Expecting the Unexpected: Beyond Teleological Information Systems Development

Daniel Bartholomeus le Roux

Thesis submitted in fulfilment of the requirements for the degree of
Magister Artium (Socio-Informatics)
Stellenbosch University
March 2008

Supervisor: Christiaan Maasdorp

Declaration

I, the undersigned, hereby declare that the work contained in this thesis is my own original work and that I have not previously in its entirety, or in part, submitted it at any university for a degree.

Signature: __________________________

Date: __________________________

Summary

Information systems have become a standard and essential feature of contemporary organisations as they are applied to enable the management of information as an organisational asset in the unstable business world of the knowledge economy. The academic field, though extremely young, is a dynamic permutation of various professional domains and scientific research areas, making information systems a complex and often confusing subject.

Traditional information systems development methodologies, like the Systems Development Life Cycle, approach systems development with a teleological paradigm. This implies that information systems should be developed to adhere to a certain set of predefined system requirements. Although organisations have widely accepted this paradigm, some experts argue that it is insufficient when organisations are subject to frequent change as a result of turbulent business sectors. They suggest that information systems will operate in a changing context that will render any predefined set of system requirements futile.

In contrast to the teleological paradigm, these experts proclaim the ateleological paradigm as a more suitable approach to information systems development in organisations that require the inherent ability to adapt to a changing environment. The ateleological paradigm approaches an information system as a living system that should have the ability to adapt continuously to emerging or changing system requirements. Instead of being driven by system requirements that were fixed at a specific point in time, these information systems are developed over time continuum to ensure that the system remains relevant with the changing context in which it operates.

Tailorable Information Systems (TIS) is an information systems development approach that embodies the ateleological paradigm. As a central principle, TIS operates around the notion that information systems development should be done by the end-users of the system as opposed to the traditional system analysts and developers. By empowering the end-users of an information system with adequate technology and relying on their technical sophistication, organisations can implement truly flexible systems that are particularly responsive to contextual changes.

In the light of the ateleological paradigm, this thesis critically evaluates traditional information systems development approaches and compares these two approaches that support the notion of an information system as a living system.

Opsomming

Inligtingstelsels het 'n standaard en essensiële element van kontemporêre organisasies geword. Dit speel 'n integrale rol ter ondersteuning van hulle pogings om inligting as 'n organisatoriese bate te bestuur in die onstabiele besigheidswêreld van die kennisekonomie. Die akademiese veld, jonk soos dit is, is 'n dinamiese samekoms van verskeie professionele gebiede en wetenskaplike navorsingsareas. Dit veroorsaak dat inligtingstelsels 'n komplekse en dikwels verwarrende onderwerp is.

Tradisionele stelselontwikkelingsmetodologieë, soos die "Systems Development Life Cycle", benader stelselontwikkeling met 'n teleologiese paradigma. Dit impliseer dat inligtingstelsels ontwikkel word volgens 'n stel voorafbepaalde vereistes. Alhoewel hierdie paradigma algemeen deur organisasies aanvaar is, bevraagteken sekere kundiges die nuttigheid daarvan in organisasies wat onderworpe is aan gedurige verandering as gevolg van turbulente besigheidsomgewings. Hulle is van mening dat die veranderende konteks waarin die inligtingstelsel moet funksioneer enige voorafbepaalde stel verseistes nutteloos sal maak.

In kontras met die teleologiese paradigma, stel hierdie kundiges voor dat die ateleologiese paradigma 'n meer gepaste benadering tot stelselontwikkeling is indien organisasies staatmaak op hul vermoë om by 'n veranderende omgewing aan te pas. Die ateleologiese paradigma benader 'n inligtingstelsel as 'n lewendige sisteem wat instaat moet wees om aanhoudend aan te pas by nuwe of veranderende vereistes. Instede daarvan dat stelselontwikkeling gedryf word deur vasgestelde vereistes moet hierdie inligtingstelsels oor die verloop van tyd ontwikkel word om te verseker dat dit in pas bly met die veranderende konteks waarbinne dit moet funksioneer.

"Tailorable Information Systems" (TIS) is 'n inligtingstelsel-benadering wat die ateleologiese paradigma inkorporeer. As sentrale beginsel, word TIS ontwikkel rondom die idee dat stelselontwikkeling deur die gebruikers van die stelsel gedoen moet word instede van deur die tradisionale stelselanaliste en -ontwikkelaars. Deur gebruikers toe te rus met die toepaslike tegnologie en deur staat te maak op hulle tegniese vaardigheid kan organisasies buigsame inligtingstelsels ontwikkel wat dinamies reageer op kontekstuele veranderinge.

In die lig van die ateleologiese paradigma, evalueer hierdie teks die tradisionele benadering tot stelselontwikkeling en vergelyk dit met benaderings wat die idee ondersteun dat inligtingstelsels as lewendige sisteme gesien moet word.

Acknowledgements

I would like to, first and foremost, acknowledge my family for their continuous support in various forms and across geographical boundaries. I have endless admiration for my parents who have provided me with every opportunity a son could wish for and so much more; also my sisters for their encouragement and unconditional love. Words cannot express my gratitude.

Secondly, I want to thank my friends and colleagues for being a part of this project. Especially those who took the time to understand some of the more complex ideas expressed in this thesis and those who took my mind away from those ideas when the need surfaced. Special appreciation must be expressed for Minnaar Pieters who endured many a long night with me while working on his own thesis.

Finally, I want to acknowledge my study leader, Christiaan Maasdorp. His academic skill is a source of inspiration that motivated me throughout the duration of this project.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

The latter half of the 19th century brought about the invention of the telephone and telegraphy. Such inventions meant that information could, for the first time, move faster than physical objects and lead to what is now commonly known as the Information Age. Technology has come a long way since then. The accessibility of computer systems for personal and professional use, coupled with the birth and speedy growth of the internet, have formed a global network of information agents that freely generate and share information across all geographical boundaries.

The development of computer technology and its introduction to organisations had a major impact on working environments in the developed world. Tasks that traditionally required manual labour could now be automated and organisations could operate with fewer employees. At the same time, however, they required a new range of skills to manage information as an essential organisational asset. This change within organisations is reflected in economies that are no longer dominated by industrialisation but by knowledge and information services. Themes such as information and, more recently, knowledge management have received increased attention as organisations strive for a competitive advantage in the turbulent business environment of the 21st century.[1] Ever advancing computer technology lies at the heart of these efforts and has enabled the development of

[1] Magalhàes argues that "the knowledge economy environment for both manufacturing and service organisations requires new capabilities for competitive success. The ability of a company to mobilise and exploit its invisible or intangible assets has become far more decisive than investing and managing tangible, physical assets." Magalhàes, R. 2004. *Organisational Knowledge and Technology.* p 6.

Information Systems to manage every aspect of business operations.[1] Fruhling and De Vreede accurately state that "information systems that are designed and developed efficiently, accurately, reliably, and meet the intended needs and expectations of the stakeholders are important goals of organisations today".[2]

Modern information systems are integrated with organisational procedures to allow the seamless communication of business data among departments and support managers in the decision making process with real-time in-depth business information. Some organisations, often referred to as E-Businesses, extend their internal information systems to connect them with suppliers and consumers through the internet.[3] Transaction data is dynamically captured and fed back into financial systems to allow real-time performance analysis. These automated processes are no longer at the cutting edge of business technology, but have become the standard in the knowledge economy.

1.2 Research Problem

The birth of the internet and, subsequently, the World Wide Web have played central roles in increasing the turbulence of the modern business world in which all organisations have one thing in common. Change. Like living organisms, organisations are expected to continuously adapt themselves to changing environments to ensure their survival. This constant organisational change leads to challenges for the information systems that operate within such organisations. The knowledge economy, much more than its predecessors[4], requires information systems to cope with frequently changing or emerging information requirements as the organisational contexts in which they operate become increasingly unstable.

---

[1] Pearlson and Saunders state that information systems are integrated with almost every aspect of business in contemporary organisations. Pearlson, K. E. 2001. *Managing and Using Information Systems: A Strategic Approach*. p 4.

[2] Fruhling, A., De Vreede, G. 2006. *Journal of Management Information Systems*. p 40.

[3] Avgerou states that in the 1990s, the technological possibility of "interorganisational information systems, such as EDI and electronic commerce, has fuelled a new shift of research emphasis on processing cutting across organisational boundaries". Avgerou, C. 2001. *Information Systems Journal*. p 48.

[4] Nicholas argues that three salient characteristics distinguish modern society from earlier periods of history: interdependency, complexity, and rapid, radical change. Nicholas, J.M. 2001. *Project Management for Business and Technology. Principles and Practice*. p 9.

Contemporary organisations invest comprehensively to fund extensive information systems development projects. Such projects involve numerous groups of professionals and a series of complex processes. Surveys like that of Jiang et al.[1], however, show that organisations are often disappointed when their expectations are not met or the developed systems become inappropriate and outdated soon after implementation. Literature in the field, accordingly, reflects an "ongoing concern that there continues to be a considerable failure rate in information systems development"[2] - a problem which has "significant organisational consequences, in terms of both wasted critical resources and lost business opportunities".[3]

A survey conducted in 2001 revealed that only 24% of information system implementations are considered a success.[4] It also claimed that 31% of all information systems development projects were cancelled before completion. Accordingly, Maguire argues that "information system misuse and rejection are more frequent than acceptance and use".[5] This large amount of information system failures was already observable in the 1960s and gave rise to the term "software crisis"; referring to the general opinion that "systems took too long to develop, cost too much, and did not work very well".[6] Researchers have since realised that systems development is "more complex than has been assumed in the past"[7], yet identifying the main obstructions to successful systems development is still a much debated problem. Baskerville and Pries-Heje state that "a quarter-century has elapsed since the field first recognised that software development schedule and budget overruns are typical, and are often coupled with low quality and functionality".[8]

Despite the discouraging statistics, organisations are continuously making larger investments into information systems development. A 2001 report by the Standish Group claims that U.S.

[1] Jiang, J.J., Klein, G., Discenza, R. 2001. *IEEE Transactions on Engineering Management*. p 46.

[2] Goulielmos, M. 2004. *Information Systems Journal*. p 363.

[3] Xia, W., Lee, G. 2005. *Journal of Management Information Systems*. p 46.

[4] The results of this survey were published in an IEEE Transactions on Engineering Management article on the impact of development strategies and risks on information system success in 2001. Jiang, J.J., Klein, G., Discenza, R. 2001. *IEEE Transactions on Engineering Management*. p 46.

[5] Maguire, S. 2000. *Information Management & Computer Security*. p 231.

[6] Fitzgerald, B. 1996. *Information Systems Journal*. p 4.

[7] Goulielmos, M. 2004. *Information Systems Journal*. p 379.

[8] Baskerville, R., Pries-Heje, J. 2004. *Information Systems Journal*. p 237.

companies invested four times more money in information systems projects in 2000 than they did annually in the 1990s.[1] Gwillim et al. point out that there "are consistent doubts that IT investments are economically sound" yet they represent "more than half of most large firms' capital expenditure" – a phenomena referred to as the "IT productivity paradox".[2] Managers comprehend the importance of harnessing the potential of IT yet often find it difficult to justify large IT-related investments, highlighting the complexity of accurately measuring their ultimate value.[3]

Since its birth, information systems development theory has been dominated by a teleological paradigm. Systems analysts and developers[4] approach information systems development as an engineering problem that must be solved through the development of a product. To do this, they go through a series of steps to investigate the problem that needs to be solved. These steps manifest in the first three phases of the Systems Development Life Cycle (SDLC) which has been the dominant approach for developing information systems over the past 30 years. The SDLC, as presented by Avison and Fitzgerald[5], has six phases:

1. Preliminary Investigation
2. Systems Analysis
3. Systems Design
4. Systems Development
5. Systems Implementation
6. Systems Maintenance

[1] Xia, W., Lee, G. 2005. *Journal of Management Information Systems*. p 46.

[2] Gwillim, D., Dovey, K., Wieder, B. 2005. *Information Systems Journal*. p 308.

[3] Kumar, R. L. 2004. *Journal of Management Information Systems*. p. 11.

[4] It is important to note that system analysts and system developers are two completely different groups of professionals. Although analysts and developers often draw on a similar set of skills, analysts are primarily concerned with the pre-implementation processes of information systems development. These include a range of activities that are discussed in the course of this thesis, but mostly exclude the actual programming of software. In practice, most analysts start out as programmers and are later promoted to analysts. Oz, E. 2004. *Management Information Systems*. p 416.

[5] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 31.

The SDLC approach advocates the development of an information system based on an in-depth investigation into the requirements of the organisation in which the system will operate. The first three phases of the process produce detailed reports, diagrams and charts that stipulate how different parts of the system should function and interact. It allows analysts to identify possible obstructions and clarify system functionality. The finalised set of system requirements, agreed upon by all the stakeholders involved, form the driving force for the implementation phase as it guides programmers towards a working product. The approach has been particularly prominent in the short history of systems development theory and many methodologies inherit its requirements-driven structure. These are often referred to as traditional or hard methodologies.

The software crisis has led to a stream of research that critically investigates hard systems development methodologies in an attempt to make sense of the obstructions that system developers must overcome to achieve a successful product. Xia and Lee[1], who specifically investigate the complexity of systems development projects in contemporary organisations, report a variety of arguments as to why information systems are often perceived as being unsuccessful. One such argument is that hard methodologies tend to neglect the *social* dimension of information systems and focus too strongly on the technological dimension. Goulielmos, accordingly, points out that "there is a growing argument that information systems development should be seen as more than a technical activity" but rather "a complex social activity"[2] that cannot be undertaken without consideration of the organisational context in which a system will operate. Numerous researchers investigate the social processes underlying systems development; a good example being the communication of business requirements between system users and system developers.[3] This has led to the creation of more extensive methodologies to ensure system developers accurately comprehend the social context in which the information system will operate before the system is designed or implemented.

Another argument, however, is that hard methodologies fail to recognise the *dynamic* context of information systems. Stamoulis et al. support this view. In what they refer to as "the

---

[1] Xia, W., Lee, G. 2005. *Journal of Management Information Systems*.

[2] Goulielmos, M. 2004. *Information Systems Journal*. p 364.

[3] These include, among others, Flynn & Jazi (1998), Gallivan & Keil (2003), Avgerou (2001), Fitzgerald (1996) and Kiely & Fitzgerald (2005). They are all referenced in different sections of this thesis.

fallacy of correct information systems requirement specification"[1] they state that the very practice of developing information systems based on pre-defined requirements can be held responsible for a number of information systems being a disappointment.[2] In essence, their argument is that an artificial freeze is imposed on the exercise of eliciting user requirements for a system. It is based on the principle that users cannot know all their requirements since the system will work over time continuum and should not be developed for requirements relevant at one specific point in time. In contemporary organisations, the development process itself can take several years to complete, by which time the business requirements of the organisation may have changed.[3] Traditional methodologies, however, limit information systems functionality to a set of requirements that were fixed before system implementation commenced. The reality is that information systems "inevitably go into decline as the requirements of the organisation change over time".[4]

In accordance to this, Patel states that "the problem of developing business information systems is not so much an engineering problem as it is a living problem, if it can be regarded as a problem at all in the sense of having a solution".[5] Patel argues that one can never separate an information system from the business processes in which it operates and emphasises that these processes have a living (human) element to them. Maguire agrees by arguing that an exclusively technological view of systems development will create the possibility of "technical successes but organisational failure".[6] To support business processes that are continuously subject to organisational change, information systems "should be regarded as products and processes simultaneously which are continuously developed to facilitate data, information and knowledge".[7]

The key phrase in Patel's writings is *continuously developed*. He implies that an information system, after being implemented, will require ongoing maintenance to keep it in line with the changing business processes that it supports. Systems development is thus not viewed as a

---

1 Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 295.

2 Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 295.

3 Maguire, S. 2000. *Information Management & Computer Security*. p 230.

4 Maguire, S. 2000. *Information Management & Computer Security*. p 232.

5 Patel, N.V. 1999. *Proceedings of the 1999 Americas Conference on Information Systems*. p 4.

6 Maguire, S. 2000. *Information Management & Computer Security*. p 234.

7 Patel, N.V. 1999. *Proceedings of the 1999 Americas Conference on Information Systems*. p 4.

once-off project, but rather an ongoing process that enables system adaptation to changing requirements. Like a living organism, the system should be able to adapt to the change in its environment and continue to survive. Stamoulis et al. agree when they state that "information systems must be approached as living systems"[1] and not dead systems.

The notion of *Living Information Systems* implies a fundamentally different approach to systems development projects, both before and after the system becomes operational. It proclaims that an information system should be developed with the "inherent ability to flex and adapt to unforeseen circumstances"[2], dynamically shaping itself to fit the changing context within which it operates. This thesis explores that possibility by investigating the implications of adopting an *ateleological* paradigm for information systems development as opposed to a teleological paradigm.

1.3 Research Questions

1.3.1 Primary Research Question

What are the implications of teleological and ateleological paradigm adoption for information systems development, use and maintenance?

This thesis is primarily concerned with a qualitative analysis and comparison of two paradigms underlying information systems development, use and maintenance in contemporary practice. By exploring the implications of teleological and ateleological paradigm adoption I hope to further understanding of information systems failure and provide a new, more appropriate approach for organisations that operate in turbulent environments.

[1] Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 296.

[2] Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 296.

1.3.2 Secondary Research Questions

1.3.2.1 Why do traditional methodologies mostly fail to produce successful information systems?

The first question this thesis attempts to answer deals with the inability of traditional development methodologies (i.e. those that fall within the teleological paradigm) to produce systems that can cope with changing and emerging requirements. The high amount of information system failures and uncompleted development projects indicate that the lifecycle approach and its sequential steps are not always effective in its purpose. Determining the reasons for its incompetence is fundamental to the development of a new and successful approach that can overcome the complexities of information systems development in the 21st century. The software crisis has sparked a large amount of research on this topic as many experts question widely accepted systems development practices.

1.3.2.2 What are the nature and origin of emerging or changing information system requirements?

A systems development approach that can overcome the problems associated with dynamically changing or emerging requirements obliges an understanding of their source and the implications they have on the system. These requirements are directly related to the context in which the system operates and entails not only an understanding of the organisational environment, but also the larger business environment within which the organisation operates.

1.3.2.3 How do contemporary organisations cope with emerging or changing information system requirements?

The dominance of traditional development methodologies in the modern practice of information systems development necessitates that organisations employ certain strategies to cope with changing or emerging requirements. Despite the dismal statistics often associated with information systems development, organisations are still reliant on information systems functionality and must continue to develop and maintain them despite ongoing organisational change. It is of crucial importance that organisations maintain their agility in the increasing competitiveness of the knowledge economy and information systems play a central role in their efforts to do so.

1.3.2.4 How does the ateleological paradigm overcome the negative aspects of the teleological paradigm?

The theoretical development of the ateleological paradigm in contrast to the teleological paradigm is the central theme of this thesis. The comparison must be done with specific focus on organisational change and its impact on an information system's context. It also requires the identification of the key enablers of ateleological development as well as an investigation into the implications of ateleological development for the stakeholders of the information system. A logical extension to this question would involve an investigation into the technological dimension of the ateleological paradigm in an attempt to identify the technological enablers of the paradigm.

1.3.2.5 How will an information system developed within the ateleological paradigm be structured?

Information systems developed within the ateleological paradigm will most certainly be of a different structure than teleologically developed systems. This necessitates an investigation into a systems architecture for ateleologically developed systems and how this architecture can be implemented at various organisational levels. By focusing on Tailorable Information Systems (TIS) as having an implementable systems architecture, the theoretical ideas within the ateleological paradigm can be expressed more clearly.

1.3.2.6 What are the challenges to and pitfalls for the ateleological development paradigm?

The unstructured nature of ateleological systems development exposes the developed information systems to a series of challenges and pitfalls. The identification of these is an essential part of this thesis as it questions the plausibility of the paradigm within contemporary organisations and in the context of the technology that is currently available. It also serves to direct future research on the topic by identifying the areas where the paradigm is still theoretically weak.

1.4 Research Design and Methodology

The qualitative approach adopted for this study necessitated that a detailed literature study had to be conducted and that the arguments presented in this thesis are built around a strong theoretical core. Before information systems development practice and the two paradigms could be studied an attempt to make sense of the academic field of information systems on a conceptual level was made. This revealed a lack of consensus within published literature that results mainly from the intricate relationship between information systems and information technology, combined with the relative youth of the discipline. Using Checkland and Holwell as key authors, this conceptual ambiguity was investigated by focussing on the central concepts in the field and their meanings in various publications. This enabled the development of a conceptual framework within which further research could be conducted. Other authors of note in this regard are Carvalho, King and Lyytenin. Systems theory was an immiscible tool used throughout the study and the work of Von Bertalanffy was of notable value.

Information systems research is generally dominated by studies on the techniques and tools that are applied in the methodologies of the teleological paradigm.[1] To ensure that this section of the literature was thoroughly covered, an overview of the paradigm was done before research became focused on aspects specifically applicable to this study. Firstly, introductory texts to the field were studied to ensure a broad understanding of the various technical, social and organisational aspects involved. Focus areas for the study were identified and further analysed through more books; articles published in various journals and magazines; as well as conference proceedings. These articles were further supplemented by studies that specifically criticise or question traditional thinking in the field, often drawing on empirical data to supplement arguments. Specific attention was paid to case studies that recognise the effects of organisational change on information systems developed through teleological methodologies. Table 1.1 presents a summary of the most notable contributions in this regard.

[1] Avgerou states that a great deal of IS research has been preoccupied with technological change and that such studies typically focus on traditional information systems development practices. Avgerou, C. 2001. *Information Systems Journal.* p 45.

<table>
<tr><th>Authors</th><th>Nature of contributions</th><th>Keywords and phrases used in searches</th></tr>
<tr><td>Avison & Fitzgerald
Satzinger, Byrd & Jackson
Laudon & Laudon</td><td>Overview of information systems development practices.</td><td rowspan="4">Information systems development; systems development project; software development; method; methodology; SDLC; systems development life cycle; spiral; v-model; prototyping; software engineering; maintenance; evolution; evolutionary methodologies; agile development; requirements; requirements engineering; requirements elicitation; empirical evaluation; social process; social complexity; statistics; user-led design; project failure; change; adaptation;</td></tr>
<tr><td>Baskerville & Pries-Heje
Avgerou
Goulielmos
Flynn & Jazi
Kiely & Fitzgerald
Gallivan & Keil
Jiang & Klein</td><td>Investigation and evaluation of traditional development practices based on empirical data.</td></tr>
<tr><td>Fitzgerald
Maguire</td><td>Qualitative studies on traditional development practices.</td></tr>
<tr><td>Xia & Lee</td><td>Conceptualisation of development project complexity based on empirical data.</td></tr>
</table>

Table 1.1: Notable authors; the nature of their contributions and keywords used to locate literature on traditional development practices.

The next part of the literature study entailed a detailed analysis of published articles and conference proceedings on the ateleological paradigm and TIS. These articles enabled a theoretical understanding of the paradigm and the identification of the key differences between itself and the teleological paradigm. Most of the articles were authored by a combination of Greek and British academics and IT experts, among which Stamoulis, Kanellis, Martakos, Patel and Paul are the most notable. A search was done to locate and study as many as possible of their publications and some of the authors were personally contacted. They were kind enough to supply yet to be published research for supplementary reading. The publications highlight different aspects of the proposed concepts, approaching them from various standpoints and drawing on a range of related research to substantiate arguments. They reveal, however, a limited conceptual understanding of the phenomena and various authors noted that the theoretical roots of these phenomena are still being explored. Table 1.2 presents a summary of the most notable literature contributions focussing specifically on the ateleological paradigm and TIS.

Authors	Nature of contributions	Keywords and phrases used in searches
Stamoulis, Kanellis & Martakos	Conceptual development of TIS.	living information systems; ateleological paradigm; amethodical development; TIS; tailorable information systems; ateleological development; organisational change; flexibility; system adaptability; deferred systems design; self-organisation; self-organising theory; operational closure; structural coupling; changing requirements; emerging requirements; flexibility; dynamic complexity; systems context; misfit.
Patel	Qualitative investigation of TIS and DSD (Deferred Systems Design). Investigation of the implications of dynamic system requirements.	
Patel & Irani	Qualitative investigation of dynamic business environments and TIS.	
Kanellis & Paul	Investigation of information systems misfit based on empirical data.	
Stamoulis, Martakos & Introna	Conceptual development of TIS.	
Stamoulis, Patel & Martakos	Investigation of a systems architecture model for TIS.	
Kanellis, Lycett & Paul	An empirical study on information systems failure.	

Table 1.2: Notable authors on the atelelogical paradigm and TIS; the nature of their contributions and keywords used to locate literature.

Figure 1.1 presents a taxonomy of key concepts in the study and the authors relevant to each.

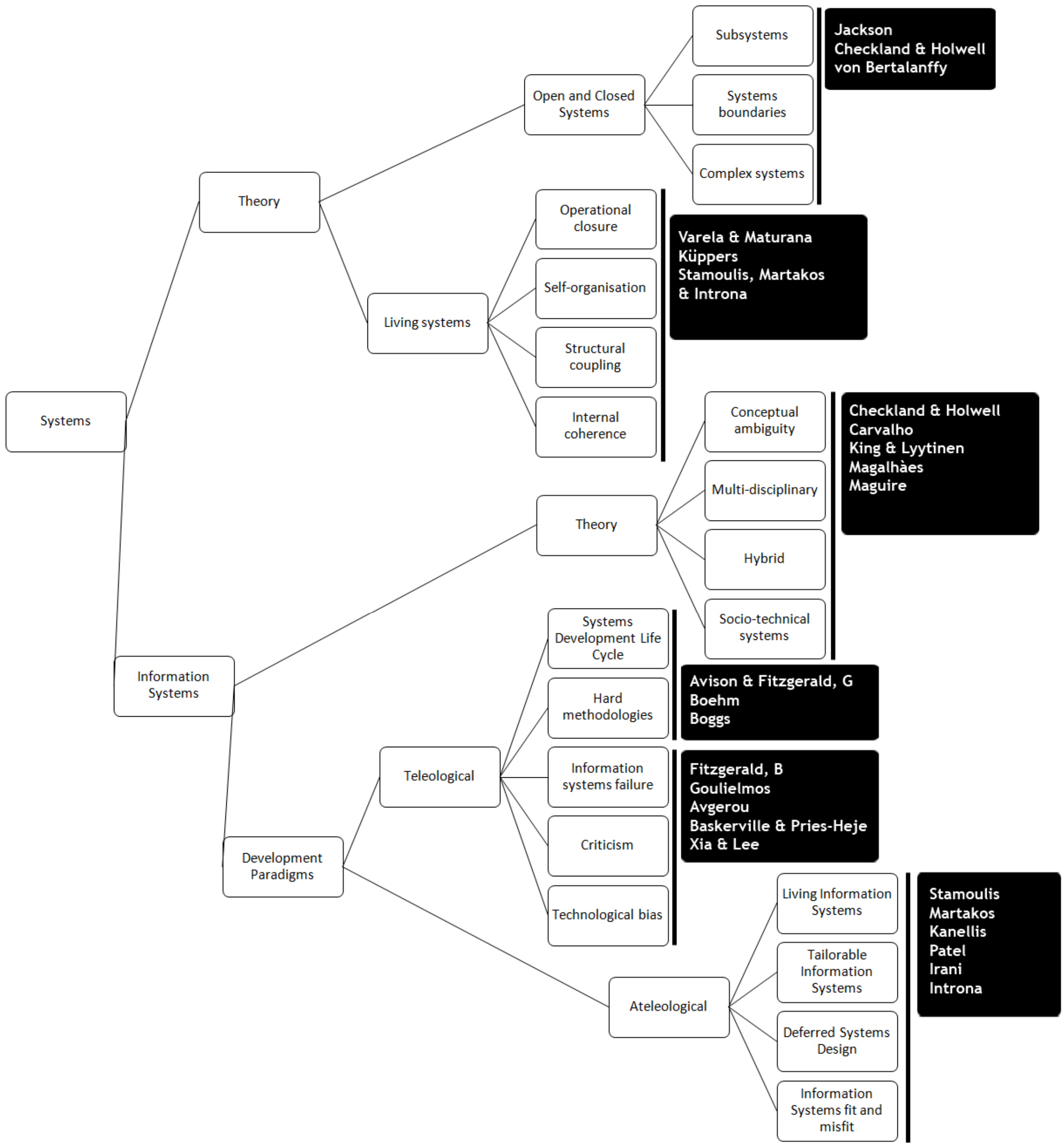

Figure 1.1: A taxonomy of key concepts and related authors.

1.5 Value for Research and Practice

Because information systems operate in almost every industry of a modern economy, research conducted for this thesis is of value to a wide group of stakeholders. The thesis is not written in a style that assumes significant technical knowledge and should be understandable to professionals outside traditional IT departments. It discusses the ateleological paradigm only after an in-depth study of traditional thinking in the field is presented and therefore lays a solid foundation upon which the essential arguments of the thesis can be understood.

The primary stakeholders of the research are academics who study the complexity of information systems development projects and search for more successful approaches to such projects. So far, research on this topic has focused mainly on the social complexity of these projects and proclaimed it to be the major obstacle for system success. This study, although acknowledging social complexity, offers a different perspective on the problem. Academics who are familiar with the systems development methodologies or have experience in current systems development practices will be the best qualified to critically evaluate the arguments presented in this thesis.

Secondary stakeholders include managers involved in information systems development or maintenance projects. Based on the research done, they will most likely identify with some of the challenges to systems development that are under scrutiny here. The thesis should also serve to encourage broader theoretical understanding of information systems development and the crucial relationship between system and context in contemporary organisations.

1.5.1 Value for Research

A key problem in the academic field of information systems is the conceptual ambiguity that surrounds the theoretical core of the field. This problem is discussed at the outset of the thesis as it is seen as a corner stone of the study. Information systems research is in desperate need of a more solid theoretical core and by clearly expressing the meanings of concepts used in research, the resulting reports will be of more value to the wide variety of readers. Information systems are, for several reasons, especially vulnerable to variants in theoretical interpretations and researchers should therefore pay particular attention to the formation of a theoretical foundation before continuing research. This thesis attempts to do exactly that.

A second valuable aspect of this thesis involves the critical evaluation of traditional thinking in the field. Statistics discussed in the course of this thesis point out that the theory developed throughout the short history of information systems must not be accepted blindly. This thesis questions traditional thinking in the field by investigating the inability of development methodologies that are broadly accepted in practice, to consistently produce successful information systems. By exposing the weaknesses of these methodologies, the thesis cultivates the need for new research in the field and the development of methodologies that are more applicable in the knowledge economy.

Finally, the thesis proposes a direction for future research in the field by investigating the ateleological paradigm. The limited research surrounding this paradigm reveals a weak conceptual understanding of phenomena already visible in practice. By introducing the notion and proposing TIS as an implementable solution to changing or emerging system requirements, the thesis significantly pushes theory forward and presents ideas around which more researchers can contribute. Objectivity is maintained by placing the ateleological paradigm in the context of its challenges and identifying key areas for future research.

1.5.2 Value for Practice

Although this thesis is primarily concerned with a theoretical comparison of paradigms, it holds value for various stakeholders in practice. It promotes a scholarly accurate and more holistic understanding of information management in practice by recognising not only the technological, but also the social and dynamic dimensions of information systems. This perspective is of particular value to managers as it emphasises the need for synergy between the various subsystems of the information system with the goal of effective overall information management.

The critical evaluation of broadly accepted practices in information systems development projects, should lead practitioners to question the affectivity of their own departments. Many researchers have promoted more rigorous application of traditional development methodologies in projects to overcome the high failure rates witnessed in practice.

Practitioners, however, face a host of unique challenges that are particular to their context and complicate the application of these methodologies. This thesis advocates that a step by step application of any methodology can be ineffectual independent of the completeness or accuracy of its implementation. It also places the information system in the context of a changing organisation which brings to light numerous obstacles to the application of traditional methodologies.

Many practitioners may find that the ateleological paradigm, as discussed in this thesis, has already been adopted (in part) by their organisations. The study should serve to confirm some of their beliefs but also further their understanding of the phenomenon in the broader organisational sense. For other practitioners the paradigm may seem futile. Large, bureaucratic organisations or those operating in less turbulent business environments are less likely to require, or gain advantage from an ateleological development paradigm. The thesis, however, might serve to encourage lateral thinking in these areas as organisational change is likely to increase in all sectors of business.

Finally, this study culminates in a model for paradigm selection and guidelines for flexible information systems design. These are presented in the final chapter and should provide practitioners with executable strategies for developing a responsive, dynamic information system.

1.6 Acknowledging a lack of Consensus

Checkland and Holwell, in 1998, emphasised the lack of consensus that characterised information systems literature at the time. They describe the academic field of information systems[1] as "crucial but confused".[2] King and Lyytenin reiterate this view in a 2004 article by arguing that the "perceived inadequacy of information systems as an academic enterprise" is the foremost concern in what they refer to as an "anxiety discourse" in the field, visible from as early as 1972.[3] This discourse typically evolves from the academic weakness that has become the target of the field's critics.[4]

It is therefore crucial that academic literature on information systems should start with adherence to the variations of interpretations that are typical to the field. To enable clear communication of theories and arguments, writers in the field (scholars and practitioners) should provide a conceptual framework within which publications can be interpreted accurately. This, however, is easier said than done, especially in the light of the large amount of conflicting views of what the field's main concerns should be. This thesis, though not primarily concerned with general information systems theory, requires the reader to form a clear understanding of the key concepts involved, from which further arguments can be constructed. It is therefore unavoidable to delve into the conceptual intricacy of the field to some degree and construct a sound theoretical platform for the study.

[1] Although it should be clear from its use in the text, it should be noted that the term *information systems* also denotes an academic domain. The study of information systems and their development is a "multi-disciplinary subject and addresses the range of strategic, managerial and operational activities involved in gathering, processing, storing, distributing and use of information, and its associated technologies, in society and organisations." Magalhàes, R. 2004. *Organisational Knowledge and Technology*. p 2. The distinction between these two meanings of the term should, however, be clear from the context within which they are used in the text and the discussion on information systems definitions (see section 1.7.2) focuses particularly on the actual systems and not the academic domain.

[2] Checkland, P., Holwell, S. 1998. *Information, Systems and Information Systems*. p 3.

[3] King, J.L., Lyytenin, K. 2004. Reach and Grasp. *MIS Quarterly*. p 540.

[4] King, J.L., Lyytenin, K. 2004. Reach and Grasp. *MIS Quarterly*. p 549.

Since its birth in the late 1940's[1], the field has expanded rapidly. The introduction of ever-advancing computer technology to organisations has fuelled the field's increasing importance in the business world, simultaneously transforming it into a complex, multidisciplinary academic domain.[2] Checkland and Holwell provide three reasons for the conceptual ambiguity witnessed within literature of the academic field of information systems: [3]

- Firstly, they acknowledge the relative youth of the field - "emerging only in the late 1940's with the introduction of the first computers".
- Secondly, they point out the inability of theoretical background to develop in support of fast developing technology by stating that thinking about the field "has its own rolling pace that cannot always be hurried".[4]
- Finally, they recognise that the field of information systems draws on a variety of other fields of study (not only technology) and consequently describe it to be a "hybrid" field.

Their arguments are in line with those reported by King and Lyytinen who state that "the most-powerful factor in explaining the information systems field's anxiety is its relative youth".[5] Another contributor to the anxiety is the field's "close ties to technology and the traditional antipathy toward technology found in management schools that are often the home of information systems groups".[6] Maguire points out that information systems research has been dominated by the technical issues of computer systems development and "less research

[1] Checkland, P., Holwell, S. 1998. *Information, Systems and Information Systems.* p 8.

[2] In a review of the first ten years of the publication of the *Information Systems Journal,* Avison et al. report that the field's multidisciplinary nature is reflected in the variety of fields from within which information systems research is conducted. They state that articles published in the journal were authored by people from the following institutes or departments: 25% information systems departments; 25% computer science departments; 5% practitioners; 40% business or management schools; 7% from either sociology, anthropology, statistics, design, psychology, management science, engineering and medicine. They conclude that although multidisciplinary is a positive factor, it obstructs the field's coherence and can result in a lack of focus. Avison, D., Fitzgerald, G., Powell, P. 2001. *Information Systems Journal.* p 11.

[3] Checkland, P., Holwell, S. 1998. *Information, Systems and Information Systems.* p 8.

[4] Checkland, P., Holwell, S. 1998. *Information, Systems and Information Systems.* p 9.

[5] King, J.L., Lyytenin, K. 2004. Reach and Grasp. *MIS Quarterly*. p 543.

[6] King, J.L., Lyytenin, K. 2004. Reach and Grasp. *MIS Quarterly*. p 543.

has been undertaken on the non-technical issues".[1] He quotes Lein and Hirschheim who view "information systems development as a technical process with social consequences".[2]

A key result of this dilemma and a primary obstacle to research in the academic field is the gap between the scholarly understanding of information systems and conventional wisdom about information systems. Checkland and Holwell partly blame this phenomenon on the inaccuracy of introductory texts to the field and state that they are out of touch with reality being "rooted in ideas about organisations in the 1960's".[3] This gap is reflected, for instance, in the incorrect usage of the term in everyday conversations to signify a software system. Phrases like "What information system are you running?" or "We are installing a new information system", which are of course harshly inaccurate in a scholarly context, are often used in practice. The gap is also, and more importantly, evident in the various fields of study that draw on the term. A study by Carvalho is an example of this. He focuses purely on how information systems are defined differently in literature from various fields and concludes that there are no less than four completely different systems all referred to as information systems in contemporary academic literature.[4] They are:

1. Organisations, like libraries, whose business it is to provide information to their clients.
2. A sub-system of any system capable of governing itself.
3. Any combination of active objects (processors) that deal with symbolic objects (information) and whose agents are computers or computer-based devices.
4. Any combination of active objects (processors) that deal with symbolic objects (information).

Carvalho accurately concludes that these various definitions for the same term "lead to confusion about what is the object of interest of a professional activity or a scientific domain".[5] This is especially significant as the field is characterised by its hybrid nature and lies at the intersection of a combination of academic domains.

[1] Maguire, S. 2000. *Information Management & Computer Security*. p 232.

[2] Maguire, S. 2000. *Information Management & Computer Security*. p 233.

[3] Checkland, P., Holwell, S. 1998. *Information, Systems and Information Systems.* Preface.

[4] Carvalho, J.A. 2000. *Proceedings of the 1999 Conference on Information Systems Concepts*. p 266.

[5] Carvalho, J.A. 2000. *Proceedings of the 1999 Conference on Information Systems Concepts*. p 280.

What makes the general lack of consensus specifically notable in this study is the recent ubiquity of information systems and information technology. Checkland and Holwell go as far as to say that some information technology professionals "even assume that information systems and information technology are synonyms".[1] Academic research on information systems that are developed and used through information technology draws on literature produced by both information systems scholars and information technology professionals. This necessitates the extrication of numerous conflicting definitions, assumptions and interpretations before the true subject of the research can receive any real attention. Likewise, it complicates the accurate communication of arguments to the reader.

To deal with this problem, papers and texts often start off with the development of working definitions for the key concepts under scrutiny. In doing so they supply readers with a theoretical framework within which they can interpret the work. The problem, however, is that the theoretical framework is often out of line with those developed by authors in some of the literature resources used during research and cited in the text. This results in further ambiguity and creates the possibility of misinterpretation. Iivari et al. agree by arguing that "if we are all interested in contributing to practice, we as a research community have a clear need for cross-paradigmatic interaction, regardless of our philosophical biases".[2]

1.7 Dealing with conceptual ambiguity

This thesis is concerned with information systems that are implemented through computer technology. Such systems, often referred to as Computer-Based Information Systems (CBIS), lie at the centre of the conceptual ambiguity as they bring together people and computers in a business context. King and Lyytinen emphasise this by arguing that "the fact that the essence of the information systems field lies at the intersection of the technical and the social makes it inherently tense".[3] It is therefore of critical importance that the key concepts used in this

[1] Checkland, P., Holwell, S. 1998. *Information, Systems and Information Systems.* Preface.

[2] Iivari et al. investigate the possibility of coding a practically relevant body of knowledge (BoK) for information systems and its benefits for the field, both in practice and theory. Iivari, J., Hirschheim, R., Klein, H.K. 2004. *Information Systems Journal.* p 313.

[3] King, J.L., Lyytenin, K. 2004. Reach and Grasp. *MIS Quarterly.* p 543

study are interpreted accurately to allow arguments that are void of ambiguity. The temptation to attempt a "conceptual cleansing of the field"[1], like that of Checkland and Holwell, had to be fought continuously as the literature used revealed numerous frustrating conflicts. The scope of this paper, however, does not justify such an attempt and attention was only paid to the concepts that are specifically applicable in the context of information systems development projects.

1.7.1 Systems

This thesis draws extensively on systems theory in the construction of arguments within the academic field of information systems. It is thus useful to develop a basic understanding of these ideas and grasp their significance in the academic literature in the field. A general definition for a system by Hall and Fagen reads as follows:

> A set of objects together with relationships between the objects and between their attributes.[2]

A system is "the result of viewing the active world from a certain point of view"[3] and provides us with a framework to study active objects. The basic system ideas were established by the classical Greek philosophers[4], but the systems language was developed and enriched by the "encounters of holism with philosophy, biology, control engineering, organisation and management theory, and the physical sciences".[5] Holism, as opposed to reductionism, is not primarily concerned with the study of the various parts that make up a system, but considers "systems to be more than the sum of their parts".[6]

This thesis draws on theories developed by numerous systems scholars. The most notable of these theories is probably the differentiation between open and closed systems that was

[1] Checkland, P., Holwell, S. 1998. *Information, Systems and Information Systems.* Preface.

[2] Hall, A.D., Fagen, R.E. 1956. *Yearbook of the Society for the Advancement of General Systems Theory*. p 19.

[3] Carvalho, J.A. 2000. *Proceedings of the 1999 Conference on Information Systems Concepts*. p 262.

[4] Jackson, M.C. 2003. *Systems Thinking: Creative Holism for Managers*. p 4.

[5] Jackson, M.C. 2003. *Systems Thinking: Creative Holism for Managers*. p 4.

[6] Jackson, M.C. 2003. *Systems Thinking: Creative Holism for Managers*. p 4.

originally published by Von Bertalanffy, a biologist, in a 1950 article.[1] "Open systems take inputs from their environments, transform them and then return them as some sort of product"[2] while closed systems engage in no interactions with their environments. Naturally, a computer system would be a closed system while an organisation can be viewed as an open system. The idea of open and closed systems is especially important in forming the foundation for research on information systems, as will be done later in this chapter.

The influence of biology on the systems language is of further significance and particularly applicable to this thesis. Besides the differentiation between open and closed systems, Von Bertalanffy also studied the similarity between living systems, like organisms, and open systems in other domains. Lilienfeld argues that his work "has never been purely that of a biologist; it has included philosophical, psychological, and sociocultural themes from the very beginning".[3] It is therefore not surprising that it was "embraced by management thinkers who transferred the open system model to their study of organisations".[4] This thesis discusses discusses the possibility of developing an information system as a living system and explores the theoretical implications of such an approach. Other notable scholars in this regard are Maturana and Varela who challenge Von Bertalaffny's open system concept by emphasising the "closed system of interactions that occurs in living entities".[5] Their theories will be discussed in a later chapter.

1.7.2 Information Systems

In the context of information systems development and, more specifically, the development of CBIS, there are two primary schools of thought regarding the definition of information systems. As with much of the conceptual ambiguity in the field, this conflict results from the intricate relationship between information systems and information technology. In this section these two schools of thought will be compared in terms of their perspective on information system boundaries and subsystems.

---

[1] Jackson, M.C. 2003. *Systems Thinking: Creative Holism for Managers*. p 5.

[2] Jackson, M.C. 2003. *Systems Thinking: Creative Holism for Managers*. p 6.

[3] Lilienfeld, R. 1978. *The Rise of Systems Theory. An Ideological Analysis.* p 21.

[4] Jackson, M.C. 2003. *Systems Thinking: Creative Holism for Managers*. p 6.

[5] Jackson, M.C. 2003. *Systems Thinking: Creative Holism for Managers*. p 7.

1.7.2.1 Information Systems as Computer Systems (IS1)

The first school of thought is that of information systems as computer systems. Definitions that correlate with this school of thought are the following:

- "An information system is a business application of the computer".[1]
- "An information system is any computerised system with a user or operator interface".[2]

This interpretation of information systems probably best resembles conventional knowledge. It bounds the term to the technological infrastructure of an organisation and subsequently implies that information systems are, by definition, closed systems. It is a simplistic perspective that neglects the true nature of information systems and fails to recognise their multi-dimensionality. Avgerou confirms this view by stating that "a distinction between technology as content and society as context is a simplification obscuring the complex processes in which technology and human actors jointly take part to form socio-technical entities".[3]

1.7.2.2 Information Systems as Human Activity Systems (IS2)

The second school of thought adopts a more holistic view of information systems and is in line with the definitions originally proposed by Earl and Buckingham:

- "The set of people, procedures and resources that collects, transforms and disseminates information in an organisation".[4]
- "A system which assembles, stores, processes and delivers information relevant to an organisation (or to society), in such a way that the information is accessible and useful to those who wish to use it, including managers, staff, clients and citizens. An

[1] Whiteley, D. 2004. *Introduction to Information Systems, Organisations, Applications.* p 6.

[2] Carvalho, J.A. 2000. *Proceedings of the 1999 Conference on Information Systems Concepts*. p 275.

[3] Avgerou, C. 2001. *Information Systems Journal.* p 46. The term *socio-technical* is discussed in more detail in section 1.7.2.2.

[4] Earl's definition as referenced by Stamoulis, D.S., Martakos, D.I., Introna, L.D. 1998. *Proceedings of the 1999 European Conference of Information Systems*. p 1017.

> information system is a human activity (social) system which may or may not involve the use of computer systems”.[1]

The scopes of these definitions are much wider. The information system includes stakeholders that interact with the computer system as a subsystem of the information system. It defines an open system that is, unlike the first school of thought, complex and interacts with its environment. Maguire refers to this as a “socio-technical design” that finds a balance between the vision of management, technology and the business processes.[2] These definitions recognise the various dimensions of information systems, articulating the complexity of the academic domain in which they are studied.

1.7.2.3 The Significance of System Boundaries

It would be inaccurate to argue that information systems were originally thought of as computer systems until our knowledge of organisational information management expanded and more holistic definitions were adopted. On the contrary, the more holistic definitions supplied here were already published in 1987 (by Buckingham) and 1989 (by Earl) while Whiteley’s definition was only published in 2004. What is more upsetting is that most scholars tend to agree with the second school of thought, while the conventional knowledge and usage of the term supports the first school. Checkland and Howell’s explanation for this (the inaccuracy of introductory texts on information systems) is confirmed by Whiteley’s definition which appears in such a text.

To illustrate the implications of defining an information system as either IS1 or IS2, it is useful to investigate the system boundaries implied by each school of thought. One might argue that the definition of an information system is of little interest in the bigger process of systems development or use, however the field’s theoretical weakness necessitates that care

---

[1] Buckingham’s definition as referenced by Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 23.

[2] Maguire, S. 2000. *Information Management & Computer Security*. p 234. The term *socio-technical*, according to Magalhàes, has its roots in efforts to decrease the number of system failures in information system implementation by a stronger emphasis on the “human aspects involved in applying information technology to the workplace”. Magalhàes, R. 2004. *Organisational Knowledge and Technology*. p 146.

be taken when the extent of a concept's meaning is discussed, as it inherently leads to assumptions and conclusions being made.

The first school of thought leads to the development of a model as displayed in figure 1.2.

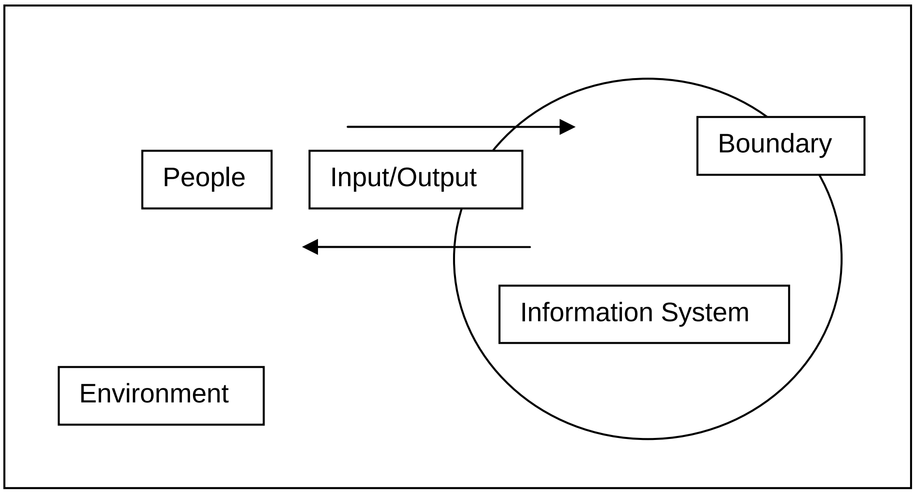

Figure 1.2: A closed information system.

Figure 1.2 illustrates an information system of which the boundary encompasses the computer technology (hardware and software) used in the organisation. The system's stakeholders fall outside the boundary and subsequently become part of the system's environment. In accordance to this, Stamoulis et al. state that "if the scope of the system is constrained to the IT systems which start with the input terminal devices and end with the storage devices, then the input-type description of the system is fine and unquestionable".[1]

The second school of thought produces a model as displayed in figure 1.3.

[1] Stamoulis, D.S., Martakos, D.I., Introna, L.D. 1998. *Proceedings of the 1999 European Conference of Information System*. p 1017.

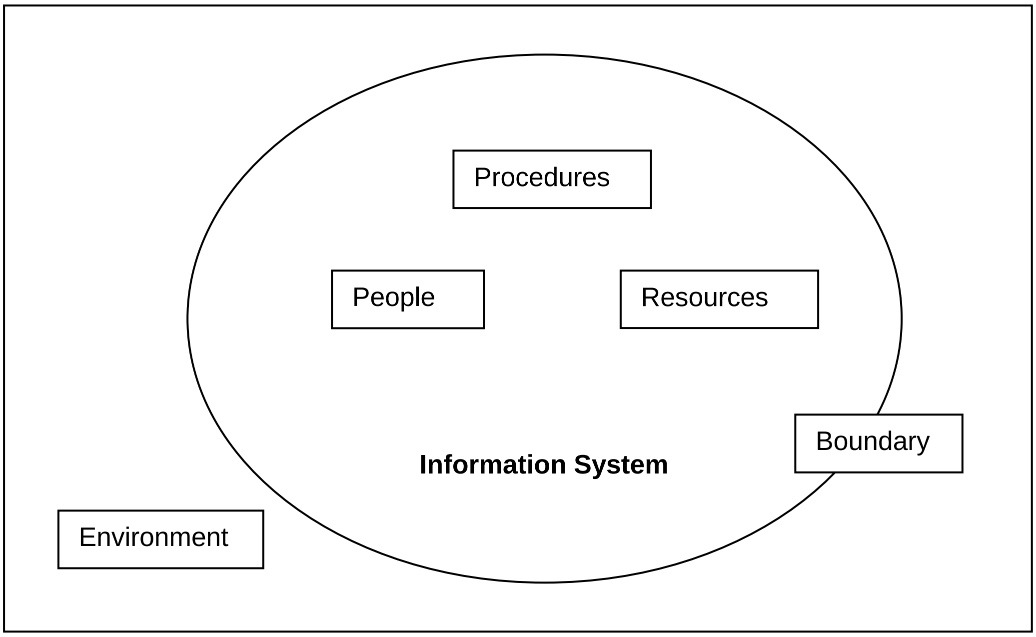

Figure 1.3: An open information system.

Figure 1.3 is significantly different from figure 1.2. It resembles a complex, open system containing three primary subsystems and does not include information technology by definition. The system's boundary does not only enclose the resources used to handle information, but also the people applying these resources and the procedures[1] in which they are applied. Although information technology is not by definition part of this system, the focus of this paper is information systems in contemporary organisational context. It is therefore logical to assume that information technology would form part of the resources subsystem, implying that the information system is computer-based.[2]

The people that form part of IS2 are often referred to as the system's stakeholders. They are impacted, directly or indirectly, by the system in unique ways depending on their specific role. Beynon-Davies identifies 6 groups of stakeholders for an information system[3]. They are:

[1] Lucas, in his definition for information systems, emphasises the central role of procedures. It reads: "an information system is a set of organised procedures that, when executed, provides information to support the organisation." The definition proclaims that the system is not a combination of components, but rather a combination of procedures by which components interact. Lucas, H.C., 1994. *Information Systems Concepts for Management*. p 17.

[2] Willcocks state that "information systems may be more or less IT based, though the obvious developing pattern in the industrialised and industrialising economies is toward the former". Willcocks, L. 1996. *Investing in Information Systems, Evaluation and Management.* p 1.

[3] Beynon-Davies, P. 2002. *Information Systems: An Introduction to Informatics in Organisations*. p 183.

- *Producers*. They are the team of people responsible for the planning and development of an information system. A variety of experts and professional are involved in the development process and may include analysts, programmers, developers, consultants, IT practitioners and information systems development professionals.[1]
- *Clients*. Clients are normally managerial groups within the organisation that sponsor and provide resources for the continuation of an information systems project.
- *End-users*. They are the people who typically rely on the information system during daily operations and business procedures.
- *Customers*. The organisation's customers are directly or indirectly impacted by the information system. They are of particular importance in e-businesses where they can directly interact with computer systems.
- *Suppliers*. Although suppliers are not directly impacted by the information system, an increasing amount of organisations are implementing business-to-business e-commerce solutions that will have an effect on their relationship with suppliers.
- *Regulators*. These are groups or agencies that are responsible for the regulation of information systems procedures.
- *Competitors*. Although competitors will not play a specific role within the information system, they are impacted by and have an impact on an organisation's information system. Beynon-Davies notes that this is especially true for strategic information systems.

1.7.2.4 Differentiating between IS1 and IS2

Laudon and Laudon sharply differentiate between IS1 and IS2 by stating that:

> "Although computer-based information systems use computer technology to process raw data into meaningful information, there is a sharp distinction between a computer and a computer program on the one hand, and an information system on the other. Electronic computers and related software programs are the technical foundation, the tools and materials, of modern information systems. Computers provide the equipment for storing and processing information. Computer programs, or software,

[1] Goulielmos, M. 2004. *Information Systems Journal.* p 364.

are sets of operating instructions that direct and control computer processing. Knowing how computers and computer programs work is important in designing solutions to organisational problems, but computers are only part of an information system".[1]

Their view is supported by Beynon-Davies who argues that "information technology systems are component elements of modern information systems" but that information systems are primarily concerned with the "communication between a group of people".[2] Gupta expresses the same view on the relationship between information technology and information systems by stating that "technologies by themselves do not do anything" and that it is only when they are "applied in meaningful ways" that they can contribute to the organisation.[3] Magalhàes, however, note that the separation between information systems and information technology is difficult and sometimes impossible.[4] "The distinction is far from being clear-cut and for that reason the dual acronym IS/IT (information systems/information technology) is often used in the literature".[5]

In this thesis, information systems as defined in the first school of thought (IS1), and information systems as defined in the second school of thought (IS2), are both relevant. Tailorable Information Systems are by definition social systems (IS2) that contain computer systems (IS1) as subsystems. The reader should be aware that a lot of the literature cited in this paper was authored by IT experts or professionals whose scholarly correctness in the use of concepts is, at times, questionable. Some authors define and use the term "information system" purely as IS1 while others tend to define it as IS2, yet in the text use it as IS1.

A good example of the confusion resulting from these two schools of thought is the use of the term "Systems Development Life Cycle" (SDLC) which is a development approach that will be discussed in more detail in the next chapter. Avison & Fitzgerald call it the "*Information* Systems Development Life Cycle"[6], yet the same approach is commonly known as the

---

[1] Laudon, K.C., Laudon, J.P. 2004. *Management Information Systems*. p 9.

[2] Beynon-Davies, P. 2002. *Information Systems: An Introduction to Informatics in Organisations*. p 159.

[3] Gupta, U.G. 1999. *Information Systems Success in the 21st Century*. p 17.

[4] Magalhàes, R. 2004. *Organisational Knowledge and Technology*. p 4.

[5] Magalhàes, R. 2004. *Organisational Knowledge and Technology*. p 4.

[6] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 31.

"Software Life Cycle"[1] in software engineering literature. The approach focuses on the development of a technological infrastructure for information systems, hence the conflicting interpretations.

To allow the reader to interpret this thesis as accurately as possible, it is essential to clearly stipulate the context within which it is written. Consequently, two assumptions are made:

- Firstly, it is written from the perspective that an information system is a social system (IS2), which, by default, harnesses computer systems as a crucial resource (thus CBIS) and that it is not possible to develop an "effective information system incorporating significant amounts of technology, without treating it as a social system".[2] The definitions proposed by Earl and Buckingham will be accepted working definitions but the reader must be constantly alert for conflicting interpretations by cited authors.

- Secondly, the reader should take note that information systems are implemented in virtually every professional domain. This thesis, however, is primarily concerned with information systems that govern the flow of information in organisations that operate in the turbulent business environments of the knowledge economy and experience dynamically changing system requirements. There exist a myriad of information systems that operate in stable environments and are successfully implemented with a teleological paradigm; this thesis is not concerned with them.

By making these two assumptions and supplying a theoretical framework for the interpretation of key concepts in the remainder of the thesis, I hope to communicate arguments void of ambiguity.

---

[1] Boehm, B. 1988. A Spiral Model of Software Development and Enhancement. *Computer*, p 61.

[2] Maguire, S. 2000. *Information Management & Computer Security*. p 231.

Chapter 2

The Teleological Paradigm

2.1 Introduction

Stamoulis et al. state that the dominant contemporary systems development methodologies are defined by "teleology".[1] The term has its roots in the Greek word "telos" meaning "end". In an information systems development context, teleology is probably best translated as "goal-directed" and implies that the development process is driven by an accurate and complete understanding of the deliverable product to be produced. Such an understanding is formed during the analysis and design phases of a development project and culminates as a set of information system requirements. Following the teleological paradigm, information systems are approached as products that must adhere to these requirements to be successful. The methodologies in this paradigm, often referred to as "hard methodologies"[2], are aimed at producing an accurate set of system requirements before any actual development commences to ensure that the system's stakeholders are satisfied and the system is successful. Fitzgerald summarises the teleological paradigm successfully:

> "The underlying paradigm for many development methodologies is the scientific reductionist one. They rest on the a priori assumptions that the solution can be arrived through a sequence of technically devised steps, and that the developer can obtain detailed knowledge about the situation".[3]

Before analysing these methodologies in some detail, however, it is worth considering their origins and forming a basic understanding as to what constitutes an information systems development methodology.

[1] Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 295.

[2] Maguire, S. 2000. *Information Management & Computer Security*. p 230.

[3] Fitzgerald, B. 1996. *Information Systems Journal*. p 14.

When organisations originally implemented computer systems, systems development was void of any structured methodology and relied on "unsystematic and random methods".[1] Programming languages used in the early 1960's were not as modular as the object-orientated[2] ones available today and system developers spent a lot of time searching through code to make requested software changes. These technological difficulties often meant that systems lacked user-friendliness and users required a high level of system knowledge to use the system efficiently. The advances in computer technology allowed systems development theory to evolve and sparked the birth of systems development methodologies. Avison and Fitzgerald point out three significant changes:

1. Developers realised the importance of systems analysis and design in developing maintainable systems. Naturally, more time was spent on systems planning before the actual software coding started. This meant that programmers were not always involved in the design of the system but rather followed instructions from a systems analyst. Analysts could now spend more time eliciting user requirements and needs to ensure a more acceptable system. Programmers, on the other hand, were freed from communicating with users directly and taking responsibility for system acceptance; they could now concentrate on coding more efficient systems.
2. Developers recognised the need of organisations to move away from once-off solutions to more integrated solutions that could run for longer periods of time. A system that required regular changes led to frustration on the side of both users and programmers.
3. As information systems development became a common practice, the need for an accepted methodology grew. Such a methodology would not only serve to assist developers but also ensure that the needs of users are taken into account during the development process and that the final product is satisfactory.

[1] Fitzgerald, B. 1996. *Information Systems Journal*. p 4.

[2] Object-orientated is a programming paradigm that approaches the problem to be solved at application level, in terms of objects and interactions needed to describe the application. Object-orientated programming owes much of its current form to the notion of objects and classes introduced by Simula, a language that originated in simulation. Sethi, R. 1996. *Programming Languages. Concepts and Constructs*. p 15.

Information systems development methodologies have become immiscible in the efforts to develop systems that are on time, within the budget, and ultimately successful. A variety of these methodologies exists today and they are used as strategies which imply a subdivision of the development process.[1] Avison and Fitzgerald define an information systems development methodology as:

> "A collection of procedures, techniques, tools, and documentation aids which will help the system developers in their efforts to implement a new information system. A methodology will consist of phases, themselves consisting of sub-phases, which will guide the system developers in their choice of techniques that might be appropriate at each stage of the project and also help them plan, manage, control and evaluate information systems projects".[2]

Systems development methodologies must be understood within the socio-technical context in which they are implemented. It is extremely rare that any methodology is applied according to the exact specifications of its creators. In his case studies on the use of methodologies in practice, Goulielmos found that "it is clear that there is no pure use of methodologies".[3] He argues that methodologies assist the development process but should not receive too much emphasis in the discipline of systems development. The ultimate success or failure of an information system depends on a range of technical and social factors both within and outside the organisational boundaries. Goulielmos notes that information systems do not succeed or fail "because of methodologies, but because of the way people use them".[4] This view, however, has been opposed in literature. Some researchers argue that more rigorous implementation of methodologies is the most effective way to combat the high rate of system failures.[5] Fitzgerald describes this view accurately:

> "An examination of the literature reveals a two-fold bias, which, firstly, construes the software crisis as a problem arising from the sloppy, ad hoc and irrational approaches of system developers in practice; and secondly, views the

[1] Fitzgerald, B. 1996. *Information Systems Journal*. p 6.

[2] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 24.

[3] Goulielmos, M. 2004. *Information Systems Journal.* p 381.

[4] Goulielmos, M. 2004. *Information Systems Journal.* p 381.

[5] Fitzgerald, B. 1996. *Information Systems Journal*. p 3.

solution to the software crisis in terms of more widespread adoption of rigorous and formalised systems development methodologies".[1]

The term "technique", in this context, refers to "a way of doing a particular activity in the information systems development process"[2] and the term "tool"[3] refers to "the artefacts used in information systems development".[4] The various methodologies available today differ in their main objectives and the circumstances in which they are most effective. Some of these objectives include:

- To accurately record the user requirements of the system.
- To provide a systematic method of development so that progress can be effectively monitored.
- To provide an information system within an appropriate time limit and at an acceptable cost.
- To produce a system that is well documented and easy to maintain.
- To provide an indication of any changes that need to be made as early as possible in the development process.
- To provide a system that is liked by the people affected by it.

In the remainder of this chapter I will further analyse the teleological development paradigm by focusing specifically on the SDLC as the systems development approach that originally introduced a structured systems development. Two other methodologies that are prominent in contemporary development projects are briefly discussed thereafter. This is followed by a discussion on the general criticism associated with the teleological paradigm and, thereafter, a more in-depth exploration of the implications of dynamic requirements for information systems developed through this approach. According to the *Fallacy of Correct User*

---

[1] Fitzgerald, B. 1996. *Information Systems Journal*. p 3.

[2] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 24.

[3] Satzinger et al. specifically define tools as "*software* support that helps create models or other components required in the project." CASE (Computer-Aided Systems Engineering) tools are the most comprehensive of these and facilitates the development of diagrams, models and computer code in an integrated manner. Satzinger, J.W., Jackson, R.B., Burd, S.D. 2004. *Systems Analysis and Design in a Changing World*. p 47.

[4] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 25.

Requirements[1], any methodology that bases systems development and implementation on system requirements will produce information systems that are irresponsive to organisational change. This results in the ongoing maintenance of the information system and associated organisational dependencies which are analysed in the final part of the chapter.

2.2 The System Development Life Cycle (SDLC)

2.2.1 Background

The SDLC is the result of work done by members of the United Sates Department of Defence in the 1960's. Henry Rowan and A. Enthoven were developing a heuristic process for managing large information systems projects which developed into "a linear list of stages" for development projects.[2] These SDLC has strongly influenced systems development theory and practice over the last 30 years as a great number of the prominent methodologies used today stem from it. It has the following stages[3]:

1. *A feasibility study*. The feasibility study "looks at the present system, the requirements that it was intended to meet, problems in meeting these requirements, new requirements that have come to light since it was first implemented, and briefly investigates alternative solutions". An information system must be feasible in a legal sense, an organisational and social sense, a technical sense, and in an economical sense.[4]

---

[1] This phrase was coined by Stamoulis et al. Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 295.

[2] As discussed in an article by Boggs on the SDLC and Six Sigma. Boggs, R.A. 2004. *Issues in Information Systems*. p 37. Magalhàes confirms the engineering roots of the SDLC by arguing that its "major emphasis is on how to make technology work, that is, how hardware, software and data can be utilised to serve a particular organisational need". Magalhàes, R. 2004. *Organisational Knowledge and Technology*. p 138.

[3] Many variants of the SDLC exist today; these are the phases as discussed by Avison and Fitzgerald. Satzinger et al. only describe 5 phases. Satzinger, J.W., Jackson, R.B., Burd, S.D. 2004. *Systems Analysis and Design in a Changing World*. p 37. Magalhàes state that the number of stages of published SDLC variations, range from four to seven or more. Magalhàes, R. 2004. *Organisational Knowledge and Technology*. p 138.

[4] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 31.

2. *Systems investigation.* During this phase the system analysts obtain detailed information of both old and new system requirements through thorough investigation of the current system and a series of interviews, questionnaires, observations and samples. This phase produces a combination of documents much more detailed than that of the feasibility study, pointing out the possible system constraints, exception conditions and problems with the current working methods.
3. *Systems Analysis.* This phase investigates all the methods and associated problems of the current system. It should serve to give the system analyst an understanding of the reasons for a new system and indicate to him why certain working methods have developed within the information system. Naturally, this will serve to further stipulate the requirements of the new system.
4. *Systems Design.* The design of the new system is done during this phase. It will produce documentation that outlines the data used in processes, the processes themselves, structure of information sources, security and back-up measures and system testing and implementation plans.[1]
5. *Implementation.* In this phase the computer programmers will develop the actual software packages that have been designed. The organisation must buy the required computer hardware as specified by the system analysts. When the technological infrastructure is in place, the new system can replace the old one. This is referred to as cutover and it is suggested that it happens in stages to avoid confusion and data loss. User training and the production of documentation such as manuals are crucial to ensure user acceptance of the new system and a smooth cutover.
6. *Review and maintenance.* This is the final phase of the SDLC and takes place after the new system is operational. Developers monitor the new system and correct errors that occur or make changes where a requirement was misunderstood or forgotten. This phase is usually the source of frustration for both users and developers as users need to cope with continuous system changes while developers have to adapt program code.

Each phase of the SDLC is dependent on its predecessor. The output of the system investigation, for instance, serves as input for the systems analysis. Thus the SDLC is often referred to as a sequential model. Many systems development models closely resemble the

[1] Avison, D., Fitzgerald, G. 2006. *Information Systems Development.* p 33.

SDLC, of which the most notable is the Waterfall model. Introduced by Winston Royce in 1970, the model also follows a set of sequential phases and is often referred to as the Classic Life Cycle.[1]

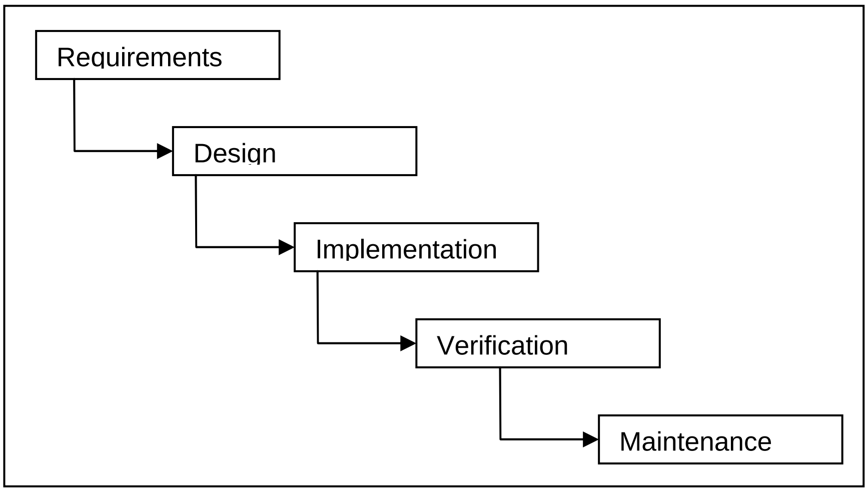

Figure 2.1: The Waterfall model

The SDLC "is being used successfully"[2] in many organisations and has "a number of features to commend it".[3] The approach is useful for "building large complex, systems that require a rigorous and formal requirements analysis, predefined specifications, and tight controls over the systems-building process".[4] Any business capable of accurately stipulating their information systems requirements can successfully implement the SDLC, especially if the business environment is stable for extensive periods of time. If followed correctly, the phases of the SDLC ensure that specifications for a new system are recorded accurately. This is especially true if the phases are documented using the correct standards.[5] Documentation serves to ensure clear communication between developers and users, allowing developers to produce a system that meets user expectations. By breaking the systems development project up into phases, developers can do a review after each phase, allowing them to consider the outcomes of the phase.[6] Such a review will allow investors to monitor the development and give their input where necessary. Time and budget constraints for every phase of the project

[1] Boggs, R.A. 2004. *Issues in Information Systems*. p 37.

[2] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 43.

[3] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 38.

[4] Laudon, K.C., Laudon, J.P. 2004. *Management Information Systems*. p 395.

[5] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 38.

[6] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 38.

can be set out, allowing developers to manage the overall project timeline and ensure completion before a deadline.

2.2.2 General criticism

The use of traditional development approaches such as the SDLC has, however, "garnered many criticisms" [1] regarding their application. Such criticism, sparked by the high level of failing development projects[2], focus on various aspects of the SDLC and other traditional development methodologies.

- Although the SDLC allows project managers to plan phases according to a time-line, the estimations made at the outset of development projects "can be unreliable because of the complexity of some phases and the inexperience of the estimators".[3] This problem is further complicated by the fact that "later phases depend on the successful completion of earlier phases which requires perfect foresight".[4] This is especially true true when poor communication between users and developers result in a misunderstanding of the system requirements. As a result of this, "computer people have been seen as unreliable and some disenchantment have ensued in relation to computer applications".[5] A lack of understanding on the side of the user plays an equally important role. Users are unfamiliar with the technical difficulties that developers face in each phase and their idea of a system requirement is often completely different than that of a developer.

- Notoriously, the SDLC has not produced what Avison and Fitzgerald refer to as "complete" systems. What they imply is that systems neglect to cater for exceptions but rather focus on the routine, basic tasks of the organisation. Exceptions are scenarios that do not occur frequently and clash with standard system procedures. They are often expensive to cater for from a development point of view, as they might

[1] Kiely, G., Fitzgerald, B. 2005. *The Electronic Journal on Information Systems in Developing Countries.* p 1.

[2] Kiely, G., Fitzgerald, B. 2005. *The Electronic Journal on Information Systems in Developing Countries.* p 2.

[3] Avison, D., Fitzgerald, G. 2006. *Information Systems Development.* p 41.

[4] Fitzgerald, B. 1996. *Information Systems Journal*. p 14.

[5] Avison, D., Fitzgerald, G. 2006. *Information Systems Development.* p 41.

involve intricate workarounds that take a long time to develop. When exceptions are ignored or forgotten, the final system is falsely believed to be complete.

- Although the various phases of the SDLC are critical in their purpose, they result in a much longer process than users expect. Some systems can take several years to be developed and many development projects are cancelled as a result of this. An application backlog is not unfamiliar to developers and they are often unable to respond timely to client requests. This goes hand in hand with the fact that many developers are tied up in the maintenance of older systems. Maintenance, which occurs in the final phase of the SDLC, ultimately becomes a full-time responsibility for many developers. Organisations often employ a team of developers to ensure that their systems are well-maintained and user requirements can be met in relatively short periods of time.

- Finally, the underlying assumption on which incremental development processes, like the SDLC, are based must be questioned. In the complex, turbulent organisational environments that become the contexts of information systems development projects, the principle of requirements-driven development is seen to be flawed. The notion that an accurate and complete understanding of the information system requirements can be devised "is somewhat naïve" and systems development in such practices will almost certainly be an "iterative process".[1]

In defence of the SDLC, it is important to place it in the correct slot on a timeline of systems development theory. It was the first systems development approach and was developed when computer technology was unable to produce anything close to what is possible today. Regardless, it has dominated systems development theory and is still prominent in any text on the subject. In the early 1990's more models started to appear that supported iterative design and development, but the emphasis still stayed on system requirements as a driving force for development, thus a scientific reductionist approach. In an article published in 2001, Stamoulis et al. claim that "there can be no argument that each and every method is based on the same basic principle of getting the stakeholder requirements right first so as to proceed

[1] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 42.

and build the system".[1] This view is again expressed by Kanellis and Paul in 2005 when they state that "any approach, methodology, or group of tools begin with and base their eventual success on one objective: to achieve a correct set of system requirements".[2] Before critically evaluating this statement by discussing some of the development methodologies developed through the 1990's, it is crucial to explain why Stamoulis et al. argue that a system developed based on fixed user requirements (i.e. goal-directed development) is bound to be unsuccessful.

2.3 The Fallacy of Correct User Requirements

"Requirements are regarded by many as the most important and crucial part of the systems development process"[3] and has traditionally been at the heart of systems development methodology.[4] It is the anchor on which the entire development project depends and "errors made in this phase have a potentially high impact on the eventual information system".[5] The process of eliciting the requirements for a new information system, often referred to as requirements analysis or engineering, is not only a fundamental part of systems development but also a field of study and a full-time occupation to many. Accordingly, a common belief is that system success depends mainly on the accuracy of pre-defined system requirements made during the pre-implementation phases of the SDLC and similar reductionist approaches.

Requirements analysis draws on a series of techniques (interviews, meetings, surveys, workshops, models, diagrams, prototypes etc.) in an attempt to capture the requirements for a new system in a coherent whole that reflects the needs of all the system stakeholders. During this process, which may take several years to complete, system analysts compile documentation that stipulate how the new system must be developed and implemented. From an engineering perspective, this requirements driven approach makes perfect sense. However,

---

[1] Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 295.

[2] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 67.

[3] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 99.

[4] Baskerville, R., Pries-Heje, J. 2004. *Information Systems Journal*. p 245.

[5] Flynn, D.J., Jazi, M.D. 1998. *Information systems Journal*. p 53.

the problem of developing information systems in contemporary organisations is, as Patel puts it, not so much an engineering problem as a living problem.[1]

Many statistics serve to support the fact that system maintenance requires ongoing, often extremely large, investments.[2] None more so than the fact that between 70% and 85% of information systems costs in organisations are spent during the maintenance phase of the SDLC, while only the remaining 15% to 30% are spent on new systems.[3] These statistics concur with a study which reports that 50% of design activity in the systems development project does not occur in the design phase.[4] System changes can be up to 100 times more expensive to implement during the maintenance phase as opposed to the design phase.[5] Graphs that depict these statistics indicate an increase in cost to implement system changes as the SDLC stages follow each other with a considerable increase in costs after the implementation phase. A simplified version is shown in figure 2.2.

[1] Patel, together with Stamoulis, Kanellis and Martakos, has done groundbreaking research in the field of Tailorable Information Systems. Patel, N.V. 1999. Developing Tailorable Information Systems through Deferred System's Design. *Proceedings of the 1999 Americas Conference on Information Systems*. p 4.

[2] It should be noted here that some information systems are developed to be more *maintainable* than others. Thiadens defines *maintainability* as "the swiftness with which one is able to trace faults in IT facilities; the testability of the facility and its reusability". Thiadens, T. 2005. *Manage IT.* p 28.

[3] As reported by Sprague in 1993 and referenced by Stamoulis et al. Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 295. This statistic is confirmed by Leffingwell in 1997 as referenced in Avison and Fitzgerald Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 99.

[4] Fitzgerald, B. 1996. *Information Systems Journal*. p 15.

[5] Avison and Fitzgerald state that Pfahl et al (2000) reports the following: "Finding and fixing a software problem after delivery is 100 times more expensive than finding and fixing it during the requirements and early design phases." Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 99.

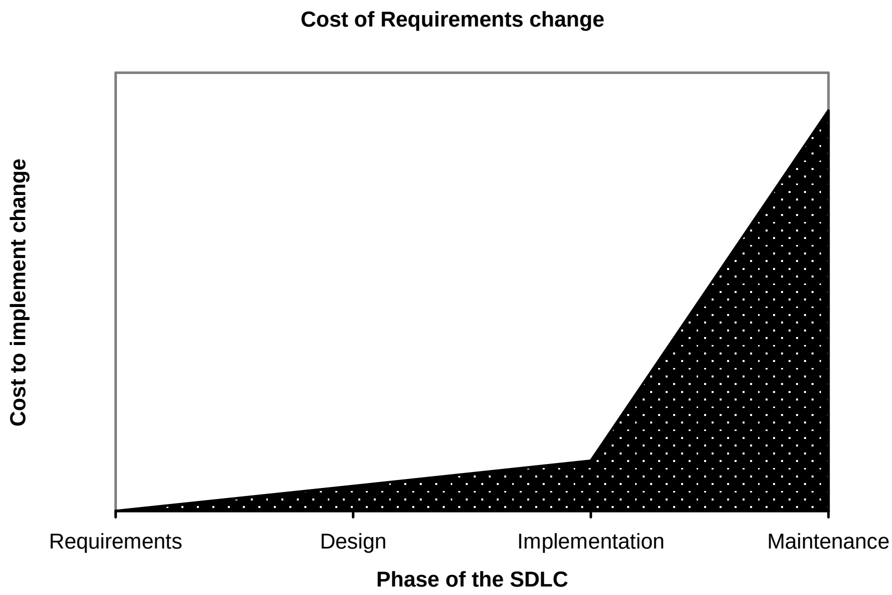

Figure 2.2: The cost of implementing requirements changes in the phases of the SDLC.

The complexity of advanced technology used in contemporary information systems makes it extremely difficult for developers to implement changes during or after the system has been integrated into its working environment. Maintenance procedures most likely result in system *downtime*[1] that has further financial consequences (hence the high costs). This forces developers to work with a predefined set of requirements that stipulate system functionality and operation, making the entire project dependent on the accuracy of those requirements. In what they refer to as *The Fallacy of Correct Information System Requirements*, Stamoulis et al. state that the exercise of eliciting system requirements leads to a variety of "false paradigms for information systems development".[2]

The complexity of requirements analysis is gravely underestimated and often proclaimed as the reason for many unsuccessful information systems development projects.[3] Theory tends to over-simplify the process, not recognising the intricacies that are produced by the complex social dimensions within an organisation in a competitive business environment. This notion has sparked a wave of literature that proclaims a stronger emphasis on the social context

[1] The term *downtime* typically refers to a period of time during which system maintenance tasks are performed and the system is offline or not functioning as a result.

[2] Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 296.

[3] Avison and Fitzgerald discuss the problems associated with requirements analysis in a detailed list; I will only provide a couple of paragraphs which focus on the main problems.

within which the information system will operate. Gallivan and Keil state that "user participation in systems development has been one of the most widely researched topics in the information systems literature, and is widely viewed as instrumental in the successful design and implementation of information systems".[1] Researchers have focused specifically on the involvement of users during the early phases of the SDLC to ensure that the system's social context is analysed by the development team before implementation gets under way.

The inability of analysts to identify all the system stakeholders, for instance, results from the disregard of the social dimension of systems development. When system requirements are finalised they might be correct, yet lack completeness as a result of this error. Incomplete requirements will result in a system that is only useful to a certain group of stakeholders, yet ultimately inadequate. This problem is often intensified by the fact that certain stakeholder groups might be reluctant to participate in requirements elicitation activities. Whether it is as a result of time constraints or other agendas, it inevitably means that the requirements, and ultimately the system, will be incomplete.

The process of analysing the social context of the system can be further complicated by communication gaps between the development team and stakeholders, something which only a few studies have examined.[2] Most system analysts are technically qualified and tend to approach system requirements from a technological perspective, often failing to understand the needs of the user who lacks technical sophistication. This is especially true when analysts use technical terminology during interviews or workgroups, failing in their attempt to develop a thorough understanding for the role of each stakeholder group from a user perspective. Flynn and Jazi refer to it as a "user-developer culture gap"[3] and supply two main reasons for its existence:

- A lack of domain and business knowledge by developers.
- Developers have an engineering viewpoint with a bias towards technical concerns.

A communication gap, however, can also emerge as a result of users that over-elaborate on requirements. Instead of focusing on crucial system requirements, users compile lists of

[1] Gallivan, M.J., Keil, M. 2003. *Information Systems Journal.* p 38.

[2] Gallivan, M.J., Keil, M. 2003. *Information Systems Journal.* p 41.

[3] Flynn, D.J., Jazi, M.D. 1998. *Information systems Journal.* p 53.

things that would be “nice to have” in their personal opinions. As a result, analysts often have difficulty distinguishing between real system requirements and special system features. Gallivan and Keil warn that certain users might be more vocal than others and their opinions and preferences will often receive more attention, while those of less vocal users are of equal or higher importance.[1] In the same way some users tend to over-elaborate, others often fail to disclose all their requirements. Especially if a new information system will influence their business processes and they are unaware of what their information requirements will be during the new processes. Another common occurrence is disagreement among users as to what their requirements are. This puts analysts in a particularly tough spot, often having to choose between requirements, interpreting one group’s opinion to be more accurate than another’s.

The process of eliciting user requirements often leads to the discovery of the need for business process changes at other organisational levels. During requirements specification, analysts monitor the flow of information through the organisation by studying the inputs, outputs and operations of different business processes. They are often forced to decide whether the information system should shape itself around the existing business processes or whether business processes can be re-engineered more effectively.[2]

Communication within the project team, specifically between analysts and programmers can also obstruct system success. During requirements specification, analysts produce documents detailing requirements. These documents typically include various diagrams that are interpreted by programmers when coding the actual software.[3] The inability of an analyst to produce these diagrams or the inability of a programmer to accurately interpret them will lead to an incomplete or inaccurate system. In a scenario where incomplete requirements specification leads a programmer to make assumptions, he is more likely to follow his

[1] Gallivan, M.J., Keil, M. 2003. *Information Systems Journal.* p 64.

[2] On this point it is important to keep the system’s end-users in mind. Although the SDLC has its roots in software development, it recognises the social dimension of the information system by placing such emphasis on user requirements.

[3] The Unified Modeling Language (UML) is a combination of nine distinct types of diagrams that serves to create an abstract model of a system. Although most analysts only partly utilise it, Deborah Hess describes it as “a sleeping giant with significant potential to ease application development, maintenance and even integration”. Hess, D.S. 2001. *UML: An Introduction*. p 1.

personal judgment than to acquire the correct specifications from analysts or users; this mostly happens out of fear for project delays.[1]

The view that information system failures is a result of the development team's inability to involve stakeholders during the analysis and design phases of the SDLC has been the focal point around which various newer methodologies (see section 2.4) have been developed. Gallivan and Keil, however, argue that "despite the pervasiveness of these beliefs, researchers have repeatedly noted the lack of consistent empirical results in this area" and that it is therefore dangerous to draw parallels between system success and user participation.[2]

Stamoulis et al. argue that the fundamental problem which underlies the complexity of an information system's context is not as much the lack of user involvement, as it is the dynamic nature of the context. The ongoing changes that have become part of contemporary organisational culture have a direct impact on the requirements of the information system. By asking stakeholders what their information system requirements are, we force them to specify their requirements at a certain point in time, yet we aim to develop a system to work over time continuum. In other words, our system relies on the user requirements to be the same for the duration of the system's lifetime. Flynn and Jazi express the problem as follows:

> "Developers and development methods tend to assume that requirements are completely known at the beginning of the requirements process and never change".[3]

If an organisation was a closed, mechanistic system that operated independent of its environment and retained its internal composition for the duration of its existence, we could rely on requirements elicitation as development driving force. Likewise, if we were certain that the organisation's environment would remain stable, goal-directed development would be highly successful.

---

[1] Avison, D., Fitzgerald, G. 2006. *Information Systems Development.* p 103.

[2] Gallivan, M.J., Keil, M. 2003. *Information Systems Journal.* p 37.

[3] Flynn, D.J., Jazi, M.D. 1998. *Information systems Journal.* p 54.

It is worth noting here that the first information systems were developed in a time when the business environment was indeed less turbulent. Organisational theory in the 1960's was still strongly influenced by the idea of an organisation functioning like a machine. This organisational metaphor has its roots in the industrial revolution when organisations required manual labour to perform repetitive tasks in a machine-like fashion.[1] Morgan states that organisations were "designed like machines and their employees were in essence expected to behave as if they were parts of machines".[2] Fitzgerald notes that "early development methodologies were very much influenced by technical and engineering disciplines", approaching the information system as another mechanic part of the organisational machine.[3] Although change was a factor, the business environment was far less turbulent than that of the knowledge economy in the 21st century. Organisational theory has evolved accordingly. A growing awareness of the organisational ability to adapt to change is reflected in the organisational metaphor of an organism. This implies that organisations are not only open systems, but also living systems, relying on their ability to adapt to changing environments for survival.

With such a dramatic change in thinking about organisational systems, it is surprising that information systems development theory in the 1990's were still dominated by the same approach that served mechanical organisations 30 years earlier. To draw once more on Morgan's metaphors in an information systems context: the SDLC was designed for the development of machine-like, technological systems and not for open, socio-technical information systems required in contemporary organisations. Fitzgerald states that this "tendency in Western society to view certain procedures as universally applicable" has been criticised by researchers.[4] Stamoulis et al. argues that "a dead system is a malfunctioning clog in the great organisational engine".[5] Consider the information system as a sub-system of the organisation: As the business environment change and the organisation adapts itself to it, the

[1] Such organisations, often referred to as bureaucracies, are characterised by a large number of management layers. They employ a strict and often large set of rules to protect individuals and minimise uncertainty. Their strong reliance on fixed structures and procedures inhibit their ability to adapt readily to environmental change. Lucas, H.C., 1994. *Information Systems Concepts for Management*. p 38.

[2] Morgan, G. 1997. *Images of Organisation*. p 20.

[3] Fitzgerald, B. 1996. *Information Systems Journal*. p 6.

[4] Fitzgerald, B. 1996. *Information Systems Journal*. p 17.

[5] Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 296.

information system's environment (an adapting organisation) changes as a direct result. If the information system is a dead system, bounded by a pre-defined set of requirements, it will not survive in a changing environment. Bjerknes confirms this by stating that "information systems operate not in isolation but in environments consisting of people, complex organisational structures, technologies and markets."[1]

Avison and Fitzgerald refer to this problem as "changing or evolving requirements" and state that "requirements can and do change while the system is being developed and this is a major problem because specification has been frozen".[2] It is the opinion of Stamoulis et al. that changing requirements are not only bound to the period in which development takes place, but continues after implementation for as long as the system is used. Xia and Lee also emphasise this problem by arguing that because "both information technology and business environments are fast changing, it becomes increasingly difficult to determine business requirements with any certainty, making systems development a progressively more dynamic and complex process".[3]

Many methodologies aim to address changing requirements during the pre-implementation phases, while post-implementation changes are generally regarded as system maintenance. As a result, system survival becomes completely dependent on the maintenance phase or, as Avison and Fitzgerald argue, "maintenance will always be with us and is a normal part of systems development and operational systems".[4] Stamoulis et al. probably best describe the problem of changing requirements when stating that "information systems provide the glue amongst the mission, business processes and structure that define an organisation, a change in one will affect all the others, ultimately leading to organisational incompetence within the macro environment that is operating".[5]

Together with turbulent business environments, organisations must cope with constantly advancing technology. New technology can vastly influence their approach to information

[1] Stamoulis et al. quote Bjerknes when discussing the Fallacy of Correct Requirements. Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 296.

[2] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 104.

[3] Xia, W., Lee, G. 2005. *Journal of Management Information Systems*. p 46.

[4] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 105.

[5] Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 296.

system requirements. The influence of touch screen technology on the restaurant industry is a good example. If a restaurant manager considered the development of a new information system before touch screen technology was available to the public, the knowledge of such technology might pursue him to postpone development until touch screens could be implemented. Once touch screens come onto the market, the development of newer technologies continues and he might postpone systems development even further. Whenever he decides to commence the systems development project, technology will continue to develop.

Stamoulis et al. state that living businesses require “information systems with the inherent ability to flex and adapt to unforeseen circumstances”[1], the keywords here being “flex” and “unforeseen”. Flexibility has always been an important theme in information systems development and notoriously “computer applications have proved very poor at responding to changes in the environment”.[2] The concept is used in various contexts and takes on different meanings as a result. In the context of information systems, flexibility will be a product of both the social and technical dimensions of the system, describing the system’s overall ability to “prepare for and manage an uncertain future in a more proactive way”.[3] It is clearly impossible for stakeholders or system analysts to predict or know the correct requirements of an information system that is meant to run for a considerable period of time in a turbulent business environment.

The complexity of information systems development projects has received an extensive amount of interest by researchers who attempt to make sense of the factors that influence the development process. Xia and Lee argue that these projects “are uniquely more complex than other types of projects” and that the management of their complexity “has become one of the most critical responsibilities of information system managers and executives”. [4] In their study they define information systems development project (ISDP) complexity as “the characteristics of the project that constitute difficulties for the development effort”.[5] Their

[1] Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 296.

[2] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 71.

[3] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 71.

[4] Xia, W., Lee, G. 2005. *Journal of Management Information Systems*. p 47.

[5] Xia, W., Lee, G. 2005. *Journal of Management Information Systems*. p 53.

research culminates in a conceptual framework for ISDP complexity in which they identify the following key dimensions[1]:

- *Structural complexity*. The variety, multiplicity, and differentiation of project elements; and interdependency, interaction, coordination and integration of project elements.
- *Dynamic complexity*. The uncertainty, ambiguity, variability, and dynamism, which are caused by changes in organisational and technological project environments.
- *Organisational complexity*. The complexity of the organisational environments surrounding the project.
- *Technological complexity*. The complexity of the technological environment of the ISDP.

They emphasise the importance of dynamic complexity in contemporary ISDPs as a result of the "unprecedented rapidity" at which environments, both business and technology, are changing.[2]

The discussion so far of traditional development practices has focused mainly on the SDLC. It is, however, not the only approach that falls within the teleological paradigm. Other methodologies were developed and adopted by organisations influenced by the "widespread belief in literature that the formalisation of practice by adherence to methodological approaches" is the most effective way to develop systems. It is worth briefly discussing two of the models that became widely accepted in the 1990's, to critically evaluate Stamoulis et al.'s statement that "there can be no argument that each and every method is based on the same basic principle of getting the stakeholder requirements right first so as to proceed and build the system".[3]

---

[1] Xia, W., Lee, G. 2005. *Journal of Management Information Systems*. p 54.

[2] Xia, W., Lee, G. 2005. *Journal of Management Information Systems*. p 55. Magalhàes supports this view by arguing that the business environment is changing so fast that "by the time the various stages of the methodology have been completed, the initial assumptions made about the business will have been out of date". Magalhàes, R. 2004. *Organisational Knowledge and Technology*. p 145.

[3] Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 295.

2.4 More Methodologies

2.4.1 The Spiral Model

The problems resulting from requirements elicitation and the associated risks, lead Barry Boehm to the definition of the Spiral model in 1985. The model is particularly aimed at risk management and has the ability to lower development cost by allowing clearer communication between developers and stakeholders. Boehm argues that the model's risk-driven approach "incorporates many of the strengths of other models and resolves many of their difficulties".[1]

It works on the basis of cycling through four quadrants in a repetitive style. The quadrants are (starting at the bottom-left in figure 2.3):

1. Plan next phases.
2. Determine objectives, alternatives and constraints.
3. Evaluate alternatives, identify and resolve risks.
4. Develop and verify next-level product.

[1] Boehm, B. 1988. A Spiral Model of Software Development and Enhancement. *Computer*, p 61.

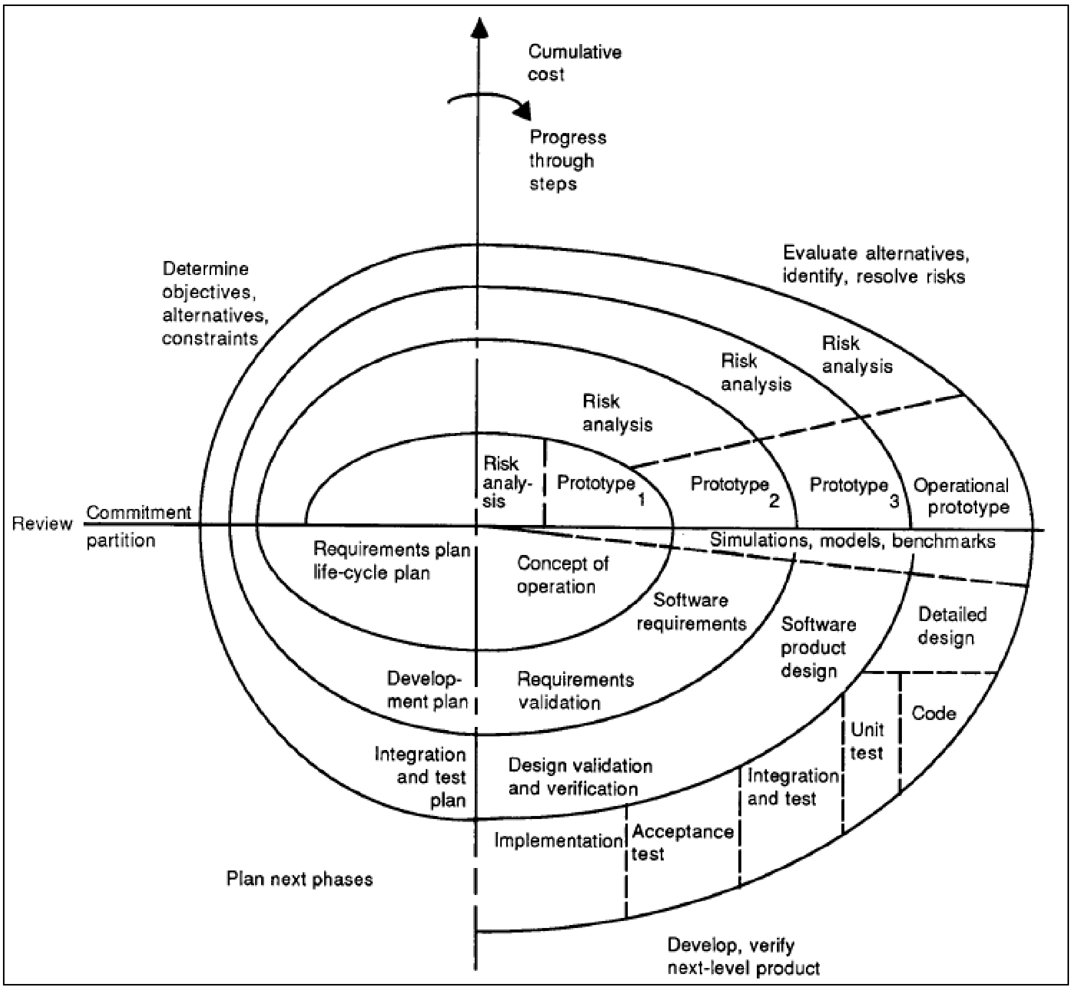

Figure 2.3: The Spiral Model.[1]

Figure 2.3 illustrates the four quadrants of the spiral model. During every cycle of the model each of the four quadrants are re-visited, creating an outward spiral. The model does not specify how many cycles are to be made but naturally (as the vertical axis indicates) cumulative costs rise as cycles increase. “An important feature of the spiral model, as with most other models, is that each cycle is completed by a review involving the primary people or organisations concerned with the project”.[2] The model relies heavily on prototyping which was “formally introduced to the information systems community in the early 1980’s to combat the weaknesses of the Waterfall model”.[3] A prototype (in information systems context) can be defined as “the preliminary working version of an information system for

[1] Boehm, B. 1988. A Spiral Model of Software Development and Enhancement. *Computer*, p 64.

[2] Boehm, B. 1988. A Spiral Model of Software Development and Enhancement. *Computer*, p 65.

[3] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 123.

demonstration and evaluation purposes".[1] By revisiting development stages, the spiral model ensures that developers and stakeholders are in unison regarding system requirements. This lowers the risks for both parties and is especially successful if functional prototypes are developed in the third quadrant.

Boehm states that the model faces three primary challenges[2]:

- Matching to contract software.
- Relying on risk-assessment expertise.
- The need for further elaboration on the model's steps.

The spiral model does have significant advantages over the SDLC in delivering a final product to meet user requirements. Its iterative approach and incremental prototype development bridge the communications gap visible in SDLC applications and clearly lowers project risks. It provides developers with a more accurate understanding of the system's context increasing the likelihood of user acceptance. It is, however, a complicated and intricate process to follow and Fitzgerald accurately notes that "when it takes so much effort to comply with methodological requirements, it may be more worthwhile to spend the time on actual development".[3]

Although the model introduces a number of improvements, it remains a teleological approach. The fact that every cycle of the model is based on more refined requirements than its predecessor enables better handling of changing requirements during the development project; this, however, plays no part in post-implementation maintenance. When the model stops cycling, it will still deliver a product based on a final, fixed set of requirements. Once this happens, new requirements cannot be met and stakeholders are again forced to do extensive maintenance to keep the system usable. The model introduces the notion of "evolution" to information systems development. During the cycles, the final product evolves from prototypes, but when developers stop cycling through the quadrants, the system, though probably more acceptable, will rely on ongoing maintenance.

[1] Laudon, K.C., Laudon, J.P. 2004. *Management Information Systems*. p 395.

[2] Boehm, B. 1988. A Spiral Model of Software Development and Enhancement. *Computer*, p 69.

[3] Fitzgerald, B. 1996. *Information Systems Journal*. p 16.

2.4.2 The V-model

The V-model, widely accepted in Germany[1], focuses on "project management, software development, quality assurance and configuration with special attention for better communication between developer and customer".[2] Graphically presented (figure 2.4), the model has a V shape, but the V is also "a synonym for verification and validation".[3]

The model's emphasis on quality assurance is the result of the connections between development and testing activities. These connections ensure that the next phase of the project only commences after the results of the previous phase are tested and Boggs states that "costs are expected to be significantly lowered through its quality assurance modules".[4]

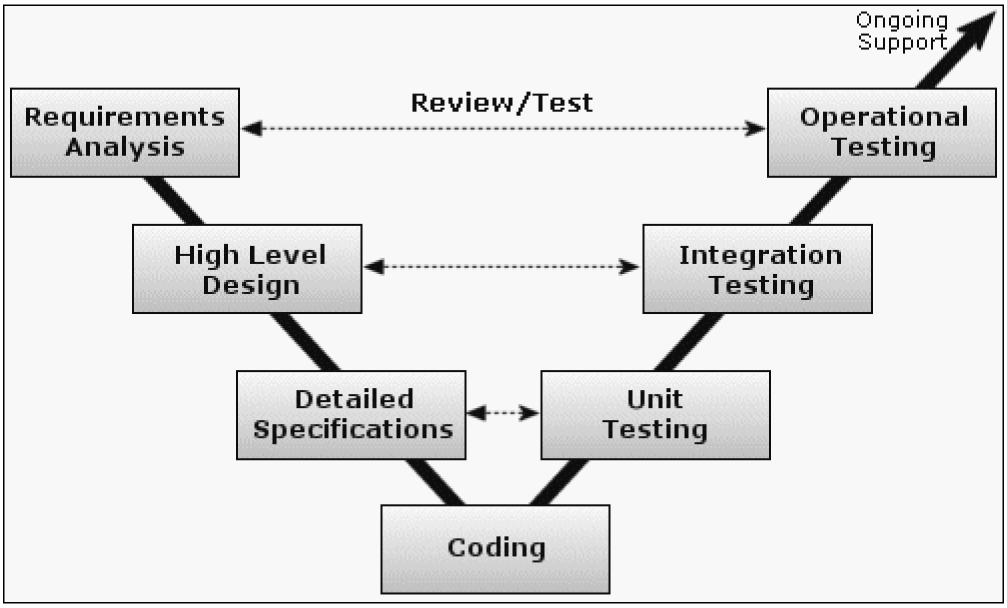

Figure 2.4: The V-model.

[1] Boggs states that it is the model used in all German military, civil and federal IT projects. Boggs, R.A. 2004. *Issues in Information Systems*. p 38.

[2] Boggs, R.A. 2004. *Issues in Information Systems*. p 38.

[3] Spillner, A. 2002. *The W-Model - Strengthening the Bond Between Development and Test*. Software Quality Research Laboratory Seminar 2002. p 2.

[4] Boggs, R.A. 2004. *Issues in Information Systems*. p 38.

Like the spiral model, the V-model has certain advantages over the SDLC. Boggs argues that it facilitates "better communication between developer and customer"[1] and that "costs are expected to be significantly lowered"[2] through its use. The extensive testing serves to capture current user requirements more accurately; forcing developers to critically evaluate every phase by performing appropriate tests. Boggs state that a newer model, the WEIT model, was said to replace the V-model in 2005 to address issues like scalability, change adoption and application formulation.

At the top-right corner of the model, an arrow indicates *Ongoing Support*. This is the V-model equivalent of the SDLC's maintenance phase. Although the model does not specify the processes involved in ongoing support, statistics clearly indicate that this is where organisations will be spending most of their money. Analysts and developers might argue that the model is designed to specifically reduce ongoing support by rigorously following accurate elicited requirements. It does not, however, affect the system's ability to cope with changing or emerging requirements post implementation.

Fitzgerald notes that although a great range of hard methodologies exist, the differences between them are often "relatively insignificant and artificially contrived".[3] He quotes Tagg who accurately summarises the practice of re-inventing similar methodologies:

> "Despite the fact that they are 95% agreed in their aims and their broad areas of getting there, they nevertheless manage to stay separate. Each set jealously guards its own style and magic ingredients".[4]

Goulielmos, accordingly, points out that the "goal of producing an all-encompassing methodology is not only unrealistic but also undesirable in practice".[5] This "search for the holy paradigm" have resulted in the, what may be, more than 1000 methodologies that have been published in pursuit of it.[6]

---

[1] Boggs, R.A. 2004. *Issues in Information Systems*. p 38.

[2] Boggs, R.A. 2004. *Issues in Information Systems*. p 38.

[3] Fitzgerald, B. 1996. *Information Systems Journal*. p 13.

[4] Fitzgerald, B. 1996. *Information Systems Journal*. p 10.

[5] Goulielmos, M. 2004. *Information Systems Journal*. p 382.

[6] Fitzgerald, B. 1996. *Information Systems Journal*. p 10.

2.4.3 The Mock Fixed Point Theorem

To understand why the concept of information systems flexibility is significant, it is crucial to shed light on the problems that face system stakeholders as direct or indirect results of a teleological paradigm. It has been shown (in section 2.4) how a growing awareness of the central role of requirements elicitation in system success has led to methodologies like the Spiral and V models that assert stronger emphasis on this process. The principle, however, of implementing a scientific reductionist approach for the development of complex, social systems remains prominent; post-implementation these systems will inevitably become less effective as the organisation continues to change.

This conclusion is supported by the Mock Fixed Point Theorem of Information Systems as discussed by Kanellis et al. The theorem states that:

> "There exists some point in time when everyone involved in the system knows what they want and agrees with everyone else".[1]

For organisations that undergo frequent change, the existence of a point in time where this theorem holds is extremely unlikely. The theorem operates on two conditions.

- Firstly, it requires that all the system stakeholders must be able to stipulate a correct set of system requirements. This implies that unknowable or emerging system requirements are completely ignored during development phases.
- Secondly, it requires agreement between all the system stakeholders regarding the combined set of requirements. This implies that conflicting views and opinions on system requirements must be completely resolved.

Any analyst or developer will realise the implausibility of the theorem and would agree that in the analysis phase of the development project, there are "limits to what can be known, both

[1] Kanellis, P., Lycett, M., Paul, R.J. 2000. *Proceedings of the International Conference on Information System Concepts*. p 202.

from a theoretical and practical standpoint".[1] Nevertheless, Kanellis et al. argue that requirements driven development methodologies are aimed at producing systems based primarily on a fixed set of correct user requirements at a certain point in time. Not only is the existence of such a point in time highly unlikely, information systems are developed to operate over periods of time. They conclude that requirements driven development results in static information systems that lack the ability to constantly undergo change in a changing context.[2]

The methodologies discussed in section 2.4 share the view that ongoing system maintenance is unavoidable, suggesting that systems development will not be a once-off project but an ongoing process that continuous for the system's entire lifespan. This assumption accentuates the fact that most of the money spent on systems development is spent post-implementation, during the maintenance phase. To address this problem by refining the pre-development phases to produce more accurate and complete system requirements might extend the need for maintenance, but it's unlikely to eradicate it.

2.5 Evolutionary Development

Kanellis et al. argue that closed information systems can only adapt to a changing environment through the "maintenance route".[3] The maintenance route generally involves the development of side-subsystems or patches that can substitute the current system operations.[4] The use of the word *maintenance* tends to suggest that developers did something wrong during systems development or that the system breaks down in some or other. While this is often the case, maintenance is frequently performed to adapt the system to new or changing requirements. As a result of organisational change, a misfit[5] between the information system

[1] Fitzgerald, B. 1996. *Information Systems Journal*. p 15.

[2] Kanellis, P., Lycett, M., Paul, R.J. 2000. *Proceedings of the International Conference on Information System Concepts*. p 202.

[3] Kanellis, P., Lycett, M., Paul, R.J. 2000. *Proceedings of the International Conference on Information System Concepts*. p 202.

[4] The maintenance route, in terms of the SDLC, refers to the various activities that make up the maintenance phase. Although the SDLC does specify this phase, the emphasis is stronger on the pre-implementation phases and less on maintenance.

[5] Information systems misfit is explored in more detail in the next chapter.

and the user requirements force developers to continuously perform maintenance. This does not necessarily mean the system was poorly developed. A more recent approach to systems development that accepts the inevitability of ongoing maintenance is evolutionary development.

Evolutionary development is a "staged or incremental approach that periodically delivers a system that is increasingly complete".[1] The basic principle behind these methodologies is a repetition of the SDLC stages in a series of short time frames. As developers move from iteration to iteration, the system evolves into one that is in line with user requirements. Evolutionary methodologies are popular in organisations that need to "react quickly and flexibly to environmental or market changes".[2] By harnessing a series of techniques to increase the speed of cycle iterations[3], extensive analysis and design activities are limited and solutions are delivered more timely. Although the process resembles the Spiral model in many ways, its ability to deliver working software more promptly makes it attractive to changing organisations. The development process typically terminates only when the system has been successfully integrated with the operations of the organisation.[4] Naturally, changes to these operations would require further system evolution.

On a theoretical level one might argue that evolutionary methodologies do not fall under the teleological paradigm as they approach the information system as an evolving and thus living system. This notion is strongly supported by the *Manifesto for Agile Software Development* which was published in 2001 by a collection of influential authors[5] and is available online.[6] The manifesto reads:

---

[1] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 119.

[2] Baskerville, R., Pries-Heje, J. 2004. *Information Systems Journal*. p 238.

[3] Techniques that are used in evolutionary development are aimed at reducing development time and are implemented in methodologies like eXtrme Programming, Unified Process, Scrum and Agile Development. These techniques are certainly effective in their purpose but rely heavily on the skills of professional developers and programmers.

[4] Maguire, S. 2000. *Information Management & Computer Security*. p 230.

[5] See bibliography.

[6] Beck, K., et al. 2001. *Manifesto for Agile Software Development.* http://agilemanifesto.org.

We are uncovering better ways of developing
software by doing it and helping others do it.
Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

That is, while there is value in the items on
the right, we value the items on the left more.

The authors state that a principle behind the manifesto is to “welcome changing requirements, even late in development” and that “agile processes harness change for the customer’s competitive advantage”. Other principles include the promotion of sustainable development that should be maintained at a “constant pace indefinitely”. These principles strongly oppose the rigorous methodological application of tools and techniques typical to the teleological paradigm, advocating light-weight, flexible development processes.

Whether evolutionary development marks a paradigm shift in information systems development thinking will depend on one’s understanding of the attributes that constitute a paradigm. Baskerville and Pries-Heje argue that evolutionary development does not mark a paradigm shift as certain key aspects of traditional development practices remain: although “evolutionary development integrates specification, design, and implementation”, the goal remains focused on a “completed system and project”.[1] They conclude that evolutionary development is leading the way away from the traditional methodologies and marks the beginning of a paradigm shift. This places the field of information systems development as we currently witness it in “a state of transition”.[2] The direction into which the field will ultimately expand is an arguable subject, however only certain techniques of the evolutionary methodologies “are likely to stabilise and endure”.[3] An important observation to make is that “the effect of growing rather than building systems may have slipped quietly past most

[1] Baskerville, R., Pries-Heje, J. 2004. *Information Systems Journal*. p 261.
[2] Baskerville, R., Pries-Heje, J. 2004. *Information Systems Journal*. p 239.
[3] Baskerville, R., Pries-Heje, J. 2004. *Information Systems Journal*. p 239.

pundits and into widespread practice without particular notice" and this major paradigm-shift in systems development "may have begun in the mid-1990s and has not yet been clearly delineated in history".[1]

Two key prerequisites for successful evolutionary development are the dedication of highly skilled, experienced developers and the organisation's commitment to the project.[2] Baskerville and Pries-Heje emphasise that the developers used in such projects "are not just replaceable programmers in a sea of developers".[3] The ability to produce technological solutions in short periods of time requires not only a rare set of skills but also hard-earned experience. Such individuals value the ability to be sharp and discover clever tricks like development process shortcuts.[4] Consequently, organisations must be committed to make large investments into the employment of people to perform the maintenance of their information systems. A study by Ravichandran and Lertwongsatien confirms this. They investigate the effects of information system resources and capabilities on firm performance and subsequently report the following: an organisation's ability to use IT to support its core competencies is dependent on the "nature of human, technology and relationship resources of the information systems department".[5] Other recent studies provide clear evidence that organisations are investing extensively into IT personnel.[6] What is specifically notable is that not only larger organisations (500 or more employees) employ IT staff but that "small organisations account for over three-fourths of IT jobs in non-IT organisations".[7]

Stamoulis et al. point out that continuous system maintenance or evolutionary development leads to a set of dependencies: "in order to innovate, the project organisation depends on the provision of continuous support in the form of material resources and help in coping with contingencies. In turn, this support will only be provided if the stakeholders see a range of potential benefits that can be obtained via the information system. The information system

[1] Baskerville, R., Pries-Heje, J. 2004. *Information Systems Journal*. p 262.

[2] Fruhling, A., De Vreede, G. 2006. *Journal of Management Information Systems*. p 42.

[3] Baskerville, R., Pries-Heje, J. 2004. *Information Systems Journal*. p 255.

[4] Baskerville, R., Pries-Heje, J. 2004. *Information Systems Journal*. p 255.

[5] Ravichandran, T., Lertwongsatien, C. 2005. *Journal of Management Information Systems*. p 257.

[6] Simon,J.C., Kaiser,K.M., Beath,C., Goles,T., Gallagher,K. 2007. *Information Systems Management*. p345.

[7] Simon,J.C., Kaiser,K.M., Beath,C., Goles,T., Gallagher,K. 2007. *Information Systems Management*. p346.

requires the effort and expertise of the project organisation for its continued existence.[1] Figure 2.5 illustrates these dependencies. System failure, in the context of figure 2.5, occurs when "the level of dissatisfaction with the system is such that there is no longer enough support to sustain it".[2] From a managerial point of view; this means either continued investment into system maintenance or project termination and system failure.

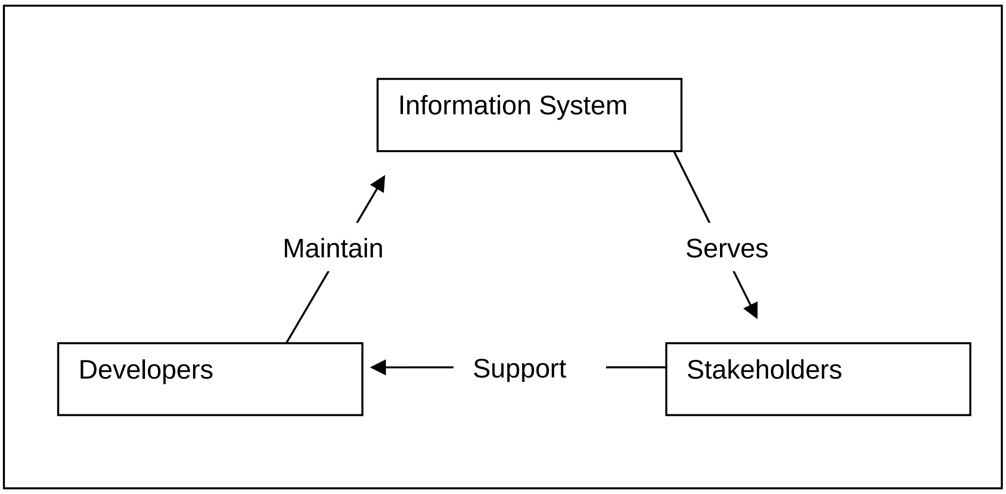

Figure 2.5: The dependencies of information system maintenance.[3]

An important observation to be made on the basis of evolutionary development is that an information system, if understood as a socio-technical system, can be self-organising. The organisation's capacity to enable the active participation of the three subsystems displayed in figure 2.5, is a critical requirement in this regard. End-users communicate contextual changes and emerging requirements to developers who have the required knowledge and skills to perform maintenance on the technological subsystem. Developers form the crucial bridge crossing the automation boundary, giving stakeholders indirect control over the functionality of the computer system. Their role is immiscible, as is the need for ongoing communication between themselves and stakeholders, emulating a never-ending repetition of analysis, design

[1] In this context the *project organisation* should be interpreted as the developers that are responsible for the innovation of the information system, i.e. to keep the information system alive. Kanellis, P., Lycett, M., Paul, R.J. 2000. *Proceedings of the International Conference on Information System Concepts*. p 193.

[2] Kanellis, P., Lycett, M., Paul, R.J. 2000. *Proceedings of the International Conference on Information System Concepts*. p 193.

[3] Adapted from Stamoulis et al Kanellis, P., Lycett, M., Paul, R.J. 2000. *Proceedings of the International Conference on Information System Concepts*. p 193.

and implementation. Though time consuming and expensive, this process has become accepted practice and is typically structured by agile development principles.

Based on a socio-technical understanding of information systems, hard methodologies clearly present an "ambiguous and narrow conception of the phenomena information system developers confront".[1] If ongoing system maintenance is an inevitable reality independent of the original requirements elicitation processes, can the investment of time into extensive analysis and design activities be justified? More importantly, if the concept of an accurate set of requirements is always a transitory one, should a set of requirements be the driving force for implementation processes?

Finally, it is worth noting the "buy versus build" dilemma here. Naturally, many organisations lack the capacity to develop and maintain information systems; a more suitable option is to buy standard commercial packages from software vendors. McManus states that "commercial packages are increasingly finding a place in corporations as information system professionals seek measures to cut costs while maintaining a high level of support to users".[2] Commercial packages are mostly sold with a certain degree of free support, which is likely to include package updates for a specified period of time. McManus lists three scenarios in which a commercial package is suitable:

1. When the organisation has a limited budget for an information system and depends largely on standard information processing functions.
2. If an organisation lacks the expertise or capital to do a detailed requirement analysis, the package becomes the basis for the core requirements.
3. The risks involved in information systems development might be too big and organisations would prefer to avoid them.

As can be expected, commercial systems also have numerous drawbacks. The most important of these is certainly the fact that they do not cater for the specific requirements of the organisation. Similarly, the updates might not necessarily match the changing requirements of the organisation and many organisations end up only utilising small parts of a commercial system, yet pay for a lot of functionality they never use. Some commercial packages allow

---

[1] Fitzgerald, B. 1996. *Information Systems Journal*. p 13.

[2] McManus, D.J. 2003. *The International Journal of Applied Management and Technology*. p 33.

customisation of the system for specific users, but this is usually associated with high tailoring costs.[1] Commercial packages are naturally popular in the more controlled business sectors that have multiple organisations with the same core operations and procedures.

2.6 The Need for Living Information Systems

To describe an information system as "living" necessitates some elaboration. So far, the concept has merely been used to differentiate between systems that can adapt to changing user requirements through system maintenance as opposed to computer systems that are restricted by a fixed set of user requirements for the duration of their organisational use. Stamoulis et al. draw on the theories developed by Varela and Maturana in biology to apply characteristics of living systems to information systems. They do this with emphasis on self-organising theory and operational closure, theories that I will introduce in the next chapter. In the remainder of this chapter, however, I will draw on Morgan's organisational metaphor of an organism to explain the need for a living information system as a subsystem of living organisations.

The central idea in the organism metaphor is "that of the organism or organisation as an open system".[2] Morgan states that organisms exist in "a wider environment on which they depend for the satisfaction of various needs".[3] This metaphor accentuates the integral role that information systems play in the survival of *living organisations*. Consider a living system (organisation) that lacks the ability to adapt readily to a changing environment as a result of a dead or latent subsystem (information system). Ultimately, the subsystem will negatively impact the larger system to such an extent that it will lose its ability to survive. Stamoulis et al. agree with this by stating that "what is required for living businesses are living systems".[4] They extend the metaphor of living systems to compare an information system to a human being and state that information systems should not be given an identity but should develop

---

[1] McManus, D.J. 2003. *The International Journal of Applied Management and Technology*. p 33.

[2] Flood, R.L., Jackson, M.C. 1991. *Creative Problem Solving: Total Systems Intervention*. p 10.

[3] Morgan, G. 1997. *Images of Organisation*. p 39.

[4] Kanellis, P., Lycett, M., Paul, R.J. 2000. *Proceedings of the International Conference on Information System Concepts*. p 202.

their own having been “brought into existence as infants with the ability to grow”.[1] Maguire argues along the same lines when he states that systems development methodologies should be business-led, allowing the information systems to be introduced rather than implemented.[2] In contrast, the reality is that most organisational information systems are unable to adapt to environmental changes as a result of inflexible computer systems that are incapable of growing with the organisation.

The contemporary practice of extensive information system maintenance activities realises the ideal of living information systems. The teleological paradigm produces an inevitable dependency on maintenance activities, separating the social and technological subsystems with a clearly delineated automation boundary. This boundary is firmly imposed by the inability of stakeholders, other than developers, to configure the technological systems on which they rely during business operations. Despite the financial implications of traditional system maintenance, the associated dependencies have further negative consequences.

The delays associated with maintenance operations result in an organisational subsystem that lacks the ability to react in real-time. IT departments normally schedule periodical system updates, allowing them to deal with a collection of emerging or changing requirements during a single system update. This implies that users have to wait for a period of time after reporting new requirements, before changes are completed and the system is updated. The resulting scenario is one in which developers form the bridge between the information system and its context, having the ability to facilitate co-evolution between the systems. The users, however, have little or no direct control over the computer systems and can merely advise developers on the organisational changes that requires information system updates. The situation is clearly inefficient as it is the users and not the developers that are dynamically aware of changes to the information system’s context. Naturally, their lack of technical

[1] Kanellis, P., Lycett, M., Paul, R.J. 2000. *Proceedings of the International Conference on Information System Concepts*. p 202.

[2] Maguire, S. 2000. *Information Management & Computer Security*. p 230.

ability to update intricate computer programs and the risk of it influencing the information system's overall integrity call for the expertise of developers.[1]

Secondly, many older computer systems (legacy systems) that still form the backbones of information systems in contemporary organisations, lack the ability to be uniquely tailored for specific users. This implies that when a change is made to the system, it affects all the users, not only those who required the change. On this basis, many emerging system requirements are ignored. Developers cannot make system changes that will only affect a minute part of the system's users and be useless or even disruptive to the rest. As a result, the information system lacks flexibility and is unable to support the changing organisation.

2.7 Summary and Conclusions

Traditional systems development methodologies are all characterised by a sequence of phases that are aimed at producing a system according to a fixed set of user requirements. Before system implementation commences, analysts and developers use a combination of techniques and tools to elicit the system's requirements from its stakeholders. Subsequently, the system is produced to adhere to these requirements as accurately as possible. This approach to systems development reveals an underlying teleological paradigm. The majority of methodologies that fall under this approach "are derived from ideas from the 1960s-1970s"[2], when systems development was a new field, and the "very nature of these methods reflects this point".[3]

The SDLC is the original systems development methodology and embodies the teleological paradigm with its requirements-driven structure. Other methodologies that became prominent towards the end of the century are all strongly influenced by the SDLC and operate on the same principle of system implementation driven by a correct set of user requirements. These

[1] Sharma and Yetton warn that one should be careful when drawing parallels between the complexity of information systems (as experienced by end-users) and successful system implementation. Although a strong stream of research suggests that technical complexity forms a barrier to successful information systems implementation, there is a lack of empirical research that supports this claim. Sharma, R., Yetton, P. 2007 *MIS Quarterly.* p 223.

[2] Kiely, G., Fitzgerald, B. 2005. *The Electronic Journal on Information Systems in Developing Countries.* p 2.

[3] Kiely, G., Fitzgerald, B. 2005. *The Electronic Journal on Information Systems in Developing Countries.* p 2.

methodologies have been implemented with limited success (see section 2.3). This is especially true in more recent years as the organisations in which information systems operate become increasingly subject to change and increasingly reliant on advancing technology.

Stamoulis et al. argue that systems operating in changing organisations cannot be developed for a fixed set of user requirements. Organisational change will result in changing or emerging system requirements that make the existence of a *correct set of requirements* a fallacy. The system will only be usable if the requirements for which it was developed remain relevant for the duration of the system's lifespan, a scenario which is highly unlikely in any turbulent business sector.

More recent development methodologies facilitate changing requirements by implementing tools and techniques that enable continuous system maintenance. These methodologies, referred to as evolutionary methodologies, operate on the principle that systems development is an ongoing process as opposed to a once-off project. They rely, however, on the commitment of system developers and stakeholders to maintain the system post-implementation. This creates a triangle of dependencies: computer systems depend on maintenance by developers, developers depend on the support of stakeholders, and stakeholders depend on the functioning of the computer system. These dependencies have become an accepted part of modern information systems as managers realise that their changing environments necessitate ongoing information systems maintenance.

The teleological paradigm has been and is still dominant in systems development practice. Although evolutionary development has shifted the focus away from requirements-driven development principles, its implementation is not financially or practically feasible in all organisations. It is the argument of Stamoulis et al., however, that dynamically changing information systems can be realised without a fundamental reliance on technically skilled professionals. In the next chapter I discuss the ateleological development paradigm, standing in stark contrast to the traditional development methodologies of the teleological paradigm and suggest a new approach to information systems development and maintenance.

Chapter 3

The Ateleological Paradigm

3.1 Introduction

Organisations in the 21st century will become increasingly reliant on their ability "to respond adroitly to constant shifts in their environments".[1] A new class of flexible information systems is required to support living organisations in their need to "continuously improve, innovate and adjust a whole gamut of organisational properties in response to environmental contingencies".[2] In chapter 2 it was argued that contemporary systems development practices, however, have revealed a limited ability to produce systems with the inherent ability to survive in a dynamically changing context. These practices are typically shaped by the requirements-driven structure of traditional systems development methodologies, characterised by a scientific, reductionist approach to information systems development. The turbulence of modern business sectors and the immense speed of technological development have accentuated the fact that "it is increasingly difficult to ensure success using the traditional paradigm of information systems development".[3]

In contrast to the traditional (teleological) paradigm, the ateleological[4] paradigm approaches information systems development from the standpoint that changing or emerging requirements are inevitable. It proclaims that the system's ability to productively and dynamically accommodate these requirements should be its critical success factor; not the system's adherence to a pre-defined set of requirements. Remenyi and Sherwood-Smith refer

---

[1] Jackson, M.C. 2003. *Systems Thinking: Creative Holism for Managers*. p xiii.

[2] Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 294.

[3] Goulielmos, M. 2004. *Information Systems Journal*. p 363.

[4] The term 'ateleological' as opposed to 'teleological' (see section 2.1), in this context, should not be interpreted as *goalless* or *purposeless*. It implies, rather, that the emphasis of the systems development project is not as much on the product, as it is on the process. Development is not driven by the idea of a perfect product, but by the ongoing process required to keep the information system functional and effective.

to this approach as "the post-modernist concept as it is applied to information systems development".[1] The ateleological paradigm implies a move away from the structured step-by-step development methodologies that dominate information systems development practice and proposes a decentralised, flexible information system that is developed and maintained by its users.

The concept of Tailorable Information Systems (TIS) is an idea in search of systems flexibility.[2] It introduces an approach to information systems development with an ateleological paradigm that frees the information system from the design decisions made during its analysis and design phases. It also promotes an information system that is free from the dependencies of traditional information system maintenance processes. Stamoulis et al. claim that such information systems will be able to "adapt to changing requirements during their operation".[3] The notion of TIS, as developed so far and investigated in this study, should not be seen as a proof of concept, but rather "stimulate thinking, seeking to enhance our understanding about information systems development and to provide a direction for the state of the art to pursue".[4] Literature on this phenomenon has been published since the late nineties and the concept's theoretical roots are still being developed. It is a revolutionary concept, proposing a significant change in thinking about systems development, but it is also a true representation of the actual form that many current information systems have taken on over time.[5]

In this chapter I will discuss the important relationship between an information system and the context[6] within which it operates before continuing to define TIS and discuss its

---

[1] Remenyi, D., Sherwood-Smith, M. 1997. *Achieving Maximum Value from Information Systems.* p 5.

[2] Stamoulis, D.S., Martakos, D.I., Introna, L.D. 1998. *Proceedings of the 1999 European Conference of Information Systems.* p 1011.

[3] Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies.* p 295.

[4] Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies.* p 295.

[5] In section 2.5.1, Baskerville and Pries-Heje are referenced stating the practice of information systems development finds itself in a transitional state. This transition is already observable in many organisations but lacks any form of formal theoretical conceptualisation. Managers and developers working in such organisations are thus likely to find that the TIS concept merely represents a holistic view of their information systems as opposed to a new, revolutionary approach.

[6] The term *context* refers to the direct "socio-organisational circumstances" within which the information system operates. Avgerou, C. 2001. *Information Systems Journal.* p 45.

theoretical roots. It is the opinion of Kanellis and Paul that information system success is largely dependent on the relationship between an information system and context. They argue that changes in the organisation's environment lead to a misfit between the information system and context, ultimately resulting in a perception of information system failure.

A central notion in TIS is the shift of power from computer system developers to end-users in terms of their ability to change the system to match their requirements. Empowered users that are not bounded by restrictions placed upon them by computer systems, can manage the fit between an information system and its context in real-time, overcoming the dependencies of ongoing system maintenance - a scenario supported by the post-modern viewpoint that users are the origin of the best ideas for improving their work.[1] This shift of power is naturally associated with numerous challenges which result mainly from the decentralisation of information system control. They are discussed in the final part of the chapter.

3.2 The Source of Change

A case study performed by Kanellis and Paul and reported in 2005[2], led to the development of a concentric model to illustrate how changes in an organisation's environment influence the context within which an information system operates. The case study was performed over a period of 11 months at a devolved organisation operating within the UK electricity sector with a turnover of over $6 billion.[3] In the following section I will briefly report the findings of the case study and draw some basic conclusions. In sections 3.4 and 3.5 these findings are generalised and investigated further.

---

[1] Flynn, D.J., Jazi, M.D. 1998. *Information systems Journal*. p 54.

[2] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 64.

[3] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 68.

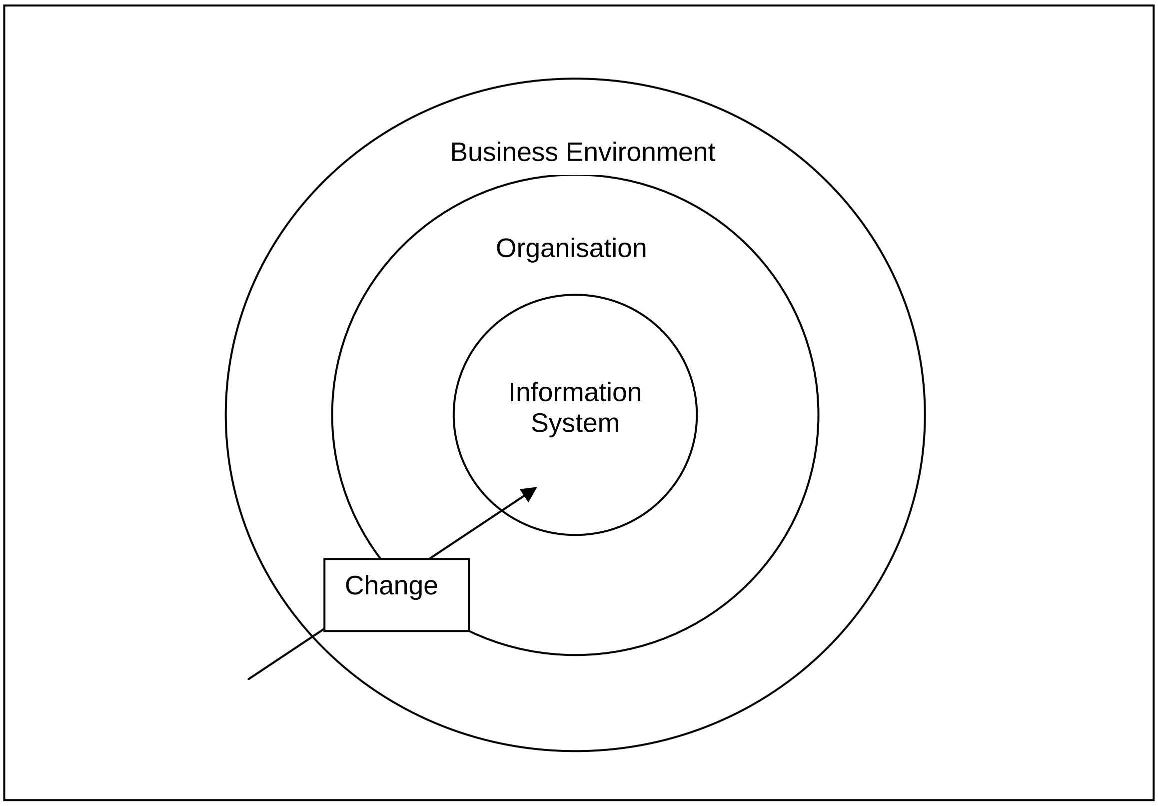

Figure 3.1: A concentric model of an information system's environment.

Figure 3.1 illustrates an information system as a subsystem of an organisation and an organisation as a subsystem of the larger business environment. The business environment is made up of a collection of systems over which the organisation has little or no control. Such systems might include competitors, regulators, unions and customers, all crucial to the organisation's survival. Changes in the business environment force the organisation to adapt and, in doing so, change the context in which its information system operates. Thus, change has a reversed ripple effect from the outside of the model towards the centre. Every time a system adapts itself to change in its environment, the subsystems within it has to deal with a changed environment. Based on the results of the case study, figure 3.1 indicates that organisational change has its birth outside the organisational boundary; changes within the organisation are mostly actions taken to keep the organisation profitable within its changing environment. These changes directly or indirectly impact the requirements upon which an information system is developed. Xia and Lee confirm this finding by stating that "information systems development is inherently complex because it must deal with not only technological issues but also organisational factors that, by large, are outside of the project team's control".[1]

There can be no argument that business environments in the 21st century will become increasingly turbulent and that continuous, rapid change is inevitable for business success. This has been a central theme in organisational theory ever since globalisation has dissolved geographical boundaries in business sectors. Organisations must be prepared to make rapid internal changes, knowing that this will influence the context of their information systems. Fitzgerald, accordingly, points out that "the continuous change that organisations are now faced with means that the economics of formalised systems development is dwindling".[2] When an organisation changes, it leads to a mismatch between the information system and the pre-defined requirements of its users, something referred to as information systems misfit.[3] This emphasises flexibility as a key success factor for modern information systems. Kanellis and Paul confirm this by arguing that information systems should, like post-industrial organisations, "be characterised by frequent and continuous change in structures, domains, goals and so forth, even in the face of apparently optimal adaptation".[4]

3.2.1 Change in the Business Environment

Avgerou points out that in several areas of information systems research the focus of the study extends beyond the single organisation and includes aspects of the organisation's environment. Such studies "tend to adopt a contingency analysis, suggesting that organisations choose courses of action according to their assessment of the type of environment they are faced with".[5] In the case study, three elements in the business environment that forced the organisation to implement changes to ensure survival are identified. None of these elements are confined to the specific business sector of the organisation in the case study and are likely to be present in most other business sectors.

---

[1] Xia, W., Lee, G. 2005. *Journal of Management Information Systems*. p 46.

[2] Fitzgerald, B. 1996. *Information Systems Journal*. p 19.

[3] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 79.

[4] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 64.

[5] Avgerou, C. 2001. *Information Systems Journal.* p 47.

- *Competition*

  Organisations are pressured to continuously adapt to maintain or expand their share of a certain market.[1] This might require organisations to continuously innovate, re-engineer or re-structure internally to keep themselves agile on all levels. Globalisation has played a crucial role in increasing the number of competing organisations in a sector, forcing managers to find new ways of setting their products and services apart from those of competitors. A competitive advantage is no longer enough to ensure long-term success, especially to those organisations supplying goods or services that can be sold through the World Wide Web.

- *Regulators*

  Business sectors are governed by regulations that protect the rights of both supplier and client. Regulations adapt periodically and organisations are forced to adapt accordingly to avoid prosecution. In the case study it was reported that regulatory issues demanded "significant management attention and represented major continuing uncertainty".[2] Naturally, some business sectors are regulated more rigorously than others, but it is unlikely that an organisation would ever be free of regulations imposed on it by a system in its environment.

- *Customers*

  The changing needs of customers force organisations to adapt their strategies. A constant awareness of customer needs and a readiness to react to them are crucial success factors in any competitive business sector. The quality of customer service naturally plays an important role in the development of long-term relationships with customers, ensuring future success. An increasing number of business sectors are becoming dominated by organisations that allow customers to customise products or services to their personal requirements.

The three external systems listed above are by no means a complete representation of the sources of change in a turbulent business environment; they merely serve as examples of such

[1] Avgerou mentions that competitiveness within business sectors has been the source of information systems literature as organisations require guidance on the actions they should take to harness the potential of IT in order to secure a competitive position for their firm. Avgerou, C. 2001. *Information Systems Journal*. p 47.

[2] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 72.

sources that were evident in this particular case study and fell outside the organisation's locus of control.

3.2.2 Change in the Organisational Environment

Changes in the business environment force organisations to change internally. Morgan's organism metaphor[1] for organisations is particularly applicable here. An organism depends on its environment for survival and a continuously changing environment requires continuous adaptation within organisational boundaries. The changes outside the organisation can result in various forms of internal change. In the case study, the organisation's environment forced it to bring about the following internal changes:

- *Strategic*

  As a result of the increasing competition in the business sector, the organisation was forced to adjust its strategy and target new markets locally and abroad. They were already losing a large share of their market to new competition and, as a result of regulations, could only be distinguished from them by price.[2]
- *Structural*

  The organisation made a decision to devolve from a centralised organisational structure to independent business units with minimal centralised control. It is reported that "this move was another attempt to increase the overall flexibility and competitiveness of the company by enabling decisions to be made closer to the operational level".[3] Structural change is a common and often regular practice in organisations to facilitate adaptation to environmental changes. Kwon et al. accordingly states that "in response to competitive pressure and environmental uncertainty, many organisations routinely undertake large-scale corporate downsizing, which subsequently results in the reconfiguration of information-processing protocols within and across firm boundaries".[4]
- *Cultural*

  Strategic and structural changes demanded cultural changes throughout the

---

[1] Morgan, G. 1997. *Images of Organisation*. p 39.

[2] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 70.

[3] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 73.

[4] Kwon, D., Oh, W., Jeon, S. 2007. *Journal of Management Information Systems*. p 202.

> organisation. The organisational culture was “extremely flexible at one end but had at the same time a great lack of trust and territorialism at the other end which meant that when a change occurred, there was an aggressive/defensive stand rather than a cooperative one”.[1]

The internal changes brought about as a result of changes in the business environment, had a profound impact on the context within which the organisation’s information system had to operate. Following the decision to devolve, the emergent autonomous business units had complete freedom regarding the development of bespoke computer applications that suited their own particular needs. This resulted in the disintegration of the original information system and a lack of overall control, but enabled the business units to survive and the overall organisation to become more flexible and adaptable to change.[2]

3.3 Information Systems Fit and Misfit

If the assumption is made that the context within which an information system operates will be subject to ongoing change, our perception of information system success requires rethinking. Current evaluation methods reveal, like traditional development methodologies, an emphasis on the system’s ability to adhere to a fixed set of requirements at a certain point in time.[3] Avgerou, however, emphasises that it is organisational context which produces information system requirements and the dynamic nature thereof makes it unrealistic to expect the determination of any definitive requirements.[4] Serafeimidis and Smithson agree that the available methodologies oversimplify the process and note that information systems evaluation is embedded in many social and organisational processes, and thus a particularly complex “decision-making process”.[5]

Evaluating information system success in a changing environment has led to research based on the argument that “any determination of information requirements must be based on the

---

[1] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 73.

[2] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 74.

[3] Serafeimidis, V., Smithson, S. 2003. *Information Systems Journal*. p 251.

[4] Avgerou, C. 2001. *Information Systems Journal*. p 45.

[5] Serafeimidis, V., Smithson, S. 2003. *Information Systems Journal*. p 251.

organisational use to which the information system is to be put".[1] Unlike the existing methodologies, this approach is free from a fixed point in time and measures (at least to a certain degree) a system's ability to adapt to its context. It is the opinion of Kanellis and Paul that "flexibility is an important fit relationship for developing contemporary information systems"[2] and that most systems that are currently being developed are "static entities whose purpose is to model a dynamic world".[3] This is a direct result of the technological bias observable in the hard methodologies, neglecting the dynamic nature of human activity. The emphasis is on the *product* being developed as opposed to the ongoing *process* of development. Technology, however, is not an objective but a social artefact as it is built by practices based on social assumptions.[4]

The notion of *information systems fit* is an approach to the measurement of information system success with emphasis on the system's ability to adapt to a changing context and cater for emerging and changing requirements. If a successful system is one which fits its context, an unsuccessful system has some degree of *misfit* with its context. In the case study, three major types of information systems misfit can be identified:

- *Structural*

  "Structural misfit refers to a change in the organisational structure that the information system has not been able to follow".[5] This type of misfit was prominent in the light of ongoing changes to organisational structure as a result of decentralisation. An employee states that "it was like trying to fit a square in a rounded hole, and the number of requests for changes to the systems increased, and have been coming non-stop ever since".[6] Further structural misfit resulted from "the very clear division of the organisation into distinct business units"[7] Kwon et al. notes the impact of organisational restructuring on the information–processing capacity of

[1] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 66.

[2] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 67.

[3] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 68.

[4] Flynn, D.J., Jazi, M.D. 1998. *Information systems Journal*. p 66.

[5] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 77.

[6] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 78.

[7] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 78.

an organisation: "a 10% reduction in workforce can lead to a reduction in information-processing-related coordination as high as 30%".[1]

- *Process*

  Process misfit occurs when a business process changes and the information system cannot adapt accordingly. In the case study, the observation is made that "processes have not only changed, but they have kept on changing"[2], necessitating that some parts of the system had to be discarded completely. These processes combine to form standard organisational routines and it is only when information system use becomes part of these routines, that "business value" is gained.[3]

- *Technological*

  Technological misfit is refers to a change in technology itself that makes the existing systems obsolete and cumbersome in the eyes of the users.[4] Managers were under constant pressure from system users who were aware of the latest technological advances and the positive effects these might have on the system. Users perceived the system to be outdated and out of touch with the technological evolution outside the organisation.

The concepts discussed above can be represented in the following analytical framework.

[1] Kwon, D., Oh, W., Jeon, S. 2007. *Journal of Management Information Systems*. p 223.

[2] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 79.

[3] Santhanam, R., Seligman, L., Kang, D. 2007. *Journal of Management Information Systems*. p 172.

[4] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 77.

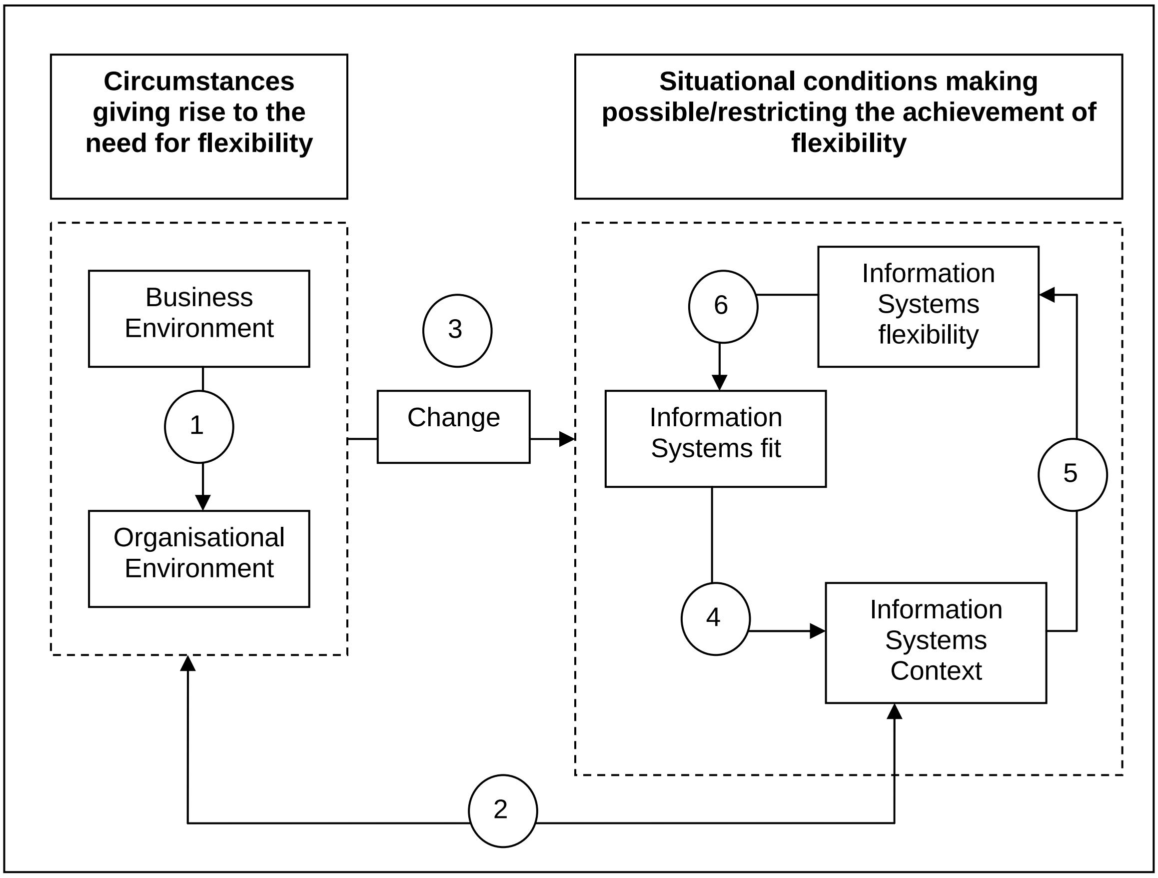

Figure 3.2: A synthesised analytical framework proposed by Kanellis and Paul.

The numbered arrows signify the following events:

1. Changes in the business environment lead to internal changes in the organisational environment.
2. Changes in the organisation impacts the information system context.
3. Changes in the business environment and the organisation itself have a direct impact on the information system fit.
4. A disrupted information system fit requires corrective action by developers.
5. Possible assumption changes towards change by developers and managers may impact systems development.
6. An information system with more flexibility will have better ability to adapt to unforeseen changes in its context.

3.4 The Emergence of Flexibility

The most notable discovery made during the case study was that the organisation managed to operate smoothly in spite of the presence of these forms of misfit. “User activities and tasks did not seem to be disrupted in any way”[1], something which came as surprising to the researchers. A closer investigation revealed that the users managed to do their work without reliance on the developed information system, suggesting that they were not “tied down by the systems”.[2] Similarly, Maguire reports that “in many instances staff work round systems to get the information they require”.[3]

This phenomenon can be explained as resulting from the process of decentralisation during which users were given almost complete autonomy and freedom with respect to meeting their own system and informational needs.

> “People used this freedom and have developed small applications of their own and along with application packages, have cannibalised the overarching systems to give themselves a system that is working by adapting it to their particular need - a truly flexible information system but certainly not a planned or intended one”.[4]

A general opinion of the end-users was that the systems developed by the organisation could not meet their dynamically changing requirements. Through the use of additional software applications they managed to extend the existing system to one which they could themselves tailor as required. Note that the developed system remained at one level, yet users developed their own system on a higher level to cater for emerging requirements. One particular form of tailoring was done by the extensive use of spreadsheets, something the managers referred to as the “Lotus Cult”.[5] The formal system lacked the ability to give users reports and statistics in the format they required, using a spreadsheet they could transform the information after

[1] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 80.

[2] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 80.

[3] Maguire, S. 2000. *Information Management & Computer Security*. p 230.

[4] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 80.

[5] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 81.

extracting it from the underlying system. Kanellis and Paul report that this form of underground activity "has come to be seen as essential even by the authorities themselves".[1]

Information system flexibility was achieved by allowing users to implement various forms of enabling technology and relying on their knowledge and skills to tailor the technology to their needs. Developers assisted users in the tailoring of applications and ensured that suitable technology could be supplied where needed. This scenario gave users a sense of ownership of the information system and subsequently the ability to get it to work for them rather than against them. Managers, on the other hand, realised the importance of technological competency of their employees and commented that they required a "new breed of sophisticated users"[2] that are able to innovate continuously. Avison and Fitzgerald support this view by stating that users will play a crucial role in information systems development "particularly given their increasing IT knowledge and the increasing sophistication and usability of the tools and packages available".[3]

Kanellis and Paul conclude that the resulting information system was not flawless and note that the real challenge is "to allow for maximum flexibility at the user and business unit level without the introduction of conflict that could jeopardise the integrity and stability of the corporate information system organisation".[4] If users are allowed to produce their own part of the system, managers are bound to lose overall control and the ability to integrate the various parts of the system. Imposing rules on them on the other hand, sacrifices flexibility, forcing managers to find a balance between flexibility and control.[5]

An observation made, emphasising the findings of chapter 2, was that the use of methodologies to develop the information system was reported to be "severely limiting any

---

[1] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 81.

[2] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 82.

[3] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 87.

[4] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 82.

[5] The balance between flexibility and control is a key aspect of ateleological development. Duane and Finnegan note that it's becoming increasingly important as more business processes are mediated through information technology. Their study, which focuses primarily on intranet environments, revealed that control structures not necessarily inhibit intranet growth, but actually empowers users by guiding their participation in its development. Duane, A., Finnegan, P. 2003. *Information Systems Journal*. p 156.

possibility of achieving flexibility"[1] and developers described them as "constraining and inadequate".[2] These findings correlate with the conclusion made by Avgerou that the "rational" principles of methodologies fail to support the "subjective, apparently irrational elements of actions within organisations".[3] The turbulent context within which systems had to operate nullified the advantages and affectivity of methodical development.

A final notable observation was the centrality of people, specifically end-users, in the process of information system maintenance. It emphasises the conclusion made in chapter 2 (2.5.1) that information systems, as social systems, can only flex and adapt when "the first and foremost of its components"[4], the people, do the same. The people that use the computer systems in everyday procedures are the ones who form the crucial bridge between the information system and its context, being constantly aware of the misfits between system and context.[5]

3.5 Operational Closure and Self-Organisation Theory

In the proceedings 6th European Conference of Information Systems[6], Stamoulis et al. argue that users are limited in their influence on the information system and mostly passively obey the system's rules. This leads to a loss of the system's internal coherence since the computer systems and their users (subsystems of the information system) lack the ability to co-evolute.[7] evolute.[7] An investigation into the applicability of self-organisation theory on information

[1] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 82.

[2] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 82.

[3] Avgerou, C. 2001. *Information Systems Journal.* p 48.

[4] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 82.

[5] Avgerou emphasises that there is a lack of information systems research that considers the broader context within which technological innovation is embedded and argues that a detailed contextual analysis is of crucial importance. Such an analysis, however, involves much more than purely consulting system stakeholders on their requirements. Avgerou, C. 2001. *Information Systems Journal.* p 43.

[6] Stamoulis, D.S., Martakos, D.I., Introna, L.D. 1998. *Proceedings of the 1999 European Conference of Information System*. p 1017.

[7] Stamoulis, D.S., Martakos, D.I., Introna, L.D. 1998. *Proceedings of the 1999 European Conference of Information System*. p 1017.

systems supports this statement and reveals a new understanding of the phenomena witnessed by Kanellis and Paul.[1]

The work of Varela and Maturana[2] that focuses on the notions of self-reference and operational closure is specifically applicable in this regard. The concept of *self-organisation* can be described as:

> "a genetic term which embraces a number of concepts – such as synergetics, autopoiesis, dissipative structures and self-referential systems – which all share the attempt to describe and understand the behaviour of complex, dynamic systems".[3]

"Systems can be self-organising if, and only if, they are operationally closed".[4] Understanding the concepts of self-organisation and operational closure necessitates a move away from the classical, input-type system description to a closure-type description of living systems and opposes von Bertalanffy's theories of open and closed systems. Instead of viewing an information system as an open system influenced by inputs from the environment, the study approaches an information system as an operationally closed system. The system maintains internal coherence through its internal, circular linkage of cause and effect. "An operationally closed system could be depicted as a collection of negative and positive feedback loops that create self-referring circular behavioural patterns".[5] Küppers explains the the concept of operational closure as follows:

---

[1] Magalhàes writes extensively on the application of these theories on organisations and notes that it brings new meaning to *organisational learning.* He remarks, however, that the concept of an operationally closed organisation is "far removed from the view of an organisation as an open system". Magalhàes, R. 2004. *Organisational Knowledge and Technology.* p 97.

[2] Maturana, H.R., Varela, F.J. 1972. *Autopoiesis and Cognition, The Realisation of the Living*.

[3] Küppers, G. 1999. *AI & Society Journal.* p 53. Markus and Robey investigate the unintended consequences of IT implementation with "complex adaptive systems theory". They argue that the distinctive characteristic of complex adaptive systems theory is that order at a higher level of analysis emerges from interactions at a lower level of analysis. They identify both self-organisation and co-evolution as key elements of complex adaptive systems theory. Andersen, K. V., Vendelo, M. T. 2004. *The Past and Future of Information Systems.* p 79.

[4] Stamoulis, D.S., Martakos, D.I., Introna, L.D. 1998. *Proceedings of the 1999 European Conference of Information System*. p 1013.

[5] Stamoulis, D.S., Martakos, D.I., Introna, L.D. 1998. *Proceedings of the 1999 European Conference of Information System*. p 1014.

> "Because of operational closure of cause and effect the states of the system are determined only by this circular dynamics. The system cannot be steered, it can only be disturbed by the environment. In contrast to classical systems theory, the environment no longer determines the system but, rather, the system determines the environment".[1]

The information system from the case study discussed in the previous section enabled internal circular dynamics by enabling interaction among its social and technological subsystems. The automation boundary that traditionally obstructed the dynamic co-evolution of these subsystems now becomes penetrable by stakeholders other than developers. This opposes traditional information systems development where users merely act as observers from outside the automation boundary. By empowering users to dynamically maintain technological systems, circular causality is created within the information system and a self-organising system emerges.

Stamoulis et al. state that "autopoetic systems have only one goal and that is to maintain themselves"[2] and explain this by arguing that inputs from the system's environment are dealt with in terms of maintaining the integrity of the system's organisation. Küppers agrees by arguing that it is the internal dynamics of the system that allows its survival in a changing environment.[3] This understanding opposes the Darwinian belief that a system's adaptation is an achievement of the environment and proclaims that adaptation is an achievement of the operationally closed system itself.[4]

By introducing the concept of operational closure to the information systems domain, Stamoulis et al. extend their argument for TIS. Further investigation into the concept's

---

[1] Küppers, G. 1999. *AI & Society Journal.* p 54.

[2] Stamoulis, D.S., Martakos, D.I., Introna, L.D. 1998. *Proceedings of the 1999 European Conference of Information System*. p 1015.

[3] Küppers, G. 1999. *AI & Society Journal.* p 54.

[4] Küppers, G. 1999. *AI & Society Journal.* p 54.

applicability leads to the notion of "co-evolution between structure and agent"[1] and its extendibility to various levels of the information system.

- On an organisational level, co-evolution between information system and context (socio-organisational environment) is required to enable information system fit. The structural coupling of system and context reveals a complex, dynamic relationship and reflects both the structure of the environment as well as the internal dynamics of the system.[2] This partly explains the phenomenon observed by Kanellis and Paul in their case study. The evolving organisation, despite various forms of information system misfit, remained reliant on the system's functionality. The result was the co-evolution of information system and context brought about by innovating end-users with enough knowledge, autonomy and technology to facilitate the maintenance of the system's integrity. Avgerou, accordingly, states that "the development and use of computer technologies have been seen as intertwined with the social fabric of organisations, and it is seen as emergent, incremental, more accurately characterised as improvisation rather than precalculated".[3]
- Within the information system itself, structural coupling between the social and technological subsystems enables a second instance of co-evolution. End-users, as revealed in the case study, are becoming increasingly proficient in their ability to cross the automation boundary and extend existing information system functionality to a more complete set of functions, catering specifically for their own emerging and changing requirements.

Stamoulis et al. emphasise that a crucial enabling factor for TIS is the locality of knowledge within the information system. Sharma and Yetton agree by stating that "end-users need to acquire new knowledge to be able to use new information system applications effectively".[4] They identify three "knowledge domains" that end-user training initiatives should deliver:[5]

---

[1] Stamoulis, D.S., Martakos, D.I., Introna, L.D. 1998. *Proceedings of the 1999 European Conference of Information System*. p 1015.

[2] Küppers, G. 1999. *AI & Society Journal*. p 54.

[3] Avgerou, C. 2001. *Information Systems Journal.* p 48.

[4] Sharma, R., Yetton, P. 2007 *MIS Quarterly.* p 219.

[5] Sharma, R., Yetton, P. 2007 *MIS Quarterly.* p 220.

- *Application knowledge* – knowledge covering the commands and tools embedded in the information system.
- *Business context knowledge* – knowledge covering the use of the information system to effectively perform business tasks.
- *Collaborative task knowledge* – Knowledge covering how others use the information system in their tasks.

It is only when users have the ability to innovatively apply technology to fit their changing requirements that the information system can maintain its internal coherence and structural coupling with a changing context. This point is supported by the view that "TIS is a reflection of one's business responsibility onto the system's infrastructure that supports or enables one's role".[1] Traditionally, this responsibility belongs only to members of the development project team, but tailorability improves every user to a developer whereas the user remains an expert in his own job and does not become a software engineer.[2]

The idea of users as software developers is not unique to TIS. Avison and Fitzgerald state that this scenario, also known as end-user computing or EUC, originally resulted from the inability of developers to deliver applications due to backlogs.[3] Users use self-taught skills to produce basic applications such as programmed spreadsheets or VisualBasic (VB) scripts to overcome their obstacles for the time being. They are often very successful in their efforts as they have the "advantage of knowing the requirements of their part of the business very well"[4] and are free from the timely procedures of system maintenance by developers and analysts. As will be explained in the next section, EUC share a lot of the core principles of TIS. This is reflected in Mirani and King's definition of EUC as "the practice of end-users developing, maintaining and using their own information systems".[5]

---

[1] Stamoulis, D.S., Martakos, D.I., Introna, L.D. 1998. *Proceedings of the 1999 European Conference of Information System*. p 1017.

[2] Stamoulis, D.S., Martakos, D.I., Introna, L.D. 1998. *Proceedings of the 1999 European Conference of Information System*. p 1018.

[3] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 84.

[4] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 84.

[5] As quoted by Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 84.

Avison and Fitzgerald explain that EUC, despite having numerous advantages, proves to be "unsustainable in many organisations".[1] The main problem is that the systems developed by users lack integrity and the ability to be integrated with systems in other departments of the organisation. Understandably, TIS face similar challenges. These will be discussed in more detail at the end of this chapter.

3.6 The Theoretical Roots of TIS

3.6.1 A Working Definition for TIS

Before introducing the main ideas that currently form the theoretical roots of TIS, I will develop a working definition for the term. It is essential that such a definition should encompass these theoretical roots and depict how TIS are distinctly different from traditional information systems. Using the following statements made by a combination of experts as building blocks, I will construct a definition:

- "An information system can be deemed as tailorable if it provides a user with control over its operation, meaning that a user should be able to regulate/operate the system, and thus direct or manipulate its behaviour".[2]
- "Maximum tailorability means utter independence from design decisions".[3]
- "TIS, in contrast to traditional development practices, require the design and implementation of new behaviour to be inextricably interwoven and carried out in real time".[4]
- "…system ownership in an organisation should shift from the information systems department/developer to the user – the main idea behind TIS".[5]
- "The purpose of these systems, termed end-user tailorable information systems, is to provide control over systems to a user".[6]

---

[1] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 85.

[2] Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 298.

[3] Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 301.

[4] Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 303.

[5] Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 295.

[6] Stamoulis, D.S., Patel, N.V., Martakos, D.I. 1996. *Proceedings of the 1996 European Conference on Information Systems*. p 316.

- "Tailorability in information systems is well-expressed as the degree of control users have over the functionality and operation of computer systems."[1]
- "The TIS view treats the development and usage of information systems as a like for like principle, and seeks to provide a dynamic information systems development and usage environment for changing information needs in a changing organisation".[2]
- "…the concept posits that systems design decisions should be deferred to users to make when using systems in particular, individual or group, changing organisational situations".[3]

From these statements it is clear that the emphasis in the definition should fall on the control of end-users over computer/technological systems within the information system. It is important to distinguish between different subsystems of people – if we generalise both users and developers to "people" in the definition; we fail to articulate the key strength of TIS, namely end-user tailorability. My working definition for TIS is as follows:

A Tailorable Information System is an information system in which end-users of technological systems have the ability to adapt these systems to fit their changing or emerging requirements.

The *ability* of end-users to adapt technological systems involves interventions at various organisational levels. In section 3.6.3 these interventions are explored in more detail and the various role-players involved are identified.

3.6.2 Deferred Systems Design

As stated in section 3.1, TIS are only examples of what information systems developed with an ateleological paradigm might culminate as. Such systems will, to a large extent, be characterised by their lack of pre-defined structure, making it difficult to propose a generalised architecture. A more valuable approach might be to focus on the paradigm shift

[1] Patel, N.V., Irani, Z. 1999. *Logistics Information Management*. p 36.

[2] Patel, N.V. 1999. *Proceedings of the 1999 Americas Conference on Information Systems*. p 4.

[3] Patel, N.V. 1999. *Journal for Requirements Engineering*. p 82.

underlying TIS, generalising the key notions of the ateleological paradigm. Stamoulis et al. summarises the crux of this paradigm shift:

> "TIS imply a paradigm shift in systems development, where the users or system designers will no longer be required to foresee every contingency or articulate every requirement for a design".[1]

If an information systems development project is approached from the standpoint that the *telos* (end) of the project is an elusive ideal for as long as the information system operates, the traditional emphasis on requirements elicitation is clearly unproductive. This is especially true in the light of the time-consuming, expensive nature of elicitation techniques. Several experts, accordingly, question the applicability of any methodical approach for the development of information systems.[2] Fitzgerald states:

> "Systems development in practice is actually an unstructured, evolutionary process. Yet, development methodologies attempt to impose complete solutions when the minimum are not yet well-defined".[3]

Goulielmos agrees by stating that "nearly half of all information systems development projects use no methodology".[4] Baskerville and Pries-Heje use the term "amethodical" to describe this scenario and define it formally as "systems development without predefined sequence, control, rationality, or claims to universality".[5] Metaphors associated with amethodical development "spring more from gardening"[6] than engineering as the systems are being *grown* as opposed to being *built*. It facilitates the ongoing evolution of an information system by merging the activities of development and maintenance into a single continuous activity.[7] Fitzgerald recognises that amethodical development is a more realistic approach when he states that "developers do not develop systems by completing a single task and

---

[1] Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 296.

[2] Fitzgerald, B. 1996. *Information Systems Journal*. p 15.

[3] Fitzgerald, B. 1996. *Information Systems Journal*. p 17.

[4] Goulielmos, M. 2004. *Information Systems Journal*. p 381.

[5] Baskerville, R., Pries-Heje, J. 2004. *Information Systems Journal*. p 261.

[6] Baskerville, R., Pries-Heje, J. 2004. *Information Systems Journal*. p 261.

[7] Baskerville, R., Pries-Heje, J. 2004. *Information Systems Journal*. p 262.

moving on to the next task following a rational sequence; rather they will have a range of tasks not concluded due to the interdependent nature of development".[1] This approach is undoubtedly in line with many principles of the ateleological paradigm as it recognises the futility of a clear-cut distinction between development and maintenance for systems that will operate in "emergent organisations".[2] It does not, however, justify development in a sloppy or careless manner.[3]

This perspective accentuates that the realisation of living information systems requires the emphasis to be on the ongoing process of development, as opposed to a once-off development project aimed at delivering a complete product. The driving force of information systems development becomes not the pre-defined requirements, but the ability to dynamically adapt the system to changing and emerging requirements. A concept that is central to this ideal is that of *deferred systems design* (DSD). Patel argues that it should be interpreted as a principle that seeks to actualise the view that information systems design should consider the non-planned nature of human behaviour.[4]

DSD implies that design decisions traditionally made during the pre-implementation phases of a development project, should be deferred to post-implementation adaptation procedures. It operates primarily on the theory that no correct set of fixed user requirements exist and, as a result, forces system analysts away from the time-consuming procedures of requirements elicitation, encouraging them to develop systems that are less "contextually sensitive".[5] The focus moves from a finished *product* to on an ongoing development *process*.

Although TIS and amethodical development share numerous development principles, the key difference is the role-players. TIS, off course, specify that post-implementation adaptation should primarily be performed by end-users as opposed to developers or IT professionals. This enables the development of an information system that allows "users to make the actual

[1] Fitzgerald, B. 1996. *Information Systems Journal*. p 14.

[2] Baskerville, R., Pries-Heje, J. 2004. *Information Systems Journal*. p 262.

[3] Fitzgerald, B. 1996. *Information Systems Journal*. p 201.

[4] Patel, N.V. 1999. *Proceedings of the 1999 Americas Conference on Information Systems*. p 6.

[5] The use of the term *contextually sensitive* ties in specifically with the study of Avgerou who emphasises the importance of information systems context in both development and use (Avgerou, C. 2001. *Information Systems Journal*. p 43). Patel, N.V. 1999. *Journal for Requirements Engineering*. p 83.

systems design decisions, depending on the organisational situation in which the information system will be used".[1] By moving design decision to users, dynamic system adaptation is facilitated to support the "changing and ongoing organisational work designed to achieve objectives".[2] This supports the notion that organisational work should not be structured around rigid computer systems, but should continuously guide information systems development.

Any systems analyst will recognise that DSD opposes current development methodologies in many ways. For one, it not only redefines the role of analysts in the development process but questions its future existence. Stamoulis et al. go as far as to say that if design decisions can be deferred, "the idea of having systems analysts and designers will fast become obsolete".[3] The ateleological paradigm certainly suggests that once-off pre-development systems analysis is futile in the face of changing organisations and that no amount of planning can cater for emerging requirements, but too little research has been done to validate this prediction. On the contrary, the ateleological paradigm might lead to continuous systems analysis to control continuous systems development by users, reinforcing the role of analysts. In the final chapter, this notion is further explored.

Table 3.1 provides a summary of the key differences between the teleological and ateleological development paradigms.

[1] Patel, N.V. 1999. *Proceedings of the 1999 Americas Conference on Information Systems*. p 6.

[2] Patel, N.V. 1999. *Journal for Requirements Engineering*. p 83.

[3] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 84.

Information Systems Development Attributes	Teleological Paradigm	Ateleological Paradigm
Aim and Deliverable	'Static' solution for a specific business problem	System with the ability to continually flex and adapt
Purpose	To meet a centrally defined goal/target, which addresses a problem as agreed at a specific point in time	To serve localised needs (disjoint or common) by continuous adaptation of the system's behaviour
Drivers	Frozen, Contractual Requirements	The current needs which evolve arbitrarily
Development Management	Top-down problem breakdown; Centralised management, realised by the analysts/developers	Problem addressed holistically; Decentralised management realised by the users themselves; allows for multiple routes to be followed: evolutionary theory's principle of selective retention applies.
Provision for Changes	Only as much can be determined in advance at design time	System is designed and being operated in such a way that allows for alteration of its behaviour at run-time in order to adapt

Table 3.1: Contrasting information systems development paradigms.[1]

[1] Adapted from Stamoulis, D.S., Kanellis, P., Martakos, D.I. 2001. *Journal of Applied System Studies*. p 297.

3.6.3 Empowering the end-user

A shift of power from developers to users frees an information system from the dependencies of ongoing system maintenance by developers and is fundamental to the idea of TIS. These dependencies result in ongoing investments to keep information systems usable in changing contexts and are associated with numerous disadvantages (see section 2.5). Empowering the end-user is a multi-dimensional notion that has implications for the responsibilities of developers, managers and, most importantly, the users themselves.

3.6.3.1 The Responsibility of Developers

The "key ingredient" for the process of strategic alignment of information systems is an environment enabling "partnership-building relationships" between IT professionals and end-users of information systems.[1] Being in charge of the computer systems that are implemented within the information system, developers play an immiscible role in contemporary organisations. Although their role might change in a TIS environment, they still form an essential subsystem of the information system. Traditionally, developers operate in an information systems or IT department that is responsible for the development and maintenance of the computer systems, hardware and software, which form the backbone for information management. They are mostly technically qualified individuals operating under the instruction of an IT manager or CIO (Chief Information Officer).

In TIS developers are required to diffuse power that was traditionally tied up in their department to the system's end-users. This power, a result of their technical knowledge and skills to manipulate the computer systems, make the other subsystems of the information system dependant on them. Diffusing this power has numerous advantages but as can be expected, also holds various risks.

[1] Magalhàes, R. 2004. *Organisational Knowledge and Technology*. p 212.

- *Diffusion through technology*

It is the opinion of Kanellis and Paul that information systems departments should be "transformed into and managed as emergent organisations".[1] Users are getting increasingly sophisticated in their use of technology and developers should foster this by supplying them with technology and tools that empower them to develop business solutions dynamically. Such a scenario is often perceived as threatening to the overall control of the information system and rightly so but, if managed correctly, "an environment where maximum independence and flexibility is allowed at user and business level"[2] can be achieved. This would depend largely on "the necessary culture, policies, and controls in place so as to avoid the introduction of conflict that could jeopardise the integrity and stability of the organisation as entity".[3]

- *Diffusion through support and training*

A second way of diffusing the power of developers is by the introduction of training initiatives aimed at increasing the technical skills of end-users. Learning opportunities such as workshops, communities of practice and tutorials will support users in their attempt to harness the applications and tools at their disposal constructively. Although a TIS environment requires users to be pro-active in developing their technological skills, it also poses challenges to developers who in turn "must have a strong understanding of the business, be aware of the changing organisational and environmental realities, and furthermore, be prepared to accept this fact even though this realisation may result in a paradigm shift with respect to the ways they carry out their work".[4] In their study of the contingent effects of training on information system success, Sharma and Yetton note the lack of consistent empirical support for the current approaches.[5] Large investments into end-user training often fail to meet the implementation expectations of developers and managers and calls for a "richer

[1] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 85.

[2] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 85.

[3] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 85.

[4] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 82.

[5] Sharma, R., Yetton, P. 2007 *MIS Quarterly*. p 219.

theoretical conceptualisation of the effect of training on implementation success".[1] Nevertheless, Santhanam et al. emphasise that failure to facilitate knowledge transfers between technical professionals and end-users could "adversely affect the successful assimilation and use of new technology".[2]

3.6.3.2 The Responsibility of Managers

As a subsystem of information systems as defined by Earl (see 1.5.2.2), procedures typically shape an information management strategy. They are often rigorously forced upon employees by managers to ensure information integrity and diligence. Employees, on the other hand, draw some security from them as they are freed from the responsibility of shaping their own working procedures.

Although essential in any organisation, detailed business procedures often inhibit organisational flexibility. Their weakness is especially evident when employees have to deal with exceptions to standard procedures on a regular basis. Managers might attempt to solve these exceptions by designing alternative procedures for every exception; this, however, creates an autocratic culture that forces humans to act as machines and centralises all decision making to management, resulting in an a loss of organisational flexibility. Bradley et al. refer to this type of organisations as "formal organisations" and state that they have a "predisposition to demonstrate cost-effectiveness continually and to be consistently rigid".[3] "Entrepreneurial organisations", on the contrary, place stronger emphasis on "spontaneity, flexibility, and individuality" and have a tendency of being first to market.[4]

These two "main corporate culture types"[5] vary in their use of information systems and the subsequent impact it has on the organisation. It is the responsibility of managers to design business procedures in a way that fosters innovation yet helps to maintain organisational integrity, thus seeking a balance between flexibility and efficiency gained through automation. By relying on the technical skills of employees to innovatively apply technology

[1] Sharma, R., Yetton, P. 2007 *MIS Quarterly*. p 219.

[2] Santhanam, R., Seligman, L., Kang, D. 2007. *Journal of Management Information Systems*. p 172.

[3] Bradley, R.V., Pridmore, J.L., Byrd, T.A. 2006. *Journal of Management Information Systems*. p 269.

[4] Bradley, R.V., Pridmore, J.L., Byrd, T.A. 2006. *Journal of Management Information Systems*. p 269.

[5] Bradley, R.V., Pridmore, J.L., Byrd, T.A. 2006. *Journal of Management Information Systems*. p 268.

to cater for exceptions and putting in place regulations that ensure information integrity, they can promote this balance.

Managers should guard, however, against the disregard of the various social complexities that surrounds a user's perceptions of IT innovation. New technology is likely to pose a significant challenge to end-users and requires mindful management. To effectively facilitate user adaptation to the introduction of new technology, managers must ensure that users are provided with adequate resources that can "improve users' performance and minimise negative emotions"[1] associated with such introductions. A typical phenomenon in contemporary organisations to facilitate the adaptation of users to new technology is the establishment of an IT help desk. Research has revealed that IT help desks are "critical sources for transferring knowledge on the conceptual understandings and procedures relating to system use".[2]

3.6.3.3 The Responsibility of end-users

The ability to develop successful TIS resides mainly with end-users. They are the system stakeholders who utilise the technological functionality of the system on a day to day basis to manage the information that fuels organisational operations. Their attitude towards technology is a deciding factor in the quest for tailorability. Although it has become standard practice that any white collar employee of a contemporary organisation is computer literate to some degree, it is their willingness to continuously learn and develop new skills that is of imperative importance. A paradigm shift in information systems development will depend specifically on a "process-oriented perspective with a central role for the learning process".[3]

Kanellis and Paul state that users should move away from the notion of doing things in a certain way because they have always done it in that way.[4] This attitude restrains organisational learning and innovation, making an organisation defenceless in the face of a changing business environment. Organisational culture plays a central role in encouraging users to develop and apply newly developed skills to solve problems. Without the restrictions

---

[1] Beaudry, A., Pinsonneault, A. 2005. *MIS Quarterly*. p 518.

[2] Santhanam, R., Seligman, L., Kang, D. 2007. *Journal of Management Information Systems*. p 172.

[3] Fitzgerald, B. 1996. *Information Systems Journal*. p 18.

[4] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 82.

and security of detailed business processes, users will have the freedom to develop and implement self-made solutions. It is at this level that flexibility will prove valuable for the organisation since it allows business processes to continue despite exceptions to standard procedures.

Table 3.2 summarises the roles and responsibilities involved in the empowerment of the end-user.

Role players	Responsibilities
Developers	Supply technology and tools for users Train and support users
Managers	Give users freedom from rigorous business procedures Maintain organisational integrity
Users	Learn to use technology Apply skills to maintain computer systems

Table 3.2: A summary of the roles and responsibilities required in TIS.

The effect of end-user empowerment can best be expressed by adapting the model for information systems dependencies (figure 2.5). Whilst stakeholders financially support developers, developers support stakeholders through the diffusion of power; and stakeholders use and maintain the computer systems by implementing this new power to tailor the system to their emerging requirements. Note that stakeholders specifically include end-users. Having stated this, the relationship between developers and computer systems now requires redefinition. Although developers support users in their system use and maintenance, they would ideally not perform the bulk of maintenance operations themselves. That would recreate the dependencies that TIS aim to overcome and take power away from users. Their role should rather be to control and manage the various elements that make up the computer systems in such a way that users are equipped to maintain them. This can include the management of hardware, networking, servers, redundancy, software installation etc.

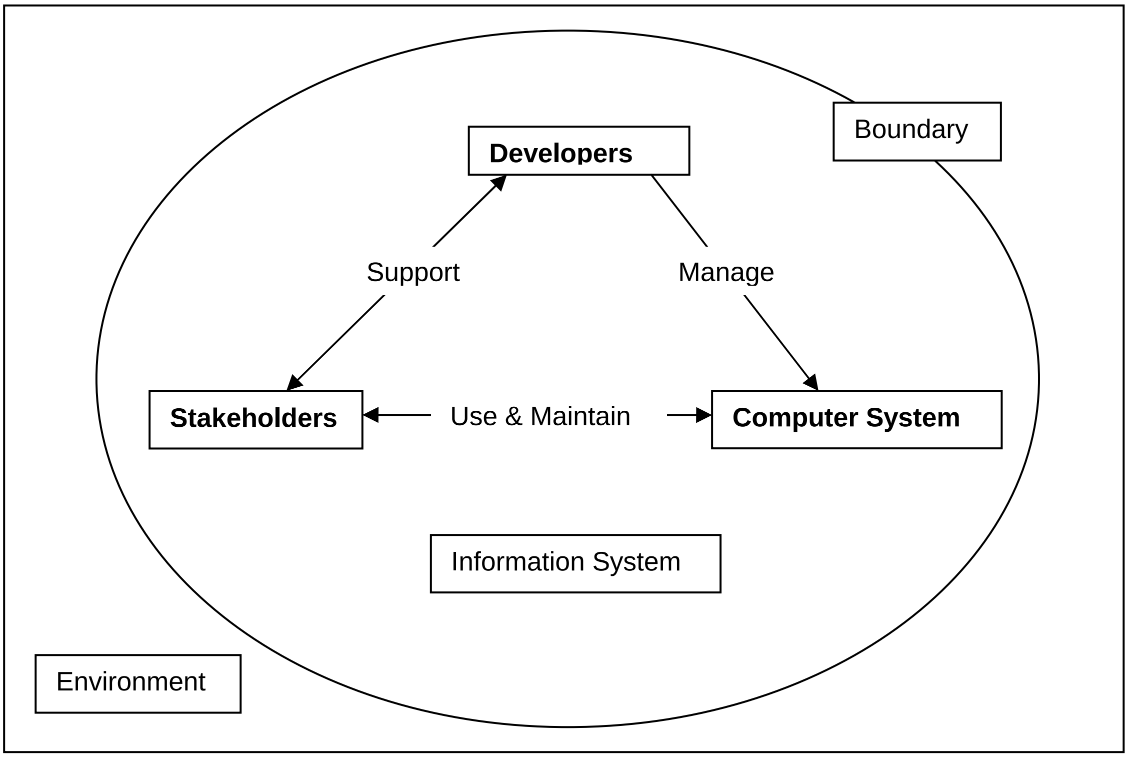

Figure 3.3: A model for TIS.

3.7 A Systems Architecture Model for TIS

TIS suggest an information system that is free from rigorous structures and predefined processes. By giving users the power to develop the system as requirements emerge, the system becomes a simultaneous combination of process and product. This does not imply, however, that TIS are void of structure on all levels. For tailorability to be achieved, a certain amount of technological infrastructure is critical to support the technology and tools that will empower users. Such infrastructure might include the hardware and network topology[1] of the system and is unlikely to undergo change as frequently as the software that operates on it. In accordance to this, Stamoulis et al. argue that different parts of an information system are subject to more frequent changes than other parts[2] and go on to develop a 3-layered systems architecture model for TIS. The model is displayed in figure 3.4.

---

[1] A computer network is typically classified into a category based on its topology or general shape. Examples of different topologies include the Star, Ring and Bus topologies. Comer, D. E. 2001. *Computer Networks and Internets*. p 103.

[2] Stamoulis, D.S., Patel, N.V., Martakos, D.I. 1996. *Proceedings of the 1996 European Conference on Information Systems*. p 318.

Areas that often undergo major changes		Information systems Structure		Control by (user empowered to tailor)
Employee tasks		User interface layer		End-user
Business functioning	Practices & routines	Application areas layer	Operational tasks	Field/Section Supervisor
	Business Rules & Regulations		Business Logic	Business Process Manager
Organisational Structure		operating systems -------------------------------- hardware systems		IT Experts (IT Dept).

Figure 3.4: A Systems Architecture Model for TIS.

The model represents an overview of the parts of the system that are subject to change and the various groups of people involved in the tailoring of these parts. It is arranged in 3 layers of which the middle layer represents two groups of people (Field/Section Supervisor and Business Process Manager). Although this study is not predominantly concerned with the implementation of TIS, the model will be analysed briefly.

3.7.1 Organisational Structure

The bottom layer of the model represents the part of the information system that is strongly associated with organisational structure. Stamoulis et al. state that this is typically the "network of machines and their operating systems"[1] in a corporate information system scenario. As discussed earlier in this chapter, structural changes in an organisation often lead to a structural misfit between the information system and its context.

[1] Stamoulis, D.S., Patel, N.V., Martakos, D.I. 1996. *Proceedings of the 1996 European Conference on Information Systems*. p 318.

Stamoulis et al. argue that the people responsible for the tailoring of this layer should be the IT experts. The installation of operating systems and hardware in a network environment requires a certain amount of expertise and it would be rather optimistic and notably risky to expect end-users to perform such operations. Technology in this layer should be designed and implemented in a way that allows IT experts to adapt them in real-time. Typically, IT experts should be able to "reconfigure the distribution of processing capacity"[1] to match emerging requirements of users without any disruption of the overall information system functionality.

To achieve tailorability at this level of the system necessitates a high level of modularity from both hardware and software. Stamoulis et al. argues that object technology is an enabling factor for software, allowing IT experts to reject, replace or modify any part of the system. Similarly, plug and play or hot swappable hardware allows physical system adaptation during operation.

3.7.2 Business Functioning

The next layer, situated in the middle, encapsulates what Stamoulis et al. refer to as business functioning.[2] This includes the business logic, routines and established practices that govern the way the organisation operates. These are all subject to change when emerging requirements create a process misfit between the information system and its context.

To achieve tailorability on this level the responsibility of adapting the system should move from developers or IT experts to business unit managers. This allows real-time adaptation that is specific to the unit or department where the requirements emerge. The technology that resides in this layer is usually built on the business logic and procedures of the department and materialises in the program configurations of applications that facilitate standard practices and routines. The key principle is that changes to business procedures or the role of an employee should never render the application software useless.

---

[1] Stamoulis, D.S., Patel, N.V., Martakos, D.I. 1996. *Proceedings of the 1996 European Conference on Information Systems*. p 318.

[2] Stamoulis, D.S., Patel, N.V., Martakos, D.I. 1996. *Proceedings of the 1996 European Conference on Information Systems*. p 318.

The model differentiates between field/section supervisor and business process manager yet they fall in the same layer of the model. Stamoulis et al. comment that their roles are "less managerial in nature"[1] and have to do with the daily business running rather than decision making. Persons at this level will be more sophisticated in their use of application software and have control over more system functionality.

3.7.3 Employee Tasks

The top layer, or user interface layer, represents the "employee's personal workspace".[2] The layer encapsulates all the tools and applications that the employee uses to perform his work on a day to day basis. Stamoulis et al. warns that the capacity of tailoring power supplied at this level is a complex issue.[3] As discussed above, the key to successful TIS is that persons working in this layer should not be dependent on the IT experts, but rather be skilled enough and have the right tools at their disposal to handle emerging requirements on their own. Stamoulis et al. admit that end users can operate in the applications layer (the one below them in the model) in an ideal scenario, giving them even more power over the technological system.

The proposed model accurately recognises that TIS cannot function without a certain degree of structure. The business processes and logic of most contemporary organisations rely on intricate software applications due to their complex nature. These applications, in turn, require the skills and knowledge of experts. This does not imply, however, that end-users can't control application software to a degree that allows them to deal with frequent emerging requirements. Stamoulis et al. discuss a combination of enabling technologies that, through DSD, allow the end-user to adapt applications at user level.[4] Some emerging requirements will involve the adaptation of systems located at the lowest level of the model, implicating

[1] Stamoulis, D.S., Patel, N.V., Martakos, D.I. 1996. *Proceedings of the 1996 European Conference on Information Systems*. p 318.

[2] Stamoulis, D.S., Patel, N.V., Martakos, D.I. 1996. *Proceedings of the 1996 European Conference on Information Systems*. p 319.

[3] Stamoulis, D.S., Patel, N.V., Martakos, D.I. 1996. *Proceedings of the 1996 European Conference on Information Systems*. p 319.

[4] Stamoulis, D.S., Patel, N.V., Martakos, D.I. 1996. *Proceedings of the 1996 European Conference on Information Systems*. p 321.

action by IT experts, but TIS require that these are done in real-time and in a way that is not disruptive to business procedures.

3.7.4 A Concentric Model

To illustrate the key differences between TIS and information systems that are subject to continuous maintenance by developers, I adapted Stamoulis et al.'s model to a concentric systems model for TIS. This model is specifically focused on illustrating the way an information system deals with various emerging requirements.

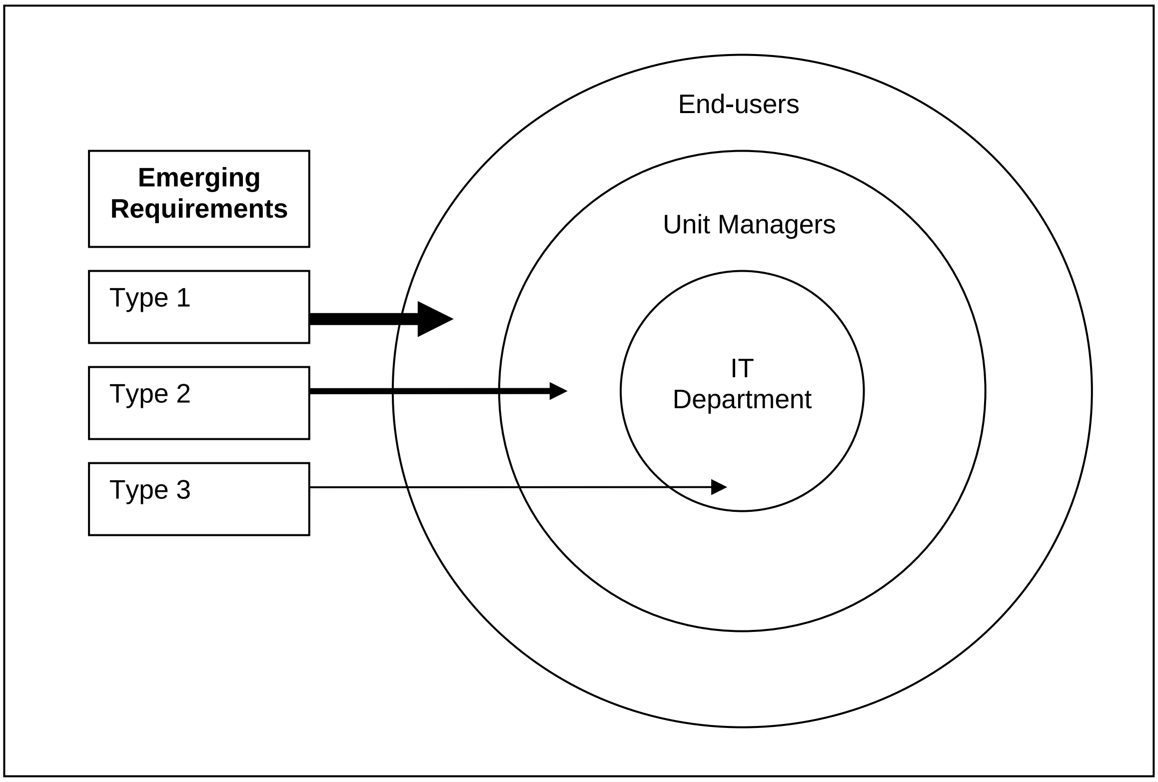

Figure 3.5: A concentric Model for TIS.

The model (displayed in figure 3.5) consists of 3 concentric circles representing the 3 layers in the Stamoulis et al. model. The innermost circle represents the IT department, then the unit managers and finally the end-users on the outside. The circles are penetrated by 3 arrows. They represent different types of emerging requirements resulting from changes to the system's context.

The first arrow penetrates only the outer circle and represents type 1 emerging requirements. The end-users of the system can adapt the information system at their level to this type of requirements. The frequency of this type is the highest as it is the end-users that are the closest to the changing context of the system; this is represented by the width of the arrow. The unit managers and IT department are protected from these requirements through the diffusion of their power towards the outside of the concentric model. The second arrow penetrates the middle circle and indicates requirements that call for a higher level of expertise and authority, but emerge less frequently. Unit managers are responsible for system adaptation when they occur. The final arrow represents emerging requirements that call for the expertise of the IT department; these might include a change in the network topology or the installation of a new operating system.

Information systems that do not allow tailoring by end-users or unit managers refer all the emerging requirements to the centre of the model, the IT department. This leads to a backlog of work for IT experts and, ultimately, an information system that is irresponsive to change. TIS, in contrast, enable the end-users to manage the fit between an information system and its context, an ideal situation since they are the people who deal with the emerging requirements in working procedures. Kanellis and Paul support this view by stating that “true flexibility at the user end is the ability of the user to develop a system that matches precisely the way the user views the world at the moment”.[1]

3.8 Challenges to TIS

The small amount of research conducted on TIS and the conceptual weakness of the academic field of information systems, subject the concept to numerous challenges even at theoretical level. They arise, mainly, from the shift of power to users from developers. This decentralisation of system control is both the core competency and Achilles' heel of TIS. I will discuss a collection of the main challenges that face TIS based on the research done so far. These challenges are by no means a complete representation of the concept’s possible pitfalls but should serve to guide future research and highlight areas of concern.

[1] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 84.

3.8.1 Information Integrity

Information integrity is a concept that resides primarily in the field of computer security. Li et al. state that it is mainly concerned with the prevention of the "accidental or malicious destruction of information".[1] In an attempt to protect itself against such destruction, organisations enforce information integrity policies to protect valuable or sensitive information.[2] These policies are embedded into information systems through various forms of technological or procedural interventions.

Li et al. expand the concept by drawing on Pfleeger's definition of information integrity that stipulates four conditions to which data must adhere at all times. These conditions are:

1. Data must be *precise* - only modified in acceptable ways.
2. Data must be *accurate* - only modified by authorised people.
3. Data must be *unmodified* - only modified by authorised processes.
4. Data must be *consistent* - meaningful and correct.

These conditions form a framework by which developers plan and implement information system security, ensuring information integrity. TIS, being continuously developed by non-developers, face a series of challenges for each of these conditions. I will briefly discuss them below.

3.8.1.1 Data must be precise

If an information system empowers users to develop or adapt applications to match emerging requirements, the ways in which data will be modified are unpredictable. Whether they are acceptable or not will depend on the user's level of sophistication and the type of data that is modified.

---

[1] Li, P., Mao, Y., Zdancewic, S. 2003. *Proceedings of the 2003 Workshop on Formal Aspects in Security & Trust (FAST).* p 53.

[2] Manton argues that the handling of sensitive information "is a complex subject in its own right" and that procedures should address the storage, transfer, ownership and accessibility of sensitive information. Manton, S. 2005. *Integrated Intellectual Asset Management.* p 60.

3.8.1.2 Data must be accurate

Information systems implement user authorisation that is based on organisational structure and the various user responsibilities. Access rights are rigorously controlled by managers and implemented by developers to ensure the protection of information from unauthorised users. These interventions naturally limit system adaptability by users as they are restricted in their access to data. TIS, however, opposes the implementation of restrictions that confine users in their ability to adapt the system. On the contrary, TIS suggest that users should be empowered with access to data, yet be sophisticated enough to guard information accuracy.

3.8.1.3 Data must be unmodified

The development of software applications through the writing of program code results in automated processes that execute commands on data. These processes are extensively tested in software development lifecycles to ensure that they do not corrupt the data in any way. Skilled programmers are aware of the pitfalls associated with data handling and have been trained to avoid them through the use of programming techniques like exception handling. To expect users to implement these techniques when developing automated data handling applications, regardless of how simple they are, is to be overly optimistic. As a result, TIS will have to implement a form of protection against processes that manipulate data without authorisation.

3.8.1.4 Data must be consistent

Finally, to ensure that data is always meaningful and correct, TIS will rely heavily on the accuracy of system adaptations done by users. If the IT department handled all system adaptations, the data consistency can be controlled and monitored by developers. This greatly simplifies the allocation and correction of erroneous parts of the system.

Kanellis and Paul point out that ensuring information integrity is one of the main challenges that face flexible information systems.[1] The systems architecture model proposed by Stamoulis et al. suggests a layered approach that will allow the implementation of

[1] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 82.

authorisation control to protect information integrity, but this will certainly be at the cost of flexibility. Ultimately, the responsibility to maintain information integrity in TIS will reside with empowered end-users who operate under “the necessary culture, policies and controls”[1] to avoid the destruction of information.

3.8.2 Systems Integration

A major challenge that results from the decentralisation of information system control is the ability to integrate systems that were developed at different places and by different people within and outside the organisation. Contemporary information systems normally draw on a variety of software applications and tools that are integrated over a network by IT experts to form the complete technological infrastructure of the information system. Fitzgerald refers to this practice as “industrialisation of information systems”[2] and states that:

> “users are constructing their own information systems out of several suppliers’ modules, integrated together by specifically written software which incorporates local practice”.[3]

This process entails important technical issues such as “reliability, security, throughput, and synchronisation”.[4] When users in a TIS environment adapt the information system by the development of side applications, they are often unable to integrate these with the rest of the system. This results in a collection of inefficient standalone solutions that require manual data input for operation.

Besides the *ability* to integrate their applications, users often also lack the *authority* to do this. Avison and Fitzgerald argue that although users can develop their own applications, these often disobey the rules of information integrity. As a result, IT departments are reluctant to allow the integration of user developed applications and the “live-running operational systems”[5] of the organisation. This was a key challenge to the early instances of EUC that, as

---

[1] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 85.

[2] Fitzgerald, B. 1996. *Information Systems Journal*. p 20.

[3] Fitzgerald, B. 1996. *Information Systems Journal*. p 20.

[4] Satzinger, J.W., Jackson, R.B., Burd, S.D. 2004. *Systems Analysis and Design in a Changing World*. p 322.

[5] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 85.

discussed previously, share many of the ideas of TIS. To overcome this obstacle, users extracted data from the operational systems and used it in their applications. This resulted in further complications, especially in the form of conflicting data in different parts of the system.

The technical intricacies of systems integration call for the skills of professional systems developers and, as a result, obstruct the shift of power to users. Integrated systems are crucial as they provide, if implemented correctly, an automated, secure and accurate form of data flow between applications.

3.8.3 Evaluation

Patel and Irani argue that "in general terms, there are no evaluation techniques for evolutionary information systems"[1] such as TIS. The problem of evaluating evolutionary or living information system calls for a new evaluation technique that focuses on a system's usability over a period of time. Traditional evaluation techniques are, like SDLC, concerned with a system's performance at a fixed point in time, treating the information system as a closed system and focussing mostly on the affectivity of technological application. Kanellis and Paul argue that "the myriad of reasons that determine whether an information system is successful or not can be matched by an equal number of explanations".[2] Checkland and Holwell also recognise the challenge of evaluating information systems by stating that "it is never easy to evaluate complex interventions which bring about significant change in complex organisations".[3]

The evaluation of living information systems that are dynamically updated is a far more complex problem than the evaluation of a closed information system. The main dilemma is that "the actual information systems product is continuously being changed because of changing organisational needs, which result in the need to continuously assess related costs and strategic issues".[4] As the system evolves, users gain both knowledge and value from it, creating the need for the recognition of social and contextual factors in the evaluation of the

[1] Patel, N.V., Irani, Z. 1999. *Logistics Information Management.* p 37.

[2] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 65.

[3] Checkland, P., Holwell, S. 1998. *Information, Systems and Information Systems.* p 194.

[4] Patel, N.V., Irani, Z. 1999. *Logistics Information Management.* p 37.

system. Although attempts to create an evaluation model for dynamic system environments have been made,[1] it remains a daunting challenge to measure the actual value that is added to the organisation through the system. Accordingly, Patel and Irani argue that the benefits associated with such systems are often transitory.[2]

TIS are specifically aimed at overcoming the vulnerability of an information system to contextual change. Its true value will thus be most prominent in the face of such change. One can argue that value will emerge, specifically when unforeseen requirements emerge, emphasising, once more, the complexity of an evaluation technique for such systems.

3.8.4 Dependencies

IT departments ideally operate in a fashion that ensures that the technological infrastructure of an information system does not depend on the skills or knowledge of a single person. In doing so, they can guarantee that the system will be maintained independent of a single person's physical presence. To manage system knowledge among the members of a single department in this way is a much easier task than to do the same for all the users of the system. This problem, also prominent in EUC, is another direct result of the decentralisation of control over the system.

Besides managing knowledge within the IT department, professional developers also obey certain rules when adapting systems to ensure transparency. One such rule, for instance, specifies the provision of comment statements within program code that explain the developer's choice of commands. These rules allow developers to interpret each other's work and adapt it if the need arises. Avison and Fitzgerald argue that users, when developing, do not obey these rules as they underestimate their importance in the larger organisational context.[3] To complicate the situation even more, code developed by users is often void of good structure and logical design making it difficult to unravel at a later stage. This inexperience on the side of users results in applications that heavily rely on the knowledge of a specific person. Once that person is removed from the information system, the code that

[1] Patel and Irani briefly mention these. Patel, N.V., Irani, Z. 1999. *Logistics Information Management.* p 37

[2] Patel, N.V., Irani, Z. 1999. *Logistics Information Management.* p 37.

[3] Avison, D., Fitzgerald, G. 2006. *Information Systems Development.* p 85.

he/she developed becomes less maintainable. The problem can be controlled if every user depended only on code they developed themselves and continued to work in the same organisation and position for an extensive period of time. But it is extremely unlikely that these conditions will hold in an organisation operating in a turbulent business sector. Any manager would recognise and guard against the dangers associated with irreplaceable personnel. Unless TIS enforce regulations to protect the information system against dependency on key employees, system failure will be a constant threat.

3.8.5 Efficiency

TIS imply that system users will undergo a continuous and often steep learning curve. The nature of the concept requires users to hone their technical skills and become proficient in the applications and tools provided to them by the IT department. In the long run, the organisation will benefit from having smarter employees, a key competency in the knowledge economy. The problem that accompanies this learning curve, however, is the efficiency of business processes.

The bureaucratic organisations that dominated the industrial era were associated with employees that acted as parts of a machine, performing basic repetitive tasks. Unlike the learning organisations of the knowledge era, these organisations did not expect their employees to continuously broaden their knowledge and skills. They were merely required to do the job at hand in the most efficient way possible. Employees were likely to be penalised when making a mistake or not following a prescribed procedure. Organisations that implement TIS stand in stark contrast to these as they depend greatly on the knowledge and skills of their employees. As can be expected, this dependency comes at a certain price. The time invested in training users to adapt the system together with the risk undertaken to decentralise system control are only two examples.

The main concern in terms of TIS, however, is the time it takes a user to develop a solution or make a system adaptation versus the time it would have taken an IT expert to do the same. Coupled with this is the amount of replica development done at different places in the organisation due to a lack of central authority. Department A, for example, might develop an application that can be used by other departments, yet unknowingly Departments B and C undergo similar development processes. This is clearly inefficient. One of the key strengths of software, especially object-orientated software, is the ability to reuse code by slightly adapting it to fit different requirements. Developers are trained to exploit this extensively to ensure efficiency; developing users can't be expected to do the same.

3.8.6 Learning from EUC

The commonalities that exist between TIS and EUC allow the identification of challenges that have been associated with EUC and will most likely be relevant to TIS. Some of the challenges discussed above are among those that led to EUC becoming "unsustainable in some organisations".[1] Checkland and Holwell note that, although having associated risks, EUC was rated as the second most important issue in the information systems field in the mid-1980s.[2]

A definition for EUC supplied in this thesis is "the practice of end-users developing, maintaining and using their own information systems" (see section 3.5). Users in any organisation will, to some degree, use applications other than the main running system to perform their daily work. Packages like Microsoft Office have become synonymous with business operations and users are expected to master and apply them habitually. It is important to distinguish EUC from such standard operations. Writing a report in Microsoft Word does not involve the development or maintenance of an information system. Programming a Microsoft Excel sheet, however, involves the translation of business logic into formulas and can certainly be seen as information systems development. Naturally it is a fine line and users in TIS will have the responsibility to know the difference.

A consequence of early EUC initiatives was the categorisation of users into groups according to their ability to develop applications. Rockart and Flannery report this as early as 1983[3] and state that the following groups were recognised:

1. Non-programming end-users
2. Command-level users

---

[1] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 85.

[2] Checkland, P., Holwell, S. 1998. *Information, Systems and Information Systems*. p 226.

[3] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 84.

3. End-user programmers
4. Functional support personnel

Organisations implementing TIS will, without a doubt, face a similar scenario. As users become more proficient as developers, they will become more valuable to the organisation in terms of maintaining the information system. Proficient users will be able to develop applications more efficiently and weaker users, as well as the system itself, are likely to become increasingly dependent on them. Managers might consider freeing such employees from normal organisational responsibilities to encourage them in their development efforts; this might also involve increased remuneration.

Stair and Reynolds refer to EUC as "end-user systems development"[1] and argue that developers should encourage it by providing guidance and support rather than fight it. They too, however, warn of the dangers that can be associated with small development mistakes by stating that "multi-million dollar mistakes can be made using faulty spreadsheets that were never tested".[2] Laudon and Laudon argue along the same lines noting that some organisations have reported an increase in development productivity of 300%-500% after implementing EUC; but the fact that EUC occurs "outside of traditional mechanisms for information system management and control"[3] poses major organisational risks.

To manage the relationship between risk and reward in an EUC environment, many organisations established information centres or IC's. IC's were business units that had both the technology and expertise to assist users in their development efforts. They also served to control EUC and ensure system integrity despite the mistakes made by inexperienced users. IC success depended on a number of factors[4]:

- Organisational commitment to the IC, including the support of top management.
- The rank of the IC executive.
- Commitment of the end-users.

---

[1] Stair, R., Reynolds, G. 2006. *Principles of Information Systems*. p 574.

[2] Stair, R., Reynolds, G. 2006. *Principles of Information Systems*. p 574.

[3] Laudon, K.C., Laudon, J.P. 2004. *Management Information Systems*. p 398.

[4] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 86.

- An adequate budget.
- The active promotion of IC services.
- The quality of IC staff.

Laudon and Laudon argue that the role of IC's is "diminishing as end-users become more computer literate".[1] This argument is supported by Avison and Fitzgerald who in turn state that their time has passed with the "increasing ability of users to develop their own systems for the new distributed Internet world".[2]

Independent of the existence of IC's, there is a general consensus that "EUC is here to stay".[3] The increasing sophistication of users and technology will combine to make development a more accessible practice at all levels of the organisation. How this development will be managed and controlled remains to be seen, but Avison and Fitzgerald argue that the IC's are required to "transfer from a tactical problem-solving perspective to a more strategic, consultant-like role".[4]

3.9 Recognising a Trend

The decentralisation of control over an information system is not bounded to information systems in organisational use. Although this study has focused mainly on such systems, it is worth recognising that this trend can also be witnessed in other system domains. Two examples are particularly applicable, the first is Web 2.0 and the second is Peer to Peer networking.

---

[1] Laudon, K.C., Laudon, J.P. 2004. *Management Information Systems*. p 87.

[2] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 86.

[3] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 87.

[4] Avison, D., Fitzgerald, G. 2006. *Information Systems Development*. p 87.

3.9.1 Web 2.0

Web 2.0 refers to a collection of second-generation web-based technologies that are aimed at dissolving the lines between the publishers and readers of the web. Services like Wikis and Blogs allow readers to dynamically change the content and design of publicly available web pages. This is facilitated by user-friendly, online interfaces that shield readers from the technological complexities of web publishing yet enables them to produce professional looking web pages.

The most popular Wiki on the web is an encyclopaedia called Wikipedia.[1] It allows any user to add, edit or remove information on any one of over 1 451 000 English articles[2] and research suggests that it is more accurate than commercial encyclopaedias like Microsoft's Encarta.[3] Nevertheless, the Wiki concept was initially received with scepticism. Many feared that Wikis would be targeted by vandalism and that too little users would commit to updating their content. This initial criticism was silenced by the significant popularity of Wikipedia and similar sites. Despite the complete decentralisation of control over the system content, Wikis continue to operate efficiently. This is primarily due to the combination of technology that enables readers to control the system and the commitment of the Wiki community.

Although a comparison between TIS and Wikis would not be justified, there are certain commonalities that deserve recognition. Wikis, like TIS, are developed through the ateleological paradigm and can be seen as open, living information systems. The decentralisation of control employed by Wikis is a prominent feature of TIS and critical to the success of both. Wikis and TIS both depend on technology to enable system control by users despite technical complexities and finally, both Wikis and TIS are responsive to environmental change through the awareness of users who adapt the system in real-time to match its environment.

The key difference between the two systems is that TIS do not merely require users to adapt information, but also expect them to adapt system functionality. Nevertheless, organisations

[1] Available at http://www.wikipedia.org.

[2] As reported by Wikipedia.org on 25/10/2006. Articles also appear in a variety of other languages. (Wikipedia, 2006)

[3] Schwall, J. 2003. *The Wiki Phenomenon*. p 10.

are more controlled environments than the World Wide Web and through the dedication of IT experts, end-users and management, there seems to be enough evidence supporting the fact that the challenges discussed earlier can be overcome.

3.9.2 Peer-to-Peer Networking

Peer-to-Peer (P2P) networking became popularly known through the controversial Napster MP3 file sharing network that was the first widely used P2P network.[1] It offers an alternative way of designing a computer network that has significant advantages over the traditional client-server model which operates around a centralised server or collection of servers. In contrast to this design, pure P2P networks combine the storage space and bandwidth of computers at the edge of the network to form more flexible networks. Some P2P networks, referred to as hybrid P2P, employ centralised servers to some degree with the purpose of improving network performance. The primary idea, however, is to decentralise the central point of control over the network by allowing peers to act as clients and servers simultaneously.

A key strength of P2P networks is their robustness. The fact that the network does not rely on the operation of centralised servers makes it less vulnerable to environmental contingencies. When the network has grown sufficiently, peers can connect to or leave the network without it having any significant effect on the network's overall performance; the network's strength thus lies in its decentralised nature. P2P technology has already infiltrated organisational information systems as many managers have realised its advantages.

A key commonality that TIS will share with P2P is the distribution of technology across a network as opposed to a central point of control. In the same way peers control their own participation in the network by sharing and downloading files, the users in TIS will control their own part of the information system. This suggests that TIS might inherit the same robustness of P2P networks by eradicating the traditional reliance on a central point of control. The information system will be able to cope more efficiently with unexpected events like server downtime.

[1] Haag, S., Cummings, M., McCubbrey, D.J. 2005. *Management Information Systems*. p 3.

Note that this does not imply the implementation of P2P networks in TIS. Although this might be a feasible design decision, it is the theoretical perspective that is applicable here. The decentralisation of technology should not only be interpreted negatively, P2P networks serve as proof that there are reasons to encourage it. A P2P network has the ability to organise itself to the changing needs of peers, acting like a living system that is responsive to change without the mediation of a network administrator and centralised servers. It thus shares many of the goals of TIS.

3.10 Summary and Conclusions

Organisations undergo internal change as a result of the turbulence of the business sector in which they operate. This sector is made up of numerous elements that may include competitors, regulators and customers over which the organisation has limited or no control. The internet has played a significant role in increasing the turbulence of business sectors by forming a foundation for a growing global economy in which organisations can cross geographical boundaries and enter previously inaccessible markets.

Organisational change has a significant impact on information system success. Information systems developed through the teleological paradigm are vulnerable to contextual change, being unable to adapt to changing or emerging requirements. The concept of information systems misfit signifies a situation where the requirements of the system's users do not match the requirements for which the system was originally developed. As the organisation continues to change, the misfit between information system and context increases; ultimately leading to system failure.

Despite being developed through teleological methodologies and operating in changing contexts, many information systems continue to function with success. The case study discussed in this chapter describes this phenomenon as the emergence of flexibility and suggests that it occurs as a result of the sophistication of system users who keep the system functional by extending it with personally developed side applications such as spreadsheets. Although users acting as developers provide challenges in terms of system control, managers have realised that the survival of the information system relies on them. To develop an understanding of information systems flexibility through users acting as developers, Stamoulis et al. apply the theories of operational closure and self organisation to information systems. In doing so they indicate that information systems can achieve flexibility by maintaining internal coherence despite existing in a changing organisational context.

These arguments lay the foundation for Tailorable Information Systems which achieve system flexibility by decentralising system control. Instead of relying on system maintenance by IT departments, TIS call for a shift of power towards the end-users of the system, allowing them to adapt the system when their requirements change or new ones emerge. This shift of power involves the diffusion of technical knowledge by IT experts and the availability of tools that enable users with limited technical ability to adapt system functionality. Such systems are continuously developed and do not operate on a fixed set of requirements. This approach to systems development is referred to as the ateleological paradigm as it opposes the development methodologies of the teleological paradigm.

The concept of TIS is subject to a series of challenges that obstruct the development and maintainability of living information systems. On introduction to TIS, one tends to think that the main challenge would be to increase the sophistication of users to enable development by them. Research, however, has indicated that this is merely a step towards the actual problem – controlling user development. The decentralised nature of TIS however, should not only be interpreted as hazardous. Other types of information systems, like Web 2.0 applications and P2P networks, have applied the same principle with much success.

Management will play a crucial role in fostering an environment where user development is encouraged yet regulations are employed to ensure system integrity. Users will have to accept that their power as developers is accompanied with a great deal of responsibility and that there are major risks associated with small mistakes. To impose strict rules on users will inhibit them in their development efforts, managers will need to find alternative ways to control the risks associated with TIS.

The role of the traditional IT department will be of essential importance in a TIS environment. It is not clear yet how their services can be best employed but it will most definitely entail closer collaboration with users to allow the diffusion of their technical knowledge. Closer collaboration will also increase their awareness of user trends and improve their ability predict danger areas and control risks. IT experts and system developers that approach TIS with open minds and have the ability to apply their skills in new ways hold the key to overcoming the challenges that face TIS.

Chapter 4
Combining Paradigms

4.1 Overview of the Study

This section will provide a brief overview of the study, outlining the findings and conclusions of each chapter. References will be made to the specific section where arguments were constructed.

To enable organisational change and adaptation, academics and managers alike have realised the need for a move away from the bureaucratic management styles of the industrial era. Themes like knowledge management, cluster structures, networking and organic organisation have received increased attention[1] as managers realise the need to decentralise organisational control and rely on the knowledge and skills of employees to act in the organisation's best interests. Academic literature on organisations reveal this transformation; and globalisation fuels the need for organisations to implement a more flexible organisational culture in turbulent business sectors. Despite this revolution in organisational science, the academic field of information systems has been slow to follow and grow out of the original ideas about systems development and implementation (see section 2.3). Contemporary information systems development projects are still dominated by a scientific, reductionist approach that fails to recognise the various complexities associated with such projects. I have argued that this is a futile practice and that it can, to a large extent, be held responsible for the large amount of failing information systems (see section 2.4.3).

The ambiguity that riddles the academic domain of information systems and the ongoing evolution of technology for organisations, are only two of the factors that make information systems an extremely complex research subject (see section 1.6); let alone the technicalities of information systems development and use in practice. The relative youth of the academic

[1] Kwon, D., Oh, W., Jeon, S. 2007. *Journal of Management Information Systems.* p 202.

field is evident in the amount of conflicting views and interpretations in information systems literature. Despite this, there have been significant developments towards information systems that can operate in frequently changing organisations.

Contemporary information systems development practices are dominated by a *teleological paradigm* (see section 2.1). It implies that the process of information systems development can be reduced to the specification and production of a system that adheres to a set of predefined requirements. The SDLC and similar methodologies typically give structure to this process and are aimed at delivering systems that adhere to these requirements as closely as possible (see section 2.4). In theory and from a purely operational point of view, the teleological paradigm seems both sensible and practical. Information systems, however, are not machines (see section 1.7.2.4). On the contrary, they should be studied as complex socio-technical systems, consisting of a combination of the people, processes and technology within organisations (see figure 1.3). The practice of developing these complex, socio-technical systems with a scientific, reductionist approach leads to an underestimation of the multi-dimensional complexity that characterises the process (see section 2.3). Maguire agrees with this conclusion by referring to the stages of the SDLC as “the key elements of the *computer systems* development process”[1]; *information systems* development entails more than “the structuring of hardware and software to achieve efficient processing”.[2] He accurately concludes that this “mechanistic bias within the systems development process may be a major factor leading to information system failure”.[3]

System analysts and developers have re-invented teleological development methodologies with an emphasis on the ongoing maintenance that the system will undergo post implementation (see section 2.5). Techniques like Rapid Applications Development (RAD), Agile development and Extreme Programming have shaped *evolutionary* development methodologies that have become standard practice in organisations with the capacity to employ experienced developers. These methodologies are successful in their purpose to reduce the time it takes developers to perform system maintenance, producing a more flexible information system. Although they produce systems that continue to rely on the technical

[1] Maguire, S. 2000. *Information Management & Computer Security*. p 231.

[2] Maguire, S. 2000. *Information Management & Computer Security*. p 231.

[3] Maguire, S. 2000. *Information Management & Computer Security*. p 232.

skills of developers, their emphasis is no longer on detailed requirements specification, but rather development as an ongoing process. This change of emphasis is a natural result of the fact that organisations spend more money on system maintenance than system development (see figure 2.2). Systems that are continuously developed through these evolutionary methodologies can be seen as *self-organising systems* as they clearly have the ability to interact with their environment and adapt themselves accordingly. This opposes the traditional systems development methodologies and indicates a movement away from the teleological paradigm. The information systems are, however, still bounded by the dependencies of system maintenance (see figure 2.5). The authority to adapt the system is centralised at the computer system developers who control the technical, automated components of the system.

It is essential to accentuate that the teleological paradigm, despite its criticism, is and will continue to be fundamental in any information systems development project. Although this study has focused primarily on organisational information systems for turbulent business environments, a myriad of information systems are developed to operate in different environments. Some of these environments are stable and system developers can accurately stipulate the requirements that the information system should adhere to. One example is the information systems used to monitor human vitals in intensive care units at hospitals which must, above all, operate accurately and consistently. Their strength lies in stability, not flexibility. Likewise, flexibility at user level is not an important prerequisite for flight control systems where the margin for error is non-existent. Applegate et al. support this view and state that the key to systems development lies in the selection and implementation of a methodology "based on the nature of the project and the experience and expertise of both business and IT professionals".[1]

The *ateleological paradigm* (see section 3.1) opposes the development of information systems based on a fixed set of pre-defined requirements. On the contrary, it suggests that any set of requirements for an information system is contextually sensitive, making the developed system fragile in a changing environment (see section 3.3). The notion of Tailorable Information Systems (see section 3.6) provides a more suitable approach for developing and maintaining information systems in turbulent business environments and

[1] Applegate, L.M., McFarlan, F.W., McKenney, J.L. 1999. *Corporate Information Systems Management.* p 32.

exemplifies the principles of the ateleological paradigm. It proclaims that users should be empowered to develop and maintain their own information systems, enabling the dynamic adaptation of the system to changing or emerging requirements (see section 3.6.3). TIS, as a result, overcome the dependencies of system maintenance (see figure 3.3).

Minimal research has been done on the ateleological paradigm and information system approaches like TIS. A common notion, however, is that the technology to develop and support these information systems are not available yet. Such technology will shield users from the technical complexities of systems development yet give them the power to control these systems without jeopardising information integrity. A systems architecture model proposed by Stamoulis et al. suggests that TIS will employ various layers of technology to enable flexibility in spite of technical intricacy and the lack of technical skills on user level (see figure 3.4). It embodies the key principle of TIS: the decentralisation of system control to enable real-time adaptation and maximum flexibility at end-user level.

The TIS approach is associated with numerous challenges (see section 3.8), not only on a technical front but also from a managerial point of view. The decentralisation of information system control will result in a series of management problems and potentially enormous risks. Technology, independent of its sophistication, will only partly enable ateleological systems development; it would rely largely on an innovative organisational culture that fosters continuous organisational learning. Besides their technical sophistication, users in such organisations will have the responsibility of continuously learning to apply technology to overcome business problems, work traditionally performed by IT professionals. If these challenges can be overcome, the ateleological paradigm will have a significant impact on the future of the information systems development in organisations. The increasing sophistication and technical ability of the average organisational employee, coupled with technology to overcome the challenges that currently face concepts like TIS, will path the way for the implementation of truly flexible information systems.

4.2 Paradigm Selection

From the research conducted and the arguments produced in this thesis, it is my opinion that both paradigms, teleological and ateleological, will ultimately prove essential in information systems development projects. Maguire notes that there have been a number of ingenious attempts to combine hard and soft development methodologies[1], a solution that seeks to find a balance between flexibility and control in information systems. Independent of user sophistication TIS will always rely, at some level, on systems that were developed to adhere to predefined sets of system requirements. Flexibility can only be achieved if these systems form a foundation upon which end-user development can be done.

This conclusion is in line with the proposed systems architecture model which specifies a multilayered approach (see figure 3.4) and introduces the need for some form of guidance in terms of paradigm selection on higher system layers, specifically those that regulate organisational practices and routines.[2] Based on figures 3.4 and 3.5, it is clear that certain components of the information system will be subject to more contextual turbulence than others. These components, specifically, require the inherent ability to manage changing or emerging requirements.

[1] Maguire, S. 2000. *Information Management & Computer Security*. p 230.

[2] Avgerou confirms this need by arguing that information system professionals must be able to judge the potential, develop and manage information systems in a changing context. He adds that this requires the development of the "capacity to take into account the processes of change across layers of context, and to judge the forces that contribute to the realisation or impede initial plans in order to pursue feasible action." Avgerou, C. 2001. *Information Systems Journal*. p 60. The paradigm selection model presented in the next section is specifically applicable in this sense.

4.2.1 A model for paradigm selection

Based on the research and findings of this study, a 3-dimensional model for paradigm selection was developed. This model combines the main factors that are applicable in the selection of a paradigm for information systems development in an organisation.[1] It encapsulates the findings of this study and facilitates a contextual analysis for information systems. The model is presented in figure 4.1.

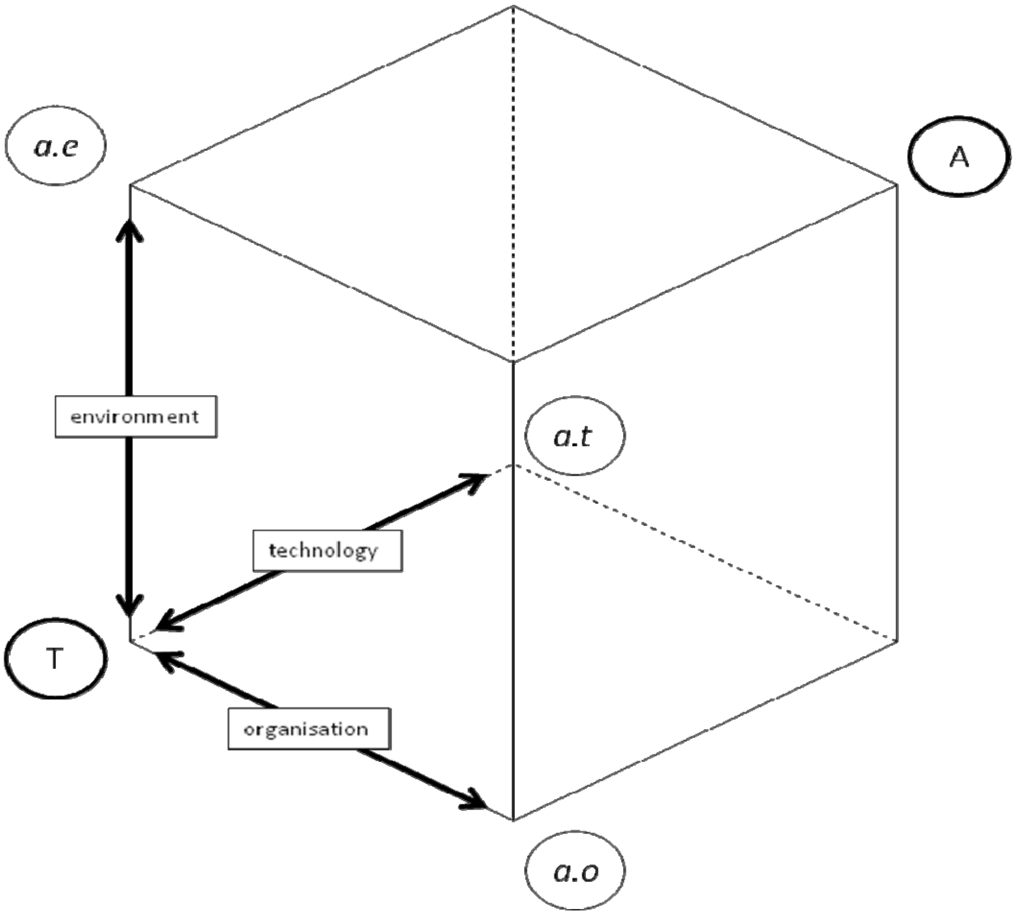

Figure 4.1: A 3-dimensional model for paradigm selection.

[1] The term *organisation* will be used throughout this model's description and discussion. It should be noted, however, that not all departments of an organisation are subject to equal turbulence. I therefore suggest that to be more effective, the model must be applied to the direct organisational context (the specific department) within which development is done, as opposed to the organisation or information system as a whole. See section 4.2.3 for a discussion.

The model employs 3 axes that represent various aspects of the organisation:

- The *environment* axis represents the organisation's external environment.
- The *technology* axis represents the technology and technological sophistication of the specific component.
- The *organisation* axis represents other organisational attributes, like management style and structure, which will influence paradigm selection.

The model has 5 marked points:

- Point *T* represents the point in the model where the teleological paradigm is most applicable.
- Point *A* represents the point in the model where the ateleological paradigm is most applicable.
- The other three points (*a.e*, *a.t* and *a.o*) mark the furthest ends of the three axes from point *T*.

Each axis has three factors which combine to determine where on that axis a component must be plotted. Point T marks *'Teleological Extreme'* while the relevant point at the other end of the axis (*a.e*, *a.t* or *a.o*), marks the *'Ateleological Extreme'*. The three positions on the axes are then combined to give a final position within the model. The factors applicable to each axis are discussed in table 4.1.

Teleological Extreme	Ateleological Extreme
The *environment* axis (T, a.e)	
T	**a.e**
Regulators	
The business sector within which the organisation operates is not affected by frequent regulatory changes and remains stable for considerable periods of time.	Regulators frequently force organisations to change various aspects of their business. This results in constant internal organisational change and frequent changing or emerging information system requirements.

Competitors	
The organisation operates within a niche market where few or no competitors threaten to decrease the current market share.	The organisation competes with a large amount of organisations that target the same market and continuously search for new competitive advantages.
Customers	
Customers do not expect the product or service to change or improve and are unlikely to support a different supplier for any specific reason. Prices may be regulated or small enough as not to affect choice of supplier.	Customers can consider a variety of suppliers and will base their decision on the supplier's ability to find a competitive edge. Organisations must constantly innovate to ensure that they maintain market share through customer communication and service.
The *technology* axis (T, a.t)	
T	**a.t**
Technological infrastructure	
The organisation relies heavily on very specific or advanced information system functionality and is rarely confronted with changing or emerging requirements that were not knowable before systems development.	Advanced technological infrastructure is in place and the organisation relies on a variety of applications that are added and removed as the need arise. A foundation of operating software is employed upon which users can add applications that they require for specific projects.
Information technology department	
The organisation permanently employs IT professionals to maintain the information system on a regular basis and users are not expected or allowed to adapt system functionality in any way. The organisation may also rely completely on the services of external contractors that update the information system at set intervals.	The organisation employs IT professionals to assist users in the development and integration of applications on an ongoing basis. IT professionals do not follow specific routines or procedures but continuously diffuse technological knowledge to users to improve their technical skills. They also ensure that the underlying technological infrastructure supports user development.

End-user sophistication	
End-users have no or very little technical skills or information system authority and merely use system functionality provided to them. The organisation does not expect nor encourage end-users to improve their technical skills.	End-users are technologically sophisticated and have the skills and authority to adapt the system dynamically. The organisation has initiatives that encourage the technological empowerment of end-users.
The *organisation* axis (T, a.o)	
T	**a.o**
Management	
Centralised management implements rigorous control structures by which employees are forced to work within pre-defined procedures and accurately adhere to the rules and regulations of information system usage.	Decentralised management encourages employees to innovate and dynamically develop the information system to match their changing working procedures. Management does not penalise employees for data errors resulting from end-user development.
Structure	
The organisation has a hierarchical structure that represents and strongly implements organisational authority. This structure is reflected in the information system by limiting users to parts of the system based on their position.	The organisation does not implement a formal structure but allows all employees to network independent of their position within the organisation. Employees are given authority in their use of the information system which does not implement rigorous authentication levels but rather trusts employees to handle information responsibly.

Information security policy	
The organisation mainly deals with sensitive information of which the privacy must be protected from unauthorised employees and outsiders. To enable this protection, the organisation employs a very strict information security policy that is implemented within the information system architecture.	The organisation mainly deals with information that is not of a private or sensitive nature and does not require policies to regulate information security. Employees are expected to use their personal judgment or consult management when an information security matter arises.

Table 4.1: The axes and variables of the paradigm selection model.

4.2.2 Applying the model

When the model is applied, the three points plotted on the axes must be combined to produce a point within the model of which the position relative to points T and A is of value. Figure 4.2 presents an example of what an organisation's profile would look like within the model, with point *f* marking its position. The example profile suggests a turbulent external environment that would require information system flexibility. However, both the technology and organisation axes suggest a teleological approach. This can be evident of a strong reliance on fixed procedures and advanced computerised technology applied in the information system.

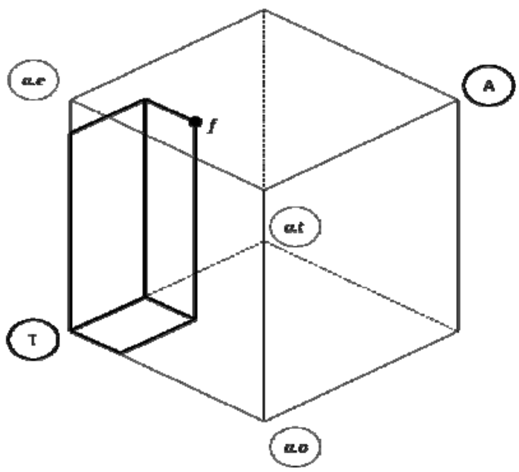

Figure 4.2: Application of the model.

4.2.3 Applying the model to a decomposed information system

The application of the model for paradigm selection can only prove successful if the information system is decomposed into its various functional subsystems (both technological and social).[1] Each component of the system should then be assessed separately and developed with the paradigm most suitable to its direct context. There are several arguments to support such an application of the model:

- The process of system decomposition is a fundamental step in the analysis and design phases of traditional approaches like the SDLC. Abstractions like DFD's (Data Flow Diagrams), Use Cases and similar process models are typical documentation produced before implementation commences. Using the paradigm selection model, the contexts of these information system components can be assessed. An ateleological paradigm can then be adopted for those components that are likely to be subject to contextual turbulence. The changes impacting these components would, unlike figure 2.2, not result in major costs, as they are not bounded to pre-defined requirements. This approach supports the combination of hard and soft methodologies as proposed by Maguire (see the start of section 4.2).
- The information system as a whole would be less sensitive to contextual turbulence. Structural coupling between the information system and its context can be achieved by ensuring flexibility at key components of the information system, minimising the need for maintenance by developers. This can ensure the dynamic handling of instances where information systems misfit negatively influences organisational productivity.
- By focusing ateleological development on specific information system components, the challenges and risks associated with end-user development are limited. By confining end-user development to these components, the information system itself can be protected from possible user errors made in these components. The primary

[1] Satzinger et al. use the phrase *functional decomposition* to refer to the process of breaking a system up into components based on subsystems that in turn are divided further into subsystems. Satzinger, J.W., Jackson, R.B., Burd, S.D. 2004. *Systems Analysis and Design in a Changing World*. p 7.

concern then would be the integration of these components into the system in a way that ensures the maintenance of data integrity, without sacrificing the productivity gained through automation. The interface between various system components and the definition of dataflow among them will be complicated by flexible system components. It is my opinion that these interfaces will become increasingly important as developers seek means to increase information system flexibility.

- Finally, this approach would assist managers in recognising those areas of the system where technologically competent end-users are essential. The specific skills required may differ from one ateleologically developed component to another, enabling managers to develop specific training initiatives for end-users, as well as focusing the efforts of IT professionals in their role of assisting end-users.

The application of the model to a decomposed information system necessitates the need for traditional systems analysis activities, yet opposes the specification of a complete set of fixed requirements that typically results from them. This hints, once more, to a combination of the two paradigms in an attempt to manage the balance between flexibility and control within the system.

4.2.4 Designing Flexibility

In section 3.4, Kanellis and Paul discuss flexibility as an emerging property of the information system under scrutiny in their case study. They argue that flexibility was not planned or designed but emerged as a result of various organisational factors. Based on the findings of this study, however, it is clear that certain guidelines can be provided to specifically promote information system flexibility. The technology and organisation axes of the model for paradigm selection, presented in section 4.2.1, can be used to categorise these guidelines. They are presented in table 4.2.

Technology	
Technological infrastructure	Flexibility at end-user level can be promoted by supplying end-users with a scalable technological infrastructure, enabling them to install new applications as the need arises. This must be done with the support of the IT department and users should have access to their expertise at all times. It is important that users have secure storage space (i.e. drive space on a file server) where business data can be protected from loss of integrity. To ensure some form of integration between applications, systems should provide functionality to import or export data in various formats. Finally, it is essential that users have good communication tools available to facilitate collaboration.
IT department	The IT department should employ highly skilled, innovative personnel; capable of diffusing their technical knowledge through continuous interaction with end-users. These individuals should have a constant awareness of the changing requirements of end-users as well as the various technologies available to solve business problems. They should focus their efforts on the integration of end-user applications, seeking a balance between flexibility and control in the system. It is of crucial importance that these individuals are good communicators that teach users to perform system maintenance as opposed to maintaining the system themselves. Finally, the IT department will be the key component in controlling the risks associated with end-user development; they should maintain and enforce the policies put in place by management to ensure system integrity.
End-user sophistication	End-user sophistication is the probably the most important enabler of flexibility within the system. End-users must be motivated to learn continuously and apply their skills innovatively to match emerging or changing requirements. The management of knowledge among end-users and IT staff is of vital importance and requires dedication on both sides. End-users should have a willingness to take on new challenges and must not be daunted by the introduction of new technology. This study has shown that universally found technology, like spreadsheets, can have a major impact when applied in meaningful ways.

Organisation	
Management	The management of a flexible information system will involve mindful consideration of various factors on an ongoing basis. It has been argued, in this study, that the ateleological paradigm calls for a process view of information systems, posing new challenges to management. Human resource planning must be done with consideration for technical sophistication required at end-user level as well as communication abilities required in the IT department. Management should implement policies that have the dual purpose of encouraging end-user development and controlling associated risks, allowing the organisation to learn by doing. Trust between managers and end-users is of decisive importance as it will enable the decentralisation of decision-making, enabling a more responsive system.
Structure	A strongly imposed hierarchical structure will inhibit the possibility of a flexible information system. The need for continuous development at all levels of the organisation calls for a structure that encourages regular interaction across levels in a network-like manner. Although this does not nullify the need for role definition within the system, it encourages innovative thinking at all levels as opposed to end-users merely following orders.
Information security policy	Sensitive or valuable information complicates the formation of an information security policy for a flexible system. It is crucial that the importance of handling such information correctly is communicated clearly to the role players involved – emphasising the need for a policy. The policy should guide end-users in their efforts and limit them where necessary. It should also protect them to some degree and encourage them to apply their skills. Aspects like trust and communication are again integral here as the policy (like the system) will require continuous maintenance.

Table 4.2: Guidelines to achieve flexibility.

4.3 Limitations and Future Research

To further develop the ateleological paradigm and TIS, research is required into both the organisational and technological dimensions of information systems. Avgerou points out that there exists a "crucial need to develop professional capacity to facilitate the technological exploitation in relation to the socio-organisational processes of change within which innovation is embedded."[1] This calls for a breed of professional, technically skilled innovators that grasp the complexities of information systems as social, complex entities. It also implies their involvement in information systems development as an ongoing process as opposed to a once-off project. Such an approach would require researchers to investigate the various implications it holds for practice. The challenges discussed in this thesis provide ample areas of concern that can guide future research to determine whether the proposed paradigm can culminate in the implementation of dynamically flexible information systems.

4.3.1 Empirical Data

The study was slightly limited by a lack of authentic empirical data to support theoretical arguments. The case study performed by Kanellis and Paul (see section 3.2) is a good example of the value of empirical data to explicate such arguments and validate their significance in practice. A large part of the articles referenced in this thesis draw extensively on empirical results, yet certain aspects that are specifically applicable to this study are not investigated. Some notable examples include the following:

- It would be useful to investigate the amount of end-user development currently performed in various organisations and subsequently attempt to describe its influence on the system's efficiency. It has been argued that the ateleological paradigm is already a reality in many organisations and it would be interesting to compare such organisations to competitors.
- The study has revealed the need for managers to find a balance between control and flexibility in information systems. Empirical data may serve to define this balance in more detail and identify best practices for its implementation.

[1] Avgerou, C. 2001. *Information Systems Journal.* p 60.

- Finally, one can attempt to find relationship between the extent to which end-users control the information system (the system's *tailorability*) and system flexibility. Such an experiment, however, would prove a timely exercise as it would only have real value in the face of organisational change.

Although the study drew on statistics of system success and development project complexity, it was notable that the multi-dimensionality of information systems development and use complicate quantitative approaches. Literature, for instance, reveals various arguments on the quantification of concepts like *information system success* (see section 3.3), realising that the social and dynamic dimensions of such concepts complicate the generalisation of variables. The view that information systems are processes embedded within a socio-organisational context necessitates that information systems cannot be studied without attention being paid to a large number of non-technological, mostly unquantifiable factors.

4.3.2 Technological Research

The ateleological paradigm will require the development of technological applications that shield end-users from the technical intricacies of systems development, yet empower them to control automated processes and system functionality. These applications should provide users with interfaces that allow effortless control over information system structure and functionality. Stamoulis et al. suggest that object technology is essential in the design and development of such "end-user tailoring tools" and discusses a combination of these that are already in organisational use.[1]

Besides enabling end-user development, these tools will also have the duty of protecting the information system's integrity from end-user errors. Inexperienced end-users will be prone to erroneous development which is potentially damaging to the organisation as a whole. Professional developers will be required to design tools that limit the risks associated with such errors, ideally supplying more experienced end-users with more authority over the system.

[1] Stamoulis, D.S., Patel, N.V., Martakos, D.I. 1996. *Proceedings of the 1996 European Conference on Information Systems*. p 319.

To enable information system control despite decentralisation, technology in TIS will have to allow transparency between end-users and IT professionals. Although end-users should be given as much freedom as possible to develop their own solutions, IT experts should be able to assist them promptly when the need arises. This, however, should not lead to a re-establishment of the maintenance dependencies (see figure 2.5), but rather encourage and enable organisational learning at end-user level.

4.3.3 Organisational Research

To allow the implementation of TIS, organisations will need to adopt a new, and significantly different, approach towards information management. Information system flexibility can only be achieved if the people in the organisation support continuous change by being flexible themselves. Naturally, a bureaucratic management style will not encourage end-users to take on the challenge of systems development with the associated risks. Management will need to support their participation by giving employees freedom and support without sacrificing their own authority. This should serve as a starting point for future organisational research in TIS.

Kanellis and Paul state that flexibility in information systems can be achieved "with the necessary culture, policies, and controls in place so as to avoid the introduction of conflict that could jeopardise the integrity and stability of the organisation as entity".[1] This statement allows the identification of three fundamental research questions:

1. What is the composition of an organisational culture that will encourage information system flexibility?
2. What policies should be introduced to enable flexibility yet maintain integrity?
3. How can management and IT departments control end-user development without inhibiting overall flexibility?

Like Web 2.0, a key enabling factor in TIS will be the community of end-users who share a common set of goals and operate in unison to achieve information system flexibility. Organisations need to develop an environment where such a community takes responsibility

[1] Kanellis, P., Paul, R.J. 2005. *Journal of Organisational and End User Computing*. p 85.

for their own information system by giving them a sense of ownership, enabled by authority over information technology.

Bibliography

ANDERSEN, K. V., VENDELO, M. T. 2004. *The Past and Future of Information Systems.* Oxford: Elsevier. ISBN: 0750661410.

APPLEGATE, L.M., MCFARLAN, F.W., MCKENNEY, J.L. 1999. *Corporate Information Systems Management: Text and Cases.* 5th ed. Boston: McGraw-Hill. ISBN: 0072902833.

AVGEROU, C. 2001.The significance of context in information systems and organisational change. *Information Systems Journal,* 11: 43-63.

AVISON, D., FITZGERALD, G. 2006. *Information Systems Development: Methodologies, Techniques and Tools.* 4th ed. New York: McGraw-Hill. ISBN: 0077114175.

AVISON, D., FITZGERALD, G., POWELL, P. 2001. Refelctions on information systems practice, education and research: 10 years of the Information systems Journal. *Information Systems Journal,* 11: 3-22.

BASKERVILLE, R., PRIES-HEJE, J. 2004. Short cycle time systems development. *Information Systems Journal,* 14: 237-264.

BECK, K., BEEDLE, M., VAN BENNEKUM, A., COCKBURN, A., CUNNINGHAM, W., FOWLER, M., GRENNING, J., HIGHSMITH, J., HUNT, A., JEFFRIES, R., KERN, J., MARICK, B., MARTIN, R. C., MELLOR, S., SCHWABER, K., SUTHERLAND, J., THOMAS, D. 2001. *Manifesto for Agile Software Development.* [Online]. Available: http://agilemanifesto.org/ [5 October 2007]

BEYNON-DAVIES, P. 2002. *Information Systems: An Introduction to Informatics in Organisations*. New York: Palgrave. ISBN: 0333963903.

BEAUDRY, A., PINSONNEAULT, A. 2005. Understanding User Responses to Information Technology: A Coping Model for User Adaptation. *MIS Quarterly,* 29(3): 493-524.

BOEHM, B. 1988. A Spiral Model of Software Development and Enhancement. *Computer,* p 61-72, May 1988.

BOGGS, R.A. 2004. The SDLC and Six Sigma: An Essay on Which is Which and Why? *Issues in Information Systems*, 5(1): 36-42.

BRADLEY, R.V., PRIDMORE, J.L., BYRD, T.A. 2006. Information Systems Success in the Context of Different Corporate Cultural Types: An Empirical Investigation. *Journal of Management Information Systems,* 23(2) 2006. 267-294.

BROWN, C.V., MAGILL, S.L. 1994. Alignment of the IS Functions with the Enterprise: Toward a Model of Antecedents. *MIS Quarterly,* 18(4) 371-403.

CARVALHO, J.A. 2000. Information system, Which One do You Mean? *Proceedings of the 1999 Conference on Information Systems Concepts,* Leiden, 20-22 September 1999, 259-280.

CHECKLAND, P., HOLWELL, S. 1998. *Information, Systems and Information Systems – making sense of the field*. Chichester: Wiley. ISBN: 0471958204.

DUANE, A., FINNEGAN, P. 2003. Empowerment and control in an intranet environment. *Information Systems Journal,* 13: 133-158.

COMER, D. E. 2001. *Computer Networks and Internets*. 3rd ed. New Jersey: Prentice Hall. ISBN 0130935816.

EVANS, A., MARTIN, K., POATSY, M. 2006. *Technology in Action.* 3rd ed. New Jersey: Pearson. ISBN: 0131878824.

FITZGERALD, B. 1996. Formalised Systems Development Methodologies: A Critical Perspective. *Information Systems Journal,* 6(1) 1996. 3-24.

FLOOD, R.L., JACKSON, M.C. 1991. *Creative Problem Solving: Total Systems Intervention*. New York: Wiley. ISBN: 0471930520.

FLYNN, D.J., JAZI, M.D. 1998. Constructing user requirements: a social process for a social context. *Information systems Journal,* 8: 53-83.

FRUHLING, A., DE VREEDE, G. 2006. Field Experiences with eXtreme Programming: Developing an Emergency Response System. *Journal of Management Information Systems,* 22(4) 2006. 39-68.

GALLIVAN, M.J., KEIL, M. 2003. The user-developer communication process: a critical study. *Information Systems Journal,* 13: 37-68.

GOULIELMOS, M. 2004. Systems development approach: transcending methodology. *Information Systems Journal,* 14: 363-386.

GUPTA, U.G. 1999. *Information Systems Success in the 21st Century*. New Jersey: Prentice Hall.

GWILLIM, D., DOVEY, K., WIEDER, B. 2005. The politics of post-implementation reviews. *Information Systems Journal,* 15: 307-319.

HAAG, S., CUMMINGS, M., MCCUBBREY, D.J. 2005. *Management Information Systems for the Information Age*. 5th ed. New York: McGraw-Hill. ISBN: 0072935863.

HALL, A.D., FAGEN, R.E. 1956. Definition of System. *Yearbook of the Society for the Advancement of General Systems Theory,* 1: 18-28.

HESS, D.S. 2001. *UML: An Introduction*. Gartner Research.

IGBARIA, M., ZINATELLI, N., CRAGG, P., CAVAYE, A. Personal computing acceptance factors in small firms: a structural equation model. *MIS Quarterly*, 18(4): 279–302.

IIVARI, J., HIRSCHHEIM, R., KLEIN, H.K. 2004. Towards a distinctive body of knowledge for Information Systems experts: coding ISD process knowledge in two IS journals. *Information Systems Journal*, 14: 313-342.

JACKSON, M.C. 2003. *Systems Thinking: Creative Holism for Managers*. Chichester: Wiley. ISBN: 0470845228.

JIANG, J.J., KLEIN, G., DISCENZA, R. 2001. Information System Success as Impacted by Risks and Development Strategies. *IEEE Transactions on Engineering Management,* 48(1):46-55.

KANELLIS, P., LYCETT, M., PAUL, R.J. 2000. Failure, Identity Loss and Living Information Systems. *Proceedings of the International Conference on Information System Concepts*, Leiden, 20-22 September 1999, 189-212.

KANELLIS, P., PAUL, R.J. 2005. User Behaving Badly: Phenomena and Paradoxes from an Investigation into Information Systems Misfit. *Journal of Organisational and End User Computing*, 17(2): 64-91, April-June 2005.

KIELY, G., FITZGERALD, B. 2005. An Investigation of the Use of Methodologies within Information Systems Development Projects. *The Electronic Journal on Information Systems in Developing Countries,* 22(4): 1-13.

KING, J.L., LYYTENIN, K. 2004. Reach and Grasp. *MIS Quarterly,* 28(4): 539-551.

KUMAR, R. L. 2004. A Framework for Assessing the Business Value of Information Technology Infrastructures. *Journal of Management Information Systems,* 21(2) 2004. 11-32.

KÜPPERS, G. 1999. Coping with Uncertainty: New Forms of Knowledge Production. *AI & Society Journal,* 13: 52-62.

KWON, D., OH, W., JEON, S. 2007. Broken Ties: The Impact of Organisational Restructuring on the Stability of Information-Processing Networks. *Journal of Management Information Systems,* 24(1) 2007. 201-231.

LAUDON, K.C., LAUDON, J.P. 2004. *Management Information Systems*. 8th ed. New Jersey: Pearson Education. ISBN: 0131206818.

LI, P., MAO, Y., ZDANCEWIC, S. 2003. Information integrity policies. *Proceedings of the 2003 Workshop on Formal Aspects in Security & Trust (FAST)*, Pisa, 8 September 2003.

LILIENFELD, R. 1978. *The Rise of Systems Theory. An Ideological Analysis.* Toronto: Wiley. ISBN: 0471535338.

LUCAS, H.C., 1994. *Information Systems Concepts for Management*. 5th ed. San Francisco: McGraw-Hill. ISBN 0071134557.

MAGALHÀES, R. 2004. *Organisational Knowledge and Technology. An Action-Orientated Perspective on Organisation and Information Systems.* Massachusetts: Edward Elgar. ISBN: 184376282.

MAGUIRE, S. 2000. Towards a business-led approach to information systems development. *Information Management & Computer Security,* 8(5): 230-238.

MANTON, S. 2005. *Integrated Intellectual Asset Management*. Aldershot: Gower. ISBN: 0566087219.

MATURANA, H.R., VARELA, F.J. 1972. *Autopoiesis and Cognition, The Realisation of the Living*. Dordrecht: Reidel Publishing Company. ISBN: 9027710163.

MCMANUS, D.J. 2003. A Model of Organisational Innovation: Build versus Buy in the Decision Stage. *The International Journal of Applied Management and Technology*, 1(1): 29-44.

MEYER, C., DAVIS, S. 2003. *It's Alive – The coming convergence of information biology and business*. New York: Crown. ISBN: 1400046416.

MILLER, J.G. 1978. *Living Systems.* New York: McGraw-Hill. ISBN: 0070420157.

MORGAN, G. 1997. *Images of Organisation*. 2nd ed. Calif: Sage. ISBN: 0761906320.

MUMFORD, E. 2003. Information Systems Research and the Quest for Certainty. *Journal of the Association for Information Systems*, 4(4): 197-205.

MYERS, M.D., AVISON, D. 2002. *Qualitative Research in Information Systems, A Reader.* London: Sage. ISBN: 0761966323.

NICHOLAS, J.M. 2001. *Project Management for Business and Technology. Principles and Practice*. 2nd ed. New Jersey: Prentice Hall. ISBN 0130183288.

OLIFER, N., OLIFER, V. 2005. *Computer Networks: Principles, Technologies and Protocols for Network Design*. Wiley: Chichester. ISBN: 0470869828.

OZ, E. 2004. *Management Information Systems.* 4th ed. Boston: Thompson. ISBN: 0619213221.

PATEL, N. V., PAUL, R. J. 1998. Towards systems tailorability: Comparative analysis of business change and systems flexibility. *Proceedings of the Association for Information Systems 1998 Americas Conference,* August 14-16, Baltimore. 122-124.

PATEL, N.V. 1999. Developing Tailorable Information Systems through Deferred System's Design. *Proceedings of the 1999 Americas Conference on Information Systems*, Milwaukee, 13-15 August, 1999, 4-6.

PATEL, N.V. 1999. The Spiral of Change Model for Coping with Changing and Ongoing Requirements. *Journal for Requirements Engineering*, 4: 77-84.

PATEL, N.V., IRANI, Z. 1999. Evaluating information technology in dynamic environments: a focus on tailorable information systems. *Logistics Information Management,* 12(1/2): 32-39.

PAUL, R. J. 1993. Dead Paradigms for Living Systems. *Proceedings of the First European Conference on Information Systems,* Henley-on-Thames 29-30 March 1993. The Operational Research Society, Birmingham. 250-255.

PAUL, R. J. 1994. Why users cannot get what they want. *International Journal of Manufacturing System Design*, 1/4. 389-394.

PEARLSON, K.E. 2001. *Managing and Using Information Systems: A Strategic Approach*. New York: Wiley. ISBN: 0471320013.

PEARLSON, K.E., Saunders, C.S. 2004. *Managing and Using Information Systems, A Strategic Approach*. 2nd ed. New York: Wiley. ISBN: 047134644.

RAVICHANDRAN, T., LERTWONGSATIEN, C. 2005 Effect of Information System Resources and Capabilities on Firm Performance: A Resource-Based Perspective. *Journal of Management Information Systems,* 21(4) 2005. 237-276.

REMYENI, D., SHERWOOD-SMITH, M. 1997. *Achieving Maximum Value from Information Systems*. Chichester: Wiley. ISBN: 0471975001.

SANTHANAM, R., SELIGMAN, L., KANG, D. 2007. Postimplementation Knowledge Transfers to Users and Information Technology Professionals. *Journal of Management Information Systems,* 24(1) 2007. 171-199.

SATZINGER, J.W., JACKSON, R.B., BURD, S.D. 2004. *Systems Analysis and Design in a Changing World*. 3rd ed. Boston: Thomson. ISBN: 061921371X.

SCHUTTE, F.G. 2000. *Integrated Management Systems*. 2nd ed. Sandown: Heinemann. ISBN: 1868539350.

SCHWALL, J. 2003. The Wiki Phenomenon. [Online]. Available: http://www.schwall.de/dl/20030828_the_wiki_way.pdf [25 October 2006]

SERAFEIMIDIS, V., SMITHSON, S. 2003. Information systems evaluation as an organisational institution – experience from a case study. *Information Systems Journal,* 13: 251-274.

SETHI, R. 1996. *Programming Languages. Concepts and Constructs*. 2nd ed. Addison-Wesley. ISBN: 0201590654.

SHARMA, R., YETTON, P. 2007. The Contingent Effects of Training, Technical Complexity, and Task Interdependence on Successful Information Systems Implementation. *MIS Quarterly,* 31(2): 219-238.

SIMON, J. C., KAISER, K. M. BEATH, C., GOLES, T., GALLAGHER, K. 2007. Information Technology Workforce Skills: Does Size Matter? *Information Systems Management*, 24(4): 345-359.

SPILLNER, A. 2002. *The W-Model - Strengthening the Bond Between Development and Test*. Software Quality Research Laboratory Seminar 2002, Ontario.

STAIR, R., REYNOLDS, G. 2006. *Principles of Information Systems*. 7th ed. Boston: Thompson. ISBN: 0619215615.

STAMOULIS, D.S., KANELLIS, P., MARTAKOS, D.I. 2001. Tailorable Information Systems: Resolving the Deadlock of Changing User Requirements. *Journal of Applied System Studies*, 2(2) 2001. 294-311.

STAMOULIS, D.S., MARTAKOS, D.I., INTRONA, L.D. 1998. Systems for Users, not for Observers: The Tailorability Concept. *Proceedings of the 1999 European Conference of Information Systems*, Copenhagen, 23-25 June, 1998, 1011-1024.

STAMOULIS, D.S., PATEL, N.V., MARTAKOS, D.I. 1996. A Systems Architecture Model and Implementation Platforms for Tailorable Information Systems. *Proceedings of the 1996 European Conference on Information Systems*, Lisbon, 2-4 July 1996, 314-322.

THIADENS, T. 2005. *Manage IT! Organising IT demand and IT supply*. Dordrecht: Springer. ISBN: 101402036396.

TURBAN, E., ARONSON, J.E. 2001. *Decision Support Systems and Intelligent Systems*. 6th ed. New Jearsey: Prentice-Hall. ISBN: 0130894656.

TURBAN, E., MCLEAN, E., WETHERBE, J. 2003. *Information Technology for Management*. 4th ed. New York: Wiley. ISBN: 0471229679.

WARD, J., PEPPARD, J. 2003. *Strategic Planning in Information systems.* 3rd ed. New York: Wiley. ISBN: 0470841478.

WEICK, K.E., SUTCLIFFE, K.M. 2001. *Managing the Unexpected*. New York: Wiley. ISBN: 0787956279.

WHITELEY, D. 2004. *Introduction to Information Systems, Organisations, Applications, Technology and Design.* Basingstoke: Palgrave Macmillan. ISBN: 0333997662.

WIKIPEDIA. 2006. *Wikipedia.org* [online]. Available: http://en.wikipedia.org [25 October 2006]

WILLCOCKS, L. 1996. *Investing in Information Systems, Evaluation and Management.* 1st ed. London: Chapman & Hall. ISBN: 041272670X.

XIA, W., LEE, G. 2005. Complexity of Information systems Development Projects: Conceptualisation and Measurement Development. *Journal of Management Information Systems,* 22(1) 2005. 45-83.