

**BUSINESS PROCESS MANAGEMENT IN AN INTRAPRENEURIAL SOFTWARE
ORGANISATION**

by

Ulrike Janke

**In partial fulfilment of the requirements for the degree
Master's in Business Administration at the North-West University.**

Supervisor: Prof JG Kotzé

2006

Potchefstroom Campus

“People are more important than any process.”

(Booch, 1996:188)

“The four ‘P’s of project management are: People Performing Perfect Process.”

(Johnson, 1999)

“While software as art has a certain aesthetic and emotional appeal, the fact is artistic endeavour has never been a reliable process.”

(Davis & Leffingwell, 1991:1)

ACKNOWLEDGEMENTS

- Chantelle and Franni, for putting up with me and my moods, and for your unconditional love. This is dedicated to you.
- My family, for your support of everything I set out to do.
- My faith and strong coffee, which got me out of bed every morning.
- My colleagues at CText, in particular Gerhard, who made time for me as well, between all your own post-graduate students and articles. You have no idea how much I appreciate it.
- Nico, Alon and Alex: You spoke with such passion about your work, it was an honour spending time with you!
- Jeff, Robin, Victor, Mari and Christo: Thank you for your interest, time and expert inputs. I feel privileged to have met and corresponded with you.
- Prof. Jan Kotzé.
- *The Bottom Line* diehards: Frans and Albert, you were awesome team mates!

Thank you all,
Ulrike

ABSTRACT

Business process management in an intrapreneurial software organisation

By Ulrike Janke

Business process management (BPM) is a philosophical approach to organisation-wide management in which the focus is on the processes through which it operates, and in particular the streamlining and optimising of these processes, for which software solutions may be used. CText is an intrapreneurial software organisation that has been experiencing problems with software development due to a lack of formal processes relating to customer support, versioning, configuration, quality, risk and project management.

The objective of the study is to determine whether the implementation of an electronic BPM system can effectively solve CText's development problems and thereby improve its overall software development capacity. More specifically, the focus is on i) the effect of the resulting standardisation on creativity and innovation, and ii) implementation matters, such as the type of processes that can be subjected to an electronic system, and how CText can overcome the time and cost constraints of such a system.

The study investigates these questions by means of a literature investigation in combination with interviews with knowledgeable respondents from other innovative and software organisations. Interviews with six employees from CText determine the relevance of these findings and highlight critical areas for process improvement.

Since BPM systems improve organisational efficiencies and are generally employed in larger corporate contexts marked by transactional and repetitive activities where they enforce administrative rules, the conclusion is drawn that a BPM system will not be suitable for an intrapreneurial organisation, and that it is likely to cause more disruption to the creative environment than improve its operations. It is further shown that although a BPM system is theoretically applicable to software development, it generally does not seem to be applied practically in the industry, and the suitability of this process as manageable through a BPM system is seriously questioned.

Instead, the research points to improvement through the application of software development methodologies and a holistic approach towards BPM. The investigation at CText

confirms that its development problems relate to flawed methodologies and that remedies should therefore focus on improving its methodologies and controlling certain aspects of the software development life cycle by means of suitable software tools.

KEY TERMS

Business Process Management, BPM, BPM philosophy, BPMS, CText, Development, Intrapreneurial, Research and Development, Software, Software development, Software development methodologies, WfMS, Workflow, Workflow solutions

OPSOMMING

Besigheidsprosesbestuur in 'n intrapreneuriese programmatuurontwikkelingsorganisasie

Deur Ulrike Janke

Besigheidsprosesbestuur (BPB) is 'n filosofiese benadering tot organisasiewye bestuur waarin gefokus word op die prosesse waardeur die werk plaasvind, en in besonder op die stroomlyning en optimering van hierdie prosesse, waarvoor programmatuuro oplossings aangewend mag word. CText is 'n intrapreneuriese programmatuurontwikkelaar wat probleme met programmatuurontwikkeling ervaar weens 'n tekort aan formele prosesse wat verband hou met kliënteondersteuning-, weergawebeheer-, konfigurasie-, kwaliteit-, risiko- en projekbestuur.

Die doelwit van hierdie studie is om te bepaal of die implementering van 'n elektroniese BPB-stelsel CText se ontwikkelingsprobleme effektief kan oplos en daardeur sy algemene ontwikkelingskapasiteit kan verbeter. Meer spesifiek is die fokus op i) die effek van die voortspruitende standaardisering op kreatiwiteit en innovasie en ii) implementeringsake, soos die tipe prosesse wat aan 'n elektroniese stelsel onderwerp kan word, en op welke wyse CText die tyd- en kostebeporinge van so 'n stelsel kan oorbrug.

Hierdie vrae word ondersoek deur 'n kombinasie van 'n literatuurstudie en onderhoude met kundiges van ander innoverende en programmatuurontwikkelingsorganisasies. Onderhoude met ses werknemers van CText bepaal die toepaslikheid van hierdie bevindinge, en lig die kritiese areas vir prosesverbetering uit.

BPB-stelsels verhoog organisatoriese effektiwiteit en word oor die algemeen gebruik in groter korporatiewe liggings wat gekenmerk word deur transaksionele en roetienewerk waar hulle administratiewe reëls afdwing. Die gevolgtrekking is dat 'n BPB-stelsel nie geskik sal wees vir 'n intrapreneuriese organisasie nie, en dat dit waarskynlik meer ontwinging vir die kreatiewe omgewing as verbetering van die werksaamhede teweeg sal bring. Daar word verder aangetoon dat hoewel 'n BPB-stelsel teoreties toepaslik behoort te wees op programmatuurontwikkeling, dit nie in die praktyk die geval blyk te wees nie, en die bestuurbaarheid van hierdie proses deur BPB-stelsels word ernstig bevraagteken.

Die navorsing wys op verbetering deur die toepassing van geskikte programmatuurontwikkelingsmetodologieë en 'n holistiese benadering tot BPB. Die ondersoek by CText bevestig dat die ontwikkelingsprobleme verband hou met foutiewe metodologieë en dat regstellende stappe behoort te fokus op die verbetering van metodologieë en die beheer van sekere aspekte in die programmatuursontwikkelingsiklus deur middel van geskikte programmatuurhulpmiddele.

SLEUTELTERME

Besigheidsprosesbestuur, BPB, BPB-filosofie, BPBS, CText, Ontwikkeling, Intrapreneurries, Navorsing en Ontwikkeling, Programmatuur, Programmatuurontwikkeling, Programmatuurontwikkelingsmetodologieë, Werkvloei, Werkvloeibestuurstelsels, Werkvloei-oplossings

ABBREVIATIONS

API: Application Programming Interface
BPA: Business process analysis
BPEL: Business Process Execution Language
BPM: Business Process Management
BPMS: Business Process Management System/Suite(s)
BRE: Business Rule Engines
CALL: Computer Assisted Language Learning
CASE: Computer-Aided Software Engineering
CMM: Capability Maturity Model
CText: Centre for Text Technology, North-West University
DSDM: Dynamic Systems Development Method
EAI: Enterprise Application Integration
ECM: Enterprise Content Management
ERP: Enterprise Resource Planning
HLT: Human Language Technology
PRINCE: Projects In Controlled Environments
RAD: Rapid Application Development
RUP: Rational Unified Processes
SCM: Software Configuration Management
SD: Software Development
SDLC: Software/Systems Development Life Cycle
UBL: Universal Business Language
UML: Universal Modeling Language
WfMC: Workflow Management Coalition
WfMS: Workflow Management System(s)
XP: Extreme Programming
XPDL: XML Process Definition Language

TABLE OF CONTENTS

ACKNOWLEDGEMENTS.....I

ABSTRACTII

OPSOMMINGIV

TABLE OF FIGURESX

1 NATURE AND EXTENT OF THE STUDY 1

1.1 INTRODUCTION AND CONTEXTUALISATION..... 1

1.2 PROBLEM STATEMENT..... 3

1.3 SPECIFIC RESEARCH QUESTIONS 5

1.4 OBJECTIVES..... 5

1.5 RESEARCH METHODOLOGY 6

1.6 SCOPE AND DEMARCATION OF THE STUDY..... 7

1.7 DEPLOYMENT OF THE STUDY..... 8

2 TERMINOLOGIES 10

2.1 BPM AND WORKFLOW: DIFFERENT TERMS FOR THE SAME CONCEPTS? 10

2.1.1 Enterprise Content Management 11

2.1.2 Workflow..... 12

2.1.3 Business Process Management..... 13

2.1.4 Working definitions for workflow and BPM 14

2.2 A TECHNICAL PERSPECTIVE 15

2.2.1 BPMS technology overview 16

2.2.2 The workflow engine..... 18

2.2.3 Standards and specifications 18

2.3 IMPLEMENTATION AREAS 19

2.4 SUMMARY..... 21

**3 BPM AND INNOVATION IN AN INTRAPRENEURIAL ORGANISATION:
CHALLENGES AND SOLUTIONS 22**

3.1 INTRODUCTION 22

3.2 DEFINING CONCEPTS..... 22

3.2.1	Intrapreneurship	23
3.2.2	Creativity	23
3.2.3	Innovation.....	23
3.3	GENERAL ORGANISATIONAL CONSTRAINTS ON INTRAPRENEURSHIP.....	24
3.3.1	Administrative barriers	24
3.3.2	Project management.....	25
3.3.3	Group dynamics	25
3.4	CHALLENGES RELATING TO BPM AND INTRAPRENEURSHIP	25
3.4.1	Control.....	25
3.4.2	Toleration of failure.....	26
3.4.3	ERP challenges.....	26
3.4.4	Practical challenges to small organisations.....	27
3.4.4.1	Limited choice.....	27
3.4.4.2	Cost and complexity	27
3.5	SOLUTIONS	28
3.5.1	Managing innovation.....	28
3.5.1.1	Strategic and operational management support.....	28
3.5.1.2	Project management.....	29
3.5.1.3	A change culture.....	29
3.5.1.4	Learning networks	30
3.5.1.5	Socialisation	30
3.5.2	BPM solutions for intrapreneurs.....	30
3.5.2.1	Incremental BPM deployment.....	31
3.5.2.2	Implement the BPM philosophy	31
3.5.2.3	Subscribing to services.....	32
3.6	BUSINESS PROCESSES AND INNOVATION AT CTEXT	32
3.6.1	Introduction.....	32
3.6.2	Survey results.....	34
3.6.2.1	Administrative barriers	34
3.6.2.2	Project management.....	38
3.6.2.3	Group dynamics	38
3.6.2.4	Control.....	39
3.6.2.5	Errors	40
3.7	SUMMARY AND CONCLUSION	41
4	BPM AND SOFTWARE DEVELOPMENT: CHALLENGES AND SOLUTIONS...	44
4.1	INTRODUCTION.....	44
4.2	SOFTWARE DEVELOPMENT CHALLENGES.....	45
4.2.1	Process flaws	45
4.2.2	Confusion over the role of project management.....	45
4.2.3	Over-reliance on software tools	46

4.3	THE CASE FOR PROCESS MANAGEMENT IN SOFTWARE DEVELOPMENT	46
4.3.1	Software development processes	47
4.3.1.1	The Software/System Development Life Cycle	47
4.3.1.2	Rational Unified Processes	48
4.3.1.3	Project management processes	50
4.3.1.4	Step Wise	51
4.3.1.5	PRINCE2 TM	52
4.4	PITFALLS OF BPM IN SOFTWARE DEVELOPMENT	53
4.4.1	Managing people	53
4.4.2	Disagreement over BPM in software development	54
4.5	SOFTWARE DEVELOPMENT SOLUTIONS	55
4.5.1	BPM solutions for software development	55
4.5.2	Methodologies	56
4.5.2.1	Agile development methodologies	56
4.5.2.2	PRINCE2	57
4.5.2.3	The Rational Unified Process	58
4.5.2.4	A Methodology audit and the Capability Maturity Model	58
4.5.2.5	CASE tools	60
4.6	BPM AND SOFTWARE DEVELOPMENT AT CTEXT	61
4.6.1	Introduction	61
4.6.2	Survey results	61
4.6.2.1	Specifications and requirements	61
4.6.2.2	Risk management	62
4.6.2.3	Testing	62
4.6.2.4	Methodologies	63
4.6.2.5	CMM	63
4.6.2.6	Software configuration management	64
4.6.2.7	Programming conventions	64
4.6.2.8	People	64
4.6.2.9	Project management	65
4.7	SUMMARY AND CONCLUSION	66
5	CONCLUSION AND RECOMMENDATION	68
5.1	SUMMARY	68
5.2	RECOMMENDATION	72
5.3	FUTURE RESEARCH	76
	REFERENCES	77
	APPENDIX A: POLARITY MANAGEMENT	89
	APPENDIX B: LEVEL 2 OF THE CMM	92

TABLE OF FIGURES

Figure 1: ECM Technologies.....	12
Figure 2: Processes cut across traditional stovepipe boundaries	15
Figure 3: The innovation process	22
Figure 4: CText's organisational structure	33
Figure 5: Conway's SDLC with deliverables	48
Figure 6: Dimensions of the RUP	49
Figure 7: Example of a project Gantt chart.....	50
Figure 8: An overview of Step Wise	51
Figure 9: PRINCE2 process model	52
Figure 10: The Levels of Software Process Maturity and key process areas.....	59
Figure 11: The breathing polarity (Johnson, 1996:21)	89
Figure 12: Standardisation-Innovation polarity map for CText	91

1 NATURE AND EXTENT OF THE STUDY

1.1 INTRODUCTION AND CONTEXTUALISATION

The Centre for Text Technology (CTexT™) is an intrapreneurial organisation within the North-West University. The concept was born from the output successes of the sub-programme *Language & Technology* within the Research Focus Area: *Languages and Literature in the South African Context* in the *Faculty of Arts* (CTexT, 2006:1).

It was decided in 2003 to consolidate these activities and CTexT began operations on 1 June 2004 as a non-profit, self-sustaining (but not separate legal entity) research and development centre at the North-West University, focusing on Human Language Technology (HLT) and Computational Linguistics (CTexT, 2006:1). The Centre received R1 455 000 from the University's strategic funding for 43 months (that is, until 31 December 2007), after which the sustainability is to be reevaluated by an advisory committee. With this in mind, the Centre is required to prove its sustainability in the business arena too, through commercialisation of its expertise.

It is stated in its constitution that CTexT has a vision to be an internationally acknowledged, innovative research and development centre for text technology sources and applications (CTexT, 2006:2).

Its mission is to:

- Do innovative research of international merit;
- Ensure long-term sustainability through software product development; and
- Obtain external funding for research and development activities.

The core business of the Centre is thus software development, supported by innovative research and funded by non-traditional sources (CTexT, 2006:2).

CTexT also functions as an outlet for products resulting from its research and development activities:

- It has released spelling checkers for five South African languages and is engaged into expanding these to other operating systems and applications (e.g. *Microsoft Office® for Mac* and *Adobe Indesign*).

- The Centre partnered with Microsoft SA to localise *Microsoft Windows®* and *Microsoft Office®* in three South African languages.
- CText has been approached by Microsoft Ireland Operations Limited for the development of spelling checkers for languages spoken in African countries, such as Nigeria, Senegal and Rwanda. These projects are to be executed in partnership with African linguists over significant geographical and technological divides.
- At the time of writing this dissertation, the Centre, together with another university, secured a national government tender to develop spelling checkers and automatic translation systems for ten official South African languages.
- CText has acquired the full copyright of three Computer-Assisted Language Learning (CALL) software packages, which it now maintains and expands; the graphical user interface has been changed and two new releases for other languages have followed in 2006. There are plans to develop similar programmes for specific language purposes (e.g. Setswana for teachers).
- At the time of writing of this dissertation, CText was engaged in the planning phase of a multimillion rand project for a new generation CALL project for eleven South African languages.

CText has grown from a group of five with a dedicated development task of one product, to a core group of seven, a number of contractual workers, and a software product range of nine. The new projects will see the Centre triple its staff to about 20 employees by the start of 2007, and the opening of a satellite office at the University of Pretoria. As a result, logistics too have grown exponentially and will continue to do so over the next three years.

Of late, it has come to management's attention that a lack of formal processes is leading to problems with software development. Examples include incomplete technical communication to the customer support centre on known software issues (**customer support management**), version control issues (**versioning** and **configuration management**) and insufficient software evaluation processes (**quality management**). There have also been some examples of flawed **risk and project management** on development projects. The cost to CText has been substantial, especially with regard to time spent on developing patches for software errors which should have been detected through pre-release testing. In one case, the entire stock of a product had to be withdrawn from the warehouse and

new software had to be printed, with significant cost implication in the context of the relative small scale on which the Centre operates.

Although the need for improvement should be obvious, there are three specific reasons to correct the situation urgently:

1. **Cost reduction:** more efficient processes should have a positive impact on productivity and ultimately profitability.
2. **Quality assurance:** CText has a reputation for high quality to maintain.
3. **Effective project management of new projects:** Projects are becoming increasingly complex, drawing upon diverse expertise of geographically dispersed team members. This increases the need for effective project management.

CText's management is of the opinion that an electronic workflow management system (WfMS) or business process management system (BPMS) might hold the solution to its software development problems, and has issued its project manager with the task of investigating the matter and making a recommendation for the implementation of an appropriate system.

1.2 PROBLEM STATEMENT

CText is famous within its operating environment for the unconventional ways in which it thinks and works. Under an exceptional leadership style from the head of the Centre, with support from the dean, an intrapreneurial culture has been created in which employees feel encouraged to explore their ideas, question established ways of doing and make suggestions for improvement, to "shoot from the hip" and to act on gut feel. Bending the rules is tolerated if it may lead to new wisdom. Independence and freedom are valued and consequently care is taken to keep the Centre free from red tape and obtrusive measures wherever possible, so that CText finds itself in an ideal intrapreneurial working environment. Most of the Centre's commercial successes so far can be traced back to innovative and creative thinking by this team.

Having said this, the first problem area to have been identified relates to the **introduction** issues, thus, the probability of resistance from within CText towards a WfMS or BPMS.

According to Trott (2005:92-93), people become discouraged when they have to change established ways of working, while feeling that their autonomy is being undermined. Relating this back to CText, one has to wonder about the reaction from some of the creative independent thinkers to such a controlling system.

A related issue is that of the possible negative effects on some of CText's core strengths. An exploratory investigation into workflow solutions has uncovered an interesting and somewhat worrying claim regarding enterprise resource planning (ERP), namely that it has a detrimental effect on creativity and innovation. ERP is a related field which focuses on information management and data integration (Cardoso *et al.*, 2003). It is aimed at re-engineering, automation and integration of an organisation's core business processes such as manufacturing, distribution, finance and human resources (O'Brien, 2004:52), and as Smith and Fingar (2003a:2) point out, most ERP systems today include a workflow component.

Trott (2005:93) questions the demands of these systems on existing organisational processes, which are forced to fit their system demands. He explains that ERP require discipline, "*whereas freedom and creativity in the form of professional autonomy is continually cited as necessary for innovation to occur*" (Trott, 2005:95). Under ERP systems there is a preference of explicit knowledge over tacit knowledge, resulting in the establishment of a culture in which decisions are made from a selection of drop-down menus, while risk taking and experimentation become undesirable (Trott & Hoecht, 2004:373-375).

Whereas ERP systems rely on an organisation adapting to its predefined processes, BPMS let an organisation define its own workflow models, and therefore, the effect of its implementation on the innovative capacity should not be a matter of concern to an organisation such as CText. However, creativity and innovation form the basis for competitive advantage at CText, and it thus seems imperative that this issue be examined thoroughly in order to either discard or confirm the ERP-BPM analogy.

The second issue relates to **implementation**. Firstly, there is the matter of considering which processes to subject to workflow or business process management (Becker *et al.*, 1999). For instance, it is not certain at CText whether a WfMS/BPMS is a viable solution to its software development processes, as opposed to its support functions. There is doubt among management as to whether or not a system can and should be imposed on

this terrain and it is asked whether it is not a rather drastic remedy step. It is possible that there are other, simpler solutions. Furthermore, time and cost to implement and maintain are critical factors pointed out in the literature (Khan, 2005; Moonley, 2005:3 and Trott & Hoecht, 2004:373-375). CText certainly does not have the capacity to opt for an expensive, resource-intensive WfMS/BPMS and furthermore, the Centre will be bound by prescriptions and restrictions of the University regarding deployment of systems within its IT infrastructure.

The general research question of this study is then: Can the implementation of a WfMS/BPMS effectively solve CText's development problems and thereby improve its overall software development capacity?

1.3 SPECIFIC RESEARCH QUESTIONS

The general research question induces the following specific questions:

1. Introduction: Can CText reconcile creativity and innovation with standardisation, by introducing a WfMS/BPMS? What level of resistance can be expected from its personnel towards such a system?
2. Implementation: Which processes should be subjected to workflow or business process management, and how can CText overcome the time and cost constraints of such implementation?

1.4 OBJECTIVES

The primary objective of this study is to investigate and develop recommendations regarding the appropriateness of introducing a WfMS/BPMS at CText.

The secondary objectives, then, are as follows:

1. To determine the risk of harm to creativity and innovation by a WfMS/BPMS to an intrapreneurial software organisation like CText, as well as the predisposition of personnel at CText towards such a system.

2. To identify all the problem areas in CText and to determine which of these can be remedied with a WfMS/BPMS, taking into account the time and cost constraints present at CText, as an intrapreneurial organisation within the North-West University.

1.5 RESEARCH METHODOLOGY

1.5.1 Meeting the Primary Objective: Recommendation

Having completed the study, the author hopes to be in a position to make an informed recommendation regarding implementing a WfMS/BPMS. By looking at the problems and concerns identified in the Secondary Objectives 1 and 2, and the extent to which each of them can be resolved through a WfMS/BPMS, a decision can be made on the feasibility of such an implementation.

1.5.2 Meeting Secondary Objective 1: Introduction

An extensive literature review will be undertaken to establish whether or not the ERP-BPM analogy holds any truth.

Semi-structured interviews with CText employees will be conducted to determine their position with regard to a workflow or BPM intervention on their daily work tasks relating to software development.

A further literature review will be undertaken with the intention of advising management on an appropriate introduction strategy to obtain buy-in from staff, while upholding the current creative and innovative culture of the Centre.

Finally, interviews with knowledgeable individuals from other organisations will add useful and practical perspectives on these issues.

1.5.3 Meeting Secondary Objective 2: Implementation

Information regarding the employees' specific development problems will be gathered through interviews, and through scrutinising documentation (that is project plans and technical reports) regarding past projects.

An investigation into actual successes of WfMS/BPMS at other software development organisations through literature reviews and interviews with managers will help answer the question of the appropriateness of a WfMS/BPMS in addressing each of these problems.

1.6 SCOPE AND DEMARCATION OF THE STUDY

This study will be executed within the broad theoretic framework of BPM, with specific focus on its role in an intrapreneurial software development organisation. The study will be limited to BPM within software development; the assumption is made that operational and support functions are carried out in accordance with the university policies and infrastructure and can therefore not be subjected to alternative management systems or practices.

Further demarcation can be done with reference to the BPM life cycle, in that the study will limit itself to the first two stages thereof. Microsoft (MC, 2006a:7-9) explicates the following iterative stages of this cycle:

- **Planning:** this involves gaining an understanding of the organisation's current business practices, and planning to improve this situation. Candidate BPM projects and key players are identified and governance established. The organisation's culture should also enjoy sufficient attention, since a lack of buy-in may complicate the change effort to the point of complete project failure.
- **Selecting the business process to improve:** Processes with considerable impact on an organisation's ability to achieve its goals or its return on investments are good candidates.
- **Model and design:** This phase is overseen by a business analyst, who works with process owners and end users to define and document the target business process and its underlying business rules from end to end (and beyond organisational borders if required). Modelling is followed by an investigation into how IT can design improved support, for example through integration, automation or workflow re-design.
- **Develop and deploy:** The IT developer abstracts the business rules to a layer independent of their systems and applications, and then joins the logical components

into a combined application that combines the functionality of underlying systems.

- **Manage and interact:** The solution is set in motion and end users interact with the business process as it runs through its stages, while business users monitor the process for potentially disruptive events and act accordingly to remedy them.
- **Analyse and optimise:** This phase completes and repeats the cycle. Service level agreements and key performance indicators are used to establish benchmarks, to which performance information is measured. The goal is to develop an iterative process which enables real-time optimisation of business rules.

1.7 DEPLOYMENT OF THE STUDY

Chapter 1 has served as an introduction into the field and purpose of this study. A brief overview of CText, its products and its growth was followed by an indication of the problems the Centre is currently experiencing as a result of its growth and flawed management principles. The general research question posed relates to whether the implementation of a workflow or business process management system can effectively solve its development problems and thereby improve CText's overall software development capacity, given its nature as an intrapreneurial organisation. The study is narrowed down to the first two stages of the BPM life cycle, namely planning and selection of the business process for improvement.

The rest of the study will be deployed as follows:

Chapter 2 deals with workflow and BPM terminology by means of a literature study, and culminates into working definitions for the purpose of this dissertation. A brief technical perspective of a WfMS/BPMS is provided, followed by some notes on specifications and standardisation. Specific areas for implementation are identified.

Chapter 3 serves to meet Secondary Objective 1, namely the determination of the risk posed by BPM to creativity and innovation at intrapreneurial organisations. This will be accomplished by examining the advantages and disadvantages of BPM in the literature and the industry. The results are then tested at CText to determine relevance and to identify particular problem areas with regard to creativity and innovation.

Chapter 4 focuses on BPM in the domain of software development, and the associated challenges and solutions. These characteristics are then mapped to CText's case. The result is an answer to Secondary Objective 2, namely to establish whether a WfMS/BPMS can effectively solve the prevalent software development issues at CText.

Chapter 5 concludes with a recommendation to CText for future management of its software development projects, based on the outcomes of the preceding two chapters.

2 TERMINOLOGIES

2.1 BPM AND WORKFLOW: DIFFERENT TERMS FOR THE SAME CONCEPTS?

A search through the literature and marketing content of vendors brings the realisation that there are about as many definitions and performance criteria for workflow and BPM as there are vendors and specialists. Software solutions which help automate and streamline certain aspects of business go around by many names, often with overlapping definitions. There are a number of reasons for this, e.g.:

- Srinivasan (2006) writes of BPM that due to its multiple uses it is hard to find a common definition, and even more problematic to find one technology market prepared to handle all the needs.
- Kemsley (2006) explains that there are many vendors jostling for position within the industry, and that their respective definitions of BPM are based on those capabilities which their particular offerings do and do not contain.
- In an article about the state of workflow Baeyens (2004) states that WfMS still find themselves at the initial phase of the technology hype curve, where there is not yet consensus on workflow terminology and standards; that these concepts and the understanding of the field still vary greatly. According to Silver (2006a:7), BPMS offerings derived from a workflow orientation differ from those derived from an integration/infrastructure perspective, and that these differences are the cause of the current debate over BPM standards.

To confuse matters further, there are even conflicting definitions.

- For example, McGovern (2005) writes that though some would maintain that the difference is not really that clear, BPM is generally seen as a superset of workflow, differentiated by the ability to coordinate activities across multiple applications with fine grain control. Silver (2006b) also sees workflow as a feature of BPM.
- Others argue that it is a marketing matter, and that BPM is a newer and more impressive term for workflow (Miers, 2005:23). A recent blog discussion supports this view; it is contended that workflow is a somewhat derogative term compared to BPM, and as a result vendors and workflow practitioners have adopted the latter

since 2001, thus creating the impression that workflow is old and BPM a newer, better technology (Swenson, 2006b; Kemsley, 2006). Chappell (2006), on the other hand predicts that Microsoft as a major vendor will soon influence the scene with a forthcoming unified process architecture for all of its technologies: Windows Workflow Foundation (also see MC: 2006a:19-20).

- It is then also suggested that Microsoft is viewing workflow as the overarching term of which BPM is a part (Chappell, 2006). Microsoft itself uses workflow to refer to *“the ability to coordinate work done by software and by people”* (Chappell, 2005:27) and has in the past attached to it the meaning of human oriented business processes (MC, 2004).
- Brown and Widell (2006:4) make the interesting statement that what may be BPM in theory, becomes workflow in practice. They ascribe this to the ever-changing working environment to which business process models cannot adapt appropriately.

Against this backdrop a more thorough investigation into these concepts needs to be conducted with the purpose of formulating working definitions for the subject of this study.

2.1.1 Enterprise Content Management

Jenkins (2005:18,30) provides a useful perspective on the link between workflow and BPM, through the topic of enterprise content management (ECM). ECM is a collective noun for all technologies that connect people with each other and information, and it consists of collaboration (communication, messaging, sharing, etc.) and content (documents, archives, searching etc.) technologies. While Jenkins does not distinguish clearly between workflow and BPM, and on occasion discusses them as one concept (2005:33, 231-245), he does indicate them as separate technologies in the illustration below.

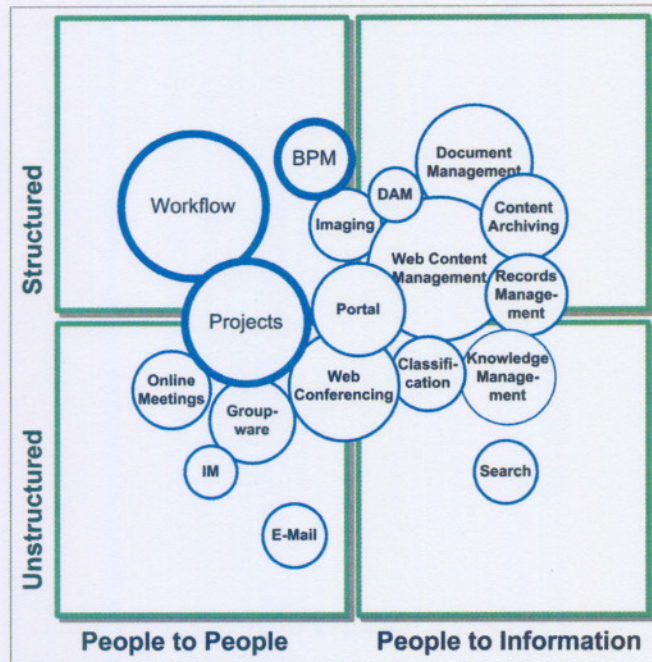


Figure 1: ECM Technologies
(Jenkins, 2005:30)

Figure 1 shows that workflow and BPM belong to the collaboration category and both deal with structured business processes between people.

Projects and project management will also be discussed in this study; therefore, project technologies too have been highlighted. Project technologies handle both structured and unstructured processes.

2.1.2 Workflow

The term **workflow** refers to the automated movement of documents and/or tasks through a work process according to a set of procedural rules (e-workflow, 2006). Becker *et al.* (1999:1) define workflow as a sequence of activities required to perform operations on economically relevant objects, whose control logic lies within the control sphere of an information system.

Cardoso *et al.* (2003) provide the following definition for a **Workflow Management System (WfMS)**:

“Under a WfMS, a workflow model is first created to specify organizational business processes, and then workflow instances are created to carry out the actual

steps described in the workflow model. During the workflow execution, the workflow instances can access legacy systems, databases, applications, and can interact with users”.

According to Kemsley (2006), early workflow systems were usually document-focused.

Enterprise application integration (EAI) is the process of linking different applications with the goal of automating the exchange of data between systems to create competitive advantages for organisations (Wikipedia, 2006a; Kemsley, 2006). EAI emerged independently from workflow in the 1980s for system-to-system integration and resulted in two types of benefits: data no longer had to be re-entered in order to transfer it between systems, and information could be exchanged between systems in near-real-time (Kemsley, 2006). Workflow and EAI started merging their capabilities into single systems by the late 1990s. Today EAI is still considered to be a vital part of the BPM landscape (Pyke, 2006:17).

2.1.3 Business Process Management

Savvion (2006a) defines a **business process** as “*an aggregation of operations performed by people and software systems containing the information used in the process, along with the applicable business rules*”. According to Burlton (2006?:3), a business process is triggered by an external event, which leads to certain transformational actions that follow logical steps and are measured by performance indicators, and ends with the delivery of a product or service to either an external stakeholder or another internal process. Microsoft extends the meaning to literally include everything that is done in an organisation to support its functioning (MC, 2006a:4).

Gilbert (2005) explains that **business process management** (emphasising the lower case to distinguish it from software solutions) relates to an organisation’s discipline as far as understanding, executing and improving their business processes are concerned. Silver (2006a:2) calls it a management discipline for thinking about the business in terms of cross-functional processes, as opposed to the traditional departmental view. BPM seeks to improve business performance through optimisation of processes from end to end (Silver, 2006c).

A Business Process Management System/Suite is an enabler to BPM (Ranganathan &

DuBrock, 2004; Burlton, 2006?): it enables the execution of business processes by assigning tasks to human or computer agents according to the predefined definitions of the processes (Ha *et al.*, 2006:64). With a comprehensive BPM platform, an organisation can set up its business processes as applications accessible via the Web, integrated with existing software and operational systems such as ERP and databases, and allow managers to monitor, analyse and control the execution of these processes in real time (Savvion, 2006a). According to Silver (2006a:2), BPMS break down departmental boundaries for end-to-end efficiency, agility, compliance and visibility to cross-functional business processes.

2.1.4 Working definitions for workflow and BPM

One might endeavour to make sense of the preceding discussion by inspecting the topics through the lenses of a non-technical business user.

Pyke (2006:17) attempts a description of BPM in terms that business users understand best. According to him, these users typically see BPM as *“a way of managing cases or tasks in a predefined sequence; getting the right information to the right place at the right time to meet a business need. To them BPM is something that reduces the risk of error, gets tasks completed sooner and more effectively and makes the whole business or running a business easier and more manageable”*.

This last quote also becomes a good outcome to Swenson's suggestion to define BPM and workflow by the results that they produce (Swenson, 2006a). He then explains that workflow and BPM are a *“philosophical approach to produce applications that allow people to work in a more coordinated manner”* (Swenson, 2006a). Incidentally, one of the themes discussed at the South African 3rd annual BPM Congress was that of BPM being a philosophy, not necessarily a technology (IQPC, 2006:1).

By adopting this “business user” view, the discussion about the relative domains, similarities and differences regarding BPM and workflow essentially becomes irrelevant. As a small intrapreneurial entity, CText is hardly in the position to create the kind of end-to-end integration across organisational borders as is one of the ambitious goals of BPM. Currently it is seeking to automate certain processes internally, although this should by no means exclude the possibility of future expansion to other processes and organisations. Therefore, for the remainder of this study, the use of the term workflow shall be restricted

to the technical perspective (c. Section 2.2), while the following view on and implied definition for BPM will apply throughout the rest of this study:

BPM is a philosophical approach to organisation-wide management in which the focus is on the processes through which it operates, and in particular on the streamlining and optimising of these processes, for which software solutions may be used. Exercised in the narrow sense of the word, the focus might be on internal processes only. In the broadest sense, BPM can facilitate work and communication flow between various databases within an organisation, and in addition extend beyond organisational borders to connect and involve other entities and their systems on which the organisation may be dependent for its operations.

2.2 A TECHNICAL PERSPECTIVE

BPM technology necessitates rather complicated systems. This Section offers a brief introduction into the technical facet of such systems.

Silver (2006a:1) explains that the need for BPM often stems from the fact that processes span various departments: a particular process can run through and be dependent on the departments of Sales, Finances and Support, and often there is a problem with integration.

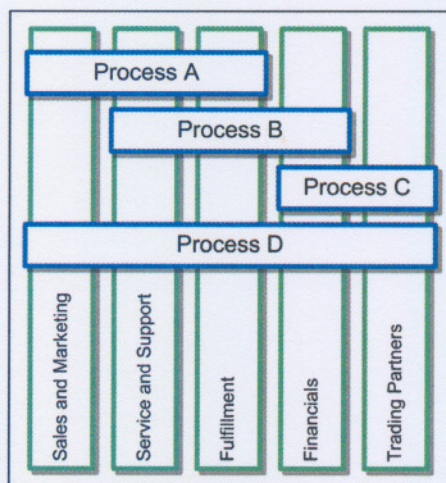


Figure 2: Processes cut across traditional stovepipe boundaries (Silver, 2006a:1)

Each of these departments may use different systems (e.g. ERP or customer management systems) which optimise operations within the department, but may at the same time hinder agility across organisational and system boundaries. BPMS break down these barriers to improve end-to-end efficiency, agility and visibility to cross-functional business processes (Silver, 2006a:2).

2.2.1 BPMS technology overview

This section is taken from the 2006 BPMS Report (Silver, 2006a:4-13).

- **Process modelling:** This is traditionally an abstract description of the process. Of late, the notion of executable process models have become popular, to refer to graphically (i.e. not programmed) built process models which can be deployed to a process engine to actually manage processes.
- **Process automation:** When triggered by conditions (events) as specified by the model, the process engine creates a new instance of the process and executes it, for example by forwarding work to a participant, or invoking an executable object for automated steps.

Routing rules based on logical expressions of process data regulate flow at branch points. Some BPMS allow for splits into parallel segments to give a single process multiple threads of control.

BPMS can also handle faults and changes, and typically exceptions. For instance, when incoming data does not match the required schema, or when an order in process is changed or cancelled, special exception handlers intervene, which can even reverse the effects of completed work.

- **Process architecture and standards:** Major differences in BPMS offerings can be attributed to the fact that some are derived from a workflow orientation, while others follow an integration or infrastructure perspective. The latter emphasises integration and is based on service-oriented architecture and the BPEL process definition language.

Most BPMS products are, however, based on workflow architecture, since the main purpose in BPM remains streamlining human work. This architecture emphasises

human workflow and is based on the reference model of the Workflow Management Coalition (WfMC) and its XPDL process definition language.

- **Business rules:** Traditional process rules are defined as Boolean expressions and consequently simple. They are bound to specific points in the process model and must therefore be replicated at each point of use in a process model. In contrast, business rule engines (BRE) are complex and powerful, and since they are defined globally and stored centrally, there is no need for replication to all areas of use when changes are made. Very importantly, BRE parameters can be modified by business people themselves through simple web interfaces, and such changes take immediate effect, improving BPM agility to changing circumstances.
- **Application integration:** Application integration middleware facilitate the execution of application programming interfaces (APIs), object methods or web service operations on external applications and information systems with little or no programming. Through a communications bus, integration adaptors and data transformation, the functions of multiple external systems can be coordinated with little code, regardless of varying platforms, languages and data modes.
- **Performance management:** During the process, snapshots of process data and timestamps of step completions are logged for use in performance management. Some BPMS can deliver graphical representations of key performance indicators, while graphical dashboards providing an overview of the process performance is a common feature to most.
- **Continuous process improvement:** BPM design tools ensure a more collaborative effort between business and IT than in conventional software development. Through modelling and simulation the process design can be tested and refined before deployment.

2.2.2 The workflow engine

The WfMC has developed a reference model to describe the bigger picture and how standards fit together (Plesums, 2002:34). This model identifies five interfaces to the workflow engine:

- The process definition relates to the procedures followed in implementing workflow, and the resources that perform the work.
- The client interface relates to the way an application program invokes workflow, such as a request to get the next piece of work.
- The third interface exists to invoke other applications, such as document management systems. By tying together dedicated applications, EAI can be achieved (Baeyens, 2004).
- External workflow services enable interoperability between workflow systems, such as between an organisation and its suppliers.
- The fifth interface facilitates administration and monitoring, for recording the history of each case and the monitoring of the total work. Managers can extract statistics on audit trails and processes (Baeyens, 2004).

2.2.3 Standards and specifications

Despite the growing interest in this field and the amount of work that has been done here, standards and specifications still vary. According to Pyke (2006:26), there are currently more than ten bodies involved in defining standards for process-based technologies and the standards specifications are extremely complex. Some of the initiatives in this regard are:

- The WfMC is a consortium of around 300 that works to develop international standards for interoperability of the different components of workflow (Plesums, 2002:34).
- The BPM Group has 16 000 members in 155 countries who exchange ideas and best practice in BPM and change management (BPMG, 2006). Members are supported through case studies, seminars, education and research (BPMG, 2006).
- The BPM Institute is a peer to peer exchange for BPM professionals who endeavor

our to advance the adoption and implementation of BPM solutions and practices (BPMP, 2006).

- XPDL is the WfMC specification that defines an XML schema for specifying the control flow of business processes (Prior, 2003:17; Baeyens, 2004).
- The Business Process Execution Language (BPEL) is the result of inputs by several major companies and was designed for platform- and code-independent integration of a variety of applications that are run to accomplish a particular business goal (Matlis, 2005).
- The Universal Business Language (UBL) defines a standard library of XML documents to be used for communication between organisations (Baeyens, 2004).
- The Universal Modeling Language (UML) is used by developers to specify, visualise and document models of software systems, and can also be applied for business modelling (Siegel, 2005).

2.3 IMPLEMENTATION AREAS

The question arises as to which business processes can be streamlined through BPM. To answer this question, it is helpful to look at vendor marketing content.

Vendors typically recommend their offerings to the following business areas:

- Customer relations;
- Operations and Manufacturing;
- Revenue cycles (sales forecasting, account and rebate management, etc.);
- Process documentation (e.g. for regulatory compliance);
- Risk management;
- Project management;
- Trading partner interactions;
- Administrative efficiencies, e.g. finances;
- Franchise management;

- Human resources;
- Six Sigma (A philosophy and method for eliminating defects in products and processes (Chase *et al.*, 2004:279); and
- Software (of all the vendor content studied, only Singularity and IBM were found to make provision for software development).

(FlowCentric, 2005; IBM, 2005; MC, 2006b; Metastorm, 2006; Savvion, 2006a; Singularity, 2006).

Based on the above, the conclusion can be drawn that BPM is well-suited for routine and repetitive processes. Order and request processing, production-line activities and human resource management consist of standard processes and draw upon information databases for certain inputs and outputs. The capability of software solutions to record audit trails make BPM ideal for industries that have to document their processes for regulatory purposes (Metastorm, 2006; Savvion, 2006b:2), or those applying Six Sigma (Savvion, 2006b:4, Smith & Fingar, 2003b:1). Goldman (2006) too notes that BPM works best for businesses where tasks are repeated continuously and have to conform to regulations, whereas processes that differ on a case-by-case basis may find BPM restrictive.

This does not exclude non-routine processes, though. Silver (2006a:13,15) distinguishes between six types of use cases, one being the complex collaborative use case, which has ad hoc flow BPM features. However, he admits that very few BPMS are well suited for this type of process flow (Silver, 2006a:15).

In the arena of software development, there is very little available information. IBM is one of the exceptions, offering software development infrastructures tailored to special organisational needs (IBM, 2005). Singularity (2006) is another; it claims to help its customers leverage process-centric software engineering approaches through BPM software and methodologies.

2.4 SUMMARY

In this chapter it was shown that workflow and BPM are related concepts; in fact, there is no consensus on their exact meaning and distinctions. For the purpose of this study BPM shall be used as a collective term to refer to a philosophical management approach for streamlining and optimising of the processes within, and possibly between organisations, for which software solutions may be used.

A summary of ongoing standardisation initiatives indicated that BPM is a growing field with a variety of standards and specifications being developed by a number of bodies.

It was shown that BPM is well-suited for transactional and routine processes, but less so for non-routine, ad hoc flows. Only two from the sample of BPM vendors made provision for software development.

The following two chapters will narrow the focus to two areas: the compatibility of BPM and innovation (Chapter 3), and the applicability of BPM to software development (Chapter 4).

3 BPM AND INNOVATION IN AN INTRAPRENEURIAL ORGANISATION: CHALLENGES AND SOLUTIONS

3.1 INTRODUCTION

As it was described in Chapter 1, CText places high value on its innovativeness and considers it a core strength. Therefore, the first objective of this study is to investigate the potential of BPM to adversely affect this core strength. This can be achieved by looking at the disadvantages of BPM with relation to creativity and innovation in intrapreneurial organisations, since the disadvantages are assumed to be representative of the risks associated with BPM.

This chapter starts with definitions for creativity, innovation and intrapreneurship. An investigation into the disadvantages associated with BPM within the context of intrapreneurship is followed by an investigation into possible solutions to these identified problems. Finally, the results of the interviews held with six employees from CText are presented in order to determine the applicability of the known BPM disadvantages and employees' predisposition towards a BPMS, so that a conclusion may be drawn regarding the danger of BPM to the Centre's innovativeness.

3.2 DEFINING CONCEPTS

According to Morris and Kuratko (2002:121), in entrepreneurial activity, creativity leads to innovation and entrepreneurship drives the process from idea to harvesting. Burns defines the relationship as one where high creativity and invention, combined with high levels of opportunity perception and entrepreneurship leads to innovation (Burns, 2005:249).

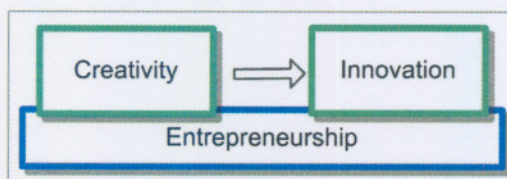


Figure 3: The innovation process

3.2.1 Intrapreneurship

Intrapreneurship, otherwise known as corporate entrepreneurship, relates to entrepreneurial behaviour inside established organisations (Burns, 2005:11; Morris & Kuratko, 2002:31). Therefore the characteristics of entrepreneurship apply to intrapreneurs too, the only real difference being the setting in which the entrepreneurial activity takes place.

Entrepreneurs are central to innovation, since they are the ones who introduce and exploit innovations (Burns, 2005:246). They have the vision to turn ideas into profitable realities and are the drivers behind the implementation of innovative concepts (Morris & Kuratko, 2002:85).

3.2.2 Creativity

Creativity relates to the capacity to think outside the norm and develop new ideas, concepts and processes through divergent, tangential thinking (Handzic & Chaimungkalanont, 2004). Morris and Kuratko (2002:104) call this the soul of entrepreneurship, pointing out that it is a requirement for identifying opportunities and developing innovative business concepts. A creative person applies her/his intellectual ability and curiosity to discover new things and relate previously unrelated things (Morris & Kuratko, 2002:104).

3.2.3 Innovation

According to Handzic and Chaimungkalanont (2004), innovation represents creativity in action; it involves refining the ideas brought about by creativity and transforming them into useful solutions.

Innovativeness is one of the cornerstones of entrepreneurship (Morris & Kuratko, 2002:39; Timmons & Spinelli, 2003:10). Innovation is a new way of doing something, and can lead to the introduction of new methods, processes, technologies, products or services (Morris & Kuratko, 2002:121). Bessant *et al.* (2003:19) explain that product and process innovation are fundamental for survival in an increasingly hostile and turbulent environment; that without it there is a considerable risk of failure and losing competitive edge.

Åmo and Kolvereid (2005:17) have proven successfully that by putting a corporate entrepreneurship strategy in place and/or training employees in innovation and entrepreneurship, innovation behaviour is likely to increase.

3.3 GENERAL ORGANISATIONAL CONSTRAINTS ON INTRAPRENEURSHIP

3.3.1 Administrative barriers

Intrapreneurship faces enemies of various sorts from within the organisation. Morris (1998:97) organises these in categories of constraints, including systems, structures and policies, among others:

- Systems that are oppressive, inflexible or require rigid planning.
- Structures that have too many hierarchical levels, overly narrow span of control, top-down management, and restricted communication channels.
- Policies and procedures that have complex approval cycles, extensive documentation requirements and over-reliance on established rules of thumb.

The high level of efficiency and profitability resulting from these structures and systems is set off by a loss of the intrapreneurial spirit (Anon, 2006?) and may lead to commoditisation (Coulson-Thomas, 2005b), both of which are most unattractive features in organisations wishing to differentiate themselves through innovative thought and action. Morris and Kuratko (2002:87-88) observe that intrapreneurs tend to dislike systems and learn to manipulate and outwit it.

Manimala and colleagues have recently made similar findings, in a study on innovation constraints on Indian organisations. They have found that inadequate supporting systems and procedures inhibit innovation behaviour (Manimala *et al*, 2006:49-56). The fifteen-item constraint list includes lack of recognition for innovation in non-core areas, ambivalent supervisor support, procedural delays and poor documentation and maintenance of records. Most interesting about their results is that they indicate that while some barriers stem from inflexible or inefficient systems (e.g. time-consuming procurement systems), most are actually the result of absent systems (e.g. documentation).

Finally, the net result of an organisation's processes and systems can cause people with new ideas to leave the organisation. Coulson-Thomas (2005a) notes that employees with new business ideas need assistance in the initial development and implementation thereof, but if the organisation is bureaucratic and ill-equipped to handle new ventures, intending entrepreneurs may take their ideas elsewhere for support.

3.3.2 Project management

Project management is a widely acknowledged driver to ensuring the success of projects, but in the context of innovation the controlling characteristic of project management has on occasion been questioned. In an article on the relationship between project management and innovation Hildebrand illuminates two schools of thought: “*Traditionalists believe that the rigor of project management slows innovation, while others understand that a project management framework frees a team to brainstorm creative solutions.*” (Hildebrand, 2005:38).

3.3.3 Group dynamics

Sethi *et al.* (2002:17), while acknowledging the importance of cohesion (in moderation), point out the danger inherent to this: social cohesion between long-time team members can lead them to suppress the exchange of opinions in favour of maintaining good relationships. The consequence is then that innovation suffers.

3.4 CHALLENGES RELATING TO BPM AND INTRAPRENEURSHIP

Literature on BPM in intrapreneurial settings is basically non-existent. BPM is seen as an innovation in itself (Smith & Fingar, 2003a:24) and an innovative approach to driving business (Mooney, 2005:1; Fingar & Bellini, 2004:20-21). However, the matter of its impact on innovation and intrapreneurial activities does not seem to have enjoyed any attention yet. Therefore, this investigation will have to rely on deductions made from significant characteristics and objectives of BPM and intrapreneurship respectively to determine whether these characteristics are complimentary or at odds.

3.4.1 Control

By comparing the characteristics of an intrapreneurial culture to that of a corporate culture, it may be deduced that while BPM complements the latter, chances are that it will interfere with the spirit of an intrapreneurial setting. Czernich (2003) describes the typical organisational environment as characterised by uncertainty avoidance, myopia and the sole use of tested methods to conduct business (which he calls “local search”). This increases efficiency of current operations while closing the door on innovative thinking. Hisrich and Pe-

ters (2005:40) observe that traditional corporate culture, distinguished by established procedures, reporting systems, lines of authority and responsibility, instructions and control mechanisms are not conducive to the guiding principles of intrapreneurs, namely creativity, flexibility, independence, ownership or risk taking.

BPMS ensure the execution of the rules and best practices regarding business processes to achieve the abovementioned corporate efficiencies, and this single fact casts a shadow over its prospects in an intrapreneurial setting, which in essence is everything that a typical corporate setting is not.

3.4.2 Toleration of failure

One particular interesting characteristic of intrapreneurship is its predisposition towards failure. By its very nature, intrapreneurial environments encourage trial and error, and failure is tolerated in developing new products (Hisrich & Peters, 2005:43). In contrast, BPM promises to support business in such a way that decisions and operations are more efficient, with an implied reduction of error. It is specifically this potential for error reduction at CText which led to this investigation in the first place.

3.4.3 ERP challenges

Goldman (2006) claims that BPM suffer the same disadvantages as ERP, which is not particularly good news to an investigation concerned about the truth of the BPM-ERP innovation analogy. To illustrate his point, Goldman provides a hyperlink to a Wikipedia listing (2006b). The following specific disadvantages should be highlighted in the context of this study:

- Systems can be expensive to install, and difficult to use and troubleshoot.
- When new managers are appointed, they may propose changes in business practices that are out of synchronisation with the best utilisation of the company's system.
- In small organisations which are often undercapitalised, the system is often operated by personnel with inadequate education concerning ERP.
- Resistance in sharing sensitive internal information between departments can reduce the effectiveness.
- Compatibility problems with legacy systems of partners.

- The system may be over-engineered relative to the customer needs.

3.4.4 Practical challenges to small organisations

Over and above the challenges to innovation, there are other practical matters to consider. Since intrapreneurial organisations tend to be small (as they are usually departments or working teams within a larger organisation), it should be helpful to look at BPM within small organisational contexts. Available literature is very limited, but the following issues are raised:

3.4.4.1 Limited choice

Goldman (2006) warns that small businesses who choose to use BPM, will have limited options. The general impression left by many BPM vendor marketing material is that their solutions are aimed at larger organisations; for instance, that they will deliver improved coordination between silo departments and across organisational boundaries to the value chain partners. Case studies and reports mention hospitals, manufacturers, banks, insurers and government departments. These are generally large entities with a wide variety of processes and high levels of complexity. It remains to be investigated whether small organisations can get the required benefits from these systems.

3.4.4.2 Cost and complexity

BPM technology is often expensive, requires the help of a business analyst and IT staff to configure and roll out (Chappell, 2005:5-6), and this can be quite an endeavour even for a larger organisation with existing IT policies and infrastructure, or one which does not use BPM itself. According to Stoker (2006), such systems can actually slow down the processes for small organisations.

BPM is more than a technology issue; the successful adoption thereof requires radical cultural changes and discipline for an organisation to become process-oriented (Kristick, 2006).

3.5 SOLUTIONS

This section offers solutions to the issues raised in the previous section. In Section 3.5.1 some general managerial advice for the preservation of innovation is presented. Section 3.5.2 looks at solutions for unique intrapreneurial challenges.

3.5.1 Managing innovation

3.5.1.1 Strategic and operational management support

Manimala *et al.* (2006:49, 56) stress the need to put innovation on the corporate agenda, with an encompassing innovation strategy that treats both research and development facilities and organisational support for innovation as complementary systems that lead to innovation within organisations. According to them, neglect of either will constrain innovation. Riederer *et al.* (2005:9-10) point out that an innovation strategy which is understood by everyone in the organisation is the first step in innovation management. Their report on innovation management furthermore continuously points to a variety of communication means as a best-practice to improve creativity (2005:15, 21, 27, 28, 39, 42, 49).

Dos Santos (2006), director of South African Innovation Hub business Applisential, agrees. According to him, their innovativeness is the result of aligning processes with corporate strategy, designing the processes and employing the right tool to enforce them. As a result, management can devote their creative time to creating differentiated products and growing the business rather than spending their time controlling processes and enforcing discipline.

Sethi *et al.* (2002:17) have found that management encouragement of employees to be venturesome can have a significant impact on team innovativeness. In contrast to teams who worked according to continuous improvement practices based on established product development strategies, innovative teams were invited to stray from established approaches and pursue untried ideas.

Finally, Morris and Kuratko (2002:114) explain the concept of creative abrasion, which refers to the fusion of different approaches and perspectives to increase the level of stimulation and variety. Creative abrasion can be increased by hiring people who are different from current employees, giving a team two seemingly incompatible goals, hiring temporary staff, or inviting speakers who hold very different points of view.

3.5.1.2 Project management

From a project management point of view, Hildebrand (2005:39) relates how senior management of the Chrysler Group define prospective projects in terms of the problem, rather than the solution, so that the team can have the opportunity to think innovatively about solving it. She also proposes that managers modify the phases of the standard project life cycle to better reflect the organisation's culture and business requirements. By adding phases, for example, opportunities are created for micro-innovation within the project process (Hildebrand, 2005:40).

3.5.1.3 A change culture

The description of workflow and BPM in Chapter 2 may have created the perception that BPM is only about setting and controlling processes to eliminate deviations and optimise productivity. This, however, accounts only for half of the reality of what seems to be a regulate-change polarity¹. Organisations in high-velocity markets such as computer hardware and software increasingly find themselves having to adapt to remain competitive (Thompson Jr. *et al.*, 2005:206). The winners among them utilise BPM to achieve this agility (Fingar & Bellini, 2003a:23; Metastorm, 2006; MC, 2006a:1).

Change management is an essential part of BPM, both in terms of guiding an organisation through the process of implementing BPM and in creating a change-*able* organisation. According to Chang (2006:29-45), continuous process improvement is a key principle and organisations need to be able to adapt to such continuous change. The ability to make changes to processes fast and simple is a key selling point of BPMS vendors (Metastorm, 2006; MC, 2006a:1). At the 3rd annual South African BPM Congress, the topic of change featured prominently in four presentations and two workshops (IQPC, 2006:1).

The conclusion drawn from this is that BPM need not weigh down intrapreneurs; in fact, the theory suggests that if applied correctly, it should support the organisation's work and continue to adapt to changing circumstances and needs.

¹ The reader is referred to Chapter 5 and Appendix A for an explanation of polarities and polarity management.

3.5.1.4 Learning networks

Bessant *et al.* endorse interorganisational learning networks as a capacity building instrument for the promotion of organisational learning which then leads to product and process innovation. Participants can discuss new perspectives, debate conventional wisdom, share experiences etc., and ultimately create a unique collective insight (Bessant *et al.*, 2003:20-23).

Yaruz and Heidelman (1999: 68, 70), who advocate that knowledge sharing takes place around the water cooler, claim that information technology has advanced to such an extent that this technology has become the water cooler of the 21st century. It provides inexpensive methods for employees to connect with peers and databases all over the world in search of solutions to their current work problems.

Dos Santos (2006) mentions peer review as a particular effective BPM tool borrowed from the academic community, which is often used in the Open Source community to satisfy the quality, reliability and repeatability criteria (which he calls essential attributes in the global community) of products. Provided that there is a mature approach to ownership of ideas, it encourages “star bursting”, thus, one idea leads to another, and the final product becomes more than the sum of the individual contributions.

3.5.1.5 Socialisation

Handzic and Chaimungkalanont (2004) have established a clear relationship between socialisation and creativity. Informal socialisation was found to contribute significantly to creativity, while organised socialisation (artificially manufactured situations where employees have an opportunity to socialise) had a somewhat smaller, but nevertheless noteworthy effect on creativity levels in the workplace.

3.5.2 BPM solutions for intrapreneurs

The majority of problems and challenges described in Sections 3.3 and 3.4 point to procedural problems within organisations; processes which are either flawed or absent. It therefore seems plausible that by scrutinising these processes and bringing them in line with intrapreneurial needs, one should be able to come to a satisfactory solution.

3.5.2.1 Incremental BPM deployment

BPM is not necessarily an organisation-wide, all-encompassing solution. Although this may be the ideal towards which analysts and vendors are striving, Douglass (2006) advises organisations to identify the critical organisational capabilities, determine how BPM can support them, and then establish which business processes will benefit from BPM. The implication for an intrapreneurial organisation is that it can choose to exclude certain activities from BPM, or apply BPM to as few as one process, with the possibility to expand it to other areas.

3.5.2.2 Implement the BPM philosophy

The view on BPM should not be limited to technology; technology is simply an enabler of the BPM philosophy. When BPM is viewed as a structured approach which employs methods, policies, metrics, management practices and software tools to manage and optimise processes (Harris-Ferrante, quoted by Gambrell, 2006), the possibilities become endless, and different tools can for instance be used to apply BPM to various processes.

Silver (2006c) too acknowledges that BPM is achievable without BPMS and that significant savings and process improvements are possible through cross-functional process thinking combined with business process analysis (BPA). BPA is in fact becoming a starting point for BPM projects, as an essential tool to optimise business processes and realise BPM cost and time savings (Blechar. & Sinure, 2006:1).

Dos Santos (2006) maintains that adherence to best practices that ensure quality, reliability and repeatability is crucial. This can be achieved through any range of tools, be they paper, e-mails, spreadsheets or BPMS. Swart *et al.* (2006) and Lyneham-Brown (2006) all agree that even the old-fashioned brown paper method, which refers to the practice to simply draw up plans and processes on a wall or board, can still be relevant and effective today amidst more fashionable technological alternatives. Only once the cost of maintaining a manual system becomes higher than the cost of an automated system, has the time come to acquire a BPMS (Dos Santos, 2006).

Stoker (2006) suggests creating business process awareness, so that each individual is aware that their work impacts on the process and that they in fact have an obligation to comment on ineffective aspects and make suggestions for improvement of the overall

process. According to him, this mode of thinking can improve understanding between the management and development functions, as everyone gains insight into the larger scope of things and the reasons behind the other parties' actions.

Prinsloo (2006) also stresses the importance of becoming aware of the business processes underlying projects, even if they are not managed through technology. She argues that it is crucial to understand these processes and adhere to them throughout development work.

3.5.2.3 Subscribing to services

Organisations who do not have the capacity to invest in and modify expensive solutions, can gain access to BPM through subscription fees. For instance, *Process Factory* is a product which promises cost effective deployment to organisations of all sizes, through on-demand and subscription services, so that customers need only pay for those BPM services they use (BPMG, 2005).

Stoker (2006) points out that a range of customised Open Source management tools can be utilised to facilitate project management, software version control, etc., which will benefit BPM without the cost and implementation constraints.

3.6 BUSINESS PROCESSES AND INNOVATION AT CTEXT

3.6.1 Introduction

The applicability of the BPM and innovation challenges to CText's development division was investigated through interviews with two members each of CText's management, support and development staff. It is often difficult to draw clear lines between the various divisions, since from a project leader's point of view, all employees are potential team members and anybody can be recruited as a team member for a particular project. This means that at times the resource manager can assist with problem solving while a programmer might set up his own hardware for testing purposes, or the head of CText can be a member of a project team of which a programmer is the leader.

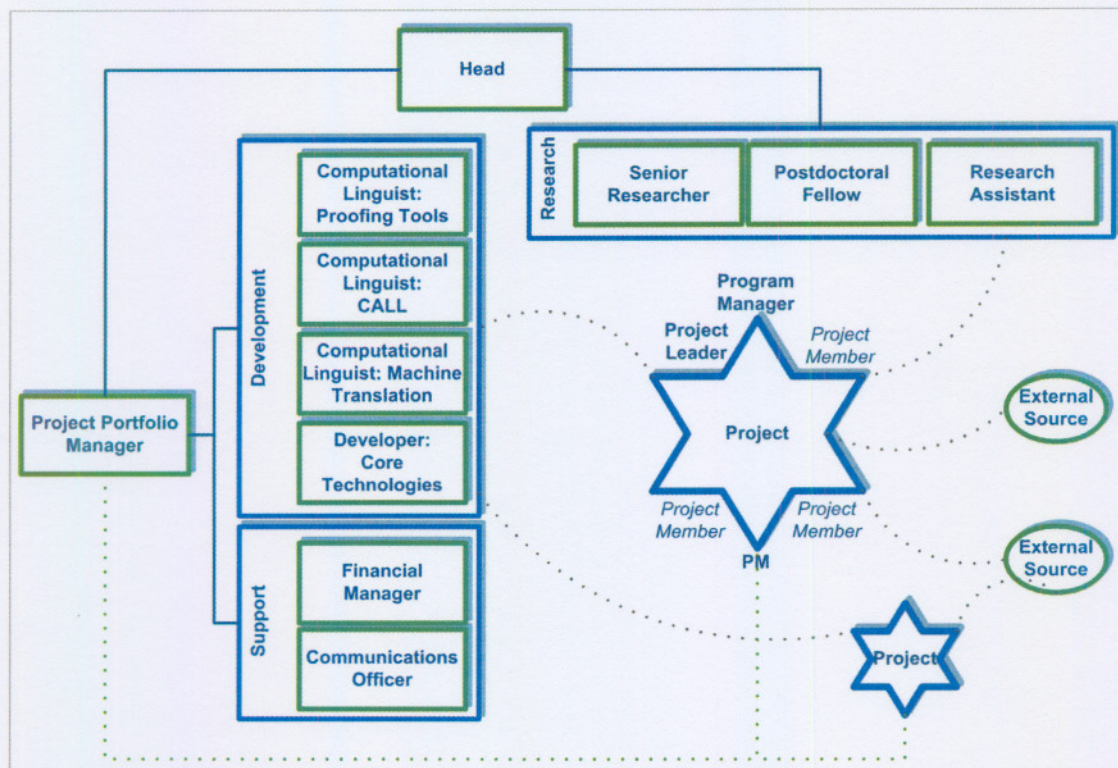


Figure 4: CText's organisational structure
(Adapted from CText: 2006:7)

Management's responsibilities are on average split equally between development and support. With these vague distinctions in mind, the respondents' involvement in software development can be depicted broadly as follows:

Management:

- **Head:** Project owner of most projects. He writes project proposals, secures contracts and funding, heads project planning and is involved on an ad hoc basis in various development phases, such as specification, design and beta evaluation.
- **Manager:** Project manager on most projects, to manage budgets, time schedules and quality, as well as distribution and customer support.

Support staff:

- **Communications officer:** Events management (workshops and meetings), product promotion and customer support, graphical user interface and graphic design co-ordination.
- **Resource manager:** Financial administration, assistance to developers with regard

to computer and software needs. Involved on an ad hoc basis in some development activities, such as testing.

Developers

- Two computational linguists who are directly involved in all software development phases and hold the responsibility for a number of them. As of 2007, they become the programme managers for proofing tools and CALL respectively.

3.6.2 Survey results

3.6.2.1 Administrative barriers

- **Meetings**

There is a clear difference in opinion between the developers and the rest of the respondents regarding meetings. While the support staff and the project manager take advantage of staff meetings for gathering the information they need to perform their work, the developers feel that these meetings is a waste of their time. The developers object to the notion of staff meetings; they feel that they generally have little to add to these meetings, as they have to report the same development activity week after week for the duration of a project. According to one developer, 90% of these meetings are in his opinion unnecessary. Both are confident that they keep abreast of current organisational issues through informal conversations.

All the support staff and management respondents say that regular meetings serve to provide an update of what each person is involved in, the status of projects and other information which they need to be aware of in order to perform their own project related tasks. One of them notes that since the abolishment of weekly staff meetings in late 2005, she often has the feeling that only employees who take smoke breaks together know what is going on, since they also use their smoking as an opportunity to share information about their ideas, current work and problems. However, another respondent, also a smoker, expresses her dismay with the lack of regular meetings as well, complaining that certain types of information which she values greatly has been lost, ranging from accounts by the head of his current plans and activities with regard to a new project he has announced some time ago, to such information as colleagues' study and leave plans.

- **Planning**

On this topic as well there is a difference in opinion. The developers are only interested in planning which is directly related to their work, while management and support staff feel that more is better. The developers agree that the following types of planning activities are valuable to their development work:

- . In-depth pre-project planning workshops with all the team members add considerable value to development.
- . Short meetings to discuss issues as they arise.
- . They believe that it is crucial to communicate with stakeholders before engaging in any development task.
- . The team commends the practice of a colleague from another university, whose department hosts regular colloquiums where presentations of colleagues' current work are given. These presentations are relevant to the work of all those who are attending, with the result that they can learn from one another.

In contrast, they get frustrated with seminars where they have to listen to introductory presentations of project-related topics for the sake of new team members who are not familiar with the field or type of project.

The other respondents added the following:

- . The head and project manager says that she would not change any of the current planning practices, because they are essential to everyone's understanding of the scope of a project and every team member's role.
- . The support staff indicate that they would appreciate to be involved in projects much earlier. They often only learn about a project when they are tasked with organising a workshop or purchasing project equipment, which they then have to do without any prior knowledge of the parties involved, the objective of the project or workshop, etc. The communications officer feels that for her to do a good job of organising a workshop, designing a product cover or promoting a product, she needs to follow the early history of a project more closely than she currently does.

▪ Documentation

Since the start of 2006, management has enforced basic policies for general document management and product development. Interestingly enough, the development team has no complaints about the additional administrative effort this has created for them. They are satisfied with their current repertoire of procedure documents and technical reports, and believe that it has contributed significantly to their development work.

One developer, however, indicates that he will seriously consider leaving if CText should ever decide to become ISO-compliant, due to the commenting and versioning control requirements pertaining to the code. According to him it is not feasible and will simply slow down development. The team believes that such measures will impede on their progress.

There are mixed feelings about documentation within the support function. Both respondents understand the need for procedure documents and that they have experienced the advantages of them in their own work. However, there are some aspect with which they are displeased:

- . One respondent is specifically dissatisfied with the document naming policy; she simply detests the requirement that she has to name her documents in a certain way and says that it has not in any way solved her persistent problem with missing e-documents. As far as work content and procedures are concerned, she has a preference for verbal transfer; she would have liked an oral introduction when she started her work and she would prefer to one day hand over her work to her successor in a conversational meeting.
- . The other respondent dislikes the level of detail she has to include in her procedure documents, since she is obliged to write down even things which she considers to be common sense or universal knowledge.

The project manager is satisfied with the progress achieved with regard to project documentation within the last year. A number of projects have a near-complete set of documents which are updated with each new visitation of a project. There is admittedly still a long way to go; teams must continuously be reminded to finish

documentation on time and there are still gaps in the documents themselves. The head also notes that the naming structure is not followed accurately by some staff.

- **Standard University procedures**

One management respondent explains that tasks such as staff appointments and leave arrangements, and the long process which must be followed before a contract can be signed is a burden and it wastes time which could otherwise have been used for creative activities.

- **Record keeping and document management**

The record keeping is in a poor state, as the following examples show:

- . Records relating to older projects are difficult to find or non-existing. There is also little documented information available for the management of new projects that are related to previously completed projects: in many ways, the same devices have to be re-invented over and over again for each new project, unless the team can rely on the memory of team members who worked on the older projects.
- . Many important documentation regarding projects are kept on backup CDs and DVDs along with project data, with the result that they cannot be easily consulted.
- . The CDs and DVDs themselves are not easy to search. There is one storage medium with a limited search function that searches through all its CDs, provided the person performing the search has a good idea of the file or folder name. Of the rest of the storage disks, a third carries useless labels which make finding a document or an application installation disk quite difficult.
- . The filing room contains around 30 thick files on finances, negotiations, contracts, staff information, project information, etc. In a number of files, documents are filed in no logical order, while related information is often spread over numerous files. The result is that the users of these files waste a lot of time looking for documents.
- . There is little security around storage media and files. The resource manager does make dual backups of electronic data and has a system for the booking

out of CDs, but everything, including files containing sensitive information, is housed in open offices where anyone has access to them.

3.6.2.2 Project management

The developers agree that project management is an aspect which makes a positive contribution towards their work. The fact that they can communicate their needs and problems with a project manager who then assists them by helping to provide in those needs or by taking care of the logistical aspects of the problem solving process, is viewed as very helpful, as it allows them to remain concentrated on the development task. They have no noteworthy problems with project management practices at CText and are generally thankful for the freedom it gives them to focus on the nitty-gritty aspects of programming.

One respondent from the support function shares this sentiment; she is glad that she is spared the responsibility to make certain decisions relating to project plans. The other respondent could not comment on this, as she has had very limited exposure to project management.

The head believes that project management frees the real innovators to concentrate on their work and to be creative. Despite the risk to innovation posed by the nature of project management to manage a project stepwise, his opinion is that by posing all the right questions to the team, the project manager can tick off steps on her/his checklist, while actually providing the team the opportunity to come up with creative answers.

The project manager relates a problem with the documentation of the progression of projects. Projects start with impressive Gantt charts which become outdated the moment the work commences. When a project turns chaotic, all energy is spent on solving issues such as bottlenecks, non-compliance, etc. and none on documenting the details of their occurrence and solutions. This type of information can be valuable input for future projects, but unfortunately it is often lost or forgotten at the end of a project. This means that CText runs the risk of repeating the same mistakes in future.

3.6.2.3 Group dynamics

CText employees are a tight-knit group who enjoys each other's company away from the office on a social level too. Asked whether this cohesion can lead team members to keep quiet when they witness something with which they do not agree, all respondents disagree

strongly. Everyone answers that in their opinion the relationships are such that individuals have the courage to be honest, and to question and criticise ideas and work when they view it necessary. According to one respondent, the group dynamics are especially conducive to creativity and that the culture of critical thought is well-established. She trusts the group to warn her if she comes up with a bad idea.

One respondent, however, says that even though teamwork invokes creativity, he thinks that it holds the risk that the group can develop what he calls a herd mentality, and that he has observed some warning signs of this. For instance, the teams are solving many HLT problems through machine learning, without considering the available alternatives. He also feels that he contributes more towards creating creative energy than drawing from it.

Two respondents admit that in some instances they themselves are hesitant to disagree with team members. In one case this is because of a conflict-avoiding personality; in the other case the reason is that the respondent considers the team members as more knowledgeable and therefore thinks that she is either wrong, or that she will not be heard.

3.6.2.4 Control

In all interviews, respondents report that find it difficult to envisage how a BPMS would impact on their work. 50%, however, do not like the idea of systemic intervention. The developers say that in a development project they prefer to be led in broad and unobtrusive ways. They know that a project manager has to remind them of deliverables and their deadlines, but thereafter they expect her/him to retreat and let them come to those deliverables in whichever way they choose. They question the ability of systemic control to accept that developers sometimes need to ignore deadlines in order to deliver a better solution. In their opinion, good project management understands this. If a BPMS were to communicate with them more than once or twice daily, they predict that they would become quite frustrated.

One respondent from the support function thinks that a BPMS sounds bureaucratic and that working under such a setup does not sound pleasant. She would not like to be assigned tasks by a system, nor would she like to log her progress daily. The other respondent from this group, as well as one management respondent, are much more positive; they predict that systemic reminders will make life more convenient and provide peace of mind. The one respondent foresees more available time to dedicate to more creative work

until the next prompt for new action, while the other refers to the possibility that a BPMS might lead to stagnation by taking over the responsibility for deciding what should happen next.

The remaining respondent is often confronted with issues of which she does not know the process rules by heart, for example in retail (order requests, complaints, technical enquiries). In this sense a BPMS could be helpful, since it would guide action through process rules and minimise the risk for the incorrect course of action. She is, however, uncertain whether a BPMS could indeed deal with these matters effectively.

3.6.2.5 Errors

When prompted for commentary on their experience with errors, all but one respondent become cautious. The developers engage in light blame shifting with regard to specific examples of product errors, while two other respondents note that they are very risk averse. Only when they understand that the purpose of this discussion is to investigate the possible advantages of errors, everyone agrees that they learn from their mistakes and that errors generally have positive outcomes for future work. To illustrate, the developers continuously adapt test cases to close the loopholes through which errors slip.

Management expresses the view that the idea of a system which limits errors is attractive, since there are many issues surrounding software development in need of improvement, such as version control, product information, etc. One respondent says knowing that a BPMS will make provision for thorough testing and validation, he can perform extreme development experiments without fear that disastrous code will ever leave CText.

One respondent from the support function remarks that she hates it when a system attempts to eliminate human error. Having a strong sense of duty, she would much rather work in a protective environment where she could apologise and repair errors when they occurred, than to work with a system that prevented errors but which made her daily life unpleasant.

Her colleague makes the interesting remark that a system should allow errors as a way to let users learn. For example, as a user with restricted access, she would become frustrated if she were unable to access something or receive error messages without explanation. She needs the freedom to explore the outcomes of a decision, rather than to be de-

nied access without explanation. From the discussion that follows, it is revealed that she would enjoy acting as system administrator; that this would serve as creative outlet for her. Her current interaction with the University's Oracle system serves as an example: during the current deployment phase, users temporarily have unrestricted access to a wide range of features, and she has familiarised herself with the system and alternative ways to access customised information which serves her much better than the options provided in her user guide.

To add to the above topic of curiosity, 50% of the respondents agree that they feel constantly chased by deadlines and that they consequently do not feel that they have enough time to spend on pet hobbies and to investigate topics that interest them. One developer in particular expresses the wish to have more time for such activities, and to meet with different people away from the office for discussions on topics of interest. Having read of innovation nurturing at companies like Google, he is fascinated by their X% playtime rules and rewards for innovations that are adopted into products. He generally feels that his own investigation needs (which he sadly calls "unimportant") are always moved back in favour of tasks with deadlines.

3.7 SUMMARY AND CONCLUSION

This chapter probed creativity and innovation as stimulants for entrepreneurial activity, and the way in which these can be supported or hampered by BPM. A lack of specific literature necessitated a creative approach towards finding answers to the BPM-ERP innovation issue in particular and the suitability of BPM in an intrapreneurial organisation in general, based on deductions from relative attributes.

The literature review revealed that there are many administrative barriers that add to the weight on intrapreneurs' shoulders. Concerns were raised about the function of BPMS to control every step of the process and eliminate errors. Entrepreneurs do not like to be controlled, and trial and error is a respected path to innovation. Other potential threats identified were excessive group cohesion, and practicalities, such as the high cost and complexity to implement, which put most BPM options out of reach of small organisations. One source specifically linked BPM with ERP with regard to shared disadvantages, thereby strengthening suspicions about the BPM-ERP innovation analogy.

Creating a change culture in which employees learn to adapt to changing circumstances and processes fast, was identified as one of the most important preconditions for BPM specifically, and maintaining competitive advantage in general. By incorporating all aspects of innovation in the corporate culture and developing learning networks, an organisation can purposefully protect and support its creative employees. Other ideas put forward for the stimulation of creativities were organised creative abrasion and socialisation.

Finally, some options for small organisations and intrapreneurs were discussed, such as incremental BPM deployment, small and Open Source BPM tools and working according to the BPM philosophy, with or without IT tools. It was suggested that until such point where the cost of maintaining a manual system becomes higher than that of an automated system, even old-fashioned methods can be employed in the philosophical approach to BPM. Business process awareness alone was argued to deliver various advantages such as improvements to operations and better understanding between the various business functions.

Interviews with six employees from CText revealed that the developers and support staff have quite different views and needs regarding administrative functions such as meetings and documentation and consequently are satisfied and frustrated with different aspects thereof. Everyone is pleased with project management, in that the fact that somebody fulfils that role frees up the other team members to be creative. Only the project manager noted some administrative problems with the process, including project documentation and data management. Group cohesion was found to contribute to a healthy creative environment; however, one respondent voiced concerns about the development of a herd mentality. Most of the respondents were sceptical about the effect of a BPMS on their free movement and declared that they would not feel comfortable receiving prompts and reminders from a system instead of a project manager. One respondent recognised the creative protection offered by a BPMS, which would ensure that no development experiments leave CText before having gone through various inspection phases.

Employees were found to draw their creative energy from various sources. Some find it in the work itself, others through interaction with peers. One respondent noted that she could find it as system administrator of a BPMS. The general consensus is that there is too little time to be creative; that time constraints limit them to known methods.

The conclusion drawn from this chapter is that a BPMS might not be suitable for an intrapreneurial organisation. This conclusion is based on two reasons:

- The comparison of the characteristics of the typical corporate organisation and an intrapreneurial organisation (c. Section 3.4.1), and the goal of BPMS to facilitate the type of efficiencies that are desirable in the typical corporate setting, creates the impression that it is not suitable for the “anti-corporate” intrapreneurial organisation.
- The interviews with CText employees uncovered considerable scepticism towards systemic control, in favour of human guidance.

4 BPM AND SOFTWARE DEVELOPMENT: CHALLENGES AND SOLUTIONS

4.1 INTRODUCTION

Booch (2001:119) speculates that software may be the most important industry in the world. It affects almost all aspects of modern life. Unfortunately, a literature review on software development problems show that the domain as a whole has been plagued by crippling problems for as long as the industry has existed. According to Batista and Figueiredo (2000), most software products suffer from quality and reliability shortcomings. A recent Standish survey report supports their view with the following revelations: 18% of all projects fail, while 53% run late, over budget, and/or with less than the required features and functions (SGI, 2004:2).

Software project failure affects everyone; it occurs all over the world to large and small organisations alike, as to commercial, non-profit and government organisations, and the costs of these failures run billions annually (Charette, 2005). Most importantly, project failure can cost an organisation its reputation and endanger its future existence (Charette, 2005).

This chapter looks at process-related problem areas in software development, and investigates the role of BPM within this field. There is a variety of reasons for software development project failures, such as the project not supporting an organisation's strategic goals (Yardley, 2002:24), or stakeholder politics (Charette, 2005). Such problems, however, fall outside the scope of this study. This chapter investigates software development as a development process, and looks at the challenges and solutions in the industry with regard to this process. Problems that occur as a result of the skill or motivation of an intrapreneurial organisation, the climate in which it operates and the business case for the software products it selects to develop, are not taken into account for the duration of this chapter.

The investigation commences with an overview of process-related problems within software development, followed by a detailed motivation to make the case for BPM as an answer to these problems. Thereafter the investigation zooms in on specific challenges that BPM would face if applied to software development, and alternative solutions to software development process problems. The ultimate goal is thus to determine whether software development processes are manageable as business processes through BPM.

4.2 SOFTWARE DEVELOPMENT CHALLENGES

4.2.1 Process flaws

Process-related issues constitute a considerable part of software development problems. Yardley (2002:303) distinguishes between three types of factors influencing project failure: technical, human and process. Elimination of non-process factors produces the following list:

- Poor technical design
- Absence of methodologies for:
 - . Project management
 - . Systems development
 - . Benefits management
 - . Quality management
- Failure to identify and mitigate project risks
- Failure to manage requirements
- Insufficient testing.

The results of a study by Tiwana and Keil (2004) also point to methodology; they indicate that the use of an inappropriate methodology for a particular project is viewed as the most critical risk driver in software development. Inadequate customer involvement in the context of requirements management ranks second (Tiwana & Keil, 2004).

4.2.2 Confusion over the role of project management

The range of publications dedicated to the topic of project management for software development (Conway, 2001; Hughes & Cotterell, 2002) illustrates that project management is seen as vital to projects. The study referred to in the previous section on the project risk drivers has found the lack of formal project management practices to be the third most critical risk driver (Tiwana & Keil, 2004).

That said, Yardley (2002:10-25) warns that it must be kept in mind that project management is only one of the ingredients to success and that project failure is often mistaken for

project management failure. While a good project concept can fail when requirements or testing are managed poorly, project management cannot solve the problems created by a weak business case or the use of unproven technology. A project sets an objective, while project management is concerned with the planning and control to achieve that objective. They have different scopes and boundaries, and are measured by different criteria. Project management consequently does not necessarily guarantee project success.

4.2.3 Over-reliance on software tools

Projects occasionally fail because they are blindsided by the technology employed to develop the software (Booch, 1996:7). Yardley (2002:76) cautions against the notion of resorting to software tools to solve what is often really a business problem, stating that the reliance on such tools by development teams keen to overcome technical challenges can compromise an otherwise healthy project. He argues that such reliance leads to unrealistic expectations of what the tools can achieve, IT managers may view it as the solution to all problems, and the business has to change the way it operates to be compatible with it.

Furthermore, Yardley (2002:20-21) expresses concern for the importance attributed to the ability to use project management tools; while they may add control to existing good project management practice, they cannot make up for a lack of basic project management skills such as communication and leadership.

4.3 THE CASE FOR PROCESS MANAGEMENT IN SOFTWARE DEVELOPMENT

Section 4.2.1 illustrated that there is a sizeable element of process problems present in software development. There are some in favour of managing software development projects as processes. More than a decade ago, Davis and Leffingwell (1995:1) attributed the failure of projects to deliver expected features, and the tendency to overshoot their budgets and time schedules to a lack of process. They compared it to art, as being "*subject to bursts of creativity and individual genius rather than teamwork and engineering discipline*" (Davis & Leffingwell, 1995:1).

This situation has not improved since. Criticising the low priority given to processes within many development teams, Booch (2001:119) recently argued that the complexity of software called for management through higher levels of abstraction in the components cre-

ated, and elimination of the points of friction in the team and its processes.

4.3.1 Software development processes

Software development takes place through activities or processes: it consists of a range of processes that need to be completed within certain parameters and according to certain rules to produce a working product. Booch (1996:63-64) writes that software development methods consist in part of processes that are responsible for specifying how and when certain objects should be developed. The reader is referred to the processes listed in Section 4.2.1; arguably they all have controllable workflows. According to Yardley (2002:228), it is failure to integrate these processes into the software development life cycle that contributes to the high failure rate of projects.

Software development and the activities within can be represented graphically as progressing through the various phases. The following examples relating to software development demonstrate this point and establish the link to BPM.

4.3.1.1 The Software/System Development Life Cycle

In Conway's version of the system development life cycle (SDLC), the approach to development follows seven key phases (Conway, 2002:1-6). Each phase has the goal to produce one or more deliverables, for example, Phase 1 is concerned with communicating with the client about her/his requirements and the deliverable is a document that defines these requirements in terms that the client can understand (Figure 5 on the next page).

Movement to the next phase is not allowed until everything in the current phase has been verified and approved. There can, for instance, be no useful requirements analysis or solution design until the client is satisfied that her/his requirements have been accurately defined. There is opportunity for iteration, though, as progressing through phases may bring about a better understanding of earlier phases, which then have to be revisited.

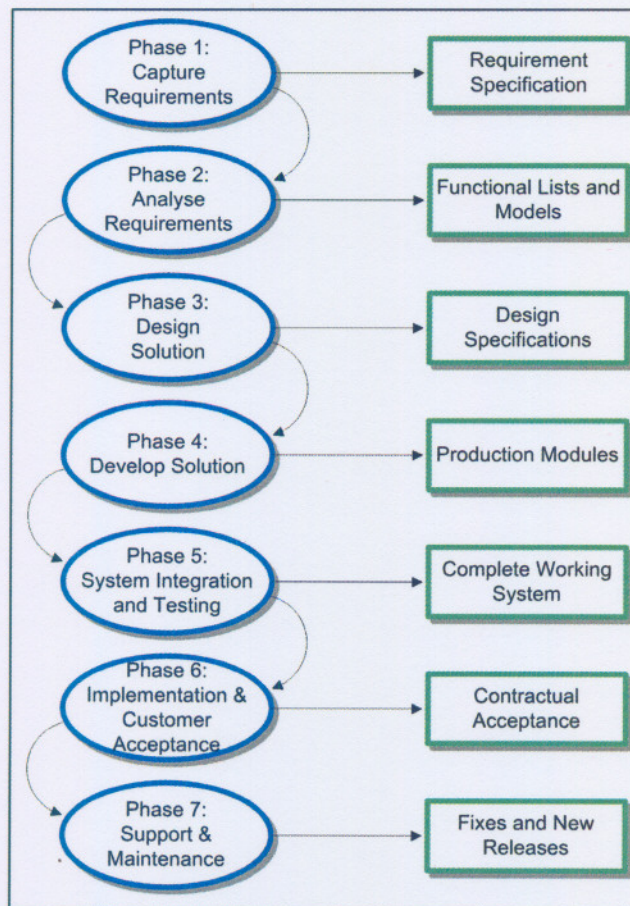


Figure 5: Conway's SDLC with deliverables
(Conway, 2002:4)

4.3.1.2 Rational Unified Processes

Another possible link between software development and BPM lies in Rational Unified Processes (RUP), a software development methodology developed by Jacobson, Booch and Rumbaugh (1999:96-97), who recognised a need for a process that integrated the many facets of software development and would:

- Provide guidance to the order of a team's activities;
- Direct the tasks of individual developers and the team as a whole;
- Specify what artefacts should be developed; and
- Offer criteria for monitoring and measuring a project's products and activities.

They believed that a well-defined and well-managed process was the key discriminator

between productive and unsuccessful projects, and the RUP was introduced as a generic use-case driven, architecture-centric, iterative and incremental process framework based on the Unified Modeling Language (Jacobson *et al.*, 1999:97).

RUP captures many modern software development best practices such as requirements management, iterative development, continuous quality verification and change control, and can provide organisations with a mature, rigorous and flexible software development process (Kruchten, 2001; Yardley, 2002:92).

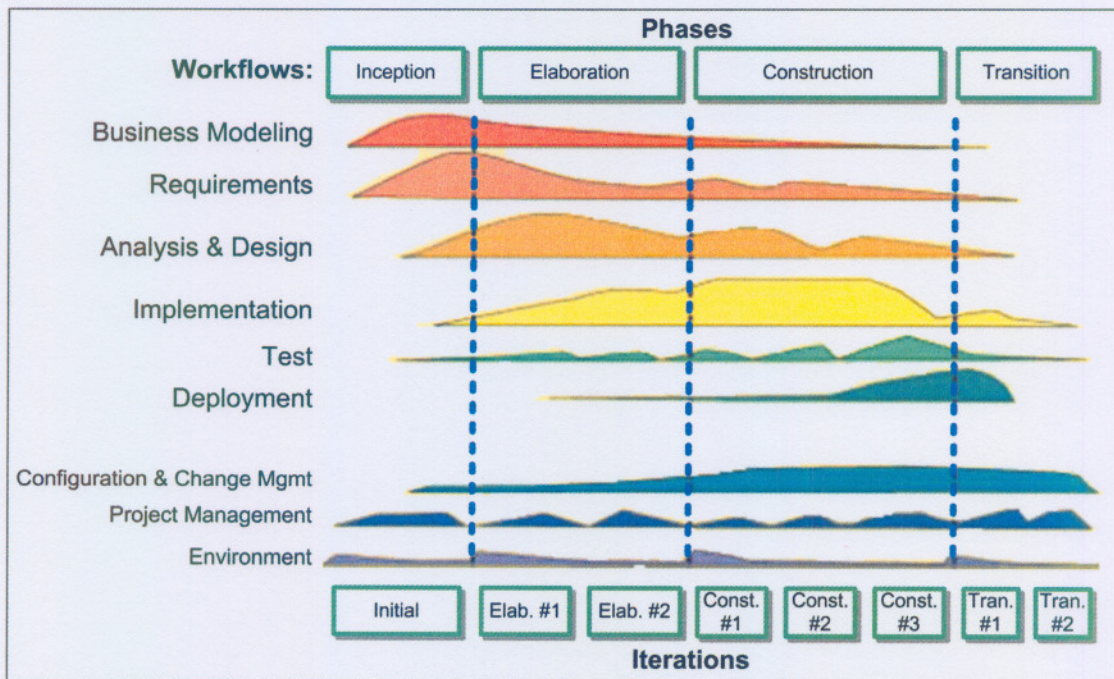


Figure 6: Dimensions of the RUP (Kruchten, 2001)

The following description of the RUP SDLC is taken from Yardley (2002:92-95):

In RUP, each project is divided into four phases: inception, elaboration, construction and transition. Each phase is marked by a number of iterations with resulting deliverables. The newest version of RUP² contains six core development process workflows (similar to the classic development methodologies) and three supporting workflows that are revisited with each iteration. During an iteration within the construction phase, for instance, all nine workflows are repeated, albeit with varying emphasis and intensity.

² The original consisted of five workflows: Requirements, Analysis, Design, Implementation and Test (Jacobson *et al.*, 1999:101).

4.3.1.3 Project management processes

Software project management has developed into a discipline within project management, to guide development through all the phases from concept to a quality software product (Conway, 2001:xix). Johnson (1999) highlights the process aspect of software development as critical to success, and argues that project management needs to manage these processes. He lists the four P's of project management as *People Performing Perfect Process*.

A number of methods and standards have been developed through which software project management can approach projects.

According to Lyneham-Brown (2006), project management and BPM have a great deal in common. This also comes to light when Yardley explains the key features of program management, which refers to the coordinated management of a portfolio of projects (Yardley, 2002:140-141). He notes centralised management of activities such as dependency tracking, resource, risk, issue and requirements management, and process tracking. The ability of BPMS to provide a single view on all work and an integrated approach to work and resource management (Pyke, 2006:23), makes it seem a potential candidate for software program management.

Another way to demonstrate the process nature of software development projects and their management, is by converting a project plan to a set of if-then statements:

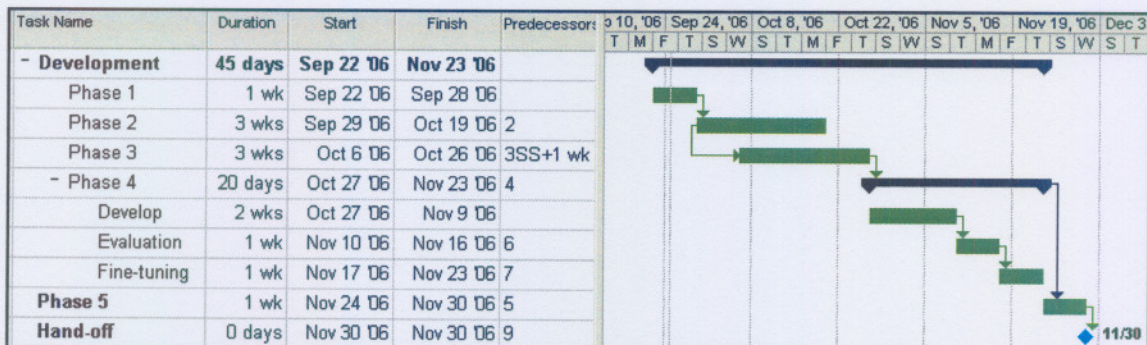


Figure 7: Example of a project Gantt chart

- If Phase 1 is completed, then initiate Phase 2.
- If Phase 2 is one week underway, initiate Phase 3.
- If Phase 4 meets requirements XYZ, continue with Phase 5, else review Phase 4.

In this fashion, it becomes clear that a project plan consists of preconditions, dependencies, rules for execution, and pre-defined outcomes, similar to business processes.

4.3.1.4 Step Wise

Step Wise is a project management framework for software development project planning developed by Hughes and Cotterell (2002:18-37).

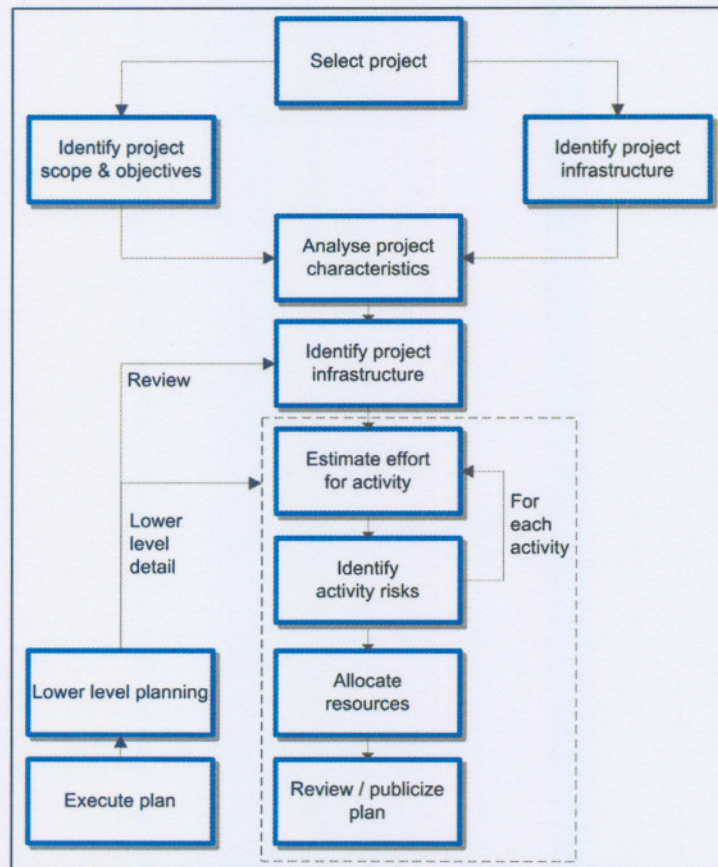


Figure 8: An overview of Step Wise (Hughes & Cotterell, 2002:21)

Each step contains a number of activities to be carried out, for example:

Analyse project characteristics:

- Distinguish the project as either objective or product driven;
- Analyse other project characteristics;
- Identify high-level project risks;

- Take into account user requirements concerning implementation;
- Select general life cycle approach; and
- Review overall resource estimates.

4.3.1.5 PRINCE2™

Projects in Controlled Environments is a project management method first developed by the British Central Computer and Telecommunications Agency³ in 1989 as a UK Government standard for IT project management (PRINCE2, 2006). PRINCE2 is the latest version which was designed to incorporate the requirements of existing users and to enhance the method towards a generic, best-practice approach for the management of all types of projects.

PRINCE2 dictates that a project should have an organised and controlled start, middle and end (PRINCE2, 2006). The method describes how a project is divided in manageable stages enabling efficient control of resources and regular progress monitoring throughout the project. Project planning is product-based and the project is driven by its business case.

At the heart of a PRINCE2 project is its process model:

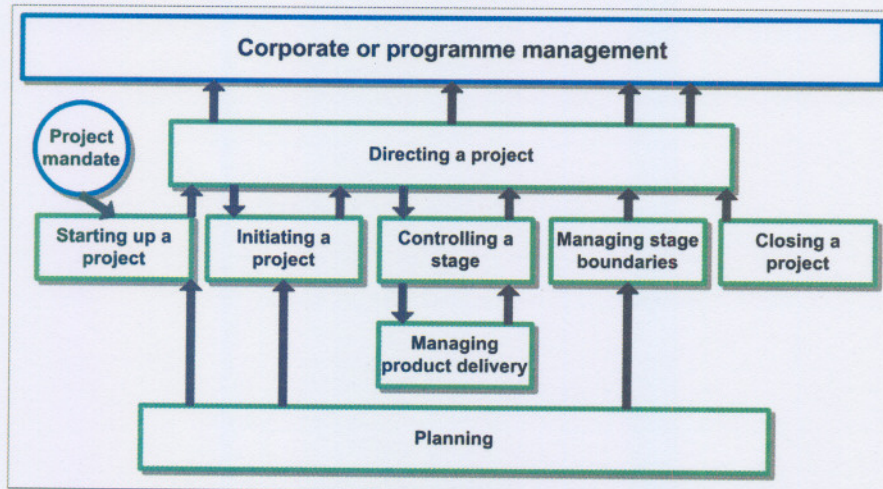


Figure 9: PRINCE2 process model (PRINCE2, 2006)

³ Now part of the Office of Government Commerce (OGC).

Each process consists of a set of activities or phases of its own, and starting up a project (SU), for instance, consists of the following activities:

- SU1: Appoint the executive and project manager;
- SU2: Design the project management team;
- SU3: Appoint the project management team;
- SU4: Prepare a project brief;
- SU5: Define project approach; and
- SU6: Plan an initiation state.

(McGuin, 2006?)

As a conclusion to this section, it should be noted that some modern approaches are much more iterative in nature, and steps need only be completed enough to move to the next step, as they can be revisited within a later iteration (Yardley, 2002:225). Therefore, the question remains as to whether or not a BPMS could successfully manage such iteration practices.

4.4 PITFALLS OF BPM IN SOFTWARE DEVELOPMENT

The available literature contributes little to the understanding of BPM problems in the domain of software development. However, the matter of the human element in management was cited from various sources and is therefore included in this subsection as a possible risk to the success of BPM within the software development domain. Furthermore, interviews with people involved in development within other organisations bring to light varied opinions on this issue, with interesting arguments, of which the negative statements have been included in this Section.

4.4.1 Managing people

Yardley (2002:81-82) explains that information systems methodologies often neglect to take into account the people involved. He explains that people are much more difficult to

manage, due to such issues as personality differences, skills and experience levels, attitudes, etc. Yet, even though managing people is key to the success of a project, many methodologies overlook the human, social and organisational aspects (Yardley, 2002:88). Miers (2005:26, 29) too observes that the people side often suffers in BPM deployment, due to ignorance of the organisational culture or the people interface, or neglect to help people understand where they fit in the workplace after implementation.

4.4.2 Disagreement over BPM in software development

Swart *et al.* (2006) claim that they have never seen BPM succeed in software development. According to them, problems relating to software development point to ailing methodologies and disciplines in the field of software development and that methodologies should therefore be the focus of an intervention plan.

Some of their arguments are substantiated by Metastorm director of product management Hudspeth (2006). To the question “Do you use your BPMS to manage the development of your BPMS?” he answers that while certain aspects such as bug fixing are managed by one of their BPMS functionalities, the management of the software development projects themselves are performed by project managers, who generally oversee these processes with *MS Project*.

While Prinsloo (2006), a business analyst at the North-West University, is extremely process aware and manages each phase of projects as sub processes within a larger process, she has never heard of a single BPMS that can take over this entire responsibility.

Tibco CEO Kristick (2006) too holds the opinion that software development is not a good fit for BPM. In his motivation, he refers to the McKinsey differentiation of work types:

- Transformational (such as the extraction of raw materials);
- Transactional (clerical); and
- Tacit (collaborative, complex problem solving activities).

(Beardsley *et al.*, 2006).

According to Kristick, BPM is very well suited for transactional work and automating these processes. In contrast, given the unstructured and collaborative nature of workers with tacit skills (such as programmers), this remains a challenged field for BPM (Kristick, 2006).

Muller (2006), a programmer at the North-West University, notes that some development projects are marked by such high levels of uncertainty, that planning becomes very complicated and processes close to impossible.

Dos Santos (2006) initially seems to hold the opinion that BPMS can be imposed on software development, and he notes that the successful software houses have them in place, of varying sophistication. However, when he provides the example of the Open Source community's BPMS as a set of agreed principles, policies and procedures to ensure the quality, reliability and predictability criteria, it becomes clear that his understanding of BPMS is much wider and representative of the holistic approach towards BPM (c. Section 3.5.2.2).

4.5 SOFTWARE DEVELOPMENT SOLUTIONS

4.5.1 BPM solutions for software development

Following the arguments against BPM intervention in software development, it should come as no surprise that an Internet search for BPM solutions for software development leads nowhere, except to provide information to systems and software developers with regard to BPMS development and deployment. The rare exceptions found are:

- IBM, who has recognised the need for what it calls “governance” of the business process of software and systems development in a globalised environment characterised by regulatory compliance, open computing and modular systems (IBM, 2006:6). Governance is defined as (1) establishing chains of responsibility, authority and communication, and (2) establishing measurement, policy and control mechanisms (IBM, 2006:3). IBM labels this governance of the business process of software and systems development “business driven development” (IBM, 2006:2).
- Singularity helps its software industry customers leverage process-centric software engineering approaches through BPM software and methodologies (Singularity, 2006).

A recent Forrester Wave evaluation report (Schwaber *et al.*, 2005) challenges the validity of the assumption that BPMS do not exist for software development. The report looks at process-centric SCM solutions, which are SCM tools that also manage the process by

which changes are made and thereby help manage the SDLC. Seven leading tools in this field are evaluated, and include tools by IBM, Borland and Microsoft. Characteristics evaluated include process management, versioning and configuration management capabilities.

4.5.2 Methodologies

Methodologies are a recurring theme in software development literature and much is written on the topic by experienced software developers. This has resulted in an extensive range of methodologies to assist software developers and project managers in the development of better software. Developers and managers are well advised to take advantage of the available literature, to familiarise themselves with various methodologies and gain a thorough understanding of the respective strengths and weaknesses before adopting any one for a particular project.

It is not possible to attempt a representative listing here, but some interesting and noteworthy examples are provided in this section.

4.5.2.1 Agile development methodologies

The focus on the management of process and structure can give way to a number of risks, such as:

- Creative and independent software developers themselves are bound to feel threatened.
- Software development can become tediously slow, which is as good as a death sentence for software development organisations.

These risks can be mitigated through the adoption of various methodologies and approaches for certain situations. At a very basic level one can look at practices within the process, such as documentation. Ambler (2006), for example, expresses concern about the effect of comprehensive documentation development and maintenance on software development, and recommends that documentation be kept lean and effective, so that more energy can be invested in the development work itself.

Booch (1996:158) notes that the individual programmers and their interests and rhythms are often ignored by the management functions of control and risk management. He ex-

plains that within a software development project (the macro process) individual programmers can find their creative outlet in the micro processes, which are opportunistic and pragmatic cycles of discovery, invention and implementation (Booch, 1996:159-161). Each project, its domain and the organisational culture must then find the right balance between these macro and micro processes.

A number of methodologies have been developed to ensure responsiveness and faster delivery, such as Rapid Application Development (RAD), the Dynamic Systems Development Method (DSDM) and Extreme Programming (XP).

The RAD approach emphasises the swift delivery of prototypes for user evaluation (Hughes & Cotterell, 2002:63). The DSDM is a collaborative approach that promotes user involvement throughout the development process and it is becoming the standard methodology for RAD (Yardley, 2002:87). XP is a further development of the RAD and DSDM principles (Hughes & Cotterell, 2002:75) which prioritises requirements and then focuses on the 80% reach through 20% functionality principle (Yardley, 2002:65) to deliver prototypes.

In XP, applications are written in increments of working software that take no more than a few weeks to complete (Hughes & Cotterell, 2002:75). XP aspires to remove artificial barriers to development productivity so that, for instance, changed user requirements can be accommodated at any time (Hughes & Cotterell, 2002:75). The thread holding this seemingly chaotic method together, is a special emphasis on communication, testing and iterative processes. Test cases and expected results are devised before design starts, to help the programmer and client agree on the exact requirements (Hughes & Cotterell, 2002:75). Schach (2005:53) cautions that while XP has been successfully used on some small projects, there is reason to be cautious of this approach and that it still needs to prove itself. As was noted in the introduction to the discussion of methodologies above, users of any methodology must be aware that all methodologies have strengths and challenges.

4.5.2.2 PRINCE2

This methodology for project management, which was discussed in Section 4.3.1.5, is widely used for managing software development projects.

4.5.2.3 The Rational Unified Process

The RUP was discussed under Section 4.3.1.2 for its workflow view on the SDLC. There is, however, disagreement regarding the applicability to small-scale software; according to Schach (2005:68), some features of RUP are inapplicable while Kohrell and Wonch (2005) maintain that with some customisation small projects can benefit significantly from adherence to RUP.

4.5.2.4 A Methodology audit and the Capability Maturity Model

The advice of Swart *et al.* (2006) to struggling software developers is an audit of their existing procedures and to take corrective measures to adopt and adhere to established software development methodologies. Considering the number of times methodologies have been mentioned thus far in this Chapter, an audit would seem the logical first step towards software development improvement.

The Capability Maturity Model (CMM) should be an effective framework for implementing this recommendation. The CMM was developed by the Software Engineering Institute at Carnegie Mellon University in the USA, having recognised that project success often vested in particular individuals rather than organisational software processes, which provided no basis for long-term productivity and quality improvement (Paulk *et al.*, 1993:1). The CMM guides software organisations to gain control of their development and maintenance processes and to evolve towards a culture of software engineering and management excellence (Paulk *et al.*, 1993:5). For each higher maturity level there are a number of key process areas an organisation must address to improve its software processes (Paulk *et al.*, 1993:30).

The CMM consists of five maturity levels:

- **Initial:** An unstable environment with reaction-driven commitment systems. The software process is ad hoc and occasionally chaotic. Few processes are defined and success depends on individual effort. Schedule and cost targets are typically overrun.
- **Repeatable⁴:** Basic project management processes are established for stable planning and tracking of cost, schedule and functionality. Process discipline is in

⁴ The reader is referred to Appendix B for more information on the key process areas of Level 2.

place to repeat earlier successes on projects with similar applications. Plans based on past performance are more realistic than in Level 1 organisations.

- **Defined:** Software processes for management and development are documented, standardised and integrated into a standard software process for the organisation. All projects use an approved, tailored version of this process and performance continues to improve.
- **Managed:** Detailed measures of the software process and product quality are collected; they are quantitatively understood and controlled.
- **Optimising:** Continuous process improvement through quantitative feedback from the process and piloting innovative ideas and technologies.

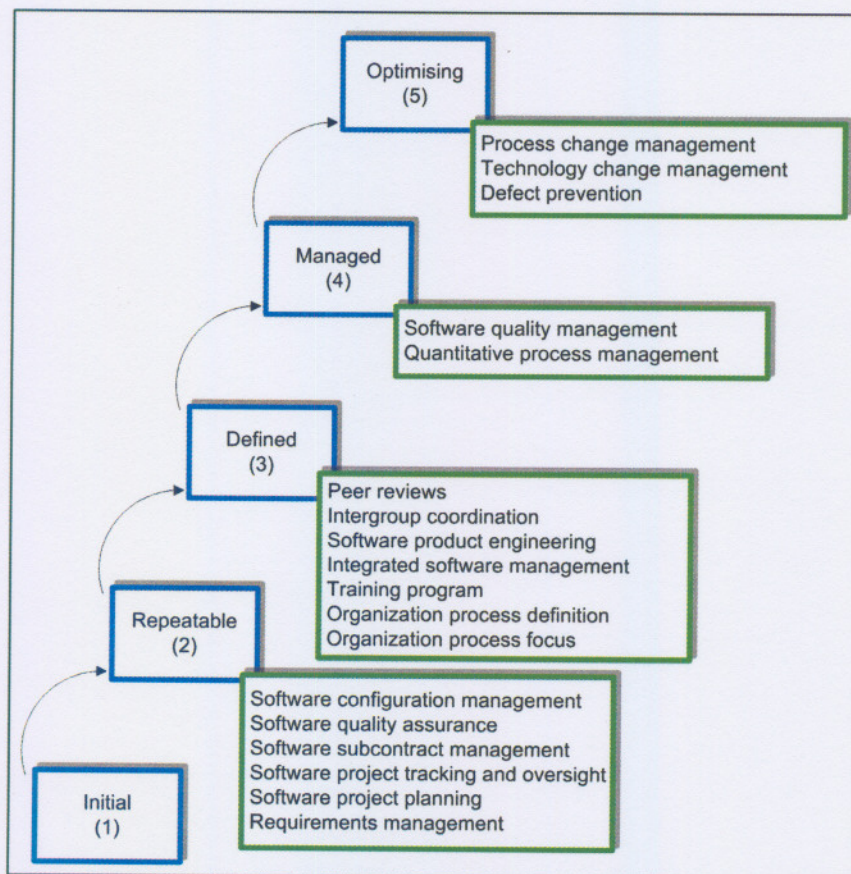


Figure 10: The Levels of Software Process Maturity and key process areas (Paulk *et al.*, 1993:31; Yardley, 2002:240)

Davis and Leffingwell consider requirements management the first step to mature development practices and management and ultimately improve product quality and fit with customer needs (1995:13).

The CMM was originally developed for very large teams; however, Batista and De Figueiredo (2000) have successfully proven its applicability to a micro team. The test team had fewer than 10 members and at project start found itself at the first level of maturity. It shared some characteristics with CText: poor version control, no engineering procedures, last-minute changes and quality problems. In order to save costs, the approach to CMM was simplified and very pragmatic. After a period of one year, a significant quality improvement of software was evident. Time and costs relating to development activities had increased, but conversely decreased for non-development activities.

It should be interesting to note as a final remark that a PRINCE2 Maturity Model (P2MM) has been derived from the CMM to enable organisations to gauge their maturity in the use of PRINCE2 (PRINCE2, 2004:3-4).

4.5.2.5 CASE tools

Computer-Aided Software Engineering (CASE) tools provide support for software development activities (SEI, 2004). These automated tools are generally acknowledged to provide the following benefits:

- Software development productivity improvements;
- High-quality software outputs;
- Reductions in maintenance costs; and
- Systems performance improvement.

(Yardley, 2002:245).

Yardley's recommendation is that a RAD methodology, supported by appropriate CASE tools and a software development methodology such as RUP can introduce a level of software process discipline and progress the maturity of an organisation to CMM Level 2 or above (Yardley, 2002:245).

4.6 BPM AND SOFTWARE DEVELOPMENT AT CTEXT

4.6.1 Introduction

The reader is referred to Section 3.6.1 for more detail on the respondents interviewed. The support staff are marginal figures in the software development process itself, who are included in some aspects thereof as their assistance is needed. Examples are graphic design, user help file development and evaluation. Consequently they could not be interviewed in depth on the topic of BPM and software development, however, whatever contribution they could make to this topic with regard to past experiences was recorded and has been included in this section. For insight into their opinion towards BPM and how it might impact their work, including their involvement with software development, the reader is referred to section 3.6.

It has been stated in section 3.6.1 that management is involved both in software development and a variety of support functions. This implies that management is knowledgeable on some, but not all aspects of software development and on some questions they were unable to provide complete answers.

4.6.2 Survey results

4.6.2.1 Specifications and requirements

The developers dislike the custom of getting full and detailed specifications before starting with programming; while they respect the importance thereof, they prefer rapid prototyping as a method of refining specifications. In their opinion this approach could work in both small and large projects; they specifically note the new generation CALL project CText is negotiating to develop, as being a good subject in which prototyping could help to capture requirements.

The project manager admits that in the past she was seldom part of the process and as a result did not explicitly manage the outcomes. She has only recently recognised that some problems with end products relate back to specifications and requirements flaws.

A review of the project documentation on specifications and requirements shows gross neglect; the majority of projects do not have these explicitly stated, while in some cases requirements are incomplete or vague. Specifications do appear in project and tender

proposals, but are in some cases not optimally translated into test cases and other evaluation criteria.

4.6.2.2 Risk management

All the respondents agree that risk management is very low on the agenda at CText. The programmers account for it indirectly through the allocation of additional time for certain phases in time estimations, while the project manager plays devil's advocate in order to ensure that the developers have taken all possible problems into account before taking a certain course of action.

Although there is a section on risk analysis in the project planning template, the project manager has only on very rare occasions made risk management an important part of a project and admits that there is much room for improvement in this division. Considering the scope of some of the projects to be embarked on in 2007, everyone agrees that risk management must become a priority. The document survey confirms the respondents' comments; generally risk assessments appear in project proposals (thus, they are performed by the head during the writing of proposals) and are rarely transferred to the project planning and management phases.

4.6.2.3 Testing

All the respondents agree that testing practices have improved significantly in 2006, and at the present take up considerable space within projects. According to the developers the test cases for current projects are extensive and regularly updated to increase reach and close loopholes.

In the opinion of the project manager, most of the development problems of the past originated from insufficient testing, and that advances in testing have resulted in improved product quality. She is satisfied that procedures and test cases are in place and are improved with every new product, and testing now takes central stage in product development. It remains a work in progress, with room for progress.

The document survey shows that a major recent CALL project has an evaluation procedure with an extensive set of test cases, which was originally developed in the second part of 2005 and has since been updated for each new product in the portfolio. The Afrikaans spelling checker is the only other project with a decent evaluation procedure, which it has borrowed from the Microsoft spelling checker project. It seems that evaluation procedures

receive sufficient attention in current projects, and that the commitment to improve this development function comes from within the development team.

4.6.2.4 Methodologies

Despite the developers' assertion that they use RAD methodologies, they cannot produce a clear description of the methodology. They have never studied it and basically view RAD as a process of iterative prototyping and doing whatever needs to be done "until the boss is happy" in order to complete the project.

The project manager has for the purpose of this investigation had to study a range of material on software development. She notes that as a result she has become aware of the variety of aspects which should enjoy prominent attention, rather than the accidental attention they receive at CText, and she believes that there are definitely some voids with regard to certain models and methodologies. Specific examples mentioned are software configuration management (SCM) and code commenting.

The head shares his opinion that the developers do not grasp the meaning of the various SDLC models and that if they did, their development work would improve, even if their work were to continue along the rapid prototyping route.

The project documentation survey reveals that some phases in the classic development models receive no more than unintentional attention, such as feasibility studies and user requirements. This can be attributed to the fact that the project manager has been unaware of these aspects and has consequently not driven projects to grant attention to them. Another possible reason is the fact that the South African market for HLT applications is still very small, and therefore product development at present cannot rely on the recommendations of feasibility studies and end user requirements.

Other issues that arose in the interviews include:

4.6.2.5 CMM

Interestingly enough, the description of the Initial Phase describes the ideal working environment to one of the developers, for small ad hoc projects. However, there is agreement among the developers that the Repeatable Phase would be much more conducive to development for projects in general. It is felt that CText has already made some good pro-

gress in moving towards this phase, through its procedure documentation in which a range of key tasks pertaining to individual projects have been documented.

The project manager is excited about the possibilities of CMM, that it can act as a vehicle through which processes and documentation can be put in place for software development.

4.6.2.6 Software configuration management

The developers are optimistic about the possible benefits through the adoption of an SCM software tool, and that this should be investigated. The project manager notes that she is not familiar with the current status of SCM and it worries her.

4.6.2.7 Programming conventions

Quite worrying with regard to sustainability is the fact that one of the programmers does not comment his code unless there is something significant about a piece of code.

It is also revealed that programmers often have to rewrite existing code originating from other programmers before they can make changes. Sometimes differences in programming style are small and contained within individual lines of code, but it can also be structural. The head programmers overcome this by reviewing and editing all pieces of code that are delivered by other programmers.

They object to the suggestion that programming conventions be agreed upon at project start, stating that it is not possible and such measures will result in negativity; one needs to accept and respect that all programmers work differently and that one should not try to change their way of thinking and working.

4.6.2.8 People

All the respondents consider the human element a crucial aspect, and therefore all are concerned about the effect of a BPMS on them.

While the developers appreciate receiving an e-mail reminder from a human project manager, they predict that they will probably become agitated if a system sends red flagged prompts to complete an overdue task. Some of the responses during the interviews highlight the fact that this team does not feel and function well when they do not have control

over their development tasks, for example if they have to solve a problem while being constantly reminded that the project is running late or that some requirement has not been fulfilled.

The project manager relates an incident where the head experimented with a web-based workflow application for small tasks, but nobody could or wanted to work with it. They ignored or responded late to the e-mail notifications and eventually the idea was abandoned⁵. She herself faces the daily challenge to determine the most appropriate communication of tasks to team members. She finds it most peculiar that a team of people with above-average computer literacy, who perform all their work on computers, seem to either dislike e-mail notifications or respond ineffectively. Some examples include:

- Recipients do not take cognisance of e-mail attachments or take in only half the message, even if it is a short message.
- Some do not respond to requests or tasks to inform her that these have been completed.
- They need to be reminded of tasks despite e-mail notifications.
- Some do not accept electronic appointments, because once they do, the appointment moves to their electronic diary and since they do not check their diaries, they will forget about the appointment.
- Nobody uses their e-mail task list function, rendering the possibility of electronic tasking useless.

She wonders how a team who does not make full use of a simple e-mail application, which is active on their computers all day, will react and adapt to a BPMS.

4.6.2.9 Project management

As was noted in Chapter 3, the team has appreciation for project management and perceives it to enable them to focus all energy on development. The developers prefer not to be troubled by non-programming responsibilities. As an illustration to this point, one de-

⁵ The application was not investigated beforehand, nor was it explained to the team – the initiator simply opened an account, set up a workflow and triggered its execution. There were some other issues which led to the abandonment, such as the fact that it was a demo version which only allowed for 4 users. Registering all CText's employees would have been quite costly and management was not yet convinced of its success, especially after the failure of the experiment. This incident was one of the factors leading to the initiation of this study.

veloper relates an incident in 2005 where a serious error slipped through⁶. At the time, management encouraged multifaceted job descriptions to give employees exposure to a wide range of tasks. This developer was put in charge of finalising the language content for the product packaging, and managing the process to the point of delivering the master CD. He could not give the usual attention to the code and consequently the error went through unnoticed.

When a developer has to engage in other tasks, for example in overall project planning as a programme manager, he welcomes the help of the project manager who gets the needed inputs from him, and then delivers drafts of budgets and time lines that he can then review and refine.

Finally, while developers find project management guidance helpful, they feel that they should retain the right to decide how to accomplish tasks, which might include missing deadlines or distancing them from other current CText events. When the project manager interferes incessantly during crunch time or when other employees disturb them with unrelated requests, it causes frustration.

The project manager notes that the capability of a BPMS to produce an audit trail is theoretically a good solution to her own recording flaws of critical project information. Also, considering the unique needs of software development, with specific reference to the process related problems, the challenge to incorporate previously neglected processes and improving existing processes is daunting and systemic assistance will possibly be needed to manage it all.

4.7 SUMMARY AND CONCLUSION

This Chapter has shown that software development is generally plagued with problems, of which process-related problems forms a significant part. Inappropriate development and project management methodologies and skills were noted as some of the most critical risk drivers. Through comparison with methodologies such as the software development life cycle (SDLC) and Rational Unified Processes development, and the Step Wise and PRINCE2 project management frameworks, a case was made for the management of

⁶ This is one of the very incidents which led management to commission an investigation into BPM.

software development processes through BPM. In addition, the importance of the process aspect of software development was cited by a number of sources as critical to success.

This deduction was, however, met with considerable criticism by knowledgeable respondents in the software development industry, who instead pointed to methodologies and an implied holistic BPM approach for improvement in software development.

The focus consequently shifted to methodologies and a number of options were discussed as candidates for improving software development processes, such as PRINCE2 and the Capability Maturity Model (CMM). The latter in particular seems to address CText's cost, quality and project management concerns, through the realisation of the Repeatable Phase of the CMM.

The interviews with CText's development team highlighted that the team is protective of its autonomy. They recognise that certain principles such as risk management, testing and project management are indispensable to software development, but they wish to retain their authority to decide how and when to meet these requirements. Judging by the procedure documents and evaluation principles in place for projects, the developers are responsible enough to maintain such authority.

The discussions did, however, reveal a possible sustainability risk in differing coding styles, a perceived lack of understanding of development methodologies, and a confirmation that risk management is virtually non-existent. It was also pointed out that despite the fact that all team members have their e-mail program open on their computers all day, not everyone is using it satisfactorily, as far as project-related tasking is concerned. The question therefore arises whether another systemic intervention would have any chance of succeeding in managing software development processes.

The conclusion drawn is that while software development management from end to end through a BPMS is not advisable, it is essential that the overall development process be managed more appropriately. There are aspects of the SDLC in need of more control, and many of these aspects can be managed through the use of software tools. Essentially, software development at CText needs to become more process aware and weak areas within processes must receive urgent management attention. In this sense the recommendation is for a holistic approach towards software development process management.

5 CONCLUSION AND RECOMMENDATION

5.1 SUMMARY

CTexT, an intrapreneurial software development organisation within the North-West University, has been experiencing a variety of problems relating to software development, in the areas of customer support management, versioning and configuration management, quality management and risk and project management. Cost reduction, quality assurance and effective project management of new projects are the primary motivators for granting urgent attention to this situation.

CTexT's management held the opinion that an electronic workflow or business process management system might hold the solution to its software development problems, and consequently issued its project manager with the task of investigating the matter and making a recommendation regarding the implementation of an appropriate system.

Since CText is an intrapreneurial organisation, initial introduction concerns regarding the possible adverse effect on innovation and resistance were expressed. Furthermore, there were implementation issues: the suitability of BPM in the domain of software development was questioned and it was asked whether a WfMS/BPMS would be the best solution to the current problems. These became the secondary objectives to the general research question:

Can the implementation of a WfMS/BPMS effectively solve CText's development problems and thereby improve its overall software development capacity?

The study was executed within the broad theoretic framework of BPM, with specific focus on its role in the software development division of an intrapreneurial software development organisation. Further demarcation was done with respect to the first two states of the BPM life cycle, namely planning and selection of the business processes for improvement.

Chapter 2 introduced the debate between workflow and BPM terminologies, where there is no general consensus on their exact meaning and application within the enterprise content management domain. A number of definitions from relevant literature followed for the purpose of arriving at a working definition for this study. It was decided to formulate such a definition from a business user perspective where the focus is on bottom-line results, such

as reduced risk and more effective business operations. From this perspective, the discussion about the relative domains, similarities and differences regarding workflow and BPM essentially became irrelevant and it was consequently decided to forthwith use only the term BPM for the remainder of this study to refer to the following:

A philosophical approach to organisation-wide management in which the focus is on the processes through which it operates, and in particular on the streamlining and optimisation of these processes, for which software solutions may be used. Exercised in the narrow sense of the word, the focus might be on internal processes only. In the broadest sense, BPM can facilitate work and communication flow between various databases within an organisation, and in addition extend beyond organisational borders to connect and involve other entities and their systems on which the organisation may be dependent for its operations.

A simplified technical perspective served to provide additional information to a non-technical audience, including a technology overview, the workflow engine and standards. A final section on implementation areas revealed that BPM is well-suited for routine and repetitive processes that draw upon information databases for certain inputs and outputs. There was inconclusive evidence about the application on non-routine processes, and very little available information on the application of BPM on the domain of software development. Only a few vendors were found to offer solutions to software developers.

The methodology applied to the investigation of the objectives was a combination of extensive literature reviews and interviews with knowledgeable respondents from other innovative and software organisations. The focus was on the disadvantages of BPM to innovation and software development, in order to determine the risk posed by BPM to these activities, and possible solutions to remedy or eliminate these risks. These findings were supplemented with an exploration of CText project documentation and semi-structured interviews with six employees to determine the relevance of the identified risks to CText.

Chapter 3 was dedicated to determining the risk posed by BPM to creativity and innovation at intrapreneurial organisations. It was stated that in entrepreneurial activity, creativity led to innovation while entrepreneurship drove the process from idea to harvesting. An investigation found that administrative barriers (systems, structures and policies), project man-

agement and group dynamics represented general organisational constraints, while control, no toleration for failure and other practical matters such as cost were specific BPM-related problems with regard to innovation and the reality of the intrapreneurial organisation.

Strategic management of innovation, development of a change culture, learning networks and socialisation were identified as possible solutions to the BPM-innovation tension, while incremental BPM deployment and the implementation of the BPM philosophy as a holistic approach towards process management were discussed as some of the options available to intrapreneurs bound by limited resources.

The relevance of the challenges and their solutions were finally tested at CText. It was found that the developers and support staff had quite different views and needs regarding administrative functions such as meetings and documentation and consequently were satisfied and frustrated with different aspects. Project management was said to free up the other team members to be creative, but the project manager noted some administrative problems with the process. Most respondents agreed that group cohesion contributed to a healthy creative environment; however, one respondent voiced concerns about the development of a herd mentality. Generally the respondents were sceptical about the effect of a BPMS on their free movement and declared that they would not feel comfortable receiving prompts and reminders from a system instead of a project manager. One respondent did recognise the creative protection offered by a BPMS, which would ensure that no development experiments left CText before having gone through various inspection phases. Finally, the interviews revealed that employees drew their creative energy from various sources, including from the work itself, interaction with peers or experiments with technology. The general consensus was, however, that there was too little time to be creative and that time constraints limited them to known methods.

In Chapter 4 the literature review pointed to a range of process-related issues in software development, which lay the foundation on which the case for BPM in software development could be built. Short introductions were given into the software/systems development life cycle (SDLC), Rational Unified Processes (RUP) and software project management, including the Step Wise and PRINCE2 project planning and management methods. These confirmed that software development consists of processes, which added weight to the hypothesis that a BPMS can be utilised to manage software development. However,

interviews with software developers and business analysts to a large extent led to the rejection of this hypothesis; given the complexity and uncertainty characteristics of software development and the tacit work type, the majority of respondents were rather negative about the suggestion to introduce a BPMS in this domain.

Throughout the investigation, the concept of methodologies continued to surface as a recurring theme. This resulted in a refocus on previously mentioned methods such as PRINCE2 and RUP, CASE tools and a short discussion on agile development approaches to prevent a loss of creativity and lengthened projects. The Capability Maturity Model (CMM) was presented as a framework for the improvement of software process maturity through five phases from unstable, reaction-driven environments to repeatable, continuously improved development processes. It was suggested that a rapid application development methodology, supported by appropriate CASE tools and a RUP methodology would facilitate growth to the second CMM phase, that is, the repeatable phase.

The interviews with CText's development team highlighted the team's protectiveness of their autonomy and their wish to retain their authority to decide how and when to meet the demands of requirements management, risk management, testing and project management. A review of project documentation indicated that recent projects have documents for requirements and specifications, evaluation procedures and post-project reports, but that there is very limited documentation for older projects and that it is very hard to locate data relating to those projects.

The discussions revealed a possible sustainability risk in differing coding styles, a perceived lack of real understanding of development methodologies, and a confirmation that risk management is virtually non-existent. The only real evidence of methodology in development at CText is the practice of rapid prototyping as part of requirements specification, and the testing procedures. The development process can be described as one of prototyping and iterative refinement to application functionalities, "until the boss is happy". This is reminiscent of Hughes and Cotterell's (2002:67) criticism of prototyping for the danger that it can become an excuse for a sloppy "hack it out and see what happens" approach.

It was also pointed out that despite the fact that all team members use their e-mail program throughout the day, not everyone is using it satisfactorily, as far as project-related

tasking is concerned. The question therefore arose whether another systemic intervention would have any chance of succeeding in managing software development processes.

5.2 RECOMMENDATION

This investigation into BPM has cast a number of important question marks over its application to a small intrapreneurial organisation and to the domain of software development. There are some very practical concerns, like cost, complexity and maintenance. Furthermore, CText employees have voiced their concern about the contribution of a BPMS to their work. Lastly and most importantly, the lack of available literature regarding BPM in software development, coupled with the testimonials from other software developers indicate that the concept of BPMS has not proven itself in software development. CText does not have the capacity to become a crusader in this field, and a BPMS will most probably cause more disruption than improvement to development work, with the result that it will not meet the most basic BPM goal of streamlining and optimising processes. Therefore, the proposal to invest in a BPMS should be temporarily abandoned. It might be reconsidered in the future for the streamlining of other operational processes within CText, but it would not be a good investment in the current context.

Nevertheless, the aforementioned concerns do not constitute enough reason to abandon BPM altogether. In fact, the author has come to the conclusion that a process consciousness has considerable potential to bring about streamlined work, smoother cooperation between team members and improved sustainability for projects to survive their teams.

The outcomes from this study show that while software development management from end to end through BPMS is not advisable, it is critical that the overall development process be managed more appropriately. There are aspects of the SDLC in need of more control, and many of these aspects can be managed through the use of software tools. Essentially, software development at CText needs to become more process-aware and weak areas in processes must receive urgent management attention. In this sense the recommendation is for a holistic approach towards software development process management.

The following specific suggestions are made in respect of preparing CText for this management approach:

- Should CText's management decide to accept the recommendations offered in this study, the decision must be a high-level strategic pronouncement. BPM and innovation should be placed on the **corporate agenda** from where they can receive support and resources for the achievement of their goals, and guide the achievement of related objectives. It is imperative that management not merely pay lip service to these objectives, but actively work towards enabling their achievement. For example, if innovation is identified as a strategic objective, management becomes responsible for implementing action steps to allocate and spend money on initiatives, grant playtime the status of a key performance indicator on employees' task agreements, etc. Thompson Jr. *et al.* (2005:18) have developed a strategy crafting and execution model to explain how this process works from vision to execution.
- The BPM initiative should be based on and guided by an acknowledged model for **motivation** to ensure optimal employee commitment towards the goals of the programme, such as the model of aligned-commitment by Coetsee (2002:27-41). Motivation is a powerful determinant of organisational success; unless employees understand the need to accommodate this change, are empowered to take responsibility for the process and are rewarded for achieving the objectives of BPM, they are not likely to be committed to give their full cooperation.
- **Change management** should become a priority to CText's management, not just to help the Centre cope with the change caused by the BPM initiative, but to enable the Centre to become truly change-able as a prerequisite to long-term BPM success. Jackson (2000:102) has developed a roadmap to help organisations achieve such dynamism, which might be one possible point of departure.
- Employees are bound to become confused with a number of perceived conflicts within the new programme: fixed processes, standardisation and structure on the one hand, and flexibility, change and innovation on the other. It would be fatal to assume that one is the answer to the other and Johnson (1996) has developed a **Polarity Management™** model through which organisations can gain an understanding of interdependent polarities, so that they may manage them effectively. The discussion of this model falls outside the scope of this study; however, it is such a powerful mechanism for the understanding of these interdependent issues,

that a short overview and a polarity map of the standardisation-innovation polarity at CText is offered in Appendix A.

- **Communication** is an important element to project, innovation and change management, as well as a critical feature of business process management. Therefore, it is imperative that the necessary measures be taken to ensure effective communication within CText. It is recommended that CText review its communication methods to address the problems that have been identified in this study and support the BPM and innovation initiatives. Some suggestions in this regard are:
 - . **Support staff meetings:** Periodic meetings between the support staff, project managers and the head during which a brief overview of recent developments is offered.
 - . **Regular newsletter:** The introduction of a regular e-newsletter is proposed as a more suitable alternative to meetings. The newsletter can be written by employees themselves who take turns to gather and publish information. In addition, more detailed communication should originate biannually from management in the form of an official newsletter, which might also be distributed to the dean and colleagues from other departments and organisations that have a vested interest in CText.
 - . **Social interaction:** An investigation into social interaction should be undertaken with the objective to make recommendations regarding social office space, organised as well as no-strings socialising, which will nurture social interaction and lead to knowledge creation and creativity.
- **First BPM initiative:** Once all the preceding suggestions have been implemented, the scene is set for a first BPM initiative. It is proposed that a workshop be held where BPM can be introduced and a first project for implementation can be identified. The process should be facilitated by a business process expert, to guide the team through the process from planning to implementation. One of the outcomes should be the acquisition of knowledge by employees of how to repeat the process in future for review and improvement of BPM initiatives, as well as the incorporation of other processes as needed for more efficient operations.

- **Software development methodologies:** The developers should familiarise themselves with popular methodologies, their uses, advantages and challenges. This presents the opportunity for a colloquium series, to which other developers can be invited to share their perspectives, and project managers too should attend these events as observers. Short presentations by guest speakers, followed by discussions and practical experimentation should facilitate practical learning about methodologies.
- **Phase 2 of the Capability Maturity Model:** Appendix B contains more information on the key process areas and their objectives. The CMM does not indicate how these goals should be achieved (Yardley, 2002:243); this remains to be determined by CText. In this sense, moving to Level 2 becomes a team effort in which everybody can collaborate to achieve these goals. This project is a natural outflow from the software development methodology project, since Level 2 is achieved through processes that will have been discussed in the prior project, for example requirements and configuration management.
- **Learning networks and knowledge management** have not been discussed in detail in this study and will require further investigations. It is foreseen that through learning networks CText can find relevant answers to the questions regarding software development methodologies, the CMM, documentation, etc.
- **Records management:** It is suggested that the existing documentation policies be reviewed and a plan for improved management be developed. An improved record management system will support Level 2 of the CMM, while solving CText's current information problems. The objective should be solid management; fast lookup, distribution of the right information to the right people and maintenance of information.

It is the author's conviction that the above recommendation for BPM will lead CText to effectively deal with its cost, quality and project management concerns, and ultimately improve its software development work in general. Managing the Centre successfully through this endeavour will prove very challenging, but the result will be worth the while.

5.3 FUTURE RESEARCH

Following the recommendation that CText should not invest in a BPMS, but look at ways to manage certain phases more effectively, future studies should look at tools and methods through which these phases can be optimised. One example might be a comparative study between a number of software configuration management tools or methods to determine which can contribute the most to software development.

With regard to BPM in general, the following opportunities for future research have been identified:

- Determining how an intrapreneurial organisation can meet basic documentation requirements without harm to its agility and creativity, and thus manage this polarity effectively. The literature consulted on innovation and intrapreneurship contained very little if any information regarding the administrative functions within intrapreneurial entities.
- An investigation into models for innovation to determine the possibility that appropriate innovation programmes can eliminate the risk contained in BPM. It is, for example, possible that a BPM initiative can in fact work as a creative impetus for an organisation, as it will require everyone to think about their work in a different way.
- The impact of BPMS on creativity and innovation. Literature on this topic is extremely scarce.
- The success of BPMS within smaller organisations, in relation to larger organisations.
- BPMS interaction with humans: successes and challenges.

REFERENCES

- AMBLER, S.W. 2006. Agile requirements best practices.
<http://www.agilemodeling.com/essays/agileRequirementsBestPractices.htm> Date of access: 7 Aug. 2006.
- ÅMO, B.W. & KOLVEREID, L. 2005. Organizational strategy, individual personality and innovation behaviour. *Journal of Enterprising culture*, (13)1: 7-19. March. Available: Business Source Premier. Date of Access: 6 Sep. 2006.
- ANON. 2006?. Intrapreneurship. *Reference for business*.
<http://www.referenceforbusiness.com/small/Inc-Mail/index.html> Date of access: 6 Sep. 2006.
- BATISTA, J. & DE FIGUEIREDO, A.D. 2000. CMM in a micro team: A case study.
http://www.iscn.at/select_newspaper/assessments/iscaa.html Date of access: 13 Sep. 2006.
- BAEYENS, T. 2004. The state of workflow.
<http://www.theserverside.com/tt/articles/content/Workflow/article.html> Date of access: 29 Jul. 2006.
- BEARDSLEY, S.C., JOHNSON, B.C. & MANYIKA, J.M. 2006. Competitive advantage from better interactions. *The McKinsey Quarterly*, 22 Sep.
http://www.mckinseyquarterly.com/article_page.aspx?ar=1767&L2=18&L3=30
Date of access: 22 Sep. 2006.
- BECKER, J., VON UTHMAN, C., ZUR MÜHLEN, M. & ROSEMAN, M. 1999. Identifying the workflow potential of business processes (In HICSS. Papers read at the 32nd Hawaii International Conference on System Sciences held in Maui on 5-8 January 1999.) 10 p. [http://www.workflow-research.de/Publications/PDF/JOBE.CHUT.MIZU.MIRO-HICSS\(1999\).pdf](http://www.workflow-research.de/Publications/PDF/JOBE.CHUT.MIZU.MIRO-HICSS(1999).pdf) Date of access: 16 Feb. 2006.

- BESSANT, J., KAPLINSKY, R. & MORRIS, M. 2003. Developing capability through learning networks. *International Journal of Technology Management & Sustainable Development*, 2(1):19-38. Available: Academic Source Premier. Date of access: 26 Sep. 2006.
- BLECHAR, M.J. & SINURE, J. 2006. Magic quadrant for business process analysis tools. *Gartner RAS Core Research Note G00137850*, 27 Feb. www.proformacorp.com/Downloads/files/proforma1713.pdf Date of access: 7 Oct. 2006.
- BOOCH, G. 1996. Object solutions: managing the object-oriented project. Reading, Mass.: Addison-Wesley. 322 p.
- BOOCH, G. 2001. Developing the future. *Communications of the ACM*, 44(3):118-121. Available: Academic Source Premier. Date of access: 18 Sep. 2006.
- BPMG. **See** BPM GROUP.
- BPM GROUP. 2005. Process Factory revolutionises delivery of BPM to businesses. <http://www.bpmg.org/Zpost1057.php> 19 Apr. Date of access: 15 Sep. 2006.
- BPM GROUP. 2006. Home page. <http://www.bpmg.org/index.php> Date of access: 26 Sep. 2006.
- BPMI. **See** BPM INSTITUTE.
- BPM INSTITUTE. 2006. About us. <http://www.bpminstitute.org/company/about-us.html> Date of access: 26 Sep. 2006.
- BROWN, R. & WIDELL, R. 2006. Managing business processes through collaborative workflow systems. (In TMCE 2006 Proceedings. Paper read at the Sixth International Symposium of TMCE held in Ljubljana, Slovenia on 18–22 April 2006. 12p.) www.qline.com/Files/WEBSWhitePaper.pdf Date of access: 16 Feb. 2006.
- BURLTON, R.T. 2006?. Effective business change through process management: strategies and architectures for integrated change. <http://www.processrenewal.com> Date of access: 7 Oct. 2006.

- BURNS, P. 2005. Corporate entrepreneurship – Building an entrepreneurial organisation. New York: Palgrave. 326 p.
- CARDOSO, J., BOSTROM, R.P. & SHETH, A. 2003. Workflow management systems and ERP systems: differences, commonalities, and applications. *Journal of Information and Technology Management*, <http://sdis.cs.uga.edu/library/download/CBS03-ITM.pdf> Date of access: 14 Feb. 2006.
- CARNEGIE MELLON SOFTWARE ENGINEERING INSTITUTE. 2004. What is a CASE environment? http://www.sei.cmu.edu/legacy/case/case_what_is.html Date of access: 18 Sep. 2006.
- CHARETTE, R.N. 2005. Why software fails. *Spectrum, IEEE*, 42(9):42-49. Sep. <http://www.spectrum.ieee.org/print/1685> Date of access: 9 Aug. 2006.
- CHANG, J.F. 2006. Business process management systems: strategy and implementation. New York: Auerbach. 286 p. <http://www.bpminstitute.org/whitepapers/research/article/business-process-management.html> Date of access: 5 Aug. 2006.
- CHAPPELL, D. 2005. Understanding BizTalk Server 2006. Microsoft white paper. <http://www.microsoft.com/biztalk/techinfo/whitepapers/understanding.msp> Oct. 32p. Date of access: 2 Aug. 2006.
- CHAPPELL, D. 2006. On defining "Workflow". *David Chappell's Weblog*. Blog post date: 20 May 2006. <http://www.davidchappell.com/blog/2006/05/on-defining-workflow> Date of access: 1 Aug. 2006.
- COETSEE, L.D. 2002. Peak performance and productivity: a practical guide for the creation of a motivating climate. *Ons drukkers*. 212 p.
- CONWAY, K. 2001. Software project management: from concept to deployment. Scottsdale, Ariz.: Coriolis. 832 p.

- COULSON-THOMAS, C. 2005a. Achieve commercial success and personal fulfilment: entrepreneurship that can benefit individuals and the enterprise. Post date: 17 May 2005. <http://www.bpmg.org/Zpost1084.hph> Date of access: 15 Sep. 2006.
- COULSON-THOMAS, C. 2005b. Creating an entrepreneurial culture. Post date: 27 May 2005. <http://www.bpmg.org/modules.php?op=modload&name=News&file=article&sid=1109&mode=thread&order=0&thold=0> Date of access: 15 Sep. 2006.
- CTEXT, NWU. 2006. CText Reglement. Potchefstroom. 14 p.
- CZERNICH, C. H. 2003. When do ideas survive in organizations? Variable demands on resource-relatedness and framing efforts of the intrapreneur. Paper presented at the Academy of Management Annual Conference held in Seattle on 1-6 August. (IIB - Institute of International Business, WP 03/1).
- DAVIS, A.M. & LEFFINGWELL, D.A. 1995. Using requirements management to delivery of higher quality applications. *Rational Software Corporation*. http://tinf2.vub.ac.be/~dvermeir/courses/software_engineering/696wp.pdf Date of access: 13 Sep. 2006.
- DOS SANTOS, V. 2006. (rvsantosmeister@gmail.com) 26 Oct. 2006. BPM in innovative organisations. E-mail to: Janke, U. (ntluj@puk.ac.za)
- DOUGLASS, R. 2006. BPM: From strategy to execution. *Bpm.com*. Jun. 26. <http://www.bpm.com/FeaturePrint.asp?FeatureID=210> Date of access: Aug. 11
- E-WORKFLOW. 2006. The key benefits of workflow. <http://www.e-workflow.org> Date of access: 29 Jul. 2006.
- FINGAR, P. & BELLINI, J. 2004. The real-time enterprise. Tampa, FL: Meghan-Kiffer. <http://www.mkpress.com/rte.pdf> Date of access: 24 Jan. 2006.
- FLOWCENTRIC. 2005. Application brochure: Typical processes. Centurion. 2p. Date of access: 28 Nov. 2005.

- GAMBRILL, D. 2006. The holistic art of BPM. *Canadian Underwriter*, Jul.
http://www.canadianunderwriter.ca/issues/ISArticle.asp?story_id=160015113332&issue=07012006&PC Date of access: 25 Aug. 2006.
- GILBERT, P. 2005. Perspectives in process: bpm vs. BPM. *Phil Gilbert Blog*. Blog post date: 12 May 2005. http://blog.lombardicto.com/2005/12/bpm_vs_bpm.html
 Date of access: 13 Feb. 2006.
- GOLDMAN, A. 2006. BPM for small business, Version 1.0. *ISP-Planet*, 15 May.
http://www.isp-planet.com/equipment/2006/colosa_processmaker.html Date of access: 7 Sep. 2006.
- HA, B.H., BAE, J., PARK, Y.T. & KANG, S.H. 2006. Development of process execution rules for workload balancing on agents. *Data & Knowledge Engineering*, 56(1):64-84, Jan. Available: Academic Search Premier. Date of access: 18 Feb. 2006.
- HANDZIC, M. & CHAIMUNGKALANONT, M. 2004. Enhancing Organisational creativity through socialisation. *Electronic journal of knowledge management*, 6(1).
www.eikm.com/volume-2/v2i1/v2-i1-art6-handzic.pdf Date of access: 23 Jan. 2006
- HILDEBRAND, C. 2005. A brighter lightbulb. *PM Network*, 19(5):36-41. May. Available: Business Source Premier. Date of access: 9 Aug. 2006.
- HISRICH, R.D. & PETERS, M.P. 2005. *Entrepreneurship*. 6th ed. Boston, Mass: Irwin McGraw-Hill.
- HUDSPETH, N. 2006. Verbal account to Janke U. Johannesburg.
- HUGHES, B. & COTTERELL, M. 2002. *Software project management*. 3rd Ed. London: McGraw-Hill. 358 p.
- IBM. 2005. Software developments for the on demand enterprise buyer's guide. 16p.
<http://www-306.ibm.com/software/tivoli/resource-center/overall/bg-on-demand.jsp>
 Date of access: 9 Aug. 2006.

IQPC **see** INTERNATIONAL QUALITY & PRODUCTIVITY CENTRE

- INTERNATIONAL QUALITY & PRODUCTIVITY CENTRE. 2006. (*In congress brochure: 3rd annual BPM Congress*). <http://www.bpmcongress.com/cgi-bin/templates/genevent.html?topic=232&event=10901> Date of access: 26 Aug. 2006.
- JACKSON, D. 2000. *Becoming dynamic: creating and sustaining the dynamic organisation*. London: Macmillan. 232 p.
- JACOBSON, I., BOOCH, G. & RUMBAUGH, J. 1999. *The Unified Process*. *IEEE software*, 4(16):96-102, May/June.
- JENKINS, T. 2005. *Enterprise content management: what you need to know*. Open Text: Waterloo. 306 p.
- JOHNSON, B. 1996. *Polarity management: identifying and managing unsolvable problems*. Amherst, Mass: HRD Press. 267 p.
- JOHNSON, J. 1999. *Turning chaos into success*. *Software Mag.com*, Dec. <http://www.softwaremag.com/L.cfm?doc=archive/1999dec/Success.html> Date of access: 23 Sept. 2006.
- KHAN, R.N. 2005. *BMP: A global view*. *bptrends.com*, Mar. <http://www.bptrends.com/publicationfiles/03-05%20COL%20Global%20BPM%20-%20Assumptions%20-%20Kahn.pdf> Date of access: 14 Feb. 2006.
- KEMSLEY, S. 2006. *A short history of BPM*. *ebizQ blog. Column 2*. <http://www.ebizq.net/blogs/column2/archives/bpmhistory/> Blog post date: 1-9 May 2006. Date of access: 31 Jul. 2006.
- KOHRELL, D. & WONCH, B. 2005. *Using RUP to manage small projects and teams*. *The rational edge*, 15 jul. <http://www-128.ibm.com/developerworks/rational/library/jul05/kohrell/index.html> Date of access: 28 Oct. 2006.
- KRISTICK, J. (jkristic@tibco.com) 19 Sep. 2006. *BPM and software development*. E-mail to: Janke, U. (ntluj@puk.ac.za)

- KRUCHTEN, P. 2001. What is the Rational Unified Process? *The rational edge*, Jan.
http://www.therationaledge.com/content/jan_01/f_rup_pk.html or <http://www.-128.ibm.com/developerworks/rational/library/content/RationalEdge/archives/jan01.html>
 Date of access: 28 Oct. 2006.
- LYNEHAM-BROWN, D. 2006. How to apply best practice principles of BPM in the real world. *Workshop presenter: 3rd annual BPM Congress, Gallagher Estate, Jhb.*
<http://www.bpmcongress.com> / <http://www.iqpc.co.za> 14 Sep.
- MANIMALA, M.J., JOSE, P.D. & THOMAS, K.R. 2006. Organizational constraints on innovation and intrapreneurship: insights from public sector. *Vikalpa*, 31(1):49-60.
 Available: Business Source Premier. Date of access: 11 Aug. 2006.
- MATLIS, J. 2005. BPEL. *Computerworld*, Jun. 20.
<http://www.computerworld.com/developmenttopics/development/webservices/story/0,10801,102580,00.html> Date of access: 26 Sep. 2006.
- MC **see** MICROSOFT CORPORATION
- MCGOVERN, J. 2005. Pay attention: The difference between BPM and workflow! *IT-toolbox Blogs*. Blog post date: 25 Jan. 2005.
<http://blogs.ittoolbox.com/eai/leadership/archives/002875.asp> Date of access: 14 Feb. 2006.
- MCGUIN, P. 2006?. Prince2 and Atlantic Global solutions. *Atlantic Global*.
<http://www.atlantic-ec.com/prince2.html> Date of access: 5 Nov. 2006.
- METASTORM. 2006. Roundtrip business process management — why you need it.
http://www.metastorm.com/products/overview_index.asp Date of access: 24 Jan. 2006.
- MICROSOFT CORPORATION. 2004. Understanding BizTalk Server 2004.
http://msdn.microsoft.com/library/default.asp?url=/library/en-us/BTS_2004WP/html/c245a8af-9f01-410f-b1bc-c43e725bfc27.asp Feb. Date of access: 2 Aug. 2006.

- MICROSOFT CORPORATION. 2006a. Enabling "people-ready processes" through business process management (BPM). Microsoft white paper.
<http://www.microsoft.com/biztalk/solutions/bpm/whitepaper.msp> Date of access: 2 Aug. 2006.
- MICROSOFT CORPORATION. 2006b. Business process management overview.
<http://www.microsoft.com/biztalk/solutions/bpm/overview.msp> 12 Jul. Date of access: 2 Aug. 2006.
- MIERS, D. 2005. BPM – too much BP, not enough of the M. (*In*: Fischer, L., *ed.* 2005 Workflow handbook. Lighthouse Point, FL: Future Strategies Inc. p.23-29.)
http://www.wfmc.org/information/Miers_too_%20much_BP.pdf Date of access: 29 Jul. 2006.
- MOONEY, L. 2005. How to choose the right business process management solution. Metastorm white paper.
<http://whitepapers.techrepublic.com.com/whitepaper.aspx?docid=140090> Date of access: 11 Jan. 2006.
- MORRIS, M.H. 1998. Entrepreneurial intensity. Westport: Quorum Books. 170 p.
- MORRIS, M.H. & KURATKO, D.F. 2002. Corporate entrepreneurship. Fort Worth: Harcourt. 364 p.
- MULLER, J.C. 2006. (itbjcm@puk.ac.za) 22 Sep. 2006. BPM in software development. E-mail to: Janke, U. (ntluj@puk.ac.za).
- O'BRIEN, J.A. 2004. Management information systems. 6th ed. Boston: McGraw Hill. 529p.
- PAULK, M.C., CURTIS, B., CHRISSIS, M.B. & WEBER, C.V. 1993. SEI Technical Report. Capability Maturity Model for software, version 1.1.
<http://www.sei.cmu.edu/publications/documents/93.reports/93.tr.024.html> Date of access: 24 Sep. 2006.

- PLESUMS, C. 2002. Introduction to workflow. (*In*: Fischer, L., *ed.* 2002 Workflow Handbook. Lighthouse Point, FL: Future Strategies Inc. p.19-38.)
[http://www.wfmc.org/information/Introduction to workflow02.pdf](http://www.wfmc.org/information/Introduction%20to%20workflow02.pdf) Date of access: 29 Jul. 2006.
- PRINCE2. 2004. PRINCE2™ Maturity Model paper. Issue 1.5.
<http://www.prince2.com/implementation.html> or www.customer-projects.co.uk/Documents/PRINCE2MaturityModel.pdf Date of access: 7 Oct. 2006.
- PRINCE2. 2006. PRINCE2 info, PRINCE2 structure.
<http://www.prince2.com/p2structure.html> Date of access: 7 Oct. 2006.
- PRINSLOO, M. 2006. Interview. Potchefstroom. (Interview notes in possession of author.)
- PRIOR, C. 2003. Workflow and process management. . (*In*: Fischer, L., *ed.* 2003 Workflow Handbook. Lighthouse Point, FL: Future Strategies Inc. p.17-25.)
[http://www.wfmc.org/information/Workflow and Process Management.pdf](http://www.wfmc.org/information/Workflow%20and%20Process%20Management.pdf) Date of access: 29 Jul. 2006.
- PYKE, J. 2006. BPM in context: now and in the future. (*In*: Fischer, L., *ed.* 2006 Workflow Handbook. Lighthouse Point, FL: Future Strategies Inc. p.17-28.)
<http://www.wfmc.org/information/handbook06.htm> Date of access: 29 Jul. 2006.
- RANGANATHAN, V. & DUBROCK, M. 2004. The benefits of adopting BPM are simple – competitive advantage. *CIO Magazine, Consultants' briefing*, May 13.
<http://www2.cio.com/consultant/report2513.html> Date of access: 7 Oct. 2006.
- RIEDERER, J.P., BAIER, M. & GRAEFE, G. 2005. Innovation management: an overview and some best practices. *C-LAB report*, 4(3). 58 p.
- SAVVION. 2006a. Business process management.
http://www.savvion.com/business_process_management.php Date of access: 13 Feb. 2006.

- SAVVION. 2006b. Making business improvement real: how people power the process - an overview of Savvion BusinessManager 6.8. (White paper.)
<https://www.bpminstitute.org/whitepapers/whitepaper/article/making-business-improvement-real-how-people-power-the-process-an-overview-of-savvion-businessmana.html> Date of access: 5 Aug. 2006.
- SCHACH, S.R. 2005. Object-oriented & classical software engineering. 6th ed. New York: McGraw-Hill. 581 p.
- SCHWABER, C., ORLOV, L.M., ZETIE, C. & DOWLING, K.Q. 2005. Process-centric software configuration management. *The Forrester Wave*, Q4.
www.borland.com/resources/en/pdf/company/analyst_news/forrester_scm_wave.pdf Date of access: 24 Sep. 2006.
- SEI. **See** CARNEGIE MELLON SOFTWARE ENGINEERING INSTITUTE.
- SETHI, R., SMITH, D. C. & PARK, C. W. 2002. How to kill a team's creativity. *Harvard Business Review*, 80(8):16-17. Aug. Available: Business Source Premier. Date of access: 9 Aug. 2006.
- SINGULARITY. 2006. Software. <http://www.singularity.co.uk/solutions-software.asp>
Date of access: 12 Sep. 2006.
- SGI. **See** STANDISH GROUP INTERNATIONAL.
- SIEGEL, J. 2005. Introduction to OMG's Unified Modeling Language.
http://www.omg.org/gettingstarted/what_is_uml.htm Date of access: 26 Sep. 2006.
- SILVER, B. 2006a. The 2006 BPMS Report: understanding and evaluating BPM suites. Bruce Silver Associates white paper.
<https://www.bpminstitute.org/whitepapers/research/article/2006-bpms-report-series-overview-understanding-and-evaluating-bpms-copy-1.html> Date of access: 5 Aug. 2006.

- SILVER, B. 2006b. Is workflow "back"? (Did it ever to away?). *BPMS Watch blog*.
 Blog post date: 18 May 2006. <http://www.brsilver.com/wordpress/2006/05/18/is-workflow-back-did-it-ever-go-away> Date of access: 1 Aug. 2006.
- SILVER, B. 2006c. Two ways to win at process improvement. *Intelligent Enterprise*,
 Aug. 1.
<http://www.intelligententerprise.com/showArticle.jhtml?articleID=190400346> Date
 of access: 5 Oct. 2006.
- SMITH, H. & FINGAR, P. 2003a. Workflow is just a pi process.
<http://www.bpm3.com/picalculus> Date of access: 31 Jan. 2006.
- SMITH, H. & FINGAR, P. 2003b. Digital Six Sigma. 14p .
[www.businessprocesstrends.com/deliver_file.cfm?fileType=publication&fileName=
 12-03%20ART%20Digital%20Six%20Sigma-Smith-Fingar1.pdf](http://www.businessprocesstrends.com/deliver_file.cfm?fileType=publication&fileName=12-03%20ART%20Digital%20Six%20Sigma-Smith-Fingar1.pdf) Date of access:
 31 Jan. 2006.
- SRINIVASAN, P. 2006. SOA and enterprise applications.
<http://msdn.microsoft.com/isv/solution/crm/soa/default.aspx> Date of access: 2
 Aug. 2006.
- STANDISH GROUP INTERNATIONAL. 2004. 2004 third quarter research report.
http://standishgroup.com/sample_research/index.php Date of access: 21 Sep.
 2006.
- STOKER, R. (robin@insurancegateway.co.za) 29 Sep. 2006. BPM in software devel-
 opment. E-mail to: Janke, U. (nfluj@puk.ac.za)
- SWART, J.N., FOWLER, A. & ALEXANDER, K. 2006. Interview. Johannesburg.
 (Digitised version of cassette recording in possession of author.)
- SWENSON, K. 2006a. BPM philosophy, not technology. *Go Flow Keith Swenson blog*.
 Blog post date: 25 Jun. 2006. [http://kswenson.wordpress.com/2006/06/25/bpm-
 philosophy-not-technology](http://kswenson.wordpress.com/2006/06/25/bpm-philosophy-not-technology) Date of access: 29 Aug. 2006.
- SWENSON, K. 2006b. 'Workflow' is back. *Go Flow Keith Swenson blog*. Blog post
 date: 6 May 2006. <http://kswenson.wordpress.com/2006/05/06/workflow-is-back>

Date of access: 31 Jul. 2006.

THOMPSON JR, A.A., STRICKLAND III, A.J. & GAMBLE, J.E. 2005. *Crafting and executing strategy: the quest for competitive advantage*. 14th ed. Boston: McGraw-Hill Irwin. 401 p.

TIMMONS, J.A. & SPINELLI, S. 2003. *New venture creation – entrepreneurship for the 21st century*. 6th ed. Boston: McGraw-Hill. 700 p.

TIWANA, A. & KEIL, M. 2004. The one-minute risk assessment tool. *ACM Queue, Programming Languages*, 2(19).
http://www.acmqueue.com/modules.php?name=Content&pa=printer_friendly&pid=239&page=1 Date of access: 21 Sep. 2006.

TROTT, P. 2005. *Innovation management and new product development*. 3rd ed. Harlow: Prentice Hall: 536p.

TROTT, P. & HOECHT, A. 2004. Enterprise resource planning and the price of efficiency: the trade off between business efficiency and the innovative capability of Firms. *Technology Analysis & Strategic Management*,16(3), p.367–379, Sep.

WIKIPEDIA. 2006a. Enterprise application integration.
http://en.wikipedia.org/wiki/Enterprise_application_integration Date of access: 29 Jul. 2006.

WIKIPEDIA. 2006b. Enterprise resource planning.
http://en.wikipedia.org/wiki/Enterprise_resource_planning Date of access: 18 Sep. 2006.

YARDLEY, D. 2002. *Successful IT project delivery: learning the lessons of project failure*. London: Addison-Wesley. 346p.

YARUZ, E.M. & HEIDELMAN, D. 1999. The “office water cooler” of the 21st century. *Medical marketing & media*, 34(4):66-69.

APPENDIX A: POLARITY MANAGEMENT

A.1 Polarities

Johnson (1996) has developed a Polarity Management™ model for managing so-called 'unsolvable' dilemmas. According to him, many trends in business are polarities to manage rather than problems to solve. Examples include centralisation v. decentralisation, competition v. collaboration, individual v. team and rigid structures v. flexible arrangements. Often the one is seen as a problem and its counterpart as the solution, whereas in reality they are sets of interdependent opposites that cannot function properly independently. Polarity management seeks to achieve the best of both opposites while avoiding their respective limits.

Johnson illustrates this with a very simple example: inhaling v. exhaling.

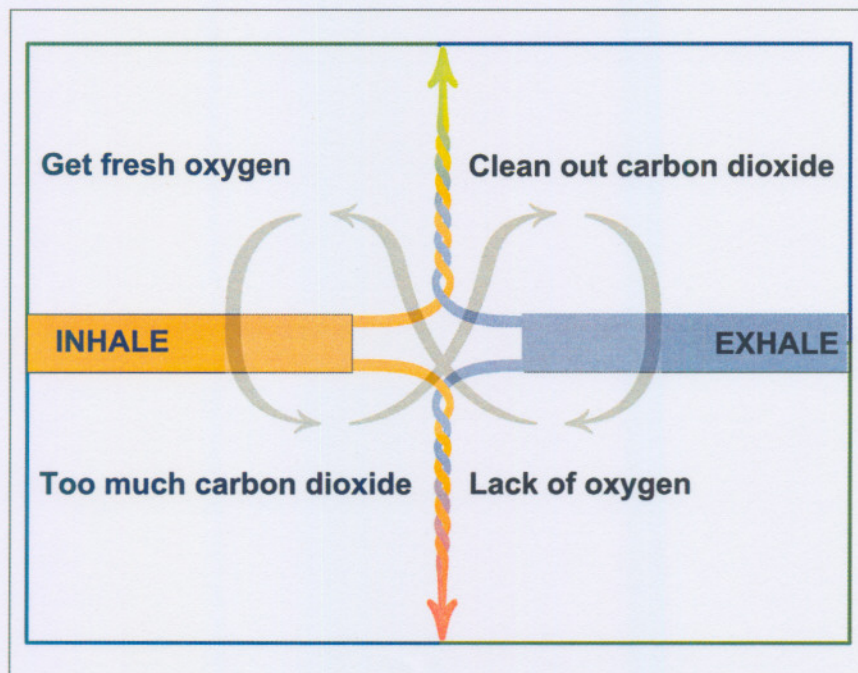


Figure 11: The breathing polarity (Johnson, 1996:21)

Neither should be seen as the solution to the other, since the solution will come to generate its own resistance. Focusing on inhaling will lead to an excess of carbon dioxide, and

a focus on exhaling will cause a lack of oxygen. There needs to be an ongoing oscillation between the poles.

By looking at the opposite of a current state as the solution to the current state, one will come to experience the disadvantages of that opposite state. The opposite also holds true: the fear of the possible disadvantages of the other pole will result in the experience of the disadvantages of the current pole.

Polarity management teaches that inhaling and exhaling are interdependent opposites which, if managed properly, will result in the experience of their combined advantages with minimised disadvantage. This is done as follows:

- For each polarity, compile a set of action steps for the maintenance of the positive results from focusing on this pole.
- For each polarity, compile a set of 'yellow flags', or measurable indicators that will serve as warning that movement is in the direction of the downside of the pole.
- Every time the yellow flags pertaining to a pole surface, it means that the oscillation is turning from the upside of that pole towards its downside and that it is time to adjust focus to the other pole, by following the set of action steps pertaining to that pole. In this way an organisation can from time to time shift its focus between centralisation and decentralisation, the team and the individual, structures and flexibility, etc. in order to get the most out of each pole at a particular time.

Polarity management also teaches the following very important lesson: all stakeholders must be involved in the process of mapping, so that everyone is heard and their view on the polarity acknowledged. By drawing a polarity map of an issue over which there is disagreement, the opponents get the full picture and thereby gain insight in the concerns and motivations of each other. Once the opponents can appreciate the validity of the other's stance, it becomes easier to start the move between the poles.

A.2 Polarities at CText

Drawing a standardisation-innovation polarity map of software development at CText creates a new perspective of the current situation. A disproportionate focus on maintaining

an innovative, no-limits environment has led teams to experience the downside of innovation, which is a chaotic environment in which each new project starts from scratch and the end products are of dubious quality.

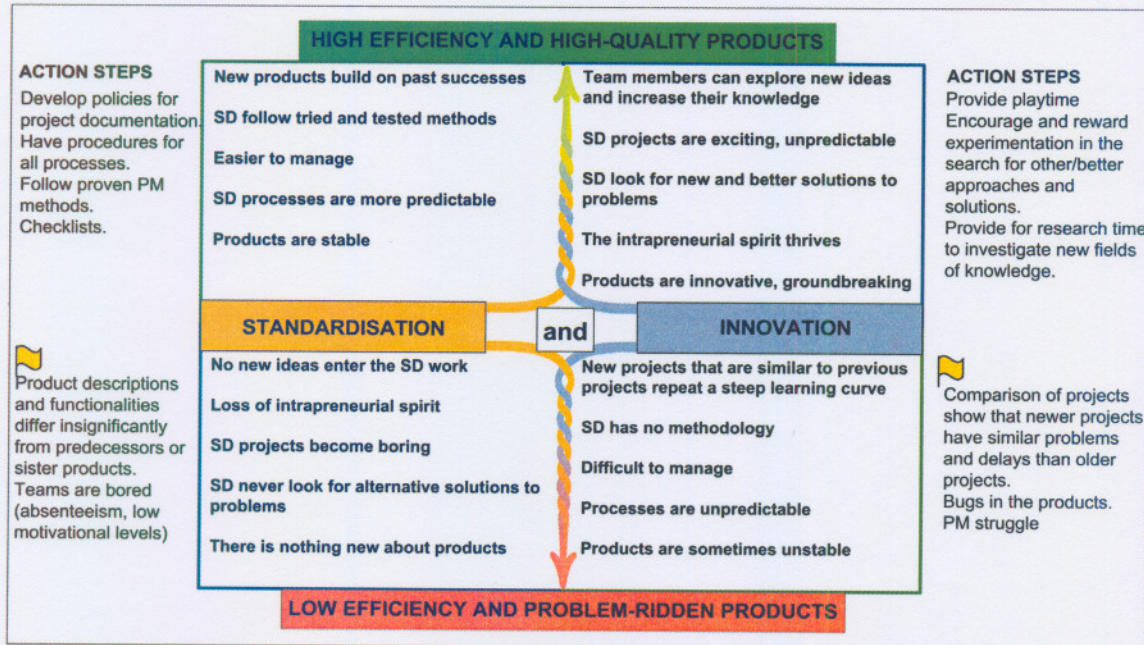


Figure 12: Standardisation-Innovation polarity map for CText

It is tempting to believe that standardised processes might be the solution to the current development problems, but this is a half-truth; neglect of innovation will eventually lead to the downside of standardisation. The challenge CText faces is to embrace both standardisation and innovation and to benefit from their respective advantages, while minimising the disadvantages.

APPENDIX B: LEVEL 2 OF THE CMM

Paulk *et al.* (1993:33-34, 59-60) provide the following description of the key process areas and their respective goals that an organisation must focus on in order to achieve Level 2 maturity:

- **Requirements Management:** establish a common understanding between the customer and the software project of the customer's requirements that will be addressed by the project. This agreement with the customer is the basis for planning and managing the software project. Control of the relationship with the customer depends on following an effective change control process.
 - . Goal 1: System requirements allocated to software are controlled to establish a baseline for software development and management use.
 - . Goal 2: Software plans, products, and activities are kept consistent with the system requirements allocated to software.
- **Software Project Planning:** establish reasonable plans for performing the software development and for managing the software project. These plans are the necessary foundation for managing the software project. Without realistic plans, effective project management cannot be implemented.
 - . Goal 1: Software estimates are documented for use in planning and tracking the software project.
 - . Goal 2: Software project activities and commitments are planned and documented.
 - . Goal 3: Affected groups and individuals agree to their commitments related to the software project.
- **Software Project Tracking and Oversight:** establish adequate visibility into actual progress so that management can take effective actions when the software project's performance deviates significantly from the software plans.
 - . Goal 1 Actual results and performances are tracked against the software plans.

- . Goal 2: Corrective actions are taken and managed to closure when actual results and performance deviate significantly from the software plans.
- . Goal 3: Changes to software commitments are agreed to by the affected groups and individuals.
- **Software Subcontract Management:** select qualified software subcontractors and manage them effectively. It combines the concerns of Requirements Management, Software Project Planning, and Software Project Tracking and Oversight for basic management control, along with necessary coordination of Software Quality Assurance and Software Configuration Management, and applies this control to the subcontractor as appropriate.
 - . Goal 1: The prime contractor selects qualified software subcontractors.
 - . Goal 2: The prime contractor and the software subcontractor agree to their commitments to each other.
 - . Goal 3: The prime contractor and the software subcontractor maintain ongoing communications.
 - . Goal 4: The prime contractor tracks the software subcontractor's actual results and performance against its commitments.
- **Software Quality Assurance:** provide management with appropriate visibility into the process being used by the software project and of the products being built. This is an integral part of most software engineering and management processes.
 - . Goal 1: Software quality assurance activities are planned.
 - . Goal 2: Adherence of software products and activities to the applicable standards, procedures, and requirements is verified objectively.
 - . Goal 3: Affected groups and individuals are informed of software quality assurance activities and results.
 - . Goal 4: Non-compliance issues that cannot be resolved within the software project are addressed by senior management.
- **Software Configuration Management:** establish and maintain the integrity of the products of the software project throughout the project's software life cycle. Soft-

ware Configuration Management is an integral part of most software engineering and management processes.

- . Goal 1: Software configuration management activities are planned.
- . Goal 2: Selected software work products are identified, controlled, and available.
- . Goal 3: Changes to identified software work products are controlled.
- . Goal 4: Affected groups and individuals are informed of the status and content of software baselines.