ADDRESSING APPLICATION SOFTWARE PACKAGE PROJECT FAILURE: BRIDGING THE INFORMATION TECHNOLOGY GAP BY ALIGNING BUSINESS PROCESSES AND PACKAGE FUNCTIONALITY

by

Wandi Kruger

Assignment presented in partial fulfilment of the requirements for the degree

Master of Commerce (Computer Auditing)

at the

University of Stellenbosch

Supervisor: Mrs Sybil Smit

Faculty of Economic and Management Sciences

Department of Accounting

December 2011

DECLARATION

By handing in this assignment electronically, I declare that all of the work contained herein is my own, original work. I hold all author's right of this document and have not previously, in its entirety or in part, submitted it to any university for a degree.

WANDI KRUGER

December 2011

ACKNOWLEDGEMENTS

I would like to express my sincere thanks to our Heavenly Father, who guided me through all my years of studies and who gave me the strength and courage to embrace all the challenges He placed upon my road.

ABSTRACT

An application software package implementation is a complex endeavour, and as such it requires the proper understanding, evaluation and redefining of the current business processes to ensure that the project delivers on the objectives set at the start of the project.

Numerous factors exist that may contribute to the unsuccessful implementation of application software package projects. However, the most significant contributor to the failure of an application software package project lies in the misalignment of the organisation's business processes with the functionality of the application software package. Misalignment is attributed to a gap that exists between the business processes of an organisation and what functionality the application software package has to offer to translate the business processes of an organisation into digital form when implementing and configuring an application software package. This gap is commonly referred to as the information technology (IT) gap.

The purpose of this assignment is to examine and discuss to what degree a supporting framework such as the **Pr**ojects **I**N **C**ontrolled **E**nvironment (PRINCE2) methodology assists in the alignment of the organisation's business processes with the functionality of the end product; as so many projects still fail even though the supporting framework is available to assist organisations with the implementation of the application software package.

This assignment proposes to define and discuss the IT gap. Furthermore this assignment will identify shortcomings and weaknesses in the PRINCE2 methodology which may contribute to misalignment between the business processes of the organisation and the functionality of the application software package.

Shortcomings and weaknesses in the PRINCE2 methodology were identified by:
- Preparing a matrix table summarising the reasons for application software package failures by conducting a literature study

- Mapping the reasons from the literature study to those listed as reasons for project failure by the Office of Government Commerce (the publishers of the PRINCE2 methodology)

- Mapping all above reasons to the PRINCE2 methodology to determine whether the reasons identified are adequately addressed in the PRINCE2 methodology.

This assignment concludes by proposing recommendations for aligning the business processes with the functionality of the application software package (addressing the IT gap) as well as recommendations for addressing weaknesses identified in the PRINCE2 methodology. By adopting these recommendations in conjunction with the PRINCE2 methodology the proper alignment between business processes and the functionality of the application software package may be achieved. The end result will be more successful application software package project implementations.

UITTREKSEL

'n Toepassingsprogrammatuurpakket implementering is 'n komplekse strewe en vereis daarom genoegsame kennis, evaluasie en herdefiniëring van die huidige besigheidsprosesse om te verseker dat die projek resultate lewer volgens die doelwitte wat aan die begin van die projek neergelê is.

Daar bestaan talryke faktore wat kan bydrae tot die onsuksesvolle implementering van toepassingsprogrammatuurpakket projekte. Die grootste bydrae tot die mislukking van 'n toepassingsprogrammatuurpakket lê egter by die wanbelyning van die organisasie se besigheidsprosesse met die funksionaliteit van die toepassingsprogrammatuurpakket. Wanbelyning spruit uit 'n gaping tussen die besigheidsprosesse van `n organisasie en die funksionaliteit wat die toepassingsprogrammatuur kan aanbied om die besigheidsprosesse van 'n organisasie om te skakel in digitale formaat wanneer `n toepassingsprogrammatuurpakket geimplementeer en gekonfigureer word. Daar word gewoonlik na hierdie gaping verwys as die informasie tegnologie (IT) gaping.

Die doel van hierdie opdrag is om te evalueer en bespreek in watter mate 'n ondersteunende raamwerk soos die **PR**ojects **IN** **C**ontrolled **E**nvironment (PRINCE2) metodologie kan help om die organisasie se besigheidsprosesse in lyn te bring met die funksionaliteit van die eindproduk; aangesien so baie projekte steeds misluk ten spyte van die ondersteunende raamwerke wat beskikbaar is om organisasies by te staan met die implementering.

Die opdrag beoog om die IT gaping te definieer en te bespreek. Verder sal hierdie opdrag die swakhede in die PRINCE2 metodologie, wat moontlik die volbringing van behoorlike belyning tussen die besigheidsprosesse en die funksionaliteit van die toepassingsprogrammatuurpakket belemmer, identifiseer.

Swakhede en tekortkominge in die PRINCE2 metodologie is as volg geïdentifiseer:

- Voorbereiding van 'n matriks-tabel wat die redes vir toepassingsprogrammatuurpakket mislukking deur middel van die uitvoering van 'n literatuurstudie opsom
- Koppeling van die redes bekom deur middel van die literatuurstudie met die redes vir projek mislukking geidentifiseer deur die *Office of Government Commerce* (uitgewers van die PRINCE2 metodologie)
- Koppeling van al die bogenoemde redes na die PRINCE2 metodologie om vas te stel of die redes wat geïdentifiseer is voldoende deur die PRINCE2 metodologie aangespreek word.

Die opdrag sluit af met aanbevelings om die besigheidsprosesse in lyn te bring met die funksionaliteit van die toepassingsprogrammatuurpakket en aanbevelings vir swakhede wat in die PRINCE2 metodologie geïdentifiseer is aan te spreek. Behoorlike belyning tussen besigheidsprosesse en die funksionaliteit van toepassingsprogrammatuurpakket kan behaal word indien hierdie aanbevelings aangeneem word en tesame met die PRINCE2 metodologie gebruik word. Die eindresultaat is meer suksesvolle implementering van toepassingsprogrammatuurpakket projekte.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1   Background

It is expected that information technology (IT) projects will become more turbulent and difficult in future (Sauer & Cuthbertson, 2003:70). This situation will result in one of the most common challenges top management face: the decision to make significant investments in application software package projects. Although top management may perceive that IT projects may result in the enhancement of the organisation performance, it is important to remember that implementing an application software package goes further than only changing components; it usually requires a complete refit of the organisation itself (Ahmad & Newman, 2009:3). The refit of the organisation entails the strategic alignment of business processes (Tillmann & Weinberger, 2004:28).

By applying application software packages in business processes, organisations believe they will ultimately improve on earnings through improved operational efficiency, decrease in costs, enhanced ability to make knowledgeable decisions and create competitive advantages by enabling innovative practices (Winter, 2006:vi, Al Neimat, 2005:1 and Al-Mashari, Al-Mudimigh & Zairi, 2003:352).

For organisations wanting to succeed in implementing application software packages within budget, within timeframe and with the specification functionality, they would need to evaluate their current business processes and where necessary, re-engineer or streamline their internal processes to suit the operational requirements (Winter, 2006:1 & Weston, 2001:1). Re-enginering internal processes is very often an ambiguous process (Bartis & Mitev, 2008:113).

However, various studies have found that a large number of significant IT investment projects result in waste and fail to provide a return to the entity as the projects fail to achieve the original functional objectives set at the start of the project (ITGI, 2008:7). In

a study conducted by PriceWaterhouseCoopers Inc. in 2004, 10 640 projects were surveyed and revealed that only 2.5 percent of organisations achieve budget, scope and schedule targets in all projects (Dalcher, 2009:43). This is in contrast with the 2004 study conducted by The Standish Group which reported a higher success rate for IT projects at 29 percent (Eveleens & Verhoef, 2010:31).

The Standish Chaos Report for 2006 showed that 35 percent of IT projects were successful, which decreased by 3 percent to 32 percent success rate according to their 2009 study (Eveleens & Verhoef, 2010:31).

Computerworld (s.a.) published a list of the top 10 corporate IT failures as:

- **AMR Corporation, Budget Rent A Car Corporation, Hilton Hotels Corporation and Marriot International Inc.**: The deadline of this IT project was missed by as much as two years and AMR Corporation took a $109 million write-off.
- **Snap-on-Inc.**: Due to improper functionality of the system this IT project cost the organisation $50 million in lost sales in the first year after implementation and a 22 percent decrease in profits.
- **FoxMeyer Corporation**: This drug company was forced to declare bankruptcy after an unsuccessful IT project implementation.
- **W.W. Grainger Inc.**: The new ERP system implemented overstated inventory and had routine crashes. Grainger made a loss of $19 million in sales and $23 million in profits.
- **Greyhound Lines Inc.**: The "Trips" system that was implemented crashed when Greyhound offered sale prices on bus fares. The company incurred a $61.4 million loss for the first six months of 1994.
- **Hershey Foods Corporation**: The rollout of the new ERP system was compressed for a number of months which lead to inaccurate inventory data. Sales went down 12 percent in the first quarter after the system went live.
- **Norfolk Southern Corporation**: Due to improper testing of custom logistics software the company lost $113 million during its railroad merger with Conrail.

- **Oxford Health Plans Inc**.: The new system implemented understated medical costs and overstated income. The company was fined $3 million by the New York state for violating insurance laws.
- **Tri Valley Growers**: This Company was forced to declare bankruptcy after an unsuccessful ERP software implementation.
- **Universal Oil Products LLC**: The end product of this software project for estimating project expenditures and figuring engineering requirements resulted in an unusable system.

However, reports of failed IT projects in the private sector are hard to come by, either because positions and reputations are at risk or organisations want to put failures behind them and move forward (Holt, 2003:1). There are numerous reasons that may contribute to application software package project failures. The reasons are listed in Chapter 2.

A number of supporting frameworks are available that may assist in the implementation of application software package projects. Supporting frameworks can be divided into two broad categories, namely generic methodologies (for example **Pr**ojects **i**n **C**ontrolled **E**nvironments – PRINCE2) and product specific methodologies (for example Microsoft Dynamics Sure Step). Many of the generic frameworks may be applied to any type of project.

Various authors (McManus & Wood-Harper, 2007:41, Taylor, 2000:25, Tillmann & Weinberger, 2004:28, Umble, Haft & Umble, 2003:251, Ehie & Madsen, 2005:546, Zand & Sorensen, 1975:541) of the IT project failure topic are of the opinion that proper business process alignment is the biggest contributor to project success. Several organisations are of the view that by placing total reliance on supporting methodologies, proper alignment of business processes with the application software package may be achieved (McManus & Wood-Harper, 2007:43).

Although there are various supporting frameworks available that may assist in the implementation of application software package projects, the question arises why

industry reports still show that the success rate (Winter, 2006:vi) of application software package project implementation is low.

Above question on the low success rate of application software package project implementation may be answered by the view of McManus and Wood-Harper (2007:43). In their analysis one of the major weaknesses they uncovered for IT project failure was the total reliance placed on methodologies such as PRINCE2. However, they argue that following methodologies may help the stakeholders involved in the project in organising and delivering application software package projects.

The view of McManus and Wood-Harper (2007:43) may be further supported by the opinion of Taylor (2000:26) that no two IT projects are the same and for that reason not one of the project management methods (supporting frameworks), such as PRINCE2, is perfect. In his opinion each supporting framework has facets which are more suitable to one IT project than another.

One major question that arises is to what extent the supporting frameworks available really assist top management in aligning business processes with the application software package.

The aim of this assignment is to determine to what degree supporting frameworks assist management with aligning business processes with the application software package. The answer will be structured by identifying shortcomings and weaknesses in the supporting framework selected for this study (PRINCE2) contributing to misalignment. Furthermore, recommendations will be made on how to align business processes with the application software package as well as recommendations for weaknesses identified in the PRINCE2 methodology.

1.2   Statement of problem

The most significant reason why IT projects in general fail is that organisational strategies are not aligned with the application software package project strategy (Velcu, 2010:160). The organisational strategies, for the purpose of this assignment, refer to the business processes of the organisation.

Misalignment is attributed to a gap that exists between business processes of an organisation and what functionality the application software package has to offer to translate the business processes of an organisation into digital form when implementing and configuring an application software package. This gap is commonly referred to as the IT gap. (Boshoff, 2011).

1.3   Purpose of study

Organisations embark on the implementation of application software package projects with the expectation that such projects will enhance improvements in one or more of the following areas (Boshoff, 2010):
- Adding value to the organisation to help the organisation stay innovative
- Lower skill requirements in order to reduce costs
- Efficient workflow in order to reduce human error
- "Dumbing" down (over-simplification of application software package)
- Top management having access to real-time information.

However from a broad review of literature a large number of IT projects are regarded as failures and do not always enhance improvements in the areas identified above.

Literature covers supporting frameworks in general and implementation of application software package projects. Previous studies have however not addressed the complex challenges faced when using PRINCE2 to assist in strategic alignment of business processes of the organisation with the functionality of the application software package.

To support above, one of the first empirical studies into the impact of PRINCE2 on the performance of a project: Creating Value in Project Management Using PRINCE2, was conducted by Queensland University of Technology. They concluded that their study conducted in 2010 should be extended to assess the impact of the strategic alignment of PRINCE2 in an organisation (Creating Value in Project Management Using PRINCE2, 2010).

The primary objective of this assignment is to examine why application software package project implementations fail even if supporting frameworks are available to assist with implementation. This assignment also proposes to examine to what extent the generic methodology supporting framework addresses the IT gap (assist in strategic alignment of business processes with the functionality of the application software package). If the generic methodology supporting framework does not properly address the IT gap, this assignment will attempt to identify the shortcomings and weaknesses.

This assignment further proposes to recommend possible additional steps, which may be followed to ensure strategic alignment of application software package projects with business processes as well as recommendations for weaknesses identified in the PRINCE2 methodology, should the generic supporting methodology framework not address the IT gap.

This assignment may assist top management of organisations and IT professionals (suppliers of application software packages) to successfully align business processes with application software packages when using PRINCE2.

1.4   Design and methodology

The approach of this assignment is non-empirical, through a review of literature in the form of white papers, academic articles, thesis and other research related to strategic alignment of application software packages and IT project failure in general.

In Chapter 2 the author gives an overview of previous research conducted on project failures and the role of application software suppliers.

The IT gap (strategic alignment) is defined and discussed in Chapter 3.

In Chapter 4 the author gives an overview of the supporting framework selected for this assignment (PRINCE2).

In Chapter 5 a matrix is provided which summarises the most frequently mentioned reasons for project failure in the literature reviewed for this assignment. The reasons identified in the literature review are mapped to the most important reasons for project failure listed by the Office of Government Commerce (publisher of PRINCE2). Both sets of reasons are then mapped to the supporting framework selected for this assignment, PRINCE2, to indicate whether, in the opinion of the authors of PRINCE2, the reasons are adequately addressed or not in the PRINCE2 methodology.

In Chapter 6 shortcomings and weaknesses identified in the supporting framework selected contributing to improper alignment are discussed.

Finally, in Chapter 7 the author proposes recommendations on how the IT gap can be bridged and weaknesses identified in PRINCE2 could be mitigated or reduced to ensure proper alignment of business processes with the functionality of the end product.

A summary of this assignment and the conclusions drawn are provided in Chapter 8.

1.5 Limitations of study

The limitations of this assignment include the following:
- This assignment will only identify weaknesses and make recommendations specifically for strategic alignment of application software packages acquired from

an application software supplier and not system software packages. Application software developments will also not be addressed.

- This assignment will not discuss technical aspects of application software package project implementations.
- The author only references the PRINCE2 2009 project management methodology.

CHAPTER 2

LITERATURE STUDY

2.1   Introduction

Application software package project implementation failures are not a new phenomenon. Much has been written about the challenges of managing and directing IT projects.

During the literature study conducted for this assignment it was noted that many researchers based their research of the reasons for IT project failure on IT project failure in general and did not distinguish between application software package project failures and system software project failure. The reasons listed in existing literature apply mostly to all IT project implementations. For that reason the literature study below is mainly based on IT project failure in general. Where a researcher based his/her study specifically on application software package project failures reference will be made as such.

2.2   Definitions

2.2.1 Defining application software packages

Wikipedia (2011a) defines application software packages as computer software designed to help the operator to perform singular or multiple related specific tasks. Examples of application software packages include accounting software, office suites, enterprise software, graphic software and media players.

Application software packages are contrasted with system software packages in that system software packages are computer software designed to operate and control the computer hardware and to provide a platform for running application software packages

(Wikipedia, 2011f). Examples of system software packages include firmware, operating systems and utility software (Wikipedia, 2011f).

This assignment will only address application software package projects.

2.2.2 Defining projects

Practically all application software package implementations are undertaken as IT projects (Jurison, 1999:3).

Jurison (1999:5) defines an IT project as a temporary assembly of resources to solve a one-of-a-kind problem. Projects may range from a small project like developing a spreadsheet-based sales plan to large enterprise-wide projects employing hundreds of resources working together. All projects display the following common characteristics (Jurison, 1999:5):

- Projects have specific goals
- Projects must be completed within a specific timeframe and budget
- Projects are carried out by a project team
- Projects are nonrecurring undertakings for a specific organisation.

The Office of Government Commerce (Common Causes of Project Failure, s.a.) agrees with the definition of Jurison by defining a project as "a unique set of co-ordinated activities with a finite duration, defined cost and performance parameters and clear outputs to support specific business objectives".

2.2.3 Defining IT project success

Various authors in the literature define IT project success differently.

IT Cortex (2005) defines IT project success as:

- A well planned, organised, clear and efficient business solution that can mature along with the streamlined business
- The epitome of business sense
- The perfect synergy between the business environment and project
- The aligning of the goals and means of the project.

In addition, in the opinion of Poli and Shenhar (2003:231) IT project success should not only be measured on cost, specification (or functionality) and time but should include criteria such as extending product lines, building market share, increasing revenue, building for the future and satisfying clients.

A research study conducted by Sofian (2003:6) surveyed 142 respondents which included project team members and project, top and functional managers from different industries in the United Kingdom. One of the questions asked to respondents was to select one or more of the definitions provided in the survey for what is meant by IT project success. The results are shown in Table 2.1.

Table 2.1: Definition of IT project success

Definition of IT project success	Percentage of respondents selecting this definition
It meets target cost, schedule, quality and functionality	88.5%
It meets client satisfaction	85.9%
It creates organisational improvement with learning from failures and success	44.9%
It was performed efficiently and effectively	43.6%
It succeeds in executing the desired changes because one cannot expect every IT project to proceed exactly as planned	37.2%

(Source: Sofian, 2003)

Sofian (2003:6) concluded that the majority of respondents regard cost, time schedules, quality and functionality as primary to the definition of IT project success.

The views on IT project success of other authors who recently conducted research on IT project success factors are summarised in Table 2.2.

Table 2.2: Summary of IT project success factors

IT project success factor	Source
Competency of all stakeholders involved	Upadhyay, Jahanyan & Dan (2011:142)
Strategic alignment of business strategies with application software package functionalities	Velcu (2010:164); IT Cortex (2005); Poli & Shenhar (2003:231)
Rigorous IT project management	Chen, Law & Yang (2009:157)
Sufficient planning	IT Cortex (2005)
Project completed within budget, timeline and within the original specification of functionality agreed upon at start of IT project	Poli & Shenhar (2003:231); Sofian (2003:6); Taylor (2000:24)

Therefore the measurement of the successful outcome of an IT application software package project does not exist in isolation, but depends on a combination of factors during the IT project life cycle (refer to section 2.4 for definition) (Procaccino & Verner, 2006:1542).

2.2.4 Defining IT project failure

Coley Consulting (2005:1) defines IT project failure as the project being:
- Not delivered on time
- Over budget
- Not meeting user requirements.

Velcu (2010:160) defines project failure as the misalignment of organisational strategies with the application software package project strategies. Unless organisations use application software packages that support their business strategies, the organisations risk of project failure is significantly increased (Velcu, 2010:160).

2.3   Reasons why IT projects fail

Different authors of the topic of why IT projects are unsuccessful place the blame of IT project failure on various factors. The one factor most of the researchers agree on is that improper IT project management is a significant contributor to IT project failure (Plotnikova, 2007:3).

Some of the authors in literature reviewed for this assignment argue that it is the sole obligation of the project manager to constantly make trade-off decisions on schedule, quality and budget limits of the IT project (Chen *et al.*, 2009:158). One example is the view expressed by Leitao (as cited by Winter, 2006:13). He states that the three main IT project constraints, namely time, cost and functionality are interrelated. He defines project failure as not meeting desired performance, late delivery or overrun on the budget.

However, Cerpa and Verner (2009:130) express the view that a combination of business, technical and project management factors contribute to an IT application software package project failure.

The studies and surveys conducted by numerous authors, listing the reasons contributing to IT project failure during the past decade, are listed below. Reasons are listed per author from most recent study to least recent study.

The Office of Government Commerce (publisher of PRINCE2) (OGC) lists the reasons for IT project failure in their best practice guide: Common Causes of Project Failure (s.a.). The reasons are:

- Lack of clear links between the project and the organisation's key strategic priorities, including agreed measures of success
- Lack of clear top management and ministerial ownership and leadership
- Lack of effective engagement with stakeholders
- Lack of skills and proven approach to project management and risk management
- Too little attention to dividing development and implementation into manageable steps
- Evaluation of proposals driven by initial price rather than long-term value for money (especially securing delivery of business benefits)
- Lack of understanding of, and contact with the supply industry at senior levels in the organisation
- Lack of effective project team integration between clients (top management), the supplier (IT) team and the supply chain.

The reasons listed by **INTOSAI** (s.a.) (professional organisation of supreme audit institutions in countries that belong to the United Nations) are consistent with the reasons listed by the Office of Government Commerce (Common Causes of Project failure, s.a.) and **Dolan** (2010:3). INTOSAI (s.a.) lists the following reasons for IT projects failure:

- Improper scope definition
- Lack of business case and business objectives
- No project sponsor (top management) to support project manager
- Decision by committee
- Absence of or little risk management
- Insufficient project management experience
- Changes in scope not managed well
- Low cost supplier selection (This issue will be discussed in section 2.5.)
- Lack of transparency between client and supplier due to different business goals
- Lack of end user involvement in project definition.

2009

Cerpa and **Verner** (2009:131) distributed a questionnaire containing 88 questions on application software success and application software failures to software practitioners. They received 304 completed questionnaires and compiled a list of application software project failure factors. The results of the survey are listed below:

- A tight deadline impacted the development of the software
- The project was under-estimated in terms of budget, time and complexity
- Risks were not re-assessed and controlled throughout the project life cycle
- Staff were not remunerated for working long hours ("people" factor)
- Decisions were made without sufficient information on requirements
- Staff had an unpleasant experience working on the project ("people" factor)
- End users were not involved in making plan estimates
- Risks were not included into the project plan
- Change control was not monitored, nor dealt with effectively
- End users had unrealistic expectations
- Processes did not have evaluations at the end of each phase
- The project methodology was inappropriate for the project
- A tight schedule had a negative effect on team member's life ("people" factor)
- The project had inadequate staff to meet the schedule
- Additional staff members were added late to the project team to meet an aggressive project schedule ("people" factor)
- End users did not make adequate time available for requirements assembly.

Cerpa and Verner (2009:132) concluded by emphasising that "people" factors are important factors contributing towards project success as is evident from the four "people" factors listed above.

Demir (2009) conducted a survey among 78 software practitioners regarding their last IT software project. The survey focussed on the challenges that the practitioners experienced in the management of IT application software projects. The results

(challenging software project area and percentage of respondents indicating the area as challenging) of the survey are indicated in Table 2.3.

Table 2.3: Challenging IT software project areas

IT Software project area	Response percentage
Scope management	52.6%
Requirements management	51.3%
Project planning and estimation	41.0%
Communication	38.5%
Staffing and hiring	33.3%
Project monitoring and control	28.2%
Risk control	26.9%
Technical complexity	26.9%
Stakeholder involvement	25.6%
Leadership	25.6%
Configuration management	25.6%
Organisational commitment	24.4%
Quality engineering	23.1%
Teamwork	21.8%
Risk assessment	19.2%
Project manager	14.1%
Other	10.3%
Support activities	9.0%

(Source: Demir, 2009)

Chen *et al.* (2009:157) list the following reasons for IT project failures:

- Changes in scope during project life cycle
- Inadequate risk management
- Insufficient allocation of human resources over time
- Improper supplier management.

Chen *et al.* (2009:157) conclude that improper project management can jeopardise the successful implementation of IT projects.

Certain authors, namely Ehie and Madsen (2005:555), Gargeya and Brady (2005:511), Chin (2003:1), Umble *et al.*(2003:245) and Jurison (1999:4), who conducted research specifically focusing on Enterprise Resource Planning software implementations agree with Chen *et al.* (2009:158) in that proper project management is critical in order to achieve project success. Chen *et al.* (2009:158) further stress that although proper project management is a critical factor for project success, it is not the only factor to consider.

2008

During 2008 **Aken** (2008:317) conducted a research study and expressed the opinion that one of the reasons for project failure is the delay between the specification of the functional requirements and the final implementation of the application software package.

The research of **Deng** and **Bian** (2008:72) was based on the prerequisite for IT project success which is setting up a set of risk management mechanisms.

2007

Aloini, **Dulmin** and **Mininno** (2007:558) conducted a literature review on application software failures and identified the top 10 risk factors from literature per project life cycle phase. The top 10 risk factors are listed in Table 2.4.

Table 2.4: Top 10 risk factors

Risk factor	Project life cycle phase
Improper application software selection	Initiation/Planning
Lack of strategic thinking and planning	Initiation/Planning
Ineffective project management techniques	Implementation
Bad managerial conduct	Initiation/Planning
Inadequate change management	Implementation
Insufficient training and instruction	Implementation
Improper project team skills	Initiation/Planning
Inadequate business process re-engineering	Initiation/Planning
Poor top management involvement	Initiation/Planning
Poor end user involvement	Initiation/Planning

(Source: Aloini *et al.,* 2007)

Aloini *et al.* (2007:559) concluded that 40 percent of the papers examined in their research study indicated that project failure is due to improper strategic thinking and planning.

In the research study conducted by **McManus** and **Wood-Harper** (2007:42) they concluded that management factors account for 53 percent of the project failure rate, technical causal factors for 27.4 percent and business factors for 19.6 percent. These factors are listed in Table 2.5.

The view that project management issues are the biggest contributor to project failure and that business issues carry the least weight, is supported by the results of the study conducted by **Thomas** and **Fenandez** (2008:736).

Table 2.5: Reasons for IT project failure

Management causal factors	Technical causal factors	Business reasons
Inability to adapt to new resources combinations	Inappropriate architecture	Business strategy superseded
Difference between management (client) and IT (supplier)	Insufficient reuse of existing technical objects	Business process change (poor alignment)
Insufficient risk management	Inappropriate testing tools	Poor requirements management
Insufficient end user management	Inappropriate coding language	Business benefits not clearly communicated or overstated
Insufficient domain knowledge	Inappropriate technical methodologies	Failure of parent company to deliver
Insufficient software metrics	Lack of formal technical standards	Governance issues within the contract
Insufficient training of users	Lack of technical innovation	Higher cost of capital
Inappropriate procedures and routines	Misstatement of technical risk	Inability to provide investment capital
Lack of management judgement	Obsolescence of technology	Inappropriate disaster recovery
Lack of software development metrics	Poor interface specifications	Misuse of financial resources
Loss of key personnel	Poor quality code	Overspend in excess of agreed budgets
Poor managing legacy replacement	Poor system testing	Poor project board composition
Poor supplier management	Poor data migration	Take-over of client firm
Poor software productivity	Poor system integration	Huge project portfolio
Poor communication between stakeholders	Poor configuration management	
Poor contract management	Poor change management procedures	
Poor financial management	Poor technical judgement	
Insufficient project management capability		
Poor delegation and decision making		
Unfilled promises to users and other stakeholders		

(Source: McManus & Wood-Harper, 2007)

2006

Kappelman, **McKeeman** and **Zhang** (2006:34) identified 53 early warning signs which could be an indication that the IT project is failing. The top 12 early warning signs for IT project failure are listed below:

- Lack of top management support

- Weak project manager

- No end user involvement

- Weak commitment of project team

- Team members lack requisite knowledge and/or skills

- Subject matter experts are overscheduled

- Lack of documented requirements and/or success criteria

- No change control process (change management)

- Ineffective schedule planning and/or management

- Communication breakdown among stakeholders

- Resources assigned to a higher priority project

- No business case for the project.

In the opinion of **Bennatan** (2009:5) many IT projects fail because top management either ignore above early warning signs indicating a severely troubled project or deal with it at a very late stage of the IT project life cycle.

Leitao (as cited by Winter, 2006:13) is of the opinion that the inability of the organisation to properly define the business needs for IT results in user requirements of the application software package not being anticipated. He further stresses that the inability of an organisation to define the business needs is because the organisation does not necessarily understand why they need IT. The organisation just believes that it has the potential to save money. (Leitao, as cited by Winter, 2006:13).

However, to potentially save money it is necessary for both top management of the organisation and IT to have a good understanding of the business case (Winter, 2006:13).

Wikipedia (2011b) defines a business case as a structured written document that captures the reasoning for initiating a project or a task. A business case should be prepared or built by top management (Wikipedia, 2011b). This document could include

the background to the project, the expected benefits, the estimated costs and expected risks (Wikipedia, 2011b).

2005

In his White Paper, **Al Neimat** (2005:3) identifies the following reasons for IT project failure:

- Improper project planning
- Unclear objectives
- Change in user requirements during the project lifetime
- Unrealistic resource and timescale estimate
- Lack of top management support and user involvement
- Failure to communicate.

Coley Consulting (2005:1) lists the most important reasons for unsuccessful IT projects as:

- Lack of end user involvement
- Unrealistic timescales
- Vague requirements with little end user input
- Change in end user requirements during the project lifetime
- No change control system
- Poor testing.

They conclude by stating that a number of factors which interact with each other contribute to IT project failure.

Kim, **Lee** and **Gosain** (2005:164) identified the following reasons for IT project failure:

- Conflict of interest among different functional users
- Inadequate human resource commitment from different functional units
- Lack of organisational change management expertise

- Business process not redesigned to take advantage of application software package
- Resistance of users to new systems
- Application software lacks some functionality to support current business processes.

Turbit (2005:5) lists in his White Paper the most likely problems that an IT project team can experience when implementing IT projects as:

- Underestimated cost budget and time schedule
- Greater than expected resources from IT and business required
- Level of outsourced expertise required higher than projected
- Changes in business processes required
- More training needed than expected
- Change in end user requirements underestimated.

Turbit (2005:3) concludes by stressing that many IT projects focus on technical aspects and neglect important people issues.

2003

The reasons of project failure identified by **Chin** (2003:1) are listed below:

- Over ambitious project scope
- Lack of project methodology
- Little end user input and requirements gathering
- Little support from top management
- Poor interpersonal skills.

Holt (2003:2) lists the following reasons for IT project failure:

- Spiralling costs which, for example, include cost due to lack of planning and proper project management
- Pressure to complete within budget and time

- Changes during project life
- Internal politics.

Research conducted by **Sauer** and **Cuthbertson** (2003:60-61) asked respondents to rate the reasons for project failure in order of importance. The reasons for IT project failures included in the survey were based on a list compiled by other researchers. The results are shown in Table 2.6.

Table 2.6: Ranking of IT project risks as causes of IT project failure

Risks	Rank
Lack of top management commitment	1
Misunderstanding of scope/objectives/requirements	2
Lack of client/end user commitment/involvement	3
Changing scope/objectives	4
Poor planning/estimation	5
Inadequate project management	6
Failure to manage end user expectations	7
Conflict among stakeholders	8
Change in top management ownership	9
Lack of adequate change control	10
Shortage of knowledge/skills in project team	11
Improper definition of roles and responsibilities	12
Artificial deadlines	13
Specifications not properly set at beginning of project	14
New or radically redesigned business process/task	15
Employment of new technology	16
Poor control against targets	17
Number of organisational units involved	18
Lack of effective methodologies	19
Staff turnover	20
Multiple suppliers	21

(Source: Sauer & Cuthbertson, 2003)

2002

Smith (2002:57) limited his research to why IT software projects fail to South Africa in particular. His research was based on a similar model to that of The Standish Group. In comparison with the results published by The Standish Group in 2004, it appears that South Africa enjoyed a lot more successful IT projects (46 percent) (Smith, 2002:94)

opposed to the 29 percent success rate internationally as published by The Standish Group (Eveleens & Verhoef, 2010:31).

Smith (2002:46) is of the opinion that, should the project objectives not be defined properly at the start of the project, the main reason for IT project failure is that the project team has no direction of what to deliver as end product.

Other IT project failure reasons listed by Smith (2002:57) are:

- Lack of end user input
- Incomplete requirements and specification
- Changing requirements and specifications
- Lack of top management support
- New technology and technology incompetence
- Lack of resources
- Unrealistic expectations
- Unclear business objectives
- Unrealistic time schedules
- Lack of IT management
- Lack of planning
- No business case
- Poor risk management
- Poor communication.

Smith (2002) concluded that the reasons for project failure in South Africa are similar to the reasons for IT project failure internationally.

2001

Keil and **Robey** (2001:87) are of the view that the decision makers (top management) in the organisation with the power to change the course of the IT project are very often uninformed of the true status of the project. Thus, while indications of a failing project

may exist in the lower positions of an organisation, accurate information about project failure may fail to move up the organisational hierarchy to top management. The reluctance to report the true status of a distressed IT project is a big contributor to project failure (Park & Keil, 2009:45, Keil, Im & Mahring, 2007:59).

2000 and older

Taylor (2000:24) covered a total of 1 027 IT projects in his research study. He divided his analysis into three parts by asking the following three questions to 38 members of the British Computer Society, the Institute of Management and the Association of Project Managers:

- What activities contribute to IT project failure? (The results are shown in Table 2.7.)
- At what stage in the project lifecycle does an IT project fail? (The results are shown in Table 2.8.)
- What are the causes of failure once the IT project has started? (The results are shown in table 2.9.)

Table 2.7: Activities contributing to IT project failure

Activity contributing to failure	Frequency mentioned	Perceived importance
Poor scope management	81.6%	24.7%
Poor project management	71.7%	15.5%
Poor monitoring and control	55.3%	10.9%
Poor risk management	47.4%	10.0%
Poor client management	39.5%	9.1%
Poor communication management	34.2%	8.5%
Poor data conversion management	15.8%	2.4%
Poor contract management	13.25%	1.3%
Poor interface management	7.9%	0.8%
Poor cost management	7.9%	0.8%

(Source: Taylor, 2000)

Table 2.8: Stage at which an IT project failure occurs

Stage at which IT project failure occurs	Frequency mentioned	Perceived importance
Requirement definition	76.3%	23.2%
Implementation	52.6%	13.5%
User acceptance	50.0%	12.8%
Project planning	42.1%	8.1%
Project identification	28.9%	6.6%
Development	18.4%	6.6%
Project initiation	31.6%	6.3%
Testing	31.6%	5.7%
Design	26.3%	5.7%
Project resource estimation	23.7%	5.5%
User training	21.1%	3.8%
Project staff training	7.9%	0.6%

(Source: Taylor, 2000)

Table 2.9: Causes of IT project failure

Cause	Frequency mentioned	Perceived importance
Unclear objectives and requirements	73.7%	18.1%
Lack of business commitment	60.5%	16.5%
Business requirements changing	57.9%	12.2%
Poor communication	44.7%	7.7%
Poor quality steering group	47.4%	7.4%
Poor project planning	44.7%	7.2%
Company and project politics	39.5%	6.9%

(Source: Taylor, 2000)

May (1998:2) cites the following factors for IT project failure as early as 1998:

- Lack of end user input
- Vague IT requirements
- Stakeholder conflicts
- Lack of proper communication between teams on IT project
- Late project failure warning signs
- Inaccurate cost and time schedule estimation
- Expertise that does not match the job
- Improper project planning.

It is concluded that many of the reasons listed by the authors in the literature are consistent.

2.4   Phases of implementing application software package projects

Jurison (1999:8) expresses the view that the IT project life cycle for application software packages may be divided into four phases. The four phases are as follows:

1. Initiation

In the initiation phase business requirements are identified, goals are established, the feasibility of the project is determined, a project proposal is prepared, time and resources are roughly estimated, key people for the project is identified and approval from top management for the project is obtained (Jurison, 1999:8).

2. Planning

Project plans, resource requirements, quality and risk concerns, budget and time schedules are prepared, the project team is assembled, feasibility of the IT project is analysed and approval for the next phase is obtained during the planning phase (Jurison, 1999:8).

Weston (2001:77) adds that care must be taken to ensure scalability of the application software product that is selected. Flexibility of the application software product must also be considered to include add-on functionality if the primary supplier does not offer add-ons (Weston, 2001:77).

3. Execution (implementation)

The execution phase entails performing the work as defined in the planning phase. Resources should be properly managed by the project manager during this phase. Further, the phase entails translating business and functional requirements into code. (Jurison, 1999:8).

Any modifications required to the original project plan should be taken into account during the execution phase and the final product (application software package) implemented should be tested thoroughly (Jurison, 1999:8).

Training schedules should be compiled during the execution phase to ensure training already starts at the beginning of the project (Weston, 2001:77).

4. Termination (controlling and closing)

The termination phase may be triggered either by early termination of the project or by successful accomplishment of the project goals (Jurison, 1999:8).

At each one of above phases there are risks that may contribute individually and/or as a whole towards IT project failure (Boshoff, 2011).

Design and implementation decisions made at the beginning of the project can have an impact on activities undertaken at a later stage during the life cycle of the project (Chen *et al.*, 2009).

In their article, Chen *et al.* (2009:158) listed the following reasons for IT project failure during the different stages:

1. Initiation and planning phases:
   - Top management may poorly define IT requirements
   - Top management may have an overly simplistic project plan
   - Top management may use unrealistic deadlines and budgets
   - Top management may fail to set and manage expectations on the application software being developed
   - Top management may fail to gain support from users, developers and functional managers.

2. Execution and controlling phases:
   - Maintaining clear communication between project staff
   - Poor consultant, top management and team participation

- Additional requirements (after the project started) due to external and internal changes
- Poor measurement of project performance
- On-going evaluation may be problematic due to participants that have different vested interests
- Organisational diversity
- Inadequate cross-functional coordination.

3. Closing phase:
   - High turnover rate of skilled professionals
   - Globalisation of IT field.

It is important to note that an application software package project may fail at any one of the above stages (Boshoff, 2011).

2.5 The role of the application software package supplier

This assignment addresses application software packages acquired from a supplier, therefore the influence of the supplier on the strategic alignment of the business process of the organisation with the functionality of the end product is discussed.

Another reason for application software package project failures may be that the end user organisations do not always have the in-house expertise to handle the technical issues relating to implementation of application software packages, due to the complexity of the application software package (Winter, 2006:2). Not having the in-house expertise will result in appointing an IT application software package supplier to assist with the implementation of the application software package. This will result in the project team consisting of both end users and the supplier of the application software package.

If the organisation decides to follow the supplier route, the end user may buy-in the product (application software package) offered by the supplier without properly evaluating

the business requirements (business processes) of the organisation. The end user will usually take the word of the supplier that the product is a perfect fit for the organisation's information needs and business processes, only realising at a later stage that the end product functionality does not meet the needs initially identified.

Above is supported by the view expressed by Umble *et al.* (2003:248) in that most application software suppliers may go as far to make assumptions about top management business processes. In some instances application software suppliers may pursue their creativity without regard to the client's business requirements (Agarwal & Rathod, 2006:359). What the supplier does not communicate properly to the organisation is that the customisation features of the purchased application software package cannot be extended in general terms as it is specific to the particular application software package (Stapelberg, 1994:6).

However, Craig (as cited by Winter, 2006:29) expresses the view that where the supplier and organisation work together as a single project team there is a higher chance that the project will be successful.

The organisation is buying more than just application software from the supplier. The organisation is actually purchasing the software supplier's interpretation for many of the organisation's business processes. The organisations that implement the application software package accept the supplier's assumptions about the organisation, without properly evaluating the business processes, and they change existing procedures and processes to conform to what the supplier is selling. The result is an end product without the functionality required by the organisation. (Umble *et al.,* 2003:248).

Turbit (2005:4) and Ke and Wei (2008:209) support the above view by stating that a common mistake made by organisations is that they try to change business processes to suit the application software package. Organisations should rather evaluate and change the business processes and patterns of workflow to improve efficiency. However, it is

important that the product selected and purchased needs to be generally compatible with the business requirements (Stapelberg, 1994:5).

In many cases organisations may choose to acquire the application software package from the supplier with the lowest bid. Low buy-in also limits the participation mix of business and IT which contributes to improper alignment of business processes of the organisation with the functionality of the end product (Turbit, 2005:4).

From the above it is clear that the supplier may well contribute towards an organisation not properly identifying their business requirements, because the supplier is selling their product and neglecting the actual needs of the organisation. Organisations should start the project by identifying the business requirements of the organisation and only thereafter select the application software product that is most suitable to address the business requirements.

2.6   Conclusion

Many of the reasons listed in literature are attributed to improper communication between IT professionals (responsible for implementing the application software package) and top management (responsible for defining requirements of the application software package) (Boshoff, 2011). Furthermore IT professionals and top management of an organisation have little knowledge of each other's environments. These two factors result in a gap between the two parties, referred to as the IT gap.

CHAPTER 3

DEFINING AND EXPLAINING THE IT GAP

3.1   Introducing and defining the IT gap

The IT gap with regards to application software packages is attributed to a gap that exists between business processes and what functionality the application software package has to offer to translate the business processes of an organisation into digital form when implementing and configuring an application software package. (Boshoff, 2011).

The above is supported by Stapelberg (1994:11). He states that there is a gap between the business requirements (or specific business processes) and the IT programmer's (supplier) interpretation of the requirements.

Authors in the past have identified the IT gap as a major contributor towards IT project failure, although they may have used different terminology (e.g. strategic alignment).

For example a research study conducted by Velcu (2010:164) tested the degree to which business strategies were aligned with application software package functionalities. The results of the study showed that the more the application software package project strategy was aligned with the business strategy, the more likely it was that project success was achieved. (Velcu, 2010:164).

The view expressed by Umble *et al.* (2003:251), Taylor (2000:25) and Zand and Sorensen (1975:541) are that a big contributor to why IT projects fail is the improper definition of business objectives at the start of the IT project. Brynjolfsson and Mendelson (as cited by Ehie & Madsen, 2005:546) support this view by stating that application software project failures are rather due to the inability of the application software package to match the organisation's requirements to solve the business problems than application software packages that were coded incorrectly.

Top management should address the IT gap by aligning information requirements (specific business processes) and the project strategy (end functionality of application software package) to achieve business performance gains (Velcu, 2010:159). Before top management can address the IT gap, they should properly understand what exactly the IT gap is.

This assignment will address the IT gap that exists between the business processes of the organisation and the functionality of the end product (application software package).

The IT gap with regards to application software packages can be divided in the following components (Boshoff, 2011):

- Business model
- Business processes
- Functionality of package
- Data attributes.

The IT gap components are illustrated in Figure 3.1 and will be explained further in the remainder of this section.

For the purpose of explaining the IT gap components the following terms will be used with the following meaning:

Supplier: The supplier refers to the provider of the application software package and represents the IT side as indicated in Figure 3.1.

Client: The client refers to the organisation acquiring the application software package and represents the business side as indicated in Figure 3.1.
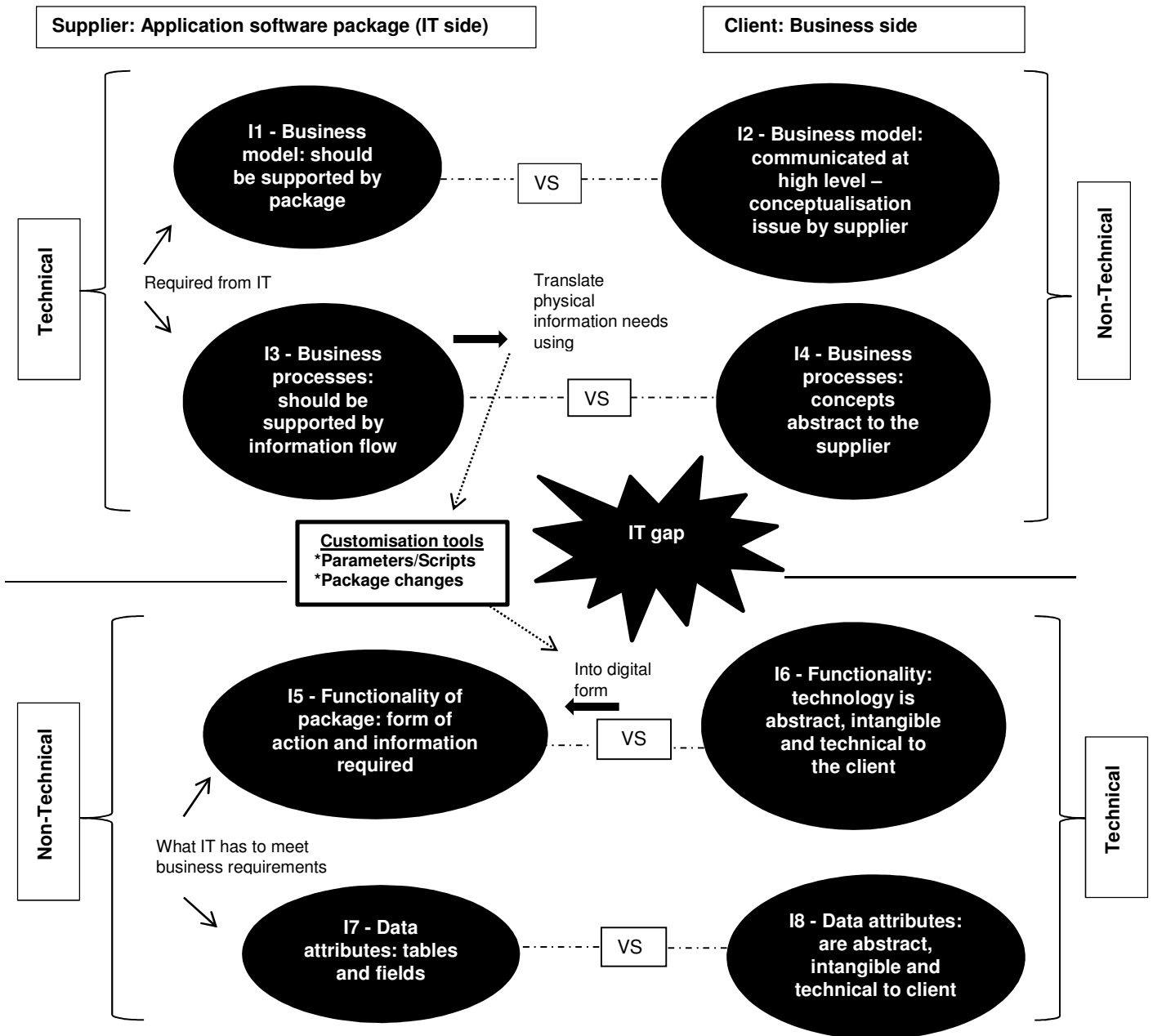
Figure 3.1: Illustration of the IT gap

(Source: Boshoff, 2011)

IT gap component: Business model

I1 - Supplier:

The business model is <u>technical</u> to the supplier and difficult to conceptualise (Boshoff, 2010).

A business model may be defined as the rationale of how an organisation creates, delivers and captures value (Wikipedia, 2011c). The business model typically consists of the industry assumptions (theory of business), strategic objectives, business imperatives (thrust of activity to meet objectives), business policies and business processes of an organisation (Boshoff, 2010).

Top management (client) expects from IT (supplier) to implement an application software package that supports the organisation's business model and specific business processes. The business model is framed within an industry context as well as the maturity scale of the organisation (Boshoff, 2010).

Both the supplier and client need to prepare a business case at the beginning of the project. A business case captures the reasons for initiating a project (Wikipedia, 2011b). To enable IT to prepare the business case they need a proper understanding of the organisation's business model and specific business processes.

Many application software package implementations already fail at the initiation stage of the project. The reason for failure at the initiation stage is answered by Paul (as cited by Smith, 2002:52) who states that it is quite obvious to articulate the business case at the start of the project. However, the supplier of the package usually does not do a business case analysis prior to the start of the project, and if they do, the business case is usually not used once the project starts.

In many instances the organisation's (client) business case differs dramatically from the supplier business case in that the organisation's business case covers the benefits to the organisation in contrast to its costs and risks (Office of Government Commerce,

2009:225). As for the supplier the business case in many instances may be simply making a profit (Office of Government Commerce, 2009:225).

I2 - Client:

To top management the business model is non-technical (Boshoff, 2010).

Establishing clear goals is difficult (Aken, 2008:317) because the supplier and client use different terminology to address the same aspects. Furthermore, these requirements are communicated to the supplier at a very high level. For example, the client would use terms such as (in an accounting environment) order-entry and invoicing, while the supplier would use tables and fields to define the exact same component (Boshoff, 2011).

IT gap component: *Business processes*

I3 - Supplier:

Business processes are technical to the supplier (Boshoff, 2010).

It is difficult for the supplier to conceptualise a business process. Business processes are part of the business model but should be defined separately as it is the business processes that need to be aligned with the application software package. Wikipedia (2011d) defines a business process as a "collection of related, structured activities or tasks that produce a specific service or product for a particular client or clients". It often can be visualised with a flowchart as a sequence of activities (Wikipedia, 2011d).

Business processes should be supported by the information flow of the application software package (Boshoff, 2011). This is supported by the view expressed by Winter (2006:1) in that application software packages used in areas such as data processing, strategies and process control, have to have a flow to the sequence of operations that need to be carried out by the application software package (Winter, 2006:1).

Information flow or workflow applications may be defined as applications that go sequentially through all the activities of a process (Collaborative Computing, s.a.).

However, it is important to remember that many application software packages have basic pre-defined workflow, basic parameters, report writing and limited customisation capabilities (Boshoff, 2011). If the supplier neglects to properly evaluate current business processes at the start of the project it may lead to an unsuccessful application software package (without the necessary functionality) being implemented (Paul, as cited by Smith, 2002:52).

I4 - Client:
Business processes are non-technical to the client (Boshoff, 2010).

Top management should select the application software package that best suits the business requirements of the organisation.

In many instances top management (client) first selects an application software package and tries to change the business processes to suit the application software package instead of changing business processes to improve efficiency (Turbit, 2005:4). If top management try to change business processes to suit the application software package it may lead to the application software package not having the functionality (improper alignment) as required by the organisation.

IT gap component: Functionality of package
I5 - Supplier:
The functionality of the package is non-technical to the supplier (Boshoff, 2010).

Functionality can be defined as what is needed (business requirements) by the user of the package as well as requested properties of inputs and outputs of the application software package (Wikipedia, 2011e).

The supplier would translate physical information requirements (business processes) into digital using customisation tools like parameters/scripts and package changes if the functionality of the package does not perfectly fit the business processes of the organisation (Boshoff, 2011). However, the intended functionality might not always be the implemented functionality due to improper evaluation of business processes.

Application software environments may be either functional rich (application software needs to perform complex calculations usually using simple data structures for calculation) or data rich (application software needs to perform simple calculations using data structures that are more complex). In order for the supplier to determine whether the application environment is functional or data rich, the supplier should first understand whether an organisation's business processes are functional or data rich. (Boshoff, 2011).

Often suppliers may use generic supporting frameworks to assist them with the implementation and configuration of an application software package. It is important for the supplier to note that generic supporting frameworks do not give guidance on "how" the functionality of the specific application software package works. It is advisable to also obtain a product specific supporting framework to assist the supplier with the implementation (Boshoff, 2011).

However, the product specific supporting framework may also have limitations which could result in not adequately addressing the alignment of the business processes with the functionality of the end product (Boshoff, 2011).

I6 - Client:
The functionality of the application software package is highly <u>technical</u> to the client.

Technology is abstract and intangible to the client (Boshoff, 2010) and the client does not understand what actions need to be performed to translate physical business processes

into digital requirements (Boshoff, 2011). The client only knows what the end product should be able to do.

IT gap component: Data attributes

I7 - Supplier:

To the supplier data attributes are non-technical.

Data attributes refer to the technical components (or building blocks) of an application software package that are used to build and configure the specifications that the application software package should meet. Technical components may include table objects, code units, form objects, reports and data ports (Hvitved, 2009:3).

Except for the guidance on implementing the application software package which is received once the package is acquired, little additional guidance is available when configuring technical components of the application software package (Boshoff, 2011).

I8 - Client:

Data attributes are very technical, abstract and intangible to the client (Boshoff, 2010).

Top management often pressure IT departments to install inappropriate technology because they are unaware of crucial technical details (Smith 2002:44).

3.2   Conclusion

It is important to note that if any one or a combination of the above components is inadequately addressed by either top management or application software suppliers it will be most likely that the application software package project will be unsuccessful.

Supporting frameworks are available to assist in the implementation of application software packages. Although these supporting frameworks are available to assist in the implementation of application software packages the success rate of IT projects remains

low (Winter, 2006:vi). The PRINCE2 supporting framework will be discussed and examined to determine to what extends this supporting framework may assist in aligning the business processes of the organisation with the functionality of the application software package (addressing the IT gap).

CHAPTER 4

SELECTION AND DISCUSSION OF SUPPORTING FRAMEWORK

4.1   Introduction

Various supporting frameworks are available that may assist in the implementation of application software package projects.

Supporting frameworks may be divided into two broad categories, namely:

- Generic methodologies, for example
    - **Pr**ojects **IN** **C**ontrolled **E**nvironment (PRINCE2) is a methodology which may be applied to any project (Office of Government office, 2009:4)
    - A guide to the **P**roject **M**anagement **B**ody **o**f **K**nowledge (PMBOK guide) is a methodology which may be applied to most projects (Wikipedia, 2011g).
- Product specific methodologies, for example
    - Microsoft Dynamics Sure Step is a methodology which may be applied to Microsoft Dynamics products
    - SAP implementation guide is a methodology which may be applied to SAP products.

PRINCE2 methodology was selected as supporting framework for purposes of this assignment due to the fact that it is generic: "it can be applied to any project regardless of project scale, type, organisation, geography or culture" (Office of Government Commerce, 2009:4). For this reason PRINCE2 methodology can also be applied to application software packages.

PRINCE2 is a project management methodology developed by the UK Office of Government Commerce (OGC). According to the Office of Government Commerce (2009:5) PRINCE2 applies four key elements to each project: **seven principles** (the guiding obligations and good practices which determine whether the project is being managed using PRINCE2)**, seven processes** (steps from getting started to project

closure), **seven themes** (aspects of project management that must be addressed continually throughout the project) and **project environment** (tailoring PRINCE2 to the specific context of the project). The development of PRINCE2 was aimed at assisting organisations to manage their projects (Office of Government Commerce, 2009:6). The PRINCE2 methodology is based on the experience drawn from thousands of projects (Office of Government Commerce, 2009:1).

The key benefits of PRINCE2 are (Office of Government Commerce, 2009:7):
- Establishment of best practice and governance of project management
- Can be applied to any project
- Provides a common vocabulary for all project members promoting effective communication and meeting the needs of the different levels in the management team
- Provides guidance for recognition of project responsibilities
- Clarifies what the project will deliver
- Provides for the economic and efficient use of management time
- Ensures that participants focus on the feasibility of the project
- Defines a comprehensive structure of reports
- Ensures all stakeholders are appropriately represented in planning
- Promotes learning in organisations
- Promotes consistency of project work
- Is a diagnostic tool, facilitating the assessment and assurance of project work
- Expert support is available for PRINCE2 projects.

PRINCE2 was designed to be applied to any project, not just information system projects (Office of Government Commerce, 2009:4). The tailoring of the methodology is critical to its successful use as "PRINCE2 is not a one size fits all solution; it is a flexible framework that can readily be tailored to any type or size of project" (Office of Government Commerce, 2009:5). If PRINCE2 is not tailored appropriately it is highly unlikely that the project will succeed and meet the requirements set at the beginning of the project (Office of Government Commerce, 2009:14).

According to PRINCE2, "a project is a temporary organisation that is created for the purpose of delivering one or more business products according to an agreed Business Case" (Office of Government Commerce, 2009:3).

The authors of PRINCE2 warn that the use of the PRINCE2 methodology is more than just the adoption of processes and documents alone. It is the adoption of the seven PRINCE 2 principles (refer to section 4.2.1 for discussion on the principles). A PRINCE2 project should be based on the seven PRINCE2 principles (continued business justification, learn from experience, defined roles and responsibilities, manage by stages, manage by exception, focus on products and tailor to suit the project environment). A project using the PRINCE2 methodology is divided into a number of management stages (planning, monitoring and controlling) (Office of Government Commerce, 2009:13) and each management stage is driven by a sequence of processes. The processes together with the principles and themes will be discussed in the remainder of section 4.2.2.

4.2 PRINCE2 project management principles, themes and processes

4.2.1 PRINCE2 principles

The PRINCE2 project management principles are based on the positive and negative experiences drawn from past projects (Office of Government Commerce, 2009:11). The aim of the principles is to provide a framework of good practices for the stakeholders involved in a project (Office of Government Commerce, 2009:11).

The seven principles are listed in Table 4.1 (Office of Government Commerce, 2009:11-14).

Table 4.1: PRINCE2 principles

Principle	Description
Continued business justification	A justifiable reason must exist to start the project, the reason should remain valid through the project life cycle although it may change and it should be documented appropriately.
Learn from experience	It is the responsibility of each project member to learn from previous experience of projects implemented whether successful or unsuccessful.
Defined roles and responsibilities	The roles of the business sponsors, users and suppliers should be defined and represented in the project management team.
Manage by stages	The project should be broken into stages and be planned, monitored and controlled from one stage to another by the project manager. A minimum of two management stages are required: the initiation stage and one or more other management stages.
Manage by exception	Each project objective should have a defined tolerance. If the tolerances that were defined are exceeded, they are directly referred up to the next management level.
Focus on products	A PRINCE2 project is output-oriented and not activity-oriented. The project outcomes are agreed prior to the start of the project and when undertaking the activities of the project.
Tailor to suit the project environment	PRINCE2 should be tailored to ensure the management method relates to the project's environment. If the PRINCE2 methodology is not tailored appropriately, it is unlikely that the project outcomes will be achieved.

4.2.2 PRINCE2 themes

PRINCE2 describes the themes that are applicable to all the management processes (Office of Government Commerce, 2009:17). The themes are integrated and should be addressed continually in the project management processes (Office of Government Commerce, 2009:17).

The seven themes are described in Table 4.2 (Office of Government Commerce, 2009:17).

Table 4.2: PRINCE2 themes

Theme	Description
Business case	The purpose of the business case theme is to establish the existence of a viable business case at the beginning of the project and to establish measures to determine whether the business case remains viable throughout the project life cycle to support the original decision making investment. If the business case becomes non-viable the project should be stopped immediately.
Organisation	This theme defines the roles and responsibilities in the project management team and establishes the project's structure of accountability.
Quality	The theme outlines the attributes of the end product to be delivered as well as laying down quality inspection methods to determine whether the requirements are delivered at the end of the project.

Theme	Description
Plans	The plan theme has to do with the planning of the project, which for example includes time estimates, cost estimates, resource estimates analysing risks and defining the end product. This theme describes the steps that should be followed to develop plans, as well as the techniques that should be applied when developing the plans. When using the PRINCE2 methodology the project continues on the basis of a series of permitted plans. The focus of the plans is continuous communication and control during the project. PRINCE2 emphasises that the proper documentation of the plans is very important.
Risk	The purpose of this theme is to outline an approach for project managers to identify, assess and address risk in all the project plans as well as the broader project environment.
Change	This theme describes how the project manager would go about to identify, assess and control any possible and permitted changes to the baseline aspects (plans and completed products) during the project life cycle.
Progress	The progress theme's purpose is to establish methods to monitor and match actual accomplishments against those set at the beginning of the project, provide a forecast of the project's continued viability and control any intolerable abnormalities.

4.2.3 PRINCE2 processes

PRINCE2 follows a process-based approach towards project management (Office of Government Commerce, 2009:113).

The PRINCE2 methodology consists of seven processes. The seven processes arrange the set of activities essential to direct, manage and deliver a successful project.

The processes are (Office of Government Commerce, 2009:115):

- Starting up a project (SU)
- Directing a project (DP)
- Initiating a project (IP)
- Controlling a stage (CS)
- Managing product delivery (MP)
- Managing a stage boundary (SB)
- Closing a project (CP).

Figure 4.1 illustrates the PRINCE2 processes.

	Pre-project	Initiation stage	Subsequent delivery stage (s)	Final delivery stage
Directing	SU	Directing a project		
Managing		SB / IP	SB / Controlling a stage	CP / Controlling a stage
Delivering			Managing product delivery	Managing product delivery
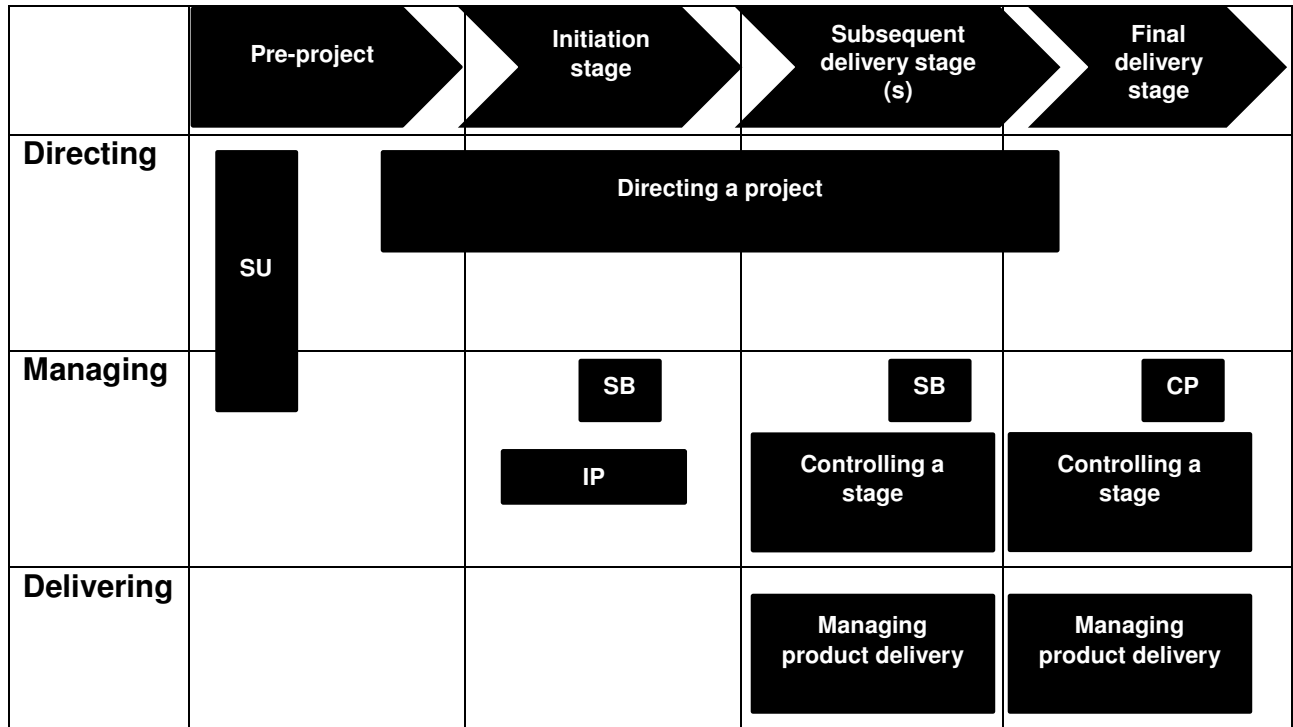
Figure 4.1: The PRINCE2 processes

(Source: Office of Government Commerce, 2009)

Key

SU	Starting up a project
IP	Initiating a project
SB	Managing a stage boundary
CP	Closing a project

The PRINCE2 processes comprise of a number of activities and activities comprise of a number of recommended actions aimed at achieving a certain result.

In the following section a brief overview of each PRINCE2 project management process is provided. Each process is supported with a figure in the form of a flow diagram of activities.

4.2.3.1 Starting up a project (SU)

This process is intended to ensure that the requisites for initiating a project are in place. The question that needs to be answered during this process is: "do we have a viable and worthwhile project?" before the project is initiated (Office of Government Commerce, 2009:121).

The end result of this process is a project brief defining what and why the project needs to be done, the outcomes to achieve, the stakeholders who need to be involved, how and when the project will be done. The starting up a project process comprises the following activities (Office of Government Commerce, 2009:122):

- Appointment of the executive and project manager
- Capturing of lessons learned from previous projects
- Design and appointment of the project management team
- Preparation of the outline business case (clients expectations)
- Decide the project approach and assemble the project brief
- Planning the initiation stage.

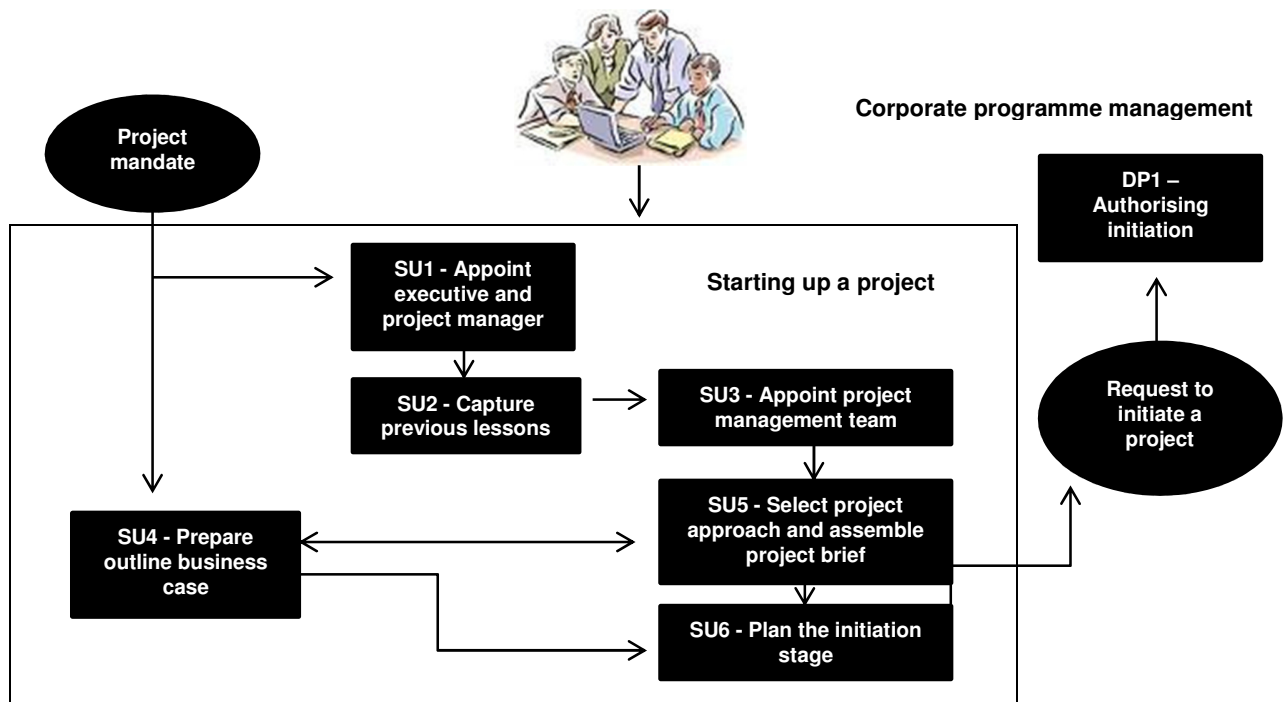The activities and their relations are illustrated in Figure 4.2.

Figure 4.2: Starting up project process (SU)

(Source: Office of Government Commerce, 2009)

4.2.3.2 Directing a project (DP)

The directing of a project process entails authorising work to be carried out and the resources to be dedicated to the project. This process covers the activities of the level of management above the project manager - the project board (Office of Government Commerce, 2009:135). The project board manages this process by exception (Office of Government Commerce, 2009:135). Managing by exception entails that only exceptions to stage plans or project plans need to be escalated for approval to the project board. The project board monitors the project via progress reports and controls the project through a number of decision points (Office of Government Commerce, 2009:136). The activities performed in this process are as follows (Office of Government Commerce, 2009:135):

- Authorisation of the initiation of the project.
- Authorisation to proceed and deliver the project.

- The project board reviews the performance of the current stage and approves the stage plan for the next stage. If an exception occurs during a stage, the board needs to approve the exception plan.
- The project board members may offer informal guidance throughout the project. Circumstances that may prompt ad hoc direction include: resolving conflict areas, responding to progress reports, responding to external influences or any other major threat to the project success.
- Authorising a controlled closure for the project.

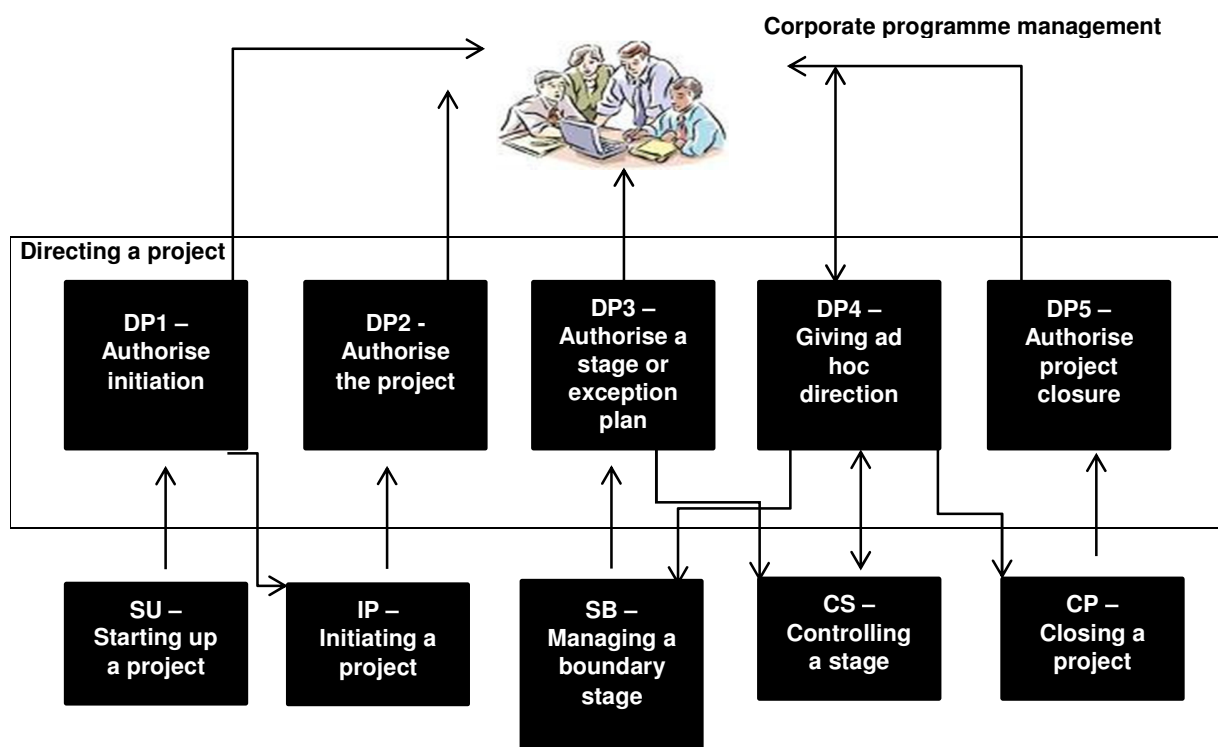The activities and their relations are illustrated in Figure 4.3.

Figure 4.3: Directing a project process (DP)

(Source: Office of Government Commerce, 2009)

4.2.3.3 Initiating a project (IP)

This process is designed to plan the entire project and define the project in terms of its business benefits, risks, products, activities, quality, and resources usage. It ensures that all stakeholders understand the what, why and how of the project. The initiating a project process consists of the following activities (Office of Government Commerce, 2009:150):

- The preparation of the risk management strategy which includes risk tolerances, timing of risk management activities, techniques that will be used and the reporting requirements

- The preparation of the configuration management strategy which includes change control procedures

- The preparation of the quality management strategy which defines how the required product quality will be achieved

- The preparation of the communication management strategy between the different stakeholders of the project

- Setting up effective project controls which is a prerequisite for identifying exceptions or deviations from the original project plan

- Creating the project plan which entails establishing the timescale and resource requirements

- The business case produced during starting up the project should be refined or updated to show the estimated time and cost and the aggregated risks

- The project initiation documentation, explaining the what, why, who, how, where, when and how much should be gathered and be made available for guidance to all stakeholders of the project.

The activities and their relations are illustrated in Figure 4.4.
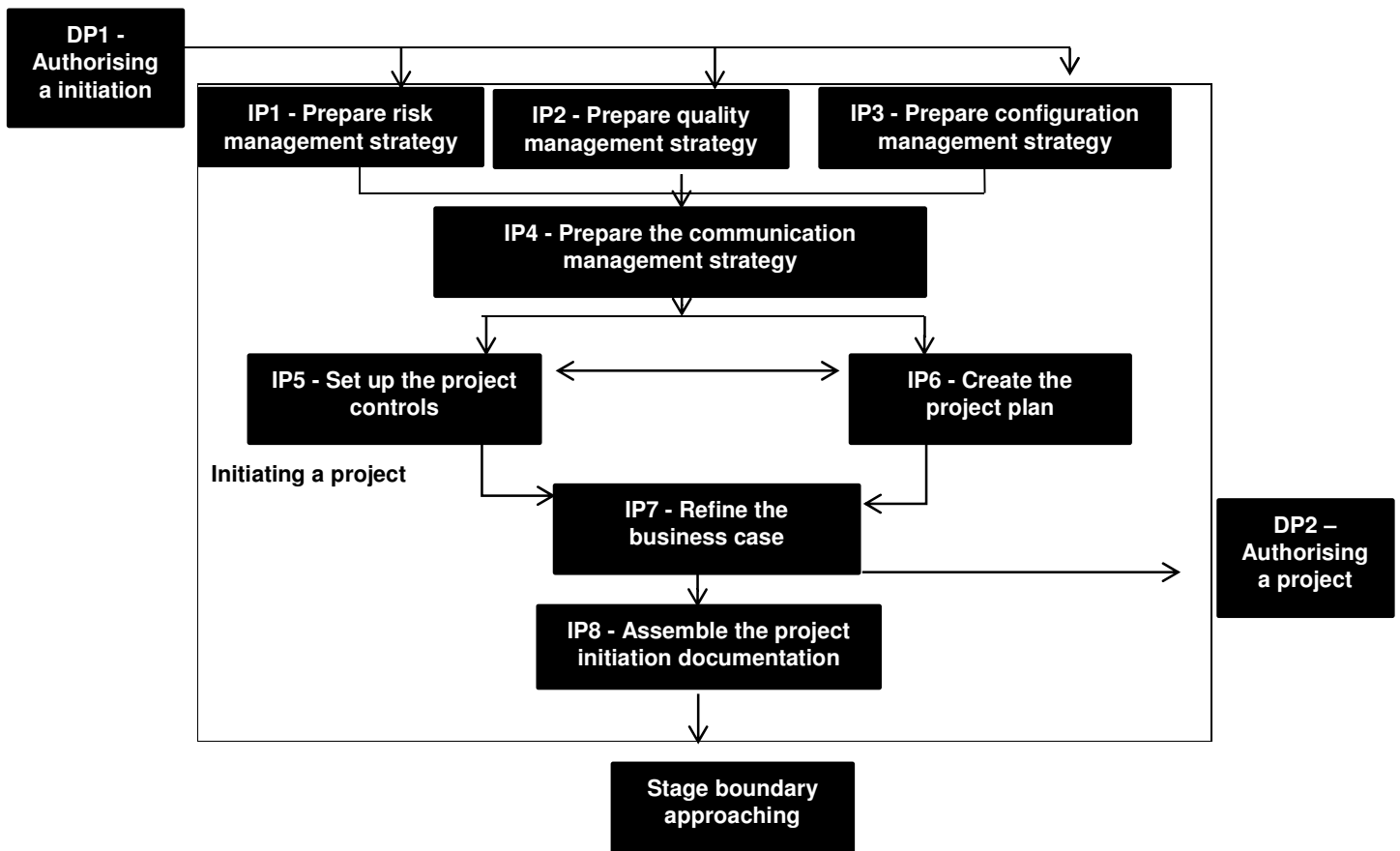
Figure 4.4: Initiating a project process (IP)

(Source: Office of Government Commerce, 2009)

4.2.3.4 Controlling a stage (CS)

The controlling a stage process is concerned with the day-to-day management of each stage (Office of Government Commerce, 2009:168). This stage includes giving authorisation for work to be conducted, the monitoring of progress information, reviewing and dealing with risk situations, reporting highlights and taking the necessary corrective action (Office of Government Commerce, 2009:168). The following activities are performed in this process (Office of Government Commerce, 2009:168):

- Authorising the commencement of work packages (set of information relevant to the creation of products)

- Reviewing the work package status at regular intervals through highlight reports
- Reporting of completed work packages
- Capturing and examining proposed changes and risks, and escalating the changes and risks where necessary, to the project board
- Taking corrective action, if appropriate, towards risks.

The activities and their relations within this stage are depicted in Figure 4.5.

Figure 4.5: Controlling a stage process (CS)
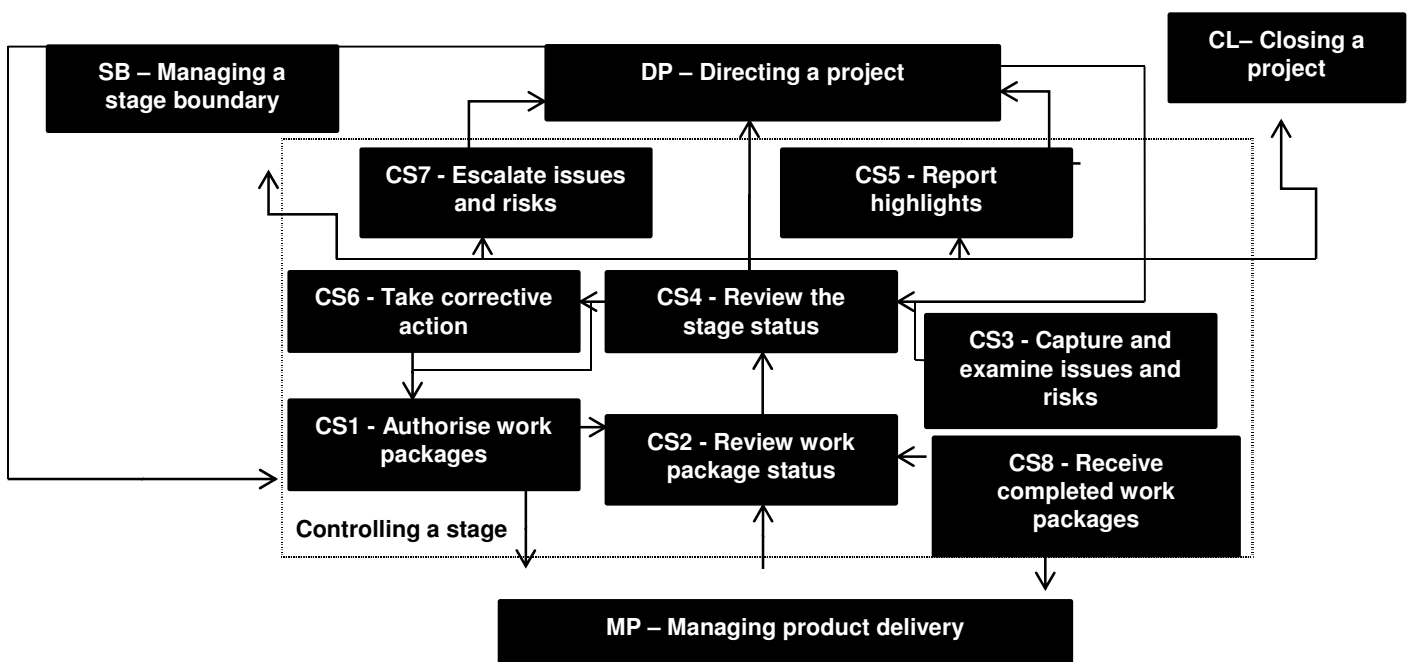
(Source: Office of Government Commerce, 2009)

4.2.3.5 Managing product delivery (MP)

The objective of the managing product delivery process is to ensure that planned products are created and delivered by the project team to the client (Office of Government Commerce, 2009:185). The following activities are performed in this process (Office of Government Commerce, 2009:186):

- The team manager should negotiate with the project manager what needs to be delivered, project constraints, reporting requirements and agree that the requirements of the work packages are achievable
- The team manager should ensure the work package is executed and monitored as per the requirements set in the authorised work package
- The team manager should ensure that the end product meets the quality criteria set and only then notify the project manager of the completion of the work package.

The activities and their relations are illustrated in Figure 4.6.
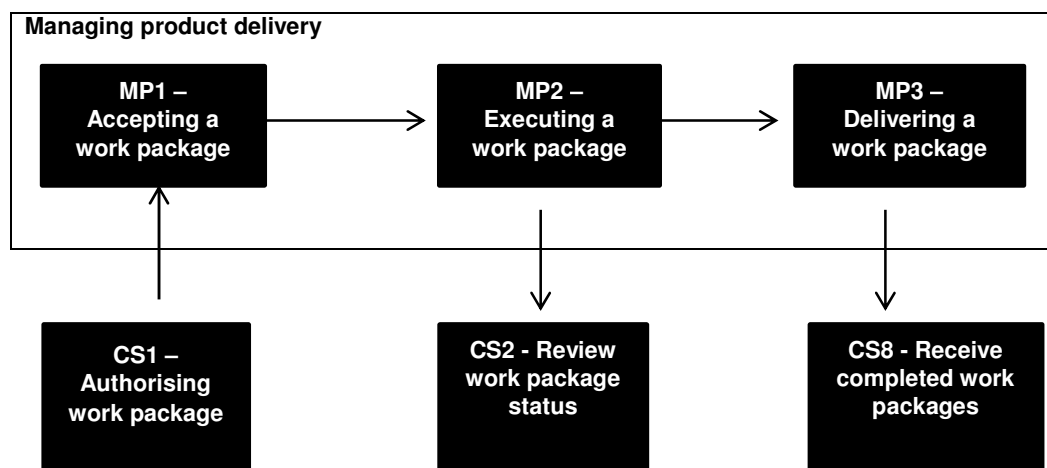
Figure 4.6: Managing product delivery process (MP)

(Source: Office of Government Commerce, 2009)

4.2.3.6 Managing a stage boundary (SB)

The purpose of this process is to provide the project board with sufficient information so that the success of the current stage can be reviewed, the next stage may be approved, continued business justification may be evaluated and the tolerability of risks can be confirmed (Office of Government Commerce, 2009:193). The managing a stage boundary process should be executed at the end of each management stage (Office of

Government Commerce, 2009:193). The following activities are defined in PRINCE2 (Office of Government Commerce, 2009:194):

- Assurance should be provided to the project board that all products planned for the current stage have been successfully completed.
- When reaching the completion of a boundary stage, the stage plan or exception plan for the next management stage is updated to show actual progress of the project versus planned progress for the current stage.
- The business case should be revised at the end of each stage.
- The end of the stage should be reported to the project board. The project manager should provide the project board with information stating the continuing ability of the project to meet the project plan and business case. The project manager should also advise the project board on the overall risk situation of the project.
- Exception reports should be produced for approval by the project board if the project deviates beyond tolerances.

The activities and their relations are illustrated in Figure 4.7.
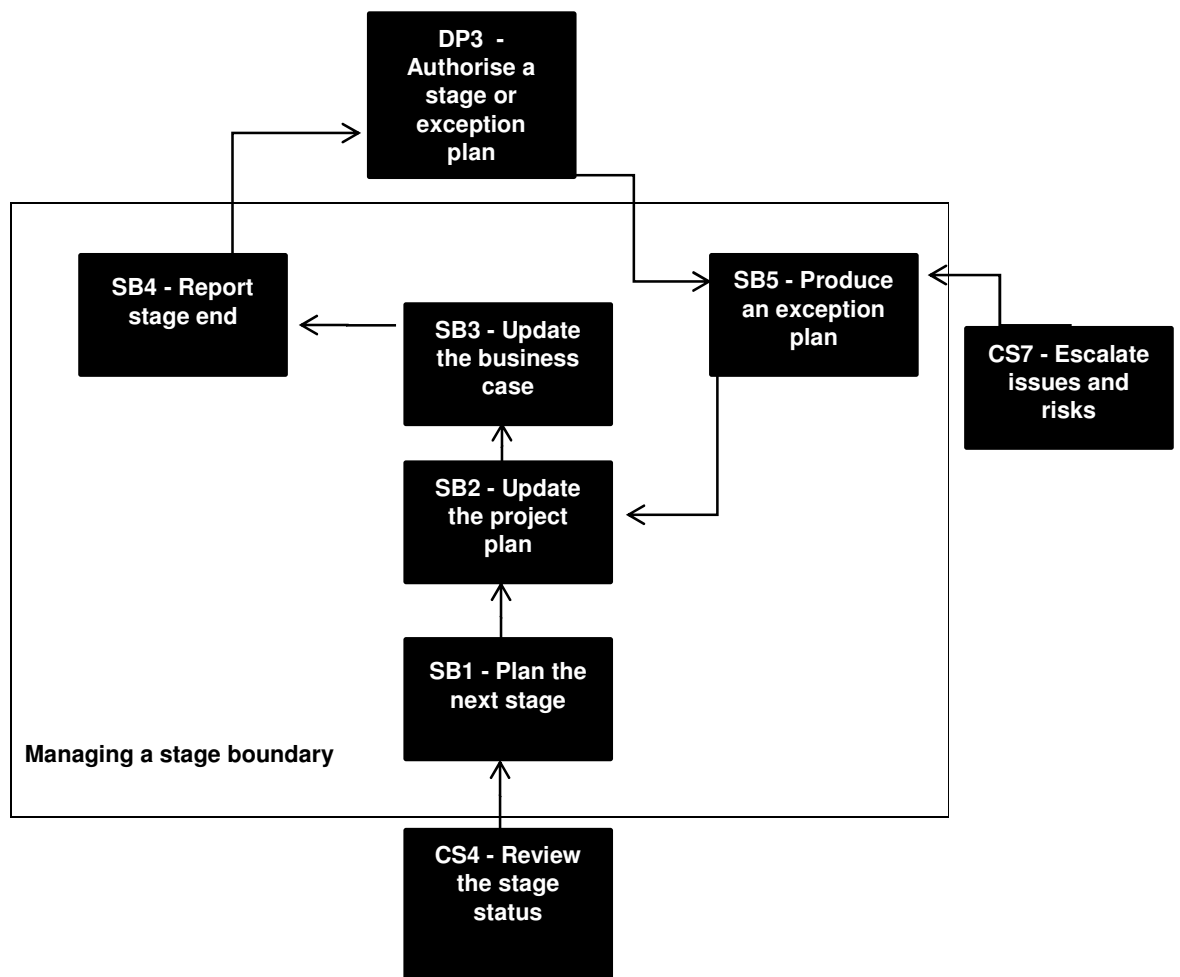
Figure 4.7: Managing a stage boundary process (SB)

(Source: Office of Government Commerce, 2009)

4.2.3.7 Closing a project (CP)

This process provides a controlled closure of the project (Office of Government Commerce, 2009:206). PRINCE2 defines the following activities that are executed at the end of the project (Office of Government Commerce, 2009:207):

- Check that objectives set out in the project initiation document have been met
- Confirm the acceptance of the product by the client
- If the project was closed prematurely, the project manager should ensure the issue register, project plan and product status account are appropriately updated
- End products should be handed over to the client

- Arrangements should be made for maintenance of the end product
- An evaluation of the project should be conducted, assessing how successful or unsuccessful the project was for guidance for future projects
- The project manager should ensure that project information is archived, recommend closure to the project board and release the resources.

The activities and their relations are illustrated in Figure 4.8.
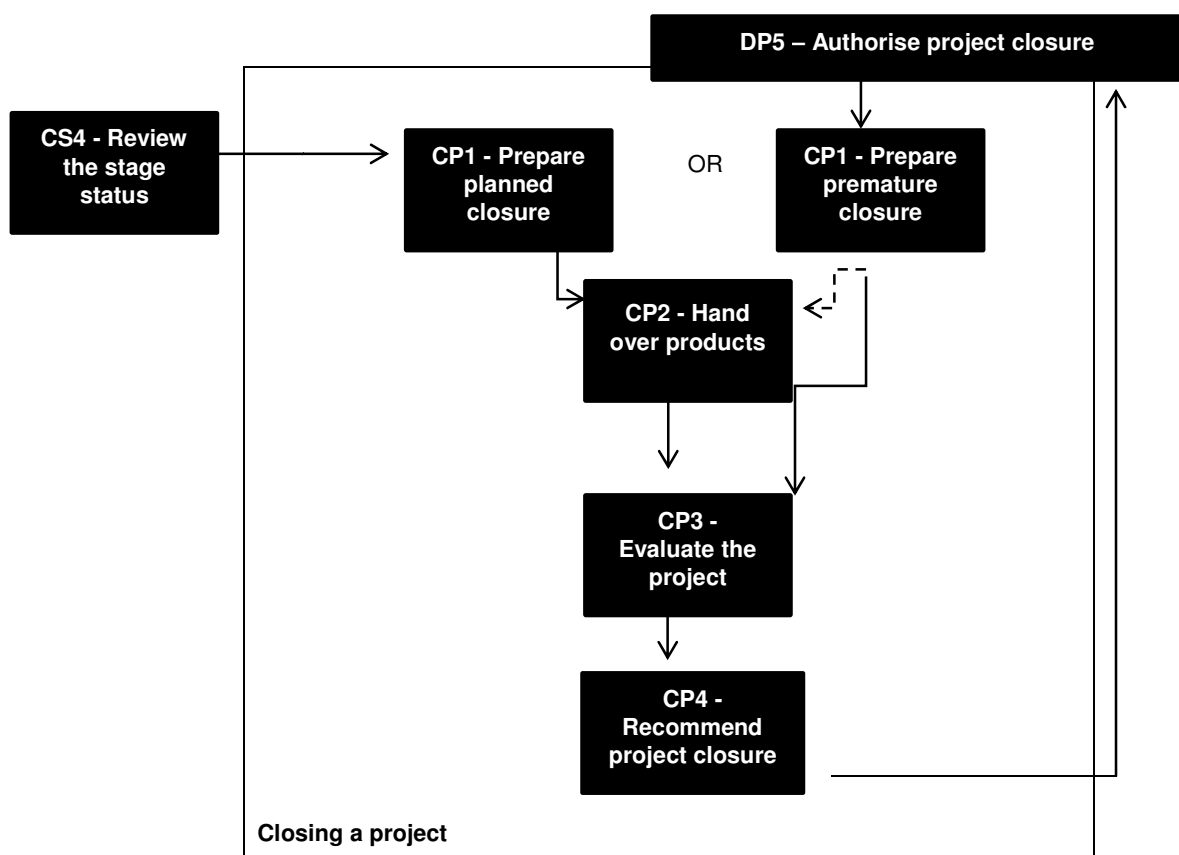
Figure 4.8: Closing a project process (CP)

(Source: Office of Government Commerce, 2009)

The principles, themes and processes are all linked to ensure effective execution of the project. The seven project management processes drive the management of the project and the processes are supported by the themes.

PRINCE2 addresses the management of the project and the management of resources. However, the following topics are outside the scope of PRINCE2 (Office of Government Commerce, 2009:6):

- **Specialist aspects** – PRINCE2 is generic and industry or type specific activities are excluded
- **Detailed techniques** – The techniques that PRINCE2 describes are only applicable to projects using the PRINCE2 methodology
- **Leadership capability** – Interpersonal skills (for example leadership skills, motivational skills) are excluded.

The Office of Government Commerce (2009:6) recommends that consideration should be given to use other best practice guides to address the topics outside the scope of the PRINCE2 methodology.

CHAPTER 5

MAPPING OF PROJECT FAILURE REASONS

A matrix table was compiled summarising the reasons for IT project failure from each source reviewed for this assignment in section 2.3 (limited to reasons recurring most frequently in literature reviewed). The reasons from literature were mapped to the reasons listed by the Office of Government Commerce in the best practice guide, Common Causes of Project Failure (s.a.) (the publisher of PRINCE2).

Lastly the reasons were mapped to the PRINCE2 methodology (Office of Government Commerce, 2009) to determine whether the PRINCE2 methodology adequately addresses the project failure reasons listed in the table. The results are shown in Table 5.1.

The following approach was followed in preparing the matrix table:
- White (as cited in Plotnikova, 2007:22) lists three categories of an application software implementation project's risk environment. IT project reasons for failures were divided into one of the three risk categories. The three risk categories are as follows:
  - **Business environment risks** – risks beyond the project manager's control that could influence the success of the project
  - **Project management risks** – risks that may lead to the improper planning and organising of the work that should be executed during the project
  - **Project execution risks or technical risks** – risks that may lead to the specification deliverables, set to align business processes with the application software package at the beginning of the project, not being properly executed.
- The most recurring reasons mentioned in literature reviewed for this assignment, were listed as reasons under one of above categories. The reasons are listed in column two of the table and the sources who mentioned the reason in their studies are indicated in column three of the table.

- The reasons from literature reviewed were mapped to the reasons listed by the Office of Government Commerce (Common Causes of Project Failure, s.a.) for project failure in column four of the table (indicated by "X").

- In column five of the table the reasons were mapped to the PRINCE2 methodology (Office of Government Commerce, 2009) to determine whether an organisation will be able to mitigate or reduce the specific reason, in the opinion of the publishers of PRINCE2, the Office of Government Commerce, if the organisation applies the PRINCE2 methodology (indicated by "X").

Table 5.1: Mapping of reasons in literature to reasons per Office of Government Commerce and PRINCE2

	Reasons	Source	Identified as reason by OGC (publisher of PRINCE2)	Reason mitigated or reduced by applying PRINCE2 principles
	Business Environment			
R1	Poor requirements management (unclear objectives or business case)	Cerpa & Verner, 2009:131; Demir, 2009; McManus & Wood-Harper, 2007:42; Al Neimat, 2005:3; Coley Consulting, 2005:1; Chin, 2003:1; Umble *et al.,* 2003:251; Smith, 2002:57; Sauer & Cuthbertson, 2003:60; Taylor, 2000:24; May, 1998:2; Zand & Sorensen,1975:541; INTOSAI (s.a.)	X	X
R2	Lack of top management commitment and support	Demir, 2009; Aloini *et al.,* 2007:559; McManus & Wood-Harper, 2007:42; Kappelman *et al.,* 2006:34; Al Neimat, 2005:3; Sauer & Cuthbertson, 2003:60; Smith, 2002:57; Taylor, 2000:24	X	X
R3	Lack of clear links between project and organisation key strategic priorities (alignment)	Velcu, 2010:160; Aloini *et al.,* 2007:559; McManus & Wood-Harper, 2007:42; Kappelman *et al.,* 2006:34; Tilmann & Weinberger, 2004:28; Ehie & Madsen, 2005:546; INTOSAI,(s.a.)	X	(*1)

	Reason	Source	Identified as reason by OGC (publisher of PRINCE2)	Reason mitigated or reduced by applying PRINCE2 principles
	Project Management			
R4	Inadequate business process re-engineering	Aloini *et al.,* 2007:559; McManus & Wood-Harper, 2007:42; Kim *et al.,* 2005:164; Turbit, 2005:5	(******6)	(**2)
R5	Underestimation of implementation timeline and budget (improper planning)	Cerpa & Verner, 2009:131; Demir, 2009; Thomas & Fernandez, 2008:736; Kappelman *et al.,* 2006:34; Winter, 2006:13; Al Neimat, 2005:3; Coley Consulting, 2005:1; Turbit, 2005:5; Holt, 2003:2; Sauer & Cuthbertson, 2003:60; Smith, 2002:57; Taylor, 2000:24; May, 1998:2	X	X
R6	Underestimation of the IT solution complexity (improper planning)	Cerpa & Verner, 2009:131; Demir, 2009; Thomas & Fernandez, 2008:736; Kappelman *et al.,* 2006:34; Winter, 2006:13; Al Neimat, 2005:3; Smith, 2002:57	X	X
R7	Insufficient risk management	Cerpa & Verner, 2009:131; Demir, 2009; Chen *et al.,* 2009:157; Deng & Bian, 2008:72; McManus & Wood-Harper, 2007:42; Taylor, 2000:24; INTOSAI, (s.a.)	X	X

	Reason	Source	Identified as reason by OGC (publisher of PRINCE2)	Reason mitigated or reduced by applying PRINCE2 principles
	Project Management (continued)			
R8	"People" issues (e.g. Not rewarding staff, no work life balance, staff added late to project, unable to work as a team or conflict among stakeholders, poor interpersonal skills, internal politics, resistance to adapt)	Cerpa & Verner, 2009:131; Demir, 2009; Chen *et al.,* 2009:157; McManus & Wood-Harper, 2007:42; Kappelman *et al.,* 2006:34; Kim *et al.,* 2005:164; Turbit, 2005:3; Chin, 2003:1; Holt, 2003:2; Sauer & Cuthbertson, 2003:60; Taylor, 2000:24; May, 1998:2	**X**	**X(***3)**
R9	Insufficient end user involvement	Cerpa & Verner, 2009:131; Demir, 2009; Aloini *et al.,* 2007:559; McManus & Wood-Harper, 2007:42; Kappelman *et al.,* 2006:34; Al Neimat, 2005:3; Coley Consulting, 2005:1; Chin, 2003:1; Sauer & Cuthbertson, 2003:60; Smith, 2002:57; May, 1998:2; INTOSAI, (s.a.)	**X**	**X**
R10	Inappropriate methodology used	Cerpa & Verner, 2009:131; Chen *et al.,* 2009:158, McManus & Wood-Harper, 2007:42; Chin, 2003:1; Taylor, 2000:26; Sauer & Cuthbertson, 2003:60	**(******6)**	**(****4)**

	Reason	Source	Identified as reason by OGC (publisher of PRINCE2)	Reason mitigated or reduced by applying PRINCE2 principles
	Project Management (continued)			
R11	Lack of resources (improper planning)	Cerpa & Verner, 2009:131; Kim *et al.,* 2005:164; Turbit, 2005:5	X	X
R12	Poor definition of scope of project	Demir, 2009; Kappelman *et al.,* 2006:34; Thomas & Fernandez, 2008:736; Al Neimat, 2005:3; Chin, 2003:1; Smith, 2002:57; INTOSAI, (s.a.)	X	X
R13	Poor communication between stakeholders	Demir, 2009; Park & Keil, 2009:45; McManus & Wood-Harper, 2007:42; Kappelman *et al.,* 2006:34; Al Neimat, 2005:3; Smith, 2002; Keil & Robey, 2001:87; Taylor, 2000:24; May, 1998:2	X	X
R14	Improper status monitoring of project (identifying early warning signs)	Demir, 2009; Bennatan, 2009:5	X	X

	Reason	Source	Identified as reason by OGC (publisher of PRINCE2)	Reason mitigated or reduced by applying PRINCE2 principles
	Project Management (continued)			
R15	Poor project management capability and planning	Demir, 2009; Chen *et al.*, 2009:157; Aloini *et al.*, 2007:559; McManus & Wood-Harper, 2007:42; Kappelman *et al.*, 2006:34; Ehie & Madsen, 2005:555; Gargeya & Brady, 2005:511; Holt, 2003:2; Umble *et al.*, 2003:245; Smith, 2002:57; Jurison, 1999:4; Sauer & Cuthbertson, 2003:60; Taylor, 2000:24; May, 1998:2; INTOSAI, (s.a.)	X	X
	Project Execution (technical)			
R16	Improper supplier management	Chen *et al.,* 2009:157; McManus & Wood-Harper, 2007:42	X	X
R17	Insufficient software metrics	Aloini *et al.,* 2007:559; McManus & Wood-Harper, 2007:42	(******6)	(**2)
R18	Insufficient training of users	Aloini *et al.,* 2007:559; McManus & Wood-Harper, 2007:42; Turbit, 2005:5; Taylor, 2000:24	X	X(***3)

	Reason	Source	Identified as reason by OGC (publisher of PRINCE2)	Reason mitigated or reduced by applying PRINCE2 principles
	Project Execution (technical) (continued)			
R19	Poor configuration management (poor change control management)	Cerpa & Verner, 2009:131; Chen *et al.,* 2009:157; Demir, 2009; Aloini *et al.,* 2007:559; McManus & Wood-Harper, 2007:42; Kappelman *et al.,* 2006:34; Al Neimat, 2005:3; Coley Consulting, 2005:1; Kim *et al.,* 2005:164; Turbit, 2005:5; Holt, 2003:2; Sauer & Cuthbertson, 2003:60; Smith, 2002:57; Taylor, 2000:24; INTOSAI, (s.a.)	X	X
R20	Insufficient user acceptance testing	Cerpa & Verner, 2009:131; McManus & Wood-Harper, 2007:42, Coley Consulting, 2005:1; Taylor, 2000:24	X	X(***3)
R21	Poor understanding by staff of solution capabilities (lack of technical competence)	Demir, 2009; Aloini *et al.,* 2007:559; Kappelman *et al.,* 2006:34; Sauer & Cuthbertson, 2003:60; Smith, 2002:57	X	X(*****5)
R22	Inability to break up implementation into manageable steps	McManus & Wood-Harper, 2007:42	X	X

Key

1*	Not addressed in PRINCE2 methodology although listed as reason by OGC. (For justification refer to Chapter 6.)
2**	Not addressed in PRINCE2 methodology, as this reason is product specific.
3***	Addressed (or referenced made) in PRINCE2 methodology, but not adequately addressed.
4****	Not specifically addressed in PRINCE2, but PRINCE2 is a methodology. It is important to note that the PRINCE2 methodology is not product specific. (For justification refer to Chapter 6.)
5*****	PRINCE2 methodology only address competency with regards to managing skills of a project.
6******	Not listed as a reason by Office of Government Commerce, as the reason is industry specific.

It is important to note that all the reasons identified by the Office of Government Commerce (Common Causes of Project Failure, s.a.) are addressed in PRINCE2 except reason three (R3), namely lack of clear links between the project and the organisations key strategic priorities. It is clear that the IT gap (alignment) is not addressed in the PRINCE2 methodology.

In Chapter 6 reason R3 will be further explored. Futhermore, the activities of the PRINCE2 processes will be examined to determine any other shortcomings and weaknesses contributing to misalignment.

CHAPTER 6

SHORTCOMINGS AND CONTRIBUTING WEAKNESSES IDENTIFIED IN THE PRINCE2 METHODOLOGY

6.1   Introduction

Project tools (e.g. PRINCE2) are available to assist in implementation of application software packages (Boshoff, 2011). The question arises why then does a gap still exist and why are application software projects still unsuccessful although the project tools are used?

Taylor (2000:26) expressed the view that none of the generic project management methods like PRINCE2 is perfect, although each one has facets which are more suitable to one IT project than another.   Each organisation should adjust the generic project management method to suit the organisations specific needs (Taylor 2000:26).

Jackson and Klobas further (2008:331) comment …

> what is happening in an ISD (information system development) project is far more complex than the simple translation of a description of an external reality into instructions for a computer.   It is the emergence and articulation of multiple, indeterminate, sometimes unconscious, sometimes ineffable realities and the negotiated achievement of a consensus of a new, agreed reality in an explicit form, such as a business or data model, which is amenable to computerization.

It is important to note that the PRINCE2 project management methodology is generic. Although it can be applied to any project assisting an organisation in reducing the risk of project failure with great success, it should be tailored appropriately to meet the needs of the organisation (Office of Government Commerce, 2009:215). However, if a project is template-driven and not tailored it can lead to robotic project management (Office of Government Commerce, 2009:215).

In Appendix B, Table B.1 of the PRINCE2 guide on governance the following is stated (Office of Government Commerce, 2009:265):

Project management principle	Addressed by PRINCE2?
"A coherent and supportive relationship is demonstrated between the overall business strategy and the project portfolio".	"Partially. PRINCE2 project should demonstrate alignment to corporate strategy through its Business Case. PRINCE2 does not provide guidance on portfolio management".

In the opinion of the authors of PRINCE2 the alignment of the business strategy and project is addressed partially. After studying the PRINCE2 manual it is concluded that the business case is discussed in the PRINCE2 methodology. However, with regards to alignment, the authors mentioned several times that corporate objectives should be aligned to the project strategy, without providing any further detail thereon (Office of Government Commerce, 2009:19-28).

From above it is clear that PRINCE2 does not address all factors that will ensure project success, leaving a gap in the PRINCE2 methodology. One factor that the PRINCE2 methodology does not address is the lack of clear links between project and organisation key strategic priorities (alignment) (refer to reason R3 in Table 5.1).

Although most authors divide the implementation of projects (specific application software package projects) into four categories, the PRINCE2 methodology divides the implementation into seven categories or processes.   In their article Chen *et al.* (2009:158) state that each one of the implementation stages (initiation, planning, execution and controlling and closing) may contribute risks that can lead to application software package project failures (refer to section 2.4 for detail thereon).

In Table 6.1 the four implementation stages identified by Jurison (1999:8) are mapped to the seven PRINCE2 processes. The mapping in Table 6.1 is to indicate that the

seven PRINCE2 processes can each be linked to one of the four implementation stages as identified by Jurison (1999:8). For this reason the risks identified by Chen *et al.* (2009:158) are also applicable to the seven PRINCE2 processes. This resulted in the examination of the PRINCE2 activities for any additional shortcomings and weaknesses.

Table 6.1: Project implementation stages mapped to PRINCE2 processes

Implementation stages: Jurison	PRINCE2 processes
Initiation	Starting up a project
Planning	Initiating a project
Execution	Managing product delivery & managing a stage boundary
Controlling and closing	Directing a project, controlling a stage & closing a project

(Source: Jurison, 1999 & Office of Government Commerce, 2009)

6.2 Shortcomings and contributing weaknesses identified

In Table 6.2 the PRINCE2 processes together with the activities per process are summarised. Activities where weaknesses may exist, specifically with regards to the implementation of application software packages, were indicated in the table (indicated with "X").

Weaknesses identified that are applicable to all PRINCE2 activities are listed in Table 6.3.

The shortcomings and weaknesses identified in the PRINCE2 activities contributing to improper alignment of business processes with the functionality of the application software package were grouped together into a number of categories. The weakness category applicable is indicated in the last column of the table and will be discussed in the remainder of this section.

Table 6.2: PRINCE2 processes and activities summarised and weaknesses indicated

Process	Activity	Weakness	Reason
Starting up a project	Appoint the executive and the project manager	X	Capability/Competence issue (W1)
	Capture previous lessons	X	Planning issue (W4)
	Design and appoint the project management team	X	Capability/Competence issue (W1)
	Prepare the outline business case	X	Capability/Competence issue (W1) & Alignment issue (S1)
	Select the project approach and assemble the Project Brief	X	Capability/Competence issue (W1) & Alignment issue (S1)
	Plan the initiation stage	X	Capability/Competence issue (W1) & Alignment issue (S1)
Directing a project	Authorise initiation	X	Capability/Competence issue (W1)
	Authorise the project	X	Capability/Competence issue (W1)
	Authorise a stage or exception plan	X	Communication issue (W2) & Capability/Competence issue (W1)
	Give ad hoc direction	X	Capability/Competence issue (W1)
	Authorise project closure	X	Capability/Competence issue (W1)

Process	Activity	Weakness	Reason
Initiating a project	Prepare the risk management strategy	**X**	Capability/Competence issue (W1) & Planning issue (W4)
	Prepare the configuration management strategy	**X**	Capability/Competence issue (W1) & Planning issue (W4)
	Prepare the communication management strategy	**X**	Capability/Competence issue (W1) & Planning issue (W4)
	Set up the project controls	**X**	Capability/Competence issue (W1) & Planning issue (W4)
	Create the project plan	**X**	Capability/Competence issue (W1) & Planning issue (W4)
	Refine the business case	**X**	Capability/Competence issue (W1) & Alignment issue (S1)
	Assemble the project initiation documentation	**X**	Tailoring and integration issue (W6)
Controlling a stage	Authorise a work package	**X**	Capability/Competence issue (W1)
	Review a work package status	**X**	Capability/Competence issue (W1)
	Receive completed work packages		No weakness – activity entails confirmation of completion and updating of the necessary registers

Process	Activity	Weakness	Reason
Controlling a stage (continued)	Review the stage status	**X**	Capability/Competence issue (W1) & Communication issue (W2)
	Report highlights	**X**	Communication issue (W2)
	Capture and examine issues and risks	**X**	Capability/Competence issue (W1) & Communication issue (W2)
	Escalate issues and risks	**X**	Communication issue (W2)
	Take corrective action	**X**	Capability/Competence issue (W1) & Communication issue (W2)
Managing product delivery	Accept a work package	**X**	Capability/Competence issue (W1)
	Execute a work package	**X**	Capability/Competence issue (W1)
	Deliver a work package		No weakness - activity entails confirmation of completion and updating of the necessary registers
Managing boundary stage	Plan the next stage	**X**	Capability/Competence issue (W1), Communication issue (W2) & Alignment issue (S1)
	Update the project plan		No weakness – activity entails mainly updating of registers and logs
	Update the business case	**X**	Capability/Competence issue (W1) & Alignment issue (S1)
	Report stage end	**X**	Testing issue (W7)

Process	Activity	Weakness	Reason
Managing boundary stage (continued)	Produce an exception plan		No weakness - activity entails confirmation of completion and updating of the necessary registers
Closing a project	Prepare planned closure	**X**	Capability/Competence issue (W1) & Alignment (S1)
	Prepare premature closure	**X**	Capability/Competence issue (W1) & Alignment issue (S1)
	Hand over products	**X**	Testing issue (W7)
	Evaluate the project		No weakness - activity entails assessing how successful or unsuccessful the project was. If the evaluation shows that the project activity is neglected it might have an effect on future projects but not on the current project
	Recommend project closure		No weakness - activity entails confirmation of completion and updating of the necessary registers

Table 6.3: Weaknesses in PRINCE2 applicable to all processes hindering proper alignment

Weakness	Weakness category
Insufficient emphasis on people issues which include leadership, motivational and other interpersonal skills e.g. team work.	Soft skill issue (W5)
Insufficient training of all parties involved in project.	Training issue (W8)
Difficulty in integrating and tailoring the methodology to match project size and context as PRINCE2 methodology is too generic.	Tailoring and integrating issue (generic issue) (W6)
Difficulty aligning project goals with business objectives (business processes).	Aligning issue (S1)
No guidance on how to perform activities.	"How to" issue (W3)

Below the shortcoming/weakness categories are explained. Under each category is indicated whether the weakness is not addressed or inadequately addressed in the PRINCE2 methodology.

S1 – Shortcoming category: Aligning issue (IT gap – as identified in Chapter 5) Addressed in PRINCE2 – Not addressed

PRINCE2 only mentions that project goals should be aligned with business requirements through its business case (Office of Government Commerce, 2009:22). In PRINCE2 the business case theme entails evaluating whether the project is and remains viable in terms of estimated costs, estimated risks and expected benefits (Office of Government Commerce, 2009:22).

However, PRINCE2 does not provide a definition on what exactly is meant by the term alignment and the approach that top management should follow to align business processes with project goals.

The following factors are contributors to improper alignment of business processes with the project (end functionality of application software package), which are not addressed in PRINCE2 as the methodology is generic: application software package requirements not adequately identified, unclear and incorrect package requirements, ill-defined requirements, lack of understanding of package capabilities and difficulty in defining the inputs and outputs of the package. Ill-defined requirements may be due to lack of understanding of the organisation's business model and business processes by the client.

Furthermore, in many instances the client changes business processes to fit into the application software package which leads to improper alignment.

W1 – Weakness category: Capability/competence issue
Addressed in PRINCE2 – Inadequately addressed

PRINCE2 recommends that the project manager as well as the project team members should have the necessary competencies and be capable of performing the assigned roles and responsibilities (Office of Government Commerce, 2009). A few competencies are listed in PRINCE2, but no definition is provided on what is meant by capability or how to determine whether the project manager and project team have the necessary capabilities.

Contributors towards the capability/competence issue may include: lack of experience by the project managers and team members in the specific application software package and difficulty to build a balanced (detailed personalities and non-detailed personalities) composition team.

79 |

***W2 – Weakness category*: Communication issue**
Addressed in PRINCE2 – Inadequately addressed

PRINCE2 recommends the preparation of a communication management strategy (Office of Government Commerce, 2009:155). The communication management strategy entails the communication procedure to follow, tools and techniques that will be used, records that will be kept and timing of communication activities (e.g. meetings) (Office of Government Commerce, 2009:156).

However, what PRINCE2 neglects to address is that in many instances lower level management may be afraid or hesitant to report any issues to top level management. Not reporting issues could result in top management being unaware of the true status of the project (Keil & Robey, 2001:87).

Furthermore, fixed communication structures as recommended by PRINCE2 might be too rigid in some cases.

Lastly, in an IT environment, the client and supplier speak different languages. The PRINCE2 methodology does not provide guidance on what approach should be followed to ensure a mutual understanding between the client and supplier.

***W3 – Weakness category*: "How to" issue**
Addressed in PRINCE2 – Inadequately addressed

The PRINCE2 methodology states who shall conduct what activities and in which order the activities should be conducted, but neglects to give adequate guidance on how to perform the specific activities. Although PRINCE2 does list a few detailed techniques (Office of Government Commerce, 2009:7) it is too generic to be of any help when implementing application software package projects.

_W4 – Weakness category__: Planning issue_
Addressed in PRINCE2 – Inadequately addressed

PRINCE2 emphasises the importance of documentation specifically during the planning phase (Initiation of a project process) as well as throughout the project life cycle (Office of Government Commerce, 2009). However, the project manager and project team members should be careful that running the project by "PRINCE2" and completing documents do not become more important than focussing on achieving project goals (Office of Government Office, 2009:12).

Although the authors of PRINCE2 warn the user of the methodology of the above issue, no guidance is provided on how to ensure that the project does not fall in the documentation trap.

Even though PRINCE2 emphasises the importance of proper planning, the planning stage of the project in many instances is neglected. The reason for neglecting the planning stage may be due to improper understanding of the business case and especially the business processes of the organisation.

_W5 – Weakness category__: Soft ("people") issues_
Addressed in PRINCE2 – Reference made but not adequately addressed

Soft or "people" issues entail many factors: lack of user participation, users resistant to change, conflict between team members, team members with negative attitudes, high turnover of managers and/or team members, users not committed to the project and the project manager may lack adequate people skills.

The soft issues are specifically excluded from the PRINCE2 methodology manual (Office of Government Commerce, 2009:7).

Although the soft issue is very important in project management, the authors of PRINCE2 are of the opinion that it is impossible to codify it in a method (Office of Government Commerce, 2009:7). They recommend the user of PRINCE2 should study other leadership models and interpersonal skills training programmes to address the soft issue.

W6 – Weakness category: Tailoring and integration (generic) issue
Addressed in PRINCE2 - Inadequately addressed

PRINCE2 recommends that the methodology should be tailored and integrated with industry-specific or type-specific activities according to the specific project needs, because PRINCE2 is not a "one size fits all solution" (Office of Government Commerce, 2009:5). If the methodology is not tailored according to the requirements of the organisation it may lead to project failure (Office of Government Commerce, 2009:216).

PRINCE2 includes a chapter on tailoring PRINCE2 to the project environment. However the guidance on tailoring is very generic. Furthermore the guidance should be tailored extensively which might be expensive.

As PRINCE2 is generic a problem is created in that no resources exist on how to tailor and integrate PRINCE2 to exactly suit the needs of an application software package project.

W7 – Weakness category: Testing issue
Addressed in PRINCE2 – Inadequately addressed

PRINCE2 emphasises that each completed work package and the end product should be evaluated and reviewed (Office of Government Commerce, 2009). When reviewing the product for quality, PRINCE2 mentions one of two appraisal methods may be used: testing or quality inspection (Office of Government Commerce, 2009:54).

However, PRINCE2 does not emphasise (or recommend) the importance of testing by the end user. PRINCE2 only recommends that the reviewer should be independent of the producer of the end product.

W8 – Weakness category: Training issue
Addressed in PRINCE2 – Inadequately addressed

PRINCE2 recommends that the project manager should evaluate which team members should be trained (Office of Government Commerce 2009:40) and training should be built into the planning of the project. However, no reference is made to the training of the other stakeholders involved in the project (or project managers).

If the training of the end user is neglected the project might seem like a failure due to the end users not properly understanding how the application software package works. Insufficient training may further lead to end users having a resistance to change and not accepting the new application software package at the end of the project.

6.3   Conclusion

Although many weaknesses exist in the PRINCE2 methodology and the alignment of business processes with the package functionality are not addressed (IT gap), the methodology may still be used to assist with the implementation of application software packages.   When the PRINCE2 methodology is used in conjunction with the recommendations made in Chapter 7 of this assignment, proper alignment between the business processes with the functionality of the end product may be achieved.

CHAPTER 7

RECOMMENDATIONS

7.1   Recommendations for addressing weaknesses in PRINCE2

Recommendations for weaknesses in the PRINCE2 methodology contributing to the improper alignment of business processes with the functionality of the application software package will be discussed below.

W1 - Weakness: Capability/competence issue
Recommendations

- **Measure technical capabilities**: Capability may be defined as the measure of the ability of a person to achieve the set objectives (Business Dictionary, s.a.). Technical capabilities may be measured by the number of years of practical experience that the project manager and team member have of successful implementations of the application software package.

- **Measure project management capabilities**: Project management capabilities may be measured by the number of years of experience in successful project management appointments.

- **Measure soft ("people") skill capabilities**: Soft skill capabilities may be measured by conducting a personality assessment of the person to be appointed as project manager.

- **Train first time project managers**: In cases where a project manager is appointed as first time project manager, the person should be trained in project management and soft skills before being appointed.

- **Mentor first time project managers**: First time project managers should be mentored by experienced project managers with the necessary capabilities.

- **Continually asses team members' performance**: It is the responsibility of the project manager to continually assess team members' performance (capabilities and competence) and to be willing to oppose and reassign people with poor performance.

- **Competencies not mentioned in PRINCE2**: In addition to the competencies that PRINCE2 lists, good team player quality, confidence, enthusiasm, energy and initiative may be added to the list as required competencies.

W2 - Weakness: Communication issue

- **Adopt less rigid communication structures**: The project manager should not only depend on reporting structures set at the start of the project, but consult whenever it seems necessary.

- **Create a "bridging" language**: To create a "bridging" language, opportunities should be created for the supplier to work with or shadow business staff (client) and vice versa. Creating a "bridging" language would give the supplier and client staff the opportunity to become comfortable with each other's terminology, methodology, frustrations and needs as well as create an understanding of each other's environments. Furthermore, creating a "bridging" language will assist both the client and supplier to perform an adequate business case.

- **Appoint staff with IT and business knowledge**: Depending on the size of the business, appoint a person with an IT and business background to facilitate communication between the supplier and client.

- **Encourage timely reporting of issues**: To address the issue of team members being hesitant to report issues, the project manager should reassure the project team at the start of the project that no repercussions will be encountered by a team member if the issue is reported timely. However, if the issue is not reported timely there will be repercussions.

- **Management should be tolerant in certain circumstances**: Top management and the project manager should be tolerant when a good reason exists for poor performance.

W3 - Weakness: "How to" issue

- **Tailor the methodology to business environment**: The "How to" and tailoring of the methodology issue go hand in hand. The selection of a supporting framework/methodology to implement an application software package would not address the IT gap. How the methodology is made applicable when implementing the application software, taking into consideration the information needs (and business processes) of the company, will address the IT gap.

  The "How to" issue should be addressed during the planning stage of the project. When the supplier decides that a specific course of action should be taken, the detailed techniques on how the action should be executed must be documented at the start of the project by a person with the necessary experience.

- **Employ staff with the necessary past experience**: Project managers (and team members) with past successful implementations of the specific application software package should be included in the team as they can be seen as the best "How to" guide. They may only fulfil a mentoring role if necessary.

W4 - Weakness: Planning issue

- **Measuring project success**: Top management should ensure that the measures for successful implementation of the application software project are not limited to meeting time and budget only. If the whole project is driven by time and cost only it will fail to meet the business imperatives (information needs and end functionality of application software package).

- **Focus on project goals instead of documentation only**: The supplier (project manager and project team) should be careful that the completion of documents

does not become more important than focussing and achieving project goals. The project manager as well as the team members should rather apply their minds and consider any other activities that may be relevant to contribute to the success of the project, rather than following the methodology blind folded.

W5 - Weakness: Soft ("people") issues

- **Evaluate project manager's soft skills**: An important issue for the supplier to address is to ensure that the project manager has sufficient people skills. The supplier may for example have discussions with team members on prior projects where the proposed project manager acted as project manager.

  If the project manager does not have sufficient soft skills he or she should attend a course on basic soft skills.

- **Educate staff members on soft skills**: It is also advisable prior to the start of the project for all team members to have a "crash course" in soft skills, specifically on how to resolve conflict between team members.

- **Introduce application software package early to address certain soft skills issues**: To address the issue of users resistant to change and lack of user participation, top management should introduce the new application software package from the initiation of the project. Top management should emphasise to all users that each one of them must and can make a worthy contribution to the successful implementation of the application software package.

  To address the soft issue of team members not committed to the project, the project manager should ensure that each team member understands what his job entails in writing. Furthermore, the project manager must document what the repercussions are should the responsibilities not be performed adequately.

- **Enhance team building exercises or social activities**: As building a team from a collection of individuals from different backgrounds is not an easy job, opportunities should be provided for socialising and interaction between the supplier (team members) and client, like team building exercises outside the office prior to the start of the project. Team building exercises may enhance the employees' functioning as a team.

- **Extra incentives for hard work**: To address the issue of negative attitudes which are usually caused by working long hours, the project team may receive additional incentives, in the form of leave or payment for overtime, for the extra efforts put into the project.

W6 - Weakness: Tailoring and integration (generic) issue

- For recommendations refer to the section on "How to" issue as well as section 7.2.

W7 - Weakness: Testing issue

- **Testing by the end user**: Detailed and thorough testing should be conducted at the end of each process as well as at the end of building the requirements of the application software package. Thorough end user testing should be performed before implementation.

  Testing by the end user will ensure adequate functionality of the application software package and user acceptance. Testing by the end user will further ensure that the application software package works technically correct and the business process configurations are practical.

W8 - Weakness: Training issue

- **Train project managers**: First time project managers should be trained before they are appointed.

- **Train project team members**: The project manager should evaluate whether any team members require training. Evaluation may be based on whether the specific team member has past practical experience of the package that needs to be implemented or whether the team member only attended a course in the past.

- **Train the end user**: If the end user does not know how to use the new application software package, training should start early; preferably well before the start of the implementation. If training starts early it will assist employees in testing the system at the end of each process and make them ready for the change (address the issue of resistance to change) to the new application software package.

  Training to the end user (and project team if needed) should be continuous throughout the project.

- **Implement on the job coaching**: On the job coaching, where team members coach one another, especially when taking over tasks from another team member, is a good way to give the necessary (or additional) training.

7.2 Recommendations for addressing the alignment (IT gap) shortcoming

Below, the IT gap issue (discussed in Chapter 3) that is not addressed by the PRINCE2 methodology (discussed in Chapter 5 and 6), will be summarised and recommendations will be made on how to reduce the IT gap that exist between the client's business processes and the supplier of application software package end functionality.

S1 – Shortcoming: Alignment issue

***IT gap component**: Business model (I1 & I2)*

Issue summarised

Top management expects from IT executives to implement an application software package that supports the business model. The adequate analysis of the business case is neglected in many instances (Paul, as cited by Smith, 2002:52) by the supplier.

In contrast it might be difficult for the client to communicate clear goals of the organisation's information requirements and business processes to the supplier (Aken, 2008:317).

Recommendations

- **Distinguish between business objectives and business imperatives**: Top management should distinguish between business objectives (essential things that need to be performed for a business to survive for example to make profit) and business imperatives (thrust of activities - things that are absolutely crucial and that need to be performed exceptionally well with regards to IT for a business to succeed in a specific industry, for example information requirements of application software package, an affordable and low risk application software package) (Boshoff, 2010).

- **Input from all stakeholders**: Although the business imperatives (business processes) should be driven from top management, all other stakeholders (board, IT and end users of the application software package) must be asked to give their input. The involvement of all parties is necessary to ensure that all stakeholders commit to the objectives of the project (Boshoff, 2011).

  End users may make valuable contributions because they are the persons who will work with the new application software package on a daily basis.

- **Create a "bridging" language**: A "bridging" language should be created by appointing a person with both IT and business background to facilitate

communication between the supplier and client. The "bridging" language would also assist in the conceptualisation issue that the supplier may experience.

IT gap component: Business processes (I3 & I4)

<u>Issue summarised</u>

Business processes are abstract to the supplier and difficult to conceptualise.

Top management may change business processes to suit the application software package (Turbit, 2005:4) which may result in an end product that has inadequate functionality.

<u>Recommendations</u>

- **Involve key people**: It is crucial that key people who have an in-depth understanding of the specific information requirements and business processes (and why processes happen as they do) are involved in the evaluation of business processes.

- **Documentation and evaluation of current business processes**: Top management should evaluate, define and document the current business processes with the vision of how they can improve the efficiency of the organisation's business processes (Boshoff, 2011).

  The business processes should be documented and defined clearly at the beginning of the project in order for the supplier to understand how users would use the application software package. Proper documentation will enable the supplier to perform their own business case analysis. The documentation of both the client and supplier should be reviewed prior to the start of the project and if necessary be explained to each other to ensure mutual understanding of the business imperatives between both parties.

  If top management is unsure how to analyse current business processes, external consultants (other than the proposed supplier responsible for implementation of

the application software package) may be employed temporarily to assist in the process. For example, an external auditor may be temporarily employed to assist top management in evaluating the current business processes.

- **Obtain proposals from multiple suppliers**: Before deciding on a supplier to implement the application software package, top management should consider obtaining proposals from more than one supplier selling different application software packages. Obtaining proposals from more than one supplier may assist top management in selecting the best way to deliver the end product through scenario comparison. Top management should not only consider cost but also the functionality and adaptability of the application software package.

- **Evaluate reputation of suppliers**: Top management should evaluate the reputation of the supplier before a final decision is made on which supplier should be appointed.

  Top management may request a list of successful implementations from the supplier which may be contacted by the organisation in order to evaluate to what extent the other organisations were satisfied with the services received from the specific supplier. Top management may inquire whether the supplier was only selling their product or actually trying to assist in improving the organisation's business processes.

- **Evaluate and compare supplier proposals**: Top management should contact the proposed suppliers and request proposals from the suppliers mapping to what extent the supplier's product (application software package) will fit the improved efficiency of the organisation's business processes.

  If the organisation does not have an IT department, top management should consider temporarily employing an IT consultant (other than the supplier) to evaluate the mapping. The client together with the assistance of the IT department or external consultant may then measure the application software

package in terms of where the organisation's business processes are now and where they want to be at the end of implementation.

- **Evaluate level of application software package customisations required**: It is the responsibility of the client to ask the supplier to what extent the application software package needs to be customised. If a lot of customisation is required the client should consider another package.

IT gap component: Functionality of package (I5 & I6)

Issue summarised

Physical information requirements and business processes would be translated by the supplier into digital requirements by using customisation tools like parameters/scripts and package changes (Boshoff, 2011).

If the business process analysis were not performed properly at the beginning of the project the intended functionality may not be the implemented functionality.

Recommendations

- **Proper understanding of business processes by supplier**: Only after the supplier has conducted and documented the business case analysis (including determining whether the business processes are data or functional rich) and properly understands how the users will use the application software package, can the supplier start translating the physical information needs (and business processes) into digital form (Boshoff, 2011).

- **Testing of functionality at end of each stage**: After the completion of each stage of the implementation of the application software package, the end users of the application software package should test the specific stage before proceeding to the next implementation stage. Testing each stage will identify any misunderstandings encountered at the beginning of the project during the analysis of the business case.

- **Limit customisations**: The supplier may only consider customisation when the functionality of the application software package fails to address critical business requirements (Boshoff, 2011). If the supplier only informs the client after the project has started that a lot of customisation is required, the client should evaluate whether it might be more beneficial to rather end the project and select another application software package than proceeding with the current project.

   The supplier should limit package changes to a minimum as customisations may create a retrofit (Boshoff, 2011) issue when new updates need to be installed. Retrofit may be defined as when upgrading the package, changes first need to be reversed before the package may be upgraded (Boshoff, 2011).

   However, if customisations are necessary the supplier should document exactly what changes were made in order for the supplier responsible for upgrades in future to know exactly what to retrofit.

IT gap component: _Data attributes (I7 & I8)_

Issue summarised

If the supplier is not accustomed to the technical components of the specific application software package, it may lead to incorrectly building the requirements of the application software package and not suiting the information needs and business processes of the organisation (Boshoff, 2011).

Recommendations

- **Define technical components**: The supplier should ensure that each technical component of the information requirements is properly defined. For example, in an accounting environment orders and invoicing should each be defined in terms of tables and fields (Boshoff, 2011).

- **Adequate knowledge of technical components required**: The supplier should ensure that the whole project team is accustomed to the technical components of the application software package that needs to be implemented. If all team members are not familiar to the technical components, the project manager should make arrangements for the necessary training of the team members prior to the start of the project.

- **Mentor first time team members**: If it is the first time a specific team member of the supplier is responsible for building the requirements of the application software package, it is the responsibility of the supplier to ensure that the team member is assisted or mentored by another team member that has the necessary experience and skills in implementing the specific application software package.

7.3   Conclusion

To achieve alignment between the business processes of the organisation with functionality of the end product (application software package), top management of an organisation should ensure that as far as reasonably possible all of the above recommendations are adopted.

When above recommendations are used together with the PRINCE2 methodology, the IT gap between business processes and the functionality of the application software package may be reduced. The end result will be less project failures and more successful application software package implementations.

CHAPTER 8

CONCLUSION AND SUMMARY

Application software package failure issues were discussed from as early as 1975 and many authors have performed extended research over the past 30 years. Despite this, many IT application software package projects still fail.

Many authors are of the opinion that the most significant reason why IT projects fail is that business processes are not aligned with the end functionality of the application software package. Misalignment between business processes and the end functionality of the application software package creates a gap between IT (application software package) and business (business processes), commonly referred to as the IT gap.

Although a number of supporting frameworks are available to assist in the implementation of application software packages, organisations who use them do not always report a successful project. Many of the supporting frameworks are generic in nature and need to be tailored appropriately to suit the individual needs of the organisation.

The purpose of this assignment was to examine and discuss to what extent the PRINCE2 methodology may assist organisations in achieving proper alignment between business processes and the functionality of the application software package. Further the assignment proposed recommendations for aligning business process with the functionality of the application software package as well as recommendations for addressing weaknesses identified in the PRINCE2 methodology.

In conclusion, this assignment found that the PRINCE2 methodology does not address the alignment of business processes with the functionality of the application software package implemented. Furthermore, other weaknesses exist in the PRINCE2 methodology contributing to misalignment. Organisations first need to understand their business processes and evaluate how the business processes may be improved to be

more efficient. Only then should the organisation decide on an application software package that would best suit the organisation's business processes.

The recommendations proposed to bridge the IT gap and address weaknesses in the PRINCE2 methodology in this assignment are practical, simple and easily adaptable to any organisation. When using PRINCE2 methodology together with the recommendations made in this assignment, proper alignment between the organisations business processes and the functionality of the application software package may be achieved. The end result will be more successful application software package project implementations.

LIST OF REFERENCES

Agarwal, N. & Rathod, U. 2006. Defining 'success' for software projects: An exploratory revelation. *International Journal of Project Management,* 24(2006): 358-370.

Ahmad, A.A. & Newman, M. 2009. The transfer of technology and knowledge: A story of an enterprise system implementation, in *17th European Conference on Information Systems.* Manchester: Manchester Business School: 1-13.

Aken, A. 2008. CHUNK: An Agile Approach to the Software Development Life Cycle. *Journal of Internet Commerce,* 7(3): 313-338.

Al-Mashari, M., Al-Mudimigh, A. & Zairi, M. 2003. Enterprise resource planning: A taxonomy of critical factors. *European Journal of Operational Research*, 146(2003): 352-364.

Al Neimat, T. 2005. *The Project Perfect White Paper Collection. Why IT Projects Fail* [Online]. Available: http://149.144.20.200/subjects/ISD/isds111web/WhyDoITProjectsFail.pdf [31 May 2011].

Aloini, D., Dulmin, R. & Mininno, V. 2007. Risk management in ERP project introduction: Review of the literature. *Information and Management*, 44(2007): 547-567.

Bartis, E. & Mitev, N. 2008. A multiple narrative approach to information systems failure: a successful system that failed. *European Journal of Information Systems,* 17(2008): 112-124.

Bennatan, E.M. 2009. *Advanced Project Solutions Inc. Project Failures: Ignoring the Warning Signs* [Online]. Available: http://www.compaid.com/caiinternet/ezine/ProjFailures_EMB.pdf [3 June 2011].

Boshoff, W.H. 2010. Technology Gap - the bridge between assurance, governance and implementation/operations computer architectures. Unpublished class notes (Computer Auditing 871). Stellenbosch: Stellenbosch University.

Boshoff, W.H. 2011. Packages 1 – Navision & Project Management – Microsoft. Unpublished class notes (Computer Auditing 872). Stellenbosch: Stellenbosch University.

Business Dictionary. s.a. [Online]. Available: http://www.businessdictionary.com/definition/capability.html [28 June 2011].

Cerpa, N. & Verner, J.M. 2009. Why Did Your Project Fail*? Communications of the ACM,* December: 52(12): 130-134.

Chen, C.C., Law, C.C.H. & Yang, S.C. 2009. Managing ERP Implementation Failure: A Project Management Prospective. *IEEE Transactions on Engineering Management,* 56(1): 157-170.

Chin, P. 2003. *Why IT projects Fail* [Online]. Available: http://itmanagement.earthweb.com/print.php/2201981 [31 May 2011].

Coley Consulting. 2005. *Why IT projects fail* [Online]. Available: http://www.coleyconsulting.co.uk/failure.htm [31 May 2011].

Collaborative Computing. s.a. [Online]. Available: www.adaptivesoftware.biz/archive/.../Collaborative%20Computing.doc [20 June 2011].

Computerworld. s.a. *Top 10 Corporate Information Technology Failures* [Online]. Available:
http://www.computerworld.com/computerworld/records/images/pdf/44NfailChart.pdf
[9 June 2011].

99 |

Creating Value in Project Management Using PRINCE2. 2010. [Online]. Available: http://www.prince2.com/downloads/CreatingValueInProjectManagementUsingPrince2.pdf [15 March 2011].

Dalcher, D. 2009. Software Project Success: Moving Beyond Failure. *The European Journal for the Informatics Professional,* x(5): 43-50.

Demir, K.A. 2009. A Survey on Challenges of Software Project Management, in *2009 International Conference on Software.* USA: Naval Postgraduate School, Monterey, CA.

Deng, J. & Bian, Y. 2008. Constructing a Risk Management Mechanism Model for ERP Project Implementation, in *2008 International Conference on Information Management, Innovation Management and Industrial Engineering.* China: Business School, Hohai University: 72-77.

Dolan, K. 2010. *Addressing Project Failure through PRINCE2* [Online]. Available: http://www.prince2.com/downloads/Addressing_Project_Failure_through_PRINCE2.pdf [13 March 2011].

Ehie, I.C. & Madsen, M. 2005. Identifying critical issues in enterprise resource planning (ERP) implementation. *Computers in Industry,* 56(2005): 545-557.

Eveleens, J.L. & Verhoef, C. 2010. *The Rise and Fall of the Chaos Report Figures* [Online]. Available: http://www.few.vu.nl/~x/chaos/chaos.pdf [11 May 2011].

Gargeya, V.B. & Brady, C. 2005. Success and failure factors of adopting SAP in ERP system implementation. *Business Process Management Journal*, 11(5): 501-516.

Holt, M. 2003. *Butler Group Review Journal Article. Why do so many IT projects fail?* [Online]. Available: http://www.brett-tech.com/Downloads/itprojectfailures.pdf [31 May 2011].

Hvitved, T. 2009. *Architectural Analysis of Microsoft Dynamics NAV* [Online].Available: http://3gerp.iwvi.uni-koblenz.de/docs/hvitved09nav.pdf [21 June 2011].

INTOSAI EDP Audit Committee. s.a. *Best Practice, Why IT Projects Fail* [Online]. Available: http://www.intosaiitaudit.org/whyitprojectsfail.pdf [31 May 2011].

IT Cortex. 2005. *Success Assessment* [Online]. Available: www.it-cortex.*com/Failure.*-htm [31 May 2011].

IT Governance Institute (ITGI). 2008. *Enterprise Value: Governance of IT investments, Getting Started with Value Management* [Online]. Available: http://www.isaca.org/Knowledge-Center/Val-IT-IT-Value-Delivery-/Documents/Val-IT-Getting-Started-Jul-2008.pdf [21 March 2011].

Jackson, P. & Klobas, J. 2008. Building knowledge in projects: A practical application of social constructivism to information systems development. *International Journal of Project Management*, 26(2008): 329-337.

Jurison, J. 1999. Software Project Management: The Manager's view. *Communications of the Association for Information Systems,* 2(17): 1-56.

Kappelman, L.A., McKeeman, R. & Zhang, L. 2006. Early warning signs of IT project failure: The dominant dozen. *Information systems management,* Fall 2006: 31-36.

Ke, W. & Wei, K.K. 2008. Organizational culture and leadership in ERP implementation. *Decision Support Systems,* 45(2008): 208-218.

Keil, M. & Robey, D. 2001. Blowing the whistle on troubled software projects. *Communications of the ACM*, 44(4): 87-93.

Keil, M., Im, G.P. & Mahring, M. 2007. Reporting bad news on software projects: the effects of culturally constituted views of face-saving. *Info System Journal,* 17: 59-87.

Kim, Y., Lee, Z. & Gosain, S. 2005. Impediments to successful ERP implementation process. *Business Process Management Journal,* 11(2): 158-170.

May, L.J. 1998. *Major Causes of Software Project Failures* [Online]. Available: http://paul-hadrien.info/backup/LSE/IS%20470/litterature%20review/MajorCausesof SoftwareProjectFailures.pdf [30 May 2011].

McManus, J. & Wood-Harper, T. 2007. Understanding the Sources of Information System Project Failure. *Management Services,* Autumn 2007: 38-43.

Office of Government Commerce. s.a. *Common Causes of Project Failure (OGC Best Practice)* [Online]. Available: http://www.prince2.com/downloads/why-projects-fail.pdf [5 May 2011].

Office of Government Commerce. 2009. *Managing Successful Projects with PRINCE2.* UK: The Stationary Office. [Online]. Available: http://www.tsoshop.co.uk [11 May 2011].

Park, C.W. & Keil, M. 2009. The Effect of IT Failure Impact and Personal Morality on IT Project Reporting Behaviour. *IEEE Transactions on Engineering Management,* 56(1): 45-60.

Plotnikova, S. 2007. Applying PRINCE2 Project Management Disciplines to Address Key Risks in ERP System Implementation Projects. Unpublished mini study project. Stellenbosch: University of Stellenbosch.

Poli, M. & Shenhar, A.J. 2003. Project Strategy: The Key to Project Success. *Management of Engineering and Technology,* 20 – 24 July 2003: 231-235.

Procaccino, J.D & Verner, J.M. 2006. Software project managers and project success: An exploratory study. *The Journal of Systems and Software,* 79(2006): 1541-1551.

Sauer, C. & Cuthbertson, C. 2003. *The State of IT Project Management in the UK 2002-2003* [Online]. Available: http://www.bestpracticehelp.com/The_State_of_IT_Project_Management_in_the_UK_2 003_2004.pdf [31 May 2011].

Smith, J. 2002. Why Information Technology Software Projects Fail in South Africa. Unpublished mini study project. Johannesburg: Technikon Witwatersrand.

Sofian, A. 2003. *Project Success in Relation with Organizational Roles and Capabilities and Project Managers' Skills and Capabilities* [Online]. Available: http://www.pmi.org/Knowledge-Center/~/media/PDF/Surveys/pp_sofian.ashx [31 May 2011].

Stapelberg, E.P. 1994. The Implementation of Customisable Software Packages for Business Practice. Unpublished doctoral dissertation. Pretoria: University of Pretoria.

Taylor, A. 2000. *The Computer Bulletin January 2000. IT projects: sink or swim* [Online]. Available: http://itnow.oxfordjournals.org/content/42/1/24.full.pdf+html [30 May 2011].

Thomas, G. & Fernandez, W. 2008. Success in IT projects: A matter of definition? *International Journal of Project Management,* 26(2008): 733-742.

Tillmann, G. & Weinberger, J. 2004. *Baseline. Technology never fails, but projects can* [Online]. Available: http://www.baselinemag.com/index2.php?option=content&task=view&id=488&pop=1&hide_ads=1&page=0&hide_js=1 [30 May 2011].

Turbit, N. 2005. *The Project Perfect White Paper Collection. ERP Implementation – The Traps* [Online]. Available: http://www.projectperfect.com.au/downloads/Info/info_erp_imp.pdf [31 May 2011].

Umble, E.J., Haft, R.R. & Umble, M.M. 2003. Enterprise resource planning: Implementation procedures and critical success factors. *European Journal of Operational Research,* 146(2003): 241-257.

Upadhyay, P., Jahanyan, S. & Dan, P.K. 2011. Factors influencing ERP implementation in Indian manufacturing organisations. *Journal of Enterprise Information Management,* 24(2): 130-145.

Velcu, O. 2010. Strategic alignment of ERP implementation stages: An empirical investigation. *Information & Management,* 47: 158-166.

Weston, F.C. 2001. ERP implementation and project management. *Production and Inventory Management Journal*, 42(3/4): 75-80.

Wikipedia. 2011a. [Online]. Available: http://en.wikipedia.org/wiki/Application_software [9 June 2011].

Wikipedia. 2011b. [Online]. Available: http://en.wikipedia.org/wiki/Business_case [29 June 2011].

Wikipedia. 2011c. [Online]. Available: http://en.wikipedia.org/wiki/Business_model [20 June 2011].

Wikipedia. 2011d. [Online]. Available: http://en.wikipedia.org/wiki/Business_process [20 June 2011].

Wikipedia. 2011e. [Online]. Available: http://en.wikipedia.org/wiki/Functional_ specification [21 June 2011].

Wikipedia. 2011f. [Online]. Available: http://en.wikipedia.org/wiki/System_software [9 June 2011].

Wikipedia. 2011g. [Online]. Available: http://en.wikipedia.org/wiki/A_Guide_to_the_Project_Management_Body_of_Knowledge [29 July 2011].

Winter, M. 2006. Investigation into project management failure within information technology systems projects. Unpublished mini study project. Stellenbosch: Graduate School of Business of the University of Stellenbosch.

Zand, D.E. & Sorensen, R.E. 1975. Theory of Change and the Effective Use of Management Science. *Administrative Science Quarterly*, 20(4): 532-545.