'; 'the play between money spent and the actual risk'; and 'I am not sure how one would quantify it'. He sees risk management as part of 'initial planning'.

Company D Structural Description

The manager initially indicated that he was involved in risk management for five years, but in the ensuing discussion, it appeared that very few of the principles of risk management were actually applied. He indicated that he felt risk management was important, but his company had not reached the level of identification. He considered risk management as identifying potential problems in "initial planning". He also indicated that he was finding it difficult to apply and rather apprehensively admitted that, 'Perhaps this is the wrong thing to say: I think I don't really see how one could apply it'. His articulation centred on, the difficulty in identifying risks, the money involved and the difficulty in quantifying risks. They seem to get around requirement risks by deciding what the company needs rather than what they think they need. He indicated that cultural awareness of risk centred around how people perceived the environment, if it is something where "stability" was important then risk awareness was more prevalent. His perspective on risk management was that it is necessary, and his company was finding means of applying it by consulting an employee who had previous experience in risk management elsewhere, but the bottom line was the difficulty in the application.

Company D Textural-Structural Description

This manager sees the need for risk management but cannot see how it would be applied. He is currently using an employee who is skilled in risk management to jumpstart the process. He indicates the need for a "balanced" structure.

Company E Textural Description

Company E focuses on Defence Systems Development and is a company with a staff complement of about forty. The manager has been involved in software development for seven years, but in project management for two years. They have a risk-taking attitude towards risk, as they feel that they need to do those types of projects to get a competitive edge. His perception of risk management comes from his tertiary level education, but indicates that his familiarity with it is ' not to the extent that I can apply it in' reality. He also indicates that risks such as personnel, deadline and schedule risk are generic risk concerns, and should be taken into account anyway. 'Deadlines affect every project, so we are aware of it. Depending on the project, we may cater for budget risks'. Therefore he believes that during planning all these risk concerns can be taken into account, 'When we plan we take into account, like the resources we will need and the personnel needed'. He does not follow a risk management strategy at all, and his rationale is, ' Well we are a small company. With IT, I believe in flexibility; we cannot have these rigid structures because creativity will be stifled. We have very flexible hours at work'.

Company E Structural Description

Company E does not follow a formal process and risk management is seen as being tantamount to planning for extra resources needed. He believes that risk management and any formal structures are not needed as they stifle creativity. They don't shy away from risky projects, as they believe these give them a competitive edge, but they don't actively deal with risk either. They know they are handling risky projects but they believe that this is inherent in the nature of the software projects they deal with and therefore they believe they cannot do much to reduce these risks.

Company E Textural-Structural Description

This manager does not deny that they deal with risky projects. In fact they seek them out but they believe that the risks are inherent and risk management would not do much to reduce this. In fact, risk management is seen as something that will stifle creativity.

Company F Textural Description

This company has a very complex structure, and the risk management is taken from a holistic point of view considering the 'cost management and risk assessment, risk management' and 'applying those portfolios in areas of industry that we are involved in '. Software development is just one aspect of the risk management process. Here the project manager was interviewed and not the software project manager but the former indicated that he also had experience of being a software project manager. This company manages big projects and has fifty or more software developers involved at any given time.

In this company all risks were identified "logically" and quantified "logically". But the project manager was responsible for making contingencies and determining the cost of those particular contingency plans: 'Let's say, we identify a risk to the program like the project manager does not have enough skill; then it is my obligation'.

The process of risk management is viewed positively: 'The value we add is billions'. One problem he cites is that 'there is a massive shortage in the market. The academic institutions train the people on cost accounting, financial accounting, marketing and strategic planning and project management, but they never teach people how to bring it all together and how to interpret what they need to do with it. The practical side of things is so wide it is difficult to teach the people what is happening out there. The biggest problem in risk management is that there is nothing out there that teaches the people ...to give the broader picture'.

Structural Description of Company F

Company F is a very big company with many aspects.. Software development is just one aspect and risk management is applied from a higher level. Here the project manager, with a strong cost management background manages this process. The emphasis is very much on cost issues and optimizing the cost to the customer. Risk management is seen as adding monetary value.

Textural-Structural Description of Company F

This company's infrastructure is enormous and the deals they make with clients involve billions of rands. Therefore risk management is not seen as just as something in software but rather as an over-arching principle over several areas. Because of the amount of money involved, the risk management strategies revolve around cost and management issues.

Textural Description of Company G

This manager has twenty-three years of international experience in software risk management. When asked about risk management he indicated that it is 'About the same. It is all combined with risk management'. This company delivers a full business solution with software, hardware, training, maintenance and backup in the commercial sector. This company considers risk before taking on projects. The manager considers whether the 'infrastructure (can) deliver the commitment that is resultant from that transaction or not' and ultimately he says that he "undermines" himself 'by not taking high risk as a principle'. He deals with risk by using his experience to determine 'what, course to steer'. He promotes a risk-aware culture by stipulating those attitudes in employees' job descriptions and but he admits that 'not all the contents of every single job description is not quite obviously definite' and therefore they supplement this with "weekly meetings" and " individual bilateral meetings with staff." The aim is to ' highlight any possible misunderstandings that can result in any risk over and above the authorisation given to that single individual member.' Risk management is 'anchored within the company's policy but he does not provide training for risk management but provides training 'more for technical aspects'. Retrospection is vital as it 'is the facts that are going to be of assistance during and after the activation of any programme that we take into our support lesson assessment. We base on its grounds any undertaking in the future.' The impact of risk management overall is very positive and makes the process "controllable". He abides by the philosophy that 'Trust is very in place but control is better'. He also indicates that the problem with risk management is that: 'What the official bodies or associations are undertaking out there, I think there is very poor coordination between the activities of various companies. I think our environment in South Africa in general is very poor in its platform of communication'.

Structural Description of Company G

This manager's concept differs quite considerably from the norm as he states ' I originate myself from an international background and that is how we have performed in multi-infrastructure, multi-social and multi-level markets.' He believes that risk management is about not taking risky projects in the first instance. When asked about his training in risk management, he said that he learnt about it through "international exposure"so this experience and management are not the traditional kind. His philosophy that 'trust is very in place but control is better' indicates that he doesn't place all control with his staff, but rather controls them. This philosophy is substantiated by his remark to the effect that by using his experience he determines 'what course to steer'.

Textural-Structural Description of Company G

This company practices risk avoidance rather than risk management. This assumption is corroborated by the following remarks: 'undermine myself by not taking high risk as a principle'; 'I am trying to safeguard my investment'; 'I try to eliminate them in my company'; 'prevent any situation that you cannot foresee'. So this strategy is risk-avoidance strategy.

The last step in this phenomenological research is to develop a composite description of the meanings and essences of the experience representing the group as a whole, from the individual textural-structural descriptions.

6.4 Composite Description of the Phenomena of Software Development Risks

Many project managers consider risk management as being implied or accounted for in the planning or specification phase. They do not see it as being distinct from planning. They tend to think of risks when allocating personnel or resources and not of risk as an ongoing process.

One manager in particular who was very interested in risk management, expressed difficulty in identifying risks and how one would go about giving a number to something so fuzzy. Others expressed their difficulties in visualising how such a process can work in reality. The other problem is with checklists: one company advocated the use of them but others considered them too generic.

Smaller companies seem less interested in risk management than larger ones. Smaller companies actively seek out risky projects and view risks as challenges rather than something to be reduced. They don't see the need for formal processes as they believe they can control crises and that formal processes stifle creativity. Many project managers in smaller companies tend to view themselves as solely responsible for reducing risk and therefore do not set up contingency plans for two reasons. Firstly, they believe they have the capacity to solve problems and should be able to handle anything should a crisis occur. Secondly, they believe that as the company is small, the crisis will not be beyond the manager's control.

Companies are simply not aware of risk management, and have only a vague idea of what it is. This could be due to academic institutions not catering sufficiently for risk management in their syllabus or due to poor coordination between official bodies and companies. One manager remarked that studying risk management at universities was not enough to apply in reality. The other reason for ignorance could be due the fact that there is often an emphasis on technical training rather than management training.

The next consideration in this study is to identify ways of promoting successful risk attitudes. Firstly, it is important not to overemphasise one particular individual's role in a project. Therefore it is important to have non-individualist expectations and to take all perspectives into account. Secondly, one should conduct meetings about risks in such a way that the risks do not become a personal issue. It might be useful to bring together different groups, such as users and managers, to gain more insight into risk issues. If an individual was responsible for a particular risk, then set mechanisms in place so that the problem is prevented from happening again by learning from past mistakes. Allocate tasks by recognising the strengths and weaknesses of individuals, and help individuals overcome their shortfalls. Become more customer-centric by optimizing the cost to the customer by promoting the view that finding risks is about adding value for the client

This study also considered how risk management can be assisted by learning. Risk management is actually about retrospection. Obviously learning from past mistakes is a matter of experience, but statistics from the past can also help. The realisation that every project is different should produce a living document that expands.

During the course of interviews, several perceptions and misconceptions surrounding risk management were identified. Some managers perceive risk management as a standard and feel that it must be given special attention, while others consider risk management to be inherent in project management. Of these there are two views of risk management, one is extremely formal while the other is ad hoc and informal. These views are not polar opposites but just different interpretations of how risk management should be applied. However, the following views are definite misconceptions:

- Equating risk management to risk-avoidance.
- Equating risk management to crisis management.
- Equating risk management to problem solving.

It is obvious that those who apply risk management consider it to be a very positive experience. Their positive perspectives indicated the following improvements in their management process:

- Drives the process forward.
- Regaining control.
- Keeps the process on track.

However, many believe that using formal techniques cannot take the place of experience. Some believe that they are practising risk management and that it is just an old concept given a new lease of life. Generic risk affects every project, and circumventing such risk just involves common sense. A lot of risk can be reduced by understanding the client's requirements. One manager perceived risk management as a process in which reducing risks should involve avoiding any high-risk projects in the first instance.

Participants were also asked how risk management can be improved. Many answered that they did not know as they were not familiar with all the literature available. Others commented on the difficulty in applying management in reality. Risk Management ideally should take up between two per cent to ten per cent of resources since beyond this, managers are not willing to accommodate it. Promoting risk-aware cultures can be accomplished by adding it in their job descriptions or making it policy.

There are several degrees of risk management practices. These range from automated processes, to using brainstorming sessions, to considering risk as an inherent element during planning. Much of what is done in the literature regarding risk management is not practiced in its entirety. While one manager used checklists and ranked items qualitatively, others identified and quantified risks logically or through experience in brainstorming sessions. Others considered risk during planning.

Checklists that address generic risks concerns are not very useful. What might be useful is advocating the use of "buffers" to address these concerns, for example not expecting that a hundred per cent of all personnel can address personnel risks. Schedule risks can also be reduced in a similar manner. Requirements risks can be reduced, not by getting the requirements right, but by making the requirements right i.e. looking at the total picture and not just giving the client what he asked for but rather forming a total business solution. If there are problem areas in the current implementation then the software should not carry the same problems because later on this system will have to be maintained to make up for that initial shortfall. Therefore software should not mirror reality but should actually make reality better by offering a credible business solution.

6.5 Comparing the Textural-Structural Data Against the Risk Performance Measure

The risk performance measure determined how well a company was managing the twenty risk items in the survey. The risk performance measure was then compared against the experience(see Table 6.1). The nature of the organization and the size of the organization were taken into account where the size was classified as follows:

- Below 25 was considered small.
- Between 25 and 40 was considered as a medium-sized enterprise.
- Above 40 was considered large.

As this research is done in an interpretive paradigm, it is important to be cautious about any correlations or generalisations made and not to extrapolate this. The comments at this juncture are just conjecture.

Table 6.1: Showing the risk performance measure against qualitative data

Company	%	Risk Management Style	Years of Experience	Nature of Applications	Size of Organization
A	90	Contingency Management	30	Medical Applications only	medium
B	58	Crisis Management	1	Various Commercial Applications	small
C	70	Total risk management from a business solution perspective	4	Various Commercial Applications	medium
D	73	No risk management	13	Engineering and Commercial Applications	large
E	58	No risk management	2	Defence Systems and Commercial Applications	medium
F	63	Risk Management not specifically software risk management	3	Telecommunication Applications	large
G	68	Risk Avoidance	23	Various Commercial Applications	medium

Company A does extremely well in alleviating risks, and this could be due to experience and the nature of the applications which are all in the medical information systems field. While companies E and F indicated that they had no need for risk management owing to their size, they had the lowest ratings. This could be due to the inexperience of the managers. Company C and company E do perform explicit risk management, but had lower ratings which could be due to the inexperience of the project manager and the complexity of the applications involved respectively. Company G performs risk avoidance and therefore does not cater for risks when they do occur, which could account for the mediocre rating. This final risk performance measure did not show how some contextual factors might influence particular risk categories, like schedule type risks and requirements risks. The next stage was to classify the 20 risk items into categories.

In a subsequent study Ropponen and Lyytinen (2000) identified six components from the original survey (Ropponen and Lyytinen,1997) by using PCA (eigenvalue 1.0, VARIMAX-rotation with Kaiser-normalization) These were the components of risk:

- Scheduling and timing risks
- System functionality risks
- Subcontracting risks
- Requirements management risks
- Resource usage and performance risks
- Personnel management risks

The performance measure for each category was determined by summing up the scores in the factors related to the category for each participant(see Appendix C). Table 6.2 provides the factors from the quantitative questionnaire that correlated with each risk category.

Table 6.2: Classifying the twenty risk items into categories of risk(Adapted from Ropponen and Lyytinen, 2000)

Risks	Factors Involved
Systems functionality risks	Satisfaction with user interface Functions and properties correct Estimation of hardware and software capabilities Estimates of personnel needs
Scheduling risks	Problems in the timetable Actual costs vs estimated costs Changes in the timetable Wrong size estimates Managing complexity Estimates for personnel needs
Requirements management risks	Requirements changes Gold plating Steady consumption of time Changes in table
Subcontracting risks	Success in externally performed tasks Shortfalls in externally furnished components Estimates for personnel needs
Resources and performance risks	Resource usage and deadline Evaluation of performance requirements Managing project complexity Estimation of hardware and software
Personnel risks	Personnel shortfalls Unrealistic expectations of personnel's abilities Steady consumption of time Insufficient expertise Evaluation of performance requirements

Some factors are defined more than once in a particular component but that is simply a reflection of the fuzziness surrounding risk, and the fact that one risk may have an impact on another. One variable (project cancelling) was dropped from the final analysis since it did not contribute to any of the components but: 'Overall, the result is statistically acceptable and represents a conservative number of factors (risk dimensions)' (Ropponen and Lyytinen,2000). All the company's performance measures are determined for each risk category.

The risk categories were discussed in chapter Two, with the exception of the subcontracting risks, which is merely the risk involved when a company outsources parts of software development to other companies. Further clarification is warranted in terms of the difference between systems functionality and requirements risks. Here requirements risk is the risk of changing requirements, and systems functionality is the risk of not getting the requirements right. In chapter Two both these categories were labelled as requirements risk. After determining the performance measure for each risk category for each company, the results are shown in Table 6.3.

Table 6.3: Showing the performance measures for each risk category for each company

Risks to Successful Software Development	A	B	C	D	E	F	G
Schedule risk	97	47	73	80	47	63	67
System functionality	90	70	70	80	80	65	95
Subcontracted risks	87	40	73	73	40	73	67
Requirements management	85	55	65	55	50	55	40
Resources management	85	55	60	75	60	55	80
Personnel management	88	60	68	64	68	60	72

The lowest performance for companies A, D, G was requirements risk, which is coping with user changes. Companies C, B, E and F managed these risks at an extremely mediocre level between 60% to 50% of the time, thereby indicating that requirements risk is one of the most troublesome risks. This could be due to the fact that changing requirements is out of the manager's control. Even though this is a weak area for company A, it still manages to cope with it 85% of the time. This could be due to the fact that the project manager does not allow any changes once the requirements document is signed off. The other problem could be the nature of the environment, where a multitude of users have to be pleased, especially in distributed systems, which could explain the problem company E has with requirements risks. Ropponen and Lyytinen (2000) concluded that requirements risks can be reduced by a commitment to applying risk methods. The same conclusion could not be reached here. This could be due to the differing contexts and experience. Those companies whose project managers had less experience in project management, that is between one to four years, were companies B, C, E and F. Of those companies, C had the strongest commitment to risk, and performed better than company F, B and E. Company F applied risk management but not specifically software risk management. This relationship could imply that software risk management may be more beneficial to those project managers with less experience than those who are highly experienced in software development.

Much of companies' success in controlling risks comes from looking at past mistakes. Company A outperforms all other companies in terms of personnel risks which may be owing to the project manager ensuring that he helps personnel overcome past mistakes. Company G also performs very well, perhaps because its project manager does not place complete trust in his employees but rather puts control mechanisms in place in terms of job descriptions and policies. The other reason could be that company G does not tackle any project where there are high risks involved. Company F, which practices risk management, has one of the lower scores. This could be due to the instability and complexity of the environment, which is large-scale and involves distributing computing software development.

Ropponen and Lyytinen (2000) concluded that 'Scheduling and timing risks seem to decrease linearly as more experience in using risk management methods is gathered'. With scheduling and timing risks, there seems to be a correlation between the experience of the project manager and the management of scheduling and timing risks. Company A does extremely well which could be due to experience, and the "fat" that the manager adds onto every project, whereby he allocates half a day's work to each staff member to account for any personnel problems.

There is an apparent correlation between the complexity of the project managed and the experience of the project manager when it comes to the way in which a company deals with resources risk. For instance, Company F deals with highly complex and huge projects involving distributed environments, while Company A deals with a familiar environment. Company G avoids, high-risk projects, which implies that they avoid highly complex projects as well.

Company B and E are small to medium-size companies and have less experience with subcontracting and therefore they do not control these risks effectively. Smaller companies tend to subcontract less and found the questions relating to subcontracting irrelevant to their situation. In fact, Company E's manager indicated that the risk factor "Externally Purchased components and equipment meet your expectations" was not applicable to him, so he could not answer and that question was given a score of 0. Interestingly, Companies C and D managed subcontracting risks at the same level, which corresponds to Ropponen and Lyytinen's (2000) study which indicated that risk management had no effect on these risks. Company C practices risk management and Company D does not.

Company F deals with functionality risks very poorly, that is they have a problem with meeting requirements. This is probably related to the complex nature of applications in the telecommunications environment. Company A deals with a stable environment where all systems are medically based and familiar. Interestingly, Company G, which performed risk avoidance, controlled systems functionality risks the best, which could be due to the fact that he does not take on projects he feels he cannot manage. Ropponen and Lyytinen's (2000) study found that experience acts in the mitigation of functionality risk. This study concurs, but finds that the familiarity of the operating environment is also a factor.

6.6 Summary of Research Findings

It is important to note that this study is not intended to form any causal relationships and that each case must be viewed in context. This research can be helpful for other organizations to apply the proposed framework in order to develop some insight into their current software development risk management practices. This is useful when considering the current risk frameworks that are documented in the literature, and how they can be improved and made more accessible to the organizations that practice software development. Software risk management described in the literature is not congruent with software risk management in practice. Therefore research into creating new risk management frameworks is not going to transcend into practical reality unless researchers take into account current processes and contexts. The framework improved organizational learning because participants were unrestricted by the confines of positivism, was able to challenge the researcher on the validity of a risk management approach and these debates were accommodated and reciprocated by the researcher. This process dispelled the misconceptions surrounding risk management.

CHAPTER 7

CONCLUSION

'Software development's risky nature is easy enough to acknowledge in the abstract, but sadly, harder to acknowledge in real-world situations'(Lister,1997:18).

7.1 Achieving the Goals and Subgoals of this Research

The literature survey indicated that the crisis in software can be alleviated through the use of software risk management, i.e. the process of identifying, analysing, mitigating and tracking risks. In spite of all the positive outcomes expected to be gained from adopting a risk management strategy, software development still remains in a crisis situation. There could be two reasons for this. Firstly, the benefits of software risk management are not significant enough to avert the crisis or secondly, software risk management is not practiced at all. Research done into risk management indicated that, contrary to the former supposition, the benefits are indeed significant, but the crisis in software still remained. This disparity leads to the conclusion that the research paradigms used in conducting research in software risk management were not sufficiently effective to capture the rationales behind the failure to use risk management. This study was not based on hypotheses testing or forming any generalisations, but merely understanding the phenomena of software development risks within varying contexts.

The main goal of this research is to provide an interpretivist framework for determining the effectiveness of risk management in field conditions and to apply it to an exploratory investigation of risk management practices in several software organizations. The framework developed was tested experimentally in several companies. The application of the framework helped focus the manager's attention on software development risks. As was indicated in chapter One, it is important to make a distinction between software development risks and security risks. The questions in the quantitative survey were created in such a way, that the principles of software risk management were not represented. As Boehm(1991) observed, although a manager may use the principles of risk management, the terminology might be unfamiliar. Hence, the quantitative survey does not use any of the terminology related to risk management rather it reflects the problems that might occur if risks are properly identified, analysed and mitigated.

The qualitative survey was used to elicit participant feedback on the existing risk management process. As this was not structured, the software manager was free to elaborate on the problems with existing risk management techniques and their perceptions of it. The two differing approaches were put together here in a complementary manner to combine their strengths. This combination is based on two previously published papers which can be seen as a justification for its validity.

The framework can serve as a tool for exploratory analysis of risk management practices in a particular environment. During the course of the discussions, the project managers surveyed, became increasingly aware of the benefits of a risk management strategy and some even indicated their willingness to adopt such a method. It is light of these comments, that indicated, the framework can posit itself in several ways within organizations to improve their risk management strategies:

- It can be used as a way of testing the reliability or efficiency of a particular risk management strategy.
- It can be a means to alert software managers to the need for a better risk management strategy.
- It can be used as a mechanism to enable and improve a risk management process concentrating on the attributes that need to be addressed in terms of culture, policy, methods, tools, skills, competence, infrastructure, and documentation.
- It can be used to determine areas of weakness in terms of schedule risks, system functionality risks, subcontracted risks, requirements management risks, resource risks and personnel risks.
- It contains both structured and unstructured questions. The structured questions help focus the discussion on software development risks only. The unstructured interview allows

free flow discussions, which is a backdrop to effective risk communication, as articulated in chapter Two.

This framework elicits two types of data, quantitative and qualitative which can be easily and quickly administered. The calculations involved do not require any specific knowledge on risk analysis or statistical analysis. This analysis, does not endorse any particular risk management strategy, as it is important that each organization develops and tailors risk management strategies according to its context. It therefore also encourages a risk management strategy that goes beyond just a risk management process but a process that considers other issues that are often neglected by risk management frameworks. Issues such as culture and policy making and learning from past mistakes.

As an experimental validation of the framework, it was applied to an exploratory investigation of risk management practices in several software organizations to determine the perspectives and perceptions of risk management. A qualitative interview was conducted with a quantitative survey for the purposes of triangulation. Here the qualitative interview served to capture the perspectives and perceptions of risk management including:

- The factors that result in successful risk management practices.
- The factors that hinder successful risk management.
- The factors that result in "risk-aversive attitudes".
- The factors that foster risk-aware attitudes.

Chapter Two began with a discussion on the differing software process models. Three aspects were considered at this juncture: the process risks that would be encountered in adopting a particular model; the level of risk consideration given in the model; and how the model can be improved through incorporating risk analysis. Here the risk categorisation provided a mechanism for evaluating the risk concerns of software managers for the formulation of the framework.

Chapter Three elaborated on risk management frameworks and considered their phases specifically. The analysis provided the sources, that is Boehm's top ten list and the stages that typify a risk management process to ensure success that is risk identification, risk analysis, risk mitigation and risk monitoring. Boehm's top ten list is a prioritized checklist of ten software risk items, spanning over all development phases as well as the external environment and used a means to test the effectiveness of a particular risk management strategy. This risk identification strategy was compared against other checklists and was found be the most comprehensive. These formulations provided the basis for the quantitative and qualitative surveys.

Risk analysis is the core function of a risk management process and required further elaboration. Chapter Four was therefore included to provide an overview of analytical techniques for risk assessment in software development. The emergent theme in chapter Four, was the problems with current risk analysis techniques be they quantitative or qualitative, with regard to their inaccuracy and the difficulty experienced in executing them.

The analysis conducted in chapter Two, Three and Four, indicated that there is a need for a structure not only to support risk management but also a structure to foster risk-aware attitudes and to counteract dysfunctional risk behaviour as well. The culmination of chapter Four provided the motivation, for considering an interpretive epistemology because it became increasingly evident, that the process of successful risk management was more than the use of appropriate tools and techniques.

The research questions that were investigated through the qualitative study were conducted with seven participants. The following summary conclusions were found in relation to the research inquiry from the themes extracted from the transcripts.

Factors that resulted in software development risks being managed successfully, irrespective of using implicit or explicit risk management, were firstly the experience of project managers and secondly the nature of the application developed. For example, developing software in similar contexts made controlling risks easier and more predictable.

There are three factors that impede the successful practice of risk management: the misconceptions surrounding the application of risk management; the difficulties encountered when trying to implement it; and ignorance about risk management practices. The misconceptions arose through viewing risk management as being implicit in the planning or specification phase or viewing risk as challenges, therefore negating the need for risk management. The other misconception is that risk management caters specifically for generic risks and therefore cannot be tailored to specific risk concerns. Some of the problems in implementing risk management were the result of difficulties in identifying quantifying risk since these processes are difficult to translate into reality. Small companies do not see the need for formal processes as they feel this will stifle creativity. Tertiary-level education in terms of risk management is not enough to allow one to apply it in reality. Software project managers exacerbated this ignorance by not providing training in risk management and placing more emphasis on technical training.

The negative attitudes towards risk management arose out of the misconceptions alluded to earlier, such as:

- Equating risk management to risk avoidance.
- Equating risk management to crisis management.
- Equating risk management to problem solving.

The ability to identify risk without fear of recriminations is vital. Therefore it is important to promote risk-seeking attitudes to facilitate the risk identification process. The following factors were found to foster risk-aware attitudes:

- Promoting client-centric values as risk identification is about adding value for the client.
- Prioritise learning from past mistakes as this can help individuals overcome past shortfalls.
- Dealing with familiar operating environments improves the project manager's ability to identify and deal with risks. Under these circumstances software developers tend to rely more on intuition than formal processes.

The reason for risk management not being translated into practical reality lies in the myths surrounding it, and the only way this can be overcome is through tertiary institutions providing adequate programmes on risk management.

7.2 Comparisons of Results with Previously Published Research

In Ropponen and Lyytinen's (1997:46) study, it was concluded that the longer the experience with risk management methods, the better the project manager's performance in estimating the project size, and the less the chances of project delay. There is an indication that experience plays a significant role in uncovering risk, but this is not necessarily owing to experience in risk management as such. It is probably due to having an intuitive handle on what can go wrong and preparing for it. Therefore the author tends to favour one of the other findings that Ropponen and Lyytinen (1997:46) uncovered, that is, performance in managing risk seems to be a function of better managerial cognition, commitment and the use of a proactive management style, rather than a specific risk management technique.

Ropponen and Lyytinen (1997:46) found that the factors that influenced development process were size, project management experience and project management training. This study also revealed two other factors, which were the familiarity of the operating environment and the nature of the application. Ropponen and Lyytinen (1997:46) offered the solution of reducing the size of projects, training project managers and standardising components as a way of minimising risks. The author offers one more solution, which is developing systems for specific environments, thereby increasing the reuse of components, and increasing experience in particular applications. This view supports Jackson's (1998) idea that product-specific knowledge as practiced in other disciplines, should also be advocated in software engineering.

In the study by Kontio et al (1998) in relation to RiskIt, it was found that in order for successful risk management to take place:

- Risk management process must be supported and enforced.
- Risk management should start before the project starts.
- Stakeholders and goals play a critical role in risk management.
- A common risk management framework makes risk management more efficient.

It was difficult to form parallels with these conclusions owing to the nature of the investigation because here risk management was not practiced in its entirety. Only one company practiced risk management and in that particular case the finding was consistent with the above conclusions.

7.3 Areas for Possible Further Research

In this study only project managers were interviewed, but interviewing subordinates in order to obtain different perceptions of the same context would have been more insightful. Although project managers indicated that they did facilitate risk-aware cultures, perhaps a different picture would have emerged if subordinates were interviewed as well.

The role risk management plays in project management has to be considered as it seems that these two issues are intrinsically intertwined. It might be useful to consider what the best possible role is for risk management to play, i.e. should it be necessary to separate these two issues at all? According to Grey (1995) there are three differing views in this respect:

- Project management is composed of risk management i.e, that risk management is just a part of project management.
- Project management is a subclass of risk management that is, without risks there would not be a need for project management as it would just be an administrative task
- Risk management must be considered in all parts of project management.

This is an important consideration as a poor connection between risk management and general project management creates 'needless bureaucracy and wasted efforts' (Ropponen,1993)

Research needs to be conducted to determine the amount of time and resources that project management must allocate to risk management in order for projects to be successful. The correlation between the time spent identifying, analysing, mitigating and tracking and the success rates should be investigated. The relationship of time spent in identifying with time spent in analysing, mitigating and tracking should also be considered. Another important aspect is looking at the type of person in terms of status and personality who would be best suited to each of these tasks. This was alluded to in chapter Three. Areas such as effective risk communications increase the productivity of brainstorming sessions.

The possibility of combining security risk management and software risk management needs to be explored. Issues such as safety and security are requirements that must be met by software developers. Therefore, not meeting safety and security requirements can also be considered as requirements risks.

7.4 Concluding Remarks on Software Risk Management

Any generalisations indicated in this study must be treated with caution given the nature of this study and the number of participants. Any comments should be considered as observations rather than causal relations. The purpose of this study was to introduce a framework for investigating risk management practices in a particular software company with the aim of gaining a better understanding of the issues related to them and improving them rather than generalising results.

The participants did indicate that risk management was a positive step. However, they took two opposing views on whether or not it is warranted. One view was that risk management is project management and no actual distinction is made between the two. Project managers indicated that the principles of risk management were applied "unconsciously". The second factor was that risk management was inappropriate to their contexts and unnecessary for smaller companies. It seems that both views are indicative of the necessity for improving the current understanding of what risk management can contribute to organizations.

It cannot be concluded that risk management as practiced by the organizations concerned is ineffective, but rather there are many misconceptions and a lack of knowledge surrounding it. The ignorance factor is perpetuated by the lack of training both at tertiary level and at company level. Another possible reason, which was not uncovered by the research itself, could be that software project managers do not keep up with current trends that are documented in scientific journals. This factor corroborates the premise for this research expressed in chapter Five on the need to link practice and research, and the fact that software project managers do not utilise the methods developed under the positivist paradigm because it is seen as being inapplicable to reality and their contexts. The ignorance factor can be overcome by universities adopting a proactive approach to teaching risk management in a reality-based fashion, while the last factor can be influenced by journals facilitating research conducted in the interpretivist paradigm, as in the example of Management Information Systems Quarterly.

In Ropponen's (1993) study conducted in Finland, he found that several participants intended to begin applying risk management. In this study a similar scenario unfolded in that project managers had very little knowledge concerning risk management and expressed keen interest in the concept and felt that it could be a positive influence. However, software risk management can only be positive with the following stipulations: it must have a balanced structure; it must not be too rigid but at the same time it must have sufficient areas of commonalities to facilitate it; the costs of implementing it must not outweigh the cost of applying it; it is also vital that risk management does not take up too much time or resources. As with software process models, risk management suffers the same fate, under restrictions. If there is no time, or money, these processes are abandoned along the way. In the research conducted by Ropponen and Lyytinen (1997:44) it was discovered that spending too much time on risk management is also detrimental. The more structured risk management becomes, the more difficult it is to apply. Risk management tends to be applied intuitively by experienced managers. It is this inductive process that must be facilitated by fostering risk-aware cultures.

The interpretive framework developed served two different but complementary functions for this research. As an experimental validation of the framework, it was applied to an exploratory investigation of risk management practices in several software organizations in order to determine the perspectives and perceptions of risk management. The analysis of the discussions with the participating software managers regarding the interpretive framework adopted for the investigation of risk management practices demonstrated the frameworks' potential as a tool, for enhancing software development risk management.

REFERENCES

Abbot, B.(2001) Requirements Set the Mark., *InfoWorld* **23**(10):45-46

Abdel-Hamid, T.K.(1993) Software Project Control: An Experimental Investigation of Judgement and Fallible Information, *IEEE Transactions on Software Engineering* **19**(6):603-612

Adler, T. R., Leonard, J.G., and Nordgren, R.K.(1999) Improving risk management: moving from risk elimination to risk avoidance, *Information and software Technology* **41**(1): 29-34

Albanna, S.J. and Osterhaus, J.(1998) Meeting the Software Challenge, *Information Systems Management* **15**(10):7-15

Alter, S. and Ginzberg, M.(1978) Managing Uncertainty in MIS Implementation, *Sloan Management Review* **20**:23-41

Avison, D., Lau, F., Myers, M. and Nielsen, P.A.(1999) Action *Research,Communications of the ACM* **42**(1):94-97

Bach, J.(1997) Software Realities: The hard road from methods to practice, *IEEE Computer* **30**(2):129-130

Bach, J.(1999) Software Realities: Risk and Requirement-based testing, *IEEE Computer* **32**(6):113-114

Baker, S. ,Ponniah, D. and Smith, S.(1998) Techniques for analysis of risks in major projects, *Journal of the Operations Research Society* **49**: 567-572

Bandyopadhyay, K., Myktyn, P.P, and Myktyn, K.(1999) A framework for integrated risk management in information technology, *Management Decision* **37**(5):437-444

Banville, C. and Landry, M.(1989) Can the field of MIS be disciplined, *Communications of the ACM* **2**(1):48-61

Barki, H. and Rivard, S.(1993) Towards an Assessment of Software Development Risk, *Journal of Management Information Systems* **10**(2):203-225

Barlas, S. ,Lutz, M. and Pleeger, S.L.(1996) Anatomy of a Runaway: What Grounded the AAS, *IEEE Software* **13**(1):104-106

Baskerville, R. and Pries-Heje, J.(1999) Grounded action research: a method for understanding IT in practice, *Accounting Management and Information Technologies* **9**:1-23

Baskerville, R.L. and Stage, J.(1996) Controlling Prototype Development through Risk Analysis, *MIS Quarterly* **20**(4):481-503

Bell, T.E.(1989) The Limits of Risk Analysis, *IEEE Spectrum* **26**(6):51

Benbasat, I. and Zmud, R.W. (1999) Empirical Research in Information Systems: The Practice of Relevance, *MIS Quarterly* **23**(1):3-16

Bennatan, E.M.(1996) *On time, within budget: software project management practices and techniques* (2nd ed.). New York: Wiley

Bennett, J.C., Bohoris, G.A., Aspinwall, E.M. and Hall, R.C.(1996) Risk analysis techniques and their application to software development, *European Journal of Operational Research* **95**(3): 467 - 75

Beynon-Davies, P.(1997) Ethnography and information systems development: Ethnography of, for and within IS development, *Information and Software Technology* **39**:531-540

Biffl, S.(2000) Using Inspection Data for Defect Estimation, *IEEE Software* **17**(6):36-43

Blum, B.I.(1992) *Software engineering : a holistic view,* New York: Oxford University Press

Boehm, B.W.(1991) Software Risk Management: Principles and Practice, *IEEE Software* **8**(1): 32-41

Boehm, B.W. and DeMarco, T.(1997) Software Risk Management. IEEE Software, **14**(3): 7-19

Boehm, B.W. and Ross, R.(1989) Theory-W Software Project Management Principles and Examples, *IEEE Transactions on Software Engineering* **15**(7):902-16

Boehm, B.W.(1996) Anchoring the Software Process, *IEEE Software* **13**(4):73-82

Boehm, B.W.(2000a) Project Termination doesn't equal project Failure, *IEEE Computer* **33**(9):94-96

Boehm, B.W.(2000b)The art of expectations management, *IEEE Computer* **33**(1):122-124

Boehm, B.W.(2000c)Unifying Software Engineering and Systems Engineering, *IEEE Computer* **33**(3):114-116

Boehm, B.W., Egyed, A., Kwan, J., Port, D., Shah, A. and Madachy, R.(1998) Using the WinWin Spiral Model: A Case Study, *IEEE Computer* **31**(7):33-44

Boehm, B.W., Port, D., and Al-Said, M.(2000)Avoiding the Software Model-Clash Spider web, *IEEE Computer* **33**(9):94-96

Boland, J.(1985) Phenomenology: A preferred Approach to Research on Information Systems. In E. Mumford, R. Hirschheim, G. Fitzgerrald and T. Wood-Harper (Eds.), *Research methods in information systems : proceedings of the IFIP WG 8.2 Colloquium, Manchester Business School,* Amsterdam New York : North-Holland New York : Sole distributors for the USA and Canada, Elsevier Science Pub:193-201

Borcz, R.(1996) Software failures, faults and errors:their risk, management and control. Paper presented at the *Proceedings of the ESA 1996 Product Assurance Symposium and Software Product Assurance Workshop (SP-377)* Noordwijk, Netherlands, March :221-226

Bröckers, A.(1995) Process-Based Software Risk Assessment, *Lecture Notes in Computer Science* **913**:9-29

Bronte-Stewart, M.(1998) A business information consulting approach for SMES, *Computing and Information Systems* 5(1):21-27

Brooks, F.P.(1986). No Silver Bullet: Essence and Accidents of Software Engineering, *Information Processing* '**86**:10-19

Brown, N.(1996) Industrial Strength Management Strategies, *IEEE Software* **13**(4):94-103

Button, G.(2000)The ethnographic tradition and design, *Design Studies* **21**(4):319-332

Cardenas-Garcia, S. and Zelkowitz, M.V.(1991) A Management Tool for Evaluation of Software Designs, *IEEE Software* **17**(9): 961-971

Carr, M. J ., Konda, S.L., Monarch, I. and Ulrich, F.C.(1993)Taxonomy-based Risk Identification, Technical Report CMU/SEI-TR-6 ESC-TR-93-183

Carr, M.J. (1997) Risk Management may not be for everyone, *IEEE Software* **14**(3):20 & 22

Chandrashekar, S., Mayfield, B. and Samadzadeh, M.H.(1993) Towards automating software project management, *International Journal of Project Management* **11**(1):29-38

Chang, C.K. and Christensen, M.(1999)A Net Practice for Software Project Management. *IEEE Software* **16**(6):80-88

Chapman, C.B. and Ward, S.C.(1997) *Project risk management: processes, techniques and insights*, Chichester : Wiley

Charette, R.N. (1989) *Software Engineering Risk Analysis and Management*, New York: Intertext Publications :McGraw-Hill Book Co

Charette, R.N.(1991) The risks with risk analysis, *Communications of the ACM*, **34**(6):106

Charette, R.N.(1996)Large Scale Project Management is Risk Management, *IEEE Software* **13**(4):110-117

Charette, R.N., MacAdams, K. and White, M.B.(1997) Managing Risk in Software Maintenance, *IEEE Software* **14**(3):43-50

Charette, R.N.(1999) The competitive edge of risk entrepreneurs, *IT Pro*, **1**(4):69-73

Chatterjee, D. and Ramesh, V.C.(1999) Real Options for Risk Management in Information Technology Projects. Paper presented at the *Thirty-second Annual Hawaii International Conference on System Sciences*, Hawaii, January

Checkland P. (1995) Model Validation in Soft Systems Practice, *Systems Research*, **12**(1): 47-54.

Chen, S.(2001) Fuzzy group decision making for evaluating the rate of aggregative risk in software development, *Fuzzy Sets and Systems* **118**:75-88

Chittister, C. and Haimes, Y.Y.(1993) Risks Associated with Software Development: A Holistic Framework for Assessment and Management, *IEEE Transactions on Systems, Man and Cybernetics* **23**(3):710-723

Clergy, C.(1994) Psychology and information technology: The study of cognition in organisations, *British Journal of Psychology* **85**(4):449-477

Collier, B., DeMarco, T. and Fearely, P.(1996) A Defined Process for Project Postmortem Review, *IEEE Software* **13**(4): 65-72

Conrow, E.H. and Shishido, P.S.(1997) Implementing Risk Management on Software Intensive Projects, *IEEE Software* **14**(3): 83-89

Cotton, T.(1996) Evolutionary fusion: A customer-oriented incremental life-cycle for fusion, *Hewlett Packard Journal* **47**(4):28-38

Covello, V.T.(1987) Decision Analysis and Risk Management Decision Making: Issues and Methods, *Risk Analysis* 7(2):131-139

Cox, E.(1994) *The fuzzy systems handbook: a practitioner's guide to building, using, and maintaining fuzzy systems* , Boston : AP Professional :435-446

Cukic, B., Ammar, H.H. and Leteef, K.(1998) Identifying high-risk scenarios of complex systems using input domain. *Paper presented at the 9th International Symposium on Software Reliability Engineering*, Paderborn, Germany, November

Cule, P.(2000) Strategies for heading off is project failure, *Information Systems Management* **17**(2):65 - 74

Davis, G.B., Lee, A.S., Nickles, K.R., Chatterjee, S., Hartung, R. and Wu, Y.(1992) Diagnosis of an information system failure, *Information and Management* **23**: 293-318

Deck, M.(2001) Managing Process Diversity while improving your practice, *IEEE Software* **18**(3): 21-27

Deephouse, C., Mukhopadhyay, T., Goldenson, D.R. and Kellner, M.I. (1996) Software Process and Project Performance, *Journal of Management Information Systems* **12**(3):187-205

Dickey, S.(2000) No Time for Quality Trade-Offs, *Information Week* **781**:177-182

Doolin, B.(1998) Information Technology as Disciplinary Technology: Being critical in interpretive research on information Systems, *Journal of Information Systems* **13**(4): 301-311

Dutta, S., Van Wassenhove, L.N. and Kulandaisamy, S.(1998) European Software Management Practices, *Communications of the ACM* **41**(6): 77 - 86

El Emam, K. and Hoeltje D.(1997) Qualitative Analysis of a Requirements Change Process, *Empirical Software Engineering* **2**(2):143-152

Eldridge, F.(1991) The integration of risk analysis and management in project management, Paper presented at *Reliability '91, The International Conference on Reliability, London*, June

Fairly, R.(1994) Risk Management for Software Projects, *IEEE Software* **11**(3):57-67

Feather, M.S. and Cornford, S.L. and Larson, T.W.(2000) Combining the Best Attributes of Qualitative and Quantitative Risk Management Tool Support, *Paper Presented at The fifteenth IEEE Conference on Automated Software Engineering*, Grenoble, France, September Fenton, N.E.(1991) *Software Metrics: A rigorous approach*, London:Chapman and Hall

Flanagan, E.B.(1995)Risky Business(Software Development Management), *C++ Report* **7**(3)75-78

Fitzgerald, B. and Howcroft, D.(1998) Towards Dissolution of the Research Debate: From Polarisation to Polarity, *Journal of Information Systems* **13**(4):313-326

Frosdick S(1997) The techniques of risk analysis are insufficient in themselves, Disaster Preventation and Management **6**(3):165-177

Galliers, R.D. and Land, F.E. (1987) Choosing an appropriate information system research methodologies, *Communications of the ACM* **30**(11): 900-902

Garvey, P.R., Phair, D.J. and Wilson, J.A.(1997) An Information Architecture for Risk Assessment and Mangement, *IEEE Software* **14**(3):25-34

Gemmer, A.(1997) Risk Management: Moving Beyond Process, *IEEE Computer* **30**(5):33-43

Gilb, T.(1988) *Principles of software engineering management*(1st ed.)Wokingham, Eng ; Reading, Mass : Addison-Wesley

Gilb,T.(1986)Deadline Pressure: How to Cope with Short Deadlines, Low Budgets and Insufficient staffing levels. Paper presented at the *Proceedings of the IFIP 10th World Computer Congress*, Dublin, Ireland, September

Grey, S.(1995) *Practical risk assessment for project management*, Chichester; New York: WileyGuarro, S.B.(1987) Principles and Procedures of the LRAM Approach to information system risk and management, *Computers and Security* **6**(6): 493-504

Hall, E.M. (1998) *Managing Risk: Methods for Software Systems Development*. Reading, MA :Addison-Wesley

Hamilton J.S. and Ives B.(1992)In E. Mumford, R. Hirschheim, G. Fitzgerrald, and T. Wood-Harper (Eds.), *Research methods in information systems: proceedings of the IFIP WG 8.2 Colloquium. Manchester Business School* Amsterdam New York : North-Holland New York: Sole distributors for the USA and Canada, Elsevier Science Pub:132-143

Hammer, T.,Huffman, L., Wilson, W., Rosenberg, L. and Hyatt L(1996)Requirement Metrics for Risk Identification, Paper presented at the *Twenty-First Annual Software Engineering Workshop at NASA-GSFC, MD*, December

Hantos, P. and Gisbert, M.(2000) Identifying Software Productivity Improvement Approaches and risks: Contruction Idustry Case Study, *IEEE Software* **17**(1):48-56

Henning, P.H.(1998) Ways of Learning, *Journal of Contemporary Ethnography* **27**(1):85-137

Hirschheim, R. and Klein, H.K.(1994). Realizing emancipatory Principles in Information Systems Development: The Case for ETHICS, *MIS Quarterly* **18**(1): 83-109

Hirschheim, R.(1992). Information Systems Epistemology: An Historical Perspective, In R. Galliers(Ed.), *Information systems research : issues, methods, and practical guidelines* Boston : Blackwell Scientific Publications:28-60

Huges, B. and Cotterell, M.(1999) *Software Project Management*(2nd ed.),London: McGraw-Hill

Jackson, M.(1998) Will there ever be software engineering?, *IEEE Software* **15**(1):30-39

Jalote, P. (1997) *Integrated approach to software engineering* (2nd ed.) New York: Springer-Verlag New York Inc

Jones, C.(1994) *Assessment and Control of Software Risks,*. Englewood Cliffs, N.J : Yourdon Press

Jones, C.(1996) Our Worst Current Development Practices, *IEEE Software* **13**(2):102-104

Känsälä, K.(1997) Integrating Risk Assessment with Cost Estimation, *IEEE Software* **14**(3): 61-67

Kaplan, B. and Duchon, D.(1988)Combining Qualitative and Quantitative Methods in Information Systems Research: A Case Study, *MIS Quarterly* **12**(4): 571-586

Keil, M., Cule, P.E., Lyytinen, K. and Schmidt, R.(1998) A Framework for Identifying Software Project Risk, *Communications of the ACM* **41**(11):76-83

Keil, M.(2000) A cross-cultural study on escalation of commitment behaviour in software projects, *MIS Quarterly* **24**(2):299-326

Kitchenham, B. and Linkman, S.(1997) Estimates, Uncertainty and Risk, *IEEE Software* **14**(3):69-75

Klein, H.K. and Myers, M.D.(1999) A set of principles for conducting and evaluating interpretive field studies in information systems, *MIS Quarterly* **23**(1): 65-93

Klein, G. and Jiang, J.J.(2001) Seeking consonance in information systems, *Journal of Systems and Software* **56**(2):195-202

Kontio, J., Getto, G. and Landes, D.(1998)Experiences in improving Risk Management Processes using the concepts of the Riskit Method, *Proceedings of the ACM SIGSOFT sixth international symposium on Foundations of software engineering* New York : ACM Press: 163-174.

Lacity, M.C. and Janson, M.A.(1994) Understanding Qualitative Data: A Framework of Text Analysis Methods, *Journal of Information Systems* **11**(2):137-155

Lam W and Vickers AJ(1997) Managing the risks of component-based software engineering, Paper presented at Proceedings. *Fifth International Symposium on Assessment of Software Tools and Technologies*, USA, Pittsburgh, June

Laporte, C.Y. and Papicco, N.R.(1998) Software and Systems Engineering Process Development and Integration at Oerlikon Aerospace, *Software Process Newsletter* **11**:10 -20

Lawson, H.W.(1998) Infrastructure Risk Reduction, *Communications of the ACM* **41**(6): 120-120

Lee, A.S(1994) Electronic Mail as a Medium for Rich Communication: An Empirical investigation Using Hermeneutic Interpretation, *MIS Quarterly* **18**(2):143-157

Lee, A.S.(1999) Rigour and Relevance in MIS Research: Beyond the approach of positivism alone, *MIS Quarterly* **23**(1):29-33

Lister, T.(1997) Point: Risk Management is Project Management for Adults, *IEEE Software* **14**(3):20 &22

Longstaff, T.A., Chittster, C., Pethia, R. and Haimes, Y.Y.(2000) Are we forgetting the Risks of Information Technology, *IEEE Computer* **33**(12):43-51

Lott, C.M.(1997)Breathing New Life into the Waterfall Model, *IEEE Software* **14**(5):103-105

Lyytinen, K., Mathiassen, L. and Ropponen, J. (1998)Attention Shaping and Software Risk-A Categorical Analysis ofFour Classical Risk Management Approaches, *Information Systems Research* **9**(3):233-251

Lyytinen, K., Mathiassen, L. and Ropponen, J.(1996) A framework for software risk management, *Journal of Information Technology* **11**(4):275-285

MacNamee, D.(2001)Risk Assessment Glossary <http://www.mc2consulting.com/riskdef.htm>

Madachy, R.J.(1995) Knowledge-Based Risk Assessment and Cost Estimation, *Automated Software Engineering an International Journal* **2**(3):219-230

Madachy, R.J.(1997) Heuristic Risk Assessment using cost factors, *IEEE Software* **14**(3):51-59

Markus, M.L. and Lee, A.S.(1999). Special Issue on Intensive Research in information systems: using qualitative, interpretive and case methods to study information technology-second installment; forward, *MIS Quarterly* **23**(1):35-40

Maude, T. and Willis, G(1991) *Rapid Prototyping: The Management of Software Risk* London:Pitman

May, E.L. and Zimmer, B.A.(1996) The Evolutionary Model for Software,*Hewlett-Packard Journal* **47**(4):39-46

McConnell,S.(2000)The Best Influences on Software Engineering, *IEEE Software* **17**(1):10-17

McFarlan, W. F(1974) Portfolio Approach to Information Systems, *Harvard Business Review* **59**:142-150

McGaughey, R.E., Snyder, C.A. and Carr, H.H.(1994) Implementing information technology for competitive advantage: Risk management issues, *Information and Management* **26**: 273-280

Melville, S. and Goddard, W.(1996) *Research Methodology*, Kenwyn:Juta

Middleton, P.(1999) Managing information system development in bureaucracies, *Information and Software Technology* **41**(8):473-482

Mingers, J.(2001) Embodying information systems: the contribution of phenomenology, *Information and Organization* **11**(2),103-128

Molka-Danielsen, J.(1996)IN 320 Strategic Use of Information Technology: Class 4. <http://home.himolde.no/~molka/in320/t99c4.html>

Monarch, I. and Gluch, D. P.(1995) *An Experiment in Software Development Risk Information*, Technical Report CMU/SEI-TR-95-014

Moore, N.(1999) *How to do research: the complete guide to designing and managing research projects*(3rd ed.), Publisher London: Library Association

Moreno, V. Jr.(2001) On the social implications of organizational reengineering, *Information Technology & People* **12**(4): 359-389

Moustakas, C.(1994) *Phenomenological research methods*, Publisher Thousand Oaks [Calif.]: Sage

Moynihan, T.(1997) How experienced Project Managers access risk, *IEEE Software* **14**(3): 35-41

Neumann, P.G.(1991a)Inside Risks: Certifying Professionals, *Communications of the ACM* **34**(2):130

Neumann, P.G.(1991b)Inside Risks: The Human Element, *Communications of the ACM* **34**(11):150

Neumann, P.G.(1991c) Inside Risks: The not-so Accidental Hotlist, *Communications of the ACM* **34** (9):122

Neumann, P.G. (1993) Inside Risks: Are Dependable Systems Feasible, *Communications of the ACM* **36**(2):146

Neumann, W.L.(1997) *Social research methods : qualitative and quantitative approaches*(3rd ed.).Boston, MA: Allyn and Bacon

Nidumolu, S.R.(1996) A comparison of the structural contingency and risk-based perspective on coordination in software development projects, *Journal of Management Information Systems* **13**(2):77-113

Nottingham, A.(1996) Risk Assessment of BPR Projects, *Computing and Information Systems* **3**(2):58-65

Olson, D.(1993) *Exploiting Chaos*, New York: Van Nostrand Reinhold

Ould, M.(1999) *Managing Software Quality and Business Risk*,. West Sussex; England: Wiley

Pescio, C.(1997) When past solutions cause future problems, *IEEE Software* **14**(5):19-21

Pfleeger, S.L.(1998) *Software Engineering: Theory and Practice*, Englewood Cliffs; New Jersey:Prentice-Hall

Pfleeger, S.L.(2000) Risky business: what we have yet to learn about risk management. *Journal of Systems and Software* **53**(3):265-273

Phillips, D.(1996) Project Management: Filling in the Gaps, *IEEE Software* **13**(4):17-18

Pighin, M. and Zamolo, R.(1997) A Predictive Metric based on Discriminant Statistical Analysis, *Proceedings of the 1997 International Conference on Software Engineering*, New York : ACM: 262-270

Powell, P.L. and Klein, J.(1996) Risk Management for information systems, *Journal of information Technology* **11**:309-311

Pressman, R. S.(1995) Software according to Niccolo Machiavelli, *IEEE Software* **12**(1):101-102

Pressman, R.S. (1996) Manager: Software Process Perceptions, *IEEE Software* **13**(6):16 &18

Pressman, R. S.(1997) *Software Engineering - A practitioner's approach* (4th ed.),New York : McGraw-Hill

Rainer, R.K.Jr, Synder, C.A. and Carr, H.H. (1991)Risk Analysis for Information Technology, *Journal of Management Information Systems* **8**(1):129-147

Ramamoorthy, C.V., Chandra, C., Ishihara, S. and Ng, Y.(1993) Knowledge-based Tools for Risk Assessment in Software Development and Reuse, Paper presented at *Proceedings : Fifth International Conference on Tools with Artificial Intelligence*, Cambridge, November

Reifer, D.(2000) Requirements Management: The Search for Nirvana, *IEEE Software* **17**(3):45-47

Ropponen, J. and Lyytinen, K.(1997) Can software risk management improve system development: an exploratory study, *European Journal of Information Systems* **6**(1):41-50

Ropponen, J. and Lyytinen, K.(2000) Components of Software Development Risk: How to address them ? A Project Manager Survey, *IEEE Transactions on Software Engineering* **26**(2): 98-112

Ropponen, J.(1993) *Risk Management in Information Systems,* Doctoral thesis, Department of Computer Science and Information Systems, University of Jyväskylä.

Royce, W.(2000) Software Management Renaissance, *IEEE Software* **17**(4):116-118

Sayer, K.(1998) Denying the technology: middle management resistance in business process re-engineering, *Journal of Information Systems* **13**(4):247-256

Schach, S.R.(1993) *Software Engineering* (2nd ed.) Homewood, IL : Aksen Associates : Irwin

Schach, S.R.(2002) *Object-Oriented and Classical Software Engineering* (5th ed.) McGraw-Hill

Schmidt, C.,Dart, P.,Johnston, L., Sterling, L. and Thorne, P.(1999) Disincentives for communicating risk: a risk paradox, *Information and Software Technology* **41**:403-411

Schultze, U.(2000) A confessional account of Ethnography about Knowledge Work, *MIS Quarterly* **24**(1): 3-41

Seaman, C.B.(1999) Qualitative Methods in Empirical Studies of Software Engineering, *IEEE Transactions on Software Engineering* **25**(4):557-572

Shepperd, M.(1993) *Software engineering metrics*, New York : McGraw-Hill

Sherer, S.A.(1992) *Software Failure Risk Measurement and Management*, New York: Plenum Press

Sherer, S.A. (1995) The Three Dimensions of Software Risk: Technical, Organizational and Environmental, Paper Presented at the *Proceedings of the 28th Hawaii International Conference on System Science*, Hawaii, USA, January

Sommerville, I.(1985) *Software Engineering* (2nd ed.), Wokingham: Addison-Wesley

Sommerville, I.(1996) *Software Engineering*(5th ed.), Reading, Mass: Addison-Wesley Pub. Co

Steen, M. (1997) Avoiding the Pitfalls of Risk., *InfoWorld* **19**(51/52):75-77

Stone, D.N.(1990) Assumptions and Values in the Practice of Information System Evaluation, *Journal of Information Systems* **94**(2):1-18

Strassberg, D.(2001) In developing measurement applications, mixing metaphors minimizes muddles, *EDN* **46**(2):129-136

Strigini, L.(1996) Quality Time- Limiting the Dangers of Intuitive Decision Making, *IEEE Software* **13**(1):101-103

Toth GA(1994) Software Technology Risk Advisor, *In Proceedings of the 9th Knowledge-Based Software Engineering (KBSE-94) Conference*, Monterey, CA: IEEE Computer Society Press:179-188

Voas, J., McGraw, G., Kassab, L., and Voas, L.(1997) A 'Crystal Ball' for software Liability, *IEEE Computer* **30**(6): 29-36

Walsham, G.(1993) *Interpreting information systems in organizations*, New York: Wiley

Walsham, G.(1995) Interpretive case studies in IS Research, *European Journal of Information Systems* **4**(2): 74-81

Williams, R.C., Walker, J.A. and Dorofee, A.J.(1997) Putting Risk Management into Practice, *IEEE Software* **14**(3):75-82

APPENDIX A

THE INSTRUMENT USED IN THIS INVESTIGATION

A1. Part One: Survey Questions (adapted from Ropponen and Lyytinen,1997)

Subjects were given this questionnaire to fill in before the interview. The survey determines how well the participant is managing the risks in software development. Participants were not told that this was related to determining the how successfully risk management was being implemented since the participant might then answer in such a way to justify its use or non-use.

Table A1, represents a list of statements describing projects. Subjects were told to mark an appropriate alternative for each statement based on *their* experience. Participants had to choose one alternative based on how often the described situation occurs

Table A1: Quantitative Survey Questions

	Hardly ever	Rather seldom	Half	Rather often	Almost always
Your project has considerable problems due to personnel shortfalls	1	2	3	4	5
Your project is completed according to the timetable	1	2	3	4	5
Resource consumption reaches its top as you approach your project deadline	1	2	3	4	5
Actual project costs and estimated costs are nearly equal in your projects	1	2	3	4	5
Your project is cancelled before completed it	1	2	3	4	5
A failure to estimate project size interferes considerably with implementation	1	2	3	4	5
Demand of personnel is estimated correctly in your project	1	2	3	4	5
Time consumption of your project is constant	1	2	3	4	5
Your personnel's expertise in methods, software and equipment is insufficient	1	2	3	4	5
The complexity of your project and its effect are easy to manage	1	2	3	4	5
Developed software functions and properties meet user's needs	1	2	3	4	5
Developed software includes complex, but only marginally useful properties	1	2	3	4	5
Software requirements are continuously changed	1	2	3	4	5
Your project timetable is changed continually	1	2	3	4	5
Users are not satisfied with the implemented user interface	1	2	3	4	5
Externally purchased components and equipment meet your expectations in your project	1	2	3	4	5
You have unrealistic expectations of the project members skills	1	2	3	4	5
Performance requirements(response time, computing efficiency) are estimated incorrectly	1	2	3	4	5
Subcontracted tasks in the project are performed as expected	1	2	3	4	5
Software and hardware capabilities are estimated incorrectly	1	2	3	4	5

A2. Part Two: Interviewing Guidelines and Questions

Introduction

This presents a structured interview template for Risk Management Experiences. This interview template is used to support consistent, semi-structured interviews for the cases that are analyzed in this study.

Study Goals

- Analyse the Risk Management Process
- Identify Potential issues and observations
- Problems, benefits, disadvantages, improvement suggestions

Interview Template

The interviewee would be briefed as follows:

The purpose of this interview is to collect your observations and experiences from the risk management perspective. It is of vital importance that you answer the questions as objectively and candidly as possible. The interview information is used for research purposes only. If you so wish, total anonymity is guaranteed for your participation.

Questions

1. Background Information

1.1 Interviewee's Name
1.2 Position at Organization
1.3 Years of Experience in Project Management
1.4 Years of experience in risk management?
1.5 How much training have you received in risk management?
1.6 What the size of your organization?
1.7 What is the nature of your applications?

Interview Questions (adapted from Kontio et al (1998))

2. In your own words, characterize your project's risk management infrastructure along main attributes:

2.1 Culture -
the level of awareness about risk management
attitude towards risk and risk management
risk averse/risk taking
is the discussion of risks encouraged
is risk management recognised as a legitimate activity

2.2 Policy -
the stated management commitment to risk management and how it is enforced

2.3 Methods -
what methods and techniques are used and supported for risk management

2.4 Tools -
what tools and templates are used in risk management

2.5 Skills and competence -

what risk management skills and competencies exist, what training is available and given to personnel for risk management

2.6 Support structure -
what type of organizational support exists to help perform risk management in projects, how much resources are made available for this task

2.7 Experience capture process -
what mechanisms exist to capture, accumulate and analyze risk management experience

3. **Concluding Questions**

3.1 Overall, what was the impact of risk management in a project?

3.2 What are the most critical problems areas in risk management?

3.3 What techniques would require more clarification or help ?

APPENDIX B

TRANSCRIPTS FROM THE QUALITATIVE PART OF THE INTERVIEWS

B1. Interview With Company A

Interviewer: Do use you risk management in your project?

Interviewee: Explain to me your software in risk management?

Interviewer: Where, you before you begin a project, you identify the types of risks that you feel will be in that project before taking it on, like for instance maybe, you have a shortfall of personnel, you don't have enough personnel that can handle the project that you have or that kind of thing or if you feel like, I'm not sure how you do your processes....

Interviewee: Let me explain, we normally when we start with a new project, we evaluate what would be required as far as the resources are concerned and that is now just people that (needs hardware?), the software that we would require etc, etc. in other words, we take the whole spectrum, we work through it and we say, we need so many people for so many hours because first of all where it comes to people you've got to evaluate whether you want to finish this specific project in a month, six months or a year, if I do project A, which is if I do it in a month I'll only need ten people, if I do it over eighteen months or a year I'll need more but many case, so obviously, or let's say within a month, I'll need twelve people, over a year I'll just need just one person to do that same project its going to take me that X number of hours, yes, the first thing is the negotiate final test and the delivery date and then we work from that we go and find out, do we need additional staff, do we need staff from outside and you've got to take into consideration somebody walks in fresh here, doesn't know how our systems works so you've got to combat all that. But what we normally do is, for any project we use Microsoft Project and we layout everything that needs to be done in other words the, the software, everything.

Interviewer: So it's not like a formal identification of risks as such, where you feel like there is a particular thing that can go wrong, you assume that you can handle whatever project that you have, you know, the specifications of that project and you know, assess it and you evaluate and you decide ok we need ten people or whatever, so you don't take it from a, or make it this point of view, what can go wrong

Interviewee: Yes, of course we do, ok, first of all, if typically any of my staff would have been involve in a major motor accident, and they are in hospital for a month or two months or are not able to come to work for two months, then yes one has got to, at the beginning of the project, you've got to say, if I lose anyone or two staff members I will obviously have to extend my resources to that point therefore I'll need additional resources which will have to be brought in from outside or whatever, if you bring in additional resources from outside what would have taken my staff member maybe a month, will take me outside resource two months, in that instance, one will obviously split it up, in other words, what we do on the project is we say ok, that task will take so long but we build in extra time for if the guy is ill for one or two days etc. you never get a hundred percent out of any person

Interviewer: Is that like a formal process or do you have like a brainstorming session, how do you decide that, you know, that you going to have that particular problem, how do you do that?

Interviewee: Ok, we've done plenty of projects, so yes, one goes on your experience, first of all, you go on statistics of what happened to projects in the past, and then you also, we call it fat, you also got to build in some fat into each and every of our field of projects, in other words, I will not promise a client a project at the end of the month, if I know I've need to use all my resources even by 75% to get to the end of the month. I work on the bases in that I found to work well, bases of 50% availability of 10%, in other words, they will be able to give me four hours production out of eight hours everyday. If the company pushes for it to come in earlier good and well, then when project A is finish and ready for implementation, in other words, we could put it in and we can say right , this thing's just waiting for implementation, we can say for project B. In other words, but if anything goes wrong in Project A, then at least I've got until then to finish Project And that's the way we would know it's not A Formal, when I say Formal, Yes we use MS Projects which we found brilliant for software development. It's got all the facilities in to provide for this and that in your dependencies etc. etc.

Interviewer: How many years experience do you have in Project Management?

Interviewee: Presently, '71 up 'till now. 1971, that's before you were born. I've been doing this for 30 years.

Interviewer: 30 Years. And so this Risk Management is not a formal thing to you, so I need to know how many years you had in Risk Management, but you said you did it informal thing. So ..... 30 years experience .....

Interviewee: Because of the experience is gained , I know the thing in and out so, yes , Risk Management, yes so comes as a , as something that you've learnt in the past. It comes naturally, yes, I can remember my first project that I was project leader of. I promised them a month, it came in after six months.

Interviewer: So you've learnt from that.

Interviewee: Over budget, over time, everything. yes, that was my first month, from that I learnt, and I learnt a lot after that

Interviewer: Ok, in you own words, in terms of your staff, what is their level of awareness of risk management, do they do it exist, or

Interviewee: I presume, they feel the pressure from me, and the other director Priscilla, she also ??? (right?), she got since, she started doing her first projects that was way back in 70... no, 80 something, somewhere there.

Interviewer: And what's the attitude towards risk and risk management, , I mean like, if they had identify a problem somewhere down the line will they come and, you know, openly come and see you and communicate it to you, ok, so they do

Interviewee: Yes, yes, yes, if they see anything that is going to cause anything in the future, then they come and we discuss now and we resolve it even if it needs some report that's going to be develop right at the end, then they come and say "we foresee that," in other words, yes, it's not a, it's not a formal process but a, they could quite easily come and talk to us.

Interviewer: Yes, in your experience, how do you get them to easily talk to you because you find that if somebody discovers the problem they always afraid that it will become their problem to sort it out. You know what I mean, it's going to be their responsibility, so how do you facilitate that, do they feel comfortable enough to still come and tell you.

Interviewee: Ok, problems within any development with us or with our environment is not received as your problem, if the three of us are working on a project, if you see a problem coming you not going to get it, if want it you can have it, the way we do it normally is, this is the problem, how are we going to solve it, and who wants to solve it, but normally in the discussion that (how?) and then who wants to do that physical work, in other words, we come up and we say, you've got to do that, not you, somebody, Person A has got to do that, Person B, or let's try call it, there is a task A and task B and task C and a task D here, ok, then we say to the people these task you can't split up into a smaller task, do you want A, no, you don't want A, you don't like that, do you want it ?

Interviewer: And what if it happens that the person who identifies the problem is response... he created that problem in some way, you know what I mean, so, I mean, do you....

Interviewer: No, no, I don't think it, is it like, you know, as you said, you know, everybody is responsible, so it is not like, you know, because you created that particular problem, now you going to get the downfall for it, so is it like an open communication, nobody is afraid to bring it out.

Interviewee: Let me tell you, outside of projects or inside of projects, it's irrelevant, my management style are the following: something happens, something went wrong, the first question I want to know is, how can we repair the damage, then the second one is, how can we prevent it from happening in the future, and the third is, who caused it and if it was you who caused it, how can we help you overcome it so that this thing won't happen in the future, people come up to me it's 100% legitimate reason, I forgot, everybody forgets, I forget to comb my hair at home and get here with my hair standing in each and every direction, how do I prevent myself from doing this from happening again to me, by putting the comb somewhere when I go past it I'll see it, if, and, this is I think where my management style in the end works and that's why people are willing to with their problems, you cause the problem, doesn't matter why, you forgot, you did something wrong (?????), doesn't matter, you are a bad girl, go and stand in the corner, for a half-a-day, because you are such a bad girl you caused this problem, ok, what do I gain from that, nothing, I lose a half-a-day's work from you, you are mad at me, you are not going to work properly tomorrow or the day after that, so why, why come down hard on you rather look at the problem, what is the problem, why did you do it, and then lets finds a way to get around it, so it doesn't happen in the future. And that I find works wonders with anybody and that causes then, yes I have design this table and there is a shortage of (?????) on these 20 programmes already (based?), finished, based on this, and all of a sudden we find out there is one field missing, it's your fault, but.....

Interviewer: But Risk Management itself is not a legitimate formal activity it's just inborn,

Interviewee: No

Interviewer: I mean it's like you don't identify with it, I mean you don't analyze them, you don't rank them and go

Interviewee: On my people,

Interviewer: Yes

Interviewee: Sub-consciously I think I do, we did, (database?) layout design review so, yes I would personally urge you to do a database, he wants to do I will allow him to help you and gain experience and confidence and then we will ? do not, ? database we will give him for (update?) test it and see how it works..... so, yes, subconsciously I presume (?????)

Interviewer: At the moment you have anybody in your staff that, you know, have the skills and the competence to handle this management formally, I mean, are you planning to have them training for them in the future or are you just, or you feel that this informal process is working for you so that you don't need to actually

Interviewee: Ok, at the moment no, we don't feel that there's any, any training, I think the people get their training as time goes along, JA, one has book learning experience, book learning is very good and you know about all the other guys', all the mistakes that they made and you now avoid it or you can make mistakes and, if you go with book learning there are some things that you do forget which you've learnt, if you go with experience then normally one remembers of course they remember that so I must remember that, so when it comes to language training in other words, oracle, , visual basic, or whatever then I do send the people on courses. Database design, yes, we do send people on a database design course, and then when we come back, we say forget about everything you've learnt, that you now that you basically now bases of the database should look, forget everything you've learnt this is the way we do.

Interviewer: And you did say you capture the process like what went wrong in previous projects is that like a formal thing

Interviewee: Well, when you still use, I can't remember the name, but then that was way back, and then we switch over to Microsoft. Yes, Microsoft project, in there, there's ample space for notes, and in our, when we get together, we've normally got a weekly meeting, to look at how people has progress etc, etc. And there we make notes and we say look this has happen, this has gone wrong, this is good, this is something somebody came up with which is working well and we keep it there.

Interviewer: So, even though you have perform risk management informally you'd have a positive impact on the project eventually.

Interviewee: Yes, I assume so. Well, I hope so

Interviewer: What other, are there any problems with risk management itself in your current format informally do you think there's any room for improvement there

Interviewee: I assume anywhere there's room for improvement, yes,

Interviewer: How

Interviewee: How, I don't know, somebody else will have to tell you, we are currently doing it

Interviewer: Well, that is very interesting because every, I have interviewed a few people so far and lots of people say they don't know how, Interviewee: Yes, no, we don't know how, how do you fit the electricity into a new building, how do you provide for it, I'm asking you.

Interviewer: I don't know.

Interviewee: You've never done it before, it's the same problem with that, the risk management takes a new concept and yes, we haven't address that yet, so

Interviewer: So it actually it's not that new, because it has been around the 1980's,

Interviewee: I know,

Interviewer: It's not that new, but I mean, relatively

Interviewee: Ok, let me tell you a secret, I've got many years of experience there's a lot of concepts that's been borne or that was borne in the past and it's going to be borne again in the future (in other words?) a new borne name, which people find and they say, it is new but it's really not something new maybe somewhere in the past it was name under another name. I think when I started, I started off with punch cards, you don't know what it is? You don't even know what it looks like?

Interviewer: I don't even know what it looks like

Interviewee: Exactly, One of that is risk management factor there was, you had to be very kind and very friendly with the operator because these punch cards came in, were carried in a box and you give it to him and he puts in the card reader and he reads it and your programmes runs etc. or it gives out some errors and then you change it. But if he doesn't like you or you shout at him once or twice or three times and all he does is he drops the cards, it's a accident, you got one or two choices, you punch the whole programme and punching the whole programme wasn't that easy or you sit and you sort the cards but there was no way of knowing you have to read the top of the card, and you'll decide this one comes after that one, that one comes after that one. so that in those stages one of the factors of risk was the operator dropping it so you took him out at least once a month for a beer so that he's friendly, he's kind and he looks when he carries your box make sure he doesn't drop it. And yes, I think there was a lot of things that in the past people did unconsciously, and they didn't realise it was, this was managing risk, this was predicting the length of the project, because (?) the first projects we went like, JA, too much.

Interviewer: Well, risk management involves firstly identifying which you do then it involves actually giving like a numerical value to that risk do you see that it's useful or you know, do you think it's better, you do it very informally, you just rank it in your subconscious and then you know which is more important than the other but do you think, do you think like that hampers the process by sitting down and giving it a numerical value because at the moment risk management is done very quantitatively.

Interviewee: I think it could be something positive, there's one proviso to it, how much additional time is it going to add to any of my projects

Interviewer: Which it probably would, you know

Interviewee: Yes, but how much? Is it going to add one percent, two percent, I can accept one percent, I can accept two percent, (????) gets to ten percent and that's hitting my pocket.

Interviewer: And it's usually found that eight percent of risk management is useful beyond that it becomes problematic, ?resourceful (?) so you can, the problem is keeping it down to the eight percent and not sitting and you know over analysing it

B2 Interview with Company B

Interviewer: What is your position ?

Interviewee: I'm the manager, The manager.

Interviewer: I am doing a study exploring risk management techniques, attitudes ...

Interviewee: What do you mean by risk management ?

Interviewer: Basically the activities like.... identifying your risks, and then analyzing those risks...

Interviewee: ok, ok our company didn't really want to do risk management that/but we are trying to put something in place ah that will basically (?) that type of thing ah I don't know how in terms of how I can tell you risk management all we do is we identify the first stage we just identify a lot of, lot of possible information about the system if we do that. When we, when we get that information we analyze that information and then we can identify things like integrating systems, you know, and that is a problem where our problem will lie, ok, so we will identify that problem and see how we will overcome that and then in the system itself that is in the databases are before we even start to develop it. We will try to accommodate that in your database. In terms of risk management or I'd rather say crisis management at this stage when we've got three systems and the data differs and we didn't know that at one stage so what we had to do is, we had to back, I don't know, in that type of crisis we just try to see what the problem was and how we can solve it ASAP, so...

Interviewer: It is normally found with research we use a controlled environment and we won't learn much from that, we want to learn factors such as, perspectives and attitudes. How many years experience do you have with project management ?

Interviewer: ok, what few years of experience do you have?

Interviewee: oh well it's four years.

Interviewer: In project management four years.

Interviewee: oh no, not in project management, in development it is four years and project manage about a year now.

Interviewer: And in risk management

Interviewee: Well, my position does everything, or whatever is necessary to make things work and make the clients happy if that is what my job is so it involves project risk management and that type of thing.

Interviewer: And how much of training have you received in risk management?

Interviewee: I Don't know?

Interviewer: None, ok, and what is it like the level of awareness of risk management within the Company amongst all the personnel?

Interviewee: In terms of developing systems or in terms of client satisfaction what...

Interviewer: yes both of the things basically

Interviewee: ok, generally I am the only one that will, will actually kill the fires if you know what I mean, so if there's a problem with the client's system at the client's whatever system here when we tested and then I will be the one that will go fix it.

Interviewer: So, and now what you are trying to say is that if somebody on the ground perceives something as a problem they are not going to tell you?

Interviewee: No, what we have done is we've divided into two teams, ok, and the team I'm leading now is, I've said to them, we are all together, three heads are better than one so I'm not going to say this how I see it and this is how we are going to do it. That's not how I'm going to do it. I'm going to take everybody's perspective, weigh it and see which one is going to be the best.

Interviewer: So, that's open communication

Interviewee: Oh definitely.

Interviewer: So the discussion of risks is it encouraged, like before you begin something do you sit around a table....

Interviewee: Oh yes, yes like the brainstorming type of session, yes, we discuss every issue so what we do first is we get, it is a planning phase, where we get all the information that we want from what the procedures are this stage, information about the processes everything, and then we draw our processes, and we see how generic can we make for each process and see what the problems involve is, if you do it this way this is going to happen so if you do that way, this is what going to happen, so then we sit together and ok what/which is the best way to do it, then we would say this is the best way.

Interviewer: How do you identify risks ? ....

Interviewee: Ok, because we've already, look every (certain?) is different, ok I can speak for the other team as well where they've got a basically a brand new system which they have to write. In that case, yes then, then you will have like questionnaires that type of thing where you will go to the user, give them the questionnaires, we don't personally do it but if you have to write a new system then I would say it is a good thing, to identify how we can help the user, how we can make his work better, how we can rule out time aspect where they can work faster and that type of thing so it's a very, it's very wide field.

Interviewer: and is risk management viewed like a legitimate activity where like...

Interviewee: no, no

Interviewer: so you don't have like a policy about risk management in your company at all

Interviewee: Nope

Interviewer: you did say you have a brainstorming sessions, once you have identify risks how would you analyze these risks?

Interviewee: Well, it will depend on what, what type of risk it will be, if it is a data risk then we will have to sort that out. I mean we didn't know that until, until we've already implemented the system, so we couldn't identify that, so we only could identify that risk later on and that was the data risk so all we did was added a new table and then write a query to update that.

Interviewer: Oh, what I'm trying to ask/ascertain is, assuming that you identify a large number of risks, obviously you can't you know you can't take care of all twenty so do you have some sort of procedure to rank them and well this is our main risk we must see to this first before we do anything else, do you have like a procedure to rank it, you know, give it like a value or something.

Interviewee: No, we rank it according to, to our processes, and as we come up with

Interviewer: Which is the most important process and then you say, well this is affecting our process and it's going to be our..... ok, great. So you don't have any automated tools to do risk management.

Interviewee: None

Interviewer: ok, what kind of skills exist, isn't there training available for people to learn more about risk management

Interviewee: In our company, well there is not a lot of project management going on, you know it's a hands on, everything you see so the problem with, with our company is there is not a lot of management skills, ok, but the guys on top must have a, must have everything, he must programme me, he must manage, he must do this, do this, make sure, I don't think every/anywhere in South Africa you'll get a company because when it comes to risk management we don't see it as a risk where we plan and brainstorm, we see it as a challenge for..., this is a problem-solver, so it is not a risk wise action, see the problem is we....

Interviewer: Basically, you feel like whatever you have, whatever problems there are, you'll be able to handle them, so you don't feel that it insolvable and that you must proactively decide you know, I don't know if I can go through with this or not, so you don't do that because you believe that you can achieve it.

Interviewee: JA, that's why you plan beforehand, you plan for that problem, if you see that problem, you'll plan for it so that you can work it out or through it. Ok, I think risk management comes in where you where you have a problem where the systems have been already implemented. The problem appears out of that of users that this is the problem and then you have to add something to get around it. That's a risk because now I add something here but I totally screw up something somewhere else. So that is a risk

Interviewer: But then didn't you say like risk management would have prevented that in the first place because if you probatively decided you know, I think that area is going to be a problem then you would have not later on have to add on code or whatever.

Interviewee: That situation occurs usually when you planning, it's not done properly. When you don't have specs that have signed off functional as well as the technical specs and as well as JAD sessions where you talk to the user about all his functionality.

Interviewer: So you believe, your belief is that if your planning is done correctly, there wouldn't be a need for risk management?

Interviewee: There will be risk management factor still where we can say, let's say we debate a (?) , if it's not, if it's not transaction systems you don't know where your data is coming from, if it is a N.S. systems, you must get your data from some where and you must make sure that the, the (?) (updated?). Now, I would say to keep the data (integrity?) very 100% correct, that is going to be a risk when it is implemented.

Interviewer: So once you've completed the project do you have a mechanism to capture what happened so that you can use that information for future projects so that you now ok well you have a problem with this particular area or there's still a shortage in this area and meaning to do something about it. Do you capture that information?

Interviewee: we don't capture it but we do analyze it. What I do is one of my job description is I go to the clients I see if they are working with the system, if they are not using the system, I go and I ask them why, and it's not captured but then they say to me they don't have to do this at all, they, we got, they have a bug in the system so that's why they didn't use it so then I can say ok, training in the one hand and (?) but it is not captured, no.

Interviewer: But don't you feel it a bit useful to gather information?

Interviewee: Because our company is so small, we don't have a (LP?) system implemented here at all. We did try to implement our best system down in the Eastern Cape but I don't know how far we've got there, that's where/when a problem exist, we go on the web/internet page and capture where they are, what they did and what the bug is and then our support guys will go there and help them.

Interviewer: ok, so overall what would you say is the impact of risk management? Is it negligible, not important, is it positive or negative

Interviewee: I think risk management is a very positive thing but I don't think it is risks, like I said I think it is problems and there must be solutions for every one of those problems.

Interviewer: And what you feel that can be improved in risk management, and what you feel like (?) I mean I get this feeling that you feel that risk management is not exactly working for you. What do you feel can be improved to make work for you.

Interviewee: I don't know because my risk management is a question of knowing what is coming and knowing what the problem is going to be and the only way you know how the problem is with your congregation, with your users and your managers.

Interviewer: And are you aware of any techniques are out there, I mean like automated packages that you can use to assist you in risk management.

Interviewee: I think Oracle designed a case course and that type of ....

B3.interview with Company C

Interviewee: I am the Branch Manager

Interviewer: Branch Manager and how many of years of experience do you have with Project Management?

Interviewee: Three

Interviewer: Your specific experience using Risk Management?

Interviewee: What we've done is that we started as a very small company about four years ago and we actually queried at the time, what technology is in place and understand what is out there and not there. We actually made our projects obviously without a clue... so based on that we design a project with technology that is suitable for the environment... that we are working in the IT industry and that is how we do it so obviously there are standards are in there and in the part of the whole thing risk analysis that we do. So technically what we would do is: Analyze the situation, and identifying the critical areas in the company and that what we do, we do business analysis, analyze it and then put the documents together and from thereon manage it accordingly. And sometimes the Company will decide ok fine and we then identify a specific area with a problem and ok what are you going to do about it and they will decide ok we are going to re-design our business processes where we are going to employ more people or whatever the case might be. That is what we do, so we basically identify the risks based on different criteria's, experiences, depends whatever you want to call it.

Interviewer: So what kind of liked methods and techniques do you specifically use like: identifying. Do you have a check list at all?

Interviewee: Yes we do, we've got a checklist. We got like criteria's. So based on those we, trying to say for instance here, a very simple example that would be: user experience on a computer, so we go around and we actually ask and we do a lot of user inputs, we sit down with the people and we ask them questions, we say to them: How typical or how long have you been working on a computer that sort of stuff, that is the easier one and we ask them a questions and they still have the actual experience.

Interviewer: And once you have identified your critical risks, how would you then analyze this, do you like rank them give them some sort of risk value or

Interviewee: We've got basically a high, medium or low risk sort of classification

Interviewer: Is it automatic?

Interviewee: JA, so we would highlight it as the lowest or highest and then analyze it and then you've got a central point where people outside of the project can actually not manage it but they will look at it and make sure that the movement on the risk levels. So they would, they would look at them and discuss it at meetings and new documents put into place and stuff like that. We look at them and say ok fine this is it and from there they create long projects, study this long report at home and they will say ok fine this risk has been highlighted, there's movement on it, there's nothing done whatever the.......

Interviewer: what's the level of the awareness of the risk management, are the people on the ground aware of it or is it thoroughly integrated into every branch as well.

Interviewee: Because we use, we've got levels of, obviously employees and that all of our employees are actually at the clients. So they are face to face with the clients, so they have to be able to say ok there's the problem, identify it or this is the problem we are working currently and this is the way it should be going on, and now everybody is dead wrong with it. Sometimes

Interviewer: What I am basically trying to ascertain is what is the attitude towards this. I mean is it positive I mean is it seen as something that is going to benefit the Company as a whole at the end or is it like a......,

Interviewee: It is a very positive attitude, we try not to get, well, the one with the one negative thing is to get personal about it. That is the one thing we, I think we manage to get away from it. It is not a personal attack towards anybody at any point in time. So that's how we tell the people to handle it so if they, if they do pick up on something, it's not a personal, I'm not attacking the person, it is a process that needs to be identified, it is a process that needs to change and obviously we going to add value to the client eventually and that is the whole thing and we also sell it to the client in that fashion and when you do, I mean, obviously, when you go to the client you analyze the given the risk document, and you are actually telling me your processes are not working for you, so obviously on like, JA, but that's not true, this is my business and we then handle the situation to an extent. This is not a personal attack, we would, the only thing we want to do here is to help you and make this project a success and that's the only thing that we are there for. I mean

Interviewer: What kind of like policy or whether you people have in place to prevent people from seeing it as a personal attack. Do you have any way of doing that or you just handle it?

Interviewee: I think it is communication, if you don't communicate it to the client all the time and create a relationship with him on a like not a personal level as per say but a business personal level if I can put it that way so that he understands you and make, and it's communication, just tell him, listen, I have identified issues but it's not, I'm not pulling down your business or anything, it's just out of maybe experience, out of best practices, out of whatever it is. You can actually highlight it and say this and this can be a problem.

Interviewer: So you can see it as a positive side or a negative. Can you just tell me what optimum tools that you have used for Risk Management, a name

Interviewee: If you asked about Technology I would say *precisely.

Interviewer: What's the process, obviously you said you did documentation so you did capture the process afterwards and then you use that for your next project, so that you could see what went wrong there. if you do that kind of experience capturing

Interviewer: So overall what would you say is the impact of risk management in your project? How would you

Interviewee: If it is not there you have got a problem, No, it is, it's crucial,

Interviewer: It's not in negative, but actually pushing you forward.

Interviewee: Yes, definitely, definitely

Interviewer: Streamlining?

Interviewee: It's streamlining and moving forward then.

Interviewer: But what would you say like other negative side of Risk Management, that actually anything negative about it? Would you feel....

Interviewee: The only negative part of it is managing the client communicating and in a way that he still understands it we are helping him, but that in the communication

Interviewer: And in such a technique that you are currently using within your risk management processing is there anything that needs more better attention or more research to be done on it, that kind of thing.

Interviewee: Yes there's always, always room for improvement and the other way I think is we actually realize that is what is management experience. The basic rules are fine but as you go along and as you get new clients you realize there's something you need to add so you continuously it's a, it's a living document as per say so...

Interviewer: So you keep feeding it back into your techniques and you keep improving it. So what I'm trying to say is as a whole you know the risk management process is like, do you feel like IT education, needs better techniques that you currently have or that kind of a thing

Interviewee: Well, like what, I don't know maybe there are techniques that I don't know of. If that there is, then maybe yes. But currently the process and the people actually does quite a bit for us.

B4. Interview with Company D

Interviewer: How do you feel about risk management ? Ok, I just got to ask you some few personal questions, because I'm doing a study within a context, what is your experience in project management, how many years do you have in project management ?

Interviewee: 13 years

Interviewer: And risk management?

Interviewee: Risk as per say, analysis of risk, I would say 5 years

Interviewer: This based on software development, not IT risk software development.

Interviewee: Ok, in engineering, we did risk analysis for a systems so, that was probably round about 5 years of that, prior to this probably about '84 to about '89?.

Interviewer: How much of training did you receive in risk management?

Interviewee: Well, let's put it this way I've got 13 years in total the follow to that in risk management, that's about 5 years...

Interviewer: ok, so basically me just ask you about the risk management in theinfrastructure, what is the level of awareness of risk management?

Interviewee: The awareness is growing, certain people is finally understanding that, they need to focus more, and clarifying exactly what client's needs are, understanding what potential problems still can crop up, but I think it's probably not, it was not really enough.

Interviewer: And are the attitudes towards this risk, risk seeking or risk adverse or risk neutral

Interviewee: I would say, it depends on the group, in some cases, we do very large project for the police? And there the people are very scared of that because they understand that it is a very stable environment and we've got to ensure its stability, our guys always goes there, on the other hand we are not going (redevelopment?) feeling as much more towards to ....but I would like that

Interviewer: So risk management, is not a formal process?

Interviewee: Yes, so they get in the risk things at the end they manage out of it ....by adding more resources, which is not the right way, so it depends on, in the very commercial environment you've got ? If we balanced investment, on the other hand, you've got a very stable company, you've got to support the risk... So that's as a spectrum but then what I am also seeing is, because it is a very commercial side is we have to bring it back as neutral as possible

Interviewer: Bring it back, what do you mean?

Interviewee: You can't take to many risks, cause then it...

Interviewer: And, has risk management seen something legitimate?

Interviewee: Yes

Interviewer: Do you have a policy about it? Like a commitment to risk?

Interviewee: Not a formal policy, no, we are writing up our procedures for systems development.

Interviewer: What kind of techniques and methods do you use for this identification and risk analysis

Interviewee: Basically, we try to understand the problems are... potential problems are...

Interviewer: So you don't plan to combine special method, just by brainstorming you decide which is more important.

Interviewee: We don't do....

Interviewer: And why not, do you feel like it is not worth the effort not to go that route ?

Interviewee: Frankly, I think it is probably necessary, it's against it not having gone to that level of identification or something like that

Interviewer: Then do you think it like it is hindering your programme in a sense of, do you think all this formal techniques is just going to slow down the process....

Interviewee: Perhaps this is the wrong thing to say, I think I don't really see how one could apply it but in the past six months or so I still need to see how we can actually use those methods?

Interviewer: And you would feel that it would be beneficial Interviewee: Absolutely, yes

Interviewer: Do you have any sort of mechanism to capture your process once you've completed it so that you could use that, you know, the types of problems you've encountered in this project and use it in the next project as a guideline, like a kind of documentation that kind of thing

Interviewee: Difficult to decide, obviously, like problems and develop understanding what the reason why other ....So it's more, it is not necessary .....here's your problem? and probably look in another way and say here is a better solution

Interviewer: So you use it in the future

Interviewee: Yes...

Interviewer: And what kind of skills and competence exists within risk management? How competent are people about it? What kind of skills ....

Interviewee: I think it is probably, people have good project management skills and some of them have just skills from that type of background...One of our guys have been in Microsoft .... that we got some guidelines and we will tailor that to our own environment....

Interviewer: How much of resources and terms of time do you spend actually analyzing the risks (????? 2% of the time)

Interviewee: I would say that is part of your initial planning and I doubt probably about 20% on client's needs?

Interviewer: In terms of planning itself, not the whole software development

Interviewee: Planning, for like in I would say that the methods that we use, understands what people should do and what requirements are, I would say it is more advanced than what we had five years ago we have already gone a long way towards eliminating a lot of risks, in terms of understanding the client's requirement, We just started off by just (mumbles) expecting the client to tell us what he wants, now we are saying no that doesn't work,.... and using that as a starting so by doing that, risk in terms of misunderstanding what people require. And that is the problem when you are used to write to them ....and expect certain things and then we have ... misunderstanding... it's a language risk or a level of understanding risk ..... or technology risk ..... the environment is very unstable

Interviewer: Even though, I would classify your risk management process very informal, even though it is informal, but what is being informal, have you seen doing software development without risk management and with risk management, how do you feel about, I get the generally sense, it is positive

Interviewee: Yes, it is positive, I think there is a need for a better awareness a better, more formalize structure to that, and what we need do is find an structure keeping a balance, knowing about it, you got to...

Interviewer: What would you see is like a critical problem in risk management, that you feel that need some improvement in area of it, needs more clarification

Interviewee: I think it is a case of finding the risk? it's got to be a play of between money spend the risk and the actual risk...I am not sure how one would quantify it....

B5. Interview with Company E

Interviewer: What is your position at your organization ?

Interviewee: I am the technical manager and I am a developer

Interviewer: How many years of experience do you have in project management ?

Interviewee: I have been involve in software development for 7 years, but in project managements for 2 years.

Interviewer: What is the nature of software developments ?

Interviewee: We built Defence Systems....

Interviewer: How large is your organization

Interviewee: About 40 ...

Interviewer: What is the level of awareness of risk management

Interviewee: Well I learnt about at university but not to the extent that I can apply it in...we don't focus on that. What do you mean by risks?

Interviewer: I mean schedule risks, budget risks , personnel risks....Those kind of issues...

Interviewee: Look, deadlines affects every projects, so we are aware of it. Depending on the project, we may cater for budget risks...We don't really have a formal process because we are a very small company.

Interviewer: What kind attitude do people have towards risk, are they risk taking, risk averse ....

Interviewee : As I said, we are a small company, so we have take on projects with risks, otherwise...

Interviewer: you're not in business ...

Interviewee: right...

Interviewer: Is the discussion of risks encouraged...

Interviewee: Yes

Interviewer: How so...

Interviewee: What do you mean ?

Interviewer: Do you brainstorm...

Interviewee: When we plan we take into account, like the resources we will need and the personnel needed...

Interviewer: So you don't have a policy...

Interviewee: No

Interviewer: Do you capture the process once you finished, so that you can learn from that experience ?

Interviewee: Yes we do

Interviewer: From what I gather, you don't not follow risk management at all? What is rationale for that ?

Interviewee: Well we are a small company. With IT, I believe in flexibility, we cannot have this rigid structures because creativity will be stifled. We have have very flexible hours at work...

B6. Interview with Company F

Interviewer: What is your position ?

Interviewee: I am a cost consultant, or in this point in time the manager running the division but as I said I am a cost consultant. I don't if I must expand on what we really do it depends on the structure of your question. What we actually do, we are in a very unique environment is optimize risk not only in software development, we talking sourcing the product, we analyze risk and develop risk profiles, from looking at various costing strategies in other words we build we do it in a pro project such way we develop cost model that address risk issues that may effect us in . We look at all kinds of things whether it is qualitative quantitative it is a very wide scope

Interviewer : How much experience do you have in project management ?

Interviewee: Lots, I am here now for about 3 years. I dealt with contracts with a total value of 5 billion rand. I must have done about 30 contracts and on top that evaluations of which 90 % was foreign based.

Interviewer: And in software risk management ?

Interviewee: In software risk management as such I have a lot of exposure in terms of costing perspective I was a project manager in developing SAP and implementing but then I not involved in implementing the system but I was responsible in analyzing the process and costing the process. I am not sitting doing nitty gritty programing. We form an evaluation group... single committees... there is a lot of people will be a guy from software a guy from technical and finance what functionality should be cost money value. Lets say, we identify a risk to the programme like the project manager does not have enough skill then it is my obligation that we get one and what would it cost that type of thing.

Interviewer: How do you promote a culture a risk active culture.

Interviewee: That is difficult question I am not involved in that. If we identify specific needs we address it by they put certain structures to address those issues. If we implement a new software system, it may have elements in it that may result in resistance to change or they maybe cultural difficulties we have doing things a specific way for years and suddenly we introduce a new way of thinking then we put committees in place to address these issues. My experience there is very limited I cannot answer that.

Interviewer: How do you go about identifying risks ?

Interviewee: We have specific process that we follow, the process is very wide and the scope is very wide and we do it by following project analysis, in other words let's say for example we buying satellite equipment we will sit down and what makes the whole process, what we actually buying what is the requirements how does it fit in with our current needs, what the critical requirements and sources in something in this..What do we want to do with it and at the end of the day we do a scope analysis and we do a overlap analysis in other words what do we have, where are we going and what the risks getting there and how we can address those risks so to answer your question we do a product understanding and that is a very wide concept and would take me days to explain but methodology various from project to project because the merits of every project is different. But you very reliant on the technical expertise that you get from your co workers sitting on this committee

Interviewer: So don't use a checklist or automated tool...

Interviewee: No there is nothing like that cos you must remember there is not generic component embedded in what we do.

Interviewer: It's a brainstorming kind of thing...

Interviewee: Yes very much

Interviewer: Once you have identified your risks you have a specific way of quantifying it

Interviewee: No, we don't, we have methodologies recording it and as far as possible we address every one of those . In other words we make sure, it doesn't have matter how tiny the risk is we address it as far as possible. So we don't have a scoring mechanism to say that one is more important than that one. We have specific issues where we apply logic we say for example this risk is top a priority, we must solve this cos it might have a massive cost implication. But then the next one is not that important cos it may have a 0.001 % on the cost but at the end of the day we try to cover both. You must remember that we have specific procedures in place that allows us to look at to optimize the cost to the customer as far as possible.

Interviewer: So you don't have a policy about risk management ?

Interviewee: If you talk of risk management, in the risk management aspect we procedures in place focusing on risk management where you split as define it as insurance.Every project is different and there you apply your own logic

Interviewer: What kind of skills and competencies do you have in risk management ?

Interviewer: I wasn't educated in software development, we bring cost management and risk assessment risk management we bring it together as one. We apply those portfolios in areas of industry that we are involved in. The experience that I have in software development, programing is very limited but I have the basics.

Interviewer: Once you finished a project, do you document it...

Interviewee: Yes, we do a benefit tracking analysis. We have a about 30 criteria and I not at liberty to disclosure those criteria. It is simple stuff, it comes do the principle where was I what did I encounter where I am now what did I do about the problems but the processes are so precise that we can cost the risks that we experiences. We say we encountered a problem X and the cost that problem was so many million rand.

Interviewer: So how do you place a monetary value on it

Interviewee: We apply logic. For example, if you encounter that a specific skill was not catered for then you have to source something out the market. You identified the risk, then you addressed it by sourcing someone out. The labor cost implication was X but then you have got to analyze the value that person added by . By doing so you can play off two against each other...

Interviewee: So do you this after

Interviewee: No we identify risk before, do the benefit tracking as we go on

Interviewer: What is the impact of risk management

Interviewer: Very positive. Risk management the procedures that we have in the market is very new. The value we add is billions. You must remember the process of determining if we adding value or not is happening continuously as you go on. At the end of the day the purpose is to see if you are still on track. The nice part of this is that we have committees and each reporting to each other. So that every committee is making sure that everyone is on the line by doing that you track as you doing.

Interviewer : what are problems with risk management?

Interviewee: The biggest hiccup I can find, I speak from a personal capacity, the biggest issue of concern I have, there is massive of shortage in the market. The academic institutions train the people on cost accounting financial accounting marketing and strategic planning and project management but they never teach people how to bring it all together and how to interpret what they need to do with it. The practical side of things is so wide it is difficult to teach the people what is happening out there. The biggest problem in risk management is that there is nothing out that there that teaches the people ...to give the broader picture

B7.interview with Company G

Interviewer: How many years experience in do you have ?

Interviewee: 23 years

Interviewer: In Risk Management ?

Interviewee: About the same. It is all combined with risk management.

Interviewer: Have you received training in risk management ?

Interviewee: Yes

Interviewer: At what level?

Interviewee: Through international marketing exposure. I started very young.

Interviewer: In terms of your company's infrastructure. What is the level of awareness of risk management ?

Interviewee: We try to calculate it. We do every single step on our way to do proper very precise calculation before we enter into risky lines that cannot be tolerable to us. So of course, I am trying to safeguard my investment. I think that, that is the common approach to any risk taking. We are also by the same token very vulnerable. There are certain situations that dictates so, we have to accept them. So the chances for risks.... I try to eliminate them in my company , if I can, but of course there are ...

Interviewer: How do you identify risks?

Interviewee: I identify my risk via proper backup system, my backup is that before I attempt any undertaking I go about my own infrastructure...can my infrastructure deliver the commitment that is resultant out that transaction or not. Minimizing any high risks is my focus. How I go about it, it is a gradual procedure where I undermine myself by not taking high risk as a principle but of course there is many ways...the most is, is to prevent any situation that you cannot foresee. . So you study it thoroughly...

Interviewer: How do you quantify risks ?

Interviewee: By or through ....taking down my mandates to my staff to the absolute minimum. It's like a ship in the middle of the ocean in high seas. You cannot delegate at this point because the risks are too high. Of course through my experience, I can judge what course to steer and that is factually the real approach . But in general, we have to operate our companies with system and our system has implemented because without infrastructure there is no backbone so risks are there but minimizing them is our commitment.

Interviewer: I am not familiar with the technique you are describing...could you elaborate on it?

Interviewee: My techniques is an international technique. I originate myself from an international background and that how we have performed in multi infrastructure, multi social multi levels markets. So it of course open to everybody. South Africa has changed slightly but that was valid strategy of the past. And I think it is still valid

Interviewer: How do facilitate a risk aware culture?

Interviewee: We give it every single every single employee a job description, and I ... I admit not all the contents of single every job description is not quite obviously definite. So though weekly meetings, through individual bilateral meetings with staff. That when we try to highlight any possible misunderstandings that can result in any risk over and above the authorization given to that single individual member. We have a solid communication in risk in place that is most important to us.

Interviewer: So risk management is seen as a legitimate activity. So do you have a policy on it ?

Interviewee: Well It is anchored within the companies policy

Interviewer: In terms of skills, do you send your personnel for training in risk management ?

Interviewee: Management as such, not necessarily, but technical, yes. Management, we provide in house

Interviewer: Do capture the process when a project is complete ?

Interviewee: Any project is a learning curve. No matter how large/small, the skill actually is not the norm. The norm, is the facts that are going to be assist during and after the activation of any programme ,that we take into our support lesson assessment. We base on its grounds any undertaking in the future.

Interviewer: What is the impact of risk management ?

Interviewee: It is controllable

Interviewer: Do see any improvement in the process of risk management ?

Interviewee: Yes, I will just use a phase "Trust is very in place but control is better" and that my philosophy.

Interviewer: Think about the techniques use in risk management...do you think that any of those techniques need clarification?

Interviewer: Frankly in our time, our time is very brief. What the official bodies or associations are undertaking out there, I think there is very poor coordination between activities various companies. I think our environment in South Africa in general is very poor in its platform of communication.

APPENDIX C

ANALYSIS OF QUANTITATIVE DATA

C1. Determining the Risk Performance Measure

The quantitative survey was used to determine the how effectively risks were being handled. The analysis used here, involved calculating the risk performance measure for each participant. This was done by "summing" up the responses, where the final score for each respondent on the scale is the sum of their ratings for all items. The responses to each question ranged from "Hardly Ever " to "Almost Always". Each response was rated by using a Likert scale from one to five. For example "Hardly Ever" was rated as one and "Almost Always" was rated as five. Some of the questions depicted a positive outcome while others depicted a negative outcome. Those questions with a negative outcome, i.e. those items that are reversed in meaning from the overall direction of the scale, had to be reversed. This implied that for those particular questions, the respondent's answer had to be reversed where:

- an item rated one by the respondent was changed to five.
- an item rated as two by the respondent was changed to four.
- an item rated as three by the respondent remained the same.
- an item rated as four by the respondent was changed to two.
- an item rated as five by the respondent was changed to one.

For example regarding "Software and hardware capabilities are estimated incorrectly", the respondent's rating had to be reversed. Table 6.1 shows the responses of the participants, together with the questions where the respondent's rating was reversed (the shaded portions).

The risk performance rate was determined by the following formula:

Risk Performance Measure = (Sum of Scores/(Number of Questions × highest score)) × 100.
where
the number questions = 20
highest score on the Likert scale = 5

The risk performance rate determines how well the participant handles risk overall. For example, company A successfully handles 90% of all risks encountered. As this is an interpretive approach, which involves considering the context of the situation, the qualitative data is analyzed first and then compared against the quantitative data. The quantitative data will only be looked at with respect to the qualitative data.

Table C1: Showing the responses for each company coded A through G where the shaded portions are those questions where the ratings had to be reversed.

SURVEY QUESTIONS	A	B	C	D	E	F	G	%	Mean	Mode	Median
Your project has considerable problems due to personnel shortfalls	4	3	3	2	4	3	1	57	2.9	3	3
Your project is completed according to the timetable	5	4	4	4	2	4	3	74	3.7	4	4
Resource consumption reaches its top as you approach your project deadline	4	2	2	3	2	3	1	49	2.4	2	2
Actual project costs and estimated costs are nearly equal in your projects	5	2	4	4	2	4	4	71	3.6	4	4
Your project is cancelled before completed it	5	5	5	5	5	4	5	97	4.9	5	5
A failure to estimate project size interferes considerably with implementation	5	2	3	5	2	2	2	60	3	2	2
Demand of personnel is estimated correctly in your project	5	2	3	4	4	4	4	74	3.7	4	4
Time consumption of your project is constant	4	3	4	2	2	2	3	57	2.9	2	3
Your personnel''s expertise in methods, software and equipment is insufficient	5	4	3	4	4	3	5	80	4	4	4
The complexity of your project and its effect are easy to manage	5	2	4	4	2	2	5	69	3.4	2	4
Developed software functions and properties meet user''s needs	4	4	5	4	4	4	5	86	4.3	4	4
Developed software includes complex, but only marginally useful properties	5	4	4	4	4	4	2	77	3.9	4	4
Software requirements are continuously changed	4	2	1	2	2	2	1	40	2	2	2
Your project timetable is changed continually	4	2	4	3	2	3	2	57	2.9	2	3
Users are not satisfied with the implemented user interface	5	4	4	4	4	3	5	83	4.1	4	4
Externally purchased components and equipment meet your expectations in your project	4	3	4	3	2	4	1	60	3	4	3
You have unrealistic expectations of the project members skills	5	2	3	4	3	3	4	69	3.4	3	3
Performance requirements(response time, computing efficiency) are estimated incorrectly	4	3	4	4	4	4	5	80	4	4	4
Subcontracted tasks in the project are performed as expected	4	1	4	4	0	3	5	60	3	4	4
Software and hardware capabilities are estimated incorrectly	4	4	2	4	4	2	5	71	3.6	4	4
RISK PERFORMANCE RATE	90	58	70	73	58	63	68				

C2. Analysis of Quantitative Data in Risks Categories

The performance measure for each category was determined by summing the scores in the factors related to the category for each participant. The total score was converted to a percentage by:

Percentage = (Total Score/ (5 × number of factors per category)) × 100

For example for Scheduling and Timing risk(Table C2) there are 6 factors.

Table C2: Showing the responses for schedule risk for each company

SCHEDULING AND TIMING RISK	A	B	C	D	E	F	G
Your project is completed according to the timetable	5	4	4	4	2	4	3
Actual project costs and estimated costs are nearly equal in your projects	5	2	4	4	2	4	4
A failure to estimate project size interferes considerably with implementation	5	2	3	5	2	2	2
Demand of personnel is estimated correctly in your project	5	2	3	4	4	4	4
The complexity of your project and its effect are easy to manage	5	2	4	4	2	2	5
Your project timetable is changed continually	4	2	4	3	2	3	2
SUM	29	14	22	24	14	19	20
PERCENTAGE	97	47	73	80	47	63	67

Table C3: Showing the responses for system functionality risk for each company

SYSTEM FUNCTIONALITY	A	B	C	D	E	F	G
Demand of personnel is estimated correctly in your project	5	2	3	4	4	4	4
Developed software functions and properties meet user's needs	4	4	5	4	4	4	5
Users are not satisfied with the implemented user interface	5	4	4	4	4	3	5
Software and hardware capabilities are estimated incorrectly	4	4	2	4	4	2	5
SUM	18	14	14	16	16	13	19
PERCENTAGE	90	70	70	80	80	65	95

Table C4: Showing the responses for subcontracted risk for each company

SUBCONTRACTED TASKS	A	B	C	D	E	F	G
Demand of personnel is estimated correctly in your project	5	2	3	4	4	4	4
Externally purchased components and equipment meet your expectations in your project	4	3	4	3	2	4	1
Subcontracted tasks in the project are performed as expected	4	1	4	4	0	3	5
SUM	13	6	11	11	6	11	10
PERCENTAGE	87	40	73	73	40	73	67

Table C5: Showing the responses for requirements risk for each company

REQUIREMENTS MANAGEMENT	A	B	C	D	E	F	G
Your project timetable is changed continually	4	2	4	3	2	3	2
Time consumption of your project is constant	4	3	4	2	2	2	3
Developed software includes complex, but only marginally useful properties	5	4	4	4	4	4	2
Software requirements are continuously changed	4	2	1	2	2	2	1
SUM	17	11	13	11	10	11	8
PERCENTAGE	85	55	65	55	50	55	40

Table C6: Showing the responses for resource risk for each company

RESOURCES MANAGEMENT	A	B	C	D	E	F	G
Software and hardware capabilities are estimated incorrectly	4	4	2	4	4	2	5
The complexity of your project and its effect are easy to manage	5	2	4	4	2	2	5
Resource consumption reaches its top as you approach your project deadline	4	2	2	3	2	3	1
Performance requirements(response time, computing efficiency) are estimated incorrectly	4	3	4	4	4	4	5
SUM	17	11	12	15	12	11	16
PERCENTAGE	85	55	60	75	60	55	80

Table C7: Showing the responses for personnel risk for each company

PERSONNEL MANAGEMENT	A	B	C	D	E	F	G
Performance requirements(response time, computing efficiency) are estimated incorrectly	4	3	4	4	4	4	5
Time consumption of your project is constant	4	3	4	2	2	2	3
Your project has considerable problems due to personnel shortfalls	4	3	3	2	4	3	1
Your personnel's expertise in methods, software and equipment is insufficient	5	4	3	4	4	3	5
You have unrealistic expectations of the project members skills	5	2	3	4	3	3	4
SUM	22	15	17	16	17	15	18
PERCENTAGE	88	60	68	64	68	60	72

APPENDIX D

CONTRIBUTORS TO RESEARCH

D1. List of Companies and Persons Who Participated in Research:

Fareed Wahab of ColorVision
Dr Vermeulen of Africon
Johan Van Staden of Virtual Health
Arno Filmalter of Syllogic
Renee Du Plooy of Resolution Software
Pierre Geldenhuys of Telkom
Hanno Botha of Qmuzik Hanno