PROJECT PORTFOLIO MANAGEMENT AS A TOOL FOR VALUE CREATION AND RISK MANAGEMENT IN THE IT OUTSOURCING VENDOR

By

Susanne Halstead

A thesis submitted in partial fulfillment of the requirements for the degree of

MASTER OF BUSINESS ADMINISTRATION

in

GENERAL

UNIVERSITY OF PUERTO RICO
MAYAGÜEZ CAMPUS

May, 2010

Approved by:

Mario Córdova, Ph.D. Member, Graduate Committee	Date
Miguel Seguí, LLB Member, Graduate Committee	Date
Rosario Ortiz, Ph.D. President, Graduate Committee	Date
Mercedes Ferrer, Ph.D. Representative of Graduate Studies	Date
Rosario Ortiz, Ph.D. Associate Dean for Research and Graduate Affairs	Date

Abstract of Thesis Presented to the Graduate School
of the University of Puerto Rico in Partial Fulfillment of the
Requirements for the Degree of Master of Business Administration

PROJECT PORTFOLIO MANAGEMENT AS A TOOL FOR VALUE CREATION AND RISK MANAGEMENT IN THE IT OUTSOURCING VENDOR

By

Susanne Halstead

May 2010

Chair: Rosario Ortiz
Department: Business Administration

Taking the vendor perspective of IT outsourcing, Project Portfolio Management is described as a tool for value creation and risk management. Superior processes and project management are at the core of the IT outsourcing vendor's value proposition. The work describes key knowledge areas that make up the theoretical foundation for the definition of management and production processes in the IT outsourcing vendor. Because estimates play a central role in the IT outsourcing vendor's business, and here especially the technique of expert estimation, this work investigates this estimation technique in more detail and in the context of Puerto Rico. We investigate the influence of incomplete expertise in terms of domain knowledge or experience with the technology on the reliability of estimates. Finally, it is shown how Project Portfolio Management can be used to integrate an IT outsourcing vendor's strategic goals and create the value proposition to clients.

Resumen de Tesis Presentada a Escuela Graduada
de la Universidad de Puerto Rico como requisito parcial de los
Requerimientos para el grado de Maestría en Administración de Empresas

GERENCIA DE PORTAFOLIOS DE PROYECTOS COMO HERRAMIENTA PARA LA CREACIÓN DE VALOR EN UN PROVEEDOR DE SERVICIOS DE INFORMÁTICA

Por

Susanne Halstead

Mayo 2010

Consejera: Rosario Ortiz
Departamento: Administración de Empresas

Describimos como se puede usar las técnicas del Manejo de Portafolios de Proyectos para crear valor y manejar riesgos desde le punto de vista de un proveedor de servicios de informática. Parte fundamental del ofrecimiento de un proveedor de servicios de informática es la ejecución de proyectos y el manejo de procesos superiores. Presentamos varios estándares y marcos teóricos en los cuales se basan estos procesos. Estimados y técnicas para estimar forman parte central de la metología de un proveedor de servicios de informática. En particular la consulta de expertos se destaca como herramienta más utilizada para estimar. Por esto, investigamos esta técnica de estimar más a fondo y en el contexto de Puerto Rico. Concluyendo, explicamos como se puede usar el Manejo de Portafolio de Proyectos para integrar las metologías y lograr los objetivos estratégicos inherentes al modelo de negocios de los proveedores de servicios de informática.

Dedicated to all children, spouses, family and friends of IT professionals that had to put up with long nights and working weekends due to faulty estimates and overly optimistic project plans.

ACKNOWLEDGMENTS

A work of investigation undergoes various transformations between its delimiters of initial conception of the idea to its final paper form. These transformations come about through discoveries made on the way. I would like to acknowledge the contribution of those who pointed me to these transforming discoveries.

I would like to thank the president of my graduate committee, Rosario de los A. Ortiz, for the ongoing support throughout all this time and the mathematical expertise. Furthermore, I thank the members of my graduate committee, Mario Córdova and Miguel Seguí for helping me finding a workable approach to my ideas.

I also would like to express special thanks to Magne Jørgensen of Simula Labs, Oslo, undoubtly one of the leading investigators in the field, for his comments on my investigation on expert estimation.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CMM	Capabilities Maturity Model.
CMMI	Capabilities Maturity Model Integrated.
COCOMO	Constructive Cost Model.
CPM	Critical Path Method.
DARPA	Defense Advanced Research Projects Agency.
IRR	Internal Rate of Return.
ISO	International Standards Organization.
IT	Information Technology.
MPT	Modern Portfolio Theory.
PDM	Precedence Diagram Method.
PERT	Project Evaluation Review Technique.
PM	Project Management.
PMBoK	Project Management Body of Knowledge.
PMI	Project Management Institute.
PMP	Project Management Professional.
PPM	Project Portfolio Management.
SEI	Software Engineering Institute.
SLA	Service Level Agreement.
SQE	Software Quality Engineer.
WBS	Work Breakdown Structure.

CHAPTER 1
INTRODUCTION

The purpose of this investigation is to show that Project Portfolio Management is a useful tool for the management of an IT outsourcing provider. Very little work in the field of Project Portfolio Management has been done from the perspective of an IT outsourcing vendor. We will show, that Project Portfolio Management integrates and supports functions needed to create value and manage risk in IT outsourcing vendors.

1.1 Motivation and Context

The focus of this work is on the vendor perspective of the IT outsourcing relationship. Even though outsourcing has been investigated, most work in this arena focuses on either the client's decision to outsource or the contractual agreements between vendor and client. However, very little research has been done from the vendor's perspective [9]. According to a case study by Levina and Ross, IT outsourcing vendors are in a unique position to create mature processes and offer an excellent level of execution at competitive prices. They conclude that an IT outsourcing vendor "*can deliver value to its clients by developing a set of experience-based core competencies, which a) address client needs, b) exhibit complementarities, which result in efficient service delivery, and c) depend on the vendor's control over, and centralization of decision rights on a large number of projects from multiple clients" [9].*" Aspect c implies that IT outsourcing vendors can benefit from project portfolio management as an instrument to create their value proposition.

Being an IT outsourcing provider is a risky business. IT projects are known to have considerable risks associated to them. These can range from making the wrong product, making the right product too late, cost overrun, schedule overrun, low quality, to litigation or any combination of these [10]. In an outsourcing relationship, the investment risk, that is the risk of investing in the wrong project, remains largely with the client organization, the process risk, however, is partially or fully shifted to the outsourcing provider. In fact, shifting the process risk to a third party is one of the motivations for a client organizations to engage in outsourcing relationships [11]. Therefore, this work will specifically investigate tools and techniques available to IT outsourcing vendors for mitigating process risks in IT projects . There are various knowledge areas, methodologies and processes that have to be combined and adapted to the specific needs of the organization in order to form its production process. This work will give an overview of project management as a discipline, particularities of IT project management, and software risk management, as these knowledge areas are the foundation of what would make up an efficient production process in an IT outsourcing vendor. We will then show, that project portfolio management can be used as a catalyst and an integration tool to forge and develop a process that avoids and mitigates process risks.

Projects have become a way of life in many organizations [12, 13]. This holds especially true for the information technology (IT) sector [5]. Project Management is viewed as a discipline proper and as such has been in the focus of researches and practitioners for several decades now. The existence of various professional organizations, academic journals, and text books attests to this. Despite a certain maturity of the discipline and the existence of standards and professional certifications, it is evident, though, that the knowledge and practices of project management vary between industries, countries and organizations. Also, with the constant changes in the business environment, the requirements of project management and points of view towards it evolve constantly. Project Management is a core competency for IT outsourcing vendors.

Project Portfolio Management (PPM) takes a strategic, birds-eye view of the collective of projects undertaken in an organization. It focuses on 'doing the right projects' , in contrast to Project Management (PM), which concentrates on 'doing the project right'. Both skill sets are necessary. The organization's strategic plan is the guideline for PPM. A successful IT outsourcing provider needs to have a strategy. That strategy, at a minimum level, should address such issues as market positioning, technology focus and product development. Independent of what the specific strategy of an IT outsourcing provider is, creating and fine-tuning efficient, effective and reliable processes can be identified as a generic strategic goal applicable across the board. The importance of reliable processes is evident in any organization, however, because of their specific business model, the impact of the quality of software development processes on the bottom line is more direct in the IT outsourcing vendor.

The business model of an IT outsourcing provider for development projects at a macro level consists of three stages: proposal stage, implementation stage, and a warranty period. During proposal stage, the outsourcing provider performs a preliminary analysis of the project and provides a proposal document which includes a statement of work to be done, assumptions, design approaches and an estimate of resources, costs and a proposed schedule. If the client organization accepts this proposal, the outsourcing provider is contracted and carries out the required work. After the conclusion of the project and the implementation of the finished product, there is typically a handover to the client organization followed by a warranty period in which the outsourcing provider will have to respond to any errors in the product. The specific contractual agreements can be manifold, however, we can distinguish two basic types: cost reimbursable contracts and flat rate contracts. Flat rate contracts have become increasingly popular with client organizations, as they shift the process risk of cost overruns onto the outsourcing provider entirely. The estimates made in the proposal stage often become the terms of the contract. It is therefore crucial for these estimates to be as accurate as possible. Estimates that fall short of the actual cost of implementation eat away at the profit margin of flat rate contracts, estimates that are too high might prevent the organization from winning contracts. Faulty estimates put the compliance with schedules, budgets, quality requirements and the bottom line of a project at risk. Faulty estimates are problematic in any software development project. In an IT outsourcing provider, however, the link between faulty estimates and adverse effects on the bottom line is more direct than in companies that do in-house development and treat IT as a cost factor rather than a product. Yet, few companies devote resources and attention to developing a more reliable methodology. Taking a portfolio view of the company's projects can aid to improve on that area.

Beyond providing a literature review of the aforementioned areas, this work presents an empiric investigation of estimation risks as perceived by IT professionals in Puerto Rico for the estimation technique of expert consult. Casper Jones points out that among the most serious software risk factors are excessive schedule pressure, which can be a result of too optimistic estimates, management malpractice, and inaccurate cost estimates. [10]. Molokken and Jorgenson report in their review of Surveys on Software Effort Estimation that expert consultation is the by far most frequent method of estimation [14]. The quality of estimates derived from expert consultation is highly dependent on the experience of the personnel asked to provide the estimates and the information available to them at the time of giving the estimates.

1.2 Goals of this Work

The main goal of this work is to show the benefits of Project Portfolio Management for the creation of an IT outsourcing vendor's value proposition, with special focus on the aspect of lowering the process risk exposure. Lowering the client's risk exposure is a key aspect of an IT outsourcing provider's value proposition. The client's risk exposure is lowered through contractual agreements that shield the client from or offer some sort of compensation for schedule or budget overruns. From the IT outsourcing vendor's perspective, this implies that the organization needs to excel at controlling process risks. Thus, independent of their particular business strategy, the creation of a reliable process that controls and mitigates risks should always be a core strategy of an IT outsourcing provider. We present a literature review of the subject areas most relevant to developing reliable processes and process risk mitigation answering research question Q1.

Q1: What competencies are required in order to successfully manage individual projects and project portfolios in Information Technology?

Answering this question will illustrate the various knowledge areas that an IT outsourcing provider can draw from in order to forge their policies and processes. These knowledge areas and frameworks are applicable to all software projects, they are not specific to an IT outsourcing vendor. The following objectives guided the compilation of the literature review:

i **Give an overview of the current body of knowledge of project management**. A research conducted by M. Martinusuo and P. Lehton suggests that the successful management of single projects is a necessary, yet not sufficient factor for project portfolio success [15]. The management of single projects is the foundation for the management of project portfolios and thus overall organizational success.

ii **Explain the particularities of IT project management**. This section will show the domain specific considerations for IT project management.

iii **Present standards and innovations of risk management processes at single project base**. This exhibit will show some of the plentiful current efforts to widen the body of knowledge and improve risk assessment and management practices. It is important to understand the risks at the single item level well in order to build portfolios based on this information.

iv **Outline the current body of knowledge on project portfolio management and present the business rules most commonly used in project selection**. This will serve as an overview of the current de facto standards, practices and discussions in this area of knowledge and will be the foundation of our further considerations and investigation of portfolio management. We will also show, that taking a portfolio view might change the selection criteria used for projects. The selection rationale for projects is typically focused on expected return of investment. When taking a portfolio view of projects, this selection rationale might change, based on strategic considerations.

The existence of well defined and controlled processes and the reliability of estimates are interdependent. In order for estimates to have any validity, the organization's methodology needs to be sufficiently reliable and controlled. A core assumption in any estimation is that the work to be carried out follows the process assumed at the time of estimation. Therefore, developing competencies in estimation has to be integrated with developing reliable processes that underpin the estimation. At the same token, estimates have an impact on the scope and quality of the work to be done [16]. Aggressive schedules as imposed by commitment to too optimistic estimates are a serious risk factor for software development [10]. Estimates are crucial inputs to contracts and project plans. A project that is planned based on erroneous estimates is set up to fail. Expert estimation techniques are the most frequently used and most easily accessible tools for estimation. Therefore, it is important to recognize experts in this skill as such and to develop estimation skills as an organizational competency. Through research question Q2 and Q3 we investigate the reliability of expert estimates and learning effects that influence the process.

. **Q2: What is the impact of developing expert knowledge for software development task duration estimations?**

This investigation is done by means of literature review.

Q3: How do IT practitioners in Puerto Rico perceive the effects of lacking domain or technology experience on the reliability of estimates? What are the perceived effects of experience and learning on the reliability of expert estimates?

To investigate Q3, we conducted a survey among IT professionals in Puerto Rico, in order to analyze this question in the local context.
Expert consultation is by far the most frequently employed estimation technique; however, not always will an organization have the required expert knowledge available to them. Experts will frequently lack experience with the specific domain problem presented to them or might be inexperienced with the technology mix to be used. Yet in practice, many organizations attempt to use the same technique even in these cases. To investigate the impact of such practices, a survey amongst IT professionals in Puerto Rico was conducted.

Integrating the findings from the previous sections, we will describe how Project Portfolio Management can be used in order to create its value proposition by implementing reliable processes that mitigate process risks.

Q4: How can portfolio management be used to create the IT outsourcing vendors value proposition and lower internal risk?

PPM can be a powerful tool for performance tracking of a projectized organization and the selection of projects that would further advance the organization's methodology. Therefore the objective is to show how:

-PPM can be employed to assure that projects that allow to further develop organizational competencies are chosen.

-PPM can enforce a standardized methodology of project management and definition of metrics and the collection of measurements.

-Historic data collected can be input to quantitative PPM, allowing a scientific base to PPM within the organization.

Very few organizations are in the position of having appropriate metrics defined and collecting the related measure across all projects conducted by the organization. PPM can drive the collection of such data. This data then serves as input to quantitative management of project portfolios.

1.3 Structure of this Work

The structure of this document follows the order of the research questions. Chapters 2 though 5 cover research question Q1 and present a literature review of knowledge areas relevant for the design of reliable software development processes: Chapter 2 focuses on project management, chapter 3 on the particularities of software project management. Chapter 4 introduces concepts of project portfolio management. Included in these considerations is an overview of frequently used project selection techniques. Chapter 5 presents a literature overview on the subject of project risk management.

Chapter 6 addresses research questions Q2 and Q3. It contains a literature review on investigations about expert judgement estimation techniques for software projects and explains the design and the results of the survey conducted.

Chapter 7 then covers research question Q4, explaining the core components of an

IT outsourcing vendor's value proposition. It then summarizes how Project Portfolio Management can be used as a tool to create and develop them.

CHAPTER 2
PROJECT MANAGEMENT

Q1: What competencies are required in order to successfully manage individual projects and project portfolios in Information Technology?

This chapter will present project management as a discipline and give a broad overview of the history and current body of knowledge of project management. Knowledge of this subject areas is necessary in order to put the investigations and results presented in the later chapters into perspective and understand their context.

2.1 Project Management as a Discipline

Project management has become a discipline of its own right. Many organizations have structured themselves based in projects. This development is often referred to as *projectization* of organizations.

Practitioners' organizations, such as the *Project Management Institute*, which publishes the *Guide to the Project Management Body of Knowledge* (PMBOK Guide), are an influential driving forces of the formalization of project management into a discipline. The PMBOK Guide is the defacto standard of project management in the USA and is also used in other countries. In addition, it is the theoretical basis of the Project Management Professional certification, which has become an important professional distinction for project managers. The existence of such certification has caused a broad dissemination of the PMBOK Guide and therefore of the practices described therein in organizations.

In academia, there is a large community of researchers that contribute to the investigation and development of project management. Among the most important publications are the International Journal of Project Management and the Project Management Journal.

In the following sections we will give a brief overview of the development of this discipline and the current body of knowledge as is disseminated amongst practitioners.

2.1.1 Historical Overview of Project Management

In his paper *'A Short History of Modern Project Management'* Alan Stretton draws a broad strokes picture of the development of project management in the decades of the 1950s through the early 1990s [17]. The development of project management reaches from initial efforts of controlling and planning schedules to the creation of a vast and integrated methodology.

The origins of modern project management can be traced back to the late 1940s, and early 1950s, where it developed in a limited number of industries, namely construction and the defense industry [17, 18]. Bechtel, a large U.S. construction company founded in 1898 [19], started using the term *Project Manager* in the beginning of the 1950s. Bechtel moved away from their traditional approach of having the project engineer also run the project in earlier phases and then have the construction manager run the project in later stages, to recognizing the role of a project manager as an individual with responsibility for the success of the project throughout its entire life [17]. The main technical achievement during this period was in the field of planning and controlling project times. Three methodologies, Project Evaluation Review Technique (PERT), Precedence Diagram Method (PDM) aka 'Activity on Node Networks', and Critical Path Method (CPM) aka 'Activity on Branch Networks' were developed during this decade. CPM was developed by the Integrated Engineering Control group and Dupont. PDM was developed by Civil Engineering Department of Standford University. PDM and CPM focus on activities and logical relationships and sequences of activities. They use deterministic time estimates for activities and serve as a measuring tool to verify if the project is on track. PERT, which was developed by the US Navy under the collaboration of the consulting firm Booz, Allen and Hamilton, focuses on project control by the definition of project milestones and uses probabilistic time estimates to determine how likely a project is to meet a certain time line [13, 17]. Even though in the 1950s these were competing methods, nowadays these methods are summarized under the umbrella name of *network planning techniques* and are viewed by some authors as almost completely interchangeable [13].

During the 1960s, techniques for project cost management and integrated with them project resource management techniques were developed as extensions to time control techniques provided by the network planning techniques. The most prominent one of them is PERT/COST. The techniques of project management where still limited to the construction industry and the defense industry. During this decade, two professional project management bodies were founded: (1) the International Project Management Association (IPMA-formally known as *INTERNET*) in Europe in 1956 and (2) the Project Management Institute (PMI) in the U.S.A. in 1969.

During the 1970s project management spread to a wide variety of industries. There were several further techniques and tools developed, including the work breakdown structure (WBS), which dissects deliverables into smaller and smaller and thus more manageable subdeliverables, and the Organizational Breakdown Structure (OBS) which integrates these deliverables with their responsible units. More and more organization conducted a transformation to matrix organizations. Project management was increasingly viewed as its own discipline and profession.

The 1980s were marked by an effort to integrate the experiences gained in the different industries into more generally applicable principles and practices. The most prominent effort in this area was the development of the Project Management Body of Knowledge (PMBOK). The PMBOK is one of the most prominent efforts to present project management as an integrated and structured discipline rather than a collection of best practices. The first version of the PMBOK was published in 1986. The PMBOK covered the subject areas of time, cost, scope and quality management. The next revision of the PMBOK in 1986 included project risk management as a fifth dimension. Currently there are eight dimensions included in the PMBOK after the inclusion of human resource management, procurement management and integration management. The 1980s also marked a shift to the concentration on the front-end phases of project, that is a shift to higher emphasis on analysis and planning instead of focusing the interest on the execution and implementation phase. Project management was also recognized as a good tool for change management within organizations. Furthermore, the PMI launched the Project Management Professional (PMP) certification in 1983, further cementing the standing of project managers as a profession. The development of more sophisticated software tools for project management allowed the management of more complex and geographically dispersed projects [17].

Academic research reflect this history in the subjects investigated and published. In their review of the first ten years (1983-1992) of the International Journal of Project Management, Betts and Lansley state that papers on project planning and project organization have declined considerably, while publications on project start-up, and project performance information have increased in number. They also noticed, that the majority of publications are reviews (196 publications) and case studies (103 publications) while only a small number (38 papers) presented empirical research. This is clearly linked to the development of project management as a practitioners' discipline that is growing into more systematic research and formalization. The fact that in the first five years the grand majority of papers came from writers in the private sector, and in the second five year period an increasing number of papers originated from universities [20] attest to this development. Crawfort et. al. review the papers published in the International Journal of Project Management and the Project Management Journal in the years 1995 through 2005. They explain, that a great number of researches in the field still comes from the construction sector, thus causing a possible industry bias in the current knowledge about projects. The topics of project evaluation and improvement and strategic alignment have gained increased attention. The investigation of risk and cost have shown to be of consistent importance [21].

2.2 The Definition of a Project and the Project Management Life Cycle

The PMBOK defines a *project* as *a temporary endeavor undertaken to create a unique product, service, or result* [1]. Temporary refers to the fact that projects have a start and an end. Projects start with the signature of the project charta and/or the kick-off meeting. Their end is brought about by the completion of its objectives or a decision to abandon the effort. It does not necessarily mean they are short in duration. This clearly distinguishes projects form operations that are ongoing efforts. The purpose of a project is to produce a measurable or tangible outcome, such as a product prototype, the implementation of a new business function or a research report. The criterion of uniqueness is another distinguishing characteristic of projects versus operations, as the latter tend to be repetitive. In addition, it is a factor that introduces risk to the realization of a project, as the project outcome is unique, thus there is a certain degree of uncertainty present in the planning and execution of projects.

Turner and Müller suggest that the pressures of uncertainty, need for integration and sense of urgency innate in projects is what distinguishes the management of projects from the management of operations rather than the balancing of schedule, cost, and quality, which is also present in operations management. They furthermore define a project as a temporary organization within the sponsoring organization where the Project Manager serves as CEO of that temporary organization and acts as an agent on behalf of the project sponsor. When taking this perspective, the role of the project manager shifts away from the emphasis on generating and monitoring plans and schedules to the tasks of objective setting, and motivating team members [22]. Kolltveit, Karlsen et. al. show that there are several different perspectives of project management present in the current literature. When examining project management from these perspectives, different skill sets and tools are described and viewed as central to project management. The existence of various perspectives on the subject of project management shows that it is a field that combines various disciplines and that there are many different possible approaches and methodologies to integrate these disciplines [23]. This is also reflected in the various efforts to generate a project management body of knowledge in different countries. The body of knowledge documents created by six different national project management organizations show considerable differences in contents and approach [24]. In the USA the Project Management Institute's Guide to the Body of Knowledge is the most commonly used reference. The PMI PMBOK has continuously included new knowledge areas and with them new perspectives of project management. In the current edition *Project Management* is defined as *the application of knowledge, skills, tools and techniques to project activities to achieve project objectives* [1]. The management of projects requires the identification and documentation of requirements, managing project time and budget, staffing the project, and determining tasks and task order, balancing the dimensions of scope, quality, cost and time and facilitating the communications within the project team and with stakeholders. In its latest, the fourth edition, the PMBOK places increased emphasis on identifying stakeholders early in the project and on thorough requirements definition [1].

Projects are marked by progressive elaboration, which means that task break down and planning become finer grained and more detailed as the team learns about and understands the project. Complex projects will be elaborated in incremental steps, using a type of divide and conquer approach. Complex projects can also be divided into phases. Projects pass through a life cycle. The division of projects into life cycle phases depends mainly on the preferences and needs of the organization. Typically, projects phases are sequential, with deliverables and some sort of handoff or checkpoints at the end of each phase. These phase exit reviews are also referred to as *phase gates or kill points*. A *deliverable* is defined as *a verifiable and measurable work product* [1]. The phase design and the deliverables checkpoints approach is designed to exert proper control over the project at all times. Even though there is no unified phase model, the following observations are valid for most of them: (1) Staffing levels and cost are low in the initial phases, increase and peak during the intermediate phases and fall during the final phases. (2) The cost of changes grows exponentially as the project progresses through its phases. (3) The influence of stakeholders on the final outcomes of the project drops as the project progresses. This is the dual effect to the rise of costs of changes throughout the life cycle. (4) The level of risk and uncertainty is highest in the initial phases and drops as the project progresses [1].

2.2.1 Project Management Processes

In a more abstract point of view, project management is achieved through the application of processes. The PMBOK presents processes according to process groups that contain processes that can be applied for initiating, planning, execution, monitoring and controlling, and closing of a single project. These collections of processes are a representation of what is generally considered good practice of project management. However, they do not constitute a methodology. It is up to the project manager to determine the right set of processes and the time, frequency and rigor of their application. This activity is called *tailoring*. The methodology chosen by

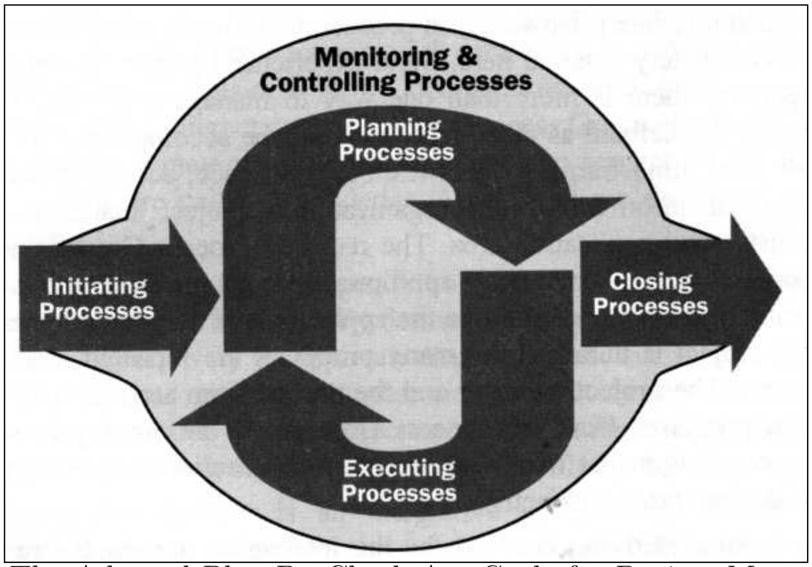

Figure 2–1: The Adapted Plan-Do-Check-Act Cycle for Project Management Process Groups. Source PMBOK [1]

the project manager must achieve the project goals by balancing the constraints of scope, time, cost, quality, risk, and resources. Project management processes are not applied strictly sequentially, in many occasions they interact, or overlap. An underlying concept is the Plan-Do-Check-Act cycle as defined by Shewart and adapted from Demming, where the cycle is linked by results: the results of one process become the input of another. Figure 2-1 illustrates the interactions of the project management process groups, using an adaptation of the Plan-Do-Check-Act cycle.

The initiating processes materialize the project by developing the project charta. A project charta describes the purpose and scope of the project, identifies sponsors, the project manager and stakeholders. The project charta formalizes the existence of the project and authorizes work to begin. After this initial phase, begins a cycle of planning and execution processes. Planning and execution are interacting, as the project is elaborated progressively, that is, there are several planning-execution cycles within a project. This is the link to the Plan-Do-Check-Act cycle. After each

Table 2–1: Project Management Processes Mapped against the Grid of Project Management Knowledge Areas and Process Groups [1]

Knowledge areas	Process groups Intiating	Planning	Executing	Controlling	Closing
Integration Management	-Develop Project Charta	-Develop Project Management Plan	-Direct and Manage Project Execution	-Monitor and Control Project Work -Perform Integrated Change Control	-Close Project or Phase
Project Scope Management		-Collect Requirements -Define Scope -Create WBS		-Verify Scope -Control Scope	
Project Time Management		-Define Activities -Sequence Activities -Estimate Activity Resources -Estimate Activity Duration Develop Schedule.		Control Schedule	
Project Cost Management		-Estimate Costs -Budget Costs		-Control Costs	
Quality Management		-Plan Quality	-Perform Quality Assurance	-Perform Quality Control	
Human Resources		-Develop HR Plan	-Acquire Project Team -Develop Project Team	-Manage Project Team	
Communications Management	Identify Stakeholders	-Plan Communications	- Distribute Information Manage Stakeholder Expectations	-Report Performance -Stakeholders	
Risk Management		-Plan Risk Management -Identify Risks -Perform Qualitative Analysis Perform Quantitative Analysis -Plan Risk Responses		-Monitor and Control Risks	
Procurement		-Plan Procurement	Conduct Procurement	Administer Procurement	Close Procurement

execution phase another planning phase is entered in which the next stage is defined and needed corrective action for the new execution phase is determined. Project closing processes follow after the last planning-execution iteration and wrap up the project. Monitoring and Controlling Processes are present throughout all project phases and linked to the execution of all project processes. They ensure that the project is on the right track, detect problems and sometimes might suggest the abortion of a project. The following table 2–1 maps the processes from the different project groups–Initiation, Planning, Executing, Monitoring and Controlling, Closing group– to the knowledge areas relevant to project management–Integration, Scope, Time, Cost, Quality, Human Resources, Communications, Risk, and Procurement Management [1].

CHAPTER 3
IT PROJECT MANAGEMENT

Q1: What competencies are required in order to successfully manage individual projects and project portfolios in Information Technology?

In order to manage projects efficiently, the project manager needs understanding of the project management tools and processes, but also needs to possess some working knowledge of the domain the project belongs to [25]. The following chapter will outline some of the particularities of software projects. It will provide examples of the types of decisions an information technology (IT) project manager has to facilitate or to take, what techniques the project manager can use, and what success criteria apply to software projects.

3.1 Software Project Specific Topics

Software development is a complex process that encompasses many interlinked and interdependent activities, such as domain analysis, specification of requirements, identifying and adopting the most appropriate technologies, end-user acceptance testing, application deployment, maintenance and multi-level support, and knowledge transfer. A project manager for IT projects needs to understand these subjects sufficiently well to know which subject matter experts to consult, to facilitate their work, to be able to communicate decisions made properly, and to be able to identify risks inherent to these issues.

Software Development Process:

The software development process is the methodology used to coordinate, prioritize and organize a series of activities in order to achieve a desirable output. The project manager has to ensure that the project follows the predefined process. What process is used depends on the type and size of project worked, the experience level of the organization with certain processes and their success rate, the culture of the organization, or even personal preferences of key subject matter experts involved in the project. Well known examples of software development processes are the Waterfall Model, the Spiral Model, the Rapid Prototyping Model, and the Unified Process for object oriented design. The waterfall model assumes that the development of a software process can be divided into separate phases and that the development process will pass through these phases sequentially without having to revisit the previous phase. Typically the phases considered for the waterfall model are: system feasibility analysis, requirements gathering, preliminary design, detailed design, module coding and testing, system integration, system testing and system maintenance. Criticism of the waterfall model is that it is not very realistic to assume that customer requirements can be frozen early on in the project, as would be required in order to be able to produce a detailed design early in the project and not to revisit the design phase. This shortcoming has given rise to a series of agile software development methods that propagate the incremental implementation of the program and continuous discovery and fine tuning of requirements. These models require various iterations of requirements analysis, design, coding and testing in which the system functionality becomes more complete with each iteration [26].

Requirements Engineering :

Requirements engineering is the process of eliciting system requirements from the stakeholders, analyzing them, and formally specifying them. It is important to understand the requirements and control changes to the requirements, as a small change in requirements can have a profound impact on the project's cost and schedule. Requirements are the basis for the system functional design, dimensioning, they drive test scenarios, and they are the source of user acceptance criteria. The project manager needs to assure that requirements are documented as completely and accurately as possible, that all team member understand these requirements well, and that changes to the requirements are avoided in later phases of the project. The main source of functional requirements is the future user of the system. The project manager has to assure the user and other relevant subject matter experts are available throughout the life of the project, so that delays or errors due to the team's lack of domain knowledge are avoided [26].

Software Architecture:

Software architecture defines the specifications of the application's high level structure. It encompasses considerations such as implementation model (e.g. client server model, three tier or four tier model, event based programming), technologies used (hardware, operating systems, programming paradigm, protocols), distribution of the application, interfaces, communication mechanisms and middleware. Key considerations for an architecture are business requirements, scale and scalability of the system, security requirements, availability and resilience requirements. Factors that guide and constrain the architecture can be enterprise level architecture guidelines, currently present architecture, currently present hardware and software, currently present expertise in the company, license costs, and availability of commercial of the shelf components [27]. As soon as high level requirements are known, the project manager needs to facilitate that the application architect develop an initial architecture. For that the requirements have to be documented as they are known up to this point. This is done in a business requirements document which can be either written by the client, the project manager, a business analyst or in a team effort.

The project manager also has to assist the architect in gathering relevant information, such as the cost of new hardware that would have to be acquired, available hardware within the company, available licenses or license costs of new installations. According to the Unified Process of Software Development, a proof of concept for the chosen architecture should be carried out early in the project. A proof of concept is an abstract, but working implementation of the system to be built that covers about 5 to 10 percent of the use cases defined in the requirements document. The software architecture gets refined as more requirements and use cases are discovered. Designing a software architecture is a complex process that considers many inputs and oftentimes requires to analyze the implications of many different alternative architectures. Therefore, the architecture is considered a product itself [26, 27] .

Organizational Aspects:

As noted in the PMBOK, organizational aspects greatly influence the way a project manager has to work the staffing aspect of the project. Team members need to be recruited, roles need to be assigned, communication channels and mechanisms need to be defined and facilitated. These activities are common to all types of projects. Software project managers need to understand the roles that are required in software projects, such as programmer, technical lead, analysts, database administrator, and staff the project accordingly. Not all roles are required in all phases of the project, so the staffing plan has to take the different personnel levels and group profiles required throughout the various phases of the development process into consideration [26].

Management Strategies and Techniques:

Software development is a dynamic process. The management techniques applied must allow for this dynamic behavior, whilst keeping the project aligned with strategic goals. It requires striking a balance between the need of structure and following defined processes and the need for flexibility, informality. For that, defining and applying meaningful measures and reviewing them frequently is essential. A constant feedback mechanism should be enforced, so that project progress is traceable at any given moment. Progress charts–often in the form of Gantt charts–that show the percentage of completion is a possible technique for this review process. The project manager should understand various measures of progress and their applicability to the current project. It is also the project manager's responsibility to keep track of all previous versions, changes, discarded designs that occurred throughout the project life cycle and commit these to some sort of repository [26].

Risk Assessment:

Project management and software development processes can never fully eliminate risk, however they can help to identify risks early on, anticipate it, and manage it. The following are essential factors for mitigating risk in software projects.

The project size should be estimated early on in the project. Project size is positively correlated to project cost, duration and risk.

The project needs the commitment of top management, so that risk factors that arise can be dealt with swiftly.

The project needs the commitment and participation of the user. User involvement is required in order to minimize the risk of the final product not meeting user expectations. The timeliness of communication of additional user requirements has to be ensured. It is most desirable to freeze specs early in the development process. Otherwise a disciplined approach to change control has to be taken.

When creating the architecture, the project manager should insist that various views of the architecture be created. These various views, even though they might have some redundancy, will avoid omissions and give all participators a clearer understanding of the specs [26].

Software Metrics:

Software Metrics provide a quantifiable description of processes and results, thus allowing for a better understanding. Numerical measures also allow for better comparison among projects. Typical areas of measurement are team productivity, project size, schedules, requirements specifications stability, and metrics of software testing. Project size used to be measured with source lines of code SLOC. With the rise of third generation programming languages this measure became less useful [10]. Measuring function points, also called functionality points or use cases, gives a better gauge of project size. Number of functionality points is a count of distinct required functionalities from the user's perspective. Team productivity can be measured through their output of functionality points or SLOC per time unit, depending on what measure is used as the base measure for project size. Metrics for schedule are the number of tasks completed, started, tasks with changed schedule or postponed tasks. The number of requests for change (RFC) submitted is a measure of requirement stability. Metrics for software testing are metrics that describe the completeness of the testing process, such as the number of use cases covered, percentage of SLOC covered. A metric for software quality is the number of errors found per group of use cases or 1000 SLOC. There are several different metrics thinkable depending on factors such as programming paradigm used, software development process used, organizational goals, and reporting tools available. The project manager has to understand, implement, apply, interpret and evaluate the most appropriate measures for the given project [26].

Software Testing:

Software testing is a complex process and takes up a considerable amount of time within a software project. It also requires a determined skill set of software testers that only partially overlaps with the skill set of software developers. Testing should be integrated from the earliest stages of the project development possible, as testing will discover functional errors, design flaws and glitches and thus indicate rework to be done. The earlier an error is discovered the smaller its impact. That is why some software development processes, such as eXtreme Programming propagate a test driven approach to programming. There, in each new phase of development the first task is to define and program test cases that simulate user interaction with the functionalities to be implemented. Software products will typically undergo a unit test phase, where each individual component is checked for its functionality, an integration test phase, where the interaction of units is tested and an acceptance testing phase, where the user verifies that the software product has the expected functionality. The project manager has to plan for sufficient time to be allocated in the schedule for testing efforts and coordinate the various work groups that participate in the integration testing [26].

Software Quality Assurance:

Quality is a measure of how a software product's operational behavior complies with specifications and user expectations. A software product has external and internal quality characteristics. External quality characteristics refer to the product's functionality within its environment, thus comprises the dimensions of usability and reliability. Internal quality characteristics are marked by how the product is developed and structured and thus includes characteristics such as reusability, scalability, ease of debugging and change, and fault rates. Quality assurance is any effort dedicated to developing processes and practices that assure that the product meets requirements and assuring that the process be followed. The project manager needs to facilitate the definition of quality criteria before the beginning of development and link external quality criteria to internal quality criteria. For this the input of subject matter experts, company policies, industry best practices and standards have to be reviewed and consolidated into the quality requirements for the project. Thus, the first step is to define a quality model. The targets defined therein should be measurable and achievable, though ambitious. This quality model could be an international standard such as ISO 9126. Once quality targets are set a process that will allow for implementation, monitoring and evaluation of the quality characteristics of the project has to be implemented [26], [28].

Software Standards:

Standards embody what is understood to be a common body of knowledge and is accepted as a best practice. They form the basis to defining processes, roles, tasks and control mechanisms independent of the individual organizations, projects, or designers. They also make it possible to view the diverse activities of software development under a common framework. Software project managers should know and understand these frameworks and be able to apply them in order to ensure that quality is built into the products. In the software industry such standards are ISO 9000 and ISO 12207. ISO 9000 is a series of standards that define a framework for Quality Management System (QMS) of a supplier organization that provides services of design and development. The ISO 9000 standard pertinent to the software industry is ISO 9000-3 Guideline for the Application of ISO 9001 to the Development, Supply and Maintenance of Software. ISO 12207 covers the entire software life cycle and explains the processes for acquiring and supplying software services and products [26], [28].

Best Practices:

Best Practices refers to processes that have proven useful through experience, but cannot be generalized into a standard. Examples of best practices are holding a face-to-face kick-off meeting for geographically disperse project participants. Project Managers should be aware of such best practices and apply them where they are suitable [26].

Software Configuration Management:

Software configuration can be defined as a collection of specific versions of hardware, software, or firmware items that are assembled into a system to fulfill a specific purpose. Software configuration Management refers to a set of technical, managerial and administrative activities that serve to identify the configurations and control changes thereof. Configuration changes need to be effected in a systematic process and should be duely documented. The project manager has to know the organizations configuration change control processes and needs to coordinate the efforts of all personnel involved in the configuration change. Furthermore, the project manager has to assure the change is properly documented. In an organization with mature processes, configuration change control can take a considerable amount of planning and documentation effort. Configuration management can also impose dates in the project schedule in case there are pre-defined maintenance windows or release dates [26].

3.2 Estimating for Software Projects

Estimations are the basis of the project plan, especially in early stages of the project where not much detail knowledge about the project is available [29]. Lederer and Prasad identified project cost estimation as an activity ranked very import to mangers (4.17 points on a 5 point scale) [30]. Software projects are known to frequently suffer from budget and schedule overrun. In their review of surveys on software effort estimation, Moløkken and Jørgensen seek to find a balanced picture of the real magnitude of estimation errors in software projects, as they feel that many authors might only cite extreme findings in order to legitimate their own research. This effect might be even more pronounced in publications by consultancies seeking to sell their services, such as the Standish Group's Chaos Report [31], or in white papers published by software houses seeking to sell estimation tools. In an effort to present an unbiased picture the authors conducted a meta-study of 10 studies published between 1984 and 2002. The investigation concluded that the average cost overrun of 89 % as reported by Standish Group is not supported, but is more in the range of 30-40 %. They found that in all investigations surveyed, expert judgement was the most frequently observed technique for estimations over any other type of formal estimation method. Furthermore, the magnitude of errors seems to be related to the size of the project [14].

In their practitioner's guide to software project estimation, Hewson and Peters [32] describe software estimates as a four step process: (1) estimating the scope of the project, (2) estimate the effort in person months, (3) estimate the schedule in calendar months, and (4) estimating the project costs. Estimating the project scope and relative size is a step oftentimes omitted by organizations, as they go directly to estimating the effort in person months. This approach, however assumes that there is a given team productivity that is known beforehand. Using a size estimate as the starting point of estimations in turn, allows for easier creation of various scenarios, assuming varying levels of team productivity. Project scope and size can be measured in SLOC, function points, number of use cases, or number of modules to be programmed. Some organizations opt for using broad categories, such as *small*, *medium* and *complex* project. Once the project size is estimated, the effort required is to be estimated. The translation of project scope to estimated effort can only be done if the organization has predefined software development processes in place and use them consistently. The effort estimates implicitly assume that the work performed will follow these development and implementation processes.

According to [2] methods for effort estimations are:

Estimation by Analogy refers to making a comparison of the current project with historical data from an already completed project.

Expert Judgement means having one or several subject matter experts estimate the effort required. When consulting several experts a consensus process such as the Delphi technique can be helpful.

Top Down Estimating: The total cost is estimated based on global properties of the project. The cost is then later on distributed among the various modules.

Bottom Up Estimating: Each component is estimated separately and the cost estimate of these components is combined afterwards.

Parkinson's Law suggests that work expands to fill available time, thus estimates are made to fit the available resource volume.

Price to Win cost estimates are based on what price is expected to win a bid.

Algorithmic Methods provide an estimate according to a predefined algorithm and dependent on given input parameters. Examples of widely accepted algorithmic methods are the COCOMO model and the Putnam methodology.

The Putnam SLIM model is based on the observation that the personnel level throughout the various stages of the software life cycle follows a Raleigh distribution. The macro effort estimation formula is $S = C_k K^{\frac{1}{3}} t_d^{\frac{4}{3}}$ where

S = number of delivered source instructions

K = life-cycle effort in person-years

t_d = development time in years

C_k = a technology constant; typically value ranges between 610 and 57314; constant can be calibrated with previous project experience.

Solving the equation for K provides the estimate of total effort required [2].

COnstructive COst MOdel (COCOMO) delivers a detailed framework for the estimation of software. The primary motivation was to develop a methodology that would help in the understanding of consequences of various alternatives in the development and support of a software products. The COCOMO framework presents a three level hierarchical approach for developing estimates with increasing complexity. The first level is a single macro estimation. The following, intermediate, stage provides an estimate based on a nominal development effort. The nominal development effort is calculated based on the estimated number of source instructions and a multiplier. The multiplier is dependent on the software development mode. Software development modes are categorizations that adjust the estimate according to how familiar problems and technologies are, how flexible the environment is or how ambitious the project is. The multiplier is derived form the scores the project is given on 15 cost driver attributes, such as reliability requirements, storage requirements, programming methodology used, or personnel capabilities. The third, detail level, stage considers all factors from the intermediate stage and adds an assessment of the cost drivers overall impact on project costs [2]

Table 3.2 outlines advantages and disadvantages of the various methods. Their evaluation shows that applying the Parkinson rule and Price-to-Win-Costing are hardly appropriate methods for generating cost estimates.

Based on the effort estimate the schedule estimate has to be developed. Factors that come into play here are the number of people involved in the project, what tasks they will work on, interdependencies of tasks, and other external constraints. Once the staffing profile of the project is estimated, the cost of the project has to be estimated. There are several cost factors to consider, such as labor, hardware purchases or rentals, travel expenses, training costs, and office space. The exact formula of deriving cost estimates from effort estimates, schedules and staffing profiles depends on the cost accounting approach of the hosting organization. Oftentimes

Table 3–1: Comparison of Software Cost Estimation Methods. Source [2]

Method	Strength	Weakness
Algorithmic Method	+objective, repeatable +analyzable formula +efficient +objectively calibrated to experience	-subjective inputs - assessment of empirical circumstances -calibrated to the past, not future
Expert Judgement	+assessment of representativeness, interactions, exceptional circumstances	-incomplete recall -only as good as experts
Analogy	+based on experience	-representativeness of experience
Top Down	+system level focus +efficient	-less detailed basis -less stable
Bottom up	+more detailed basis +more stable +forms individual commitment	-may overlook system level costs -requires more effort
Parkinson	+correlates with experience	-not best practice
Price to win	+bidder likely to win contract	-generally produces large overruns

a labor rate per hour is applied to labor costs according to the role of the project participant [32].

There is a minimum length schedule for each project, given the functionality that has to be implemented, and the minimum process that has to be followed in order to develop, test and implement the product. This minimum length schedule has the highest cost for it requires the highest concentration of resources. The shortest theoretically possible schedule is typically not achievable, as it assumes maximum team capacitation, the absence of communication failures or rework. The shortest achievable schedule, also called the nominal schedule, is a more realistic assessment of the time required. Usually it is possible to reduce overall costs by allowing for a longer schedule. There can be large differences in costs between various length schedules. Project managers are advised to produce various alternative schedules within acceptable parameters [32].

Projects that have a set delivery date, as could be forced by contractual agreements of release dates or by implementation deadlines enforced by regulatory agencies, require a different approach of estimating. Besides the end date, typically acceptable cost is also given. With these two constraints in mind the project manager, together with subject matter experts, will have to make a prioritized list of functionalities and decide the cut-off point of scope among them [32].

Another consideration is the type of project that is to be estimated. Maintenance and enhancement projects distinguish themselves from new developments. In the software industry, the bulk of work is actually in maintenance and enhancement work rather than in new developments. The four step process still applies for these type of projects, however additional concerns come into play. Depending on whether the team is familiar or not with the existing architecture, additional time for analysis is required. The existing architecture might have to be reworked to accommodate changes. Furthermore, regression testing is required in order to make sure the changes implemented did not introduce error in the other functionalities.

Estimating a new domain project is especially difficult, as much more uncertainty is present. Here a phased approach and incremental planning and estimation should be considered. That means that initial estimates have the quality of a ball park figure. Later estimates will have higher precision levels [32].

Even when applying the most appropriate estimation method for the given situation, the estimator cannot expect the technique to compensate for lack of understanding and definition of the requirements [2]. There are many factors that contribute to estimates being faulty:

-Estimating the size and complexity of the project is a challenging task, and too often it is completely omitted from the estimating process [32].

-'Overlooked tasks' is also a frequently cited reason for estimation error [33].

-Estimating is a process of gradual refinement. This means that estimates become better throughout the life of a project. However, management and customers expect estimates before the project is even started and expect these estimates to remain stable [32], [14].

-Furthermore it is oftentimes hard to get realistic schedules accepted by management and customers. Managers might push for revision of the estimates in order to report shorter delivery dates and less project cost [14].

-It has been shown that the very knowledge of customer expectation of duration and effort involved in a project skews the estimators perception and with it the estimate. This psychological effect is called external anchoring. Customer expectations of high effort led to overestimation, customer expectations of low efforts led to too low estimates. The estimators were not aware what influence of their knowledge of the customer expectations had on the accuracy of their estimates [34].

- Over-optimism is a frequently cited reason for estimation errors. [33]

-Organizations fail to collect historic data [32].

-Giving estimators insufficient time to complete their estimates leads to oversimplification of the estimated work [33]

-Inefficient change control and scope creep without readjusting estimates creates a misalignment between estimates and the actual project and thus a perceived failure to estimate correctly [35]

-The investigation of Lederer and Prasdad show that the estimation error increases with the use of estimation tools, when the estimator was not estimating their own work, when there was no revision of estimates by peers and management, when there was no revision of the development process, when there was no process of cost control, and when estimators did not receive after-completion feedback and evaluation of their estimates [30].

The following are factors that lead to more accurate estimates according to Molokken-Osvalt and Jorgensen [33]:

-The inclusion of large buffers to address unexpected events and omitted tasks do not make estimates more accurate, but compensate for possible estimation errors.

-Estimates for simple and small projects are generally more accurate.

-Estimates become more reliable if the team has experience from similar projects.

-A contributing factor is good knowledge on how to solve the required specifications. A further contributing factor is, a high degree of flexibility of how to implement the required specifications [33].

Considering the various factors that can impact the accuracy of estimates the following practices are recommended in order to achieve better estimates [32].

-There should be enough time and resources allocated to the task of estimating.

-Estimates by anyone other than the people that will actually carry out the work tend to be less accurate. Therefore team member's estimates should always be consulted.

-It is helpful to use several different estimating techniques and compare the output.

Marked discrepancies between the various estimates would allow to detect bias in expert judgement or point at overlooked tasks in some of these estimates.

-Whenever possible the project should be developed in a phased approach, as the estimation of reduced scope phases will be more accurate.

-Documentation of past project experience in an appropriate format can be used later for estimating by analogy or for the calibration of algorithmic methods.

-To reduce uncertainty, estimators can 'buy information' by investing resources into building a reduced scope prototype. This approach requires a balance between how much uncertainty is acceptable and how much investment in reducing uncertainty is enough [2].

3.3 Process Maturity and Quality of Execution

In order for estimates to be meaningful, the processes that are used to develop the software product have to be sufficiently controlled and predictable. The formal definition of processes and disciplined compliance to them is a way to reduce the probability of human error. One of the most prominent frameworks for the development and gradual improvement of processes is the Capabilities Maturity Model (CMM) along with its related Software Process Evaluation Instruments. The Software Engineering Institute at Carnergie Mellon University started work on this framework in 1987. The initial motivator for its development was a request by the US Air Force for the development of a methodology to identify the most capable potential contractors [36].

The Capabilities Maturity Model rests on two basic assumptions: (1) the process of creating and maintaining software can be defined, measured, managed and can be progressively evolved, and (2) the quality of the resulting software product depends greatly on the efficiency of that process. A software process consists of tasks, practices, guidelines and activities that lead the team in their production effort. Such a process must take the requirements, used technology, team skill level and team motivation into consideration. The CMM defines five stages of process maturity and the activities that are required at each level. CMM closely resembles the natural development of software producing organizations, it defines measures to assess organizations, it provides interim improvement goals, and points out the improvement objectives with the highest priority once an organization's position within the maturity level scale is known [36].

The maturity level indicates the organizations process capability. Each maturity level contains various process areas for which goals are defined. For the various process areas it guides the implementation and institutionalization of key practices, which describe the required infrastructure and activities. The following presents and overview of the five maturity levels defined by CMM [36], [28]:

LEVEL 1: Initial

Key Process Areas: None Defined

At level one the software process can be described as unpredictable and chaotic. Planning and control mechanisms are absent or inadequate. Tools are not used or not adequately integrated in the process. In a crisis defined procedures are abandoned and the team reverts to 'fire fighting mode'. Project success greatly depends on the capabilities and commitment and '*heroic effort*' of individual team members. Change control is inadequate. Senior management does not fully understand key software issues.

LEVEL 2: Repeatable

Key Process Areas: Requirements Management
Software Project Planning
Software Project Tracking and Oversight
Software Subcontract Management
Software Quality Assurance
Software Configuration Management

The organization has basic project management mechanisms in place to track cost, schedule and functionality. The organization has enough process discipline to be able to repeat previous successes. The strength of the organization comes from their experience in doing similar work. Organizations at level 2 face high risks when working with a new task. The organization lacks organized structures for process improvement.

LEVEL 3: Defined

Key Process Areas: Organization Process Focus
Organization Process Definition
Training Programme
Integrated Software Management
Software Product Engineering
Intergroup Coordination
Peer Reviews

At level 3 the organization has formally defined its process for both management and engineering activities. The process is standardized, documented and integrated into the organization's practices. The formal definition of processes lays the foundation for examining, evaluating and improving processes. Key elements of level 3 are designating a software engineering process group to define and document the software engineering processes for the organization, and the implementation of an adequate training program.

LEVEL 4: Managed

Key Process Areas: Quantitative Process Management
Software Quality Management

Based on the foundation of defined processes, measures are now defined and collected for the processes in order to examine and evaluate them. This is a necessary step in order to identify areas for process improvement.

LEVEL 5: Optimized

Key Process Areas: Defect Prevention

Technology Change Management

Process Change Management

The organization is focused on continuous process improvement. Performance data gathering should be automated. Senior management gives high priority to process monitoring and improvement. Numerical evidence on process effectiveness is available and the organization is equipped to identify the weakest process components. Processes are constantly re-evaluated to prevents recurring defects and lessons learnt are disseminated efficiently. The use of new technology is justified by its contribution to process improvement.

The further up a company moves on the maturity level scale, the more data collection is required. Humphrey argues, that at some point an organization will have to invest in an Computer Aided Software Engineering (CASE) tool in order to be able to produce the real time measurements required [36].

The Software Engineering Institute also developed the Capability Maturity Model Integration in order to provide a framework to integrate the various maturity models that have spawned for various disciplines since the early nineties. CMMI integrates three models: the Capability Maturity Model for Software (SW-CMM) v2.0 draft C, the Electronic Industries Alliance Interim Standard (EIA/IS) 731, and the Integrated Product Development Capability Maturity Model (IPD-CMM) v0.98 [37]. CMMI provides for two different representations, the staged representation and the continuous representation. Both representations are designed to yield the same results. What representation to choose is a matter of which representation seems more convenient or appropriate for the specific goals and situation of an organization. Though per se maturity models are designed to describe discrete levels of process efficiency and effectiveness and prescribe a recommended order of priorities and initiatives for process improvement, the continuous representation allows organizations flexibility in the order in which the various process areas are addressed. The goals and best practices included in the CMMI involve several aspects, namely process management, project management, engineering and support, making CMMI more comprehensive than CMM [28].

Maturity models are widely cited as a way for companies to gain control over the product and process quality and thus be more competitive. There are some critical voices, however. Kam Jugdev and Janice Thomas analyzed whether the implementation of a maturity model would result in an internal asset that provides competitive advantage for companies. They used various frameworks that apply the resource based view (RBV) on internal company assets and determine to what extend these assets would lead to competitive convergence or to competitive advantage. In consensus, the frameworks state that resources have value when they neutralize threads and/or take advantage of opportunities. Common or generic resources are not the source of competitive advantage, but at best can provide competitive convergence. Resources that provide competitive advantage are particular for a company and hard to imitate or move. The authors argue, that the implementation of maturity models can provide a temporary competitive advantage at best, and will mostly only provide competitive parity. Maturity models are widely available to all firms and their components are well known, so that their use is not unique to a company, or hard to copy. By their very nature they are a staged implementation of an array of best practices that apply to most companies. Additionally, the authors argue, that the main weakness of maturity models is that they provide companies with the know-what of project and process management, but not with the know-how [38].

3.4 Success Criteria for Software Projects

Solid management of individual projects is necessary for overall portfolio efficiency and success [15]. It should be noted however, that project management success is only a subset of overall project success. Munns and Bjeirmi explain that oftentimes project management success and project success are not distinguished. They propose an extended model of project life cycle, suggesting that the output of a project has a useful life beyond the stages of conceptualization and implementation. Thus the real success of a project can only be gauged long after the completion of the planning and production phase during the utilization period of the product. In current practice though, projects are evaluated at hand-off right after the termination of the planning and production phases. Thus readily available measures of accuracy of execution of these phases, expressed by the project execution variables schedule, budget, and quality, are used for this purpose. It is possible, though, to have a perfectly managed project that produced an output that is not marketable or to have a project that incurred schedule and budget slippage, but the client was able to realize more profit than anticipated. These examples serve as a caution to equating project management success to project success. [39], [40]. Good project management can contribute to a projects success but is unlikely to prevent failure if project objectives are flawed [40].

In the context of IT projects, several authors mention that the various stake holders can have very different views of what constitutes success [41], [4], [42], [43]. Wit offers the following definition of project success: *'The project is considered an overall success if the project meets the technical performance specification and/or mission to be performed, and if there is a high level of satisfaction concerning the project outcome among key people in the parent organization, key people in the project team and key users or clientele of the project effort"* [40]. Barry Boehm points out that software engineering does not exist in a value neutral setting. He suggests to organize the different project objectives and success criteria in a 'spider web' diagram, such as illustrated in figure 3-1, to identify possible conflicts of interest. The activity of becoming aware of the various success criteria and revealing possible clashes will

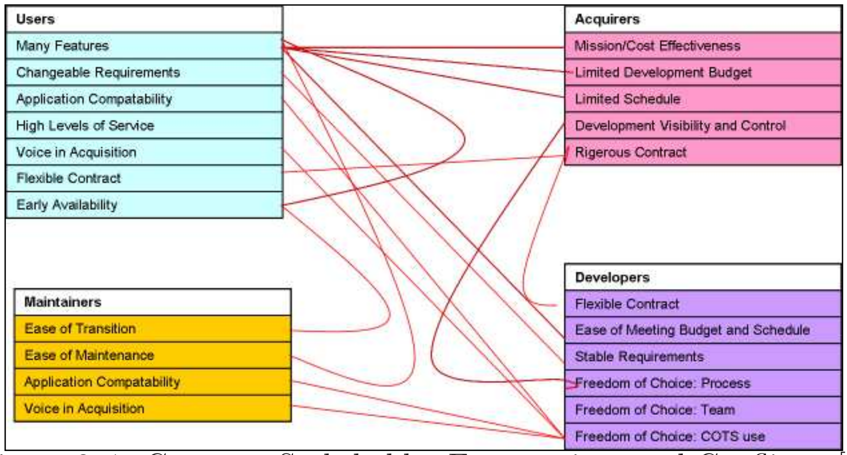

Figure 3–1: Common Stakeholder Expectations and Conflicts [4]

allow all stakeholder involved to relax their less critical expectations and reach a consensus of expectations. Such expectation management early on in the project allows for better definition of objectives and sets the project up to a good start [4]. Wateridge is attempts to define more meaningful project success measures in his investigation. He comes to a similar conclusion, as Boehm: success criteria depend greatly on the stakeholders, and should be agreed upon before the project is started [41].

CHAPTER 4
PROJECT PORTFOLIO MANAGEMENT

Q1: What competencies are required in order to successfully manage individual projects and project portfolios in Information Technology?

Project Portfolio Management refers to the management of several related and unrelated projects from a global point of view. As management of organizations by projects became a more widely used approach, the necessity of managing all of an organization's project from one central point became evident [6]. The following sections outline the definition, purpose and techniques of project portfolio management .

The idea of a portfolio as a collection of investments comes from the realm of finance, where it refers to a collection of financial instruments that was selected to mitigate risks and according to the investor's strategy. Dr. Harry Markowitz is credited with being the pioneer of modern portfolio theory (MPT) and was awarded the Nobel Price in Economics in 1990 [44]. Financial portfolios should maximize returns for a given risk, minimize the risks for a given return, avoid high correlation, and be tailored for the individual investor [5]. The required return of a financial instrument is in direct correlation to its risk, that means, the riskier the instrument, the higher its return has to be in order to be marketable [45].

The adaptation of MPT to the context of project portfolios is a logical step, as projects are always investments, however it is not a straightforward process. Projects are subject to different conditions, risks and constraints than financial instruments, like for instance, experience, management expertise, physical production capabilities and resources. The risks that affect projects can be grouped into four broad categories, *market or commercial risk*, which refers to the risk of mid project scope changes due to changes in the market, *organizational risks*, which refer to the lack of support for a project or conflicting interests within an organization, *technical risks*, which refer to unforeseen technical challenges, and *project (process) risks*, which refer to faulty project management. In contrast to financial investments, higher project risk is not necessarily directly correlated with higher returns on investment. The relationship between risk and return on investment for projects tends to be more complex [5].

Financial investments take money as their input and produce a monetary reward (or loss) as a function of their inherent risk. Projects take money, assets and human resources as their input, and their output is not limited to a monetary result, but also includes altered strategic direction of the company, the addition of new capabilities and altered efficiencies, or the creation of a new product [5]. Figure 4-1 draws a comparison between the goals of Financial Portfolio Management and the Goals of Project Portfolio Management.

Financial portfolio managers usually have fairly complete and standardized information about their portfolio assets in the form of annual reports, financial statements and analyst reports available to them. Tools and techniques for ranking and evaluating investments are well defined. In contrast, the components of project portfolios might be harder to asses, as they may not follow the same metrics or reporting standards. Finding processes and techniques to make projects easier to assess and compare, is one of the main goals of project portfolio management [46]

The Project Management Institute's standard for Project Portfolio Management defines a *Project Portfolio* as "*a collection of projects [...] and programs [...] and other*

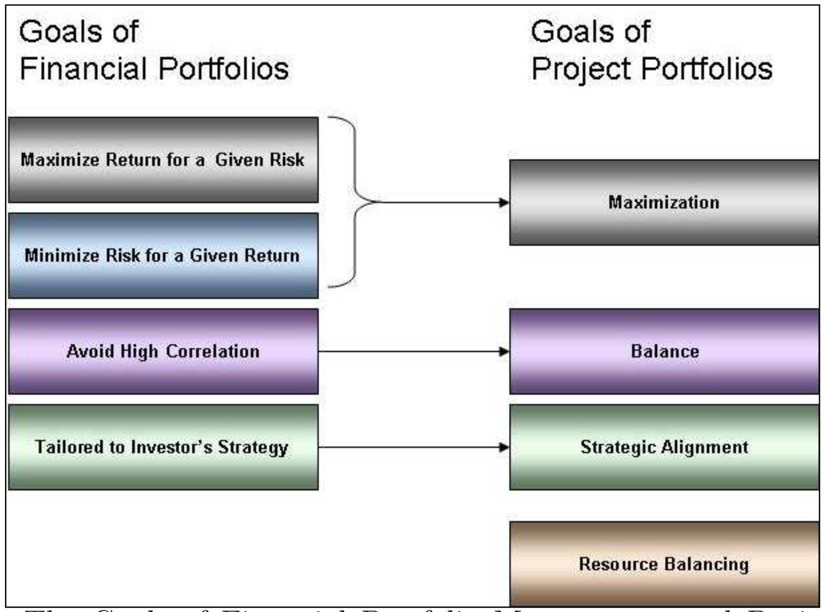

Figure 4–1: The Goals of Financial Portfolio Management and Project Portfolio Management. Source [5]

work that are grouped together to facilitate the efficient management of that work to meet strategic business objectives. The components of a portfolio are quantifiable, that is they can be measured, ranked, and prioritized [6]." A project portfolio is different from a program, as a program only contains related projects and seeks to capitalize on synergies between them in order to achieve a common strategic goal in a more efficient way than could be obtained from managing these projects individually. The components of a portfolio, in contrast, are not necessarily related and can seek to fulfill different strategic goals. The outputs of the organization's strategic planning process as expressed in mission, vision, organizational strategy and objectives and resulting operations plans forms the basis of project portfolio planning and management. Both, projects and operations are required to allow an organization to work. Organizational resources have to be allocated between operational processes and projects. Furthermore, functional groups might be involved in projects, be that as sponsors, team members or as other stakeholders. Therefore, operational and project budgets are interdependent and sometimes even intertwined. The role of the project portfolio manager in the macro view is twofold: (1) The portfolio manager must assure that the project portfolio as a whole is properly aligned with the company's operational strategy, processes and budget; (2)Projects within the portfolio have to be managed in a way as to create the best possible resource allocation that will most benefit the organization's strategy. Typically, the project portfolio planning cycle follows the organization's strategic planning cycle. With each new planning cycle the portfolio is aligned with the new strategy. Between cycles the portfolio is subject to regular monitoring and reviews. Individual projects within the portfolio are to be examined according to the monitoring processes available to project managers. Based on this periodical review projects might be discontinued or changed and resources might be shifted within the portfolio. It is the project portfolio manager's responsibility to review the performance reports of individual projects and make the necessary adjustments in order to balance the portfolio. In addition, the portfolio manager will be in charge of communicating with the stakeholders of the projects.[5].

4.1 Current Standards and Definitions

The Project Management Institute published the first edition of their Standard for Portfolio Management in 2006. Similarly to the PMBOK, the standard for portfolio management aims to describe the processes and tools that are applicable to most project portfolio management endeavors most of the time. However, it is up to the organizations and the individuals involved in project portfolio management to determine which processes and with what rigor need to be applied in their particular situation.

Project portfolio management processes fall in one of two groups: the *Aligning Process Group* and the *Monitoring and Controlling Process Group*. Table 4–1 gives an

Table 4–1: The Project Portfolio Management Processes Groups

Aligning Process Group	**Monitoring and Control Process Group**
Identification Categorization Selection Prioritization Portfolio Balancing Authorization	Review and Reporting Strategic Change

overview of the categorization of these processes into process groups. These two process groups correspond to the dependence of project portfolio management on the organization's strategic planning cycles. After the crafting of organizational strategy, the project portfolio has to be aligned to this strategy; between strategic planning cycles, the portfolio is monitored and controlled in order to ascertain it maintains its strategic alignment. The Aligning Process Group contains the processes of Identification, Categorization, Evaluation, Selection, Prioritization, Portfolio Balancing, and Authorization. Table 4-1 gives a schematic of the functions of the first six processes in the alignment group. The Monitoring and Controlling Process Group contains the processes of Portfolio Review and Reporting, and Strategic Change.

4.1.1 Processes of the Aligning Process Group

Identification

The purpose of the *Identification* process is to create an inventory of all potential portfolio components ongoing and planned within the organization. As inputs the strategic plan, a list of ongoing, and a list of proposed project is used. Furthermore, a component definition, which is a statement of criteria used in order to determine whether projects should be considered as part of the portfolio management process (e.g. a certain strategic importance, budget size), is used for initial screening of the project lists. Components are described using key descriptors and templates. The descriptors used can range form a serial number, a short description, quantitative and qualitative benefit lists, customer and stakeholder lists, to high level projects plans, including timescales and budget overviews. As a result of this process a list of included components and a list of rejected components is created. Each included component has its complete documentation and description. It is important to document all potential portfolio components according to the same standard and templates in order to achieve comparability.

Categorization

The *Categorization* process aims to map the components to a category from the strategic plan. Examples of such categories are: cost reduction, market share increase, legal obligation, risk reduction, etc. There could be several sub categories included in the main categories, such as size, duration, and geographic region. Categorization is required in order to create comparability between the components. Furthermore it is a tool to assure that the portfolio covers all strategic categories that exist for the organization.

Evaluation

The *Evaluation* process uses the strategic plan, the categorized components and key descriptors thereof to determine a score for each component. These scores might be included in a graphical comparison between components.

A typical technique for evaluation is the use of weighted scores. In weighted scores a final score is computed based on the score of a component in each category and the relative weight of that category.

Another possible approach is a graphical representation, in which the component is positioned in a grid along the axis of two criteria depending on their score in that criterion. Various criteria pairs and the components relative to them could be considered for a component. Color codes could be applied to grid in order to mark go, no-go or intermediate (caution) zones. The grid technique is especially useful in order to compare several components to each other relative to the criteria applicable to all of them.

Financial indicators play an important role in the evaluation of projects. Cost benefit analysis refers to measures that express the expected return on the investment made on the project. Expected return on investment can be gauged by various metrics, which include the Net Present Value (NPV)-Discounted Cash Flow (DCF), Payback Period, and Internal Rate of Return.

Payback Period is defined as the initial capital investment divided by the average annual net cash inflows from the project [6].

$PaybackPeriod = \frac{CapitalInvested}{AverageAnnualNetCashInflow}$

The criterion is that the projects do not surpass a certain hurdle time frame and allows projects to be categorized as short term or long term investments. This method assumes there will be enough cash inflows generated to pay back the initial investment and ignores all cash flows beyond the payback period. It also does not take the time value of money into account [13].

The ***Discounted Cash Flow*** (net present value method) method discounts all expected future cash inflows by the required rate of return (cut-off rate, hurdle-rate), thus determining the net present value of an investment in a project.

With

F_t : Cash Flow for Period t

k : Required Rate of Return

p_i : Expected Rate of Inflation for that Period

A_0 : Initial Investment

$DiscountedCashFlow = A_0 + \sum_{t=1}^{n} \frac{F_t}{(1+k+p_i)^t}$

The net present value is especially helpful when comparing the expected total benefit from projects with different payback horizons [13].

Internal Rate of Return (IRR) is defined by the following: Given a set of cash outflows and a set of cash inflows, the internal rate of return is the discount rate that equates the present value of those two cash flow sets.

PV (Inflows) = PV (Outflows).

Here the decision criterion is to prefer projects with a higher internal rate of return over projects with a lower internal rate of return [13]. The rationale for considering this measure is that the IRR of a project is its expected rate of return. In case the IRR is higher than the cost of capital required to fund the project, the project is worth undertaking [45].

Selection

The *Selection* process uses the categorized and evaluated list of components to determine which of these will be selected to be in the portfolio. The selection is based on the strategic category and evaluation of each component and the available resources within the organization. The evaluation process indicates those projects, that according to the evaluation criteria chosen appear most attractive. During the selection phase it is decided which of these projects can be realized given the resource constraints of the company. Thus, in order to arrive at a selection, the portfolio management team has to conduct a human resource, financial, and asset capacity analysis. These three areas create constraints in terms of available experts, funds, and physical assets, such as conference rooms, hardware. Selection will choose those components that have the best evaluations and can be carried out within the resource constraints found in the analyses.

Prioritization

The *Prioritization* process brings the selected components within each strategic category into an order, reflecting their rank and priority.

Techniques used for prioritization are weighted ranking and other scoring techniques.

	A	B	C	D	E	Score	
A		1	0	0	1	2	
B	0		0	1	1	2	
C	1	1		1	1	4	First Priority
D	1	0	0		1	2	
E	0	0	0	0		0	Last Priority

Single Criterion Priority Determination:
In the matrix, determine whether the project in the row takes priority over the projects in the columns relative to the criterion chosen. Then fill out the column accordingly (it will have the dual pattern). The row sum determines the score, the higher the score the higher the priority.

Figure 4–2: Single Criterion Prioritization

Projects	Criterion 1		Criterion 2		Criterion 3		Score	Priority
	Measure	Rank	Measure	Rank	Measure	Rank	Sum(rank) / #Criteria	
	Relative Importance		Expected RIO		x			
A	5 (++)	5	$16 Mio	4	14	4	4.33	5
B	2	3	$5Mio	3	2	1	2.33	3
C	2	3	$1.2 Mio	2	3.5	2	2.33	3
D	3	4	$20Mio	5	6	3	4.00	4
E	1 (--)	1	$200k	1	12	5	2.33	3

Each Criterion to be considered has a measure attached to it, be that a quantitative measure (such as expected RIO) or a qualitative measure on Likert scales. Based on the score in each criterion, the projects are ranked for this criterion. Then the rankings are summed up and divided by the number or criterio in order to arrive at the final score. The final score determines the overall ranking.

Figure 4–3: Multi Criterion Prioritization

When using one criteria ranking, such as return on investment, projects are compared to each other in a grid, as shown in figure 4-2. If project A has more priority than project B, a 1 is put in that junction field, otherwise a 0. After comparing all projects, the scores are summed up horizontally, arriving at a final total score, which expresses a project's priority relative to all other projects.

For multi-criterion ranking, as shown in figure 4-3, a series of ranking criteria are selected. The projects are then measured and ranked for each criterion. Then the average score and rank is the basis of comparison between various projects.

Portfolio Balancing

Portfolio Balancing reviews the selected and prioritized projects and seeks to accomplish the desired risk structure, strategic alignment, resource distribution and performance metrics as defined by the organization. The balancing process can suggest projects to be added to the current active project list, projects to be postponed or canceled. Project Portfolios can be balanced according to various dimensions, including, the right emphasis on the various strategic goals of the organization, the right balance between long term and short term investments, or risky and not so risky investments.

This process will also take similarities and synergies between projects into account, in order to achieve more efficient resource utilization or avoid duplicate work; in short, produce the highest possible return with a minimum investment. The process takes the list of prioritized components within the strategic categories, project evaluation reports and performance metrics, and capacity constraints into consideration in order to produce a list of approved portfolio components, inactivated and terminated components.

Tools for portfolio balancing include cost-benefit analysis, quantitative analysis, scenario analysis, probability analysis, and various graphical methods.

Cost benefit considerations evaluate the financial viability of projects. Quantitative Analysis refers to any type of calculation and tabulation of factors of interest, such as resource loading requirements or cashflows over time. As a result of this analysis projects may be scheduled so that their resource and cashflow requirements are balanced across the whole portfolio [6]

Scenario analysis refers to the consideration of various possible combinations of active and potential components. This type of consideration is also referred to as what-if analysis. Various baselines, such as resource requirements, cashflows, and time to completion can be compared between the various scenarios.

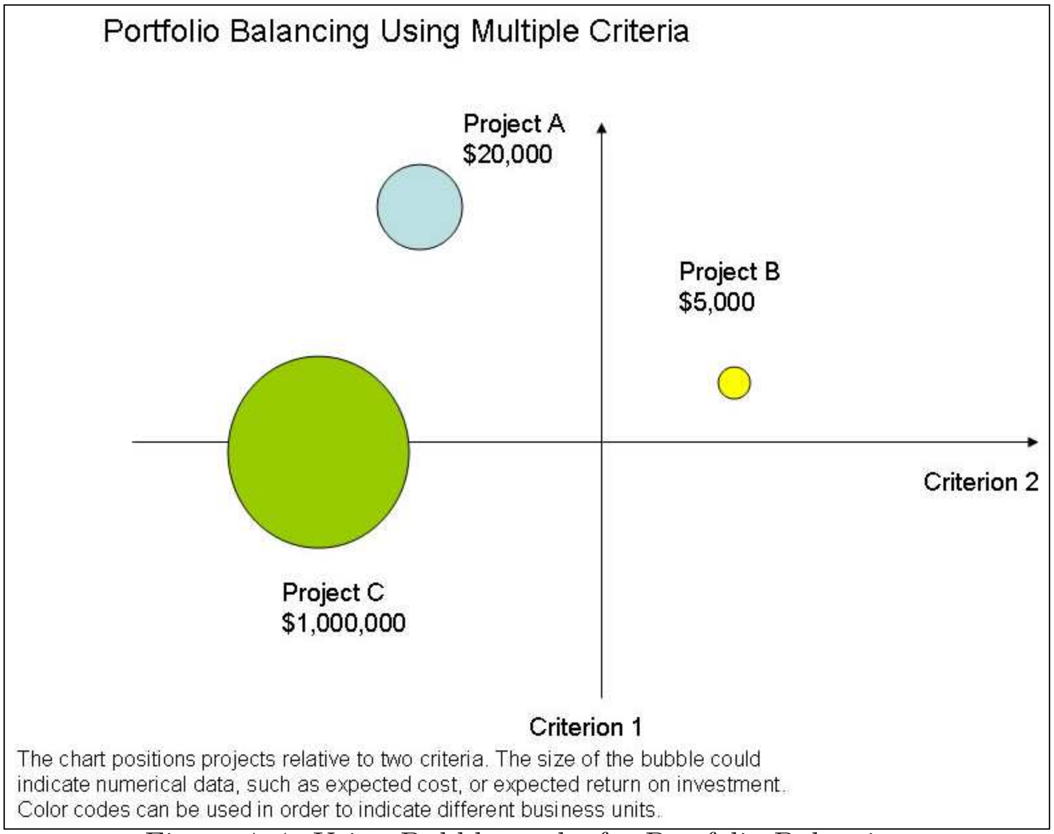

Figure 4–4: Using Bubblegraphs for Portfolio Balancing

Probability analysis uses techniques such as decision trees, flowcharts and Monte Carlo simulation in order to evaluate the probability of failure or success of a various portfolio options.

There are various graphical analytical methods that can be used for portfolio balancing, such as histograms, pie charts and bubble graphs.

Bubble graphs, such as the one in figure 4–4, allow to compare projects in various dimensions. Bubbles represent projects. They are placed on a grid according to two criteria. There are various criteria pairs possible. Bubble size could express a third variable, such as investment required, or net present value of expected returns. The bubble color could indicate a category, such as risk category or investment category [6].

Expert judgement is a tool that will be applied in all of the processes mentioned above.

Authorization

Authorization formally allocates financial and human resources to the components, using estimates provided. Furthermore the decisions about what components to include, deactivate or terminate is communicated to all stakeholders. This communication typically contains a description of approved components, their business cases, allocated budget, allocated human resources and the expected outcome. Assumptions and limitations are also discussed. In addition, a list with portfolio milestones is produced. A justification for the exclusion, halting, or termination of components is also provided.

4.1.2 Processes of the Monitoring and Controlling Process Group

Portfolio Reporting and Review is a process that considers key performance indicators of the portfolio and its components and provides reporting on them. Reviews are carried out according to predetermined review cycles. Aspects to be reviewed about components are comparing components sponsorship and ownership against governance guidelines, the alignment of selection and categorization with current strategy, resource allocations and constraints, the impact of business forecasts, and the performance indicators of individual components, such as budget and schedule. The result of this process is a catalog of recommendations for rebalancing the portfolio, directives regarding components, recommendations for the business. Additionally refined selection criteria and update key indicators can result.

The process of *strategic change* is aimed at allowing the portfolio manager to react to strategic change within the organization. The portfolio review and the updated organizational strategy are the basis of this process. The result of the process are new evaluation and categorization criteria for the portfolio components.

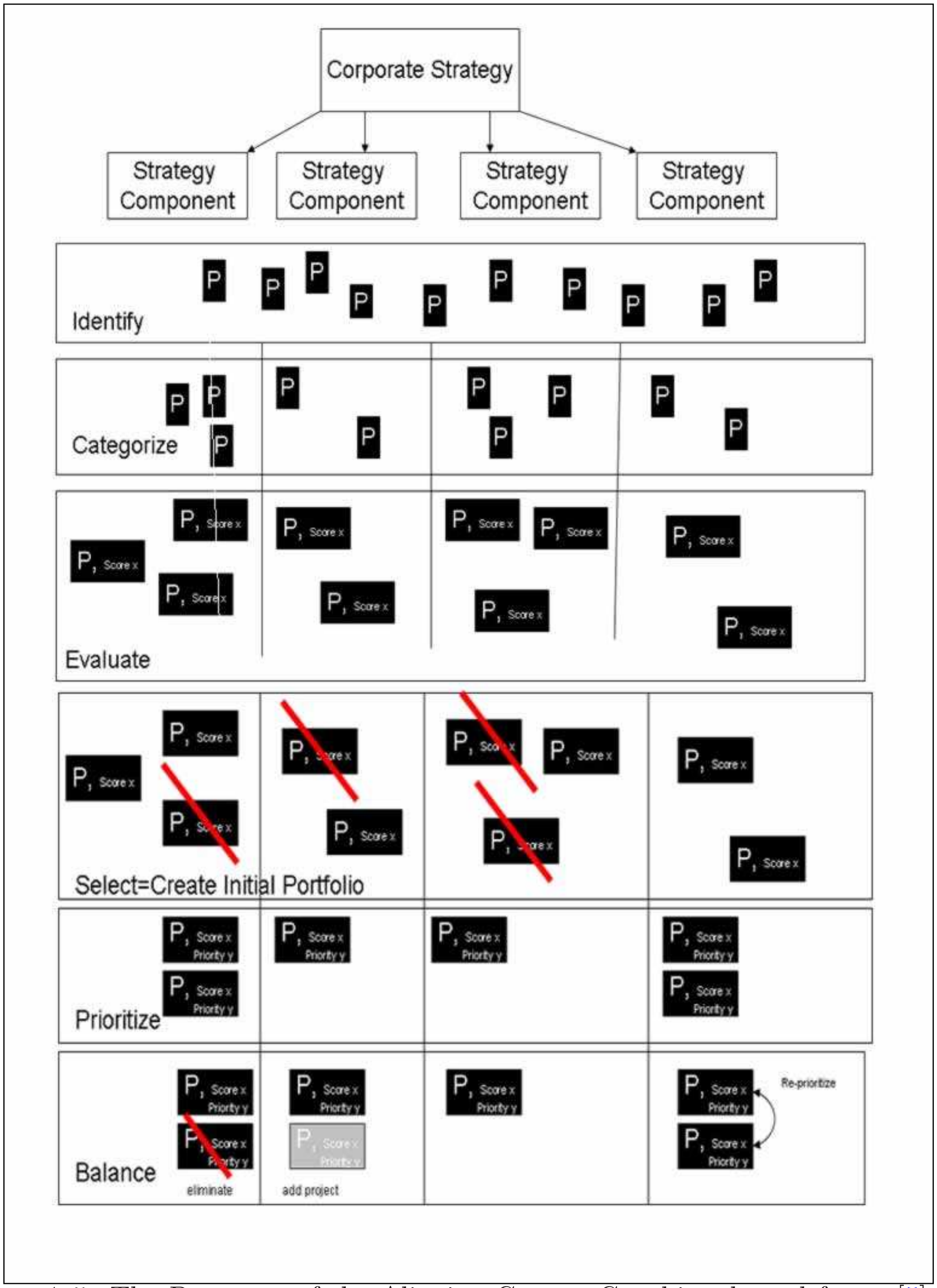

Figure 4–5: The Processes of the Aligning Group. Graphic adapted from [5] and [6]

CHAPTER 5
SOFTWARE PROJECT RISK MANAGEMENT

Q1: What competencies are required in order to successfully manage individual projects and project portfolios in Information Technology?

Projects by their very nature are risky. Since by definition a project is an endeavor to create something unique, there are many sources of uncertainty and risk involved. Therefore, the need to manage uncertainty is inherent to the project management process [7]. The following chapter will present some fundamental definitions of risk and risk management and will present a comparative overview of formal risk management frameworks as presented in standards and bodies of knowledge.

5.1 Definition of Risk

A possible definition of the term ***Project Risk*** is 'the implications of the existence of significant uncertainty about the level of project performance achievable' [7, p 7]. It is important to note here, that based on this definition, project risk is greatly influenced by what the success criteria for the project are and how project performance will be measured [7]. Negative outcomes can range from budget or schedule overruns, to lacking system performance, unmet requirements, or service interruption among others [11].

Risk Factors or Risk Items are circumstances that lead to negative outcomes, such as constantly changing requirements, personnel shortcomings, insufficient management support [11]. Ropponen and Lyytinen identified in an empirical study, that most risk items fall under one of the following six categories [47]:

1. scheduling and timing
2. system functionality
3. subcontracting
4. requirements management
5. resource usage and performance risk
6. personnel management risks

Risk Exposure is defined as $RE = P(UO) \cdot L(UO)$ with RE being the risk exposure, P(UO) denoting the probability of an unsatisfactory outcome and L(UO) representing the loss incurred should this particular unsatisfactory outcome occur. Project risk management processes consider risk exposure rather than just considering the likelihood of an unsatisfactory outcome [8]. Pfleeger et al. add a third aspect to risk exposure: the degree to which event consequences can be changed. They argue, that true risk management is only present if the possible negative outcomes are changed, otherwise risk is just noted, not managed [48]. The loss incurred in the event of an undesirable outcome can either be estimated through a quantitative analysis or by qualitatively determining the impact an undesirable event would have on the organization. Such a qualitative assessment can be done using Likert scales, for instance [11]. Possible techniques for estimating the probability of an undesirable outcomes are drawing comparisons with past projects or expert judgement and expert consensus techniques. However, probabilities are not always easy to assess and thus unreliable to estimate [49]. Given the difficulty of providing distribution functions that describe the probability of undesirable outcomes for software projects, Barki et al present a model that would determine a measure of project risk based on the absence or presence and magnitude of risk factors in a type of scoring system.

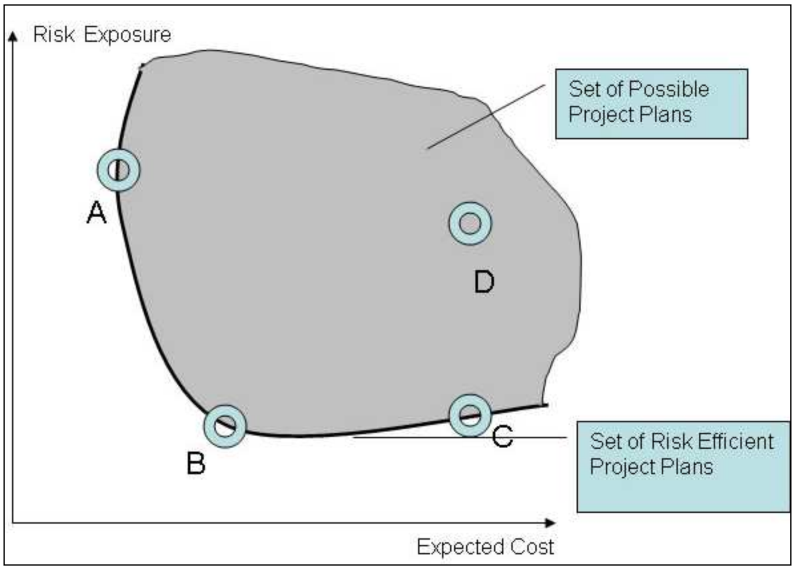

Figure 5–1: Risk Efficiency of Alternative Project Plans, adapted from [7]

The variables considered were collected from various investigations that consider risk factors of impact that are common among information systems projects [49].

Risk Efficiency when modelling performance as measured solely by cost turnout, (achieved cost versus a predetermined commitment), risk exposure is expressed by the probability and magnitude of cost overruns. In this framework, possible alternatives for project plans have the parameters of projected cost and the probability of cost overruns of a certain magnitude.

Plans that have less cost and less risk are relatively better than others. A risk efficient plan will have the minimum level of risk involved given its projected cost. The projected costs of a risk efficient plan can only be reduced by increasing risk exposure. Figure 5–1 illustrates this concept. Plans A,B,C all are risk efficient project plans. Of the three plans, A is the cheapest plan, but also has the highest associated risk exposure. Plan B offers the lowest risk exposure. Plan D is not risk efficient, because there exists a plan at the same cost line that has less associated risk, plan C [7].

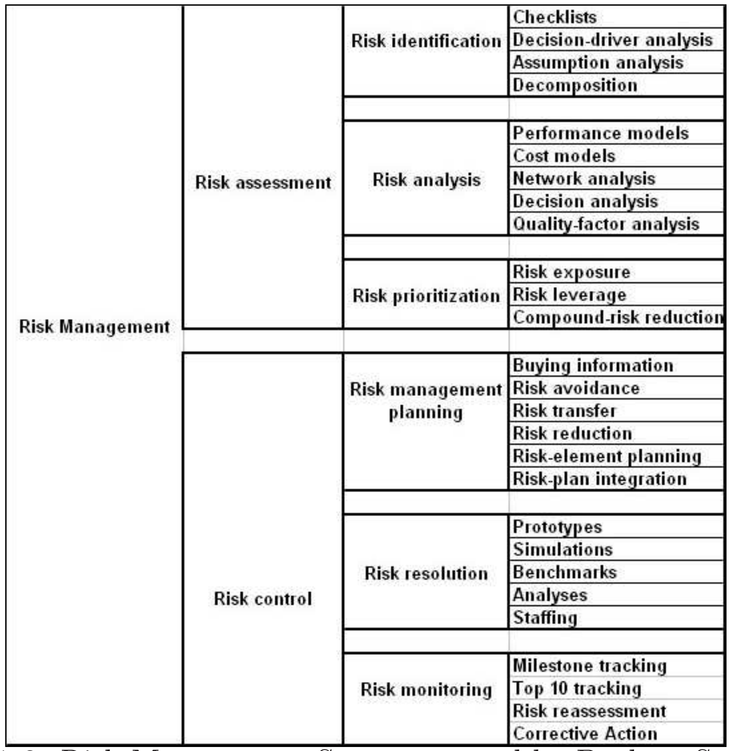

Figure 5–2: Risk Management Steps proposed by Boehm. Source [8]

5.2 Risk Management and its Standards and Frameworks

A possible definition of software risk management is: "an attempt to formalize risk oriented correlates of success into a readily applicable set of principles and practices" [50]. In a publication in 1991 Barry Boehm, then with the Defense Advanced Research Project Agency (DARPA), set forth a model that defines risk management as consistent of the two primary steps of risk assessment and risk control. Risk assessment is composed of risk identification, analysis and prioritization. Risk control is composed of risk management planning, risk resolution, and risk monitoring [8]. Figure 5.2 shows a schematic representation of Boehm's framework. It includes the techniques that apply to each substeps in the last column.

In the following we will offer a comparative summary of several standards and frameworks that describe risk management processes and have found wide dissemination in the industry . The standards and frameworks considered are ISO 9000-3, ISO 12207 and ISO 15504, Project Management Institute's PMBoK (2009), Software Engineering Institute's CMMI (2001), and the Quality Council of Indiana's process as described in the Software Quality Engineer Primer (SEQ) [51], [52], [53], [1], [37], [28].

The ISO 9000 family of standards describes processes for quality management. ISO 9001 is part of the ISO 9000 family and describes a 'model for quality assurance in design, development, production, installation and servicing of any product. In this sense it is a generic standard. ISO 9000-3 is the guideline of the application of ISO 9001 to the development, installation and maintenance of software [28]. The focus of ISO 9000-3 is on the contractor-client relationship. ISO 12207 was the first international standard dedicated to the software development process. ISO 15504, also known as ISO SPICE (Software Process Improvement and Capability dEtermination) was created in order to harmonize the various standards and guidelines for the software development process that were developed internationally. It integrates, among others, the CMMI framework (see chapter 3 for more details), Bell Canada's Trillium framework, and ESPRIT Bootstrap. ISO 12207 will be reworked in order to align with 15504 [3].

The PMBoK is the US national organization for project management's guideline for project management processes. It has also found wide international dissemination through the Project Management Professional (PMP) examination. See chapter 2 for more details.

Capabilities Maturity Model Integration is SEI's framework for the staged implementation of efficient processes for software development. It has found wide dissemination, because it is backed by the Department of Defense and it is freely available from SEI [36].

The Software Quality Engineer body of knowledge is compiled by the American Society for Quality (ASQ). It covers various subject areas relevant for software quality,

ranging from project management, to testing and validation to configuration management [28].
Standards and frameworks offer organizations orientation on what are the minimum requirements in order to implement effective, efficient and reliable risk control processes. These guidelines have to be evaluated, adapted and implemented in accordance with the organization's specific parameters, such as size, organizational structure, or expertise of personnel. A risk management process resolves a risk, when it applies to all instances in such a way, that the consequences of all possible outcomes are acceptable. The establishment of formal and structured risk management processes addresses the issue that risk management based on intuition alone is seldom effective and consistent [3].

As can be seen in table 5–1, the different frameworks use a distinct nomenclatura, however, all closely resemble each other. The continuous risk monitoring paradigm as defined by the activities of Identify, Analyze, Plan, Track, Control, Mitigate was proposed by the software engineering institute at Carnergie Mellon in 1990. The SQE primer presents this model in its pure form. The ISO standards, the CMMI model and the PMBok add a preliminary step that serves to clarifying what the organization's goals and objectives in terms of risk management are. ISO 9000-3 and SQE also explicitly mention the necessity of communication to their risk management processes. The PMBok does not explicitly mention communication as part of the risk management process, however, it defines a whole category of communications management, leaving it up to the project manager whether to include communications about risk in their communication plans. All of the process models presented only provide organizations with the know-what, not with the know how, that is, none offer concrete methods as to how to implement the processes. That is up to the organization [3].

Table 5–1: Comparison of Risk Management Processes, adapted from [3]

ISO900-3	ISO 12207	CMMI	PMBoK	SEI/SQE
Project Development Planning	Establishing the Target of Risk Management	Determine Risk Sources and Categories Define Risk Parameters Establish a Risk Management Strategy	Plan Risk Management	
Risk Identification	Risk Identification	Identify Risks	Identify Risks	Identify Risk
Potential Problem Analysis	Analyze and Prioritize Risk	Evaluate Classify and Prioritize Risks	Qualitative Risk Analysis Quantitative Risk Analysis	Analyze
Contingency Plan Definition	Risk Management Strategy Definition	Develop Risk Mitigation Plan	Plan Risk Responses	Plan
Monitoring Plan Execution	Software Risk Metrics Definition Risk Management Strategy Execution, Evaluation Execute Corrective Action	Implement Risk Mitigation Plan	Monitor and Control Risks	Track Control Mitigate
Communicate				Communicate

Chapman and Ward propose a risk management process that contains the steps *define, focus, identify, structure, ownership, estimate, evaluate, plan, manage.* Define and focus refer to the activities of defining and prioritizing goals in the risk management process with respect to the particular project. This is similar to the approach defined in ISO, CMMI and the PMBoK. Also the steps of identify, structure, evaluate, plan and manage risk mirror the frameworks set forth by the aforementioned entities. What distinguishes the approach proposed by Chapman and Ward is the activity of assigning ownership of each risk item included in the risk management plan. Ownership refers to an agreement between the various stakeholders, possibly in a client-contractor setting, of whose responsibility it is to manage and mitigate each risk item. It also defines who assumes the losses in case of an undesirable outcome occurring. The ownership issue should be clearly defined and legally enforceable. The assignment of ownership to risk items is an important activity and a useful addition to any formally defined risk management process [7], [54].

The rigor with which a risk management process is applied greatly depends on the project characteristics, most prominently the project's risk exposure in the light of the evaluation criteria of project success and organizational structures. Barki et al. suggest, that the principles of organizational contingency theory can be applied to project risk management as well. Contingency theory argues, that there is not one efficient management style, but that certain factors within the organization determine what management style is most appropriate. Similarly, Barki et al. propose that the project risk management profile has to be adapted at an individual project level; there has to be an appropriate fit between between the project's risk exposure, the risk management approach and the expected outcome [55].

CHAPTER 6
THE IMPACT OF DEVELOPING EXPERT KNOWLEDGE ON THE RELIABILITY OF ESTIMATES

Q2: What is the impact of developing expert knowledge for software development task duration estimations? Q3: How do IT practitioners in Puerto Rico perceive the effects of lacking domain or technology experience on the reliability of estimate? What are the perceived effects of experience and learning on the reliability of expert estimates?

6.1 Motivation

The methodology of project management relies heavily on the accuracy of task effort and duration estimates. Project management tools such as the identification of critical activities, the baseline schedules, milestones, resource schedules, and cost baselines depend on the accuracy of task effort and the derived task duration estimates [56]. As described in the Project Management Body of Knowledge (PMBoK) [1], there are three estimation processes in the project planning phase: Estimate Activity Resources, Estimate Activity Duration, and Estimate Activity Cost. Before estimations can be made, the requirements need to be understood and the scope of the project has to be sufficiently defined. Based on requirements analysis and scope definition, a work breakdown structure (WBS) is created, which describes in a hierarchial format what work has to be performed in order to meet requirements.

Activity definitions are derived from the WBS. Once activities are identified they need to be sequenced according to their dependencies and other constraints. An estimate of resources required for each task is the first estimate to be produced. This is necessary in order to estimate task durations, because task duration depends on the resource level assigned to a tasks and the productivity of these resources. Cost estimation in turn depends on both, resource and duration estimation. In the case of IT outsourcing vendors, the most significant cost driver is person hours invested in a project. A cost estimate for a task thus will be determined by the amount of hours a team member participated in the project and the rate at which these hours are billed. There will also be other cost factors such as licenses, materials, hardware. However, these cost factors are typically more easily determined and do not tend to change if more effort is required to complete the project than what was estimated. Thus, the investment in person hours is the variable portion of the cost estimate.

Estimates have direct impact the profits an IT outsourcing vendor can achieve. Estimates are required in order to prepare proposals. In addition, if the proposal is accepted, these estimates frequently become the terms of the contract between the client organization and the IT outsourcing vendor. Jørgensen and Sjøberg found that in theory there is a clear divide between price-to-win quote, estimated effort and planned effort. Price to win quotes use the price a potential customer is willing to pay for a project as the basis of estimation. Estimated effort is the assumed effort required to carry out a task, but contains no contingencies. Planned effort is the actual effort presumed to be required in the project. It contains contingencies and contemplates all effort required that is not necessarily task related, such as communications effort. However, in practice in many organizations the lines between these estimate concepts are blurred [16].

Overestimating required effort and with it cost does not have a direct cost to the

IT outsourcing vendor, however can have high opportunity costs in the form of rejected proposals. Underestimating required efforts leads to budget overrun and often also schedule overrun. Depending on the type of contractual agreement that exists between the client organization and the IT outsourcing partner these costs would have to be distributed between the contract parties. In flat rate contracts, the IT outsourcing provider would have to absorb these costs fully [10].

In his analysis of the most serious risks for software projects, Caspers Jones, lists the risks factors that are most likely to cause considerable damage to a software projects. He also points out that these risks are accumulative and can reinforce each other. This analysis ranks inaccurate cost estimation in fifth position. In first position he ranks inaccurate metrics, such as using Lines of Code instead of Function Points as a gauge of project complexity. Inaccurate metrics are a contributing factor to faulty estimates and lacking process control. In second place, he ranks inadequate measurements, such as not recording unpaid overtime, managerial effort or quality assurance efforts. If such flawed historic data is used to validate estimates against this data, estimation errors will ensue. Excessive schedule pressure ranks in third place. Frequently, schedule pressure is the direct result of too optimistic estimates [10]. It is evident, that estimates play an important role for project success. We therefore investigate the estimation process, specifically the technique of expert estimation, in more detail in the following sections.

6.2 Literature Review on Expert Estimation of Development Effort

Expert judgement is the most frequently used estimation method. A compelling reason for that is that expert judgement is easier to use than formal estimation models [14]. A review of studies on expert estimation found no substantial evidence in favor of using formal estimation models over expert judgement techniques. Moreover, when there is not sufficient or unreliable historical data of past projects available to calibrate the formal models, expert judgement delivers equally as good or better results than generic formal models [57]. A caveat to this statement is that expert judgement methods become increasingly unreliable with the size of the project. That explains that for example in the case of large scale military projects formal estimation methods are used far more frequently than in the development of management information systems [10].

In addition, the reliability of expert estimations depend on how closely the project tasks to be estimated resemble past experience and the ability of the expert to remember the previous projects in detail [56].

In his review of studies on expert estimation of software development effort, Magne Jørgensen also identified 12 principles that should be heeded when using expert estimation [57]:

1 Evaluate estimation accuracy, but avoid high evaluation pressure.

2 Avoid conflicting estimation goals.

3 Ask estimators to justify and criticize their estimates.

4 Avoid irrelevant and unreliable estimation information.

5 Use documented data from previous development tasks.

6 Find estimation experts with relevant domain background and good estimation records

7 Estimate top-down and bottom-up independently of each other.

8 Use estimation check-lists.

9 Combine estimates from different estimation experts and estimation strategies.

10 Assess the uncertainty of estimates.

11 Provide feedback on estimation accuracy and task relations.

12 Provide estimation training opportunities.

Points 1 through 6 are suggestions of actions to take in order to avoid human biasis. Suggestions 7 though 10 are actions to take in order to support the estimation process itself. Point 11 and 12 refer to the creation of an environment that supports learning and continuous process improvement. These 12 principles were distilled from empirical evidence found in the various studies contained in the review [57]. A conclusion from the literature review is that expert judgement can be an efficient and cost effective tool for estimation. However, experts need to have relevant experience. The organization has to provide conditions that limit estimation biases and provide feedback and learning mechanisms in order to support the estimation process. Also, this technique is not appropriate for all projects.

6.3 A Survey on Expert Estimation among Puerto Rican Practitioners

As evident from the literature review, for expert judgement to be an efficient tool, the expert has to unite several skills and experience. They need to have domain knowledge, software engineering knowledge and knowledge of the particular software engineering process of the organization, knowledge of the technology mix to be used, as well as expertise in estimation. Many organizations lack in the latter aspect, as they do not formally strive to develop estimation skills in their experts [57]. It is probable though, that the personnel entrusted with an estimate might lack in one or more of the first three skill sets as well for a particular estimate.
Even though the points mentioned above will be obvious to most project managers, it is doubtful they are aware of the magnitude of estimation risk created by relying on expert judgement of personnel that is unfamiliar with the domain or the technology mix to be used in the project. Lack of domain knowledge should be interpreted as that the estimators are faced with a business problem they have not worked on before, it does not necessarily refer to a change of industries. A change of technology mix implies the introduction of one or several new development tools, programming languages, middlewares, operating systems, data storage systems, style of computing or architectural paradigm. We investigate the following research question:

Q3: How do IT practitioners in Puerto Rico perceive the effects of lacking domain or technology experience on the reliability of estimate? What are the perceived effects of experience and learning on the reliability of expert estimates?

In particular, the goal of the investigation is to see the effect of missing domain knowledge or missing experience with the technology mix to be used in the project. Furthermore, we investigate the effect of experience with similar projects and learning.

6.4 Methodology

The investigation of Q3 was carried out by means of a survey among experienced information technology professionals in Puerto Rico.

The survey deliberately is directed at information technology professionals fulfilling the job functions of senior developer or systems analyst, not project managers. The underlying assumption is that it is senior developers or system analysts that give estimates, not project managers. Project managers will identify personnel in these job functions as experts and therefore solicit estimates from them. Asking the person that will finally carry out the work to do the estimates is considered best practice in project management practitioner's literature [32].

The study participants were selected from the extended professional network of the principal researcher, avoiding overrepresentation of one specific company. None of the companies where the survey participants work has a formal training program on software estimates, nor do they provide any formal structures for estimation process improvements. The experts depend solely on their own learning experience. Whether or not a company has a formalized effort to improve estimations processes was not a selection criterion when choosing survey participants. The fact that none of the companies formally aims at improving the estimation process is more a reflection of common practice.

For the purpose of the study, it is assumed that the personnel asked to perform the estimates has experience with software engineering practices and the organization's software development process. Therefore, only experienced personnel was included as survey participants.

It is also assumed that of the dimensions of experience with the business problem (domain knowledge) and experience with the technology mix, the person asked to provide the estimate is at the most lacking in one. A lack in domain knowledge and a lack in technology knowledge would clearly disqualify the person form giving any estimates. In such cases, the project manager would have to find alternative ways to estimate.

The survey participants were approached in person and asked to fill out a questionnaire. Since a personal approach was used, all questionnaires distributed were returned. The filled out questionnaires were collected in an envelope and had no marks that would allow their distinction, thus anonymizing the sample. The questionnaire is shown in table 6-1.

The survey considers two scenarios. Scenario 1 is a situation where the person asked to give the effort estimate for tasks has experience with the particular domain problem; however, a new technology will be used in the implementation of the solution. Scenario 2 is the inverse situation, where the person asked to provide the estimate is experienced with the technologies to be used in the implementation of the solution, however, lacks experience with similar domain problems. This does not imply that there was no analysis of requirements prior to the estimation process or that the domain problem was not properly defined. It only implies that the estimator has not worked on a similar project before. Therefore the business requirements are not as well understood as in the case of familiar problems. The estimator would, for

Table 6–1: Questionnaire Distributed to IT Professionals

SURVEY ON ACCURACY OF ESTIMATES AND EXPERIENCE EFFECTS

General Information

Education	◯ Some College	◯ Bachelor's Degree
	◯ Some Master's	◯ Master's Degree
	◯PhD	◯Other, Specify:
Work Experience	◯ < 1 yrs	◯ 1-5 yrs
	◯ 6-10 yrs	◯ > 10 yrs

I . When faced with a task where you understand the business requirements well and have experience solving similar problems, however, the technology mix (e.g.databases, programming languages, frameworks, interfaces) is NEW to you:

1. How frequently do you need more time to complete your tasks than was allocated in the estimates?
Please provide percentage estimate. Valid values are 0% through 100 % ....

2. If you use more time than was allocated to the task, by what proportion do you typically overrun the time budget?
Please provide percentage estimate. Percentage values greater than 100 are possible.
For example a value of 120% means that you needed all the time allocated for the task and an additional 120%.
So if 10 hours were estimated for a task, you actually required 22 hours [10+12] ....

3. How many similar projects would you have to work in order to significantly reduce this overrun and give more accurate estimates?
Please state number:

II . When faced with a task where you DO NOT understand the business requirements well and DO NOT have experience solving similar problems, however, the technology mix (e.g. databases, programming languages, frameworks, interfaces) is FAMILIAR to you:

1. How frequently do you need more time to complete your tasks than was allocated in the estimates?
Please provide percentage estimate. Valid values are 0% through 100 % ....

2. If you use more time than was allocated to the task, by what proportion do you typically overrun the time budget?
Please provide percentage estimate. Percentage values greater than 100 are possible. For example a value of 120% means that you needed all the time allocated for the task and an additional 120%.
So if 10 hours were estimated for a task, you actually required 22 hours [10+12] ....

3. How many similar projects would you have to work in order to significantly reduce this overrun and give more accurate estimates?
Please state number:

example, be less likely to consider overlooked requirements or tasks. The questionnaire asks survey participants to assess the probability and severity of time (task duration) overruns when compared with the original estimates under the two scenarios. In software development, person hours are the greatest cost driver and most frequently used measure for effort required. Since effort estimates are considered on a task level, an overrun in effort will typically directly translate to an overrun in time required, which in turn might lead to schedule overrun, depending on the float of the tasks considered. It is seldom possible to prevent duration overrun by adding more resources [58].

Furthermore, survey participants were asked to indicate how many similar projects they would have to work on in order to feel more confident in their estimates.

6.5 Results

The survey produced 47 datasets. The response rate was 100 percent, as the researcher approached participants personally. Participants filled out the questionnaires in private and deposited them in a collection envelope, thus maintaining the anonymity of the questionnaire.

Of the 47 participants, 13 have 1-5 years of relevant work experience, 12 have 6-10 years of relevant work experience, and 21 have more than 10 years of relevant work experience. In one questionnaire this information was not provided.

When partitioning the sample according to the highest level of formal education in computer science, software engineering or information systems, 25 participants hold a Bachelor's degree, 11 participants have partially completed a Master's degree, and 9 participants have completed a Master's degree. Two study participants answered 'other' in this category.

6.5.1 Scenario 1, Descriptive Statistics

Scenario 1: Probability of Effort Overrun							
	Whole Sample	5 years	10 years	more 10 years	Bachelor	Some Master's	Master's
SampleSize	47	13	12	21	25	11	9
Range	0.95	0.95	0.75	0.95	0.95	0.9	0.8
Mean	0.52447	0.56923	0.62083	0.44286	0.502	0.53182	0.59444
Variance	0.09814	0.11356	0.06657	0.10707	0.09635	0.11314	0.09465
Std.Dev.	0.31327	0.33698	0.25802	0.32722	0.3104	0.33636	0.30766
Scenario 1: Overrun Factor							
Range	1.35	1.2	1.2	1.3	0.95	1.1	1
Mean	0.58404	0.61538	0.62083	0.52381	0.4624	0.55909	0.57778
Variance	0.19012	0.21724	0.16112	0.2044	0.09584	0.14691	0.18194
Std.Dev.	0.43603	0.46609	0.40139	0.45211	0.30959	0.38329	0.42655

Table 6–2: Descriptive Statistics for Overrun Probability and Overrun Factor under Scenario 1

Scenario 1 describes a situation where the person asked to give an estimate has experience with similar domain problems, but lacks experience with the technologies that will be used. Table 6-2 summarizes the descriptive statistics for the probability of experiencing effort overrun for a task and the overrun factor for this scenario. The overrun factor expresses the amount of time required additionally to the time estimated relative to that estimate. Thus, the time actually required to complete the task would be

Actual Duration = Estimated Duration + (Estimated Duration x Overrun Factor).

For the whole sample the average probability of underestimating required task effort is 52 percent, with a standard deviation of 31 percent. When a time overrun is incurred, the average overrun factor is 58 percent. These results indicate that estimations under scenario 1 create considerable risk exposure, as the probability of occurrence and the impact of occurrence are large.

Average Number of Similar Projects Required to Give More Reliable Estimates

On average, survey participants indicate they need to work on 3 (calculated average is 2.6) similar projects in order to give more reliable estimates. The values ranged from 1 project to 8 projects.

6.5.2 Scenario 2: Descriptive Statistics

Scenario 2: Probability of Effort Overrun							
	Whole Sample	5 years	10 years	more 10 years	Bachelor	Some Master's	Master's
SampleSize	47	13	12	21	25	11	9
Range	0.95	0.95	0.95	0.95	0.95	0.95	0.9
Mean	0.51723	0.53923	0.5625	0.47857	0.4624	0.54091	0.56667
Variance	0.10248	0.12567	0.09869	0.10289	0.09584	0.14091	0.08438
Std.Deviation	0.32013	0.35451	0.31415	0.32077	0.30959	0.37538	0.29047
Scenario 2: Overrun Factor							
Range	2.95	1.5	1.45	2.9	1.95	2.9	0.7
Mean	0.7383	0.70385	0.69167	0.77381	0.728	0.83182	0.67222
Variance	0.34904	0.28978	0.21038	0.5064	0.34418	0.68964	0.06257
Std.Deviation	0.5908	0.53831	0.45867	0.71162	0.58667	0.83044	0.25014

Table 6–3: Descriptive Statistics for Overrun Probability and Overrun Factor under Scenario 2

Scenario 2 describes the situation where the estimator is familiar with the technologies to be used in the implementation of the solution, however has not worked on a similar domain problem before. Table 6-3 summarizes the descriptive statistics for the probability of incurring in time overrun and the overrun factor under this scenario. Under scenario 2, the average overrun probability is 52 percent, with an average overrun factor of 73 percent. Estimates made under the situation described in scenario 2 have a high risk exposure, as the probability and severity of effort overruns are high.

Average Number of Similar Projects Required to Give More Reliable Estimates

Under scenario 2, survey participants indicated that on average they have to work in 3 (calculated average is 2.65) similar projects in order to be able to give more reliable estimates. The answers ranged from 1 to 10 projects.

6.5.3 Influence of Years of Experience and Level of Education

ANOVA tests were performed in order to compare the means between the three groups partitioned by years of experience for the probability of time overrun and the severity factors under both scenarios. In all cases, the ANOVA test did not indicate statistically significant differences between the means at significance level of alpha 0.05.

The ANOVA tests comparing the means between the three groups of education level also did not indicate statistically significant differences between these means at significance level of alpha 0.05.

From these tests, it seems that neither years of experience, nor the level of formal education influence the probability and severity of incurring in time overruns under either scenario.

6.5.4 Comparison between Scenario 1 and Scenario 2

T-tests were performed in order to determine if the means of overrun probability and the means of the overrun factors are different between scenario 1 and scenario 2. Prior to performing the t-tests, f-tests were performed in order to determine if the variances are equal between the scenarios. The samples could be fitted to the normal distribution at significance level alpha of 0.05 using the Kolmogorow-Smirnow, the Anderson Darling and the Chi-Squared tests (see appendix B).

For the probability of incurring in time overruns, the p-value of the f-test of 0.44. Thus we do not reject the hypothesis that the variances between scenario 1 and scenario 2 are equal. The t-test thus was performed assuming equal variances. The two tailed p-value of 0.91 indicates that we also do not reject the hypothesis that the two scenarios have equal means.

For the overrun factors, the p-value of the f-test is 0.02. Thus, it cannot be assumed that the two scenarios have equal variances. The t-test was performed assuming unequal variances.

In personal conversations with the study participants, many indicate that the overrun factor in the case of lack of domain experience is larger than in case of lack of experience with the technology to be used. Thus we proceed with a one tailed t test, testing the hypothesis that the mean for the overrun factor under scenario 2 is larger than under scenario 1.

Whether lack of experience with the domain problem could causes larger overruns than lack of experience with the specific technology could not be conclusively decided in this study: The one tailed p-value is 0.08. Applying the alpha of 0.05 selected for this study, the differences in means between scenario 1 and scenario 2 are not statistically relevant. However, the p value indicates that for alpha larger than 0.08 the mean of scenario 2 can be assumed to be larger than the mean of scenario 1. Further investigation with larger samples would be advised to fully clarify this point.

Comparison of Overrun Factors		
	Scenario 1	Scenario 2
Mean	0.5840	0.7383
Variance	0.1901	0.3490

6.6 Conclusions and Recommendations

Based on these findings, the most obvious recommendation is to carefully evaluate the level of experience with the domain problem and technology of the estimators when using the technique of expert consultation. In cases where the estimators lack domain knowledge or experience with a specific technology, more than one person should provide estimates and/or more than one estimation technique should be used and the differences between the estimates should be investigated.

Many project managers might underestimate the magnitude of uncertainty in the estimates introduced by lack of experience with the domain problem or the technology mix. Contingency buffers of 10 to 20 percent might not sufficient considering the large average overrun probabilities and overrun factors. On the same token, considering the significant uncertainty under these scenarios, it might not be a fair practice to hold estimators responsible for faulty estimates, for instance in performance evaluations.

Neither more formal education, nor more years of experience seem to have a significant impact on average overrun probability and overrun factors under either scenarios. This makes the need to formally train personnel in estimation techniques and to provide them structured opportunities to compare their estimates to actual results evident. Organizations will have to implement processes that will aid estimators in learning.

Also, reconciliation of estimates with historical data is important. Quantitative portfolio management and scientific methods of estimation, as described in [59] and [60] should be used more frequently.

6.6.1 Using Project Portfolio Management in order to Improve Expert Estimation Performance

Project Portfolio Management can be a useful tool to implement the aforementioned recommendations.

(I)Project Portfolio Management categorizes projects in order to trace them to strategic goals and in order to create comparability of projects within the same category.

Such systematic cataloguing of projects allows for identifying appropriate candidates for historic references in analogous estimates, because it defines project similarity. The application of a common categorization framework across all projects makes historic data searchable, readily accessible and comparable.

(II) PPM enforces the collection of project management information, including personnel management, time management and cost management across all projects.

Records of personnel that participated in past projects will support the identification of suitable experts for the estimation of current projects.

A core set of metrics used in Project Management and Portfolio Management are comparison of baselined effort estimates and actual effort expended. The data collected can be used as a historic reference for expert estimates, as input to calibrate parametric estimation models and as basis for providing feedback to experts on their estimation performance.

These core functions of Project Portfolio Management make it an enabler to better estimation processes.

6.7 Weaknesses of the Methodology

Since the investigation was carried out by means of questionnaire, the results obtained reflect the IT professionals perception of estimation uncertainty and learning effects in the estimation process. Therefore, results depend on the participants' ability to recall past experience. This recall might be incomplete, inaccurate, or tainted by psychological effects, such as self-perception or reluctance to admit failures in their full magnitude. In an investigation by M. Roy et al it was found that people underestimate the duration of future events, not only because they take a too optimistic outlook, but also because their memories are systematic underestimations of how long past events lasted [61]. This effect might be reflected in the survey participants' answers.

The fact that none of the employers of the survey participants offer ex post feedback on estimation accuracy, means the survey participants depend fully on their perception of magnitude of time overruns. They have had no opportunities to compare their conception with actual results.

Furthermore, the group of study participants could have been more carefully selected in order to control interdependencies between categories: Almost all more experienced programmers work with legacy systems, while younger programmers were more likely to work in an open systems environment. The possible effect of the different computing styles on answers could not be determined, yet might be influential. In addition, younger study participants are more likely to have a master's degree or to be currently pursuing a Master's degree, as competition for entry level positions is fierce in Puerto Rico. So the categories of years of experience and education level are not completely independent.

6.8 Future Research

As shown in the previous section, the methodology used in this research has certain weaknesses. For future research, it would be advisable to use a methodology that does not depend on the perception or memory of participants. In their research on the reliability of experts' estimates of task durations in software development projects, Hill et al. [56] asked experts for estimates in the initial stages of projects and then compared these estimates with the recorded actual outcomes. This approach could be repeated for various companies in Puerto Rico, thus giving more accurate results. Projects could be clustered according to dimensions of similarity and considered according to a timeline, in order to see if there are any organizational learning effects present. This approach was not taken in this work, because possible study participant organizations are reluctant to share such data.

Another possible methodology is to set up an experiment under laboratory conditions. Participants would be assigned a simple programming task and then be asked to implement the same. Estimates would then be compared with actual time required to fulfill the task. Repeated assignment of sufficiently similar tasks would then show learning effects. This approach was not taken, as it requires facilities in order to carry out the experiment and funds in order to compensate the time of study participants. Neither resource was available for this work.

CHAPTER 7
PROJECT PORTFOLIO MANAGEMENT IN THE IT OUTSOURCING VENDOR

Q4: How can portfolio management be used to create the IT outsourcing vendors value proposition and lower internal risk?

The following section outlines how Project Portfolio Management can be used as a tool to create the IT outsourcing vendors value proposition and lower risk. As shown in chapter 4, Project Portfolio Management is driven by strategy. Individual organizations' strategies can be very diverse. However, some generic strategies inherent to the business model of an IT outsourcing vendor can be identified. These generic strategies are mainly based in value creating activities as described by Levina [9]. We will show how Project Portfolio Management can be used as a tool to implement these generic strategies and realize the related goals. The following table 7-1 links the generic strategies identified with the value creating activities and their respective associated practices as described by Levina and shows what aspects of project portfolio management support the implementation of these strategies.

7.1 The Business Model of IT Outsourcing Vendors

IT outsourcing refers to contracting resources outside of the organization to perform part of or all IT activities of the client organization. These activities include the development and customization of software, application maintenance, production support and IT operations. Entities that contract outsourcing vendors can do so for various motives, such as looking for a cost advantage, gaining access to scarce specialized resources, economies of scale, and being able to focus on their core business. Decades of experience with IT outsourcing have shown that these sought after advantage might not always materialize and there are plenty of examples of escalating costs, unacceptable service levels, overdependence and lock-in in contracts, and loss of expertise in the client organization [11].

The success of an outsourcing relationship hinges on three factors: client characteristics, vendor characteristics and characteristics of the client-vendor relationship. A client characteristic that is crucial to the success of outsourcing arrangements is the clients ability to manage resources external to the firm. This management effort encompasses the technical aspect of the relationship, such as architecture management, as well as all aspects of contract administrations and communications management.

Generic Strategy	Value Creating Activity as identified in Levina [9]	Vendor Practices [9]	Portfolio Management Technique
-Build core competencies and lower process risk	-Develop methodology	-Project Office -Identification of best practices -Standardization of processes -Process documentation -Methodology training -Work tasks documentation	-Identify Performance Measures -Record measures across all projects -Regular evaluation of project performance -Regular comparison among projects
	-Personnel development	-Staffing Office -Promotion from within -Staff rotation -Junior Staff use -Redundant skill creation -Mentorship -Skill and project management training -Team-based environment -Collaboration across	-Assign staff from a global point of view -Include opportunity for learning as evaluation criterion for selection of projects
-Diversify external risk by limiting dependence on one client or industry and creating lasting client relationships	-Client relationship management	-SLA based contracts -Sharing efficiency benefits with clients -Management of clients goals and priorities -Communicating priorities and work status -Sharing expertise with clients Staff	-Balance Portfolio across industries and clients -Discover overdependencies -Performance Assessment
-Maintain Profitability			-Identify troubled projects -Financial analysis of projects -Risk management

Table 7–1: Generic Strategies of the IT Outsourcing Vendor with Corresponding Best Practices and Portfolio Management Techniques

The vendor-client relationship contains contractual elements, but also non-tangible elements, such as goodwill, communication process agility, balance of power between vendor and client, and interpersonal relationships. Vendor characteristics refer to competencies of the outsourcing vendor and the outsourcing vendor's value proposition to the client. Both, the client and the outsourcing vendor must understand the outsourcing value proposition and align their expectations. Outsourcing vendors need to be able to provide economic, technological and management benefits that will outweigh the costs of contract administration and the risks faced by the client organization [9].

7.2 Creating the Value Proposition

In their analysis of the vendor perspective of the value proposition of IT outsourcing, Levina and Ross, state that an IT outsourcing vendor "*can deliver value to its clients by developing a set of experience-based core competencies, which a) address client needs, b) exhibit complementarities, which result in efficient service delivery, and c) depend on the vendor's control over, and centralization of decision rights on a large number of projects from multiple clients" [9].*" They found that if an IT outsourcing vendor manages a large quantity, but also large variety of projects from various different clients, it enables the company to manage these projects in such a manner that it builds the companies core competencies. The more a vendor excels at these core competencies, the better the vendor-client relationship, the level of client satisfaction and with it the vendor's reputation. The better the vendor's reputation is, the more likely it is that the vendor be entrusted with more projects. Levina furthermore points out, that possible economies of scale are not a primary factor that would drive larger clients to opt for choosing outsourcing. These clients are large enough that they could produce said economies of scale in house. It is the fact that IT competencies are the core business of an outsourcing provider that puts them in the position to create value to large clients. These large clients might find, that the resources and policies needed to create IT competencies conflict with their efforts to optimize their core business. The core competencies of an outsourcing vendor contain the following categories: personnel development, process and methodology, and customer relationship management. All these competencies are required and they are complementary.

Competencies in customer relationship management imply understanding the client's needs and priorities. These needs should then be translated into service level agreements (SLA). SLAs are agreed upon metrics that allow for a quantitative evaluation of contract fulfillment. They also clearly assign roles and responsibilities and thus allow to clearly define boundaries and interface points between the organizations.

Personnel development means allowing team members to gain experience and build skills in terms of technological expertise and in terms of domain knowledge of the clients' industries. There are several techniques, such as mentoring, formal training, staff rotation, team work and collaboration across teams that support the creation of such experience based skills.

Process and methodology competencies refer to the definition of processes, experience and skills in the software engineering processes, project management processes and other management processes. Having efficient processes for IT in place is important to any organization. However, for an IT outsourcing vendor it is crucial. Their excellence and reliability of execution in terms of quality, cost and time *is* their product. Clients that opt for outsourcing expect a level of execution superior to what they could have achieved in house. Furthermore, in a fixed price contract scenario, which is frequently the case, any slippage in schedule and cost translates to diminished profits from that project, directly affecting the company's bottom line.

Developing and refining a methodology requires that the vendor be allowed to manage the project according to their methodology. The vendor is bound by their

statement of work and their Service Level Agreements (SLAs) on the client facing interface. However, how to manage projects in order to comply with these contractual agreements is controlled by the vendor. It is costly for an organization to develop such processes, as their development requires analyses, compliance audits, possibly the installation of management tools, and detailed documentation. However, since the outsourcing vendor applies this methodology to a large number of client projects, the initial up front investment is spread. Once the processes are developed, the marginal cost of applying the methodology to a new project is low.
Experience based skill development of employees is dependent on being able to provide a large number and great variety of experiences. An IT outsourcing vendor is better positioned than a non-IT centric company to provide opportunities for skill building. This is again, dependent on controlling a large number of different client projects.

These three areas of core competencies are complementary and mutually reinforcing:
Methodology Development and Customer Relationship Management: Improvements in methodology lead to higher efficiency and thus improved service levels that can be provided to the customer with no additional costs, thus improving the customer's perception of value created. Maintaining open communications and receiving regular feedback from customers allows the outsourcing vendor to further refine their methodology.

Personnel Development and Customer Relationship: Maintaining good customer relationships creates better acceptance of personnel development measures, such as rotating junior staff through various client projects. Personnel development measures allow for improvement of customer relationships, as staff with better competencies installs more trust in the customer.

Personnel Development and Methodology Development: Having a clear methodology in place allows personnel to be productive and clearly understand what is

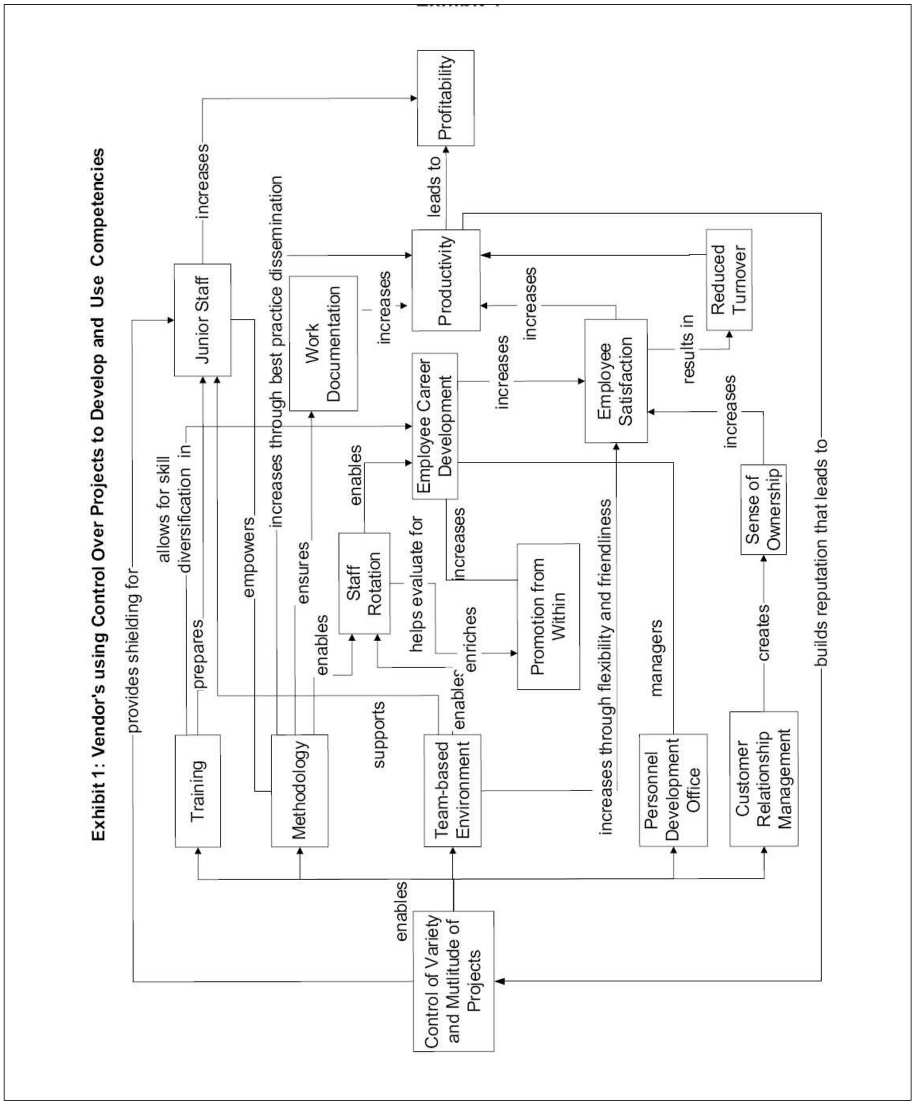

Figure 7–1: Creation of the IT Outsourcing Vendor's Value Proposition, adapted from [9]

expected of them. Experience based learning of personnel allows for better application of methodologies and less variation in the level of execution [9].

7.3 Generic Strategies for IT Outsourcing Vendors

Even though the possible strategic goals for outsourcing vendors are manifold, we can identify generic strategic goals that apply to most IT outsourcing vendors:
(1) Diversify external risk and achieve a wide client base to limit overdependence on one specific client,
(2) Build core competencies, through personnel development and process improvement
(3) Maintain profitability.

In organizations that are structured around mainly projects, project portfolio management is the tool for realizing strategy. An outsourcing vendors definition and use of portfolio management will be different from the approach described in the chapter on portfolio management or most practitioners literature. These frameworks are developed from the perspective of an organization whose core business is not the provision of IT services, but that depends on IT as a facilitator of their core business. These organizations view an IT portfolio as a portfolio of investments. In contrast, from the outsourcing vendors perspective the management of IT portfolios is the management of a collection of client projects and client relationships.
The selection of the projects that comprise the project portfolio is controlled through the selection of clients and decisions of whether to further pursue client relationships or to not enter into new contracts, once current contracts run out. The request for projects comes from the client organization. Therefore, the outsourcing vendor is not in complete control of the project selection process. The vendor is in full control of project prioritization and the distribution of resources among the different projects.
In the following we will describe how Project Portfolio Management can be applied

to each of these strategies.

7.3.1 Diversify external risk and achieve a wide client base to limit overdependence on one specific client

The strategy of not fully depending on only one client, or few main accounts, is intuitive. However, it should be noted, that in the case of an IT outsourcing vendor this needs to be weighed against the danger of losing focus or not being able to build the necessary expertise. IT projects are always also domain knowledge intensive. Thus, it is not practical for most but few very large IT outsourcing providers–such as Tata Consulting Services (TCS)–to cover a large array of domains. Most IT outsourcing providers will be some sort of specialists for a specific domain.

Thus, the diversification of risk is achieved considering dependence on specific clients, not for the decline of the industry the IT outsourcing company serves.

Project Portfolio analysis can reveal how balanced the IT outsourcing provider's portfolio is in terms of diversity of client base. Adequate data collection can support what-if analysis in the case of losing client contracts. Projects should be selected with the goal of creating a balanced portfolio in terms of client dependency in mind.

7.3.2 Build core competencies

Personnel in an IT outsourcing vendor needs to unite the following skills: technical expertise, domain knowledge, client-facing skills, process knowledge, estimating skills, project management skills, and knowledge of the client's applications and installations. Each organization should self-analyze and identify key skills required for their specific profile. Projects should then be categorized according to what competency is most required for them. A deliberate effort should be made to team up experienced personnel with personnel that needs to develop specific skills.

Portfolio management can aid in this process, as it would consider the allocation of personnel at a global level and support more strategic planning of career paths. As pointed out in the introductory section of this work, superior execution is an integral part of the outsourcing vendor's value proposition. Reliable estimates are an integral part of that, as they are needed in order to prepare proposals and to manage projects. In chapter 6 on expert estimation, we could see that it is important to allow expert estimators structured feedback on their past estimates. For this, it the organization needs to record such data. Also, having a database of estimated and actual task durations for similar past projects allows for analogies experts can use to validate their estimates. Thus implementing the portfolio management approach will create these process assets.

In order to monitor and improve processes, the processes in question have to be defined and key measures associated with them have to be identified. The collection of relevant information at portfolio level is the basis for the application of quantitative portfolio management. In an ideal situation is that data needed to assess and compare project performance is uniformly and historically continuously available from a portfolio database. Key project information includes, but is not limited to short description, comparable size measure, technology category, client, software engineering specific indicators (testing strategy used, implementation methodology, etc.) initiation date, finish date, baselines and updates thereof, staff buildup, staffing levels at various phases, total development costs, annual costs of operations, total cost of ownership, scores on various risks factors, net present value, return on investment, internal rate of return among others. Having such data available is necessary in order to be able to adequately compare projects and measure the performance of the organization as a whole. The data can also be used in order to calibrate prediction models and thus aid at making estimates and risk analysis of future projects more reliable [60].

7.3.3 Maintain profitability

Project portfolio management offers the tools to analyze the company's portfolio in terms of overall profitability, but also on the basis of overall risk. As demonstrated in [60], [62], data collected on a portfolio can help estimate the yield of the same and to assess risk factors inherent to it. Being able to identify unusually risky or unprofitable or unusually profitable projects will help to balance the portfolio and minimize the company's risk exposure. In order to determine what project is unusually risky or unusually profitable, it is necessary to take a portfolio view of all projects.

7.4 Conclusions

In contrast to an internal IT department, where all projects spring from business needs of the parent organization and can be traced to its strategy, projects in an IT outsourcing vendor are a collection of other organization's projects. IT outsourcing vendors are an organization of their own right, not a support function of a larger organization.

As any business entity, IT outsourcing vendors need to engage in environmental scanning of opportunities and threats, and internal assessment of strengths and weaknesses in order to develop a business strategy. This strategy should encompass the generic strategies described in the previous section and concerns such as market positioning and product development among others. The fact that the operational activity of an IT outsourcing provider consists of the assessment and implementation of client projects, might blur the organization's strategy. Project Portfolio Management is a useful tool for aligning client projects with the IT outsourcing vendor's strategy, because its main function is categorizing projects and tracing them to strategic goals. It allows for a global perspective of all projects and introduces selection criteria beyond generating a profit. Thus PPM allows the IT outsourcing vendor to integrate client projects with their own strategy. It becomes evident that running such a business as a projectized organization and applying Project Portfolio Management to these projects seems a logical and viable option for the implementation of organizational strategy and ultimately for value creation.

Besides being able to trace projects to strategic goals, Project Portfolio Management helps to define relevant measures, collect the data required and analyze project data in a global context.

The data collection effort involved in Project Portfolio Management inherently drives the organization to reach higher maturity levels of their organizational processes.

Having relevant historic data available is a prerequisite for process analysis and improvement, including the process of determining project task estimates. We could show that estimation accuracy does not necessarily improve with growing on the experience, but rather depends on concrete ex post evaluation of estimation accuracy. In addition, having historic data available for analogous estimates further enhances the estimation accuracy. Thus, Project Portfolio Management can make a significant contribution to this central process.

Beyond process monitoring and improvement, Project Portfolio Management can help identify overdependencies on certain accounts or industries.
PPM helps identify underperforming projects in comparison to the other projects run in the organization.

Finally, Project Portfolio Management allows to include projects in the organization's portfolio that, while not as profitable as alternative projects, will allow the company to grow their experience and knowledge in a key area. Only a global view of projects, with all aspects of the organizational strategy in mind, as could be defined in a balanced score card, will allow for a optimal portfolio selection.

APPENDICES

APPENDIX A
APPENDIX A : QUESTIONNAIRE USED

UNIVERSIDAD DE PUERTO RICO

RECINTO UNIVERITARIO DE MAYAGÜEZ

COLEGIO DE ADMINISTRACION DE EMPRESAS

DEPARTAMENTO DE ESTUDIOS GRADUADOS

INSTRUCTION SHEET FOR SURVEY ON ACCURACY OF ESTIMATES AND EXPERIENCE EFFECTS

Instructions:

* Read this instruction sheet. If you agree to participate, sign the consent sheet and return it to envelope A.
* Please fill out all points of the survey.
* DO NOT put your name on the survey.

* Put the filled out survey in envelope B.

Contact Information:

The author of this questionnaire is Susanne Halstead. You can reach me at Halstead.Susanne@uprm.edu or 787 475 7549

Purpose of Data Collection:

The purpose of this questionnaire is to capture the experience of professionals in the field of information technology with regards to how accurate time estimates for project tasks are, considering the dimensions of experience with technologies and business problems.

The data collected through this questionnaire will be incorporated into a simulation model to be included in a Master's thesis. The data collected will not be used for any other purpose.

Protection of Privacy of Participants:

Participation in this survey is voluntary. The questionnaire sheets are all identical and bear no special marks that would allow the identification of individual survey participants or their organization. Professionals surveyed will be from various organizations, so that the information collected here is not specific to any organization.

The questionnaires and consent sheets will be stored a maximum of two years after collection (30. March 2011) in the office of Prof. Rosario Ortiz in the UPR Mayagüez Campus. They will be shredded after this date.

Time Requirements

It will take you between 5 and 15 minutes to fill out the questionnaire.

Reward

There is NO reward payable to participants nor can you expect any other type of benefit from participating.

CONSENT SHEET

I have read and understood the instruction sheet for the questionnaire SURVEY ON ACCURACY OF ESTIMATES AND EXPERIENCE EFFECTS. All questions I had about the purpose of the survey, possible benefits for me, and the protection of my personal information have been answered in the instruction sheet and/or by the presenter of the survey. I have also been given the copy of the questionnaire for my evaluation prior to consenting to my participation.

I voluntarily participate in the survey.

I am aware that : I can withdraw from participating at any time prior to turning in the questionnaire. Once turned in, the questionnaire cannot be given back, as it is undistinguishable from the other questionnaires collected. Questionnaires are anonymous and bear no special marks that would allow the identification of individual persons or organizations. The questionnaire is for academic use exclusively and will not be shared with third parties (e.g. employers). Results of the questionnaire will be published in a master's thesis. These results are summaries and analyses of the questionnaires. The work will not contain names of survey participants. The questionnaires and consent sheets will be stored a maximum of two years after collection (30. March 2011) in the office of Prof. Rosario Ortiz in the UPR

Mayagüez Campus. They will be shredded after this date.

Signed:

Name:

SURVEY ON ACCURACY OF ESTIMATES AND EXPERIENCE EFFECTS

General Information

Education	◯ Some College	◯ Bachelor's Degree
	◯ Some Master's	◯ Master's Degree
	◯PhD	◯Other, Specify:
Work Experience	◯ $<$ 1 yrs	◯ 1-5 yrs
	◯ 6-10 yrs	◯ $>$ 10 yrs

I . When faced with a task where you understand the business requirements well and have experience solving similar problems, however, the technology mix (e.g. databases, programming languages, frameworks, interfaces) is NEW to you:

1. How frequently do you need more time to complete your tasks than was allocated in the estimates?

Please provide percentage estimate. Valid values are 0% through 100 % ....

2. If you use more time than was allocated to the task, by what proportion do you typically overrun the time budget?

Please provide percentage estimate. Percentage values greater than 100 are possible. For example a value of 120% means that you needed all the time allocated for the task and an additional 120%. So if 10 hours were estimated for a task, you actually required 22 hours [10+12] ....

3. How many similar projects would you have to work in order to significantly reduce this overrun and give more accurate estimates?
Please state number:

II . When faced with a task where you DO NOT understand the business requirements well and DO NOT have experience solving similar problems, however, the technology mix (e.g. databases, programming languages, frameworks, interfaces) is FAMILIAR to you:

1. How frequently do you need more time to complete your tasks than was allocated in the estimates?
Please provide percentage estimate. Valid values are 0% through 100 % ....

2. If you use more time than was allocated to the task, by what proportion do you typically overrun the time budget?
Please provide percentage estimate. Percentage values greater than 100 are possible. For example a value of 120% means that you needed all the time allocated for the task and an additional 120%. So if 10

hours were estimated for a task, you actually required 22 hours [10+12]

....

3. How many similar projects would you have to work in order to significantly reduce this overrun and give more accurate estimates? Please state number:

APPENDIX B
APPENDIX B: STATISTICAL ANALYSIS OF QUESTIONNAIRE DATA

B.1 Raw Data

The following table shows the raw data collected in the questionnaire. There are 47 usable data sets of 47 questionnaires distributed and returned. All questionnaires that were distributed were returned.

The first column, *DS*, indicates the data set number. This number was assigned to the questionnaires in the order they were found in the collection envelope after all questionnaires were returned. This order does not allow the identification of individual study participants. It is only used to be able to match the data set with the questionnaire it was transcribed from.

The columns marked with *Education* indicate the level of relevant education the study participant has obtained. The level marked in the questionnaire is marked with the value 1 in the data sets. Possible values are *some Bachelor's* studies, a *Bachelor's Degree*, *some Master's Degree* , *Master's Degree*, *PhD*, and *other*.

Of the 47 study participants, 25 have a Bachelor's degree in Computer Science, Computer Engineering or Management Information Systems, 11 are in the process of completing a Master's degree in one of these fields, and 9 have completed a Master's degree. 2 Study participants answered 'other'. The columns marked with *Experience* indicate the years of work experience the study participant has. Possible categories were *0-1 year*, *1-5 years*, *6-10 years*, *more than 10 years*. Of the 47 study participants, 13 had 2 to 5 years of experience working, 12 study participants have 6-10 years of experience working, and 21 indicate more than 10 years of relevant work experience. In one questionnaire (Data Set 29) this field was not marked.

The next three columns describe scenario 1, which consists of the study participant being asked to provide a task duration estimate on a project where she/he knows the business problem well, however is faced with a new technology mix. Under this scenario, the column of *Frequency* indicates the probability value of experiencing a time overrun in tasks during the actual execution phase as compared with the original estimate. The column *Impact* indicates the factor of magnitude of this overrun. The calculation of the actual time required to carry out the task would be

Actual Duration = Estimated Duration + (Estimated Duration x Overrun Factor).

The column *Number of Projects* indicates the number of similar projects the study participants understands she/he would have to work in order to give accurate estimates.

The next three columns describe scenario 2, which consists of the study participant being asked to provide a task duration estimate on a project where she/he knows the technology mix well, however is faced with a new domain problem. The nomenclature of columns is the same as under scenario 1.

Data Set	Education some Bachelor	Bachelor	some Master's	Master's	PhD	other	Experience 0-1	2-5	6-10	¿10	Scenario 1 Probability	Impact	Projects	Scenario 2 Probability	Impact	Projects
1	0	1	0	0	0	0	0	1	0	0	0.15	0.15	3	0.05	0.1	1
2	0	1	0	0	0	0	0	0	1	0	0.4	0.5	3	0.9	1	-
3	0	0	0	0	0	1	0	0	1	0	0.8	0.6	2	0.8	1	2
4	0	1	0	0	0	0	0	0	1	0	0.7	0.3	8	0.4	0.2	3
5	0	0	0	1	0	0	0	0	1	0	0.2	0.2	1	0.75	0.5	1
6	0	0	0	0	0	1	0	0	0	1	0.1	0.15	2	0.9	0.3	3
7	0	1	0	0	0	0	0	1	0	0	0.05	1.05	1	0.1	0.15	2
8	0	0	0	1	0	0	0	0	0	1	0.2	0.3	-	0.6	0.5	3
9	0	1	0	0	0	0	0	0	0	1	0.8	1.1	2	0.75	1.05	4
10	0	0	1	0	0	0	0	1	0	0	0.6	1.3	3	0.2	1.05	2
11	0	1	0	0	0	0	0	1	0	0	1	0.15	3	0.8	0.1	2
12	0	0	1	0	0	0	0	1	0	0	0.85	0.25	3	0.9	0.25	3
13	0	0	0	1	0	0	0	1	0	0	0.9	1	3	1	1.1	3
14	0	0	0	1	0	0	0	1	0	0	0.75	0.8	3	0.3	0.4	1
15	0	0	1	0	0	0	0	1	0	0	0.6	0.4	3	1	1.4	2
16	0	1	0	0	0	0	0	1	0	0	0.5	1.2		0.2	0.5	2
17	0	0	1	0	0	0	0	0	0	1	0.4	0.5	2	1	3	
18	0	0	1	0	0	0	0	0	1	0	0.75	0.25	3	1	0.5	3
19	0	0	0	1	0	0	0	1	0	0	1	0.1	2	0.5	0.5	1
20	0	1	0	0	0	0	0	0	0	1	0.5	1.2	3	0.75	1.5	6
21	0	0	0	1	0	0	0	0	0	1	0.25	0.5	3	0.35	0.65	3
22	0	0	1	0	0	0	0	0	0	1	0.1	0.2	2	0.5	0.8	4
23	0	1	0	0	0	0	0	1	0	0	0.2	0.4	2	0.66	1.6	2
24	0	0	1	0	0	0	0	0	0	1	0.05	1.05	2	0.05	1.05	2
25	0	0	0	1	0	0	0	0	0	1	0.6	1.1	2	0.1	1.05	2
26	0	0	1	0	0	0	0	0	1	0	0.5	0.5	3	0.25	0.1	
27	0	1	0	0	0	0	0	0	0	1	0.9	0.2	2	0.8	0.2	3
28	0	1	0	0	0	0	0	0	1	0	0.8	1.25	2	0.6	1.2	2
29	0	1	0	0	0	0					0.5	1	2	0.5	1	1
30	0	0	1	0	0	0	0	0	1	0	0.95	1	4	0.6	0.5	3
31	0	1	0	0	0	0	0	0	0	1	0.7	1.4	2	0.3	1.2	3
32	0	1	0	0	0	0	0	0	0	1	0.8	0.2	2	0.9	0.5	1
33	0	1	0	0	0	0	0	0	0	1	0.4	1.25	2	1	2	3
34	0	1	0	0	0	0	0	1	0	0	0.15	1.1	2	0.4	1.4	2
35	0	1	0	0	0	0	0	0	0	1	0.1	0.1	1	0.15	0.3	1
36	0	0	0	1	0	0	0	1	0	0	0.65	0.1	2	0.9	0.6	2
37	0	1	0	0	0	0	0	0	1	0	0.4	0.6	3	0.75	1	5
38	0	1	0	0	0	0	0	0	0	1	0.1	0.1	2	0.5	0.75	1
39	0	1	0	0	0	0	0	0	1	0	0.9	1.1	3	0.05	1.5	2
40	0	1	0	0	0	0	0	0	1	0	0.25	0.05	3	0.05	0.05	3
41	0	0	1	0	0	0	0	0	0	1	0.1	0.2	3	0.25	0.3	2
42	0	0	0	1	0	0	0	0	1	0	0.8	1.1	3	0.6	0.75	3
43	0	1	0	0	0	0	0	0	0	1	0.2	0.2	3	0.2	0.2	3
44	0	0	1	0	0	0	0	0	0	1	0.95	0.5	3	0.2	0.2	2
45	0	1	0	0	0	0	0	0	0	1	0.75	0.5	3	0.25	0.3	3
46	0	1	0	0	0	0	0	0	0	1	1	0.1	3	0.1	0.1	2
47	0	1	0	0	0	0	0	0	0	1	0.3	0.15	5	0.4	0.3	10
Total	0	25	11	9	0	2	0	13	12	21						

B.2 Descriptive Statistics

The following tables summarize the descriptive statistics for schedule overrun probability and severity in both scenarios. The descriptive statistics were calculated for the whole sample and the subsets that can be derived by grouping participants either education level or years of relevant work experience. The categories of *some Bachelor's* degree *Other* for the level of education and the category of *0-1* for years of experience were left out, for the sample size was too small or zero for these categories. The descriptive statistics were generated using the statistics tool 'Easy Fit' [63].

Sample	all	1-5	6-10	greater 10	Bachelor	Some M	Master's
SampleSize	47	13	12	21	25	11	9
Range	0.95	0.95	0.75	0.95	0.95	0.9	0.8
Mean	0.52447	0.56923	0.62083	0.44286	0.502	0.53182	0.59444
Variance	0.09814	0.11356	0.06657	0.10707	0.09635	0.11314	0.09465
Std.Deviation	0.31327	0.33698	0.25802	0.32722	0.3104	0.33636	0.30766
Coef.ofVariation	0.59731	0.592	0.4156	0.73888	0.61833	0.63247	0.51755
Std.Error	0.0457	0.09346	0.07448	0.0714	0.06208	0.10142	0.10255

Descriptive Statistics for Probability of Time Overrun under Scenario 1

Sample	all	1-5	6-10	greater 10	Bachelor	Some M	Master's
SampleSize	47	13	12	21	25	11	9
Range	1.35	1.2	1.2	1.3	0.95	1.1	1
Mean	0.58404	0.61538	0.62083	0.52381	0.4624	0.55909	0.57778
Variance	0.19012	0.21724	0.16112	0.2044	0.09584	0.14691	0.18194
Std.Deviation	0.43603	0.46609	0.40139	0.45211	0.30959	0.38329	0.42655
Coef.ofVariation	0.74657	0.7574	0.64654	0.86312	0.66952	0.68555	0.73826
Std.Error	0.0636	0.12927	0.11587	0.09866	0.06192	0.11557	0.14218

Descriptive Statistics for Severity of Time Overrun under Scenario 1

Sample	all	1-5	6-10	greater 10	Bachelor	Some M	Master's
SampleSize	47	13	12	21	25	11	9
Range	0.95	0.95	0.95	0.95	0.95	0.95	0.9
Mean	0.51723	0.53923	0.5625	0.47857	0.4624	0.54091	0.56667
Variance	0.10248	0.12567	0.09869	0.10289	0.09584	0.14091	0.08438
Std.Deviation	0.32013	0.35451	0.31415	0.32077	0.30959	0.37538	0.29047
Coef.ofVariation	0.61892	0.65743	0.5585	0.67026	0.66952	0.69398	0.5126
Std.Error	0.0467	0.09832	0.09069	0.07	0.06192	0.11318	0.09682

Descriptive Statistics for Probability of Time Overrun under Scenario 2

Sample	all	1-5	6-10	greater 10	Bachelor	Some M	Master's
SampleSize	47	13	12	21	25	11	9
Range	2.95	1.5	1.45	2.9	1.95	2.9	0.7
Mean	0.7383	0.70385	0.69167	0.77381	0.728	0.83182	0.67222
Variance	0.34904	0.28978	0.21038	0.5064	0.34418	0.68964	0.06257
Std.Deviation	0.5908	0.53831	0.45867	0.71162	0.58667	0.83044	0.25014
Coef.ofVariation	0.80022	0.76481	0.66314	0.91963	0.80587	0.99835	0.37211
Std.Error	0.08618	0.1493	0.13241	0.15529	0.11733	0.25039	0.08338

Descriptive Statistics for Severity of Time Overrun under Scenario 2

B.3 Possible Probability Distribution

Using the tool 'Easy Fit' [63] goodness of fit tests for various probability distributions were done. The following table shows the top three possible distributions ranked according to the Kolmogorow Smirnow test for the probability of time overrun and the severity of time overrun in scenario 1 and scenario 2.

Distribution	Kolmogorow Smirnow		Anderson Darling		Chi Square	
	Statistic	Rank	Statistic	Rank	Statistic	Rank
Scenario 1, Probability of Time Overrun						
Gen. Pareto	0.07339	1	0.46131	1	1.6427	2
Uniform	0.07559	2	0.4731	3	1.64157	1
Error	0.07566	3	0.47259	2	1.6446	3
Scenario 1, Severity of Time Overrun						
Johnson SB	0.10374	1	21.334	54	N/A	
Gen. Gamma (4P)	0.13357	2	1.0396	1	10.031	41
Beta	0.13768	3	1.2222	3	4.1296	28
Scenario 2, Probability of Time Overrun						
Johnson SB	0.06497	1	0.31868	1	1.3561	5
Gen. Pareto	0.07513	2	0.4365	4	0.8549	3
Error	0.07857	3	0.43089	2	0.78812	2
Scenario 2, Severity of Time Overrun						
Pert	0.10509	1	0.72912	20	3.4571	26
Dagum (4P)	0.10627	2	4.2228	46	N/A	
Gen. Pareto	0.10647	3	0.38696	1	0.91528	7

Possible Probability Distributions

B.4 Goodness of Fit Test for the Normal Distribution

In order to verify whether the t-Test and the ANOVA test can be used for further analysis of the data, goodness of fit tests for the Normal Distribution were run in the Easy Fit [63] tool. At a significance level alpha of 0.05, the Kolmogorov Smirnov Test, Darling Anderson test and the Chi Square test do not indicate that we can reject the hypothesis that the data follows a normal distribution. The following tables summarize the results of these tests

	Kolmogorov-Smirnov	Anderson-Darling	Chi-Squared
Sample Size	47	47	47
Degrees of Freedom	n.a.	n.a.	4
P-Value	0.38506	n.a.	0.08362
Statistic	0.12867	1.1858	8.2266
alpha	0.05	0.05	0.05
Critical Value	0.1942	2.5018	9.4877
Reject Hypothesis?	No	No	No

Goodness of Fit Tests for Normal Distribution for Probability of Time Overrun, Scenario 1

	Kolmogorov-Smirnov	Anderson-Darling	Chi-Squared
Sample Size	47	47	47
Degrees of Freedom	n.a.	n.a.	2
P-Value	0.10926	n.a.	0.11905
Statistic	0.17217	2.4512	4.2565
alpha	0.05	0.05	0.05
Critical Value	0.1942	2.5018	5.9915
Reject Hypothesis?	No	No	No

Goodness of Fit Tests for Normal Distribution for Severity of Time Overrun, Scenario 1

	Kolmogorov-Smirnov	Anderson-Darling	Chi-Squared
Sample Size	47	47	47
Degrees of Freedom	n.a.	n.a.	4
P-Value	0.39025	n.a.	0.19062
Statistic	0.12812	1.0472	6.1165
alpha	0.05	0.05	0.05
Critical Value	0.1942	2.5018	9.4877
Reject Hypothesis?	No	No	No

Goodness of Fit Tests for Normal Distribution for Probability of Time Overrun, Scenario 2

	Kolmogorov-Smirnov	Anderson-Darling	Chi-Squared
Sample Size	47	47	47
Degrees of Freedom	n.a.	n.a.	3
P-Value	0.12822	n.a.	0.42091
Statistic	0.16729	1.3394	2.8157
alpha	0.05	0.05	0.05
Critical Value	0.1942	2.5018	7.8147
Reject Hypothesis?	No	No	No

Goodness of Fit Tests for Normal Distribution for Severity of Time Overrun, Scenario 2

B.5 t-Tests to Compare the Scenarios

In order to analyze if there is any significant difference between scenario 1 and scenario 2 in terms of probability of schedule overrun and severity of schedule overrun, two t-tests were performed, using the Excel data analysis plugin.

B.5.1 Comparison of Probability of Schedule Overrun

In order to perform the t-test, first an f-test of variances is calculated. The following table summarizes the results of the f-test.

F-Test Two-Sample for Variances	Variable 1	Variable 2
Mean	0.524468085	0.517234043
Variance	0.098138298	0.102481314
Observations	47	47
df	46	46
F	0.957621389	
P(F <=f) one-tail	**0.441942783**	
F Critical one-tail	0.612570986	

Comparing Probability of Schedule Overrun between Scenario 1 and Scenario 2

Hypothesis: The samples have equal variances.
Alt. Hypothesis: The samples have different variances.

The probability value P(F<=f)=0.44 is larger than the level of significance, α of 0.05, thus we accept the hypothesis that the probability distribution of the occurrence of schedule overruns in the two samples have equal variances. We therefore proceed with the t-Test, assuming equal variances.

We can use a two sample t-Test: Even though the samples are provided by the same person, the survey participants were asked to assess two separate, mutually exclusive scenarios.

t-Test: Two-Sample Assuming Equal Variances	Variable 1	Variable 2
Mean	0.524468085	0.517234043
Variance	0.098138298	0.102481314
Observations	47	47
Pooled Variance	0.100309806	
Hypothesized Mean Difference	0	
df	92	
t Stat	0.11072439	
P(T<=t) one-tail	0.456038048	
t Critical one-tail	1.661585397	
P(T<=t) two-tail	**0.912076096**	
t Critical two-tail	1.986086272	
Comparing Probability of Schedule Overrun between Scenario 1 and Scenario 2		

Hypothesis: The samples have equal means.
Alt. Hypothesis: The samples have different means.

The probability value for P(T<=t) two tail is 0.9121, thus greater than α of 0.05. We therefore accept the hypothesis that the two samples have equal means.

As a result of the f-Test and the t-Test we cannot reject the hypotheses that the probability distributions of incurring schedule overrun have equal means and variances under both scenarios.

B.5.2 Comparison of Severity of Schedule Overrun

In order to determine the appropriate t-test, first an f-test of variances is calculated. The following table summarizes the results of the f-test.

F-Test Two-Sample for Variances	Variable 1	Variable 2
Mean	0.584042553	0.738297872
Variance	0.190120259	0.349044866
Observations	47	47
df	46	46
F	0.544687167	
P(F <=f) one-tail	**0.020987837**	
F Critical one-tail	0.612570986	
Comparing Severity of Schedule Overrun between Scenario 1 and Scenario 2		

Hypothesis: The samples have equal variances.
Alt. Hypothesis: The samples have different variances.

The probability value P(F<=f)=0.021 is smaller than the level of significance, α of 0.05, therefore we reject the null hypothesis that the two samples have equals variances, and accept the alternative hypothesis that the samples have different variances. We therefore proceed with a two sample t-test, assuming unequal variances in the samples.

t-Test: Two-Sample Assuming Unequal Variances	Variable 1	Variable 2
Mean	0.584042553	0.738297872
Variance	0.190120259	0.349044866
Observations	47	47
Hypothesized Mean Difference	0	
df	85	
t Stat	-1.440217823	
P(T<=t) one-tail	**0.076739498**	
t Critical one-tail	1.6629785	
P(T<=t) two-tail	0.153478996	
t Critical two-tail	1.988267868	

Comparing Severity of Schedule Overrun between Scenario 1 and Scenario 2

Hypothesis: The samples have equal means.
Alt. Hypothesis: The samples have different means.

The observed variance and mean of overrun severity under scenario 2 is larger than under scenario 1. This coincides with observations expressed by study participants in personal interviews. We therefore consider the one-tailed t-test to test the hypothesis that the sample mean of the overrun severity under scenario 2 is larger than the sample mean under scenario 1. The probability value for P(T<=t) one tail is 0.077, thus greater than α of 0.05. We therefore have to reject the null hypothesis that the two samples have equal means, and reject the alternative hypothesis that the means are different. However,for confidence level α of 0.10 we would accept the alternative hypothesis that the mean is larger under scenario 2 than under scenario 1.

B.6 ANOVA Tests for Detecting Differences between Populations

In the following section, we analyze whether the probability distributions vary if the data sets are grouped by either level of education or years of experience.

The results of the ANOVA tests can be summarized as follows:

Partition: Measure:	Level of Education	Years of Work Experience
Probability of Schedule Overrun	do not reject hypothesis	do not reject hypothesis
Severity of Schedule Overrun	do not reject hypothesis	do not reject hypothesis
Hypothesis: The groups have equal means.		

B.6.1 Comparison of Groups with Different Education Levels

The following is the result of the ANOVA test for the probability of schedule overrun for the groups of survey participants with Bachelor's degrees, some Master's degrees or Master's degrees.

Scenario 1, Probability of Schedule Overrun

SUMMARY				
Groups	Count	Sum	Average	Variance
Bachelor's	25	12.55	0.502	0.09635
Some Master's	11	5.85	0.531818182	0.113136364
Master's	9	5.35	0.594444444	0.094652778
ANOVA				
Source of Variation	SS	df	MS	F
Between Groups	0.056791919	2	0.02839596	0.283892958
Within Groups	4.200985859	42	0.100023473	
Total	4.257777778	44		
	P-value	F crit		
	0.754280363	3.219942293		

The F value is smaller than F critical. Therefore the differences between the groups are not statistically relevant for the level of education and the probability distribution of schedule overrun under scenario 1.

Scenario 1, Severity of Schedule Overrun

SUMMARY				
Groups	Count	Sum	Average	Variance
Bachelor's	25	15.35	0.614	0.233441667
Some Master's	11	6.15	0.559090909	0.146909091
Master's	9	5.2	0.577777778	0.181944444
ANOVA				
Source of Variation	SS	df	MS	F
Between Groups	0.025753535	2	0.012876768	0.063423081
Within Groups	8.527246465	42	0.203029678	
Total	8.553	44		
	P-value	F crit		
	0.93863602	3.219942293		

The F value is smaller than F critical. Therefore the differences between the groups are not statistically relevant for the level of education and the distribution of schedule overrun severity under scenario 1.

Scenario 2, Probability of Schedule Overrun

SUMMARY				
Groups	Count	Sum	Average	Variance
Bachelor's	25	11.56	0.4624	0.095844
Some Master's	11	5.95	0.540909091	0.140909091
Master's	9	5.1	0.566666667	0.084375
ANOVA				
Source of Variation	SS	df	MS	F
Between Groups	0.093484202	2	0.046742101	0.447767543
Within Groups	4.384346909	42	0.104389212	
Total	4.477831111	44		
	P-value	F crit		
	0.642068281	3.219942293		

The F value is smaller than F critical. Therefore the differences between the groups are not statistically relevant for the level of education and the distribution of schedule overrun severity under scenario 2.

Scenario 2, Severity of Schedule Overrun

SUMMARY				
Groups	Count	Sum	Average	Variance
Bachelor's	25	18.2	0.728	0.344183333
Some Master's	11	9.15	0.831818182	0.689636364
Master's	9	6.05	0.672222222	0.062569444
ANOVA				
Source of Variation	SS	df	MS	F
Between Groups	0.137458586	2	0.068729293	0.184362998
Within Groups	15.65731919	42	0.372793314	
Total	15.79477778	44		
	P-value	F crit		
	0.832303247	3.219942293		

The F value is smaller than F critical. Therefore the differences between the groups are not statistically relevant for the level of education and the severity of schedule overrun severity under scenario 2.

B.6.2 Comparison of Groups with Different Years of Experience

The following is the result of the ANOVA test for the probability of schedule overrun for the groups of survey participants with 1 to 5 years of work experience, 6 to 10 years of work experience, or more than 10 years of work experience.

Scenario 1, Probability of Schedule Overrun

SUMMARY				
Groups	Count	Sum	Average	Variance
1-5 Years	13	7.4	0.569230769	0.113557692
6-10 Years	12	7.45	0.620833333	0.06657197
more than 10	21	9.3	0.442857143	0.107071429
ANOVA				
Source of Variation	SS	df	MS	F
Between Groups	0.277337454	2	0.138668727	1.407501088
Within Groups	4.236412546	43	0.098521222	
Total	4.51375	45		
	P-value	F crit		
	0.255803214	3.214480328		

The F value is smaller than F critical. Therefore the differences between the groups are not statistically relevant for years of relevant work experience and the probability distribution of schedule overrun under scenario 1.

Scenario 1, Severity of Schedule Overrun

SUMMARY				
Groups	Count	Sum	Average	Variance
1-5 Years	13	8	0.615384615	0.21724359
6-10 Years	12	7.45	0.620833333	0.161117424
more than 10	21	11	0.523809524	0.204404762
ANOVA				
Source of Variation	SS	df	MS	F
Between Groups	0.101440018	2	0.050720009	0.257574176
Within Groups	8.467309982	43	0.196914186	
Total	8.56875	45		
	P-value	F crit		
	0.774108298	3.214480328		

The F value is smaller than F critical. Therefore the differences between the groups are not statistically relevant for years of relevant work experience and the severity of schedule overrun under scenario 1.

Scenario 2, Probability of Schedule Overrun

SUMMARY				
Groups	Count	Sum	Average	Variance
1-5 Years	13	7.01	0.539230769	0.125674359
6-10 Years	12	6.75	0.5625	0.098693182
more than 10	21	10.05	0.478571429	0.102892857
ANOVA				
Source of Variation	SS	df	MS	F
Between Groups	0.062262506	2	0.031131253	0.287782963
Within Groups	4.651574451	43	0.10817615	
Total	4.713836957	45		
	P-value	F crit		
	0.751357315	3.214480328		

The F value is smaller than F critical. Therefore the differences between the groups are not statistically relevant for years of relevant work experience and the probability distribution of schedule overrun under scenario 2.

Scenario 2, Severity of Schedule Overrun

SUMMARY				
Groups	Count	Sum	Average	Variance
1-5 Years	13	9.15	0.703846154	0.289775641
6-10 Years	12	8.3	0.691666667	0.210378788
more than 10	21	16.25	0.773809524	0.506404762
ANOVA				
Source of Variation	SS	df	MS	F
Between Groups	0.066517359	2	0.03325868	0.08983429
Within Groups	15.9195696	43	0.370222549	
Total	15.98608696	45		
	P-value	F crit		
	0.914253739	3.214480328		

The F value is smaller than F critical. Therefore the differences between the groups are not statistically relevant for years of relevant work experience and the severity of schedule overrun under scenario 1.

REFERENCE LIST

[1] Project Management Institute. *A Guide to the Project Management Body of Knowledge (PMBOK Guide), Fourth Edition* . Project Management Institute, 2009.

[2] Barry Boehm. Software engineering economics. *IEEE Transactions on Software Engineering*, 10(1), 1984.

[3] Cristine Martins Gomes de Gusmao and Hermano Perrelli de Moura. ISO, CMMI, and PMBOK Risk Management: a Comparative Analysis. *The International Journal of Applied Management Technology*, 1(1), 2003.

[4] Barry Boehm. Value based software engineering. *Software Engineering Notes*, 28(2), 2003.

[5] Stephen S. Bonham. *IT Project Portfolio Management.* Artech House, 2004.

[6] Project Management Institute. *The Standard for Portfolio Management.* Project Management Institute, 2006.

[7] Chris Chapman and Stephen Ward. *Project Risk Management: Processes, Techniques and Insights.* Wiley, 1997.

[8] Barry Boehm. Software risk management: Principles and practices. *IEEE Software*, 8(1), 1991.

[9] Natalia Levina and Jeanne Ross. From the vendor's perspective: Exploring the value proposition of it outsourcing. *MIS Quarterly*, 27(3), 2003.

[10] Capers Jones. *Assessment and Control of Software Risks.* Prentice Hall, 1994.

[11] Benoit Aubert, Michel Patry, and Suzanne Rivard. Assessing the risk of it outsourcing. *Proceedings of the Thirty-First Annual Hawaii International Conference on System Sciences-Volume 6*, 1998.

[12] R. Whitley. Project-Based Firms: New Organizational Form or Variation of a Theme. *Industrial and Corporate Changes*, 15(1):77–99, 2006.

[13] Jack R. Meredith and Jr. Samuel J. Mantel. *Project Management, A Managerial Approach*. Wiley, 2003.

[14] Kjetil Moløkken and Magne Jørgensen. A review of surveys on software effort estimation. *Proceedings of the 2003 International Symposium on Empirical Software Engineering (ISESE'03)*, 2003.

[15] Miia Martinsuo and Päivi Lehtonen. Role of Single-Project Management in Achieving Portfolio Management Efficiency. *International Journal of Project Management*, (24):56–65, 2007.

[16] M Jørgensen and Dag I. K Sjøberg. Impact of effort estimates on software project work. *Information and Software Technology*, 43(15):939–948, 2001.

[17] Alan Stretton. A Short History of Modern Project Management. *International Journal of Project Management*, (24):175–184, 2006.

[18] Terence J. Cooke-Davis and Andrew Arzymanow. The Maturity of Project Management in Different Industries: An Investigation into Variations between Project Management Models. *International Journal of Project Management*, 21:471–478, 2002.

[19] http://www.bechtel.com/history.htm, last accessed 11 April 2008.

[20] Martin Betts and Peter Lansley. International Journal of Project Management: A Review of the First Ten Years. *International Journal of Project Management*, 13(4):207–217, 1995.

[21] Lynn Crawfort, Julien Pollack, and David England. Uncovering the Trends in Project Management: Journal Emphases over the last 10 Years. *PM World Today*, 9(10), 2007.

[22] J Turner and R Müller. The Nature of the Project as a Temporary Organization . *International Journal of Project Management*, (21):1–8, 2003.

[23] Bjørn Johs. Kolltveit, Jan Tere Karlsen, and Kjell Grønhaug. Perspectives on Project Management. *International Journal of Project Management*, (25):3–9, 2007.

[24] Itzhak Wirth and Douglas E Tryloff. Preliminary comparison of six efforts to document the project-management body of knowledge. *International Journal of Project Management*, 13(2):109 – 118, 1995.

[25] Itzhak Wirth. How generic and how industry – specific is the project management profession? *International Journal of Project Management*, 14(1):7 – 11, 1996.

[26] Vladan Devedzic. Software project management. In S. K. Chang, editor, *Software Engineering and Knowledge Enginerring volume 2: Emerging Technologies*. World Scientific Publishing, 2002.

[27] Ian Gorton. *Essential Software Architecture*. Springer, 2006.

[28] Phil Marriott Barbara Frank and Chett Warzusen. The software quality engineer primer, third edition, 2002.

[29] Roger Atkinson. Project management: cost, time and quality, two best guesses and a phenomenon, its time to accept other success criteria. *International Journal of Project Management*, 17(6):337 – 342, 1999.

[30] A. L. Lederer and J. Prasad. Information systems software cost estimating: a current assessment. *Journal of Information Technology*, 8:22–33, 1993.

[31] Standish Group. *The Chaos Report*. The Standish Group, 1991.

[32] Geoffrey Hewson and Kathy Peters. Fundamentals of software project estimation, 2007.

[33] Kjetil Moløkken-Ostvold and Magne Jørgensen. Reasons for software effort estimation error: impact of respondent role, information collection approach, and data analysis method. *IEEE Transactions on Software Engineering*, 30(12), 2004.

[34] Magne Jørgensen and Dag I. K. Sjøberg. The impact of customer expectations on software development effort estimates. *International Journal of Project Management*, 22, 2004.

[35] Jeffrey Kaplan. *Strategic IT Portfolio Management*. Pittiglio Rabin Todd and McGrath, 2005.

[36] Watts S. Humphrey. Case planning and the software process, 1998.

[37] CMMI Product Team. Capability maturity model integration cmmi version 1.1, 2001.

[38] Janice Thomas Kam Jugdev. Project management maturity models: The silver bullet of competitive advantage? *Project Management Journal*, 33(4), 2002.

[39] A K Munns and B F Bjeirmi. The Role of Project Management in Achieving Project Success . *International Journal of Project Management*, (14):81–87, 1996.

[40] Anton de Wit. Measurement of project success. *International Journal of Project Management*, 6(3), 1988.

[41] John Wateridge. How can is/it projects be measured for success? *International Journal of Project Management*, 16(1), 1997.

[42] Nitin Agarwal and Urvashi Rathod. Defining 'Sucess'for Software Projects: An Exploratory Revelation. *International Journal of Project Management*, (24):358–370, 2006.

[43] Francis Hartman and Rafi A. Ashrafi. Project management in the information systems and information technology industries. *Project Management Journal*, 33(3), 2002.

[44] http://en.wikipedia.org/wiki/harry_markowitz, last accessed 11 April 2008.

[45] Eugene F. Brigham and Michael C. Ehrhardt. *Financial Management*. Thomson Southwestern, 2005.

[46] Bryan Maizlish and Robert Handler. *IT Project Portfolio Management Step by Step-Unlocking the Business Value of Technology*. Wiley, 2005.

[47] J. Ropponen and K. Lyytinen. Components of software development risk. *IEEE Transactions of Software Development Risk*, 26(2):98–111, 2000.

[48] S. Pfleegerer, L. Haton, and L. Howell. *Solid Software*. Prentice Hall, 2001.

[49] Henri Barki, Suzanne Rivard, and Jean Talbot. Towards an assessmant of software development risk. *Journal of Management Information Systems*, 10(2), 1993.

[50] K. Lyytinen, L. Mathiassen, and J Ropponen. A framework for software risk management. *Journal of Information Technology*, 11(4), 1996.

[51] ISO 9000-3. Guidelines for application of iso 9001 to the development, supply and maintenance of software, 1991.

[52] ISO. Iso 12207 information technology. amendment to iso 12207, 2002.

[53] ISO. Iso 15504 part 5: An assessment model and indicator guidance, 1999.

[54] Chris Chapman. Project risk analysis and management-pram the generic process. *International Journal of Project Management*, 15(5), 1997.

[55] Henri Barki, Suzanne Rivard, and Jean Talbot. An integrative contingency model of software risk management. *Journal of Management Information Systems*, 17(4), 2001.

[56] D. E. Allen J Hill, L.C. Tomas. Experts' estimates of task durations in software development projects. *International Journal of Project Management*, 18(1), 2000.

[57] M Jørgensen. A review of studies on expert estimation of software development effort. *Journal of Systems and Software*, 70(1-2):37–60, 2004.

[58] Fred Brooks. *The Mythical Man Man Month. Essays on Software Engineering, Anniversiry Edition. (2ed)*. Addison Wesley, 1995.

[59] R.J. Peters G.P. Kulk and V. Verhoef. Quantifying it estimation risks. *Science of Computer Programming*, 74(1-2):900–933, 2009.

[60] V. Verhoef. Quantitative it portfolio management. *Science of Computer Programming*, 45(1):1–96, 2002.

[61] Nicholas J. S. Christenfeld Michael M. Roy and Craig R. M. McKenzie. Underestimating the duration of future events: Memory incorrectly used or memory bias? *Psychological Bulletin*, 131(5):738–756, 2005.

[62] C. Verhoef R.J. Peters. Quantifying the yield of risk-bearing it-portfolios. *Science of Computer Programming*, 71(1):17–56, 2008.

[63] http://www.mathwave.com/, last accessed, 12 March, 2010.