Web Information Systems: A Study of Maintenance, Change and Flexibility

A Thesis Submitted for the Degree of Doctor of Philosophy

Jason Christian Peters

School of Information Systems, Computing and Mathematics

Brunel University
London ▪ United Kingdom

March 2010

Do not forsake wisdom, and she will protect you; love her, and she will watch over you. Wisdom is supreme; therefore get wisdom. Though it cost all you have, get understanding.

(Proverbs 4: 6-7, Holy Bible)

Abstract

Information Systems (IS's) have provided organisations with huge efficiency gains and benefits over the years; however an outstanding problem that is yet to be successfully tackled is that of the troublesome maintenance phase. Consuming vast resources and thwarting business progression in a competitive global market place, system maintenance has been recognised as one of the key areas where IS is failing organisations. Organisations are too often faced with the dilemma of either replacement or the continual upkeep of an unwieldy system. The ability for IS's to be able to adapt to exogenous influences is even more acute today than at any time in the past. This is due to IS's namely, Web Information Systems (WIS's) increasingly and continually having to accommodate the needs of organisations to interconnect with a plethora of additional systems as well as supporting evolving business models. The richness of the interconnectivity, functionalities and services WIS's now offer are shaping social, cultural and economic behaviour on a truly global scale, making the maintenance of such systems and evermore pertinent issue. The growth and proliferation of WIS's shows no sign of abating which leads to the conclusion that what some have termed as the 'maintenance iceberg' should not be ignored.

The quandary that commercial organisations face is typically driven by two key aspects; firstly, systems are built on the cultural premise of using fixed requirements, with not enough thought or attention being paid to systems abilities to deviate from these requirements. Secondly, systems do not generally cope well with adapting to unpredictable change arising from outside of the organisations environment. Over the recent past, different paradigms, approaches and methods have attempted to make software development more predictable, controllable and adaptable, however, the benefits of such measures in relation to the maintenance dilemma have been limited. The concept of flexible systems that are able to cope with such change in an efficient manner is currently an objective that few can claim to have realised successfully.

The primary focus of the thesis was to examine WIS post-development change in order to empirically substantiate and understand the nature of the maintenance phase. This was done with the intention to determine exactly 'where' and 'how' flexibility could be targeted to address these changes. This study uses an emergent analytical approach to identify and catalogue the nature of change occurring within WIS maintenance. However, the research framework design underwent a significant revision as the initial results indicated that a greater emphasis and refocus was required to achieve the research objective. To study WIS's in an appropriate and detailed context, a single case study was conducted in a web development software house. In total the case study approach was used to collect empirical evidence from four projects that investigated post-development change requests in order to identify areas of the system susceptible to change. The maintenance phases of three WIS projects were considered in-depth, resulting in the collection of over four hundred change requests. The fourth project served as a validation case. The results are presented and the findings are used to identify key trends and characteristics that depict WIS maintenance change. The analytical information derived from the change requests is consolidated and shown diagrammatically for the key areas of change using profile models developed in this thesis. Based on the results, the thesis concludes and contributes to the ongoing debate that there is a discernable difference when considering WIS maintenance change compared to that of traditional IS maintenance. The detailed characteristics displayed in the profile models are then used to map specific flexibility criteria that ultimately are required to facilitate change. This is achieved using the Flexibility Matrix of Change (FMoC) tool which was developed within the remit of this research. This tool is a qualitative measurement scheme that aligns WIS maintenance changes to a reciprocal flexibility attribute. Thus, the wider aim of this thesis is to also expand the awareness of flexibility and its importance as a key component of the WIS lifecycle.

Acknowledgements

It is often very difficult to articulate the efforts of all those involved at some stage throughout the entire research experience. Therefore I will take this opportunity upfront to thank all those who have contributed to this research in some shape or form. However, to name but a few I would like to acknowledge the following:

First and foremost I want to praise and thank my heavenly father for giving me the strength and courage in pursuing as well as concluding this major milestone. It is through his will and watchfulness that I have achieved everything I have set out to do in life and I pray his favour remains with me forevermore.

I am also sincerely grateful to Professor Guy Fitzgerald my supervisor for giving me the opportunity and guidance in achieving this research. He is the inspiration behind me experiencing a career as a researcher and his constant support has been invaluable in the completion of this thesis.

Needless to say, my family and friends played a major role in supporting and encouraging my growth throughout the research process. Therefore I especially extend my gratitude to the following individuals. My parents, Victor and Jane for their unconditional love that helped me reach this end. To my brother Sean and sisters Hazel and Michelle whose combined efforts have sustained me through any difficult and stressful times. To Eric Vithal for his continuous prayers and support whenever I needed it most.

Lastly, I would like to thank my colleague's for sharing their opinions and all those at my case study site that provided me with an insight into their working activities. In particular my appreciation goes to Howard Harris, Sungmin Park and Bruno Bose.

In loving memory of my grandmothers, Alice Vithal and Daisy Peters.

Table of Contents

List of Appendices

List of Figures

List of Tables

Part I
The Problem Area

Part I of this thesis seeks to elaborate on the issues and context of the problem area. Chapter 1 provides an overview of the thesis with a synopsis of each of the subsequent chapters. Attention is paid to introducing the problem domain of WIS's and the issue of maintenance facilitation. The chapter proceeds by defining the research aims, objectives and specific questions that the researcher will address during the course of the investigation. Research design, approach and its operation are then outlined along with the research contributions. Lastly, a diagrammatical representation of the logical research progression coupled with the thesis structure is presented.

Chapter 2 consolidates and collates the existing literature on the three broad topics of Web Information Systems, maintenance and Information Systems flexibility. This chapter provides a comprehensive and rounded academic perspective on these topics that pertain to the research aims, objectives and questions. These perspectives form the necessary backdrop required to understand the procedural and analytical workings of the thesis.

Chapter 1 ▪ Overview

1.1 Introduction

This thesis investigates the phenomena of managing and implementing Web Information Systems (WIS's) change efficiently during the maintenance phase. The issue of organisations' having to respond to often unpredictable and turbulent environments is a recognised concern within the IS domain. The fluctuating business climate and consequential organisational adjustments can cause unprecedented effects to IS business activities and processes. Typically, emergent issues such as business expansion, deregulation, re-regulation, competition, acquisitions, globalisation and brand adaptation are factors that drive constant evolution. The needs of organisations to cope with such on-going change should be reflected and synchronised with an equally malleable IT (Information Technology) structure. New requirements derived from internal or external factors undoubtedly have an effect on the fabric and operational procedures within the organisation's IT/ IS. It is recognised that the adoption of technologies can benefit and enable organisations in a changing climate. However, the commonly perceived viewpoint is that IT can often present itself as a disabler to organisational change due to its relative unresponsiveness in accommodating the necessary change. The panacea of IS's that are adaptable and can support the dynamics of a changing environment proactively and efficiently is yet to be achieved. The relevance of twentieth century software engineering techniques should be questioned in-situ of the dynamic requirements to support the 'living' IS's of today (Paul, 1993; Paul, 1994; Grisogono, 2007). Some authors have suggested that the disparity between organisational needs and what IT/ IS can deliver has resulted in a ‘software crisis’ (Boogaard, 1994; Veldwijk et al., 1994; Layzell, 2001; Livschitz, 2005; Yu, 2009). Boogaard’s (1994) review of the software crisis states that a dominant cause of disappointing systems is primarily attributed to the maintenance activities associated within software development. The work of this thesis is to investigate to what extent Boogaard’s supposition’s apply to a WIS’s context as well as to propose flexibility as a solution to alleviate the ‘software crisis’. A comprehensive understanding of the WIS maintenance phenomenon and the incorporation of the proposed solution will contribute to the development of systems that are easier to sustain and allow for maintenance change to be implemented more efficiently.

This chapter continues to explore the specific issue of maintenance and the adaptability of IS’s to manage continual change within a WIS’s context (Section 1.2). This issue provides the necessary backdrop to the research, resulting in the aim and objective of the thesis (Section 1.3). Section 1.3 also provides the research questions posed within this thesis derived from the problem definition. The means by which the aim and objectives are met are detailed and the relevance of the research design and approach is outlined in Section 1.4. Contributions of the thesis are stated in Section 1.5 and the structure and content of each chapter is summarised (Section 1.6). Lastly, the content and relationships of each chapter of the thesis is consolidated diagrammatically.

1.2 The Problem Context

The problem domain of this thesis is set within a WIS context. The web context cannot be understated enough, as this medium is changing social, cultural and commercial behaviour on a globally unprecedented scale. Maintenance for WIS’s is one facet of the web that is often neglected and underrated as an area of consideration. The reported extent, depth and nature of the maintenance concern specifically for WIS's is not yet as abundant when compared to traditional IS's. The maintenance issue could be even more accentuated and pronounced in the twenty-first century of web orientated technologies. The maintenance problems in the web era are at least as significant as those that have existed in the past, thus making it an ever more pertinent issue to discuss in the present. This thesis recognises the issues surrounding the similarities and differences when comparing traditional IS's to WIS's. Essentially, the academic debate can be divided into two sides, one side advocating traditional software engineering techniques for all software systems. The other supporting evolutionary concepts and processes specifically designed for WIS's or adaptive type systems. A resulting paradigm shift specifically for WIS's has large implications when considering its design, development and management processes. By understanding its nature, methods for maintaining WIS's can be endorsed to facilitate the change process.

The modification activities of software systems are perhaps one of the most important concerns within the development and evolution of IS's. Since the early 1970's, there has been a growing interest among practitioners and academics to develop alternative ways of delivering, developing, controlling, managing and maintaining software systems to cope with the fluctuating demands of the organisation. Faced with change, organisations are posed with two options, either replacement or enhancement through maintenance procedures. Traditionally, catering for change has been mostly managed in the maintenance phase of the Software Development Lifecycle (SDLC) (Avison and Fitzgerald, 2003).

The 'maintenance' phase of the SDLC has been recognised as a critical and important activity and represents the general name given to a set of post-development activities that relate to the continuous modus-operandi of the IS. The IEEE Standard (1998) defines software maintenance as 'the modification of a software product after delivery to correct faults, to improve performance or other attributes, or to adapt the product to a modified environment'. The established perspective on the maintenance activity is based on the pioneering research of Canning (1972), Swanson (1976), Lientz et al. (1978) and Swanson and Beath

(1989). Typically organisations make new demands that include long-term adaptation and the enhancement of the IS to meet new and evolving requirements. This is in addition to modifying the software to run on new or upgraded platforms and to integrate to an increasing array of disparate systems. All of which leads the researcher to the conclusion that software systems are not static in nature but are changing and continuously having to be evolved over time.

The concerns and issues relating to the maintenance phase can be thought of by using the analogy of an ‘iceberg’, indicating that a large proportion of the problem remains unseen (Canning, 1972; Martin and McClure, 1983). The maintenance burden becomes apparent when reviewing the numerous empirical surveys that have been conducted and reported on in the literature (Lientz et al., 1978; Lientz and Swanson, 1980; Guimaraes, 1983; Boogaard, 1994; Yourdon, 1997; Dishaw and Strong, 1998; Palmer, 1999; Schach and Tomer, 2000). They all purport that the maintenance phase accounts for a disproportionate amount of time and effort compared with the entire SDLC cost. Thus, from an organisational perspective the maintenance activity essentially presents itself as a problem due to the excessive amount of time and resources dedicated to it. Consequently, expenditure upon maintenance demands a large commitment to sustain base functionality as well as incorporating enhancements. These issues can cause IS maintenance to become a disabling factor for organisations.

More specifically, problems organisations encounter during maintenance are:

- IS’s can be unstructured or become unstructured through years of modifications or up-grades. Resulting in existing IS’s being difficult to understand.
- Documentation of IS maintenance changes is often insufficient.
- Personnel involved in maintenance tend to have a larger degree of turn-over as the work is perceived to be mundane. This results in the key skills to perform effective maintenance being lost to the organisation.

- The lack of trialability during maintenance results in changes being applied where the consequences are not thoroughly understood, resulting in regression and an increased unpredictability to the system.
- Processes and mechanisms are less rigorous when applying changes to the IS. The controllable SDLC principles applied to the development phase are largely ignored within maintenance.
- Maintenance costs are disproportionately high when compared to implementation costs and often exceed initial estimates.

IS adaptations are made in response to on-going requests for change, which can occur pre or post-development. Despite the many different system development techniques and methods that have evolved to support the management of complexity in the design process (Avison and Fitzgerald, 1995; Avison and Fitzgerald, 2003); the rate of 'change' demanded has outpaced the attempt to make improvements to the system's design. The problems associated with system complexity generally increase during the maintenance phase of the system lifecycle due to the reasons listed above. The current attempts to handle change can be broadly thought of as those which are proactive and those being reactive. Reactive responses to change assume that maintenance change will occur and focus on increasing the efficiency of how these changes are dealt with once they have arisen. The Information Technology Infrastructure Library (ITIL) change management processes is one example of how the IS industry uses mechanisms to manage changes that occur. Proactive approaches are focused on solutions that purport to increase an inherent capability to cope with future changes, object orientation in programming languages can be considered as one example of this.

Flexibility as a concept has also been championed as a proactive approach to facilitate future changes. In essence the flexibility concept proposes that inherent structures and design features should be implemented within the software system design, such that if future changes do occur then they can be facilitated more efficiently. Thus, the flexibility solution can be perceived as a preventative mechanism to accommodate change. The wider rationale of this thesis is to expand the concepts of flexibility within the WIS's realm.

1.3 The Research Questions: Aims and Objectives

The maintenance phase is subject to continual change as requirements on the system evolve and adapt according to business needs. It is assumed that the dynamic nature of change is unpredictable within the maintenance phase, therefore activities associated with maintenance are often undeterminable and difficult to pre-empt. As a consequence of this, the research begins with the notion of better understanding the impact and nature of maintenance change in a WIS context. IS development methods and techniques have evolved to become more agile and responsive to change, however the implementation of maintenance changes still appears to be inefficient and the overheads disproportionately high. It is proposed that through incorporating flexibility in the IS design, maintenance change can be addressed in a more proactive and efficient manner.

The principal aim of this thesis is to examine how maintenance change in a WIS context can be better facilitated. The rationale driving this aim is to reduce the effort and complexity associated with maintenance activities. In the web era it is extremely important that systems have the ability to be adaptable and responsive in the face of an ever-changing climate. Understanding the nature of change is a prerequisite step in order to isolate 'where' and 'how' flexibility can be applied to facilitate change. The implementation of flexibility processes and mechanisms need to be considerations that are fundamental to the design of the system. In order to achieve this, specific questions such as 'where does flexibility need to be applied?' and 'what type of flexibility is required?' must be addressed.

An empirical means to substantiate findings posed by the research questions have not been achieved to the best of the researchers knowledge. Therefore the objective of this thesis is segregated into two broad questions with the intent to investigate the aims discussed above.

1. **What is the nature of change in WIS maintenance?**

    A case study approach is adopted to examine and collate change request data, for the purposes of empirically quantifying maintenance change. Based on this analytical data, profile models for the most significant areas of change are created and used as the input for research question two. This question is further divided into two sub-research questions:

    i. **What are the areas of the software system most susceptible to change?**

        The focus of the question is to identify and isolate areas of the WIS that are impacted by change during the maintenance phase. Through an emergent analytical technique, change requests are classified allowing for specific WIS areas of change to be identified and categorised.

    ii. **What are the characteristics of change?**

        The characteristics of WIS maintenance change are analysed from the perspective of 'Form', 'Complexity' and 'Context'. The occurrence and interrelationships of these constructs are used to gain an in-depth and comprehensive understanding of the nature of change.

2. **How can WIS maintenance change be better facilitated?**

   By recognising potential areas of the system requiring change, the suggested solution of flexibility can be focused and tailored to the particular needs of the WIS.

   i. **How can flexibility facilitate WIS maintenance changes?**

      A conceptual qualitative framework (Flexibility Matrix of Change - FMoC) for the identified change categories is developed to identify which dimensions of flexibility would be required to facilitate maintenance change.

1.4 Research Approach

In order to handle change effectively requires an understanding of the changes that occur. The research starts with a given theoretical perspective of existing areas of change categories that have been inferred from literature. The existing model that was selected was proposed by Fitzgerald et al. (1999) that provided suitable IS 'areas of change' (AoC) categories based upon synthesis of existing knowledge.

After utilising this AoC model in the initial research framework, the findings indicated that the framework was not entirely suitable, as it became evident that more granularity regarding the areas of change could be attained. Further investigation and a newly designed research framework was necessary to investigate WIS maintenance changes. The new framework used a case study approach with embedded units of analysis as well as pattern-matching and clustering techniques to allow the emergence of WIS areas of change group categories. The incorporation of the new research framework prompted the additional sub-question of, 'What are the characteristics of change?' to be added to research question one.

1.5 Contributions

The contributions of this research can be listed in five themes that relate to the theoretical and practical implications on WIS maintenance.

- A research method to categorise change is developed based on the pattern-matching and clustering principles of Miles and Huberman (1994). Specifically, a bottom-up process is designed within this research, that allowed for the emergence of WIS AoC categories. A bespoke categorisation workflow process is also created for the purposes of validation. This research method offers a new and different perspective to analyse change. Thus, the research framework and analytical procedures are set-up such that the process can be replicated. This technique can be applied to various contexts other than this thesis to categorise and analyse change.

- Detailed analysis that depicts the nature of WIS maintenance change based on empirical evidence is provided by this thesis. A comprehensive analysis of the characteristics of WIS maintenance change contributes to knowledge in this embryonic area of research.

- A distinction is made between IS and WIS Maintenance change. This thesis supports the view with empirical evidence that the maintenance change characteristics are different for both IS and WIS.

- Profile models of change group categories is specifically developed within the scope of this research to graphically represent and consolidate the vast analytical data collected. The profiling model technique is context and granularity independent and can be used in various other situations other than that of this thesis.

- A qualitative flexibility measurement framework (FMoC) is devised by the researcher which utilised existing flexibility dimensions prescribed by Knoll and Javenpaa (1994). The FMoC tool can be used to manage, control and plan for flexibility within the SDLC.

1.6 Thesis Outline

The remainder of this section provides an overview of the content of each chapter within this thesis:

- Chapter 2 ▪ The Maintenance Quandary: A Literature Review

  This chapter sets the context of the research with an explanation and definition of the problem as well as its proceeding issues. This provides the basis of a common vocabulary which is used throughout the remainder of the thesis. A literature based analysis of the WIS domain is given along with the traditional perspective of IS maintenance, its problems, issues and methods. The concept of flexibility is introduced as a solution to facilitate the handling of change more efficiently in order to reduce the maintenance overheads. The chapter uses existing literature to supplement and build upon issues that have been raised within the thesis.

- Chapter 3 ▪ Research Design: Building a Research Framework to Investigate Maintenance change in Web Information Systems

  This chapter discusses the available tools, techniques and approaches that are obtainable to elicit and elucidate data/ information for the research. The requirements for a suitable case study that enables the objectives of this research to be fulfilled are addressed. The design of two research frameworks is detailed within the chapter that incorporate different perspectives to provide cogent evidence in addressing the research issues and questions. An explanation of the change in research direction and unsuitability of Research Framework A is briefly discussed. It is from the design of Research Framework

B, that change data is collected and analysed from embedded units of analysis within the case study approach. Research Framework B utilises an emergent and bottom-up approach to category building, as well as using pattern-matching and clustering techniques.

- Chapter 4 ▪ Operationalisation of the Research Framework

  The aim of this chapter is to report on the operationlisation of the research frameworks. The chapter begins by describing the case study which provided the data for both research frameworks. A description is given as to why Research Framework A was deemed inappropriate and the transition to Research Framework B is fully described. The process to identify the impacted areas of change using the design stated in Chapter 3 is detailed using the categorisation workflow process. The chapter concludes with stating that 425 individual maintenance change requests were analysed resulting in the crystallisation and defining of twelve areas of change group categories within the WIS maintenance phase. This enabled the research to answer the sub-research question of *'What are the areas of the software system susceptible to change?'*.

- Chapter 5 ▪ Results and Analysis of Web Information Systems Maintenance Change

  This chapter provides an analytical examination of the dataset repository created from the operationlisation of Research Framework B. The chapter presents quantitative results that are extrapolated from the dataset to substantiate and answer the sub-research question of *'What are the characteristics of change?'*. The results produced are further evaluated, in order to provide a comprehensive answer to research question one, that of *'What is the nature of change in WIS maintenance?'*. The top two emerged group categories of TUI Content and Functions are explored in detail and key findings and results are stated.

- Chapter 6 ▪ Maintenance Change Facilitation: Designing a Conceptual Flexibility Framework

  The chapter begins by further answering the first question of *'What is the nature of change in WIS maintenance?'* by the use of profile models of change. The models themselves, their use and implications are also described in detail. Following on from the profile models, the formulation of a qualitative flexibility measurement framework (FMoC) and its use is described. The key objective of the FMoC framework is to answer the second research question, that of *'How can WIS maintenance change be better facilitated?'.* The chapter concludes by providing implications and inferences regarding flexibility based on the findings of the chapter.

- Chapter 7 ▪ Conclusions and Implications

  The objective of this chapter is to provide a synopsis of the research and to summarise its findings. The chapter is comprised of two broad sections, the first section summarising each chapter of the thesis and its findings relating to the research questions. The second section discusses the implications, contributions and the further research that emanate from this thesis. The chapter then progresses to address wider implications and subsequently leads to broader discussions on the topics of maintenance, the software development lifecycle and the concept of flexibility. Research contributions are also listed into five main themes that collectively summate the contribution of the thesis. Finally, the further research section suggests four potential streams of work that can be conducted to supplement and validate the conclusions and findings of the thesis.

The below diagram is a representation of the outline of the thesis (Figure 1.1).

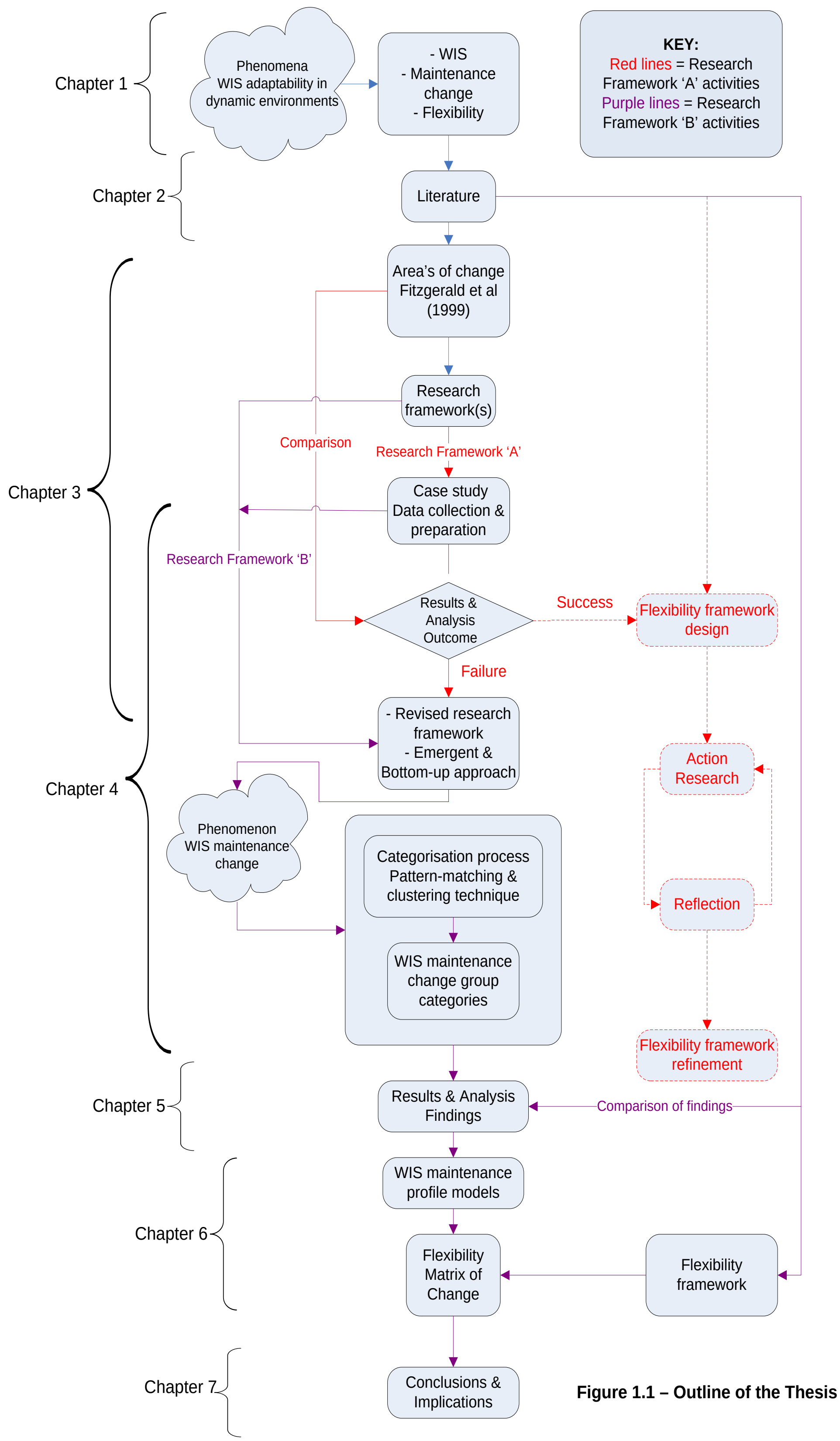

Figure 1.1 – Outline of the Thesis

1.7 Conclusion

This chapter provides an introduction and overview to the thesis and how it is constructed. It begins by familiarising the reader with the context and the issues surrounding the problem domain. Two research questions are derived from the problem domain, that of ‘What is the nature of change in WIS maintenance?’ and ‘How can WIS maintenance change be better facilitated?’. The chapter proceeds to outline the research approach used, the contributions the research makes and finally outlines the content of the remaining six chapters. The chapter concludes with a diagrammatic representation mapping the logical flow of the thesis to the various chapters.

The following chapter discusses and expands upon the issues initially located within this chapter and provides further explanations.

Chapter 2 ▪ The Maintenance Quandary: A Literature Review

2.1 Introduction

This chapter frames and positions the IS literature for three key domains, namely, the WIS's context, the maintenance problem and flexibility as a potential solution to facilitate this problem. The chapter analyses the views and opinions of various WIS academics (Bieber et al., 1998; Isakowitz et al., 1998; Russo and Graham, 1999; Pressman, 1998, 2001; Liu and Arnett, 2000; Murugesan et al., 2001; Deshpande et al., 2002; Ginige, 2002; Barry, 2004; Tonella and Ricca, 2004; Smith and Lewis, 2008) to ascertain the attributes of a WIS environment (Section 2.2) and specifically the similarities and differences with what is considered to be a traditional IS environment (Section 2.2.1). Development and management methodologies are studied to add further weight to the IS versus WIS debate as well as to establish maintenance considerations that can be derived from these methodologies (Section 2.2.2).

The chapter progresses by exploring the literature relating to the WIS maintenance phenomenon specifically (Section 2.2.3). These considerations are set against the established literature on IS maintenance and its characteristics (Sections 2.3 and 2.3.1). Within the scope of the IS maintenance study, the problems (Section 2.3.2), complexity (Section 2.3.3) and methods (Section 2.3.4) are clarified and presented.

One potential solution, that of flexibility to the maintenance phenomenon is developed in detail (Section 2.4). The perceptions and definitions of the current understanding of flexibility are investigated with various views presented (Section 2.4.1). The proposed solutions to flexibility including approaches, methodologies and dimension based frameworks are reviewed and discussed (Sections 2.4.2 and 2.4.3). This section ends by presenting a study of the barriers and constraints to the incorporation of flexibility (Section 2.4.4). A synopsis of the key findings and issues from the chapter are summarised in the conclusion (Section 2.5).

2.2 Web Information Systems

The economic climate of globalisation in the twenty first century has established the World Wide Web (WWW) as the prevailing way that people gain access to computer applications and information (Bieber et al., 1998). As a new generation of Internet-savvy consumers emerge, that are no longer solely computer professionals and enthusiasts (Sørensen et al., 2001), organisations are now prompted to develop innovative ways to solve business problems or redefine an existing business process that incorporates the Internet platform (Lang, 2001). Organisations are also utilising WIS's to respond to the increasing dynamic complexity in business to obtain competitive advantage (Lipnack and Stamps, 1997). Hamill (1997) asserts that if used properly, the Internet can be a powerful source of competitive advantage to organisations and further, that the addition of Internet-based strategies by senior management to support overall business development is increasing (Austin, 2000). This has in some cases shifted organisational perspectives from functioning as centralised structures to restructured decentralised environments, that are integrated into the global information society (Currie, 2000).

The Internet has enabled the integration, dissemination and communication of disparate information repositories to be connected globally in an effective and efficient way. Market pressures, financial benefits, enterprise mergers, outsourcing benefits, broader market presence, value-chained relationships, external environment changes and government based legislation changes are all factors that are driving organisations to reengineer business processes to be inclusive of Web-based applications as an information resource and portal to mainframe systems. Liu and Arnett (2000) further state in the context of electronic commerce that 'EC can help business organisations cut costs, interact directly with customers, run more smoothly and in a more timely manner, and even better, it can help an organisation outperform its competition.'. Web-based applications have emerged within almost every industry sector and are being widely deployed throughout commercial businesses, education, government and other institutions (Liu and Arnett, 2000). Initially there had been a tremendous increase in utilising the Web as a means to publishing documentation on-line. This represented a cultural migration from traditional distribution of paper-based documentation to publishing electronically on the Web (Berkman, 1997). Over a relatively short space of time, the web platform has evolved from a humble brochure-ware marketing tool to a platform that can sustain all facets of organisational work (Isakowitz et al., 1998).

Applications that use the Web platform are growing in complexity and have moved away from purely publishing static content (Lang, 2001). Tonella and Ricca (2004) recognise that static web sites are gradually being replaced by dynamic sites; where through the integration with sub-systems (for example, databases and middleware) non-trivial computation is now being performed and suggest that in order to improve/ maintain quality, evaluation tools will be required (Ricca and Tonella, 2006). There exists today several operational use categories of web applications (for example social networking - communication/ connectivity, search engines – information/ directories as well as procurement of products and business services). A new field called Web Information Systems (WIS's) has emerged as a consequence of organisations grafting '… Web interfaces to their legacy systems and conceive of entire new information systems that take advantage of the Web infrastructure' (Bieber et al., 1998; Smith and Lewis, 2008).

Scharl (2000) perceives WIS's as a sub-category of mass IS's that is complimentary to interactive television, kiosk systems, or automated teller machines. He further states that WIS's rely on the distributed hypertext functionality and transfer mechanisms of the WWW and are characterised by 'interactivity, dynamic updating, hypertextuality and global presence …'. With hindsight it can be seen that these rather limited expectations of WIS's have been greatly exceeded with the WIS's of today (Berners-Lee and Fischetti, 2008). Isakowitz et al. (1998) recognises in their definition the breadth and scope of WIS's. Their definition encompasses four dominant types of WIS's:

- Intranets – supportive to the internal work procedures of the business.
- Web-presence sites – marketing tools deigned to reach beyond the boundaries and scope of the business.
- Electronic commerce systems – to support consumer/ client interaction.
- Extranets – internal and external systems to support business-to-business communications.

A WIS is defined within this thesis as the conglomeration of interlinked, but semi-independent Web-based systems and applications managed by coordinated individuals and groups within an organisation and are developed for external (Internet and Extranet) and/ or internal (Intranet) parties. This perception of WIS's originates from the definition proposed by Eschenfelder and Sawyer (2001) that is inclusive of Internet and Extranet sites; however, in their explanation exclude Intranet sites and the coordinated actions required by individuals and groups to develop, manage and control WIS's.

The IDC report quoted in Lowe and Henderson-Sellers (2001, p.11) predicted almost a decade ago that US expenditures on WIS initiatives would grow from US $12 billion in 1999 to $43.6 billion in 2002. With this ever increasing growth in expenditures, technical or business failures of WIS's would be publicly visible and disastrous (Lowe and Henderson-Sellers, 2001). Ginige (2002) suggests that although organisations have successfully developed large complex WIS's, there are other cases that have failed or are confronted with the potential of major failings. WIS's failures are caused by '… a flawed design and development process, and poor management of development efforts' (Ginige, 2002). He further summarises the 'research-brief' findings by the Cutter Consortium (November 2000) on web-based application development. The findings from the research brief indicated that:

- 84% of delivered systems do not meet the needs of the business.
- 53% of delivered systems do not conform to the required level of functionality.
- 79% of projects experienced schedule delays.
- 63% of projects did not meet and surpassed the budget.

The high percentages indicated above from the survey study are indeed reminiscent and similar to the project management issues that have plagued the development of many traditional IS's. Large-scale IS's have often been reported as suffering from incomplete specifications, quality issues in relation to performance, budgetary issues and delivery within agreed timelines. The development and management activities associated with WIS's display familiar problematic attributes faced by traditional IS's.

2.2.1 WIS versus IS

There are various attributes that distinguish traditional IS's from WIS's. One of the key ones being that traditional IS's have typically been thought of as stand alone or individual systems. The trend towards an ever increasing interoperability (Maier, 1996; Maier, 1998; Maier, 2007), greater complexity (White, 2005; Cherinka et al., 2005) and scale (Northrop et al., 2006; Northrop, 2008) typify the WIS's of today.

Characteristics of large scale WIS which can be thought of as systems of systems include (Maier, 1996; Maier, 1998; Boardman and Sauser, 2007):

- There is operational independence of the elements.
- The elements are managed independently.
- The WIS under goes evolution and is never fully formed.

- The WIS displays emergent behaviour in which the system performs functions that do not reside in any component system.
- There is geographical dispersion of the WIS.

Vidgen (2002) identifies three key areas in which WIS's differs from traditional IS development:

- A strong and direct link with business strategy.
- The need to incorporate sales and marketing skills to address the needs of the user as customers.
- A bolstering of traditional IS development skills with a graphic design sensibility.

WIS's can be considered as being different from traditional IS's in terms of their purpose and audience for which they are developed. They may attract many different types of visitors and their requirements may be very different (Stroud, 1998). Furthermore, WIS's use of communication technology, multi-platform accessibility, their non-sequential nature (hypertext links) (Russo and Graham, 1999) distinguishes them from traditional IS's. Lang and Fitzgerald (2005) emphasise that a strong distinguishing aspect is when WIS's assume hypermedia functionality and thus become substantively different from a design perspective. They further state that 'Hypermedia technologies support much richer user interfaces, more complex navigation mechanisms, and more varied forms of information than traditional computer systems ...'.

Whether traditional IS's and WIS's should be thought of as similar or different is still however a contentious topic (Bieber et al., 1998; Pressman, 1998; Lowe and Henderson-Sellers, 2001; Vidgen, 2002; Barry and Lang, 2003; Barry, 2004; Lang and Fitzgerald, 2006). A conference was held amongst several academics to actively engage and debate on the subject of 'What's so different about the World Wide Web anyway?'. The members of the discussion, in particular, Conger and Janko, share their opinions in answering the question (Bieber et al., 1998):

> 'Web IS are different from traditional IS in very fundamental ways: analysis and design, content and purpose, users, and designers skill set.' – Sue Conger

> 'Technically, the difference seems to be small. The difference lies in intent.' – Wolfgang Janko

In another 'roundtable' debate chaired by Pressman, the participants were asked to focus the debate based on the fundamental tenants of IS engineering. The question 'Can Internet-Based Applications Be Engineered?' was proposed. The majority of perspectives offered by the participants had a strong bias towards the endorsement of an engineering approach to building WIS's. Nevertheless, an interesting and contrariness view was taken by one of the contributors, Lewis (Pressman, 1998):

> '"Old-fashioned software engineering" techniques that evolved out of the 1970s structured programming moment haven't worked, so why do we want to impose them on the new generation of Web designers? Today's Web programmers are trying to escape the old school by blending art with engineering. They know the problem is fuzzy – that is, unspecified; that their customers don't know how to specify their requirements; and that it doesn't matter anyhow, because as soon as the system is completed it will be obsolete! The waterfall model is dead, the spiral model is dying, and the rapid prototype is becoming the product. Darwinism has forced developers to adopt a free-wheeling approach to Web-based software development that makes traditional 'structured' software engineers roll over in their cubicles.' – Ted Lewis

Lewis acknowledges the failings of traditional engineering methods for IS development and clearly calls into question their applicability to WIS's development. In keeping with Lewis's perspective there is a school of thought that methodologies appropriate for WIS should be 'amethodical' and less reliant on rigid processes that have been the basis for traditional IS management and development (Baskerville et al., 1992).

2.2.2 Development Methods

Proponents who view WIS's as being different to traditional IS's have been active over the last decade in defining new development and management approaches. These include RMM, OOHDM, WebML, WISDM, HyDev, OO-HMethod, ERMIA and EORM (Gellersen et al., 1997; DeTroyer and Leune, 1998; Lowe, 1999; Ceri et al., 2000; Gómez et al., 2000; Gómez and Cachero, 2003). Collectively this field of enquiry has been termed as Web Engineering (Lowe, 1999; Ginige and Murugesan, 2001a; Ginige and Murugesan, 2001b; Murugesan et al., 2001; Alatalo, 2001; Ginige, 2002; Deshpande et al., 2002; Bahli and Tullio, 2003; Navarro et al., 2005). According to Murugesan and Desphande (2002), Web Engineering can be defined as the application of a systematic, disciplined, quantifiable approach to development, operation, and maintenance of Web-based applications or the application of engineering to Web-based software (Murugesan and Desphande, 1999; Murugesan and Desphande, 2000). These methods are based upon the principles and philosophical views of the IS engineering practices, however, these methods are still very much orientated to the development phase of the WIS with no 'special' consideration for the maintenance activities.

Some academics claim to have developed WIS methodologies further which seek to address WIS maintainability and extensibility (Ginige, 1998; Warren et al., 1999; Ginige and Murugesan, 2001c). Warren et al. (1999) states that the ultimate objective is to produce a predictive model, so that evolution of any given website can be categorised and taken under control, with consequent reductions in maintenance effort and improvements in product quality. On the same theme Zafiris et al. (2001) build on Lehman's earlier work on the concept of evolutionary software (See Section 2.3.3), proposing a model to cope with such demands (Figure 2.1). The characteristics of the model rely on the WIS having an open architecture for viable content management and for efficient deployment of application/ services on top of the integrated content. If such a infrastructure is achieved then the software lifecycle is considered to be in a constant state of evolution with feedback from the organisational domain being incorporated into the WIS (Cusumano and Yoffie, 1998; Bennett and Rajlich, 2000).

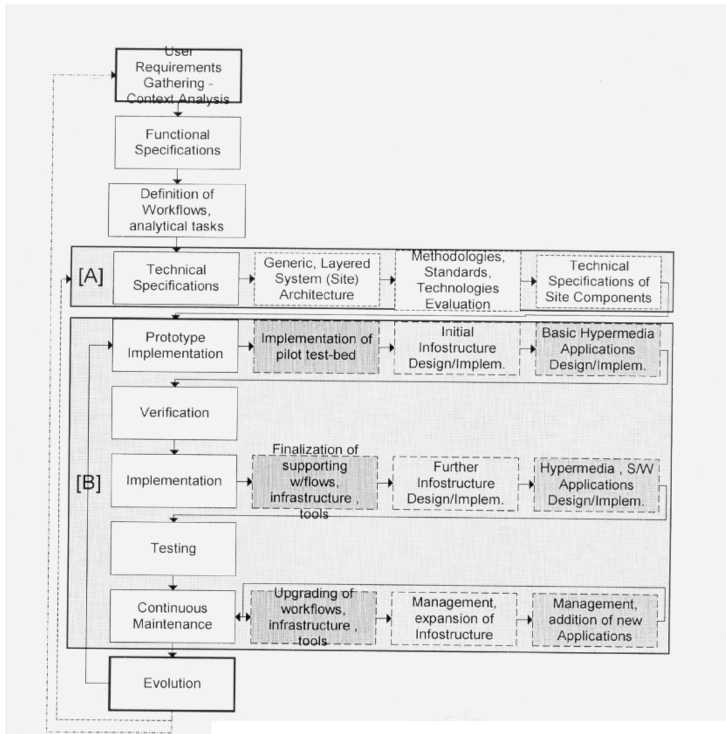

Figure 2.1 – Generic Process Model (Zafiris et al., 2001)

This view is supported by other academics in the IS maintenance realm, who recognise that systems development should be able to achieve the goals of continuous redevelopment, continual change, incorporation of dynamic requirements and their negotiation and work from incomplete or purposefully ambiguous specifications (Truex and Baskerville, 1998; Truex et al., 1999; Orlikowski, 2000; Eschenfelder and Sawyer, 2001). Orlikowski (2000) ascertains that certain technologies for WIS's should be considered to be 'reconfigurable' to allow constant and significant post-implementation change. This view that IS's planning and development should be more of an informal and incremental approach in order to cope with dynamic change has been purported by academics (Mintzberg, 1979; Quinn, 1980; Mintzberg, 1994; Mintzberg and Quinn, 1996). Truex et al. (2000) deviate further from the highly controlled and regulated software engineering practices by suggesting that an 'amethodical' paradigm should be preferred. They suggest that the intuition and rationality of experienced developers of how to act within the unique constraints of projects is more important than rigid structures (Cusumano and Yoffie, 1998; Ciborra, 1999). It is envisaged that the amethodical approach would rely on a tool kit of methods, processes and techniques that developers could use to proactively map to temporal events during the system lifecycle (Löwgren and Stolterman, 1999).

The most commonly used methodologies for WIS development over the last decade have sought to move away from a highly structured IS engineering approach, however, have stopped short of fully assimilating the amethodical approach. The focus of these 'light' methodologies are claimed to be more 'adaptive rather than predictive' (Beck, 2000; Hawrysh and Ruprecht, 2000; Cockburn, 2000) and emphasise improvisation and placidity in their approach. Examples of such methodologies include Spiral models of development (Boehm et al., 1998), Agile Software Process (Aoyama, 1998; Sutherland, 2001; Glass, 2001; Moss, 2001; Boehm, 2002; Cockburn, 2002), Scrum (Rising and Janoff, 2000; Schwaber, 2000) and eXtreme Programming (XP) (Beck, 2000; Taber and Fowler, 2000; Beck and Fowler, 2001; Jeffies et al., 2001). Although these methodologies are thought of as being 'light', if they are not practiced with an adequate amount of discipline, their use can result in the loss of control from a project management perspective. This can lead to an inefficient and chaotic software system and an unwieldy architecture (Fowler, 2000, Baskerville et al., 2009).

Logic would suggest that using 'adaptive' WIS methodologies would facilitate in the maintenance overhead and its associated activities. It has however been noted that maintenance can still be as problematic in systems based on these methodologies as compared to traditional software engineering practices.

2.2.3 WIS Maintenance

As the widespread uptake and dependency on Web based technologies increases for commercial purposes, so does the need to assess factors associated with its maintainability (Kajko-Mattsson, 2004). It is often reported that maintenance of large WIS's involves significant overheads (Brereton et al., 1998; Prevelakis, 1999; Ginige, 1998; Pressman et al., 1998). However, the Web maintenance topic is considered as a neglected subject matter and has received relatively little empirical evidence and theorising that addresses its intricate and complex disposition (Kemerer, 1995). Indeed, Glass (2001) states in relation to his 'living project' characteristic of WIS's, that maintenance by the Web developer is usually a must and the software engineering maintenance body of knowledge (See Section 2.3) is 'skimpy' and 'treacherous'. He further suggests that Web development teams are on their own in this area and that they may well lead the way in better understanding this 'vitally important topic'. Taylor et al. (2002) purports that there are few actual case studies in academic or professional literature regarding the overall process of WIS maintenance. As the functional attributes of WIS's increases as well as their scale and size, the maintenance dilemma is becoming as prominent as that faced by traditional IS's. It is foreseen in the case of WIS's that the perpetuated concerns of traditional IS's are accentuated. Reasons for this include the dynamic demands of WIS's, shorter lifecycle development times (quicker time to market), multimedia diversity, broad and diverse user base (catering for a greater cultural range), a higher emphasis on usability characteristics and an increased interoperability architecture (sub-system integration and transparency).

In her paper Cartwright (2000) presents a review of the empirical perspectives taken on maintaining web systems. She begins by posing the question – 'Why should anyone, care about maintenance of web sites?' and provides an answer by stating that '… you cannot conceal a poorly maintained site from the user/ customer, in the way a poorly maintained internal software system can be.'. The acknowledgement of WIS's as being at the forefront of the user/ client view implies that the concealment of untimely and irrelevant information cannot be hidden from the user/ client base (Lowe and Henderson-Sellers, 2001).

The flexible design of the Web infrastructure through the use of standardised protocols (for example, HTML) can prompt users and organisations to make increased demands for change. Thus customers, suppliers, partners and development teams can all have legitimate business reasons for wanting changes or enhancements that impact the WIS. However, if not managed efficiently/ effectively or designed for, the increase in changes can give rise to serious maintenance problems (Linos et al., 2001). As a WIS expands due to highly dynamic information needs (Sørensen et al., 2001), with time its management and maintenance processes may become insurmountable. Ginige (1998) supports this notion by suggesting that ‘After a while in a large web site if the information changes, making appropriate changes to all relevant web pages becomes very difficult, if not impossible.’. Warren et al. (1999) suggests that there is a growing interest in industry of the large manpower commitment required for maintenance of a WIS and that many problems seen in web maintenance are familiar to software engineering. Thus, a WIS is subject to the same pressures of evolution as with traditional IS’s.

In proposing several unanswered questions, Cartwright (2000) indicates that a key enquiry within WIS maintenance is the ability to ‘… detect maintenance “hotspots” as the focus of remedial activity?’. Through detecting areas (hotspots) of a WIS susceptible to change, WIS’s may well manifest patterned and predictable behaviour. Ginige (1998) suggests that the process of incorporating ongoing updates is continual, and that the design of a WIS requires built-in facilities to keep the information up-to-date, as ‘we cannot develop it and forget about it.’.

The introduction of WIS services to the WIS design has caused many organisations to migrate and integrate existing mainframe IS’s to take advantage of the Web platform, where methods and specific reengineering processes are utilised (Boldyreff and Tonella, 2004; Lavery et al., 2004). Web services are progressively being realised as highly vulnerable and subject to constant change (Kajko-Mattsson, 2004). Thus, having ‘successfully’ developed a WIS, businesses are faced with the same challenges on their Web applications that they experience elsewhere in the organisation, that of, change is constant. As businesses and corporations become increasingly reliant on WIS’s as a means to manage crucial information as well as provide transactional facilities, their reliability, usability and overall quality become imperative concerns to the business organisation (Boldyreff and Tonella, 2004). Boldyreff and Tonella (2004) state that web services make ‘… the domain of web applications a very dynamic one, where the evolution is rapid and unavoidable.’. One cause of this rapid rate of evolution of WIS’s has resulted from the relative ease with which changes can be made to certain aspects of WIS’s.

Web technologies and infrastructures change rapidly (Turau, 1998), coupled with increased life expectancies of web applications, longitudinal data will likely be needed in the future because of the dynamics of web-enabled commerce (Liu and Arnett, 2000). Warren et al. (1999) discusses the issues for WIS maintenance as relating to contents of pages as a free form medium and that semantic meaning is required to ensure correctness. By this they address the key issue of timely and relevant information that is at the forefront of the users. The constant analysis and maintenance of the content is a considerable challenge for businesses that adopt WIS’s; the data has to be well organised and accessible, and need to be continually updated and expanded (Martin and Martin, 2001). The concern of modifying the structure of web pages and the ripple effect it can have on other associated parts of the WIS is also a concern. Results indicate that WIS’s are subject to changes (predominantly growth) both at website (hyper document) and page (document) levels where new pages are added and existing pages extended (Martin and Martin, 2001).

The thrust of much of today’s maintenance phase analysis is directed on the source code level (for example, ripple effect analysis, slicing and broken/ redundant hyperlinks or pages). It has been recognised that WIS’s present an additional set of challenges that require its study to go beyond the source code level and require consideration at a more macro level. WIS's maintenance and its evolution should consider (Smith and Lewis, 2008):

- Analysis of WIS architecture for integration.
- Interoperability of new systems.
- Impact analysis across systems.
- Data modelling at a high level.
- Analysis of capabilities across applications.

There are also organisational and management elements to WIS maintenance which can assist in the process (Smith and Lewis, 2008):

- Coordination of release cycles between elements of a WIS when constitute systems are independent.
- Development of post deployment management procedures in a decentralised environment.
- Monitoring and enforcement of service level agreements in a WIS environment.
- Processes and rules for the incorporation, monitoring and management of components and services from third party vendors or internal departments in the evolution of the WIS.
- Determination of 'return on investment', from the perspective of the WIS and from the perspective of the constitute system.

Warren et al. (1999) acknowledge that WIS maintenance change can be categorised according to the widely accepted maintenance change type definitions (perfective, adaptive, corrective, or preventative maintenance) (IEEE glossary from 1990). However, they suggest that a new fifth type of maintenance occurs for WIS. 'Speculative' maintenance refers mostly to external objects (hyperlinks) that may have been moved, deleted or edited without being explicitly noted (Zafiris et al., 2001).

Empirically based methods can be a useful source in benefiting the decision making process of maintenance. These methods focus on documentation techniques, different types of commenting or tools or methods of source code analysis (Smith and Lewis, 2008) (See Section 2.3.4). Over the last two decades a series of tools sometimes referred to as web robots, spiders, wanderers or crawlers (for example ReWeb, Rigi, WebCopier, SiteSeer and Web site-Watcher) have been developed to traverse web sites for indexing purposes. These tools assist in evaluating process, content and reasons for changes occurring during WIS maintenance (Maes, 1994; Lieberman, 1995; Krulwich, 1997; Maglio and Barrett, 1997; Fagrell and Ljungstrand, 1998; Joshi and Singh, 1999; Warren et al., 1999; Martin and Martin, 2001; Linos et al., 2001; Sørensen et al., 2001; Tonella and Ricca, 2004; Mendo and Fitzgerald, 2005; Burgess et al., 2006; Ricca and Tonella, 2006).

Desired features that can facilitate large scale WIS maintenance should aim to address a commonly agreed upon infrastructure and set of technologies, enforcing the use of standards, and targeting loose coupling with new systems that join the WIS. In addition to this the following recommendations would also facilitate maintenance for WIS (Smith and Lewis, 2008):

- The discovery of design patterns that are appropriate for WIS maintenance and the application of these patterns to the WIS design.
- The development of domain specific WIS knowledge to identify unique rules about specific domains.
- An understanding of how security vulnerabilities in a particular new component could impact existing components in the WIS.
- Experiments and analysis of the interactions of the technology base, standards, infrastructure, and middleware for allowing components to communicate in a standard way.
- Federation of multiple WIS infrastructures.

There is an existing body of knowledge regarding systems maintenance that predates that of WIS maintenance. Although these are orientated to traditional IS's many of the concepts and solutions are viable for use in a WIS context. The following section presents an insight into the traditional IS maintenance domain.

2.3 Traditional IS Perceptions of Maintenance

The maintenance phase of the IS's development lifecycle has been recognised as a critical and important activity. The traditional perspective of IS maintenance considers it to be the final stage within the software development process. The general view on the maintenance process is as a 'black-boxed' activity separate and distinct from the development and implementation phases. Maintenance is the general name given to the set of activities on an IS after delivery to correct faults, to improve performance or other attributes, or adapt the IS to a changed environment (Taylor et al., 1997; IEEE standard, 1998, 2006).

Ligezinski (1988) proposes a technique aimed at creating a 'zero maintenance' environment for IS. The concept of zero maintenance entails allowing users to change and enhance IS's without the need of modifying any of the applications code or programs. This approach is based on the principles of the 'generalised systems approach'. The author of this approach envisages systems that are self-adapting in the face of external change (Katz and Kahn, 1978), i.e. that inputs to this type of system are generated from the system itself. In its purest form this type of homeostatic, self-equalising behaviour can only be fulfilled in IS's that are based on artificial intelligence. Currently systems based on artificial intelligence are still in an embryonic stage of development, however, self-adjusting systems do ultimately aid the flexibility of the IS without the need of a maintenance overhead.

2.3.1 Characteristics of Maintenance

The need to decompose the rather vague term of maintenance into manageable subcomponents has been addressed by IS academics. Swanson (1976) differentiates maintenance changes into three categories, these being of the corrective, adaptive and perfective types:

- Maintenance performed in response to processing, performance and implementation failures may be termed corrective maintenance.
- Maintenance performed in response to changes in data and processing environments may be termed adaptive maintenance.
- Maintenance performed to eliminate processing inefficiencies, enhance performance, or improve maintainability may be termed perfective maintenance.

Expanding on the initial definition above, corrective maintenance can be considered as when the IS behaviour does not conform to its specification. These failures may contradict the specification, or it may demonstrate that the specification is incomplete, and manifest itself in the form of an error, these are often termed ‘bugs’. Adaptive maintenance includes responding to changes in the environment in which the IS operates. The changes can occur in the data or processing environment. For example, it includes converting to a new operating system, responding to changes in government legalisations or restructuring a database. Perfective maintenance consists of dealing with changes in user requirements of the IS’s. These changes include enhancing processing efficiency, performance and IS maintainability, improving IS documentation and making enhancements for users (Lientz and Swanson, 1980; Boogaard, 1994). This nomenclature regarding maintenance change types has been commonly accepted in academic literature as well as being defined in the IEEE glossary (up till 1990).

Academic’s in the maintenance realm have looked to build upon and supplement the three initial change categories of corrective, perfective and adaptive (Reutter, 1981; Kitchenham et al., 1999; Chapin et al., 2001). Reutter (1981) identified supportive maintenance as a distinct category related to the communication between the user and maintainer as well as planning for system support. Another category which was accepted in the IEEE glossary in 1990 is that of preventative maintenance. This type of maintenance is based on the definition of changes that are undertaken on an IS to anticipate future changes and make subsequent maintenance change easier. This type of change does not necessarily change the baseline IS functionality (Bennett, 1991; Chapin, 2000; Scharl, 2000; Chapin et al., 2001; Takang and Grubb, 2003).

2.3.2 Perceived Problems of Maintenance

Due to the disproportionate amount of effort, cost and resources dedicated to the maintenance activity within the IS lifecycle, this phase has been compared to the analogy of an 'iceberg' (Canning, 1972; Trapnell, 1995). Empirical studies indicate that 50-70% of the IS resources are dedicated to the maintenance activities (Bennett, 1991; Lientz and Swanson 1980; Doke and Swanson, 1991). Even though it is evident from these findings that maintenance activities are an obvious candidate to improve efficiency, most methodologies and tools are oriented towards IS design. To amplify the issue, maintenance is often viewed as a mundane and tedious activity (Schneidewind, 1987; Bennett, 1991) resulting in a high turnover of staff associated with maintenance/ support procedures. As a result of this, temporary replacements are often drafted in by organisations, these replacements work with undocumented systems, and in doing so become indispensable and not easily exchangeable. This in turn causes serious difficulties and delays when they leave the project or the organisation (Polo et al., 2002a). However, these issues can be considered as being only the 'tip of the iceberg'. Specifically the difficulties associated with maintenance can be summarised in the points listed below (Martin and McClure, 1983; Schneidewind, 1987; Bennett, 1991; Ketler and Turban, 1992; Boogaard, 1994):

- IS's can be unstructured or become unstructured through years of modifications or up-grades. Resulting in existing IS's being difficult to understand.
- Documentation of IS maintenance changes is often insufficient.

- Personnel involved in maintenance tend to have a larger degree of turn-over as the work is perceived to be mundane. This results in the key skills to perform effective maintenance being lost to the organisation.
- The lack of trialability within maintenance results in changes being applied where the consequences are not thoroughly understood, resulting in regression and unpredictability to the system.
- Processes and mechanisms are less rigorous when applying changes to the IS. The controllable SDLC principles applied to the development phase are largely ignored within maintenance.
- Maintenance costs in the long run often exceed implementation costs.
- Maintenance overhead causes an increase in the ‘application backlog’ since resources cannot be used to develop new IS’s.

2.3.3 Complexity of Maintenance

A number of studies have linked a high software complexity with a resultant increased software maintenance effort and decreased performance (Potier et al., 1982; Gremillion, 1984; Basili and Perricone, 1984; Banker at al., 1991). In addition to this observation, Lehman’s law of increasing entropy states that ‘As a large program is continuously changed, its complexity, which reflects deteriorating structure, increases unless work is done to maintain or reduce it.’ (Lehman and Belady, 1985). This view is supplemented by the proposal that IS ‘age’ is positively correlated to its entropy, i.e. as the IS ages its entropy increases (Vessey and Weber, 1983; Jones, 1989; Gode et al., 1990). Older software modules will be more likely to receive enhancements and repairs. The longer the system is in place, the more the business will change, requiring more enhancements to the system. As systems age, they tend to become less stable with frequent modifications resulting in an increased entropy and complexity.

The characteristics of IS complexity make it difficult to understand and modify (Curtis et al., 1979; Basili, 1980; Lehman and Belady, 1985; Rosenhead, 1989; Ramanujan and Cooper, 1994; Banker et al., 1998). Attempts to quantitatively address complexity are often founded on McCabe's (1976) Cyclomatic complexity measure (Martin and McClure, 1983; Nikora and Munson, 1998), which can also be referred to as Decision Density (Kremerer and Slaughter, 1997). These measures are based upon the relative amount of decision or control paths in the software per line of code. It is reported that programs with a high cyclomatic complexity measure lead to an increased maintenance effort caused by a greater frequency of corrective maintenance that is required by such programs (Gill and Kemerer, 1991; Kemerer and Slaughter, 1997). Other more crude measures of complexity to assess change can be based on the number of lines of code added, changed or deleted. Organisations do not explicitly consider complexity when analysing the maintenance effort, even though metrics such as these are available. This has negative consequences, implying a lack of maintenance control (Polo et al., 1999a).

The issues discussed above have led to a perception common to most IS managers, that maintenance is a 'necessary evil' and that there is little that can be done to reduce this workload (Canning, 1981). Although attempts have been made to address many of the maintenance issues, it still presents itself as a problematic realm of IS.

2.3.4 Maintenance Methods

Change management (Rochester, 1990) processes such as ITIL (Information Technology Service Management) exist to manage change during the IS development lifecycle. ITIL consists of change management techniques that create a standardised set of processes as means to control and monitor requests for change. This is achieved by documenting, filtering, managing (prioritising), creating change acceptance boards and reviewing and closing requests for change (Dugmore and Taylor, 2008). Part of these overall management processes entails risk and impact analysis on the effects of introducing change to the IS (Talyor et al., 1997; Fyson and Boldyreff, 1998; Bengtsson and Bosch, 1999; Bengtsson et al., 2004). However, it is noted that there are only limited methods or techniques that have been developed in order to address the maintenance phase specifically. One such methodology is the Mantema Methodology for Software Maintenance (Polo et al., 1999b; Polo et al., 1999c; Polo et al., 2002b). This methodology is one of the few examples where processes specifically orientated for the maintenance phase has been applied to the commercial sector. This methodology has evolved from the ISO/IEC 12207 standard (IEEE, 1998) and has been incorporated during a project by the consulting firm Atos Orgin (European consultants on software outsourcing and maintenance).

Two further approaches have been developed in order to control the maintenance process. The first one being a methodology that seeks to define a formalised approach to the maintenance phase, as prescribed by Yau and Collfello's (1980) Methodology for Software Maintenance (Bennett and Rajlich, 2000). The model attempts to assist in cost effective software maintenance and the development of easily maintainable software (Black, 2001). In summary the model consists of four phases as shown in Figure 2.2:

- Phase 1 – involves a comprehensive understanding of the program. The factors that contribute to the ease of understanding are the complexity, the documentation and the self-descriptiveness.
- Phase 2 – consists of proposing a specific maintenance solution to accomplish the implementation of the program.
- Phase 3 – accounts for all the ripple effects caused by the program modifications. The effect may not be local to the modification but may also affect other portions of the program.
- Phase 4 – the modified program is tested to ensure that it has at least the same reliability as before.

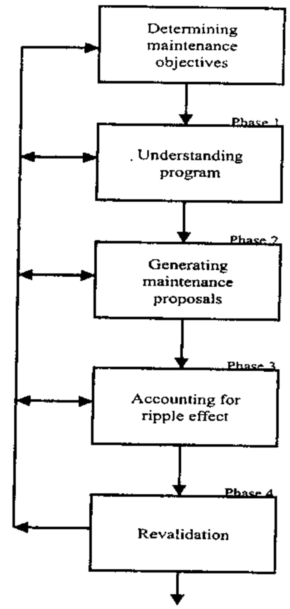

Figure 2.2 – A Methodology for Software Maintenance (Yau and Collofello, 1980)

The second approach is the Structured Analysis and Design Technique (SADT) (Pfleeger and Bohner, 1990) which has evolved from Yau and Collofello's (1980) methodology. The SADT model (Figure 2.3) differs from Yau and Collofello's methodology by having two additional phases. Furthermore, it includes the 'analyse software change impact phase' much earlier within the process. The model provides feedback outputs from each of its phases such that management can make an informed assessment before progressing to the next activity. These activities are summarised below:

- Manage Software Maintenance activity consists of two procedures, firstly to moderate the sequence of activities based on the feedback received and secondly to establish the inputs into the consecutive activity.
- Analyse Software Change Impacts examines the proposed change and decides to what extent it would affect the rest of the software.
- Understand Software Under Change involves analysing the software (i.e. complexity and documentation).
- Implement Maintenance Change provides a relevant solution for the change.

- Account for Ripple Effect assesses the extent of the impact on other modules as a result of the change that has been implemented.
- (Re)test Affected Software involves the change being validated and verified with respect to the new requirement and the system as a whole is regression tested (Kung et al., 1994).

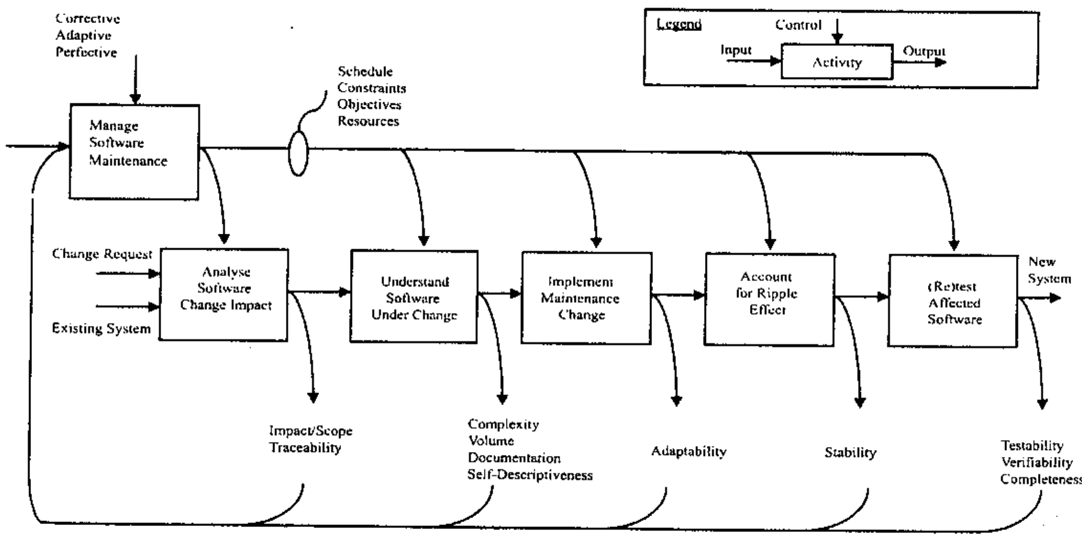

Figure 2.3 – SADT diagram of software maintenance activities (Pfleeger and Bohner, 1990)

The Methodology for Software Maintenance and the SADT model are contingent upon the use of impact analysis. The impact analysis technique determines the possible effects of proposed software change (Arnold and Bohner, 1993; Lee at al., 2000). It supports planning for and making software changes, accommodating predetermined types of change in the software, and examining previously performed changes. Impact analysis has been practiced in various forms, yet there is no consensus definition (Arnold and Bohner, 1993). Pfleeger and Bohner (1990) define change impact analysis as 'the evaluation of the many risks associated with the change, including estimates of the effects on resources, effort, and schedule.'. Turver and Munro (1994) define change impact analysis as 'the assessment of a change, to the source code of a module, on the other modules of the system. It determines the scope of a change and provides a measure of its complexity.'. Arnold and Bohner (1993) define change impact analysis as identifying the potential consequences of a change, or estimating what needs to be modified to accomplish a change. They emphasise the estimation of the impacts. The Pfleeger and Bohner (1990) definition extends their definition to the evaluation of impacts. Lee (1998) succinctly defines change impact analysis as the mechanism to understand how a change to a software system will impact the rest of the system.

Whilst impact analysis (Lee, 1998) is not yet a part of many of the established software development methods, it is increasingly being recognised (IEEE Standards, 1998) within the realm of change management (for example ITIL). Developers are also becoming more aware of the need of this type of analysis prior to making extensive software changes (Bohner and Arnold, 1996; Bohner, 1996; Chaumun et al., 1999; Bennett and Rajlich, 2000; Chaumun et al., 2002; Chen and Chen, 2009). Although impact analysis for changes seems an obvious process to apply during maintenance, there are challenges to using this approach which make it time consuming and thus costly to implement effectively. These issues can be summarised as the following (Bohner, 2002):

- Information source volume – there are a great number of software information sources to relate and analyse.
- Change semantics – methods for describing meaningful software change relationships are limited for the range of software artefacts.
- Analysis methods – methods of analysing the software work product dependency and traceability structures have not been fully explored for the growing areas of software work products.

A significant component of impact analysis dwells on the ripple effects (Haney, 1972; Collofello, 1989; Talyor et al., 1997; Fyson and Boldyreff, 1998; Bengtsson et al., 2004) that may be caused by inducing maintenance changes. Ripple effects have been defined as the consequential effects on other parts of the system resulting from a change. These effects can be classified into a number of categories such as logical effects, performance effects or understanding effects (Lee, 1998). Impact analysis is difficult because it is not clear what the impact will be on the rest of the source code once those changes are introduced; analysing ripple effects can certainly assist in this task (Black, 2001). It can highlight modules with high ripple effects as possible problem modules, which may be especially useful in preventative maintenance. It can show the impact in terms of increased ripple effect during perfective and adaptive maintenance where the functionality of a program is being modified or its environment has changed. During corrective maintenance it may be helpful to look at the ripple effect of the change program and its modules before and after a change in order to ascertain whether the change has increased, or perhaps decreased, the stability of the program. The stability of a program is fundamental to its robustness, if the stability of the program is poor then the impact of any modification is large, hence the maintenance cost will be high, and reliability may also suffer.

Computing all the possible permutations and combinations potentially associated with empirically valuing the ripple effects has proven to be a troublesome task. However, various algorithms (Haney, 1972; Myers, 1975; Soong, 1977; Yau and Collfello, 1980; Li and Offutt, 1996; Black, 2001) have been proposed and have led to the development of tools which aid in the calculation of ripple effects (for example Ripple Effect and Stability Tool - REST). To supplement this algorithmic approach Koskinen et al. (2004) recommends the use of ‘backwards slicing’. Program slicing was originally introduced by Weiser (1982) to support program comprehension and, particularly, maintenance tasks related to debugging.

Some consider that a shift in mindset is required by IS managers in the way maintenance is viewed. Schniedewind (1987) discusses the issue of the appropriateness of the lifecycle model for maintenance. He suggests that the traditional view of the maintenance stage would be better reflected as 2^{nd}, 3^{rd} …. n^{th} development and that the maintenance activity should be incorporated within the building process of the software application as opposed to the ‘black-box’ mentality.

As can be seen from the literature presented, attempts have been made to address maintenance overheads. The solutions range from management (for example ITIL) and development (for example Mantema Methodology for Software Maintenance and SADT) methodologies to more low level activities such as better documented change requests and code, as well as impact and ripple effect analysis. Maintenance management and its techniques are only slowly working themselves into the main stream system development practices. Most systems development may adopt one or more of these components, however a coherent and comprehensive maintenance strategy is still lacking during post production. This apparent neglect of maintenance considerations leads to various problems for organisations. Systems tend to experience an increased complexity and decreased stability that require a disproportionate expense in order to sustain them. The following section discusses the potential solution of flexibility as a means to facilitate the challenges posed during the maintenance phase.

2.4 IS Flexibility

The evolution of IS's can be viewed as firstly being designed to automate manual tasks (save costs), then to improve the quality of decision making and are currently at the stage where they are attempting to attain flexibility (Lucas and Olson, 1994; Suarez et al., 1995; Eardely et al., 1997). When circumstances change, often organisations are not in a position to react to accommodate changes because their IS's cannot be adjusted to new situations quickly and cost efficiently. Ashby (1956) purports that a system requires internal regulatory controls as diverse as the environment with which it interacts, thus providing the potential for the system to cope with the externally induced influences of the environment. Inflexibility of existing information architectures arises from the fact that they are geared towards particular competitive conditions and that new systems for new conditions contain an organisational change challenge (Boynton, 1993). Avison et al. (1995) state that specialists need to redress the balance so that less effort needs to be spent on maintenance because possible future changes can be taken into account during the

IS design (Fitzgerald et al., 1999). Flexible IS's should be part of IS development strategy as it is often considered to be a critical success factor. IS's should be capable of incorporating a range of possible or most probable futures (Rochester, 1989). Thus, the need for flexibility in today's very highly dynamic systems (Paul, 1993) is a recognised viewpoint (Frazelle, 1986; Allen and Boynton, 1991). However, paradoxically, technology acts as an enabler as well as a disabler to organisations flexibility depending upon the time horizon considered.

Lucas and Olson (1994) considered two types of flexibility; organisational and technological. They find that newly incorporated technology increases flexibility with regards to achieving strategic and target changes. However, in the medium to long term technology causes the organisation to become more inflexible due to the technology itself becoming outdated and difficult to maintain. It is also often stated that in order for a successful IS to be achieved, responsiveness, flexibility and innovation are key (Lambert and Peppard, 1993). Rhenmen (1973) has postulated that for different kinds of change, different kinds of response devices should be used to respond to change. 'In our terminology, this would mean that organisations need different types of flexibility to tackle different types of change. The types of change an organisation faces, determine the types of flexibility it needs.' (Eppink, 1975, 1978). The benefits of flexible IS's has been advocated in academic literature since the 1960's (Ackoff, 1967), this view has been reinforced ever since. Ittner and Kogut (1995) stated that the value of flexibility lies in increasing an organisations ability to respond to changing and uncertain environments (Ansoff, 1968, 1984, Eppink, 1978). Allen and Boynton (1991) acknowledge this by reporting that organisations need the most flexible IS's they can find. In addition Avison et al. (1995) also suggest that there are three broad advantages in adhering to flexibility. Firstly, it improves the quality of internal process in ways that may offer a variety of performance improvements. Secondly, it may give firms a competitive edge and lastly it forms part of the organisations survival kit.

2.4.1 Perceptions of Flexibility

Flexibility is an ambiguous term within research, it is not always clear what it means and often the definitions can be varied and diverse depending on the contextual setting. One reason why flexibility is so difficult to define is that definitions are often coloured by particular managerial situations or problems (Upton, 1994). Some researchers have attempted to define it succinctly (Golden and Powell, 2000).

Hees and Monhemius (1970) asserts that flexibility is the swiftness of the change, whereas Scott (1965) defines flexibility as the ability to adjust or adapt to change. An alternate view on flexibility is given by Veld (1978) who states that it is the ability to recognise changes in goals and the rapid alteration to achieve these new goals by anticipating rather than reacting. Veldwijk (1993) defined the concept of flexibility in the field of IS's as 'the extent to which information systems can be changed without the need for re-programming, re-testing, re-debugging and re-documenting.'. McClure (1989) discussed software flexibility and mentioned that a flexible system excepts abbreviations, user declarations of synonyms, missing words, run-together words, incorrect word orderings, spelling errors and semantically meaningful but syntactically incorrect user requests. She ascertained that fifty percent of the error messages are caused by inflexible IS's that do not permit any variations when entering inputs. According to Fitzgerald (1990), flexibility is a relative measure of how easy (or difficult) it is to make changes to an IS in the future. Crowe (1992) has suggested that flexibility is the ability to make substantial changes in schedules and volumes for existing products and to handle frequent product revisions and introductions.

Adaptability is the most common term whose meaning overlaps with flexibility. It differs from flexibility in one way i.e. adaptability implies a singular and permanent adjustment to a newly transformed environment, whereas flexibility enables successive but temporary approximations to this state of affairs (Stigler, 1939). The concept of 're-adaptation' better resembles the concept of flexibility, whereby 'the organisation and the environment interact and evolve toward more mutual acceptable exchanges' (Lawrence and Dyer, 1981).

2.4.2 Attempts to Enhance Flexibility

Attempts have been made to enhance flexibility within IS at various levels. Strategic IS's planning is built on the foundation of the alignment model, that seeks to map the current business strategy to the IS strategy (Hendenson and Venkatraman, 1993). This alignment model clearly needs to incorporate the possibility of future strategic change as suggested in future analysis techniques (Land, 1982; Mumford, 1983). Mumford (1983) recognises the need of exploitative future analysis and socio-technical analysis methods. Future Analysis as suggested by Land (1982) is a process that consists of four distinct stages that aim to enhance flexibility within IS. The first stage includes the identification of groups of people with which future changes can be discussed, the selection of tools and techniques that support forecasting and categorisation of the types of change which could impact the IS. The changes he identified can be classified into the following categories:

- Changes in available technology.
- Changes in legal requirements.
- Changes in economic and other environmental factors.
- Change in attitudes, expectations, tastes, or in the climates of opinion.
- Changes within the organisation.

The second stage endeavours to pre-empt the types of factors and changes that the IS will experience. The participates and researcher involved in Future Analysis attempt to answer the following questions:

1. What future changes are conceivable at the present time, and when may they occur?
2. What impact might such changes have on the IS?

3. Can the probability of such changes occurring be estimated? If so, what are the probabilities?

The third stage consists of the design team analysing components of the IS that are susceptible to any change in requirements. Stage four considers the output from stages two and three in order to construct recommendations on the planned lifecycle of the IS. These recommendations focus on how much flexibility should be built into the IS.

Flexibility Analysis as explained by Fitzgerald et al. (1998) aims to pre-empt organisational change during the early design through to the development stage (Fitzgerald, 1988, 1990). This concept builds on Land's (1982) Future Analysis proposition. Flexibility Analysis advocates the use of a team composed of senior functional and strategic management groups that would not traditionally interface with the IS community (lawyers, suppliers, trade unionists) (Galliers, 1987; Earl, 1988; Galliers, 1993; Salmela et al., 2000). The benefit of having this impartial and forward looking perspective as part of the initial systems requirements compels designers and developers to 'design systems for change' (Hart, 1964; Longworth, 1985), thus resulting in systems that can cope with change more efficiently and in turn be more flexible.

Although Flexibility and Future Analysis have led to an increased awareness in terms of flexibility as well as contributing to IS's adaptability, the authors of the techniques agree that they cannot make precise predications about the future. Indeed, Land (1982) argues that it is inherently impossible to build flexible IS's capable to cope with any change especially 'unknown unknowns'. However, it is important to develop solutions from the future needs of a system rather than for the needs of today's business (Clemons, 1991; Corbato, 1991; Clemons et al., 1995). Even authors that do not support the notion that predicting future change is a valuable technique (Paul, 1995), agree that flexibility and the need to cope with dynamic change is a necessity for contemporary IS's. The FEAST (Feedback, Evolution, and Software Technology) hypothesis formulises the views of Belady and Lehman and states that (Lehman, 1998):

> 'The software process constitutes a multilevel, multi-loop feedback system and must be treated as such if major progress in its planning, control, and improvement is to be achieved.'

Lehman (1998) views 'ceaseless maintenance' as an integral part of the software process and advocates the re-conceptualisation of system development and system maintenance. The grounding of this view is based upon an operational domain that is dynamic and must be accommodated by software processes that are evolutionary.

Boogaard (1994) states that a prerequisite for flexible IS's is a system architecture that isolates the distinct sources of IS changes; changing business rules and information needs, technological evolution, and changing in the organisation of work. To some extent object orientated technology attempts to compartmentalise and modularise components of the IS, based on human conceptualisation. A common set of object classes isolate and provide a comprehensive set of building blocks that can be used to construct IS's. The advantages gleaned from having a generic library of object classes is that it increases flexibility by offering reusability and extendibility achieved by the concept of inheritance, encapsulation, modularity and polymorphism (Rumbaugh, 1991; Coad and Yourdon, 1991a, 1991b; Ambler, 1995; Graham, 1995; Berard, 1996; Riel, 1996; Tudor and Tudor, 1997). From an IS perspective changes to object orientated based systems can be incorporated more efficiently and with less risk of destabilising the system through regressions.

Moitra and Ganesh (2005) assert that the facilitation of organisational adaptation has not been realised by the expected prospective role of IT. They suggest that the cause of the disappointment is a consequence of existing systems not possessing the required levels of flexibility to sustain changes in business processes. Web services have emerged as a disruptive technology, with a real potential of enabling flexible business processes. Web services are the realisation of Component Based

Development (CBD). It is an approach to software development in which all artefacts from executable code to interface specifications, architectures, and business models; and scaling from complete applications and systems down to small parts, can be built by assembling, adapting and ‘wiring’ together existing components into a variety of configurations. On the other hand, a component is a coherent package of software artefacts that can be independently developed and delivered as a unit and that can be composed, unchanged with other components to build something larger (D’Souza and Willis, 1999). Also Kozaczynski and Booch (1998) states that a component is a physical and replaceable part of the system that conforms to and provides the realisation of a set of interfaces. The concept of flexibility within CBD has been recognised to enhance IS’s adaptability, corrigibility, manoeuvrability, robustness and versatility (Siddiqi, 2002). CBD has led to the proliferation of web service based architectures in contemporary WIS’s by providing a common and reusable interface that allows interoperability with diverse and disparate sub-systems. CBD can support capabilities that are impractical for small objects such as language-independent access of interface so that the components can be used that are written in other languages, which makes them versatile and capable of transparent interaction between distributed components (D’Souza and Wills, 1999). Web service based architectures are the realisation of earlier work by practitioners and academics such as Buschman et al. (1999) that envisaged components having an interface that provides access to its services and these serve as building blocks for the structure of a system. On a programming language level, components may be represented as modules, classes, objects or a set of related functions.

Flexibility on a micro level, i.e. even within programming languages can contribute to the overall system flexibility. Forth generation languages give developers the ability to prototype solutions before implementing them fully into the IS, thus reducing large scale changes and debugging on a system level (Avison et al., 1988; Pressman, 1994). The rigorous testing involved with prototyping increases the trialability dimension of flexibility (Knoll and Jarvenpaa, 1994).

Multiview is a systems analysis methodology (Avison and Wood-Harper, 1990; Avison and Fitzgerald, 2003) that attempts to address flexibility on two broad fronts. Firstly, by providing a contingency framework and secondly by providing key flexibility throughout its process. Multiview as a methodology shifts the emphasis in IS's away from technical systems to a more socially orientated system. The methodology recognises that tools and techniques that may be appropriate for some IS's may not be appropriate for others. It provides a flexible framework where the techniques and tools available are tailored to the system/ development context. The contextual input to the methodology is a combination of issue and task related activities with the output from the methodology being flexibility requirements that the system should encapsulate.

2.4.3 Determinants and Dimensions of Flexibility

There has been an awareness to move away from considering flexibility as an all encompassing term. Academics in the field of IS have sort to delineate the term into constituent parts to reflect a more granulated perspective on the flexibility term. A sample of the key research that have adopted this approach is listed below. Appendix A includes a further list of research authors and a summarised description of their output regarding the de-segmentation of the flexibility term. Rochester (1989) states that there are three different points of view on flexibility:

- The business view of IS flexibility means the capability to build or adapt IS's in concert with corporate changes.
- The user view of IS flexibility means both intuitive and adaptable user interfaces.
- The IS department view of IS flexibility means portability, connectivity and maintainability or the ability to change IS's quickly.

Allen and Boynton (1991) suggested that architecture strategies to incorporate flexibility should consider:

- Low road – IS technology and its management are dispersed widely throughout the firm.
- High road – Core IS activities of the business are centralised. IS infrastructures built around the central precepts of corporate wide networks.

Boogaard and Spoor (1993) took the approach that a conceptual framework for flexibility should consider the following features:

- Agent – This feature indicates whether the mobilisation of the potential arise from within the IS itself (internal) or from an outside source (external).
- Process – The process feature refers to the application area of the flexibility potential. Use, development and operations.
- Recipient – Recipient feature involves the implications of an alternation on the elements of the IS. Architecture, database structure and function.
- Method – Method feature emphasises the procedure that is followed when the flexibility potential is mobilised. Parameters, library and development.
- Inception – Indicates the moment of mobilisation of the flexibility potential. Time of disturbance (stimulus), time of counter measure (response) and time of perception. Reactive or anticipative.

Victor (1995), Nelson and Nelson (1997) and Nelson and Ghods (1998) segregate flexibility into two types; structural and process. Structural being the design and organisation of a technology to be successfully adapted to business process changes. Process is the ability of people to make changes to the technology using management processes that support business process changes. These two types are further divided into six flexibility determinants:

- Modularity (structural) – The degree of formal design separation within a technology.
- Change acceptance (structural) – The degree to which a technology contains built-in capacity for change.

- Consistency (structural) – The degree to which data and components are integrated consistently across a technology.
- Rate of response (process) – The degree to which changes can be made to a technology in a timely manner.
- Expertise (process) – The degree to which up-to-date knowledge about the operations and maintenance of a technology exists and is communicated.
- Coordination of action (process) – The degree to which the technology maintenance and user organisations operate according to the requirements of each other and total organisation.

Golden and Powell (2000, 2004) investigated organisational and strategic flexibility and identified the following dimensions:

- Temporal – How long it takes an organisation to adapt.
- Range – The number of options that an organisation has open to it for change that was foreseen and the number of options that it has available to react to unforeseen change.
- Intention – Whether the organisation is being proactive or reactive.
- Focus – Flexibility gained internally to the organisation or by managing external relationships with trading partners. Flexibility is created.

Byrd and Turner (2000) summarised their findings into three flexibility dimensions which are:

- Integration (technical) – Transparent access, IT connectivity and IT compatibility.
- Modularity (technical) – Application functionality and database transparency.
- IT personnel flexibility (human infrastructure) – Technology management, business knowledge, management knowledge and technical skills.

Knoll and Javenpaa (1994) identified three types of flexibility: flexibility in functionality, flexibility in use and flexibility in modification. The three types relate directly to the management of IS's performance and are further divided into a comprehensive set of flexibility dimensions as shown in Table 2.1. The authors propose the flexibility dimensions as a possible means to generate IT requirements in light of highly dynamic environments.

Type	Determinants	Description
Flexibility in Functionality	Robustness	Technology with the necessary cohesion to withstand a variety of pressures.
	Scalability	Technology that can be increased or decreased by small increments to accommodate load levelling.
	Slack	Technology whose excess capacity is able to withstand increasing loads without any modification.
Flexibility in Modification	Commensurability	Size of change effort is proportional to benefits of change.
	Feedback Sensitive	Technology that can be adjusted based on feedback.
	Goal Adjusting	Technology that can change internal goals based on external or internal feedback.
	Just-in-Time Adjusting	Technology that can be adjusted close to the time of its use.
	Polyadjustable	Technology that can be adjusted by a variety of mechanisms.
	Self-Adjusting	Technology that has all required adjustment capabilities within its boundaries.
	Trialability	Ease of demonstrating a technology's features through prototyping.
Flexibility in Use	Concurrency	Technology that appears to give simultaneous access because it can manage two or more rapidly fluctuating relationships.
	Connectivity	Technology that has similar processes and structural features at its boundary allowing exchanges of information between parts.
	Modularity	Self-contained technology that fits with other basic parts for more complex subsystems. The technology can be reused for the same task or a different task.

	Multiple Forms	Technology capable of producing essentially the same information using various representation systems.
	Responsiveness	Technology that provides feedback of the kind and speed that is acceptable to the user.
	Reusability	Technology that can repeat its functions for use in similar task.
	Spatial Decoupling	Technology whose use is easy to infer based on its similarity to other structures and processes in the task domain.
	Temporal Decoupling	Technology that can exchange information without regard to start or stop times, or rate of functioning.
	Transparency	Technology whose use is easy to infer based on its similarity to other structures and processes in the task domain.
	Versatility	Technology that can incorporate other technologies for expanded capabilities.

Table 2.1 – Dimensions of Flexibility (Knoll and Jarvenpaa, 1994)

2.4.4 Barriers to Flexibility

Commercial pressures against building flexible systems have become more apparent in the recent past. Globalisation has resulted in a reduced cost of specialised expertise which offers organisations the possibility of developing ‘simple’ (without consideration of flexibility) applications at a low cost. Even if these systems suit the organisations needs for a short period of time for example one year, it may be more cost effective to create three simple ‘disposable’ applications in three years rather than creating one system which has flexibility built into it that has a lifespan of three years. This proposition is concurrent with Argyris (1997) who equates this view to systems which are orientated for single loop learning at the expense of double loop learning.

The principles of standardisation within IS by their very nature imply concepts of closure, stabilisation and irreversibility (Bijker et al., 1987; Callon, 1991, 1992, 1994;

Misa, 1992, 1994). As can be seen from the various definitions and dimensions of flexibility stated above these concepts suggest that standardisation and disciplined methodology sits at the polar opposite of flexibility (Behrsin et al., 1994; Hanseth et al., 1996; Iansiti and McCormack, 1997; Thomke and Reinertsen, 1998; Cusumano and Yoffie, 1999a; Cusumano and Yoffie, 1999b). Indeed, what is required is a trade-off between the application of standardisation and flexibility within an IS. Both factors are fundamentally important to the success of an IS. Figure 2.4 shows the balance between flexibility and standardisation.

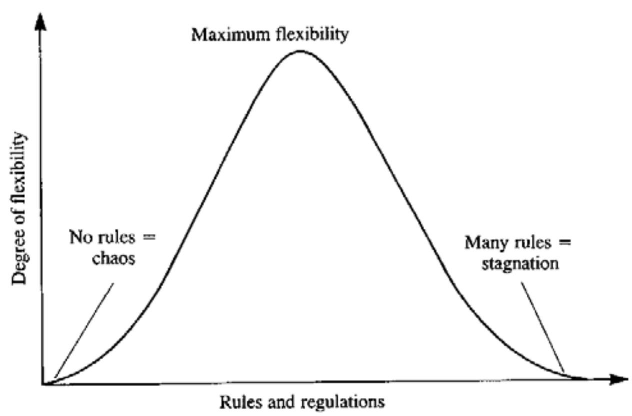

Figure 2.4 – Balance between chaos and stagnation (Behrsin et al., 1994)

2.5 Conclusion

It has been suggested by leading practitioners and academics that WIS's is and should be thought of as being different to traditional IS's. Certainly the effect of WIS's on social, cultural and on a commercial level has been unprecedented. IS's technology has branched out of the confines of organisations to the forefront of a globally interconnected and inter-correlated modern society. WIS's have swiftly evolved from a substitute for publishing media (brochure ware) into highly diverse, large scale, complex entities that can now subsume existing traditional IS's as well as assimilating other WIS's. This has been made possible because the Web itself has evolved from a simple global hypertext system to a system that is composed of a network of networks and has the potential to allow the interconnectivity of any application to any application. WIS's of today are systems of systems and this trend appears only to be increasing. The proliferation and growth of WIS's are leading to the view that WIS development needs to be reconceptualised and recalibrated in order to accommodate this constant growth. Today's view on WIS development is based on the tenant that WIS's are never totally 'finished', and are in a constant state of evolution. Using traditional software engineering methods and techniques that have a clear 'beginning' and 'end' with respect to the development process seem outdated and inappropriate for the highly adaptable needs of WIS development. In an attempt to address this, development methodologies have also evolved. This being exemplified in the inception of firstly web engineering techniques and latterly 'light' and adaptive development methodologies (for example agile and XP), that claim to be better suited to the needs of a WIS context. These claims may be valid, however, from a system maintenance perspective progress appears to be limited, maintenance is still as problematic in the WIS domain as it has been traditionally in the IS domain.

A large body of knowledge regarding traditional IS maintenance depicting its nature has been long established. These studies have led to the classification of the key maintenance change types of corrective, perfective, adaptive and preventative. Tools, techniques and methods have been developed to address the issues of control and management of maintenance change (for example Mantema Methodology for Software Maintenance and SADT), however, the uptake of these methodologies and approaches appear to be limited. Individual techniques that facilitate maintenance change seem to have been more successful and are finding themselves incorporated into formal frameworks such as ITIL, which utilise mechanisms such as impact analysis and comprehensive change request documentation as part of their processes for maintenance. On a micro level, techniques such as ripple effect analysis and slicing are also gaining traction with developers. Unfortunately a coherent approach to the maintenance phase is still lacking.

The flexibility concept has been alluded to as another potential solution to cope and facilitate change during maintenance. Flexibility and Futures Analysis are based on the premise of pre-empting the changes that will occur during the maintenance phase of the system lifecycle. Once this has been ascertained it is proposed that flexibility mechanisms can be proactively built within the system. This predicative 'designing for change' approach has to some extent defined how WIS's have evolved. The characteristics of having an open architecture, that allows component and object based elements to be built upon one another and to be integrated across common communication protocols make the web and WIS's inherently more adaptable on a macro level. However, WIS's can still incorporate a greater degree of flexibility within their scope to further facilitate maintenance change. Building upon the Flexibility and Futures Analysis techniques, it would be possible to identify where and how to apply flexibility once the potential change areas of the WIS have been identified.

The flexibility concept as a panacea to cure the ills of the maintenance phase do not come without its own set of obstacles. It has been recognised that a system that is highly structured and ruled based diminishes the scope for the application of flexibility. The conditions to allow flexibility to flourish require a system where structure, standardisation and formulised processes are diminished, however, such conditions would lead to a state of chaos. Incorporation of flexibility thus needs to strike a fine balance between these constraints to establish an optimum solution. Perhaps the greatest obstacle that IS managers and organisations face is that flexibility solutions are often expensive when viewed over the short term and need a large initial investment. In today's fast paced business climate it is often difficult to obtain funding for solutions that only reveal their cost-effectiveness over the long term.

There is a school of thought that views maintenance itself as a misnomer (Martin and McClure, 1983). Proponents of this view argue that systems development and systems themselves should be thought of as being in a constant state of evolution i.e. the system is never complete and maintenance as viewed traditionally never begins. They advocate systems that are composed of elements that allow adaptation to occur easily and propose amethodical processes and techniques for the development to work in tandem with continuous change.

The following chapter outlines and discusses the research design that will be undertaken to identify areas of change within a WIS maintenance context. The view and opinions expressed in this chapter have aided the researcher to focus on key considerations that are taken into account during the research design and analysis. In particular the concepts taken forward into this research can be summarised as follows:

- WIS and IS context in-situ of maintenance.
- Influence of development processes on the maintenance phase.
- Maintenance change categorisation.
- Existing maintenance management and techniques.
- The consideration of ripple effects, impact analysis and complexity.
- Adaptive and predictive flexibility philosophical perspectives.
- Acknowledgement of flexibility determinants and dimensions.

.

Part II ▪ The Research Process

Part II of this thesis will address the philosophical underpinnings of the research design and method, as well as the subsequent implementation of the framework to answer the research questions posed at the outset. The research process had to address two high level objectives, firstly to ascertain the nature of WIS maintenance change and secondly to construct a model by which this change could be facilitated by the use of flexibility.

The following two chapters relate to the design of the research framework and its operationalisation, respectively. The design chapter (Chapter 3), shows how the final research framework evolved from an initial design, therefore two distinct research strategies are described here. This was to emphasise to the reader the logical progression and thought processes that culminated in the final research strategy selected, to address the thesis objectives.

The operationalisation of both frameworks are described in Chapter 4 to emphasise how the inner workings specified by the designs, evolved into a final research framework. Crucially the initial research framework design and operation revealed to the researcher that this strategy would not be adequate enough to substantiate the first objective that relates to the nature of WIS maintenance change. Results from the initial framework naturally suggested that greater granularity was required within the research strategy. Thus, the second research framework evolved on the premise that greater granularity into the WIS areas of maintenance change was required. Additionally a new metric of analysing change would also be required in order to provide a greater insight and perspective into these types of changes (constructs of analysis).

The following two chapters are not necessarily presented in the manner that is common to most IS based PhD’s. However, in terms of research approach and strategy it is not uncommon for these to change or evolve during the course of an investigation. Indeed as part of this organic process questions, objectives and issues are clarified and refocused which was the case within this research.

Chapter 3 ▪ Research Design: Building a Research Framework to Investigate Maintenance change in Web Information Systems

3.1 Introduction

The literature review in the previous chapter highlighted issues purporting to the IS maintenance quandary and how this may be potentially resolved. The raison d'être of IS's is to provide timely, concise and appropriate information in response to fluctuating business requirements in an effective and efficient way (Ives and Learmonth, 1984). In order to remain contemporaneous, IS's should be able to adapt to changes that can influence its scope and form. A prominent area of discussion in existing literature, relates to the difficulties of maintenance in the software lifecycle. The maintenance phase of a system indicates the extent to which the overall IS is subject to change. In most cases the development phase should ensure that the IS conforms and meets its initial operational objectives. However, in some instances projects are operational even though the development phase is incomplete, this being due to time and cost issues. The adaptability and responsiveness of an IS in a turbulent environment is all important, however, IS's are difficult to adjust to changing requirements. This inflexibility of IS's during development, signals increasing entropy and lengthens the maintenance phase (Boogaard, 1994). As Chapter 2 discussed extensive research has been conducted on IS maintenance, however knowledge within the WIS maintenance realm remains a largely unexplored area. It appears that little research has been undertaken in order to determine how to develop WIS's that can make future maintenance more efficient and cost effective (Taylor et al., 2002). Academics and IT practitioners have suggested that a very few WIS methodologies and frameworks have been advanced, which do not have an emphasis on the maintenance processes of the system's lifecycle (Taylor et al., 2002).

Management and development methods are amongst the many attempts made in addressing the needs of changing business requirements. Another perspective on the problem of rigidity of IS's appears to be justifiable. The concept of inherent systems flexibility could be an additional solution to facilitate change more efficiently. It is recognised that flexibility is not the panacea for all the problems that a business faces, however, the advantages of a flexible software system to redress the instability that change can induce is evident. Flexibility still remains a rather ambiguous concept within IS (Fitzgerald 1990; Allen and Boynton, 1991; Golden and Powell, 2000) and is often used within specific contexts (Ackoff, 1967). Furthermore, in addition to the costs that can accrue from development and maintenance processes, the cost benefit analysis of implementing flexibility is difficult for businesses to quantify. Another possible point associated with flexibility is the delicate balancing act of robust stable systems that benefit from IS standardisation and an increasing need for inherent flexibility within the software system.

It is suggested in literature (Land, 1982; Fitzgerald, 1990) that flexibility can be better achieved by determining where it is needed in the software system. In order to determine where flexibility is needed, the areas where change is occurring have to be distinguishable. This line of reasoning serves as the platform of this research. The research starts off from the premise that by identifying areas of a IS that are susceptible to change, flexibility solutions can then be identified and reciprocally mapped to facilitate this type of change in the future. Academics suggest that the techniques of Flexibility Analysis and Future Analysis are necessary in the development phase of the system's lifecycle to better accommodate change (Land, 1982; Fitzgerald, 1990). Fitzgerald (1990) asserts that future changes can be accommodated for because they are previously known by someone in the organisation. Although this information may exist in the business, it may nonetheless be difficult to elicit in such a pre-emptive manner due to internal political business issues. This research posits the view that change is unpredictable in nature and should be studied in-situ of when it occurs rather than what the business organisation predicts; where it could be difficult to forecast change when it has not even materialised. Through investigating changes as they are actually impacting the software system, a knowledge base can be drawn that is based on factual information as opposed to presuppositions. This approach limits ambiguity and the potential for errors where time and money may be invested unnecessarily into an area of the system where change may not ultimately occur. This results in unwarranted redundancy in parts of the software system and its overall inefficiency. The factual information pertaining to change can then be utilised in a proactive way to develop flexibility mechanisms that ultimately facilitate future changes.

As elaborated on in Chapter 2 software systems need to be adaptable in the face of dynamic business environments and this is particularly so for fast evolving systems such as web technologies. Research on change and its effects in the maintenance phase of WIS's with the proposed solution of flexibility is limited. Addressing this issue serves as the focal point of this research. The nature of WIS post-development change (research question one) must be substantiated as a prerequisite to addressing the wider issue of facilitating maintenance (research question two). An issue that resonates from Chapter 2 which will be considered in this chapter, is the comparability of IS's and WIS's, and forms a basis of enquiry when considering change in the maintenance phase. Another contentious topic, is the point at which maintenance begins in the system's lifecycle and this will be determined within the design of the research framework.

The aim of this chapter is to design a research framework that will assist in investigating the phenomenon of post-development WIS change and its facilitation through flexibility. The following section will acknowledge the plurality of research strategies and will justify the use of a case study in this thesis (Section 3.2). Section 3.3 is the design of Research Framework A that adopts an existing IS perspective of ‘areas of change’ (AoC) as per Fitzgerald et al. (1999). An account of how and why Research Framework A was suspended will be provided in Section 3.4. It will be shown that there was a necessary change in the research perspective and design framework of the enquiry, in order to answer the research questions more cogently. Finally, a new Research Framework B is designed based on a contextually ‘grounded’ emergent approach (Section 3.5).

3.2 The Research Approach and Strategy

Research design approaches within the IS discipline has often remained contentious in that academics draw upon a wide variety of foundations to support their position. This has resulted in IS research having a multi-disciplined perspective, encapsulating a plurality of approaches, ranging from qualitative to quantitative, positivism to interpretivism or exploratory to confirmatory, as can be seen in Table 3.1. It is this pluralism of research philosophies, methods and techniques that can afford a better understanding of the phenomena being studied.

Research Approach	
Interpretivist No universal truth. Understand and interpret from researcher’s own frame of reference. Uncommitted neutrality impossible. Realism of context important.	**Positivist** Belief that world conforms to fixed laws of causation. Complexity can be tackled by reductionism. Emphasis on objectivity, measurement and repeatability.
Qualitative Determining what things exist rather than how many there are. Thick description. Less structured and more respective to needs and nature of research situation.	**Quantitative** Use of mathematical and statistical techniques to identify facts and causal relationships. Samples can be larger and more representative. Results can be generalised to larger populations within known limits of error.
Exploratory Concerned with discovering patterns in research data and to explain/understand them. Lays basic descriptive foundation. May lead to generation of hypothesis.	**Confirmatory** Concerned with hypothesis testing and theory verification. Tends to follow positivist, quantitative modes of research.

Induction Specific instances used to arrive at overall generalisations. Criticised by many philosophers of science but plays an important role in theory/hypothesis conception.	**Deduction** Uses general results to ascribe properties to specific instances. Associated with theory verification/falsification and hypothesis testing.
Field Emphasis on realism of context in natural situation, but precision in control of variables and behaviour measurement cannot be achieved.	**Laboratory** Precise measurement and control of variables, but at expense of naturalness of situation, since real-world intensity and variation may not be achievable.

Table 3.1 – Types of research dichotomies (adopted from Fitzgerald, 1997)

There has been a tendency in the literature to view the philosophical research approaches presented in Table 3.1 as mutually exclusive and polar to one another (Lee, 1999). Nevertheless, some researchers have counter-argued that these should not be viewed in complete isolation (Morey and Luthans, 1984) but rather several of the contrasting philosophical research approaches in Table 3.1 are recommended during the course of the research (Gable, 1994; Kaplan and Duchon, 1998; Lee, 1999). This is to ensure the validity of the appropriate approach in answering the actual question(s) being researched. This thesis embraces three of the above (Table 3.1) philosophical research approaches, that of exploratory, confirmatory and qualitative (these will be discussed in context of Research Frameworks A and B).

In addition to the plethora of philosophical research approaches, there exist several research strategies to collect and examine data of a particular phenomenon. Yin (2003) outlines five main research strategies as being: experiment, survey, archival analysis, history and case study. A popular misunderstanding is that a certain strategy can only be compatible with a certain philosophical research approach; however in some instances they can be interchangeably applied. Yin (2003) establishes three conditions for selecting the correct research strategy. The first condition relates to the nature of the research question which normally takes the form of “where”, “what”, “how”, “why” or “who”. It can be seen from Table 3.2 how certain forms of research questions apply to the various research strategies. In this thesis research question one is a “what” question and research question two is a “how” question. Both are exploratory in nature and therefore any one or a combination of the research strategies seen in Table 3.2 can be used.

Strategy	Form of Research Question	Requires Control of Behavioural Events?	Focuses on Contemporary Events?
Experiment	How, why?	Yes	Yes
Survey	Who, what, where, how many, how much?	No	Yes
Archival analysis	Who, what, where, how many, how much?	No	Yes/ No
History	How, why?	No	No
Case study	How, why?	No	Yes

Table 3.2 – Relevant situations for different research strategies (Yin, 2003)

The second condition is where the researcher has no control over the phenomenon, and in this event any research strategy can be used except for the experiment strategy according to Yin (2003). The third condition relates to whether the events being researched are contemporary. If the research focus is on contemporary events then a survey, case study or experiment can be used.

Based on the criteria presented above a combination of strategies was used, however the main focus within this research was dominated by the case study approach. Within the design scope of Research Framework A the additional strategies of survey and action research were to be used (Section 3.3). Research Framework B consisted entirely of a case study strategy, the reasons for which will be explained in Section 3.4.

3.2.1 A Case Study Strategy

A case study strategy was adopted in this thesis in order to investigate the phenomenon of change in the post-development phase of WIS’s. The advantages and disadvantages of using this strategy are set out in the ensuing discussions.

The case study strategy as 'a way to systemise observation, describing ways of collecting evidence and indicating the type of tools and techniques to be used during data collection' (Weick 1984; Galliers, 1992) has gained recognition as an applicable and well used research methodological approach within the IS research field (Galliers and Land, 1987; Cash and Lawrence, 1989; Galliers, 1992; Avison, 1997; Klein, 1999). Case studies are well suited to understanding IS's in an organisational context because it allows the views of several philosophical perspectives, for example a positivist approach (Galliers, 1992; Yin, 2003) or an interpretive approach (Walsham, 1993). It is a strategy that encroaches on both research stances. A broad definitional explanation for the case study strategy is provided by Fitzgerald (1997) who purports that according to one or more common interpretations, the case study strategy describes a single situation at any given moment and provides rich data that is both qualitative and quantitative (Benbasat et al., 1987; Lee, 1989; Yin, 2003).

Benbasat et al. (1987, p.370) provides a succinct definition that is used as the basis for this research: 'A case study examines a phenomenon in its neutral setting, employing multiple methods of data collection to gather information from one or a few entities (people, groups, or organisations). The boundaries of the phenomenon are not clearly evident at the outset of the research and no experimental control or manipulation is used'. The definition highlights the fact that the strategy allows the phenomenon to be viewed as it is taking place. The case study strategy is however not totalitarian in its approach, but rather polymorphic in nature, allowing for the theoretical aspects of the strategy to be adapted to the particular epistemological research stance (Cavaye, 1996). The multi-faceted nature of case studies has given rise to a diverse variation in how they can be used, and thus can be viewed, designed, applied, conducted or reported (Cavaye, 1996; Yin, 2003, Vander Blonk, 2003). The strength of the case study is its ability to provide a framework to collect various types of evidence: documents, artefacts, interviews, observations (Yin, 2003) and to give a holistic view of the phenomenon being studied. The case study does not call for a particular type of evidence nor does it require the use of a particular data collection method. The researcher determines how to collect, organise and present relevant data to substantiate the research questions. This is supported by the community of the case study research strategy in that the methods are ones that have been used by other published and accepted researches (Yin, 2003).

3.2.1.1 Limitations of a Case Study

A main limitation of the case study approach is its lack of generalisability (Galliers, 1991, 1992), for some this is due to weak internal and external validity (Cooper and Zmud, 1990; Benbasat and Zmud, 1999) and others the fact that generalisations by their nature will be contextually and temporally relative (Lincoln and Guba, 1985). The criticisms surrounding generalisability of a case study has been approached from three aspects. Firstly, from the rigorous usage of the case study strategy from an interpretive position, Orlikowski and Baroudi (1991) argued that validity is maintained through the cogency of the logic used in results analysis by drawing inferences from extrapolation of one or more individual cases. The second issue was from the approach of the strategy itself as Lincoln and Guba (1985) state that it is ‘a powerful means for building – intuitive, empirical, based on personal, direct and vicarious experience’. The third frequent criticism of case studies are that they take too long and result in huge amounts of unreadable documents (Feagin et al., 1991); but case studies do not necessarily have to take this form (Yin, 2003). However, Lee and Baskerville (2003) assert that ‘In a case study, the researcher may appropriately strive to develop a theory that is generalisable within the case setting.’. They further state that ‘In neither case [case study and statistical study], however, would it be appropriate to criticise a theory for a lack of generalisability to other settings …’. By this they mean it is for the theory to survive an empirical test in other case study settings (Baskerville and Lee, 1999).

3.2.1.2 Case Study in Context of the Research

The case study aim in this thesis is to collect change requests and to understand the phenomenon of change in context of WIS maintenance. A case study is particularly appropriate as it enables the capturing of a ‘snapshot of reality’ (Guba and Lincoln, 1981). This thesis utilises the case study strategy with the purpose of examining change in its ‘real-life events and contextual conditions’ (Yin, 2003). The case study provides a rich pool of contextualised data from which dynamic and evolving organisational WIS’s can be studied. Change in the maintenance phase is unpredictable, emergent and continuous, and for these reasons needed investigation over a significant period of time. Each instance of data collection within the remit of the research frameworks was conducted over a three month post-development period. This strategy is also useful where the phenomenon has no boundaries and is not clearly defined from the outset. Yin (2003) suggests that researchers can retain the holistic and meaningful characteristics of real-life events such as individual life cycles, organisational, managerial processes (in this research it is the maintenance process of WIS) and the maturing of industries. An additional justification for using a case study strategy in this research is articulated by Taylor et al. (2002) who asserts that ‘There appears to be few actual case studies in academic or professional literature regarding the overall process of developing a company Web site and even fewer regarding the maintenance of company Web sites’.

Research Framework A takes on board the existing debate identified in Chapter 2, which holds that differences or similarities may exist in methodologies, processes and management techniques between traditional forms of IS’s and WIS’s. The argument underpinning the debate is largely based on fundamental differences between traditional IS and WIS (Section 2.2.1). The proposition within this research is to consider the issue in the realm of the WIS maintenance phase of the project lifecycle, by using Fitzgerald et al.’s (1999) ‘areas of change’ IS categories as an existing literature based model. Additionally, the use of the case study approach builds upon the survey study conducted by Fitzgerald et al. (1999). Empirically little is known about the activities associated with web maintenance (Taylor et al., 2002) and thus the theorisation and categorisation of change was required as a prerequisite in understanding the nature of this phenomenon.

Another consideration relevant to the research design is what constitutes the maintenance phase. For the purposes of this thesis and as discussed in Chapter 2, this is regarded to be everything after the point of development and implementation where corrections, amendments and enhancements in the form of requests for change (henceforth referred to as change requests) arise, and the system is in operational use. The data collection begins at this point and change requests need to be collated from the maintenance phase of the WIS in order to produce the initial dataset. Change requests are considered to be user-triggered requirements that lead to the modification or enhancement of the system, within the maintenance process. The dataset derived from the change requests was used in this research to identify areas of the WIS that require change.

A suitable case study candidate for this investigation required the reflection of the current state of the domain and represents issues and problems in-situ. Thus a relevant case study had to adhere to the following research criteria:

- A WIS development and management perspective for the collection of data material.
- The development of contemporary WIS's that represent its matured complexity and growth from basic brochure-ware type project initiatives to functional commercial business applications. The scope of the WIS's should extend over medium to long term life expectancies.
- The ability to collect and document change requests that improve and enhance the WIS within the remit of the maintenance phase.

3.3 Designing a research framework: Research Framework A

The design of Research Framework A firstly incorporated an exploratory approach where the researcher addressed research question one by identifying areas of the WIS requiring change. Secondly, a confirmatory philosophical research approach was taken to confirm the use of Fitzgerald et al.'s (1999) categories in a WIS context. Thus, the purpose of Research Framework A was to:

- Ascertain areas of the WIS that are susceptible to change (an exploratory approach).
- Confirm whether WIS maintenance change can be studied by applying IS 'areas of change' categories (Fitzgerald et al., 1999) into a WIS context (a confirmatory approach).
- Determine how WIS maintenance change can be facilitated through designing and applying a flexibility model.

Figure 3.1 illustrates the overarching process of Research Framework A and includes the six stages of: data collection and preparation from the case study, the categorisation process, conducting a survey, conceptual design of a flexibility model, action research, and a revised flexibility model. Each stage of Research Framework A is discussed in turn.

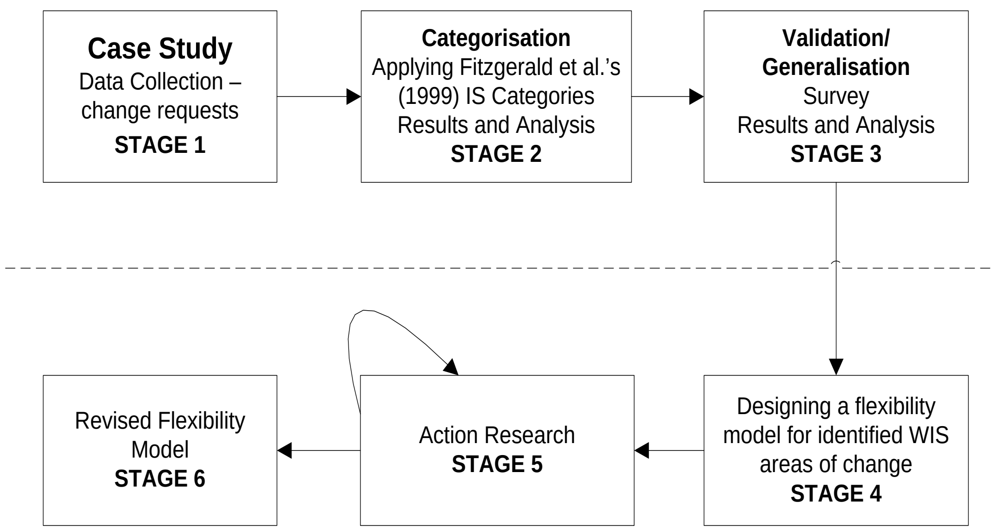

Figure 3.1 – The research process of Research Framework A

The first stage of Research Framework A relates to the collection of change request data that was attained from the case study itself. Research Framework A, begins by monitoring and recording post-development change requests that are used in the second stage to identify the areas of the system that require change. It is during this stage that the multiple sources of change request data is consolidated into a coherent dataset. Changes to the system originating from the client organisation during the maintenance phase are relayed to the software vendor and were regarded as ‘change requests’. Therefore, it is assumed that these change requests were raised due to the client company’s operational feedback and changes from the internal and external environment.

Sources/ Techniques of Data Collection

‘Evidence for case studies may come from six sources’ (Yin, 2003). These sources are: documentation, archival records, interviews, direct observation, participant-observation, and physical artefacts. For purposes of this case study, documentation, interviews, and direct observation will be discussed below. As per Yin (2003), there are some overriding principles to any data collection effort in doing case studies. One such principle is the use of multiple sources of evidence that converge on the same finding. The second principle is a case study database which is evidence distinct from the final case study report. In Research Framework A the data collection ended by consolidating and recording the collected maintenance change requests into a comprehensive and coherent dataset. The third principle is the chain of evidence that links the research questions to the data collected and the conclusions drawn. These principles have a bearing on all the sources used in this research.

Yin (2003) asserts that strengths and weaknesses exist for the sources of data collection, but a good case study will adopt most if not some of these sources. Although it is a useful comparison to make, it should be noted that no single source has a complete advantage over all the others, but serve to complement each other. In this case study the sources/ techniques used were documentation, direct observations and interviews. The documentation included formal and informal records of maintenance change requests, for example written reports, meeting minutes, emails or formal change request documents. Direct observations were made by making field visits to the case study site and were made by observing formal meetings and impromptu telephone calls regarding change requests that the client organisation would discuss with the development team. As the researcher directs set questions pertaining to change requests at the development team, interviews are also said to be a reliable source of evidence according to Rubin and Rubin (1995). Focused interviews were conducted by the researcher (Merton at al., 1990; Silverman, 2001) where set questions were posed to the development team in terms of what change requests they had received on a particular day. The data collection activities produce quantitative data results and the techniques of data collection may be qualitative, but the overall findings of the case study will be the convergence of them both (Gross et al., 1971).

The second stage of Research Framework A involves the change requests that have been collated from the data collection process being filtered and classified into Fitzgerald et al.'s (1999) categories. This is referred to as the 'categorisation process' of each individual change request. The 'areas of change' categories formulated by Fitzgerald et al. (1999) are used as the 'literature-based' model in which the maintenance change requests from the case study are categorised. The categories identify several different areas of IS's and can serve among others (Longworth, 1985) as a model for the categorisation process in this research. These categories are selected as they relate to the study of areas of change, which forms the basis of research question one and because they incorporate recognised components of any IS.

Fitzgerald et al. (1999) identified and defined the areas of change as being: 'Data' - new/ amended fields in files or databases; 'Processes' - logic/ code changes or new procedures; 'Interface' - user/ systems interfaces; 'New/ Upgrade Packages' - software releases or patches to render services and 'Other' - miscellaneous category of change requests that do not fit distinctly into the above categories (Table 4.1, Chapter 4).

The categorisation process involved the researcher conducting an exercise on individual change requests of assigning each to an appropriate category. Change requests from the dataset are filtered and classified as per the definitions stated in Fitzgerald et al.'s (1999) categories. The interpretation involved the researcher identifying key aspects of the change request that were affiliated to these category definitions. For example, the details of a change request (from any source of evidence, documentation, observation or interview) would be analysed and where it perhaps refers to a system function, the researcher would allocate it to the category of 'Processes'.

After the initial categorisation process, the process was repeated by an independent third party. A random selection of data was selected and the categorisation process was re-run by a fellow researcher who had no previous or subsequent knowledge or involvement in the case study. The purpose of the independent researcher's activity was to validate the categorisation process, and also to verify the researcher's results from the process. In the case of any discrepancies further evidence and opinions in regards to the change request was gathered from the development team. In such instances a consensus was required before assigning the change request to a category.

In terms of chronology it was deemed that the appropriateness of Research Framework A to answer research question one was unsuitable. The researcher reveals, a-priori, that Research Framework A had to be suspended at stage two. This was due to the categorisation process producing a significantly different result when applied to a WIS context. The reasons behind this decision are further elaborated on in Section 3.4 and Chapter 4.

Had Research Framework A proceeded to the next stage, the next step would have been conducting a survey. The purpose of such an exercise would have been to overcome the limitations on generalisability. For the purposes of this research, the survey strategy of Sarantakos (2005) would have been utilised. A survey is a research strategy rather than a method or technique (Bryman, 1989). Because of the ubiquity of surveys, it is likely that a good appreciation would have been needed of what the term means. The purpose of using a survey for Research Framework A would have been to collect information pertaining to areas of change in a standardised form from a relatively large number of participants, thus attaining a cross-section of change that was occurring in other WIS development projects. The selection of samples would have been from known companies.

Stage four of Research Framework A would have addressed research question two by designing a conceptual flexibility model that would facilitate the implementation of flexibility in specified areas of the WIS. Stage 5 would entail the use of an action research strategy as prescribed by Baskerville and Pries-Heje (1999). The purpose of doing action research would be to test the flexibility model in a live environment. This step would have been iterative and thus repeated until the flexibility model was refined. The penultimate stage (Stage 6) of Research Framework A produces a modified flexibility model that identifies what area of a WIS could be made more flexible to change and how it could possibly be achieved.

The next section will discuss in further detail why Research Framework A was found to be ultimately inadequate as a research approach for this thesis investigation.

3.4 The Changing Direction of the Research and its effects upon the Research Design

The analysis conducted after the completion of stage 2 revealed a large discrepancy between the findings of Fitzgerald et al.'s (1999) original study and the findings of the categorisation process of Research Framework A within a WIS context. Specifically stage 2 revealed that almost 50% of change requests were allocated to the Data category as opposed to only 15% in the original study. Additionally, clustering patterns formed within the Data category suggested that emergent sub-categories were becoming evident.

As stated in Section 3.3 the core objectives of Research Framework A are listed below. The decision to change the direction of the research was based on the premise that the first two objectives were only, and could only be partially fulfilled using Research Framework A.

- Ascertain areas of the WIS that are susceptible to change (an exploratory approach).
- Confirm whether WIS maintenance change can be studied by applying IS 'areas of change' categories (Fitzgerald et al., 1999) into a WIS context (a confirmatory approach).
- Determine how WIS maintenance change can be facilitated through designing and applying a flexibility model.

Objective 1: *Ascertain areas of the WIS that are susceptible to change (an exploratory approach).*

The exploratory process undertaken within this case study and the techniques adopted for the collection and preparation of the change request data was proven to be valid (Chapter 4). It was also confirmed that the use of Fitzgerald et al.'s (1999) areas of change categories are also suitable in context of this research. The question of unsuitability arises as the Data category appears to be too broad in its definition. As stated before a large proportion of change requests were labelled in this category (Data) and clearly upon further analysis indicated that subcategories could and should be formed.

Objective 2: *Confirm whether WIS maintenance change can be studied by applying IS 'areas of change' categories (Fitzgerald et al., 1999) into a WIS context (a confirmatory approach).*

Research Framework A adopted an overarching exploratory philosophical research approach, as well as applying a confirmatory approach within the categorisation process. From a confirmatory perspective the usage of Fitzgerald et al.'s (1999) AoC model to investigate WIS areas of change was discarded. As mentioned above, due to the emergence of WIS areas of change, a more granular/ bottom-up method of investigation would be relevant as a research framework.

The changing direction of the research was unforeseen at the outset and it is also recognised as being unconventional in terms of Ph.D. thesis formats. However it was a necessary process to undergo and report on, in order to explain the logical progression required to this investigation. Myers (1997) purports that all research has its source in some latent assumption about what constitutes valid research and which research methods are congruent with that belief. The process by which an investigation is conducted is elementary to the soundness of the research question(s) itself and how it is substantiated empirically. Eisenhardt (1989) further asserts that research objectives change during the course of the research and this actually transpired as a result of Research Framework A's categorisation process. The validity of the results of a research is contingent on the epistemological stance taken, as well as the general acceptance of a set of values that have knowledge claims which have withstood the test of time within the larger community (Klein and Myers, 2001). Smith (1990) asserts the view that '… meaningful research demands a sound epistemological base to the research methods'. Thus the philosophical perspective and the research strategies are dependent on the context of the problem being investigated and the research objectives.

Although Research Framework A was ultimately suspended, a important conclusion can already be drawn regarding the nature of traditional IS maintenance change when compared to that of WIS maintenance change. Using the same analytical technique of the AoC model, studying the same phase i.e. the maintenance phase, the results yielded suggests that a distinction exists. This empirically supports the argument within the debate (Section 2.2.1) that IS is different to WIS in context of maintenance change.

It was recognised that a redesign of the research framework was required which allows for the categorisation process to examine the emergence of WIS AoC categories. Subsequently, this resulted in an emphasis shift within the remits of this research to further explore the nature of WIS maintenance changes occurring. It was deemed that the more that is known regarding the characteristics of change the greater the efficacy of any potential flexibility framework will be. Due to this refocus, research question one was supplemented with the following sub-question: “What are the characteristics of change?”. Additionally, the second research question related to flexibility was diminished in significance, and was considered at a conceptual level as opposed to a tried and tested framework.

The following section pertains to the design of the new research framework.

3.5 Designing a research framework: Research Framework B

Research Framework B was designed on the premise that the study of change is contingent upon the context in which it arises. IS research often entails contemporary phenomena emerging in business enterprises that have software system structures that are in a continual state of flux. Consequently, this framework takes qualitative and exploratory approaches that are able to absorb the intricacies of a contextualised phenomenon. Change is a social and contextual phenomenon induced by external and internal environmental factors that can result in dynamic, spontaneous and evolutionary effects on software systems. The environments that surround software systems are fast changing and becoming more turbulent in particular for systems such as WIS’s, reasons for which are given in Chapter 2. Many studies have addressed this relationship from a management organisational context (Galbraith, 1977; Mintzberg, 1979, 1981, 1983). The requirements for a IS qualitative research approach is based on the reason that they should be able to reflect the dynamic nature of the contextual social systems and in relationship to its IT provision, in this instance with the added variable characteristics of change, specifically change requests. Therefore Research Framework B and the selected research approaches that form its constitution, attempts to accommodate the research questions and the sub-questions in a ‘ground-up’ manner, where issues and themes emerging from change are captured.

This section begins by describing the redesign of the case study to embedded multiple units of analysis as opposed to the holistic analysis that governed Research Framework A. A process to identify and emerge ‘areas of change’ categories was formulated based on clustering and pattern matching recognition techniques (Miles and Huberman, 1994) that was used to develop change group categories. For purposes of validation a new categorisation process was designed that verified these newly defined group categories. As discussed in Section 3.4 a greater understanding of the characteristics of change was required to gain a holistic perception of the phenomenon, and on this premise constructs of analysis were developed and discussed. The concluding part to the chapter shows diagrammatically the overall processes enveloping both research frameworks.

3.5.1 Embedded units of the Case Study

Within the single case study there may be incorporated subunits of analysis, so that the design becomes more complex or embedded. These subunits can create greater levels of extensive analysis that not only enhance the insight into the case study but allow for a greater pool of data. The organisation that was studied in Section 3.3 remains as the case study for purposes of Research Framework B, however the units of analysis specifically consists of three projects and one validation project within the remit of a single case study. The research design that was implemented in this framework was that of Yin’s (2003) single-case embedded design (Figure 3.2).

An embedded design can often be an important device for focusing a case study inquiry. However, an embedded research design has its weaknesses too. If too much attention is placed on the subunits and the larger and holistic issues in the case are ignored, the case study itself will change in nature and will run the risk of shifting its orientation. If there is any shift then they must be addressed or justified in context of the original inquiries (Yin, 2003). In order to mitigate against this, contextual issues for each embedded unit are retained and recognised, however ultimately all the data from the embedded units are consolidated before analysis.

This research design was used as a result of the outcome of Research Framework A, because the embedded units of analysis (the WIS projects) would produce a wider dataset. A wider dataset (greater change request pool) that was taken from the various projects would have the affect of diminishing anomalies and clarifying trends that arise from the analysis of change requests. The results may be skewed by context, if the data is only from one development company, or one type of technology, or one client or one project within a case study. However, this is addressed by establishing a wide-ranging dataset that balances context with consolidated data pertaining to change, thus promoting generalisability within the boundaries of this research.

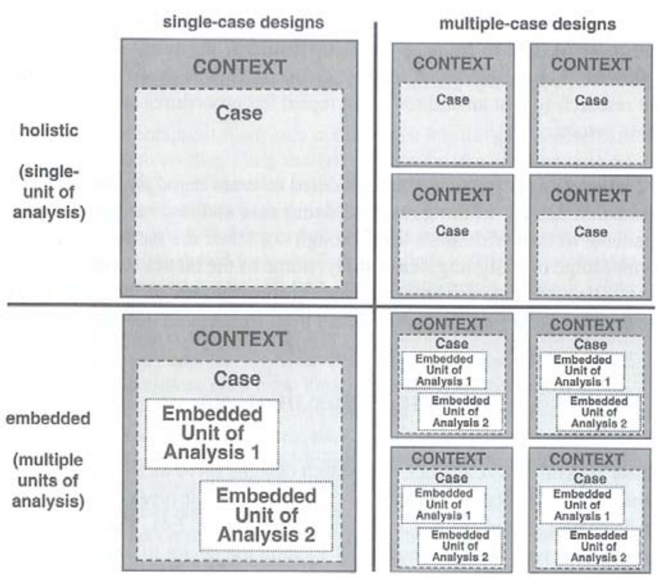

Figure 3.2 – Basic types of design for case studies (Yin, 2003)

3.5.2 The Process of Emerging Categories

A bottom-up strategy was taken to develop a research process whereby areas of change categories are allowed to emerge. In a bottom-up approach the individual base elements of the system are first specified in great detail. This stage of the research approach was adapted from Strauss and Corbin (1998, p.206) 'Open coding' approach to 'discover, name and categorise phenomenon according to their properties and dimensions... providing the greatest opportunity for discovery.'. The elements are then linked together to allow the emergence of larger subsystems, which then in turn are linked, sometimes in many levels, until a complete top-level system is formed (Bowker and Star, 2000). This strategy often resembles a 'seed' model, whereby the beginnings are small but eventually grow in complexity and completeness (Xu et al., 2007). This approach is also sympathetic with the semiotic type analysis (Feldman, 1995), whereby 'identifying signs and understanding the processes by which they come to have meaning' can be clustered into connotative meanings and institutional concerns (Feldman, 1995, p.22), whereby change requests are indicators for signs of significance.

The bottom-up approach will form the basis on which the emergent process will operate, where the low-level units of analysis are the areas of change, which permits the emergence of higher level categories. Briefly summed up, 'emergence' in this context relates to how complex systems can be organised, without any predefined direction or definition. Individual units of areas of change are 'self contained' without knowledge of any overarching aim or scheme, but out of this change 'data', order, pattern and categories emerge. A classic emergent system brings self-organised order to category building. It allows for patterns to emerge (Miles and Huberman, 1994) and gives scope to a 'creative leap' (Mintzberg, 1979). Categories are able to emerge out of the data and the approach has to be allowed to 'unfold, cascade, roll, and emerge' (Lincoln and Guba, 1985). From this viewpoint, the researcher developed a 'process of emerging categories' that can be seen in Figure 3.3 and which will be further explained below.

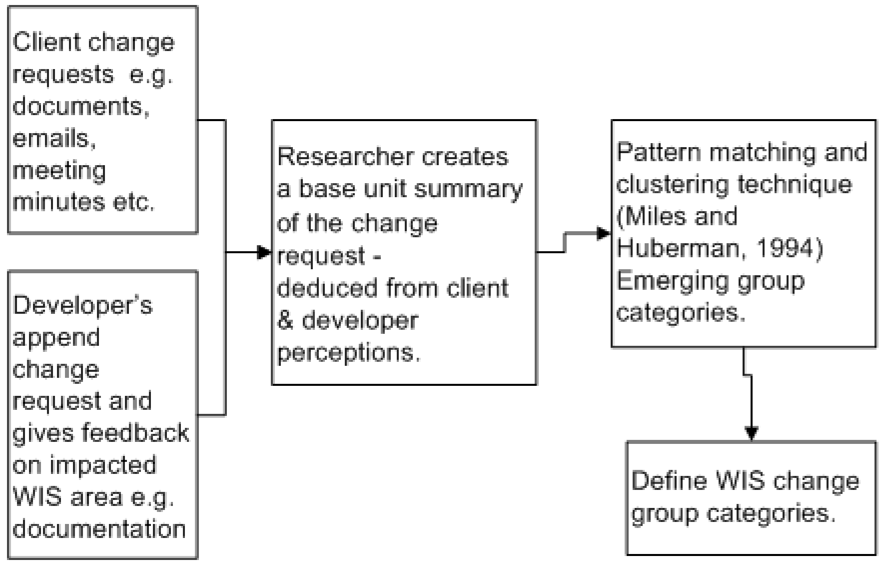

Figure 3.3 – Process of emerging categories

As with Research Framework A the same methods were employed for the data collection of change requests, however, in Research Framework B the researcher collates additional information to gain a deeper understanding of each change. Select information (See Section 3.5.4) retrieved from the data collection process was prepared and tabulated (examples shown in Chapter 4). The process of emerging categories begins with the researcher extrapolating the client company's and developer's interpretation of the area of change, as it is acknowledged in IS literature that both parties may have varying perspectives (Howcroft et al., 2004; Doherty et al., 2006). The techniques employed to acquire these interpretations, was to analyse the client company's change requests and by analysing developer feedback that may reveal what areas of the system have been impacted by development team actions. Where this information was not apparent from the documentation, the researcher directed questions in focused interviews with development team members. In this thesis the researcher ensures that their own perception of the change request base unit is validated through the convergence of both client-company's and developer's perceptions of the change request (as illustrated in Figure 3.3).

The researcher’s change request base units are then clustered and pattern-matched to develop higher level group categories as per a bottom-up approach. Miles and Huberman (1994) explained that identifying patterns from data and drawing some sort of understanding of a phenomenon is such a common human practice that ‘we all are well capable of doing it’. This endorses a natural ability to coordinate data in recognising patterns and through practice, develop a pattern-matching or textual analysis technique. Academics state that what is meant by ‘textual analysis’ is the process whereby the meaning of ‘texts’ is elicited and comprehended through the researcher’s interpretation (Boland, 1985; DeSanctis, 1993; Kanellis, 1996; Silverman, 2001). The pattern-matching technique (Miles and Huberman, 1994) can be used to identify similar characteristics between areas of change (in the dataset) in order to cluster them into group categories. Yin (2003) explains that the pattern-matching technique is the most desirable strategy for case study analysis and in particular, it strengthens the internal validity of the case study. He argues that in using the pattern-matching technique, if the pattern is found, it validates the prediction and if the pattern does not fit, it allows the modification of the prediction. Where a pattern is found and a group category created, it confirms the validity of the bottom-up approach.

3.5.3 The Validation Process

The process of identifying the emerged group categories was conducted once the data for all three embedded units of analysis (WIS projects) had been collected. This data was then consolidated into one aggregated dataset which underwent the process as shown in Figure 3.3. The fourth embedded unit of analysis serves to act as a validation control within Research Framework B. The validation exercise aims to ascertain whether the research had reached a saturation point with respect to identifying new group categories, thus using a confirmatory approach. After a trial period of data analysis the process was further refined by using a definition based method that was used to identify change group categories. Group categories can either emerge or existing ones can be confirmed from the emergent approach which had been applied to the first three embedded units of analysis. If no new group categories were found, this serves as a distinct cut-off point for the investigation. In the instance that new group categories were identified then the fourth embedded unit of analysis would be consolidated with the first three units of analysis and subsequently validation would be attempted with a fifth embedded unit of analysis. This process would continue until the saturation point would be achieved. It is however recognised that all research is subject to a delicate balancing act of achieving its objectives with estimating what would be a reasonable cut-off point for the investigation. The validation process is shown diagrammatically in Figures 3.4 and 3.5.

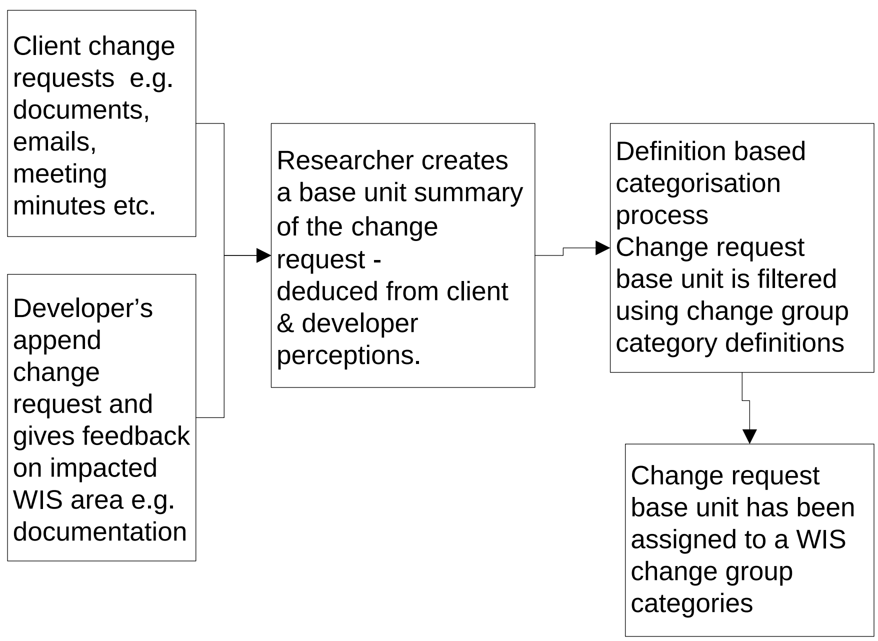

Figure 3.4 – High level view of the validation process

3.5.3.1 The Categorisation Process - Validation

The validation process is designed to effectively assign change request base units to an appropriate WIS change group category; the process is depicted as a flow diagram (Figure 3.5). The flow diagram is based upon the use of the WIS change group category definitions that have been formulated by the use of the emergent process as pre-conditions to assigning change request base units from the validation dataset to WIS change group categories permanently. The definitions of the WIS change group categories are described in Table 4.9 (Chapter 4).

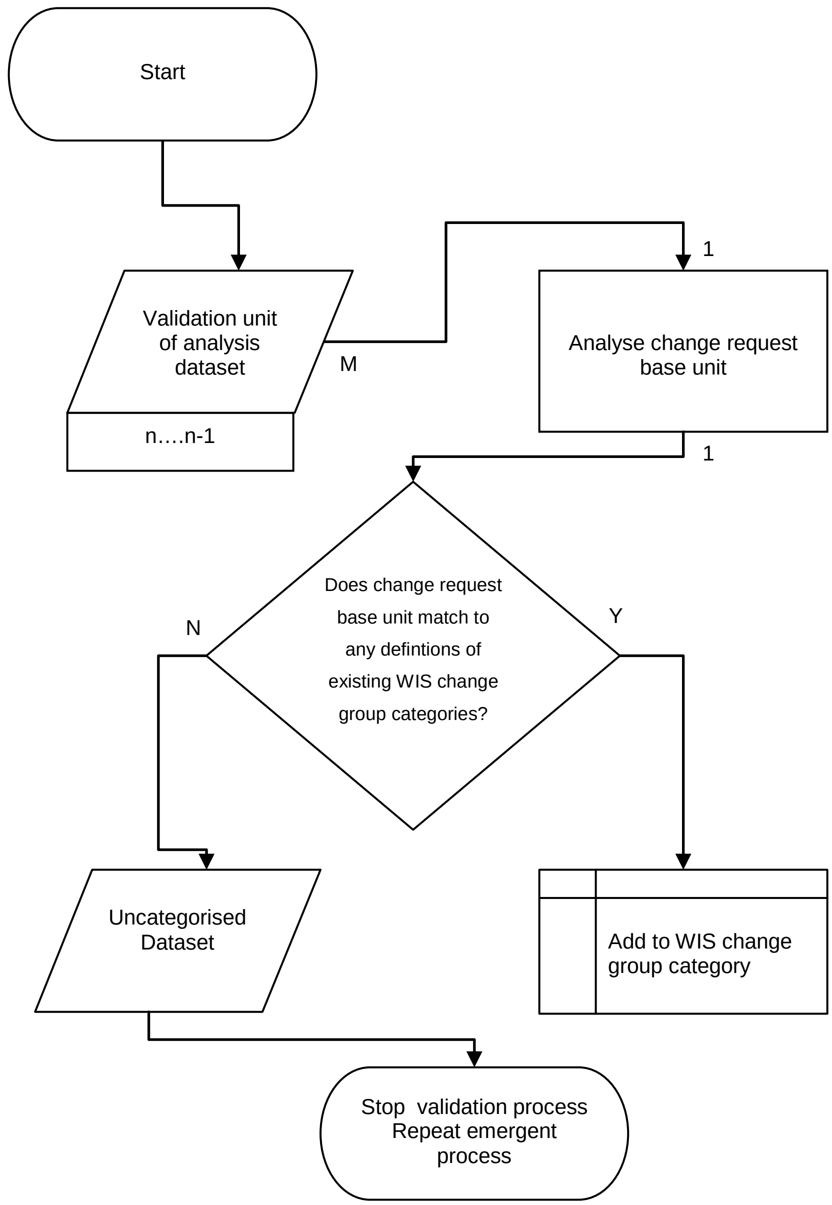

Figure 3.5 – The categorisation process: Validation

The process of the flow diagram is explained in the steps below:

- A change request base unit is taken from the validation dataset.
- The change request base unit is examined by the researcher. Additional documentation and feedback consulted if further information is required.
- The researcher accesses whether the change request base unit map to any existing WIS change group category. If the question is answered in the affirmative, then the change request base unit is placed in the relevant WIS change group category.
- Where the change request base unit cannot be assigned to a WIS change group category, this will indicate that the emergent process has not been saturated with all WIS change group categories. In this instance the confirmatory nature of this validation process would have been deemed to have failed. Subsequently the validation is terminated and all data from the validation embedded unit of analysis is added to the existing consolidated dataset pool. This pool is then subjected to the emergent analytical process to reveal the additional change group category(s).

In the next section the constructs of analysis will be explained in order to gain a deeper understanding of the phenomena of WIS maintenance change, which was the driver for the second sub-question to research question one.

3.5.4 Constructs of change

A conclusion from the results produced from the operation of Research Framework A, was that a greater degree of analysis of each change request was required. The additional depth and perception of change requests is based on three constructs that are used as parameters to collect data and analyse WIS maintenance change further. A combination of existing metrics (Chapter 2) are used within this investigation to formulate a triangulated analysis framework (Section 3.5.5). In the opinion of the researcher this novel approach utilises the three most relevant analytical constraints to study the nature of WIS maintenance change. The constructs of change used as part of the design of Research Framework B are, ‘complexity’, ‘form’ and ‘context’. The constructs are discussed below in relation to the dimensions that constitute them.

3.5.4.1 Form of change

The Form of change is defined in this research as being the cause or motivation as to why the maintenance change request emerged. A common perception as to why change requests arise within the maintenance phase as modifications to the software system can be summarised as (Canning, 1972; Lientz and Swanson, 1980; Reutter, 1981; Swanson and Beath, 1989, 1990; Boogaard, 1994):

- The non-compliance of the software system to original specifications due to residual errors or failures. It is a modification that is initiated by defects in the software system that is often referred to as bug-fixing.
- As a consequence of internal and external environmental induced changes in which the software system operates. The changes can occur in the data or processing environment.
- The expansion of the software systems scope to be inclusive of new requirements that were not perceived of during the early development phases of the system lifecycle. However, it is noteworthy to mention that new requirements that do emerge during the development phases often get propagated to the maintenance phase.

Takang and Grubb (2003) refers to the motivation factors as being; to support continuity of service, to support mandatory upgrades, to support user requests for improvements and to facilitate future maintenance work. To facilitate future maintenance work specifically relates to the maintenance change type of preventative (Chapter 2). They further explain that in practice the classifications of software changes are often intertwined but also emphasises that ‘a good understanding of the distinction between them is important’.

In an early paper on the subject of software maintenance, Swanson (1976) identified three types of maintenance work that are defined in Table 3.3. This typology has been consistently used and is the foundation for several subsequent empirical studies (Lientz and Swanson, 1980; Reutter, 1981; Boogaard, 1994, Fitzgerald et al., 1999). They have become widely accepted among practitioners (Martin and Osborne, 1983) and regulatory societies such as the IEEE and British Standards (IEEE Standard, 2006; BSI Corporate, 2006). Section 2.3.1 alluded to the discussion that several authors have endeavoured to further classify the type of maintenance change, resulting in detailed taxonomies. However, their contributions largely disseminate from and are based on the research activities of Swanson (1976, 1999). The basic criteria of the types of maintenance change according to Swanson (1976) was applied in context of WIS's for the purposes of clarity and comparative analysis (IS versus WIS). The three types of changes established by Swanson (1976) are considered as the dimensions to the 'form' construct of change established within this research. Their types of changes are extended to a WIS's context to provide additional empirical evidence in this area.

Form	Definitions
Corrective	Performed in response to processing, performance, or implementation failures. Emergency fixes and routine debugging are characteristics.
Perfective	Performed to eliminate processing inefficiencies, enhance performance, or improve maintainability. Responsive to the evolving needs of its users and ongoing business support.
Adaptive	Performed in response to changes in the data and processing environment. Responding to changes in the environment in which the IS resides.

Table 3.3 – Types of maintenance work (Swanson, 1976)

3.5.4.2 Complexity of change

Defining the complexity of IS's has often remained a contentious and difficult task. Therefore no generic IS definition for the term 'complexity' can be used as its semantic intent and measures are relative to the level of perception and context of its application (Lehman and Belady, 1985, p247-274). In-situ of software maintenance, complexity is often perceived in relation to coding and programming practices (Section 2.3.4), for which numerous algorithmic measures have been proposed in the literature (Takang and Grubb, 2003). For the purposes of this research the complexity of change requests are considered by abstracting relevant information to the analysis of WIS maintenance change. Thus, this research seeks to gain a diagnosis of WIS maintenance change by evaluating it according to the dimensions of ‘effort’, ‘structural layer’ and ‘ripple effects’.

The effort dimension relates to complexity as it can be assumed within reason that change requests that take a considerable amount of time to resolve have an increased complexity. For example a change request that takes one hour to resolve can be considered as having fewer complexities when compared to a change request that may take four hours to complete. Therefore the size of a change request can be determined by the effort dimension. The distribution of the effort dimension can provide additional information when considering areas of the WIS that are more costly or time consuming to resolve. However, the assumption behind the relationship between effort and complexity is that all developers have the same skill levels. It is realised that this may not always be the case, nevertheless it is assumed within reason that organisations hire specialists with a degree of industry knowledge.

WIS’s are typically built using the distributed client-server architectures in which the user interface, functional process logic and data storage components are developed and maintained as independent modules (often as separate platforms). Each component represents a layer and is interchangeably referred to as tiers. The software platform and design pattern of WIS’s in a client-server infrastructure environment are often founded on the basis of a 3-tier architecture model (Orfali et al., 1999), however this archetype has been further segregated into other classifications (Fowler, 2003). The segmentation of the above processing components is specifically designed as distinct procedure layers for the purpose of reducing complexity of the system through the provision of modularity (Orfali et al., 1999). The architecture layer dimension was considered to be the logical structuring mechanism for the elements that make up the software system. WIS maintenance change requests can be further isolated to the layers they affect and when used in conjunction with the effort dimension, the complexity of each layer can be deduced in regards to the types of changes that are impacting the layers. The architectural layers in context of WIS's are described as:

- Presentation layer

  This is the uppermost level of the software application. The presentation layer/ tier displays information related to services on the client side. It communicates with other layers/ tiers by outputting results to the browser and all other tiers in the network. The presentation layer is a front end web browser that presents content and in some instances embedded functionality (JavaScript).

- Application (System)/ Business-logic layer

  The application (system) or business-logic tier is separated from the presentation tier as its own layer, and its purpose is to control an application's functionality by performing detailed processing. The application (system)/ business-logic layer is a dynamic content processing layer of the software system - application server.

- Data layer

  This layer consists of database servers and it is where the data and information of the software system is stored and retrieved. This layer/ tier keeps data independent from application servers or business-logic. Giving data its own tier also improves scalability and performance and comprises of both datasets and the software (database management system) that is used to manage as well as provide access to the data.

The ripple effect dimension is derived from the maintenance literature that provides frameworks and metrics that examine the impact of implementing changes into the software system (Chapter 2). Ripple effects are included as a distinct vital stage in these frameworks and represent a complexity measure that is analysed algorithmically (Yau et al., 1978). The term ripple effect was used to describe the way that a change to one aspect of the IS would necessitate subsequent changes to other parts of the IS. The benefits of accounting for ripple effects are stated by Black (2001) as being:

- A means to help maintainers by indicating IS components which may cause concerns in the course of the maintenance process.
- To show the maintainer how great the effect of a change will be on the rest of the IS.
- To help maintainers to determine what the impact will be on the rest of the source code once those changes are made.

WIS's are inherently highly integrated as a result of the dependencies among system components, thus accounting for the cascading affects of change requests on subsequent parts of the system are an important issue to consider within this research. Ripple effects of WIS maintenance change requests are included as a dimension to the construct of complexity to gauge a better understanding about the characteristics of WIS maintenance change. However, it is noteworthy to mention that this investigation does not intend on using or developing an algorithmic calculation to analyse ripple effects, but will rather use a qualitative measure to account for instances and details of ripple effects.

3.5.4.3 Context of change

The construct of context is a pertinent attribute in understanding the surrounding issues encompassing the WIS maintenance change request phenomenon. In particular the contextual issues help in describing or informing the conditions in which the phenomenon is involved, or in planning interventions (Watson and Wood-

Harper, 1996; Howcroft et al., 2004; Doherty et al., 2006). The planning of interventions has a particularity in addressing the second research question relating to the flexibility solution. Through the recognition of the wider environmental conditions a better understanding may be achieved to explain trends and anomalies in the results produced by the research. Change requests are a socially driven phenomenon that emerge as a consequence of either internal or external environmental factors (Pettigrew, 1985). Therefore change requests are considered to be contingent on the context in which they arise and cannot be taken on face value or viewed in isolation. However, accounting for all the contextual conditions is a difficult task and thus value judgments need to be made. For the purposes of this research, issues such as the business case context, change management procedures, project management and technological features are determined to be key facets surrounding change requests.

3.5.5 Construct relationships within the Analysis Framework

The constructs discussed above formulate the framework of analysis to investigate the characteristics of WIS maintenance change. This framework is depicted in Figure 3.6 and was used as the basis for discussions and presenting results on the nature of WIS maintenance change (Chapters 5 and 6). Each construct can be seen to form an interconnected trinity relationship that encompasses the centric issue of WIS maintenance change. The triangulated structure is used as a model to view the WIS change group category dataset from the three construct perspectives and their associated dimensions. Pettigrew (1985) proposed a similar triangulation based analytical framework to investigate change within an organisational context. The key contribution of the framework is that it provides a three way perspective on the central issue of change. Each peak (Content, Context and Process) and its interrelationship with associated peaks emphasise that change should not be analysed by only focusing on one or two perspectives but rather a multi dimensional view is required. This appears to be a valid and constructive method and thus was another reason for the adoption of the triangulated framework.

The interrelationships between the constructs and their associated dimensions are used to provide additional analytical data results on the subject of WIS maintenance change (discussed further in Section 5.3). In summary the key interrelationships explored are:

- The Ripple effects crossed with Architecture Layer dimensions.
- The Form dimensions crossed with the Architecture Layer dimensions.
- The Effort crossed with Ripple effects.
- The Effort crossed with Architecture Layer dimensions.

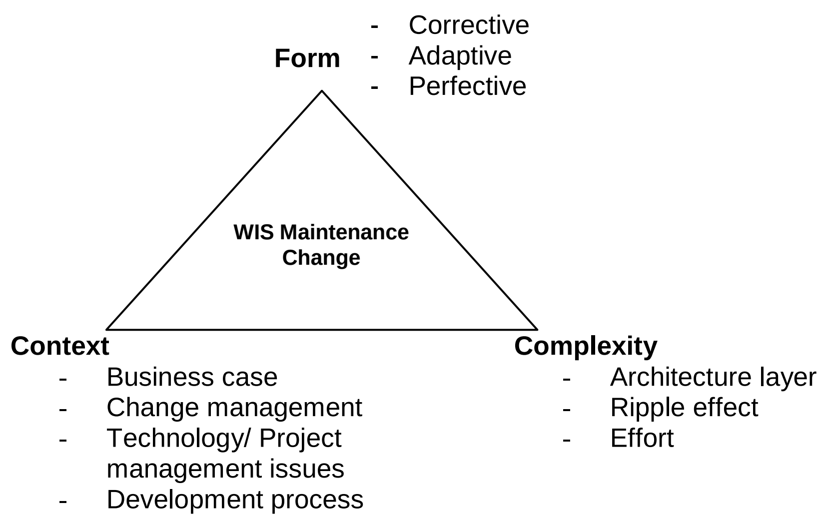

Figure 3.6 – Construct based analysis framework for WIS change

The main contribution of the analysis framework is that it attempts to depict the interrelationships between the three constructs and its dimensions, emphasising that change should not be analysed by only focusing on any one aspect. However, through analysing the intricate relationships, an enriched insight was achieved that substantiates the goal of understanding the nature of WIS maintenance change. The three constructs were selected from identifying key issues within the academic literature and from the opinions of industry professionals. It is worth noting that there are several taxonomies and perspectives when investigating change, however, these attributes often have a specific focus and are contextualised within the study (Kanellis, 1996; Buckley et al., 2005).

The results produced from the analytical examination of the dataset, provided both quantitative and qualitative results. In order to produce quantitative results, frequency analysis was performed to obtain an insight into how often a certain change feature occurs. Other statistical analysis tools such as SPSS and Matlab could have been adopted, however basic percentages were used to identify initial key trends and patterns emerged from the dataset.

3.6 Conclusion

This chapter described the case study strategy that was selected on the basis of Yin's (2003) three principles of choosing an appropriate research strategy and as a means by which change request data could be collected.

Research Framework A was designed on the premise that maintenance change in WIS could be studied from an IS perspective by extending 'areas of change' (AoC) categories as espoused by Fitzgerald et al. (1999) into a WIS context. This was done in order to address an existing issue amongst academics that there are either similarities or differences between IS and WIS (Section 2.2.1). Findings from Research Framework A suggest that within the context of maintenance change, differences certainly do exist between IS and WIS. The underlying philosophical research approach of Research Framework A was both exploratory and confirmatory in design. It was exploratory due to the fact that areas of the WIS impacted by change needed to be determined. The confirmatory aspect of the framework was due to the applicability and validity of the areas of change categories within the WIS and IS contexts.

The early analytical results from Research Framework A showed a large difference in the Data category between the WIS and IS contexts. It was evident that the Data category in the WIS context could be decomposed into sub-categories, however the limitations of the design of Research Framework A constrained this type of investigation. In order to attain the necessary level of granularity required to study WIS maintenance change, it was recognised that the research framework needed to evolve from its initial design. This was the key driver for changing the research framework from A to B, which used a bottom-up discovery mechanism. As a result of this, there was a shift of emphasis from research question two to one where a in-depth understanding of the nature of change was required. It was for these reasons that a research sub-question was added to research question one - “What are the characteristics of change?”.

This emphasis shift resulted in a re-conceptualisation of the new research framework (Research Framework B) in which the study of change became a dominant feature when compared to flexibility. Research Framework B incorporated both an exploratory and qualitative research approach and is based on the process of identifying emerging categories through the utilisation of a bottom-up approach. A bottom-up approach allows for areas of change and their relative group categories to emerge in an organic manner, where change requests are not derived from predefined definitions. The study of the nature of change in WIS maintenance was further enhanced through the formulation of analysis constructs namely; ‘form’, ‘complexity’ and ‘context’. Each construct provided firstly, an insight into an aspect of the inherent characteristics of change and secondly, provided contextualised information that enriched the scope of the dataset.

What is often required in IS research is a dialectic process that has a multi-disciplinary research approach that efficiently achieves the aims of the research. This research adopted a mixture of approaches and strategies across both research frameworks in order to achieve the goal of cogently answering the research questions. From the plurality of research strategies, approach, techniques and methods that exist in IS research, this investigation was able to change and thus endorse a more relevant research framework (B), that took a emergent, bottom-up approach to identify areas of change and relative group categories. It was Research Framework B that provided the necessary empirical data used for analysis and discussion within this thesis. The following chapter discusses the operation of both Research Frameworks A and B, however, in the case of Research Framework A the presentation of the results and analysis are also shown.

An overview of the research design process that is inclusive of both Research Frameworks A and B activities of this investigation can be seen in Figure 3.7.

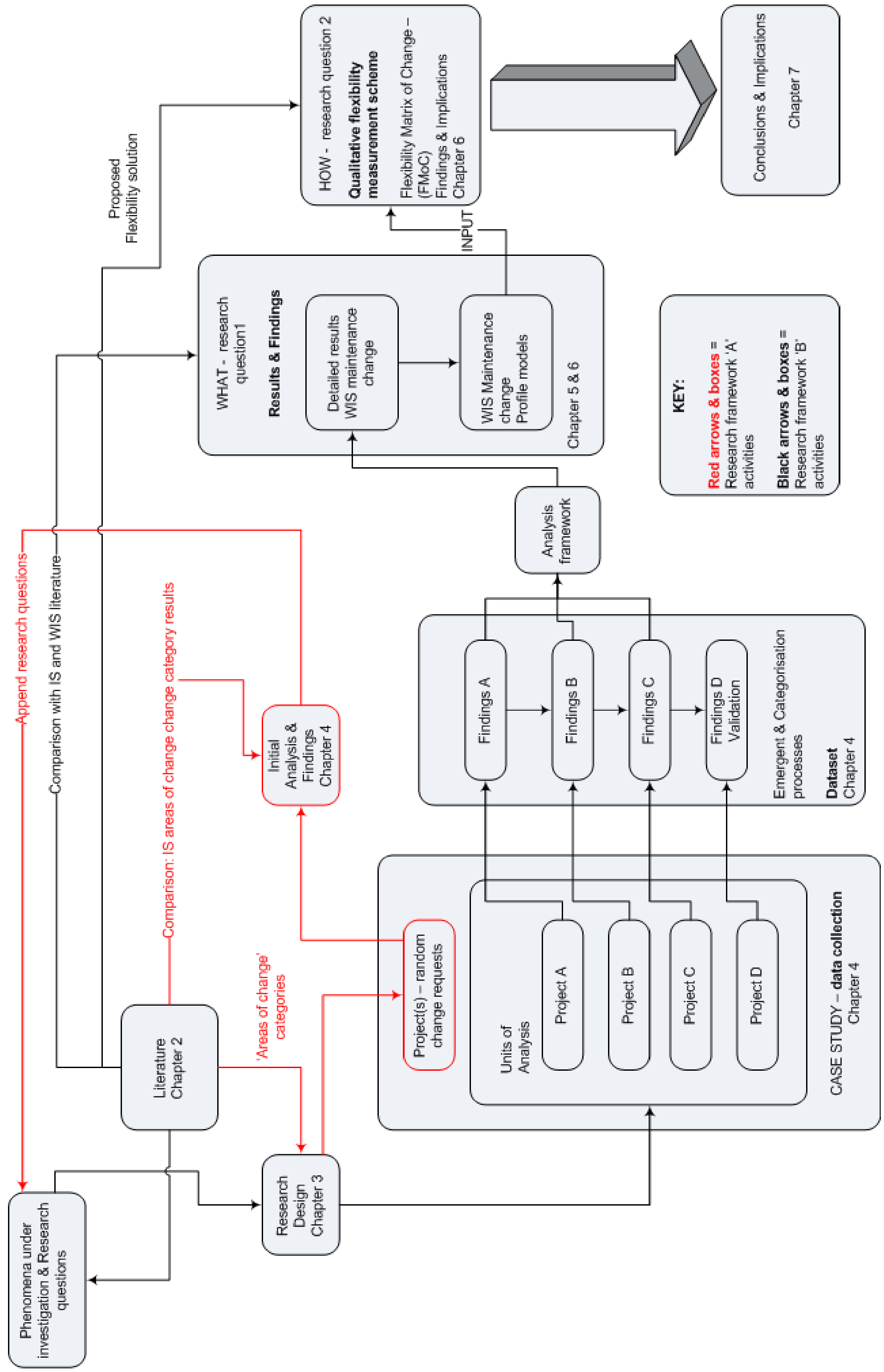

Figure 3.7 – Research Process Overview

Chapter 4 ▪ Operationalisation of the Research Framework

4.1 Introduction

This chapter demonstrates both the research frameworks in their operational state. Research Framework A's objective was to be able to firstly, collate maintenance change requests and secondly, to analyse them using the 'areas of change' (AoC) categories developed by Fitzgerald et al. (1999). Early use of these categories within the case study indicated that it would not be sufficient enough to address changes within the realm of WIS maintenance. Due to the observations made, new categories of areas of change were deemed necessary to encompass the changes in WIS post-development. The research perspective changed and the research framework evolved accordingly to allow WIS categories of change to emerge.

The initial sections of this chapter will focus on Research Framework A, associated processes and its relevant analysis. The latter part of the chapter will concentrate on the evolution of Research Framework B. Section 4.2 will provide a profile of the case study. Section 4.3 addresses the implementation of Research Framework A, that had the dual purpose of confirming previous IS maintenance investigation methods as prescribed by Fitzgerald et al. (1999), as well as to explore the nature of WIS maintenance change. Section 4.4 will contextualise the embedded units of analysis within the case study by presenting the profiles of the WFO (P1), ABN (P2) and LJO

(P3) projects. It also reflects the changes made to the research approach within Research Framework B based on a bottom-up category building approach. Lastly, Section 4.5 concludes and summarises the chapter.

4.2 The Case Study

The selected case study was carried out within the IT-development division of a small to medium sized enterprise, namely Best & Co. All empirical data pertaining to the thesis was collected from this case study source. Best & Co. is an established web services development and consultancy provider with over fifteen years experience in the design and production of on and offline marketing collateral, with a specialist focus on the niche market of financial and corporate commercial sectors. Its core business operation is to act as a development consultancy that provides solutions for organisations requiring professional design and integrated technical solutions for corporate brochures, annual reports, identity products and corporate literature. Best & Co. has an extensive client base that is growing in size partly due to a market demand of web-based resources. Due to audit and compliance requirements from end clients, providing reporting via the medium of the web has formed an important sub-niche of services that Best & Co. offer within their proposition. A substantial portion of their client base are shifting from providing traditional paper based reports, to an on-line multimedia facility.

Best & Co. employs IT specialists and have the capability to offer design, development and maintenance of bespoke web-based solutions. These include intranets and extranets, transactional capabilities, e-promotions and interactive online communication tools. They also offer consultancy on site hosting, support, management and site marketing.

As a multi-disciplined team working in conjunction with each other, they tender solutions that work equally well across several platforms and media. Therefore Best & Co.’s role can be described as that of a supplier, providing design and IT solutions to business client organisations seeking specialised expertise in the development of web-based services. The diagram below shows the value chained relationship between Best & Co. and their clients (Figure 4.1).

Best & Co. was selected from a range of potential case study candidates on the basis of it offering a diverse set of web-based solutions to a varied client base, and they are responsible for the support and maintenance of all developed solutions.

The development methodology adopted by Best & Co. was largely based on the sequential waterfall model (Fitzgerald, 1997; Fitzgerald and Avison, 2003). As per the model the development activities were segregated into several recognisable phases; requirements analysis, design, development, testing and maintenance. Entry and exist criteria of the phases were not ‘formally’ adhered to, with many tasks overlapping with one another. This methodology seemed to suit Best & Co., however there were instances where the methodology led to inefficiencies and problems during the lifecycle. For example it was noted in one particular case that the design phase had not been completed even though the development of the database schema had began. This resulted in a significant revision of the actual schema used as the design was eventually changed. In another instance, due to time constraints user acceptance testing was conducted on the WIS even though development was still in progress. As a result many of the issues that were raised as change requests during maintenance were either caused by a lack of testing on functionality (as these were not ready during user acceptance testing), or as a result of regressions occurring with the implementation of the extra development.

The start of the maintenance phase can sometimes be an ambiguous point. This can be due to interpretations of phased development strategies and also client/ supplier contractual arrangements may vary the definition of maintenance. This is significant as maintenance contracts are often subject to different criteria and cost structures when compared to the commissioning of a new development. For the purposes of this research the hand-over date was regarded as the point where the new development has been formally accepted, deployed in the productive environment and is in an operational state for the end client. The change requests were only collected from this phase.

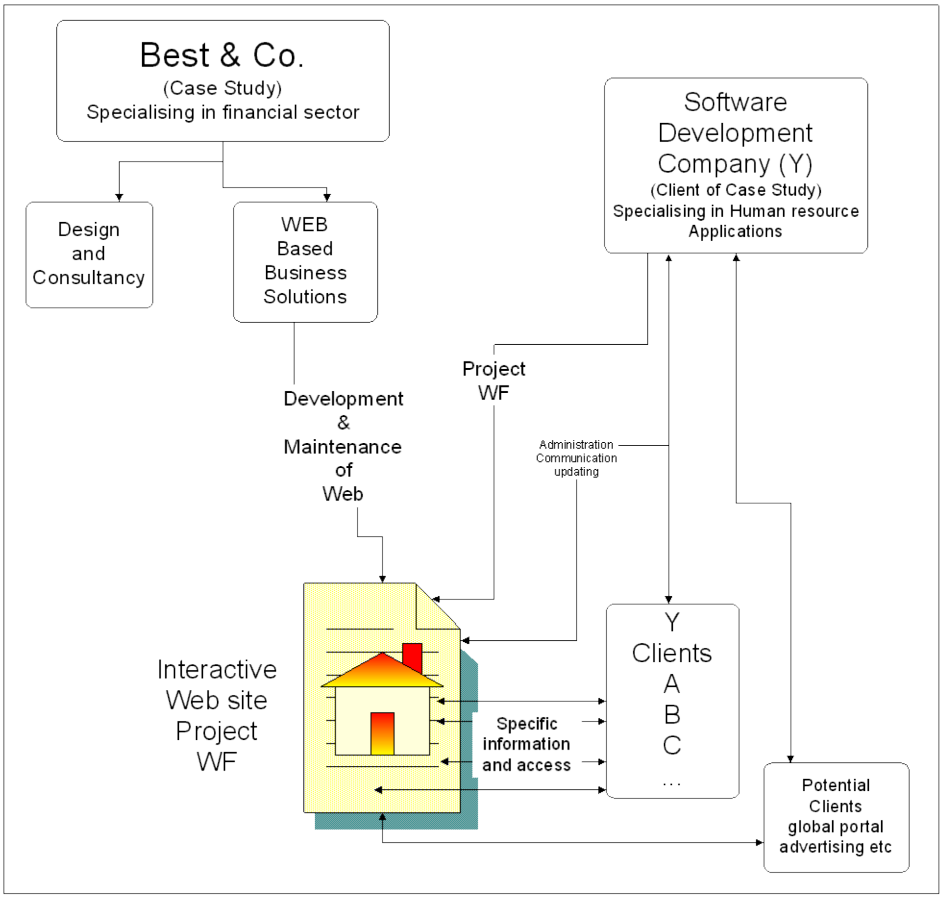

Figure 4.1 - Relationship of the case study (Best & Co.) to client

4.3 The Process of Research Framework A

Research Framework A, as designed in Chapter 3, will be explained in the following sections. The data collection and preparation will be described, followed by a discussion of the categorisation process of change requests into pre-defined categories. The results of the categorisation process will be shown and analysed and a comparison made between Fitzgerald et al.'s (1999) results (traditional IS) and the researcher's results (WIS). Thereafter, implications will be drawn from this comparison and conclusions reached on Research Framework A.

4.3.1 Data Collection and Preparation

Research Framework A collected empirical data from various WIS's that Best & Co. support and maintain. The purpose of this was to diversify the dataset and reduce the likelihood that any one particular project could skew the results as a consequence of contextual issues.

The investigation began by monitoring and recording maintenance change requests which were used to identify areas of WIS's influenced by the change. These changes were recommended by the client organisation (end client) and relayed to the development company (web services supplier (WS supplier)) as change requests. These change requests were raised by the end client during the maintenance phase i.e. after operational go-live date of the delivered solution. The investigation took place on the premises of the WS supplier and the researcher had no direct individual contact with the end client themselves. Rather, the researcher's communications were directed to the senior developer, change manager, project manager and to the database administrator within the WS supplier's organisation.

Change requests were collected from three key sources of evidence by making field visits to the case study site, which were documentation, observations and interviews. There was a particular focus upon documentation, as change requests were received by the WS supplier mostly in this form. The documentation came in several formats, which included written reports, formal meeting minutes, e-mail's and formal client change request documents (examples of which are contained in Appendix B1-B4). The WS supplier used an internal issue-tracker management tool to aid the process of recording and implementing changes to the WIS (Appendix B5). Direct observations were made as another means of data collection. This type of data collection activity was by way of observing formally arranged meetings or telephone calls between the WS supplier and the end client company that pertaining to change requests. The data collection activities produced qualitative data (Gross et al., 1971). The interviews were directed at the development team within the WS supplier organisation and were guided conversations rather than structured queries (Rubin and Rubin, 1995) where the stream of inquiry was free-flowing rather than meticulously planned. Focused interviews were conducted (Merton et al., 1990) where set non-leading questions were asked of the developer in terms of what change requests they had received on a particular day. The development team provided the necessary evidence for the data collection by directing the researcher to specific references within their internal issue tracking management tool.

A total of 150 WIS maintenance change requests over a three month period were gathered, forming the empirical data using the techniques defined in Research Framework A. The data collection activity concluded by consolidating and recording the collected maintenance change requests into one comprehensive dataset document (example in Appendix B6). Field visits were made weekly to the case study premises until the end of the research investigation period.

The following section describes the categorisation process, where each change request was categorised as per the definitions prescribed by Fitzgerald et al.'s (1999) AoC categories.

4.3.2 The Change Request Categorisation Process

The change request categorisation process entailed each individual change request being filtered and classified into areas of change categories. These predefined areas of change categories formulated by Fitzgerald et al. (1999) were used as the 'literature based' model in which the maintenance changes would be categorised. The areas of change categories are defined and listed in Table 4.1.

Area of Change Category	Definition
Data	New and amended fields in files or database.
Processes	Logic and code changes or new procedures.
Interface	User and systems interfaces.
New/ upgraded	Software releases and patches to render services.
Other	Miscellaneous category for data items that do not fit distinctly into the above categories.

Table 4.1 – Change Categories as defined by Fitzgerald et al. (1999)

The categorisation process consisted of two parts, where the first pertained to the researcher categorising change requests, and the second part was where an independent researcher repeated the process. The independent researcher was a fellow Ph.D. student who was not affiliated in any other tasks relating to this investigation and provided an objective as well as an unbiased perspective on the categorisation process. The purpose of the second part of this process was to provide validation for the findings of the researcher's interpretations. The independent researcher results from the categorisation process, were largely consistent with the results attained by the researcher. However, the independent researcher allocated twenty two change requests into different categories from that of the researcher conducting the process initially. Where there was a difference of opinion between the researcher and the independent researcher, it resulted in a third opinion being requested from the WS supplier organisation to arbitrate on which of the opinions is more valid. In this case a final decision on the categorisation was based on a consensus between the three parties.

Two examples are provided of change requests that underwent the categorisation process as can be seen below. The first example will show a change request that falls squarely into an existing category. The second example of a change request is where the researcher categorised it into one category, but the independent researcher categorised it into another and subsequently a third opinion was needed to settle the matter.

<u>Example 1</u>

> 'Can you re-align the style sheets across the whole site to be centralised on the browser – keeping a consistent and even spacing of the margin area around the window.'

This change request relates to visual components (spacing determined by style-sheet formatting) residing on the front-end user interface (browser) of the WIS. Due to these characteristics it was deemed that this particular change request best mapped to the Interface area of change definition as shown in Table 4.1.

<u>Example 2</u>

> 'It doesn't seem possible to re-size the separate browser windows in which the forum is displayed. Would it be possibly to have the facility (zoom in and out within reason - not sure!) to re-size secondary pop-up windows according to what users will want.'

The researcher deduced that to implement such a change would involve a modification to the existing logic and functionality (zoom in and out) resulting in new code needing to be developed. Hence, the researcher categorised this change request in the Processes area of change category. The rationale of the independent researcher however was based upon the fact that because the visual component (browser pop-up window) was being impacted that this change should be categorised in the Interface area of change. In this instance and as per the design of Research Framework A a third opinion was required to arbitrate between the two perspectives. The WS supplier organisation and specifically the developer responsible for exercising this change had to decide which opinion he agreed with more. In this particular instance the consensus was reached that this change request should be categorised as that of Processes.

Table 4.2 offers five other examples of change requests that were assigned to an area of change category.

Change Request	Area of change category
When viewed via the admin area, the add new forum section of the 'All forums' page now includes an 'active' checkbox to suspend a forum. However, this Checkbox actually belongs on the 'Topic Summary' page for each forum so you can suspend, or re-state the Forum as required. Can we now provide extra data fields to view all the forum entities in one outputted table?	**Data**
Several clients have reported 'problems' with opening PDF's on the site (e.g. the PDF won't open) which we believe are really due to problems with the speed of their own PCs, their Internet connection etc. One way to mitigate this perceived problem would be to display a 'progress' message function, whilst PDF's are loading. Would this be possible?	**Processes**
Right hand margins in pop-up pages are tight to the edge of the page. Good example is products/applications/human resource manager/absence and holiday manager link. Can we enlarge the window frames and add in a border line to all pop-up pages	**Interface**
When viewing the reporting page the check boxes in the left hand column are misaligned in IE6. In IE5 this problem does not exist. Ensure that all pages can be viewed correctly using IE6.	**New and upgraded**
Can you please modify the work order contract to include hand-over criteria specifying the acceptance and go-live requirements.	**Other**

Table 4.2 – Examples of client change requests and their placement into relevant areas of change categories

The categorisation process produced results that will be discussed in further detail in the following section.

4.3.3 Results and Analysis

Table 4.3 summarises the results from the categorisation process. From this table it can be seen that Processes and the Data categories together were the most frequent areas of the WIS's requiring changes (81.33%). The Other, Interface and New/ upgraded Packages categories collectively represent only 18.67% of the dataset. Just over a half of the 18.67% is specifically relating to the category of

Interface, and the results also show that the New/ upgraded Packages category contains the least amount of changes overall (4%). As 47.33% of the change requests constituted the Data category, it became evident that this phenomenon needed further investigation. Although not part of the initial Research Framework A design, the further work conducted entailed applying techniques to decompose the Data category.

Areas of Change Categories	No. of change requests	% (2dp)
Data	71	47.33
Processes	51	34.00
Interface	15	10.00
New and upgraded packages	6	4.00
Other	7	4.67
Total	**150**	**100.00**

Table 4.3 – Categorisation results of change requests from Research Framework A

Change requests that fell into the Data category were further organised and grouped into sub-categories according to their similarities. The technique adopted was to cluster and pattern match the individual change requests until naturally forming sub-categories emerged (Miles and Huberman, 1994). This refining process identified three new sub-category types that had distinct characteristics. The sub-categories that were formed and their statistical distribution can be seen in Table 4.4. The sub-categorisation process involved the researcher conducting an exercise on the Data category by assigning each change request to an appropriate sub-category using a top-down approach.

Sub-categories of Data	No. of change requests	% (2dp)
Graphical	9	12.68
Content	59	83.10
Database	3	4.23
Total	**71**	**100.01**

Table 4.4 – Results and sub-groupings of the Data Category

It is noted that all three sub-categories share the common feature of being a data component requiring changes that relate to new or amended fields in files or databases (Table 4.1). The sub-categories of the Data category are defined as follows:

- The *Graphical* sub-category in this instance refers to distinct graphic components of the interface such as image files (for example .gif, .jpeg, .png) that required specific amendments. The Graphical sub-category represented 12.68% of change requests within the parent category of Data. These graphical components are encompassed within the presentational aspects of the graphical user interface (GUI). The change requests in the Graphical sub-category are specific to one component of the GUI and localises the change to one particular point of the WIS interface. The modifications of the GUI components were further refined by the researcher into manipulation (simple edit of graphic or size), major (requires designer involvement and coding) and minor (requires designer involvement with no coding requirement). An example of a change request that was found within the sub-category of Graphical is shown below:

  > 'The original 'girl with the newspaper' image will be much improved by adding the company logo to the laptop (61_14.jpg). Could you change this to take effect on all the relevant pages?'

- The *Content* sub-category relates to the static text element within the WIS. This sub-category formed 83.10% of change requests within the Data category, inferring that 'published' website material is a component of the WIS that is likely to be modified. Additionally this is an important finding as it is purported by this research that a significant difference between traditional IS and WIS maintenance exists based on the volume of change that occurs in the Data category. As stated the vast majority of the Data category changes relate to Content type changes. This issue of Data and Content will be further discussed in the following sections within this chapter. Below is an example of a change request that relates to the sub-category Content:

> ‘The content on the following pages (see attached document) needs to be changed and up-dated to show personal details and contacts of new staff members.’

- The *Database* sub-category relates to changes that explicitly involve data that resides in the database. This could typically be insertions, deletions or data transformations within the database. Out of the three sub-categories identified, this is the least significant area requiring change (4.23%). An example of such a change request is provided below:

> ‘Add indexing to client list in database.’

The results from the categorisation process in this case study and results attained from Fitzgerald et al.’s (1999) survey will be considered in the following section.

4.3.3.1 Comparison of Results and Findings

A comparison between the survey results and the case study results, give rise to several findings. It can be shown that definite differences exist between the findings of Fitzgerald’s et al.’s (1999) survey and this case study.

Table 4.5 shows that the Data category was the most significant area of change within WIS. When compared to the results of studying traditional IS maintenance it is found that over three times as many change requests are affecting WIS’s. The Processes category had the second most influence as an area of change in the IS study as well as the WIS study. Between the survey and this study there is a 10% increase of process logic/ code and new procedure type changes occurring. This suggests that modifications and enhancements of this particular component of WIS’s are more frequent than indicated within traditional IS maintenance change. In contrast to the finding above, the results of the Interface category shows that there are almost double the amount of user/ system interface changes that affect the IS's studied by Fitzgerald et al. (1999), than the WIS's studied here. The Interface category constituted 10% of the overall changes in the WIS. An even larger difference is noted for both the areas of change categories of New/ upgraded Packages and Other (New/ upgraded Packages are greater in IS compared to WIS by a factor of four and the Other category by a factor of six).

Areas of Change Categories	IS Survey Study (Fitzgerald et al. (1999)) %	WIS Case Study % (2dp)
Data	15	47.33
Processes	24	34.00
Interface	18	10.00
New and upgraded packages	16	4.00
Other	27	4.67
Total	**100**	**100.00**

Table 4.5 – Comparison of IS and WIS areas of change category results

For reasons that will be described below the design of Research Framework A needed to be revised in order to adequately address the research questions posed by this thesis.

4.3.4 Implication and Conclusions from Research Framework A

The results yielded from the operation of Research Framework A clearly show that a significant distinction can be drawn when regarding maintenance change between a traditional IS context and that of a WIS. Table 4.5 exemplifies the distribution of the areas of change detected when analysed using Fitzgerald et al.'s (1999) AoC model. No common trends between the change categories across both studies were shown to be evident. Although the categorisation process proved to be confirmatory between the IS and WIS contexts, the findings of the results were non-confirmatory between the two studies. This finding which has been empirically confirmed during the operation of Research Framework A lends weight to the debate that WIS maintenance change is inherently different to traditional IS maintenance change (Section 2.2.1).

Another important conclusion to be drawn is the significance of the Data category as an area of change within WIS maintenance. The categorisation process within Research Framework A reveals that over three times as many change requests occur for the Data area of change in WIS maintenance when compared to traditional IS. This finding led the researcher to question the composition of the Data category in greater depth and resulted in the natural progression of modifying the Research Framework A design to further elaborate on the properties of this area of change category. Further work was conducted beyond the scope of Research Framework A by deviating from Fitzgerald et al.'s (1999) definition based categorisation model to an exploratory clustering and pattern matching technique. This was done with the intention to explore the Data category further. The process revealed three clear sub-categories derived from the parent Data category, namely, Graphical, Content and Database. The successful grouping of sub-categories was the driver to reconceptualise the research framework to incorporate a more emergent category building activity, which was required to investigate the areas of change in WIS's.

The further work conducted in Research Framework A indicated that the categories of the AoC model were too broad. Since a greater understanding of all categories of change is required (not only Data) Research Framework A evolved to accommodate these requirements. Some elements of Research Framework A were retained within the revised scope of the new research framework (Research Framework B):

- Retention of exploratory philosophical research approach.
- Case study strategy.
- Change request data collection techniques.

- Analysis technique of clustering and pattern matching (Miles and Huberman, 1994).

In addition the following attributes were added to the scope of the revised research framework:

- Incorporation of a bottom-up discovery process allowing for granular category building without any predefined assumptions.
- Case study approach with embedded units of analysis to increase data pool and generalisability within the remit of this study.
- Deeper qualitative understanding of the change achieved by the development of additional constructs for analysis.
- Data preparation technique using a spreadsheet repository to produce the dataset for analytical purposes.
- A mechanism to validate the discovered categories emerged by the bottom-up approach.

The change in the research approach also prompted an emphasis shift regarding the objectives of the thesis. It became evident that in order to adequately build a WIS flexibility framework the prerequisite question regarding the WIS nature of maintenance change would have to be thoroughly investigated. To address this issue, research question one was supplemented with a sub-question that sought to ascertain the characteristics of WIS maintenance change further ("What are the characteristics of change?"). In addition the focus of research question two became the conceptual design of a flexibility framework as opposed to an implemented and validated flexibility framework.

Had stage two of Research Framework A proved to have been successful, the research would have continued to the survey and action research stages. This would have resulted in the construction of a flexibility framework for the purpose of facilitating WIS maintenance change (Figure 3.1).

Section 4.4 will now describe the revised research framework and its implementation will be discussed in greater detail.

4.4 The Operation of Research Framework B

Best & Co. was retained as the case study for Research Framework B. The data collection in Research Framework A consisted of change requests from multiple projects to form the unit of analysis. In this section four distinct projects form the embedded units of analysis within the case study strategy (Yin, 2003). A detailed profile of three projects will be given below; as per the research framework design a fourth project was used as a validation/ control project (Section 3.5.3) and will not form part of the analysis dataset on which the findings of this thesis are based. Research Framework B comprises of a data collection and preparation process of change requests that were aggregated into one coherent dataset. The process of developing emerging change group categories as well as using the constructs of analysis for each change request will be discussed in the following sections.

4.4.1 The WIS Projects investigated within the Case Study

Figure 4.2 illustrates the WIS embedded units of analysis within the case study. The four projects each map to an embedded unit of analysis according to Yin’s (2003) case study model. A reason for using this strategy was to highlight phenomenon trends and simultaneously reduce the effects of anomalies within the case study. This was achieved by using a large data pool derived from diverse data sources. Project one was an WIS application for a human resources company that provides services to multiple clients. Projects two and three provides WIS applications that relate to a global financial institution and a provider of private banking services to high net worth clients, respectively. Project four provides a WIS solution for a hedge fund and private equity group that serve institutional and high net worth clients. All four projects are comprised of differing technical specifications and encompass various businesses, however all fit the common criteria of WIS’s providing online services (Section 2.2).

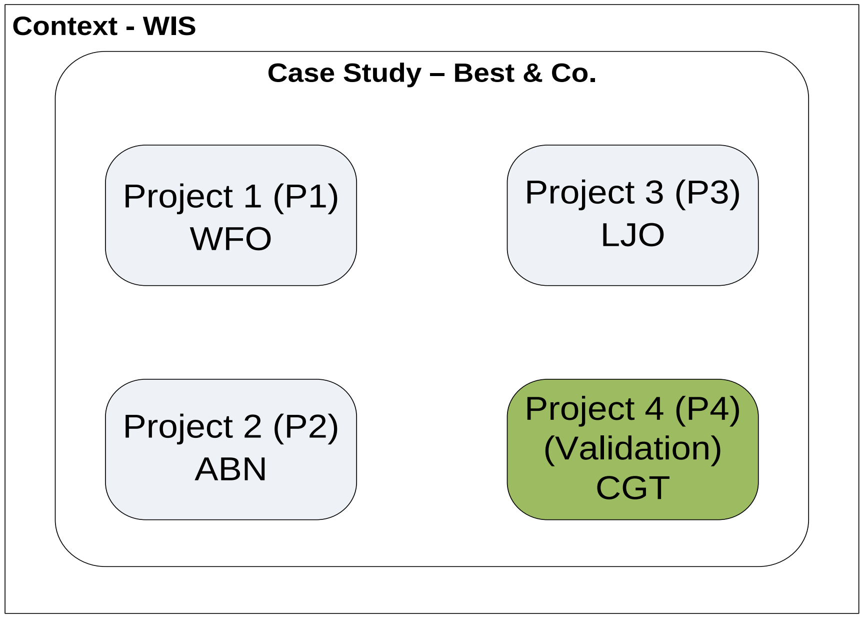

Figure 4.2 – Embedded case study units of analysis (adapted from Yin (2003))

Project 1 (P1) – WFO

The WFO project (P1) was commissioned to Best & Co. by a privately established software development company. The end client company provides business services and support to organisations seeking human resource management applications, for example payroll systems. The end client company is a software development house in their own right and thus are familiar with the methods and management processes associated with the development of IS's. In this instance the client company opted to outsource the complete development of the WFO project (P1) to specialist web service providers.

The project involved the design, development and implementation of a web-based IS. An integrated corporate online system was required, that disseminates and markets human resource products and support services. The end client company wanted to expand their business model by including an on-line repository service to both their existing and potential user base. The broad specifications of the system were inclusive of a backend database storage system, a content management system (CMS) (Friedlein, 2003) and an open forum communication facility. Included within the requirements was an administrator function that allowed the manipulation of access rights and the CMS. The scope of the application provided information and services to various types of user groups based on an authorisation mechanism, these groups being the client company's administrator, personnel and the existing/ potential clients. Access rights and permission based functions were considered as primary security specifications. The measures that were taken involved the locking of IP addresses for the accessibility of the administrator facility to be restricted within the boundary of the client organisation. Additionally the specification of the authorisation credential required encryption methods to maintain confidentiality of client information.

The entire development process of P1, in particular the maintenance phase, was investigated. The researcher monitored change requests over the first three months of the maintenance phase. The study took place at the WS supplier site with minimal interaction with the end client company itself. The researcher's interactions were mainly with the project managers, change managers, designers, senior developers and the database administrators.

The organisational structure of the P1 team consisted of a managing director from Best and Co., as well as two other project managers one of whom represented Best & Co. (WS supplier) and the other the end client company. The end client company's project personnel were also involved in communicating the change requests to the WS supplier as well as performing quality assurance activities, for example user acceptance testing. Other key roles in the P1 development team consisted of technical architects, GUI designers and configuration managers. The GUI designers were responsible for creating interactive multimedia components of the site, and the senior developers, technical architects and configuration mangers were responsible for the technical specifications of the solution.

Having a non-financial project as one of the embedded units of analysis assisted in diversifying the dataset in terms of context. Figure 4.3 is an example of the P1 GUI, showing the homepage.

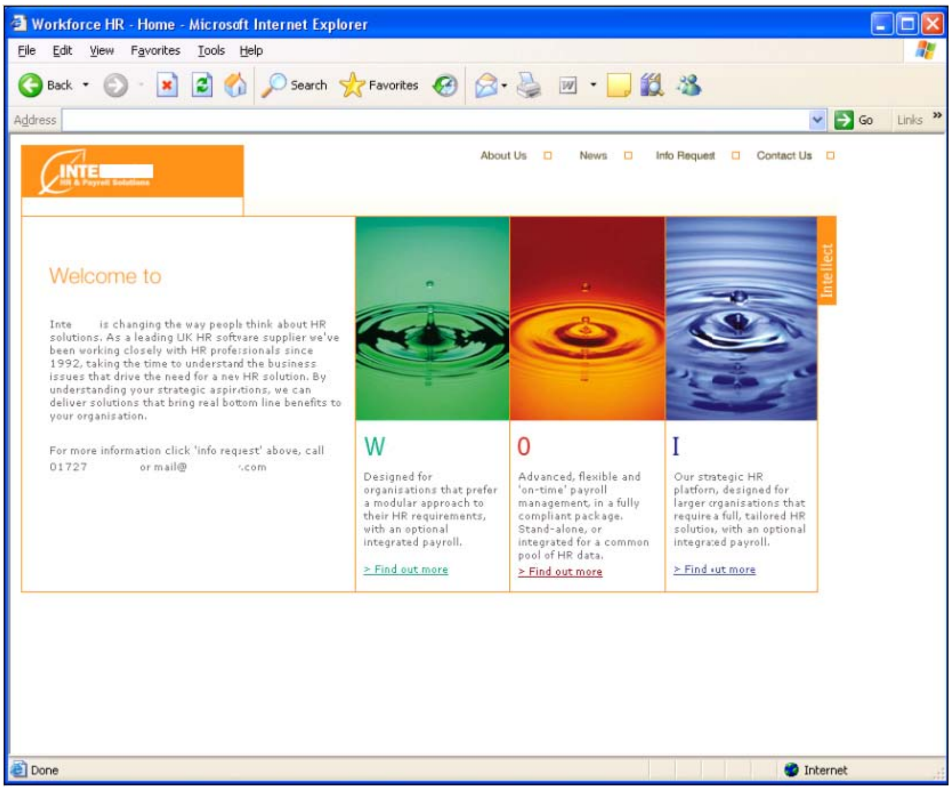

Figure 4.3 – Example of the P1 GUI

Project 2 (P2) – ABN

The P2 project involved the online implementation of the end clients discretionary portfolio management services (DPM) team to allow their customers to delegate the management of their assets. Depending on the selected service, assets can be invested in different classes, such as equity, fixed income, liquidity or various alternative investments. The proposed online application tool provides customers with continual access to investment portfolios usually attainable and restricted through traditional forms of communications.

The design and implementation of the web-based DPM application took the form of a fully interactive real-time WIS that was targeted towards a specific type of customer base. The P2 project brief was inclusive of the following key objectives: to serve as a tool for relationship managers within the team and to ascertain the investment profile needs of the customer. The WIS also provided a calculation facility to compute complex financial queries and generate reports based on the customers current portfolio positions. The requirement for a multi-lingual site was also specified that accommodated the English, French and German languages. Due to the sensitive and classified nature of the financial data, security was a paramount concern and was a key requirement of the WIS. Encryption methods based on token key affirmation techniques needed to be stable and robust components of the WIS. Furthermore, the application intended to provide a highly multimedia based GUI to handle cumbersome and complex financial data inputs as well as outputs.

As with the P1 project the researcher focused on the maintenance phase within the P2 project lifecycle. The observation period spanned the first three months of the maintenance phase. Similarly to the P1 project, the research was conducted at the WS supplier site with minimal contact with the end client company. The researcher interacted with the same development team members as mentioned in the previous project profile.

Contrary to the end client company’s interaction with the WS supplier on the P1 project, the P2 project development team had a single point of contact at the end client organisation. Although the point of contact was not an IT specialist, they were however supervised by a project manager who was. They signed-off key stages of the project, tested the solution as well as supplying relevant feedback to the WS supplier. Weekly updates and continual involvement of the end client organisation with the WS supplier was a key facet of the project lifecycle. The WS supplier development team structure was the same as in the P1 project, however during the

P2 project, the designers had a far greater influence due to the intricacies of displaying multimedia financial data (for example market performance projection graphs). Figure 4.4 provides an example of P2's GUI.

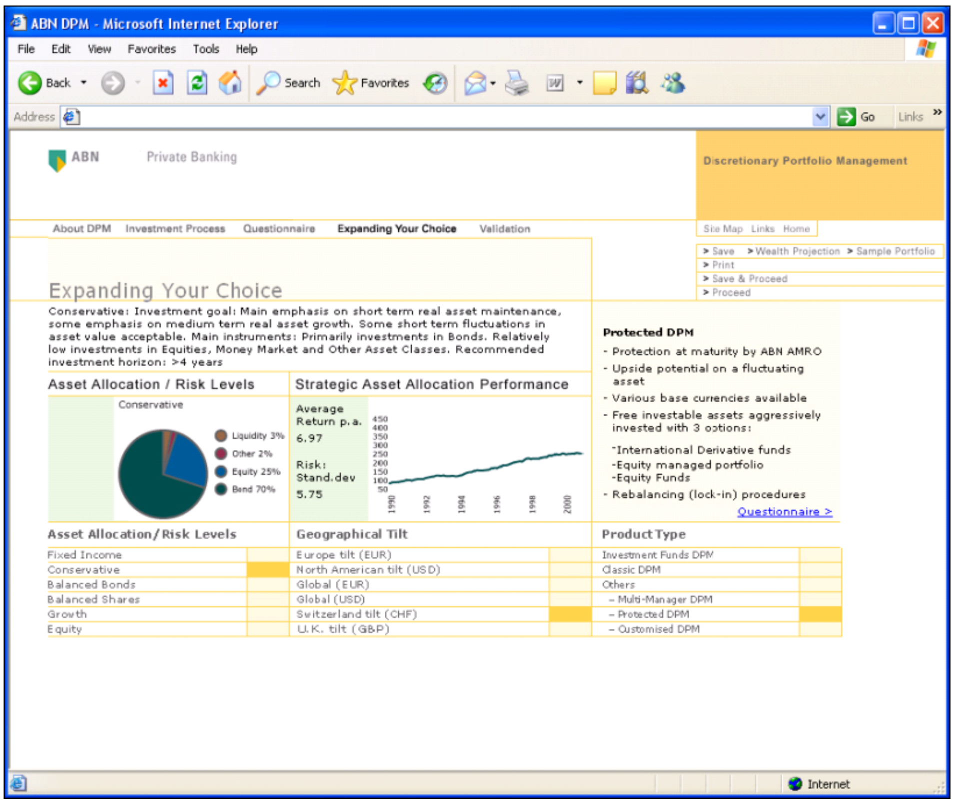

Figure 4.4 – Example of the P2 GUI

Project 3 (P3) – LJO

The P3 end client company are an independent, specialist private banking and investment management group that operate in London and various off-shore locations. The London based merchant bank has a range of private banking services, spanning tailor-made wealth management activities to high net worth individuals. The U.K. business activities consist principally of banking, client-driven treasury transactions, loans, mortgages, specialist pensions and trust services as well as investment management processes. The P3 project's objective was the online implementation of financial business services, particularly the investment and wealth management portfolios for its customers.

The P3 project involved the design, development and implementation of a web-based resource. The WIS online interface required the interconnectivity between corporate wide existing sub-systems that were to be linked to the web application. A communications facility between customers and the end clients relationship managers were required in order to respond to user queries. The WIS included document files like PDF's, bulletin messages, financial portfolios, investment reports and reviews. Due to the end clients aggressive growth strategy, the WIS had to be scalable enough to cope with an expected increase in the volume of transactions in future releases. Thus a mechanism whereby the end client company could manage site content information was specified as a WIS component. A CMS was included in the specifications that enabled the up-dating and modification of information on the web site. By adding the CMS the access rights of particular users were determined. The access right component of the WIS additionally provided a security measure to access the online application. Due to the confidential nature of client's financial data, security issues were addressed with encrypted credentials that were authenticated with a token key mechanism to verify and confirm the users. Due to the volume of documents and reports that were to be included in the web application, an intuitive GUI design and navigation mechanism was required to display complex financial information.

The researcher monitored the first three months of the maintenance phase. As with the previous projects, the researcher interacted with the WS supplier on a regular basis. The end client were represented by a single point of contact who was not an IT specialist. All communication to the WS supplier was channelled via this solitary point of contact. Figure 4.4 is an example of P3's GUI.

Figure 4.5 – Example of the P3 GUI

4.4.2 Data collection for Research Framework B

The purpose of collecting change requests in Research Framework B was to ascertain the nature of change in WIS maintenance. The first objective is to identify areas of the system that are susceptible to change, which was also the purpose of Research Framework A. However, the second and additional objective of Research Framework B was to gain a deeper insight into the characteristic nature of the change itself. The second objective was answered using the additional use of analysis constructs, these being Context, Complexity and Form. This section will discuss in detail how evidence was firstly, collected and secondly, how it was prepared for each change request. It is the ultimate collation of this evidence in a spreadsheet (Microsoft® Office Excel) that becomes the dataset. The spreadsheet aided in tabulating the data, and was preferred over using a self-created database because it provides the tools for data analysis such as pivot tables and filters. From this dataset an in-depth understanding of the inherent nature of WIS maintenance change was achievable.

The analysis construct of Context will be discussed first. A first categorisation process that allows areas of change group categories to emerge through the use of clustering and pattern matching techniques (Miles and Huberman, 1994) will then be described. Following this a second definition based categorisation process utilising pre-existing definitions derived from the first categorisation process will be used for validation purposes. The section concludes with a discussion of the analysis constructs of Form and Complexity.

4.4.2.1 Context

For the purpose of the analysis construct of Context; change management, technical aspects, management issues and the business case (Section 4.4.1) were evaluated.

The objective of the WS supplier's change management process was to ensure that standardised procedures were used to efficiently handle change requests. This was done in order to manage the workflows and delivery of change requests raised during the maintenance phase.

Each project was subject to the internal change management process adopted by the WS suppliers project management methodology. The change management process included the development, testing and implementation of the software changes.

The WS suppliers change management process comprises of the raising and recording of changes, assessing the impact and cost, managing and co-ordinating change implementation, monitoring and reporting on implementation, reviewing and lastly closing change requests. The main activities involved in change management at the WS supplier included:

- Filtering changes.
- Managing changes and the change process.
- Reviewing and closing of requests for change.

- Management reporting and providing management information.

Figure 4.6 shows the specific change management process that was used by the WS supplier. The researcher through observations of the change management process and relevant documentation was able to retrieve information pertaining to each construct of analysis.

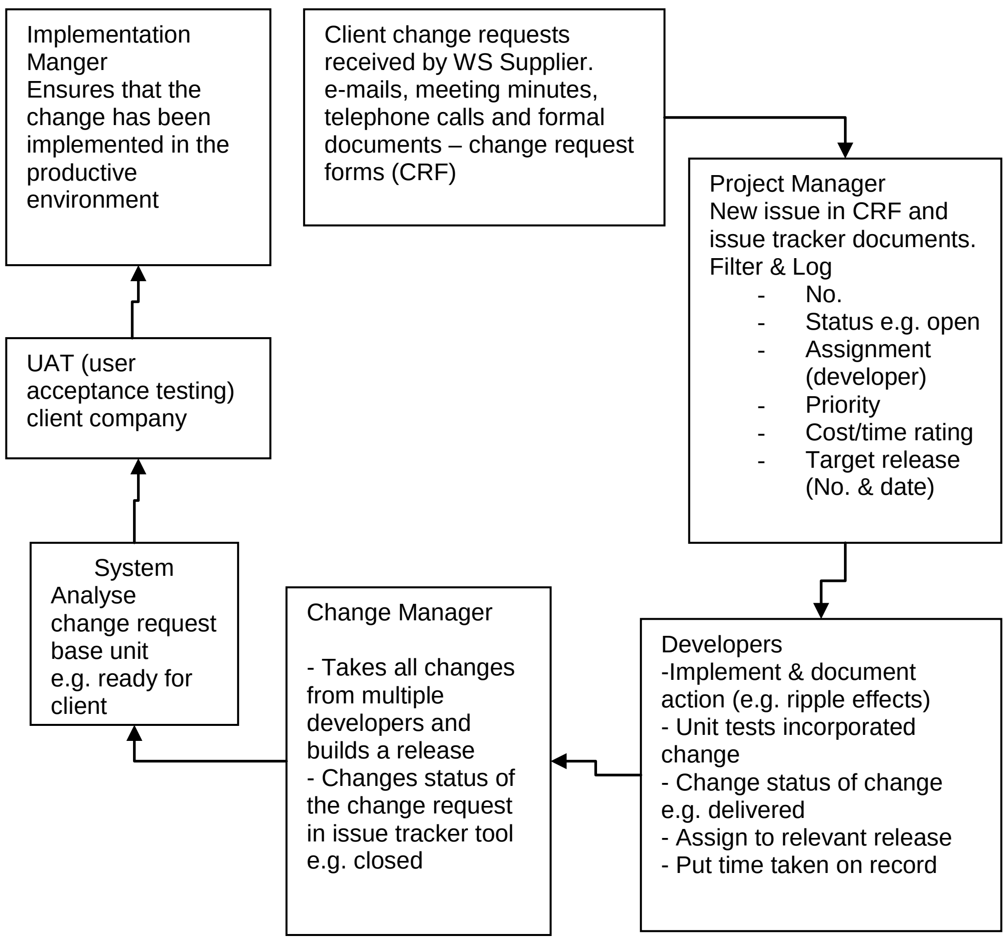

Figure 4.6 – Change Management Process

The researcher during the course of the case study had informal discussions and conducted set question interviews with the WS supplier development team members. The questions can be seen in Table 4.6. The high level profiles of each project provides additional contextual information as to what technologies were used by the WS supplier development team and management issues surrounding change requests. The profile details of each embedded unit of analysis is summarised and presented in Table 4.6.

No.	Questions:	P1	P2	P3
1	**What was the end client sector?**	Software Dev.	Banking	Banking
2	**Is it content managed?** Level of content management 10 = high usage 0 = low usage	Yes (9)	No (0)	Yes (8)
3	**What was the hosting platform?**	Microsoft IIS	Microsoft IIS	Microsoft IIS
4	**What was the backend platform (Database)?**	SQL Server 2000	Oracle db	SQL Server 2000
	Reason for database platform?	High concurrency required (multiple logons) plus Speed (Forum) (T-SQL)	High concurrency, speed and power of server side scripting (PL-SQL)	High concurrency, speed and power of server side scripting (T-SQL)
5	**What Languages/Scripting is used?**	Cold fusion (8) JavaScript (8) HTML (10) T-SQL (7)	ASP (6) JavaScript (8) HTML (10) PL-SQL (6)	ASP (9) SQL (9) JavaScript (7) T-SQL (9) HTML (10)
	10 = high usage 0 = low usage			
6	**What are the main site components?** Excluding public GUI	Site Admin Real Time Forum Real Time Stats	Investment portfolio Real time Stats Calculation facility Financial reporting Multi-lingual	Trading Platform Trading Admin Site Admin Bloomberg Admin Real Time Stats
7	**What is the site complexity level?**	9	7	8

	(development team view) Based on database complexity, scripting used and special GUI elements (e.g. Navigation features – scroll devices etc.) 10 = High complexity 0 = Low complexity			
8	**Where there any project management issues?** Client company	None	None	Poor documentation
9	**Where there any project management issues?** Development team	Forum Component specification underestimated Poor documentation	Poor documentation	Trading Component specification underestimated Poor documentation
10	**Was the project delivered on time?**	No	Yes	No
	If not why?	Client side content delay	/	Client side content delay
11	**Was the project delivered within the budget scope?**	No	Yes	No
	If not why?	See project management issues	/	See project management issues
12	**Was the project brief achieved?**	Yes	Yes	Yes

Table 4.6 – Project profile details

4.4.2.2 Data Preparation and the Category Building Process

As with Research Framework A, documentation relating to change requests came to the WS supplier in the form of change request documents, emails, formal meeting minutes and ad-hoc requests over the telephone. The change management process of the WS supplier involved these documents being converted into their own change request forms (CRF) (Appendix B4) and subsequently inputted into the issue tracking management tool (Figure 4.6). By analysing the CRF's and the formal release notes, the change requests that were not implemented or charged for were not included in the dataset of this research. The change requests that were collected from all three project cases were consolidated and prepared into a coherent dataset repository, using the Microsoft® Office Excel spreadsheet (Appendix C). The data from all three project cases was complete and stored in the repository before the process of category building was undertaken.

Table 4.7 shows a sample from the dataset repository. Data in column A of the spreadsheet relates to the WIS project that the change request originated from. For example, it can be clearly seen from table that the first change request (row 2) is from the WFO project (P1).

The second column (Table 4.7) was where the researcher gave each change request a unique identification tag in order to make it distinguishable from other change requests within the dataset table (for example column B row 2 shows the change request identification as being WFM1).

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
2	WFO	WFM1	Scrolling facility restricted	Bespoke Navigation Scroll Bar	Bespoke scroll bar	Scrolling device	Adaptive	Presentation	Yes	6 instances of same element on interface (GUI)	4	High	Object or entity of GUI interface control mechanism
3	WFO	WFM2	Up-date of On-screen message display	Text Heading	Acknowledgment output function	Function	Corrective	System	Yes	3 instances of same element on interface (GUI)	3	High	Acknowledgment process status. Business process
4	WFO	WFM3	Remove text	Character text content	Text content edit	TUI content	Adaptive	Presentation	No	None	0.5	Low	Symbols and characters included within text
5	WFO	WFM4	New mouse-over picture refresh	Mouse-over function	Roll-over picture refresh function	Function	Adaptive	System	Yes	New images for mouse over function	3	High	Graphic up-date link function
6	WFO	WFM5a	Replace incorrect numbers	Numeric content	Numeric content	TUI content	Corrective	Presentation	No	None	1	Medium-Low	
7	WFO	WFM5b	Numeric format consistency	Numeric content	Format numeric content	TUI content	Adaptive	Presentation	No	None	1	Medium-Low	Template for tel/fax numbers
8	WFO	WFM6a	Change text of link	Link Text Change	Text navigation link	Textual navigation hyperlink	Adaptive	Presentation	No	None	0.5	Low	Style sheet
	WFO	WFM6b	Change Text	Text Change	Text content	TUI content	Adaptive	Presentation	No	None	1	Medium-	

Table 4.7 – Dataset table (spreadsheet excerpt)

The third column of table is labelled as ‘Change Request Detail’ and represents the researcher identifying the area of change from the end client’s original change request. A summary of the area of change is then inputted into the spreadsheet. For example in column C row 5 the change request detail is ‘New mouse-over picture refresh’.

The forth column D called ‘Impacted WIS Area (WS supplier view)’ was where the researcher substantiated the clients perception of the change, by gaining the developer’s feedback on the area of the WIS being impacted. This was achieved by reviewing the CRF and issue tracker (Appendix B5) documents with a development team member from the WS supplier. For example, in column D row 5 the impacted WIS area from the view of the developer is summarised as being ‘Mouse-over function’.

The fifth column E, namely ‘Base Unit Area of Change (Researcher)’ was the researcher’s perspective of the area of change which is derived from the client company’s perception in column C and the WS suppliers view in column D. The same example provided in columns C and D row 5, now becomes ‘Roll-over picture refresh function’ in column E.

The base unit area of change determined by the researcher then goes through a process of clustering and pattern matching as per the technique espoused by Miles and Huberman (1994). Once the base unit area of changes are clustered according to their similarities, high level change group categories are formed. The third column of the example shown in Table 4.8 (corresponding to column E of the spreadsheet – Table 4.7) are the base unit area of changes that through clustering and pattern-matching, emerge as a higher level change group category of ‘Function’ (Table 4.8).

Case Study Project	Change request I.D	Base unit area of change (Researcher)	WIS change group category
WFO	WFM2	Acknowledgment output **function**	**Function**
WFO	WFM4	Roll-over picture refresh **function**	
WFO	WFM7	Story sort **function**	
ABN	ABN105	E-mail response **function**	
ABN	ABN132	Probability Calculation **function**	
LJO	LJ57	Performance & Standings **function**	

Table 4.8 – Example showing the emergence of the Function group category

When the group categories have emerged at the end of the clustering and pattern matching process, the relevant group category for each change request is added in column F (Table 4.7). The processes described above ranging from column C to F is diagrammatically shown in Figure 3.3. At the end of the category building process twelve WIS change group categories were found to have emerged. Each one and its reciprocal definition is listed in Table 4.9.

Change group category	Definition
Image	A graphical visual component of the GUI – e.g. graph, photograph, picture.
Function	A procedural component e.g. pop up function.
Admin Function	Any function that is used to Administer the application e.g. Setting up user permissions.
Navigation Function	Functional operations on GUI e.g. underlying function of the hyperlink.
Textual Navigation Hyperlink	String object that allows manoeuvre on GUI e.g. text based hyperlink.
Multimedia Hyperlink	Visual component that allows manoeuvre on GUI e.g. image based hyperlink.

Scrolling Device	Aesthetic control widget allowing navigation on the webpage.
Textual User Interface Content Format (TUI Content Format)	Formatting of non-hyperlink string object presented on GUI.
Textual User Interface Content (TUI Content)	Editing of non-hyperlink string object on GUI.
Database Data	Stored information in fields contained in the database.
Style Issue	Issue relating to the look and feel of the GUI presentation e.g. reduce spacing on web page.
Business Analysis	Business change that requires analysis only, requiring no immediate system development.

Table 4.9 – Definitions of emerged WIS change group categories

4.4.2.3 Validation of the Emerged WIS Change Group Categories

As discussed previously, the change requests of all three projects went through the emergent categorisation process collectively. The result of this was that twelve WIS change group categories emerged (Table 4.9). In order to validate these twelve categories an independent confirmation method was required. The design of this confirmation method is described in Section 3.5.3 and 3.5.3.1. The validation process uses definition based criteria to assign change requests to a particular change group category, in this respect it is similar to Research Framework A's use of Fitzgerald et al.'s (1999) AoC model. A validation process was conducted on a forth embedded unit of analysis with the intention of ascertaining whether a saturation point was reached where no new change group categories were detected. Figure 3.4 and 3.5 diagrammatically shows how change requests are mapped to a WIS change group category in the validation process.

The fourth embedded unit of analysis was the P4 project (CGT). P4 involved the implementation of a WIS for a private banking organisation end client that required the development of an on-line service to its banking customers. This project was similar in scope to the P2 project (Section 4.4.1). This secondary data collection stage involved the researcher observing the first three months of the maintenance phase of the P4 project, in which one hundred randomly selected change requests were chosen. Each change request was run through the definition based categorisation process (Figure 4.7) to allocate it to a WIS change group category. It was found that the P4 project change requests fell into the parameters of the established WIS change group categories discovered from the other three embedded units of analysis. As no new change group categories emanated from the validation process, the researcher concluded that a saturation point had been attained with respect to the discovery of WIS change group categories. This proved to be a distinct end point for the data collection activities within Research Framework B.

4.4.2.3.1 An Example of the Validation Process

The change request shown in Table 4.10 will be used as an example for the purpose of describing the definition based categorisation process. Table 4.10 outlines the change request and the description is shown in a similar format to the spreadsheet dataset table (Table 4.7).

Case study project	Change request I.D	Change request detail	Impacted WIS area (Developer)	Base unit area of change (Researcher)
CGT	CGT25	Change text of link	Link Text Change	Text Navigation Link

Table 4.10 – Example of change request used in validation process

Change request CGT25 originated from the P4 validation project. To assist with the validation process the definitions from Table 4.9 had been supplemented with an additional rule that relates to each WIS change group category definition (Table 4.11). The description of the base unit area of change was then examined (as can be seen in Figure 4.7).

Change group category	Rule	Definition
Image	Is it a Graphical component?	A graphical visual component of the GUI – e.g. graph, photograph, picture.
Function	Is it used as a systems tool?	A procedural component e.g. pop up function.
Admin Function	Is it used as a USER tool?	Any function that is used to Administer the application e.g. Setting up user permissions.
Navigation Function	Is this a USER tool for navigation?	Functional operations on GUI e.g. underlying function of the hyperlink.
Textual Navigation Hyperlink	Is this a TEXT based Hyperlink?	String object that allows manoeuvre on GUI e.g. text based hyperlink.
Multimedia Hyperlink	Is this an image for Hyperlink navigation?	Visual component that allows manoeuvre on GUI e.g. image based hyperlink.
Scrolling Device	Is this a Scroll aesthetic?	Aesthetic control widget allowing navigation on the webpage.
Textual User Interface Content Format (TUI Content Format)	Is this text presentation?	Formatting of non-hyperlink string object presented on GUI.
Textual User Interface Content (TUI Content)	Is this text relevance – amendments?	Editing of non-hyperlink string object on GUI.
Database Data	Are these stored information files that contain data?	Stored information in fields contained in the database.
Style Issue	Is this to do with site aesthetics?	Issue relating to the look and feel of the GUI presentation e.g. reduce spacing on web page.
Business Analysis	Is this a global business issue that requires no development?	Business change that requires analysis only, requiring no immediate system development.

Table 4.11 – Definitions and rules of emerged WIS change group categories

The question of ‘Does the change request base unit match to any defintions and rule of existing WIS change group categories?’ was posed as per the workflow defined in Figure 4.7. In the case of the CGT25 example it was found that using the rules and definitions this change request would best fit the Textual Navigation Hyperlink WIS change group category. In the cases where change requests did not map to an existing definition and rule it would be placed into a separate dataset called ‘Uncategorised Dataset’.

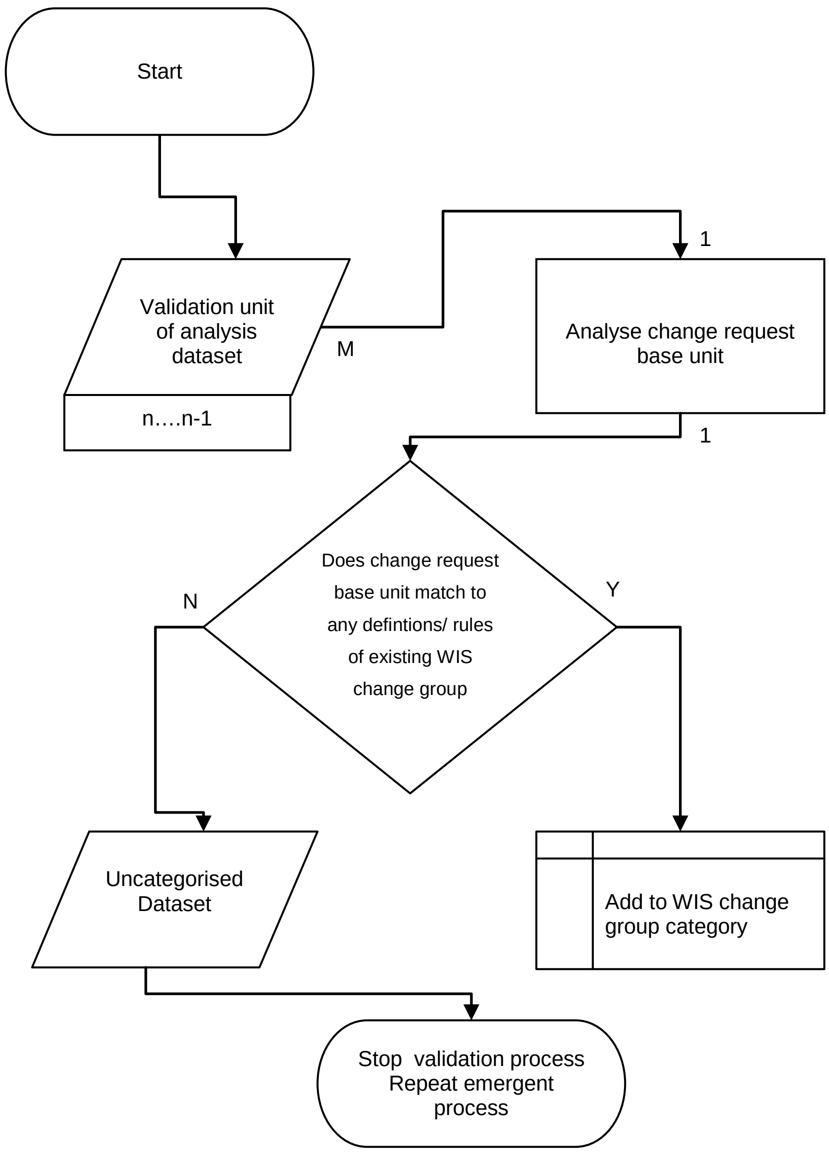

Figure 4.7 – The group categories validation process

The WIS change group categories discovered in this research have emerged in context of this case study and its embedded units of analysis. The contention is that these WIS change group categories serve as a 'data-dictionary' reference point, which should be confirmed, potentially built upon or refuted with additional investigations in various contextual settings.

It was previously discussed in Section 3.5.4 that the constructs of analysis consisted of Context, Form and Complexity. Context has already been addressed (Sections 4.4.1 and 4.4.2.1), thus the following section will discuss how Form and Complexity data was gathered. Aside from identifying WIS areas of change, the constructs of analysis provides a secondary source of analytical data of WIS maintenance change.

4.4.3 Form

The construct Form is defined within this research as being the cause or motivation as to why a maintenance change request emerged. The dimensions of analysis for the Form construct are: Corrective, Adaptive and Perfective which have been defined in Table 3.3.

The Form of each change request was determined by returning to and analysing the original sources of evidence. The documentary evidence comprised of the client's original change request, the appended CRF document and data from the WS suppliers issue tracking tool. The change requests were analysed and labelled as a Form of change in accordance with the definitions presented in Table 3.3. Three examples are provided below showing change requests for each Form type respectively.

Corrective

> 'The text of the on-screen message displayed after you submit the application is spelt incorrectly. It currently says "Thanks for **youre** application.", it should say "Thanks for **your** application."'

The above change request was labelled as being that of corrective type of maintenance change, as it can be seen that the on-screen message is wrong. The change request is referring to a failure in the WIS, such that its behaviour does not conform to its specification (Lientz and Swanson, 1980).

Adaptive

> 'Can you **replace** the **existing** PDF financial reports within the forum section to show the latest reports produced by our partner company.'

From the above change request example, the area of the WIS either needs to be replaced or adapted because of environmental changes to the WIS. Therefore this change request can be classified as that of an adaptive type, as a third party (to the end client) is causing the WIS to be modified.

Perfective

> 'Would it be possible for you to now **create an additional page** linked to the homepage? **The new page** will have **extra information** relating to our client portfolios. **It will also require** a sub-navigation tool bar such that the user can link back to the homepage and to other pages with the client section. This section will **also require a sorting function** to show latest PDF reports. Could we arrange a time for us **to discuss the new change** and how much it would cost.'

This example clearly shows the change request as being an enhancement and a new requirement to the WIS. Therefore this change request requires new components which were beyond the scope of the original specification to be added to the website. Therefore based on the definitions in Table 3.3, this change request can be considered as a perfective maintenance type.

The assigning of a Form attribute was undertaken for all the change requests in the dataset repository. The results were recorded into Column G of the repository (Table 4.7).

Complexity constitutes the third construct of analysis and consists of the Architecture Layer, Ripple Effect and Effort dimensions. These were further elaborated on in Section 3.5.4. and 3.5.5. and its application within the context of Research Framework B will be discussed.

4.4.4 Complexity

As stated in Section 3.5.4 the architecture layer that the change request impacts is deemed to have an effect on the overall complexity of the change. All the embedded units of analysis and the majority of WIS's adhere to a 3-tier architectural typography. Therefore each change request was assigned one of the following three parameters; Presentation, System or Data. In order to ascertain what structural layer the change request effected the most, the researcher interviewed the WS supplier development team, especially the developers who implemented the changes. The main question that was directed at the developer was whether the change request impacted the Presentation, System or Data layers. It can be seen from the excerpt of the dataset, that the analysis of the WIS's structural (architecture) layer has been recorded for all change requests, in Column H (Table 4.7).

The nature of change was further elucidated by analysing the Ripple Effects that emanated from a particular change request that was implemented into the WIS. This information was obtained from the CRF document that the WS supplier development team used as part of their change management process. An example of a CRF can be seen in Appendix B4, in which the developer records other parts of the WIS that were effected as a consequence of the original change. In the dataset repository, in Column I, whether a change request had a ripple effect, was answered in the affirmative or negative. Where a change did have a ripple effect, in some instances a brief description of it was provided in Column J (Table 4.7).

The change request was also analysed in terms of the Effort that was required to implement it into the WIS. This was qualified according to the length of time (hours) the developer(s) took on implementation. The effort in hours was calibrated according to quartile ranges to provide an additional qualitative perspective of ‘low’, ‘medium-low’, ‘medium-high’, and ‘high’. These quartile ranges were statistically calculated by creating a cumulative frequency graph from the aggregated dataset and was further used to evaluate the effort in terms of time taken. The cumulative frequency graph can be seen in Appendix D. The effort ranges (Table 4.12) can be used to identify the ratings of any future change requests based on the aggregated results.

Effort	Time taken (Hrs)
Low	0 < 0.7
Medium-Low	0.7 < 1.44
Medium-High	1.44 < 2.71
High	2.71 < 11.52

Table 4.12 – Effort ratings

Effort in terms of hours for each change request is recorded in Column K of the dataset repository, whilst the effort rating for each change request can be found in Column L (Table 4.7). The researcher gauged the effort taken for each change request, by referring to the issue tracker tool and time sheets used by the WS supplier development team.

Lastly, any additional information relating to a change request, that the researcher deemed as relevant to the investigation was put into Column M of the dataset repository (Table 4.7).

4.5 Conclusion

The researcher collected change requests and other relevant evidence's from the case study site on a weekly basis. Four hundred and twenty five individual change requests were collected from three embedded units of analysis, that were consolidated into a dataset repository. The researcher's study spanned an elapsed time period of eighteen months.

Research Framework A was designed in order that areas of change may be determined for WIS's. The conjecture of the framework was that the IS 'areas of change' categories as developed by Fitzgerald et al. (1999), was applicable for the purposes of this research. The research framework was designed and applied with a view to determine the similarities and differences in context of maintenance change in IS and WIS. From this perspective Research Framework A clearly achieved this goal. Section 4.3.4 outlines the implications of the results from this study that supports the argument that there is indeed a difference in the nature of maintenance change between IS and WIS.

However, a decision was taken by the researcher to adapt the design of Research Framework A in order to reveal a greater granularity and insight into the nature of WIS maintenance changes. As a consequence of this an additional sub-research question was included within research question one, namely that of "What are the characteristics of change?". The emphasis of the research reciprocally evolved in the direction of understanding the nature and characteristics of the phenomenon of WIS maintenance change to a greater degree. It was understood that this deeper understanding was required as a perquisite to adequately answer the second research question pertaining to the facilitation of maintenance change by using flexibility. As a result research question two evolved into a conceptual rather than empirical study into the flexibility mechanisms that could assist WIS maintenance change. The categorisation process within Research Framework A had to be fundamentally altered from that of being definition based to that of an emergent, bottom-up method for the purposes of category building. This change in research direction highlights an important assertion that research by its very nature is evolving. Eisenhardt (1989) asserts that research objectives change during the course of its action.

Research Framework B was designed and implemented to address the above requirements. The result of the categorisation process saw the emergence of twelve WIS change group categories contextualised within the scope of this study. The emergence of the change group categories, afforded the researcher with purposeful information as to what areas of the WIS were susceptible to change. As a control feature the researcher incorporated a validation for these detected change group categories within the scope of Research Framework B. This evolved research framework was able to provide the level of granularity regarding areas of change (emergent categorisation process) as well as a greater insight into the characteristics of change (using constructs of analysis) to investigate the WIS maintenance change further.

The prepared dataset was used to produce results and analysis in the proceeding Chapter 5.

Part III ▪ Results, Analysis and Implications

Part III of the thesis relates to the results generated from the operation of Research Framework B. It is composed of two chapters, the first relating to the analysis of the results to answer the research question regarding the nature of WIS maintenance change (Chapter 5). The second chapter being devoted to a further analysis component that investigates how WIS maintenance change can be facilitated by the use of flexibility to address the additional research question (Chapter 6).

To form an adequate base dataset from which to draw conclusions on, Chapter 5 consolidates the data collected from all three project cases into one coherent repository source. The analysis of this source data was differentiated into three clear stages. Stage one of the analysis focused on the areas of change indentified within the scope of Research Framework B, which resulted in the emergence of twelve distinct categories of change. Their relative distributions and attributes were subsequently identified. Stage two of the analytical process viewed the source data from the perspective of the change construct analysis framework (Chapter 3). Trends and anomalies based on the dimensions of this framework are shown in isolation. The final stage of analysis involved the evaluation of the two most significant areas of change with respect to the change construct dimensions. To present the complex cross relational findings relating to the complexity construct, a qualitative model was developed. This qualitative model further evolved to incorporate additional attributes of the analysis. The interrelationships were studied at length to reveal a holistic and rounded view of WIS maintenance change that was necessary to answer the first research question.

Chapter 6 further condenses the change findings and develops an additional model that summarises the nature of change succinctly (change profile model). To answer the second research question, a conceptual framework was designed (Flexibility Matrix of Change) that allowed the mapping of the change profile model to reciprocal flexibility dimensions that would facilitate the implementation of these changes. This process allowed for higher level implications and inferences to be made regarding flexibility and the maintenance phase within a WIS context.

Chapter 5 ▪ Results and Analysis of Web Information Systems Maintenance Change

5.1 Introduction

The previous chapter reported upon the operationalisation process of the research strategy. It was deduced that an alternative strategy to the method of inquiry was required to produce a valid dataset (Section 4.3.4). Chapter 4 specifically addresses the framing of the research case study context and the preparation of the data collected into a consolidated dataset repository. The dataset repository as detailed in Section 4.4.2 was constructed by using the Microsoft® Office Excel spreadsheet tool and is used as the reference point to perform detailed analysis. Section 4.4.1 also considered the boundaries of constituent project cases investigated as embedded units of analysis within the single case study. The empirical dataset repository used for analytical purposes was specifically collected and produced from the application of Research Framework B. The collated change request dataset repository was the primary source from which analytical examinations were conducted for the purposes of this chapter. The maintenance changes studied are grounded in the context of the individual project cases as well as WIS in general. The change request dataset repository is the subject matter to be further elaborated upon within this chapter and conveys quantitative data that affords the researcher with an enhanced understanding of WIS maintenance change. The aim of this chapter is to present detailed analysis on the data collected.

Sections 5.2 and 5.3 reintroduces the identified WIS areas of change (AoC) group categories and the construct of analysis framework, respectively. Table 5.1 presents the definitions of the emerged WIS AoC group categories supplemented with examples. Analysis conducted on the data and the interrelationships between the change construct dimensions are presented in Section 5.3. Section 5.4 shows the distribution and detailed analysis of the AoC group categories. Section 5.5 describes the detailed analysis and results derived when viewing WIS maintenance change from the change constructs perspective.

Based on the repository dataset source, the investigation progresses onto conducting analysis upon the prominent AoC group categories for WIS's (Section 5.6). Data analysis is performed for the identified AoC group category of TUI Content and a synopsis listing the key trends for the Functions AoC group category is shown. These two change group categories were selected to exemplify the process of studying the WIS AoC. As shown in Table 5.2, these two categories occupy the first and second most important AoC based on their frequency of occurrence. In order to provide a holistic and rounded view of WIS maintenance change these two most significant AoC group categories were correlated to the change constructs and interrelationships were studied. The section also provides some contextual explanations to the emerged trends and anomalies induced by individual project cases that give rise to the reported results. This analysis further distils and clarifies the characteristic nature of WIS maintenance change.

Lastly, Section 5.7 concludes the chapter with a summation of some of the key findings purported from the empirical data. Figure 5.1 diagrammatically shows the structure of this chapter.

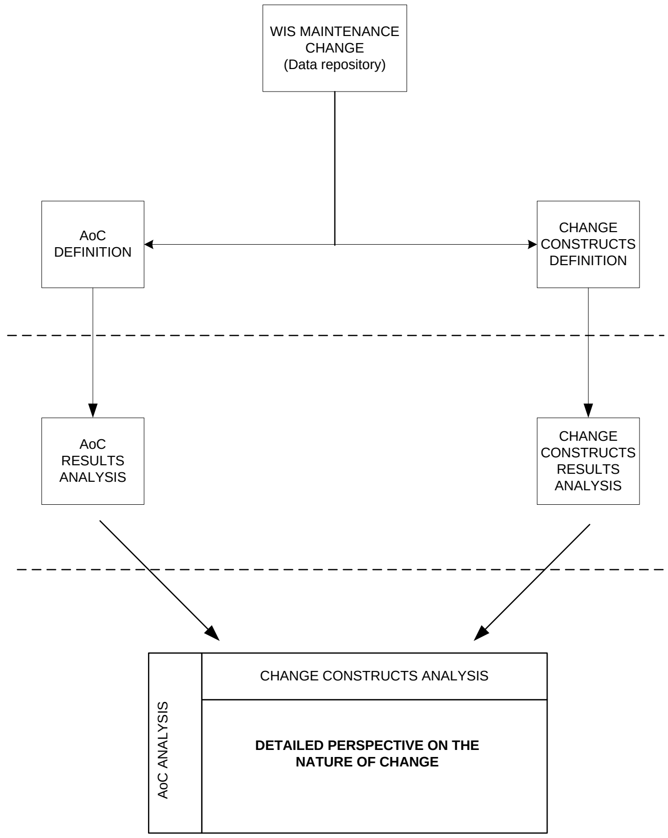

Figure 5.1 – Summary of analysis progression

5.2 The AoC Group Categories for WIS's

Prior to any analysis occurring, the dataset repository had undergone all the data preparation processes within the scope of Research Framework B. Categorisation of the base unit area of change was conducted using clustering and pattern matching techniques (Section 4.4.2.2) in order to categorically sort and arrange similar base units to groupings of a broader hierarchical structure. The total number of emerged change group categories was twelve for the entire WIS case study investigation. It was this classification of AoC group categories for WIS upon which the conduction of analysis is performed and results produced.

Table 5.1 presents the AoC group categories, definitions and examples from the dataset repository. For detailed information relating to the distribution of AoC per project as well as aggregated across all projects is provided in Appendix C.

AoC group category	Definition	Dataset repository examples Change Request I.D (Appendix C)
Image	A graphical visual component of the GUI – e.g. graph, photograph, picture.	WFM90, WFM91, WFM111, LJ21, LJ28b, LJ29a, ABN63a, ABN67, ABN69
Function	A procedural component e.g. pop up function.	WFM70, WFM79, WFM96, LJ6b, LJ13, LJ17, ABN64, ABN76, ABN78
Admin Function	Any function that is used to Administer the application e.g. Setting up user permissions.	WFM26, WFM43, WFM52, WFM74, LJ2c, LJ7, LJ16, LJ19, LJ30d
Navigation Function	Functional operations on GUI e.g. underlying function of the hyperlink.	WFM8, WFM16, WFM53, WFM62c, WFM85
Textual Navigation Hyperlink	String object that allows manoeuvre on GUI e.g. text based hyperlink.	WFM113, WFM114, WFM115, LJ9, LJ23b, LJ29e, ABN1, ABN9, ABN16
Multimedia Hyperlink	Visual component that allows manoeuvre on GUI e.g. image based hyperlink.	WFM19, ABN103
Scrolling Device	Aesthetic control widget allowing navigation on the webpage.	WFM1, WFM22d(i), WFM49a, ABN51b, ABN82, ABN139
Textual User Interface Content Format (TUI Content Format)	Formatting of non-hyperlink string object presented on GUI.	WFM54, WFM68, WFM99, LJ4a, LJ11, LJ28a, ABN2a, ABN37, ABN62
Textual User Interface Content (TUI Content)	Editing of non-hyperlink string object on GUI.	WFM104, WFM105a, WFM106a, LJ1a, LJ1b, LJ2a, ABN68, ABN73, ABN75
Database Data	Stored information in fields contained in the database.	WFM27, WFM30, ABN5, ABN65, ABN66, ABN83,
Style Issue	Issue relating to the look and feel of the GUI presentation e.g. reduce spacing on web page.	WFM46, WFM50, WFM51, LJ4b, LJ5, LJ6a, ABN116a, ABN117, ABN134
Business Analysis	Business change that requires analysis only, requiring no immediate system development.	ABN14

Table 5.1 - Definitions and examples of the AoC group categories for WIS's

5.3 Change Construct Analysis Framework

Section 3.5.5 introduced and established the analysis framework to be used to study WIS maintenance change. The constructs of change, their associated dimensions and units of analysis are shown in Figure 5.2. Most of the results are quantified using the frequency of occurrence of each of the dimensions, an exception to this is the metric of effort. The effort data was collected and measured in terms of man-hours spent on resolving and implementing the required change request. This metric was dichotomised into two types, namely an effort value (quantitative) and an effort rating (qualitative). The effort value was deduced on the average (mean) time calculations for each area of change group categories. This result was supplemented by the calibrated effort rating of low, medium-low, medium-high and high values (Section 4.4.4).

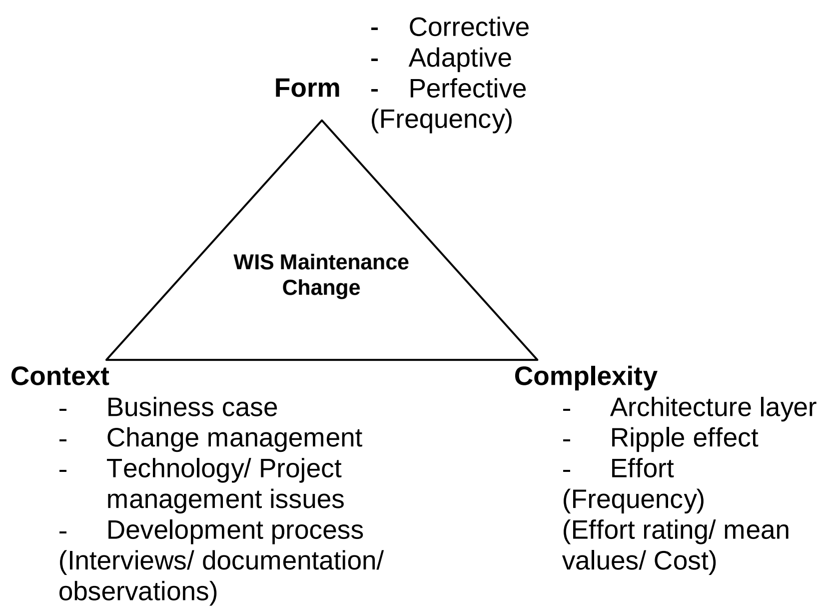

Figure 5.2 – Units of the change construct analysis framework

The purpose of this analysis was to gain a deeper insight and a second perspective into the nature of WIS maintenance change. Section 5.5 describes the analysis of the change constructs (Form, Complexity and Context) when studied individually.

Section 5.6 goes on to analyse the two most significant AoC group categories identified (Section 5.4) with respect to the interrelationships with the change construct dimensions. These interrelationships are shown in Figure 5.3, and summarised below. The context dimension was used to provide trend explanations.

- Ripple Effects taking place on the various Architecture (structural) Layers.
- Maintenance change Forms occurring on the Architecture (structural) Layers.
- Effort involved for Ripple inducing changes.
- Effort involved with changes on specific Architecture (structural) Layers.

Detailed information relating to the distribution of change constructs per project is provided in Appendix E1-E3.

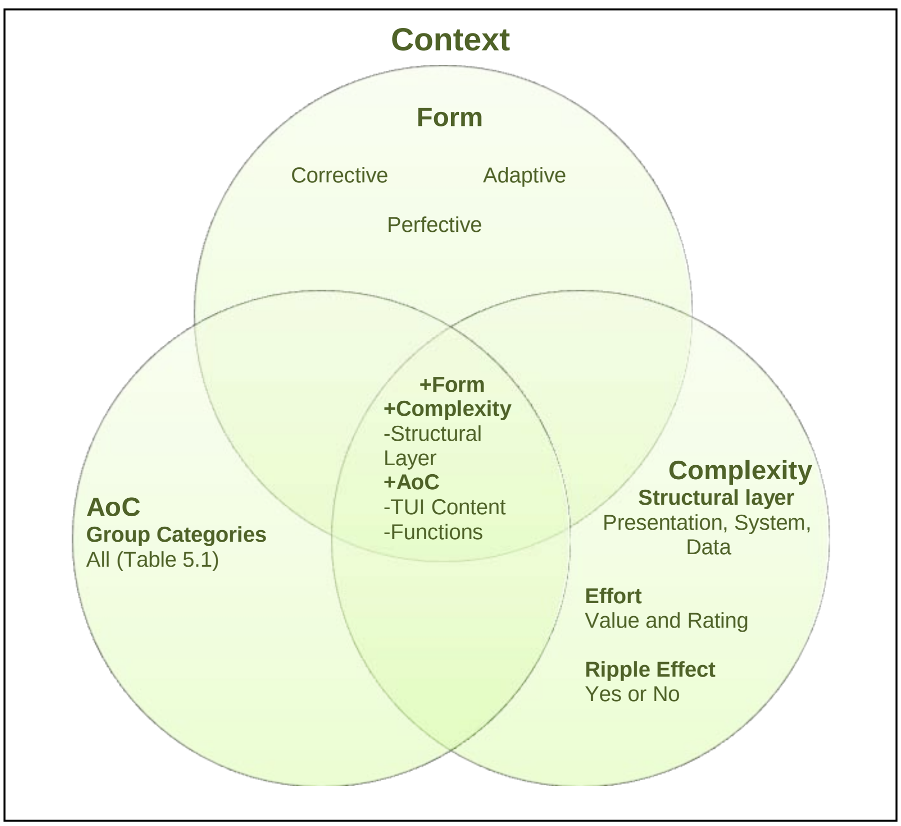

Figure 5.3 – AoC and change construct interrelationships

5.4 Identifying and Investigating WIS AoC

A total of 425 individual WIS maintenance change requests were collected to create the dataset repository (Appendix C). This data originated from three project cases and the distribution is presented in Figure 5.4.

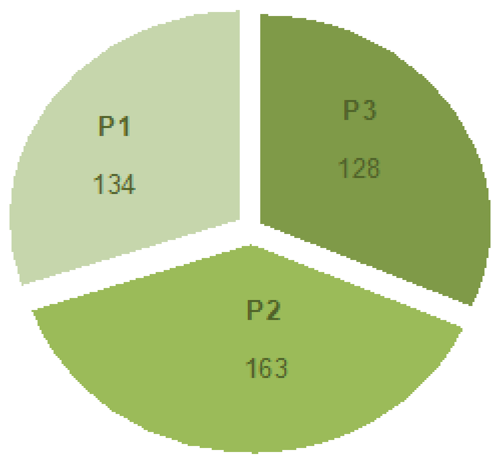

Figure 5.4 – Distribution of change requests across project cases

These were further classified and grouped into one of the twelve change group categories identified in Table 5.1. The twelve emerged change group categories were sorted on the basis of its relative weighting using the frequency metric across the dataset repository. Identifying the primary AoC that require attention from a WIS and business perspective was thus an important consideration to be made. The criteria to judge the primary AoC within this research was based on the frequency of occurrence for each change group category. This was supplemented with identifying change group categories with an arbitrary threshold of greater than 10% to distinguish the prominent AoC from the dataset. Setting this filter allows the researcher to focus and analyse the areas of the WIS that had experienced the most change. Based on this criteria it can be shown from Table 5.2 that the key AoC for the WIS's investigated are the change group categories of TUI Content, Functions, Textual Navigation Hyperlinks, Image and Style Issue. The top two change group categories will be discussed in Section 5.6.

AoC Group Category	Ranking	Frequency/ % (2dp)	
TUI Content	1	105	(24.71%)
Function	2	79	(18.59%)
Textual Navigation Hyperlink	3	61	(14.35%)
Image	4	58	(13.65%)
Style Issue	5	50	(11.76%)
TUI Content Format	6	29	(6.82%)
Admin Function	7	17	(4.00%)
Database Data	8	12	(2.82%)
Scrolling Device	9	6	(1.41%)
Navigation Function	10	5	(1.18%)
Multimedia Hyperlink	11	2	(0.47%)
Business Analysis	12	1	(0.24%)
Total		**425**	**(100%)**

Table 5.2 - Frequency of change group categories across the dataset repository

The distribution of the dataset repository as per the change group categories across the investigated project cases are graphically shown as a bar graph in Figure 5.5. Independent of context, the WIS project cases indicate patterns and trends across the whole dataset; for example TUI Content remains a consistently high occurring area of change across all three WIS projects cases investigated.

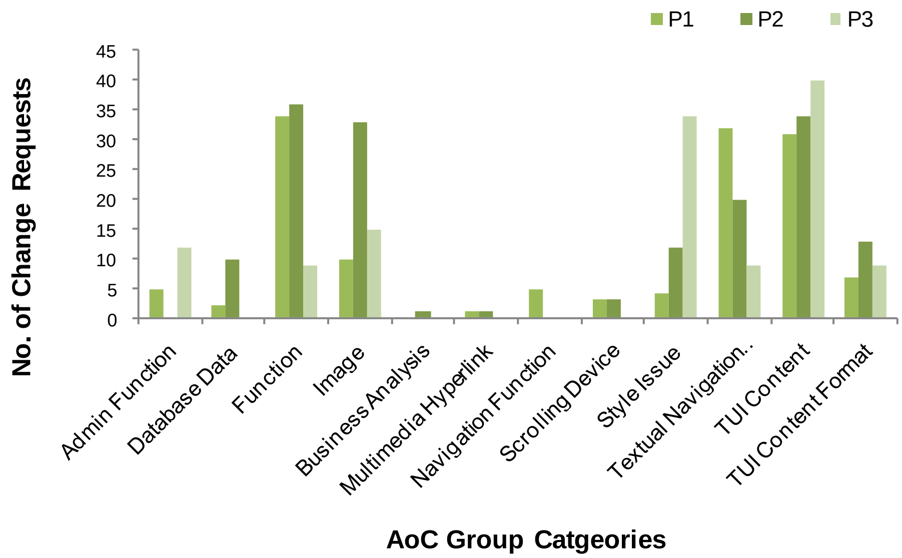

Figure 5.5 – Distribution of change group categories across the three project cases

5.5 Analysis of Change Constructs

This section will consider WIS maintenance change from the change constructs perspective. Each dimension of the change construct will be analysed individually to reveal trends and anomalies as well as higher implications. In order to conduct this individual analysis each dimension will use its own metrics as per Figure 5.2.

5.5.1 Form of WIS Maintenance Change

From the dataset repository, it was discovered that both Corrective and Perfective traditional maintenance types had an almost equal number of instances, 23.06% and 24.94%, respectively.

Source	Form (% - 2dp)		
	Adaptive	Corrective	Perfective
WIS Maintenance	**52% (221/425)**	**23.06% (98/425)**	**24.94% (106/425)**
Fitzgerald et al. (1999)	30%	28%	42%
Boogaard (1994)	30%	15%	55%
Lientz and Swanson (1980)	25%	20%	55%
Lientz et al. (1978)	18%	17%	60%

Table 5.3 – Comparison of Form maintenance types

The key distinguishing aspect for the Form of maintenance change for WIS is the high percentage value for the Adaptive type of change (52%). Another dominant trend that emerged from the dataset was the relatively reduced number of Perfective types of change that were introduced to enhance the system (24.94%).

From the studies presented in Table 5.3, Corrective maintenance types of change was found to be the smallest Form. The IS literature (Table 5.3) concerning these types of change have suggested that Corrective maintenance was found to be in the range of 15-28%. The Adaptive maintenance change types were found to be in the range of 18-30% with Perfective types being in the range of 42-60%. IS literature suggests that the distribution of the Form of change within maintenance has been relatively constant with this finding being confirmed by various studies. The results from this investigation however show a marked variance when comparing individual Forms of change to these studies.

Source	Enhancements Adaptive + Perfective (%)
WIS Maintenance	**76.94% (2dp)**
Fitzgerald et al. (1999)	72%
Boogaard (1994)	85%
Lientz and Swanson (1980)	75%
Lientz et al. (1978)	78%

Table 5.4 – Adaptive and Perfective maintenance change types

Interestingly, when considering Perfective and Adaptive changes as a whole the results from this investigation proved to be consistent with previous IS investigations. Collectively both the Adaptive and Perfective maintenance types of changes for WIS amount to 76.94% (Table 5.4). The results of a study within a WIS maintenance context are listed in Table 5.5. The reasons why these results have not been included as part of the comparison benchmark is primarily because the categorisation did not adhere to those Form types listed in Table 5.3 (see Section 3.5.4), additionally the researcher also wanted to emphasis differences between traditional IS and WIS. Assuming that the Preventative change requests detected are a subset of either (or both) Adaptive or Perfective types then the addition of these three categories also amount to 78%. This further confirms the finding that a super set Form type of 'Enhancements' has a consistent distribution when considering maintenance change either in a WIS or IS context.

Source	Form (%)			
	Adaptive	Perfective	Preventative	Corrective
Scharl (2000)	40	20	18	22

Table 5.5 – WIS study including additional preventative type of maintenance

5.5.1.1 Adaptive Maintenance Type

Table 5.6 indicates those AoC group categories that had a high proportion of change requests relating to the Adaptive Form. These relate to editing, formatting and styling issues pertaining to existing user interface components of the WIS. It is sometimes difficult to differentiate between the Form types of Adaptive and Perfective as both can be considered as enhancements to the system (Takang and Grubb, 2003). However, perfective maintenance changes did not seem appropriate (i.e. editing, formatting or style issues) as no major reprogramming was necessary and no additional aspects or components to the existing infrastructure was included/ added. Analysing a specific example of an Adaptive type of change (LJ62 - Investment challenge section - text content edit); this cannot be considered as a Perfective type because it was the manipulation of an existing aspect of the system and not the extension of the software beyond its original functional requirements. The change request instances were recognised as being induced by environmental pressures and thus were considered as Adaptive. In the particular case of the group category of Database Data these were mostly of the Adaptive Form of change. They specifically relate to financial reporting PDF generation that emerged as change requests in response to environmental fluctuations. These fluctuations were dependent on the end client requirements for chronologically based reports.

Further consideration of the Adaptive Form of maintenance change for this research was significantly higher (52%) when compared to other studies which found this type to be in the range of 18-30%. The results intrinsically suggest that the WIS's need to be more reactive to environmental conditions. This view is further supported by an independent WIS study into maintenance change (Table 5.5). This study reported that the Adaptive Form of changes occurred at a frequency of 40%, indeed this figure may well have been higher had it not further differentiated the additional category of Preventative. Nevertheless both findings show a much higher occurrence of Adaptive change when compared to traditional IS studies. The ubiquitous user base and interactive functionalities that WIS's can offer compared to IS's, suggest that the scope of influencing environmental factors that apply to WIS's are greatly increased. The analysis suggests that WIS's need to increase their responsiveness to the change of environmental factors for the highlighted AoC group categories listed in Table 5.6.

AoC Group Category	**Frequency (No./ % - 2dp)**		**Form Adaptive (No./ % - 2dp)**	
Admin Function	17	4%	-	-
Function	79	18.59%	18	22.78% (18/79)
Image	58	13.65%	42	72.41% (42/58)
Business Analysis	1	0.24%	-	-
Navigation Function	5	1.18%	2	40% (2/5)
Style Issue	50	11.76%	35	70% (35/50)
Scrolling Device	6	1.41%	3	50% (3/6)
TUI Content	105	24.71%	65	61.9% (65/105)
Textual Navigation Hyperlink	61	14.35%	20	32.79% (20/61)
Multimedia Hyperlink	2	0.47%	1	50% (1/2)
TUI Content Format	29	6.82%	25	86.21% (25/29)
Database Data	12	2.82%	10	83.33% (10/12)
Total	**425**	**(100%)**	**221**	**(52%)**

Table 5.6 – Distribution of Adaptive changes by AoC

Overall the Adaptive types of changes recognised can be considered as manipulation or even reconfigurations of existing components. The identified AoC group categories highlighted in Table 5.6, shows that the modifications are relating to presentational aspects and the Database Data category. This gives the notion of the term maintenance in its truest sense of the word, that of the upkeep, update or replacement of existing components. The concise Oxford dictionary defines maintenance as the process of maintaining or being maintained (Oxford Corpus, 2008). However, Boogaard (1994) argues that maintenance in IS is a misnomer and it is a process of maintaining, these results indicate that indeed the maintaining *as-is* aspects of the site is an important consideration within WIS; such that the relevance, accuracy and conformance of the WIS is maintained. Thus, the Adaptive type could potentially be thought to include a sub-type of change defined as *reconfigurative* in the WIS context. For example, TUI Content changes are not necessarily new requirements but require continuous configuration procedures. The TUI Content change can be perceived as a new requirement nevertheless fundamentally the system remains the same. Therefore it does not conform to being an absolute new requirement that changes the dynamics of the web application. Replacement activities of changing existing components is also recognised by Kitchenham et al. (1999) as a sub-type of enhancements. Chapin et al. (2001) also under the definition of business rule refers to this as a reductive classification.

Indeed these results show that Adaptive Forms of change are emerging within the early stages of the maintenance phase.

5.5.1.2 Perfective Maintenance Type

The results produced within this research indicate that 24.94% of changes were of the Perfective type. This result is considerably less when compared to the previous studies presented in Table 5.3. The dominant trends across the AoC group categories identified for WIS relate to firstly Admin Function (76.47%) and secondly Functions (51.9%) (Table 5.7). It was noted that although the Business Analysis AoC displays a 100% of changes relating to the Perfective type, this result will be disregarded due to the limited occurrence for this AoC (only one detected within the dataset repository).

AoC Group Category	Frequency (No./ % - 2dp)		Form Perfective (No./ % - 2dp)	
Admin Function	17	4%	13	76.47% (13/17)
Function	79	18.59%	41	51.90% (41/79)
Image	58	13.65%	5	8.62% (5/58)
Business Analysis	1	0.24%	1	100% (1/1)
Navigation Function	5	1.18%	2	40% (2/5)
Style Issue	50	11.76%	5	10% (5/50)
Scrolling Device	6	1.41%	2	33.33% (2/6)
TUI Content	105	24.71%	11	10.48% (11/105)
Textual Navigation Hyperlink	61	14.35%	24	39.34% (24/61)
Multimedia Hyperlink	2	0.47%	-	-
TUI Content Format	29	6.82%	-	-
Database Data	12	2.82%	2	16.67% (2/12)
Total	**425**	**(100%)**	**106**	**(24.94%)**

Table 5.7 - Distribution of Perfective changes by AoC

The studies presented in Table 5.3 show that the Perfective type of changes range from 42-60%. In comparison to these IS investigations, the results produced from this research indicate that almost 25% of Perfective type of changes were occurring for the WIS's. Clearly a large difference can be seen.

These findings lead the researcher to infer that WIS's require less development effort (i.e. less change requests) to incorporate new requirements (See Section 3.5.4 for an explanation of the Perfective Form). It would appear that the new requirements originating from the end client are being implemented to the WIS with a reduced development effort, suggesting that the WIS's analysed can facilitate change more easily than traditional IS's. Following this finding through suggests that the WIS's studied within the context of this investigation exhibit a greater degree of inherent flexibility than traditional IS's. The theme of inherent flexibility within WIS's is further developed in Chapter 6, however, using Knoll and Jarvenpaa's (1994) dimensions of flexibility definitions, it could be deduced that these WIS's show a degree of versatility and polyadjustable flexibility; resulting in fewer Perfective change requests emerging. It can be further inferred that the AoC group categories that did exhibit a high degree of Perfective changes (Admin Function and Functions, Table 5.7), do not share this innate flexibility. Therefore there is scope to increase the flexibility potential particularly for the functional components of WIS's.

A caveat to the results produced from this study is the length of observed period of the maintenance phase. In order for the maintenance phase to indicate the Perfective Forms of change, a longer period of investigation may be required. Nevertheless, a clear trend is apparent that Perfective Forms of change are occurring at a lower frequency, within the scope of this research.

5.5.1.3 Corrective Maintenance Type

The Corrective maintenance Form of change shows a similar trend between traditional IS (17-20%) and WIS (23.06%) results. Although similar there was a slightly higher occurrence of Corrective Form changes for WIS's. This could be as a result of a lack of testing and quality assurance that occurred during the development phase. Indeed it was observed at the WS supplier that only two to three weeks was spent out of a total of six months of the development lifecycle on this process. The Corrective Forms of change are those changes that are required to fix bugs, defects and errors. Corrective type changes can emerge due to mistakes by the developer or by the regression of the system as multiple change requests are applied to the WIS. These directly or indirectly can have an effect on other parts of the system (ripple effects).

Within the traditional ISD paradigm, systems are assumed to have met their specifications once the application has been operationalised (gone live). The requirements of the system are documented at the out-set of the development and when the system has been made operational it is assumed that it conforms to the originally specifications. In order to validate the system, quality assurance testing mechanisms are conducted to make sure the system works in accordance with its specification, hence all corrections to the system should be ‘ironed out’ during the final stages of development and certainly before the system ‘rolls-out’ to the end clients.

What governs the completion of a project is the hand-over criterion of many IS’s and particularly with WIS’s it is the site operational date. In practice very few projects ‘go-live’ with 100% of corrective changes complete or resolved. There is always a trade-off between whether end clients can live with a few low priority corrective changes outstanding rather than to increase the time to market**.** It was observed that the end client agreed to go-live even though minor defects were known. In the projects studied (a conditional go-live sign off) was given on the proviso that these defects would be resolved within an agreed time frame.

An implication that can be drawn is that programming and development methodologies for WIS’s are not more efficient to their predecessors. The results do not indicate improvement in efficiency or quality gains within the maintenance phase of WIS’s. The results also suggest that testing and quality assurance issues during the development cycle have not improved substantially. These results indicate the constant and non-context specific (i.e. IS or WIS) necessary evil of corrections that are encountered within the maintenance phase.

5.5.2 Complexity of WIS Maintenance Change

The complexity issues involved with maintenance change has long been recognised as a problematic arena ranging from management, technology and sustainability (Harrison et al., 1982; Kafura and Reddy, 1987; Gill and Kemerer, 1991; Banker et al., 1993; Bhatt et al., 2004; Candell et al., 2009; He and Carver, 2009). The dimensions of Structural Layer, Ripple Effects and Effort were used to analyse the complexity involved with WIS maintenance change (Section 5.3).

5.5.2.1 Change Complexity by Architecture Layer

The dataset results produced from this research indicate that the Presentation Layer has the vast majority of change requests affecting it, 61.18%. A further 32.71% of changes were impacting upon the System Layer with the remaining 6.12% on the Data Layer. This suggests that the Data Layer is a stable component of the WIS's and that 93.89% of WIS maintenance change requests will impact either the Presentation or System Layers.

Presentation Layer

The Presentation Layer is often perceived as the interface aspect of software systems between users and the backend processing components of the IS. The WIS architecture is based upon a client-server environment where the processing aspects are mostly based on the server side, however recent WIS trends have resulted in more sophisticated presentational layer computation using Javascripts, ActiveX and Java Applets. The web environment is a medium to communicate and organise often diverse information from multiple sources in a coherent, homogenised and intuitive fashion. WIS's rely heavily on the GUI, making the common day browser an integral and essential part of these systems. This client side prominence has given rise to an increased acknowledgment of usability issues and it is now recognised in its own right as a discipline within WIS (Bieber and Isakowitz, 1995; Kirovski et al., 2001; Rubart, 2008). It was observed that at the case study WS Supplier, designers and front end developers had very distinct skill sets from developers working on the middleware and database components.

As a consequence of these issues the finding that most WIS maintenance changes relate to the Presentation Layer (61.18%) is not surprising. Examples of the AoC group categories effecting the Presentation Layer are Images, Multimedia Hyperlink, TUI Content, Style Issues and TUI Content Format.

TUI Content AoC is one group category that is highly isolated on the Presentation Layer (88.57%). This finding could be attributed to web site ‘content’ being fundamental to WIS’s function. Contrary to this finding the AoC category of Function, was identified as having a minimal presence upon the Presentation Layer. This could be attributed to the view that functional processes are more likely to reside on the Systems Layer. For the Textual Navigation Hyperlink AoC, it was discovered that 40.98% of change requests were affecting the Presentation Layer. This signifies that a sizeable amount of these types of change requests are likely to affect the Presentation Layer. Further analysis on the change requests upon the Presentation Layer were relating to the renaming, formatting and editing of textual hyperlinks. This once again highlights the significance and relevance of the GUI aspects of WIS’s.

Systems Layer

Almost a third (32.71%) of all change requests studied as part of this thesis were isolated to the Systems Layer of the WIS’s. The dominant AoC category influencing the Systems Layer is that of Functions (94.94%). In terms of the client-server architecture for WIS’s, most procedural processing (functional calls – induced by the interface components or other subsystems) occurred on this layer. Therefore it is appropriate for the Functions AoC category to reside and have an effect on the Systems Layer. This relates predominantly to ‘server side’ processing.

Another AoC worthy of consideration on the Systems Layer is that of the Textual Navigation Hyperlink group category (60%). The Textual Navigation Hyperlink AoC in this study is commonly referred to in the academic literature as hyper-links (Garzotto et al., 1995; Rossi et al., 1997). This category specifically addresses the textual forms of hyperlinks which can be distinguished from the Multimedia Hyperlink AoC. The Textual Navigation Hyperlink category has both presentation predicates such as textural descriptions as well as functional attributes in the form of path coordinates that direct the users to relevant destinations within the WIS or beyond. The hyper-linked characteristic is what makes the web infrastructure a unique environment (due to a highly organised linking system that integrates distributed information) as well as making WIS's an intuitive logical tool. The Textual Navigation Hyperlink group category for WIS's indicate that it is the third most significant AoC within the WIS context (Table 5.2). This may be due to the continuous up-dating of the website to include new or amended navigation link paths. Thus the important group categories to consider at the Systems Layer are both Function and Textual Navigation Hyperlink.

Data Layer

The general trend indicates that the Data Layer is the least frequently addressed architecture component influenced by WIS maintenance change. The Data Layer represents 6.12% of overall change requests analysed and is constituted by one main AoC group category, that of Database Data. Some TUI Content AoC affect this layer (10%), however, not enough to draw any firm conclusions on. The Database Data category changes consisted of data integrity and batch file issues, all of which effected the Data Layer (100%). The Data Layer seems to be isolated from change and it can be deduced that it was a stable component of the WIS's studied. One possible reason for this was that the WS Supplier had adopted a data orientated approach to the WIS architecture.

5.5.2.2 Ripple Effects

The results indicate that overall the WIS's studied had an almost equal distribution, with respect to ripple effects (48.71% with ripple, 51.29% without ripple). This result was slightly misleading and what is required is a deeper analysis conducted on AoC group categories. It can be shown from the results that it was the AoC group categories of Functions, Admin Function, Textual Navigation Hyperlink and Style Issue that experienced the most amount of ripple. The categories of Image, TUI Content Format and TUI Content experienced the least amount of ripple effects.

Just over two-thirds of TUI Content category changes did not experience a ripple effect, suggesting that in most cases they are stand alone components displaying no dependencies on subsequent components. Where change requests did have a ripple effect, it was affecting some style or aesthetic issue of the general GUI dimensions. Contrary to this finding are the results produced by the Functions category. These indicated that just under two-thirds of functional type changes will result in ripple effects. Functions are inherently formed as parts of larger processes and are based on inputs and producing outputs. Thus it was not surprising that implementing a function change caused ripple effects to occur. This highlights the particular problems of integrating new requirements for functions into existing systems. The category of Textual Navigation Hyperlink indicated that just over two-thirds of changes on this AoC will produce ripple effects.

Ripple effects are one cause of the degradation of the software systems as developers strive to implement changes with careless regard for the system design and maintainability. Lehman’s law of increasing entropy, states that ‘As a large program is continuously changed, its complexity, which reflects deteriorating structure, increases unless work is done to maintain or reduce it’ (Lehman and Belady, 1985, p. 253). Modular coding methods such as Component Based Development or OO based languages (.NET, Java and style-sheets) can diminish the ripple effects. Efficient testing and quality assurance control (including impact analysis techniques) can also facilitate in detecting such regression inducing changes before they are implemented to the WIS.

5.5.2.3 Effort

The mean effort as measured by time spent (man-hours) was calculated based on the total dataset (425 change requests). However, as a general statistic it can be shown that on average the mean time spent on addressing a change request was 2.15hrs (2hr 9mins). This correlates to the qualitative rating of medium-high range on the cumulative frequency graph (Section 4.4.4). It was evident that the resolution of changes originating from the maintenance phase were consuming a significant amount of time.

In order to provide more granular results for the Effort dimension, the top three prominent AoC categories (Table 5.2) will be used as examples to show the effort required in implementing these changes.

The AoC category of Textual Navigation Hyperlink displayed an overall average effort rating of medium-low. These changes had an empirical mean effort value of 1hr 20mins. On further analysis it was shown that over half of these AoC category changes had a medium-low rating with just under a third of the category having a medium-high. Together they represent 80% of effort being in the range of between medium-low and medium-high and thus suggests a fair amount of time being spent incorporating these types of changes into the WIS. When evaluating the value of frequency of occurrence multiplied by the mean effort (total effort), overall the importance in terms of effort is better exemplified. In the case of Textual Navigation Hyperlink this equates to 61x1.34 = 81.74hrs (81hrs 44mins) to implement all the Textual Navigation Hyperlink type change requests.

For the AoC group category of Functions the overall mean effort equated to 4.38hrs (4hrs 23mins) and had an effort rating of high. This identifies the Function group category as consuming a lot more effort to implement to this type of change when compared to the Textual Navigation Hyperlink category. Further analysis showed that 21% of change requests were in the range of medium-high and that 78% of change requests fell under the bracket range of high. Therefore a lot more changes were indeed consuming a high percentage of effort to implement when considering procedural aspects of the WIS's. When considering frequency of occurrence of the Functions group category and the mean effort, the total effort is 79x4.38 = 346.02hrs (346hrs 1min) which is significantly greater than Textual Navigation Hyperlink changes.

Lastly, the TUI Content AoC category was shown to have an overall mean effort value of 0.75hrs (45mins) per change request with an effort rating of medium-low. This identifies the category as having the least amount of effort when compared to the other two AoC categories discussed in this section. Upon further analysis of the category it was deduced that 55% of changes fell into the low range and that 40% was in the medium-low range. With the highest frequency the TUI Content category can be deemed the most important in terms of the area of the system most susceptible to change. However, when calculating the total effort involved with these changes the result is 105x0.75 = 78.75hrs (78hrs 45mins). This is less than both Textual Navigation Hyperlink and Functions group categories. Even though TUI Content group category changes occurred the most often, it is still relatively easier to incorporate these changes then either the Function or Textual Navigation Hyperlink group category changes based on the total effort involved.

The cost calculation for the maintenance changes is calculated on a time and materials basis that Best & Co. charged their clients. For the three project cases studied the time and material costs is charged at twenty five pounds (sterling) per hour.

Table 5.8 summarises the key findings relating to effort for the three AoC categories discussed.

AoC Change Category	Frequency	Mean Effort (Hrs)	Total Effort (Hrs)	Total Effort (man-days)* *1 man day = 8hrs	Cost (£)
Textual Navigation Hyperlink	61	1.34	81.74	10.22	2043.50
Function	79	4.38	346.02	43.25	8650.50
TUI Content	105	0.75	78.75	9.84	1968.75

Table 5.8 – Effort summary for top three AoC categories

An interesting finding from the data was that from the sizable number of change requests collected (425), the longest effort required to implement a change was only twelve hours. In the context of software development this is a rather low value which indicates the success of reusability and modularity within the WIS development domain. The development team at Best & Co. attempted to reuse the architecture, object model and data schema where possible for their web developments. The development expertise itself was also constant throughout the development and maintenance phases of the three project cases. These factors may have contributed to this finding.

5.5.2.4 Summary of WIS Complexity

A model was designed in order to illustrate how the Complexity construct is influenced by its constituent dimensions (Structural Layer, Effort and Ripple Effects). This model is qualitative in its nature and it presents a holistic view on complexity using data generated from this investigation. How the calibrated measure using the Effort, Structural layer and Ripple Effect dimensions was developed is described below. For reasons stated in Section 3.5.4 it was assumed that changes that take longer to resolve are inherently more complex than changes that take less time to resolve.

The empirical evidence of this research indicates that changes applied to the Presentation Layer had a mean effort value of 1.30hrs (337.5/260), changes applied to the Systems Layer had a value of 3.64hrs (505.5/139) and lastly the changes relating to the Data Layer equates to a value of 2.65hrs (69/26). Based on the mean effort value the ability to rank the Structural Layer components according to the most time consuming and the least time consuming to implement can be deduced. In accordance with the rationale stated above the effects of the Structural Layer dimension on Complexity can be seen in Figure 5.6.

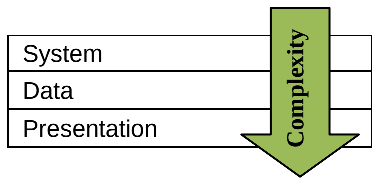

Figure 5.6 – Effect of Structural Layer on complexity

Figure 5.7 shows the relationship between the Structural Layer and Effort dimensions on the calibrated Complexity Scale.

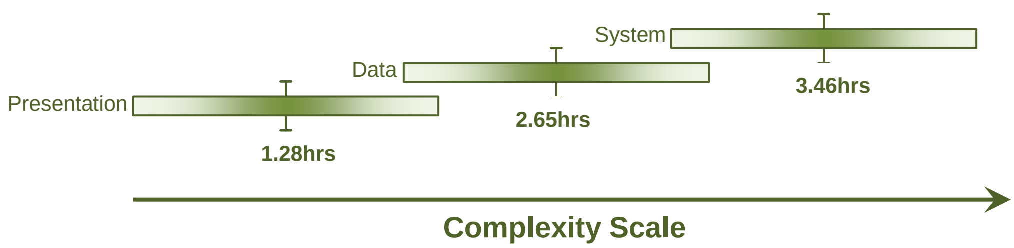

Figure 5.7 – Complexity scale using Structural Layer and Effort dimensions

The empirical data gathered on ripple effects indicates that changes inducing ripple have a higher complexity when compared to changes that do not induce ripple. Table 5.9 shows the findings of this investigation when considering the mean effort value and cost for change requests with ripple and without.

With Ripple	Without Ripple
2.64hrs (546/207) mean effort value	1.68hrs (366/218) mean effort value
£ 66 per change	£ 42 per change

Table 5.9 – Distribution of Ripple Effect

Figure 5.8 displays the evolution of the calibrated Complexity Scale when including the Ripple Effects dimension. The relationships between Ripple Effects, Structural Layer and Effort on Complexity are diagrammatically shown.

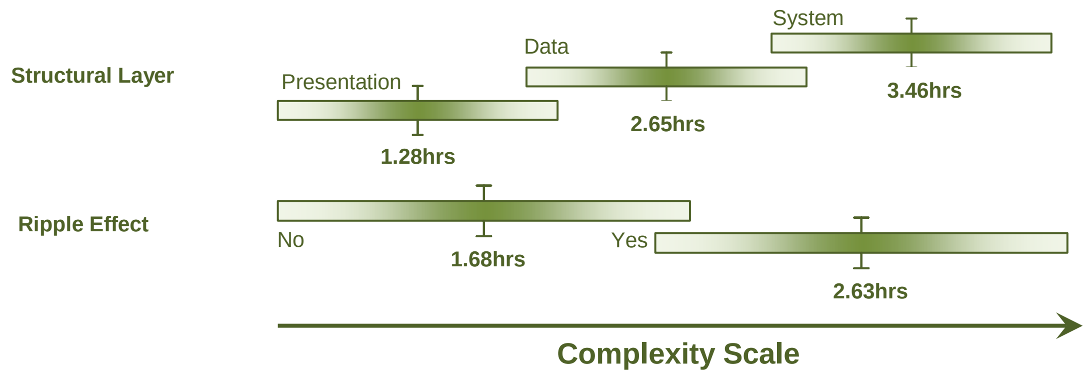

Figure 5.8 – Complexity scale using Structural Layer, Ripple Effect and Effort dimensions

Although the AoC was not a dimension of the Complexity construct, a similar approach can be taken to rank AoC in terms of complexity. Based on the mean effort value (per AoC group category) the ability to rank the AoC categories according to the most time consuming (i.e. the effort involved in resolving the change requests) and the least time consuming can be deduced. By implication if one AoC category takes longer to resolve than another, for example Functions category (mean value 4.38hrs) and TUI Content (mean value 0.72hrs), it can be assumed that the complexity involved with the Function changes are inherently more complex then TUI Content changes. For the purposes of clarity only a sample range from the twelve AoC group categories are shown in the calibrated AoC complexity diagram (Figures 5.9).

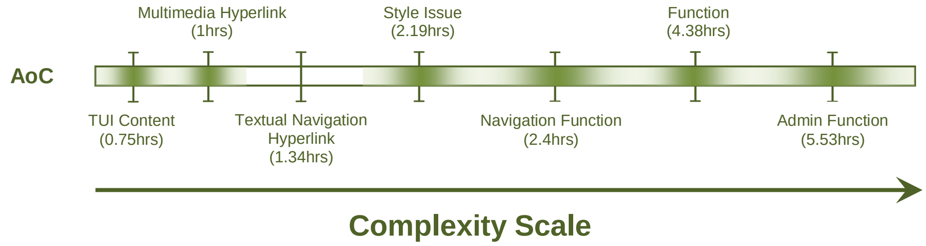

Figure 5.9 – AoC complexity

Figure 5.10 provides a diagrammatical representation of the overall view of change complexity based on the analytical components of AoC and the Complexity construct dimensions. The mean effort (total time spent divided by frequency) required for each AoC group category is shown as well as the Effort, Structural Layer and Ripple Effect dimensions. Based on this information the complexity is shown as a qualitative measure and displays the increasing complexity in sequence for each dimension analysed. Findings from the model suggest that Functions changes are more complex than TUI Content changes. Also the System Layer is shown to be more complex than either the Data or Presentation Layer. Lastly, change requests displaying Ripple Effects are more complex than those that do not have a Ripple Effect.

Figure 5.10 establishes a change complexity model to further enhance the researchers understanding of WIS maintenance change. Subsequently any solutions regarding the facilitation of these changes using flexibility are likely to be more focused and relevant (Chapter 6). The model can also be used to assess how complex a change or group of changes are assuming that the input data relating to AoC, Structural Layer affected and Ripple effects are known.

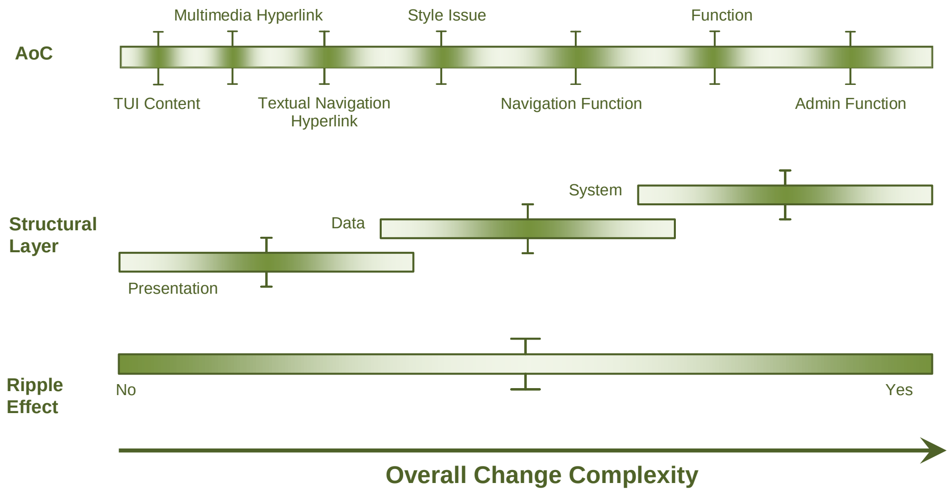

Figure 5.10 – Overall change Complexity Model

5.6 Interrelationships between AoC and the Change Constructs

Detailed analysis was performed upon the top two AoC group categories of TUI Content and Functions. The overall process of analysis is shown for the group category of TUI Content (Section 5.6.1) and a synopsis of the findings is provided for the group category of Functions (Section 5.6.2). Analysis within this section displays the interrelationships across the dimensions of the change constructs with respect to the two AoC mentioned. This holistic analysis was done in order to answer research question one (‘What is the nature of WIS maintenance change?’). Using all the change data at hand as well as gaining a perspective into the interrelationships provides the necessary information to answer this question appropriately.

5.6.1 AoC Group Category 1: TUI Content

The results presented in Table 5.2 indicate that the change activities associated with the group category TUI Content is the most significant AoC. When considered as part of the dataset repository TUI Content changes are the cause of almost a quarter (24.71%) of the total change requests analysed.

AoC Group Category	Project Cases		
	P1	P2	P3
TUI Content	29.52% (31/105)	32.38% (34/105)	38.10% (40/105)

Table 5.10 - Distribution of TUI Content changes across the projects

Table 5.10 reflects the proportion of the TUI Content category changes across the three project cases. These findings consistently show that around a third (between 29.52% - 38.10%) of TUI Content category changes occurred for each of the project cases. The trend suggests that TUI Content types of change are context independent. This may be indicative of the critical nature of WIS's where the temporal accurateness is often paramount to the business model (for example financial reporting systems). This proposition is certainly critical to all technology driven IS's, however it is even more pertinent of an issue within WIS.

5.6.1.1 TUI Content Analysis by Form

Table 5.11 shows the distribution of the TUI Content AoC when considering the Form construct. The results clearly demonstrate that the vast majority (61.9%) of TUI Content changes fall into the maintenance type of Adaptive. Over a quarter of the TUI Content changes were in the Form of Corrective (27.62%) and a lesser 10.47% was attributed to Perfective types of change. The Perfective type is shown to be minimal and thus relatively insignificant to the group category of TUI Content.

Table 5.11 also reflects the breakdown and contributions of the individual Form types of change across the three project cases investigated. In terms of the Adaptive type of change, the results show that almost a quarter of the change requests for TUI Content was attributed to the P3 project case. The second most contributing project within the Adaptive type was P2 (20.95%) and lastly P1 having a 15.23% influence. A similar trend emerges when considering Perfective type changes, although minimal in all three project case instances (less than 10%) the sequence from most to least is P3 (6.67%), P2 (2.86%) and P1 (0.95%). However, when considering Corrective types of change it can be shown that P1 had the most impact (13.33%) followed closely by P2 (8.57%) and lastly P3 (5.71%). It was noted that both Corrective (between 5%-13%) and Adaptive (between 15%-24%) Forms of change had a fairly sizeable occurrence between the three project cases.

Project Cases	Form (% - 2dp)		
	Adaptive	Corrective	Perfective
P1	(16/105) 15.23%	(14/105) 13.33%	(1/105) 0.95%
P2	(22/105) 20.95%	(9/105) 8.57%	(3/105) 2.86%
P3	(26/105) 24.76%	(6/105) 5.71%	(7/105) 6.67%
Total	(65/105) 61.90%	(29/105) 27.62%	(11/105) 10.47%

Table 5.11 – Distribution and comparison of Form across projects

Upon further investigation into the factors and reasons for the above trends, the context of the project cases were re-examined. Firstly, the attributes of the change requests raised within the Adaptive type relating to TUI Content are almost exclusively changes pertaining to the actions of editing, formatting and style configurations. These general aesthetic issues are the key driver as to why such a large proportion of change requests are Adaptive in nature.

Although both the P1 and P3 project specifications incorporated a content management system (CMS) (Friedlein, 2003) into their WIS's, it did not result in diminishing change requests relating to the TUI Content AoC. This gives the notion that although the CMS had given end-users the ability to change content as their business needs dictate, it did not give them the ability to change the styling and formatting issues of the TUI Content AoC. This reflects the limited nature of the CMS in place with particular reference to the P3 project. The absence of a CMS, as seen in the P2 project, results in a significant number of change requests for the category TUI Content.

As noted above, the most Corrective type of changes were taking place within the P1 project case. P1 was considered to be the most complex application developed with several project management issues being observed. In some instances the WS Supplier had to interpret much of the site content material as it had not been clearly defined by the end client. This consequently led to misunderstandings in content specifications that resulted in an increased number of bug fixing and corrections taking place. The P2 project had a short development lifecycle, however with a higher ratio of testing to development. The Corrective changes that were raised by the end client were predominantly due to translation errors. The P3 project was entirely content managed leading to a small number of Corrective changes, however the limits of the CMS meant that not all alphanumeric type changes could be amended by the end client. There may also be evidence to support a ‘learning curve’ by the WS Supplier with respect to implementing TUI Content changes (P1 (13.33%), P2 (8.57%), P3 (5.71%)).

Perfective type changes were overall very low for TUI Content (10.47%). This may be an indication that new requirements are not being generated by the end client for this particular AoC. Further breakdown of the P3 project reveals that the change requests were relating to the additions of new site content to the WIS. However, it seems that generally Perfective type changes are not a significant category for TUI Content.

5.6.1.2 TUI Content Analysis by Complexity

This section analyses the TUI Content AoC, Form and Complexity constructs (Architecture Layer, Ripple Effects and Effort).

Architecture Layer

Presentation	System	Data
88.57% (93/105)	0% (0/105)	11.43% (12/105)

Table 5.12 - Distribution of all TUI Content changes across Architecture Layer's

Table 5.12 indicates the Architecture (structural) Layer upon which the TUI Content AoC category had an effect. Based upon the dataset repository, it shows a clear indication that TUI Content AoC changes predominantly reside on the Presentation Layer with some changes also occurring on the Data Layer. The Systems Layer had no TUI Content changes occurring on it.

Presentation Layer

P1	P2	P3
33.33% (31/93)	36.56% (34/93)	30.11% (28/93)

Table 5.13 - Distribution of Presentation Layer TUI Content changes across projects

Analysis of the Presentation Layer as shown in Table 5.13 indicates that an almost even distribution is generated from each of the three project cases. When considering each project case, the table results show that the project P2 and P1 each contributed over a third of TUI Content changes to this particular layer with the P3 project representing the remaining 30.11%. This was evidence of a strong trend relating to 'TUI Content' AoC being isolated upon the Presentation Layer.

Presentation Layer and Ripple Effects

Investigations into Ripple Effects occurring relative to the total number of TUI Content changes on the Presentation Layer per project case are shown in Table 5.14. The results indicate that for the Presentation Layer over half of the number of TUI Content changes for the P3 project case had a ripple effect taking place. Both the P2 and P1 projects presented low percentages in regards to having a ripple effect taking place. The analysis upon Ripple Effects for TUI Content changes were generally attributed to other graphical components, for example page alignments or generic styling issues. It was observed that the limited CMS that was specified for the P3 project, did not have the capability in this case to incorporate textual type changes without having an effect upon other graphical components of the site.

P1	P2	P3
12.90% (4/31)	8.83% (3/34)	53.57% (15/28)

Table 5.14 - Distribution of TUI Content changes showing Ripple Effects on the Presentation Layer across projects

Data Layer breakdown by Project

Another significant finding from the Architecture Layer statistics shown in Table 5.12, is that of the Data Layer. As shown in Table 5.15 the twelve instances of Data Layer TUI Content changes are skewed by the P3 project, which is contributing 100% of the changes in this area. This is due to the changes that were made to the CMS of the P3 project, that relate to new content managed sections to be included in the CMS structure and thus entailed a database redesign. All the TUI Content changes found on the Data Layer for the P3 project had a ripple effect that consequently affected the GUI. The P2 project was not content managed and thus makes sense that no Data Layer changes occurred for the TUI Content category unless the requirement of a CMS was acknowledged. The P1 project suggests that no major additions or modifications were made to the existing CMS. It is worth nothing that overall the Data Layer experienced minimal TUI Content changes (11.43%).

P1	P2	P3
0% (0/12)	0% (0/12)	100% (12/12)

Table 5.15 - Distribution of TUI Content changes on the Data Layer across projects

Architecture Layer and the relationship to Form of change

When considering the relationship of Architecture (structural) Layers and the Form of change, the most significant Form types of Adaptive (61.90%) and Corrective (27.62%) will be further explained.

Architecture Layer and Adaptive type of change

Form Type	Project Case		
	P1	P2	P3
Adaptive	51.61% (16/31)	64.71% (22/34)	67.5% (27/40)

Table 5.16 - Distribution of Adaptive changes per project

Table 5.16 reflects the number of change requests for the TUI Content category that had the maintenance Form type of Adaptive. Relative to the number of instances per project it can be shown that over two-thirds of the TUI Content changes were of type Adaptive for the P3 project. Just over half of the TUI Content changes were of this Form type for P1 project. Lastly just under two-thirds of TUI Content changes were of form type Adaptive for the P2 project.

Project Case	Architecture Layer		
	Presentation	System	Data
P1	(16/65) 24.62%	(0/65) 0%	(0/65) 0%
P2	(22/65) 33.85%	(0/65) 0%	(0/65) 0%
P3	(16/65) 24.62%	(0/65) 0%	(11/65) 16.92%
Total	(54/65) 83.08%	(0/65) 0%	(11/65) 16.92%

Table 5.17 - Distribution of Adaptive changes by Structural Layer and projects

The total number of Adaptive Form type changes equated to 65 instances for the TUI Content AoC across all the project cases. When considered as part of this aggregated data, most TUI Content changes (83.08%) for Adaptive Form type were found to reside on the Presentation Layer (Table 5.17). The Systems Layer had no effect in terms of the Adaptive Form type for TUI Content changes, with the Data Layer having 16.92% of Adaptive Form type occurring for TUI Content changes.

Breaking down the results on a per project basis, it was noted that all of the Data Layer occurrences are stemming from the P3 project and that an almost even distribution is presented on the Presentation Layer. However, it is clearly visible that the P2 project had the most influence in terms of the Adaptive Form type of change occurring for TUI Content upon this layer (33.85%). The findings relating to the Data Layer are associated with the P3 project case (see section 'Data Layer breakdown by Project' (page 169) for further explanations).

Architecture Layer and Corrective type of change

Form Type	Project Case		
	P1	P2	P3
Corrective	45.16% (14/31)	26.47% (9/34)	15% (6/40)

Table 5.18 - Distribution of Corrective changes per project

The data presented in Table 5.18 provides an insight into the Corrective Form type of changes that were raised for the TUI Content AoC category. It was clear from the analysis that the P1 project had a high proportion of Corrective Form type of changes when viewed independently (45.16%), and that just over a quarter of the P2 project changes for TUI Content was 26.47%. The P3 project had the least number of occurrences of the Corrective Form type of changes happening for the TUI Content AoC category (15%).

Project Case	Architecture Layer		
	Presentation	System	Data
P1	(14/29) 48.28%	(0/29) 0%	(0/29) 0%
P2	(9/29) 31.03%	(0/29) 0%	(0/29) 0%
P3	(5/29) 17.24%	(0/29) 0%	(1/29) 3.45%
Total	(28/29) 96.55%	(0/29) 0%	(1/29) 3.45%

Table 5.19 - Distribution of Corrective changes by Architecture Layer and projects

Breaking down these statistics further on the basis of project cases, it was firstly evident that no or very minimal instances of Corrective Form type of changes for the TUI Content category were occurring on the Systems or Data Layers (Table 5.19). Secondly, further analysis on a per project basis showed that the P1 project maintains a large amount of Corrective Form type of changes (Table 5.19). Almost half of the changes investigated were of this Form for the TUI Content AoC category. Just under a third of the changes of Corrective Form type originated from the P2 project with the remaining 17.24% from the P3 project. Due to the inherent information qualities of the TUI Content AoC category for WIS's, it was not surprising that corrections and bug-fixing change requests reside on the Presentation layer. The P1 project had the highest number of Corrective Form type of changes and suffered most from project management issues (see Section 5.6.1.1 for further explanations).

Ripple Effects

Yes	No
31.43% (33/105)	68.57% (72/105)

Table 5.20 - Comparison of Ripple Effects for TUI Content

Table 5.20 shows the distribution in the form of 'yes' or 'no' of the ripple effect for the AoC group category of TUI Content. The results indicate that over two-thirds of TUI Content AoC did not experience a ripple effect. This suggests that this AoC more often than not are isolated and have a limited capacity to cause cascading effects upon other components of the WIS. In the cases of where Ripple Effects were taking place many of the changes were effecting some form of style or aesthetic issue of the GUI (for example page alignments or new pop-up window parameters).

Ripple breakdown by Projects

P1	P2	P3
12.90% (4/31)	8.82% (3/34)	65% (26/40)

Table 5.21 – Distribution of changes that have Ripple Effects across projects

The overall Ripple Effect trends across the projects indicated in Table 5.21, show that the P3 project (65%) has significantly more changes with ripple then the P1 (12.9%) and P2 (8.82%) projects. A reason as to why TUI Content changes for the

P3 project had significantly more Ripple Effects was discussed in the above section ('Presentation Layer and Ripple Effects' (page 168) and 'Data Layer breakdown by Project' (page 169). If the P3 project was skewing the results and is considered as an anomaly then the non-Ripple Effect would be even more significant for this AoC. It was also noted that both the P1 and P2 projects had few instances where Ripple Effects were detected.

Ripple breakdown by Projects and Architecture Layer

Project Case	Architecture Layer		
	Presentation	System	Data
P1	(4/33) 12.12%	(0/33) 0%	(0/33) 0%
P2	(3/33) 9.09%	(0/33) 0%	(0/33) 0%
P3	(15/33) 45.45%	(0/33) 0%	(11/33) 33.33%
Total	(22/33) 66.67%	(0/33) 0%	(11/33) 33.33%

Table 5.22 – Distribution of changes that have Ripple Effects by projects and Architecture Layer

Table 5.22 further examines the relationship of projects to the Architecture Layer. The results show that with the exception of the P3 project that all change requests for the TUI Content category had Ripple Effects located on the Presentation Layer. In total two thirds (66.67%) of TUI Content category changes on the Presentation Layer induced a Ripple Effect. Just under a half of these changes were pertaining to the P3 project with both P1 and P2 projects displaying a more even distribution of 12.12% and 9.09%, respectively. When considering the Data Layer Table 5.22 indicates that a variance is caused by the P3 project.

<u>Effort</u>

Table 5.23 displays the average effort (mean effort) and the effort rating for TUI Content AoC. It shows that the mean time taken to resolve a TUI Content change was 0.72hrs which correlates to an effort rating of Medium-Low (Section 4.4.4).

AoC Group category	Effort	
	Effort rating	Mean effort
TUI Content	Medium- Low	(79/105) 0.75 hrs 45.14mins

Table 5.23 – Effort rating and mean for the TUI Content AoC

Low	Medium-Low	Medium-High	High
54.29% (57/105)	40.00% (42/105)	4.76% (5/105)	0.96% (1/104)

Table 5.24 – Distribution of Effort for TUI Content changes

It can be deduced from Table 5.24 that the distribution of the effort rating for the TUI Content category was as follows. Firstly, over half of TUI Content changes investigated displayed a Low effort rating with a further 40% for the Medium-Low rating. When combined together the results reveal a very strong trend for TUI Content changes and suggests that almost 95% of changes will have an effort rating between Medium-Low and Low ratings. However, some instances are shown that have an effort rating of Medium-High with minimal ratings of High (0.96%).

Effort rating	Project Case - % (2dp)		
	P1	P2	P3
Low	(11/105) 10.47%	(23/105) 21.90%	(23/105) 21.90%
Medium-Low	(16/105) 15.24%	(11/105) 10.47%	(15/105) 14.29%
Medium-High	(4/105) 3.81%	(0/105)	(1/105) 0.95%
High	(0/105)	(0/105)	(1/105) 0.95%
Mean Effort Value	**(29.5/31) 0.95hrs 57min**	**(18.5/34) 0.54hrs 32.4 min**	**(18.5/34) 0.78hrs 46.5 min**
Cost	**£ 23.75 per change**	**£ 13.50 per change**	**£ 19.50 per change**

Table 5.25 – Distribution of Effort across projects

Table 5.25 presents the breakdown of the effort rating and mean effort for the TUI Content AoC across the projects. Both P3 and P2 projects equate to the same value of just over a fifth (21.9%) of changes having a effort rating of Low. The P1 project has the least instances (10.47%) of Low level effort changes. The Medium-Low effort rating shows even more of a similarity, ranging from between 10.47 to 15.24% across the projects. The identified trend shows that the P1 (15.24%) project had the most number of instances followed closely by the P3 (14.29%) project and then the P2 (10.47%) project. The highest contribution of the Medium-High effort rating is more for the P1 project (3.81%), with the results for P3 and P2 projects being minimal or nonexistent. The P1 project had more Medium-Low (15.24%) instances followed by Low (10.47%) effort ratings then Medium-High (3.81%). Both P3 and P2 projects display the same trend, that of Low, Medium-Low and High distributions.

The results of the mean effort values shown in Table 5.25 identifies a marked difference between the times taken to resolve the changes across the projects, P1 (57min), P2 (32.4min) and P3 (46.5min).

Effort Value	Ripple Effect	
	Yes	**No**
Effort Rating	Medium-Low	Medium-Low
Mean Effort	(28/33) **0.85hrs 51mins**	(51/72) **0.71hrs 42.6mins**
Cost	£ 21.25 per change	£ 17.75 per change

Table 5.26 - Effort of Ripple Effect changes

Table 5.26 presents the results of the mean effort value for TUI Content AoC changes that had a Ripple Effect when compared to those that did not have a Ripple Effect. In terms of this AoC, it can be shown that changes that had a ripple effect did take a longer time to resolve, however, the mean effort statistics between ripple and no ripple changes is relatively small.

Effort Value	**Architecture Layer**		
	Presentation	**System**	**Data**
Effort Rating	Medium- Low	0	Medium- Low
Mean Effort	(67/93) **0.72hrs 43.23mins**	0	(12/12) **1hr 60mins**
Cost	£ 18.00 per change	£ 0.00 per change	£ 25.00 per change

Table 5.27 - Effort of Architecture Layer changes

The effort ratings and mean values of Architecture Layer changes are displayed in Table 5.27. The results show an effort rating of Medium-Low is consistent for the TUI Content AoC that affect the Presentation and Data Layers.

5.6.1.3 TUI Content Analysis Summary

The descriptions that are produced in Table 5.1 indicate the definition for the group category TUI Content. The analysis provides an examination of the nature of the category to afford the researcher with a deeper understanding into the TUI Content AoC. Based on the frequency of occurrence the TUI Content category has emerged as an important facet of the WIS's investigated. As Table 5.2 exemplifies, the TUI Content AoC is ranked as the most prominent change category with almost one quarter of the total WIS maintenance changes (24.71%) falling under this definition. When scrutinising projects on an individual basis, the emergence of the TUI Content category instances are consistently significant throughout (Table 5.10). This strong trend on both an aggregated and project specific basis indicates that TUI Content is likely to be a key area of change in WIS maintenance.

It was deduced that the TUI Content changes effecting the WIS's mostly fit into the Form of Adaptive. Adaptive change was the most common Form of change when considering the total data (61.9%) (Table 5.11) and was in a similar range on a per project basis (51.61% – 67.5%) (Table 5.16). The only other Form of change noteworthy was that of type Corrective. This was found to be 27.62% across the aggregated dataset (Table 5.11). The trend for the Corrective Form type of changes does however vary quite significantly between projects (Table 5.18); for example

45.16% of all TUI Content changes for the P1 project being Corrective with only 15% for the P3 project and lastly the P2 project had a value of 26.47%. The potential skewing of the trend may be explained by quality issues relating to the P1 project and secondly due to a learning curve process across the three projects (Section 5.6.1.1). The vast majority of TUI Content changes of the Adaptive (83.08%) and Corrective (90.55%) Form types were found to reside on the Presentation Layer (Table 5.17 and 5.19).

It was also evident from the results that the vast majority of changes pertaining to the TUI Content AoC reside on the Presentation Layer (88.57%) (Table 5.12). The dominant trend for the TUI Content AoC suggests that it has a direct influence upon the Presentation Layer of the WIS's. The Presentation Layer of WIS's was perceived as the browser based GUI aspect upon which TUI Content components have an influential presence. Just above two-thirds (68.57%) of changes attributed to the TUI Content AoC do not experience a ripple effect upon other components of the WIS's. This statistic would have been even higher if the P3 project did not skew the results (Table 5.21). This point is further demonstrated in Table's 5.14 and 5.22 which specifically looks at the effect of ripple upon the Presentation Layer. It was found that two thirds of the changes on the Presentation Layer had ripple also one third of changes had a ripple on the Data Layer.

The Effort involved in implementing TUI Content changes fell into the Medium – Low range (Table 5.23). A sizeable majority (54.29%) occurred in the Low range with the remaining bulk (40%) in the Medium – Low range. This data suggests that TUI Content changes although the most populous, require a relatively small effort to resolve. When considering the effort involved, it was found that implementing ripple effect changes take longer to resolve. Interestingly the divergence in time between resolving ripple/ non-ripple TUI Content changes is marginal, 51 minutes and 46.2 minutes, respectively (Table 5.26). The Architecture Layer also has an effect on the effort to resolve TUI Content changes. It was found that Data Layer changes took an effort of 60 minutes compared to Presentation Layer changes which took on average 43.23 minutes (Table 5.27).

5.6.2 AoC Group Category 2: Functions (synopsis)

Based on the frequency of occurrence the Function AoC category has emerged as an important type of WIS maintenance change. As Table 5.2 exemplifies the Function AoC is ranked as the second most prominent change category, with almost one fifth (18.59%) of all the WIS maintenance changes investigated falling under this definition.

On further scrutiny of the projects on an individual basis, the emergence of the Function AoC category are significant for both the P1 (25.37%) and P2 (22.09%) projects. However, the number of Function AoC changes are markedly low for the P3 project (7.03%). It was noted that the P3 project had an aggressive growth strategy and had specified a broad range of functionalities that would cater for a projected new customer base. During the maintenance phase observation period (three months), the end client had not managed to mobilise the expected new customers, however, the existing customer base were fully migrated onto the WIS platform. It was also observed that the P3 project was very well specified from the outset in terms of the end clients requirements. This apparent inherent slack (Section 6.3.3) as well as the detailed specifications received may have been the drivers in reducing the amount of functional changes arising during the early phases of maintenance; thus contributing to the anomalous effect when considering Function AoC changes. However, when considering the aggregated data it would appear that the Function group category is likely to be a key area of change in WIS's.

When considering the Form of changes for the Functions AoC category, the results indicated that over half (51.90%) of Functions Form type changes were Perfective. Thus, when considering Function AoC changes it can be deduced that a large proportion will be pertaining to new functional requirements to the WIS. It can also be shown that both the Adaptive and Corrective Form's of change have a similar number of instances for the Functions AoC category, 22.78% and 25.32%, respectively. The Corrective Form of change displays a consistent trend of one quarter (25.31%), as displayed with the TUI Content AoC category.

A clear trend exists regarding the Architecture Layer and the Function AoC changes. It was evident from the results that the vast majority (94.94%) of changes pertaining to the group category of Functions reside on the System Layer. This was a consistent trend even when considering each project individually (P1 (100%), P2 (94.44%), P3 (77.78%)). The finding may be due to the middleware components of WIS's (System Layer) providing most of the computational and logic processing capabilities. When considering the System Layer further, 52% of changes were found to be of the Form type Perfective; Adaptive (22.67%) and Corrective (22.33%) were found to be less significant on this layer. This suggests that the majority of Function AoC changes relate to the addition of new requirements that predominately impact the System Layer.

Based on the aggregated dataset, the Ripple Effects for the group category of Functions indicted that over two-thirds (63.29%) of Function type changes will have an knock-on effect upon other system components. This was consistent with the view that Functions often have dependencies on other functions that ultimately form a larger process. This may explain the finding of why the majority of Function AoC changes resulted in cascading changes throughout the WIS on their implementation. This trend for the Function AoC is also evident when considering the System Layer individually, 61.33% of all changes on the System Layer induced Ripple Effects.

When considering the effort involved in implementing function type change requests, it can be shown that the vast majority of Function AoC changes required effort in the High (75.95%) or Medium-High (22.78%) ranges. As mentioned before it was deduced that the majority of Function AoC changes were Perfective in nature. This suggests that Function AoC changes relate to new requirements that take a considerable amount of effort to implement into the existing WIS. Upon further consideration of Effort it was shown that Function AoC changes that induced ripple took 4.68 hours to resolve on average (Cost per change = £ 117.00). This shows a significant variance to changes that did not invoke cascading effects, which on average took 3.86 hours to resolve (Cost per change = £ 96.50). Changes that caused ripple were ranked as having a High effort rating and this finding was also true with respect to changes that did not have a ripple. In terms of the effects of Architecture Layer on Effort, changes effecting the Data Layer took the longest to resolve, suggesting that these are the most complex. Function AoC Data Layer changes on average took 7.5 hours to resolve (Cost per change = £ 187.50), System Layer took 4.31 hours (Cost per change = £ 107.75) with the Presentation Layer taking the least time of 4 hours to resolve (Cost per change = £ 100.00).

The result tables for the analytical work conducted for the Functions AoC category can be found in Appendix F.

5.7 Conclusion

This chapter analyses the results that were generated from the design and operation of Research Framework B (defined in Chapters 3 and 4), in order to provide a rich contextual backdrop from which to analyse the nature of WIS maintenance change. A source repository was generated using a case study approach that encompassed three projects to create a base dataset. This repository was the primary source of data used to produce analytical results and implications on WIS maintenance change.

The analytical process was differentiated into three clear stages. The first stage related to the analysis of the AoC identified within WIS, stage two analysed the data with respect to the change constructs and its relevant dimensions, and the final stage created cross relationships between the AoC and constructs of change data. This approach was undertaken in order to answer research question one (‘What is the nature of change in WIS maintenance?’). The results produced were able to provide quantitative (empirical) and qualitative results that provided the researcher with a means to quantify and answer this research question. Higher level implications regarding WIS maintenance change were also identified and these were consequently used as inputs in answering the second research question in Chapter 6 (‘How can WIS maintenance change be better facilitated?’).

The construction of Research Framework B allowed twelve AoC group categories to emerge within the scope of WIS maintenance. These twelve categories were defined and their relative distributions were deduced from the source data. The analysis showed that the top five categories were considered as significant and thus proved to be candidates of where potential flexibility solutions should be targeted for WIS's. The top two AoC categories (TUI Content and Function) were shown to be the most dominant of the group and as a result were consequently used as the criteria for the stage three analysis.

Stage two of the analysis considered the findings relating to the dimensions of the change constructs Form, Complexity (Architecture Layer, Ripple Effect and Effort) and Context. The contextual influences inherently form the backdrop to the results and analysis conducted in this chapter. The analysis of change by Form was done in accordance with established parameters defined by Swanson (1976) which created a valid benchmark upon which to compare the findings from this investigation to previous studies. The analysis from this study confirms the findings of previous IS studies on two counts. Firstly the amount of Corrective types of change occurring in WIS maintenance was shown to be 23.06% compared to a benchmark IS range of 17-20%. Secondly, Enhancements to the system (as measured by empirically adding the number of Adaptive and Perfective changes) were shown to be 76.94% in this investigation comparing to results from IS studies that showed Enhancements in the range of between 72-85%. However, when the results for the Form types of change were considered individually a significant variance was detected in the proportion of Adaptive and Perfective type changes. The findings from this study suggest that the Adaptive types of maintenance change are far more prevalent in WIS maintenance when compared to IS maintenance. By the same token Perfective types of maintenance change are less prevalent when compared to IS benchmark studies. A high degree of Adaptive type changes occurring specifically within a WIS context is also supported by Scharl (2000, p. 35). This finding supplements the augment that WIS and IS do exhibit different characteristics and should be considered as distinct entities.

Stage two continues with analysis of the Complexity construct. Complexity of change is often a subjective measure therefore this study uses three dimensions to qualify implications relating to this construct. Change is analysed by the architecture layer they effect, the extent of ripple they cause and the effort required to implement them. This criteria gives rise to the dimensions of complexity that were used to analyse WIS maintenance change within this stage of the analytical process. Each dimension was analysed individually based on the consolidated dataset and the findings can be summarised as follows:

- Key trend observed regarding the Architecture Layer was that the majority of maintenance changes effected the Presentation Layer (61.18%).
- Almost half (48.71%) of all maintenance change observed induced cascading ripple effects.
- Average effort required to resolve a WIS maintenance change was found to be in the Medium-High range with a mean value of 2 hours 9 minutes taken to resolve the change.

A complexity model was devised in order to assimilate the data generated from these dimensions which allowed the researcher to derive implications and inferences on this construct of change (Figure 5.8). This model provides a template on how to ascertain the complexity of a change (specific change request) or a group of changes (AoC). The complexity model evolved to include the AoC allowing the researcher to infer on the complexity of each AoC group category (Figure 5.10).

Stage three of the analytical process considered the top two most significant AoC for further investigation (TUI Content and Function). Each AoC was analysed using interrelationships to the change construct dimensions of Form and Complexity. To enhance the understanding of Context the AoC distribution across the project cases was also considered in this stage. The interrelationships studied are as follows:

- Ripple Effects taking place on the various Architecture Layers.
- Maintenance change Forms occurring on the Architecture Layers.

- Effort involved for Ripple inducing changes.
- Effort involved with changes on specific Architecture Layers.

Further analysis could be conducted on all twelve of the AoC group categories identified but due to time constraints the researcher has only described two. However, the process has been outlined such that further work could be conducted on the remaining AoC categories in the future.

This chapter provides a granular insight into the nature and characteristics of WIS maintenance change and these findings provide the foundation for further work in this thesis. The following chapter will now address the second research question and uses the results from this chapter to create inferences upon the desired flexibility solution.

Chapter 6 ▪ Maintenance Change Facilitation: Designing a Conceptual Flexibility Framework

6.1 Introduction

The purpose of this chapter is to further build on the results and analysis of Chapter 5 in order to answer the two research questions posed by this thesis. The previous chapter presented and consolidated the findings related to the characteristics and AoC within WIS's. These findings were brought forward into this chapter and will be used as the input to determine the attributes of flexibility required to enhance and optimise the maintenance change process.

This chapter firstly intends to further crystallise the 'nature of change' observed by designing and utilising a graphical profile model of change (change profile model) (Section 6.2). This model ultimately became the input to the conceptual flexibility management and control framework (Flexibility Matrix of Change) that was used to deduce the relevant dimensions and attributes of flexibility that are required to facilitate WIS maintenance change (Section 6.3). Section 6.4 discusses the broader flexibility implications and inferences that can be drawn from the analysis conducted in this chapter. Lastly, Section 6.5 presents the conclusions of this chapter.

6.2 What is the nature of change in WIS maintenance?

This section answers the question posed by research question one that of ‘What is the nature of change in WIS maintenance?’. This was achieved by isolating and directing the discussion in relation to the top two most frequently occurring group categories of TUI Content and Function. In addition to this influencing factor, these two categories are diverse enough in their features to allow a broader perspective when answering the research question. Figures 6.1 and 6.2 summate and assimilate the findings from Chapter 5 relating to the AoC category and change construct dimensions. The change profile models express the key characteristics of WIS maintenance change which is ultimately required to address the second research question. A qualitative measurement of overall change complexity was constructed for each Form type within the change profile model. Furthermore, the overall change complexity measurement is interpreted and deduced on the basis of the statistical analysis for each of the dimensions relating to complexity.

6.2.1 Description of the Change Profile Models

The change profile models below address each of the two AoC categories individually. These models superimpose the Form of change and the mean effort based on Form and the AoC onto the existing overall change complexity model which was developed in the previous chapter (Figure 5.10). The relationships depicted in the change profile models provide a holistic view on the nature of change for each AoC. The model also differentiates and displays the nature of change for each Form type within the AoC (Figures 6.1 and 6.2).

For each group category the frequency of each Form of change is shown with the mean effort involved in implementing that particular group categories of change. The effort value specifically refers to the mean time taken to resolve changes for particular Form types of the AoC. Impacted Architecture Layer values have been derived from the frequency of occurrence of changes upon each Architecture Layer.

Similarly the Ripple Effect values have also been derived and displayed based on frequency of changes that cause or do not cause ripple effects.

In the example of TUI Content it is evident from the frequency graphic which Forms of change are the most prominent (Adaptive followed by Corrective followed by Perfective). The first consideration to formulating the overall change complexity measure is the dimension of Effort. In the case of TUI Content and the Form Adaptive, it can be shown that the mean effort was 0.69 hrs and thus it was placed at the very low end of the scale. Next when reviewing the impacted Architecture Layer dimension, the change profile model indicates that a large percentage of changes were located on the Presentation Layer (83.1%), however, a sizeable instance was also located on the Data Layer (16.9%). Inherently this increases the overall change complexity measure when compared to just considering the Effort dimension. Lastly, the Ripple Effect dimension for this Form shows that almost a third of change requests did generate Ripple Effects. This culminates in a slight increase to the overall change complexity measure for this type of Form. The interrelationships that have been discussed in this example, allows for an understanding into the nature of change for this particular change category and Form. When considering the change profile model for the group category of Function it displays discernibly different characteristics of change then that of TUI Content AoC.

By identifying AoC, analysing the key trends and patterns for the constructs of Form and Complexity, the change profile models offer a succinct mechanism to answer research question one. The models offer a graphical representation showing the nature of WIS maintenance change, based on the results and analysis conducted.

The change profile models were required as a prerequisite to answering research question two. Nevertheless, the models in their own right can be of benefit to the WIS development and maintenance organisations as they provide useful information when considering the distribution, complexity and effort involved in resolving these types of changes. Additionally, the models can also be used by other IT specialists such as project managers, developers, change managers, system architects and business owners as a reference point to facilitate control and planning processes within the maintenance phase.

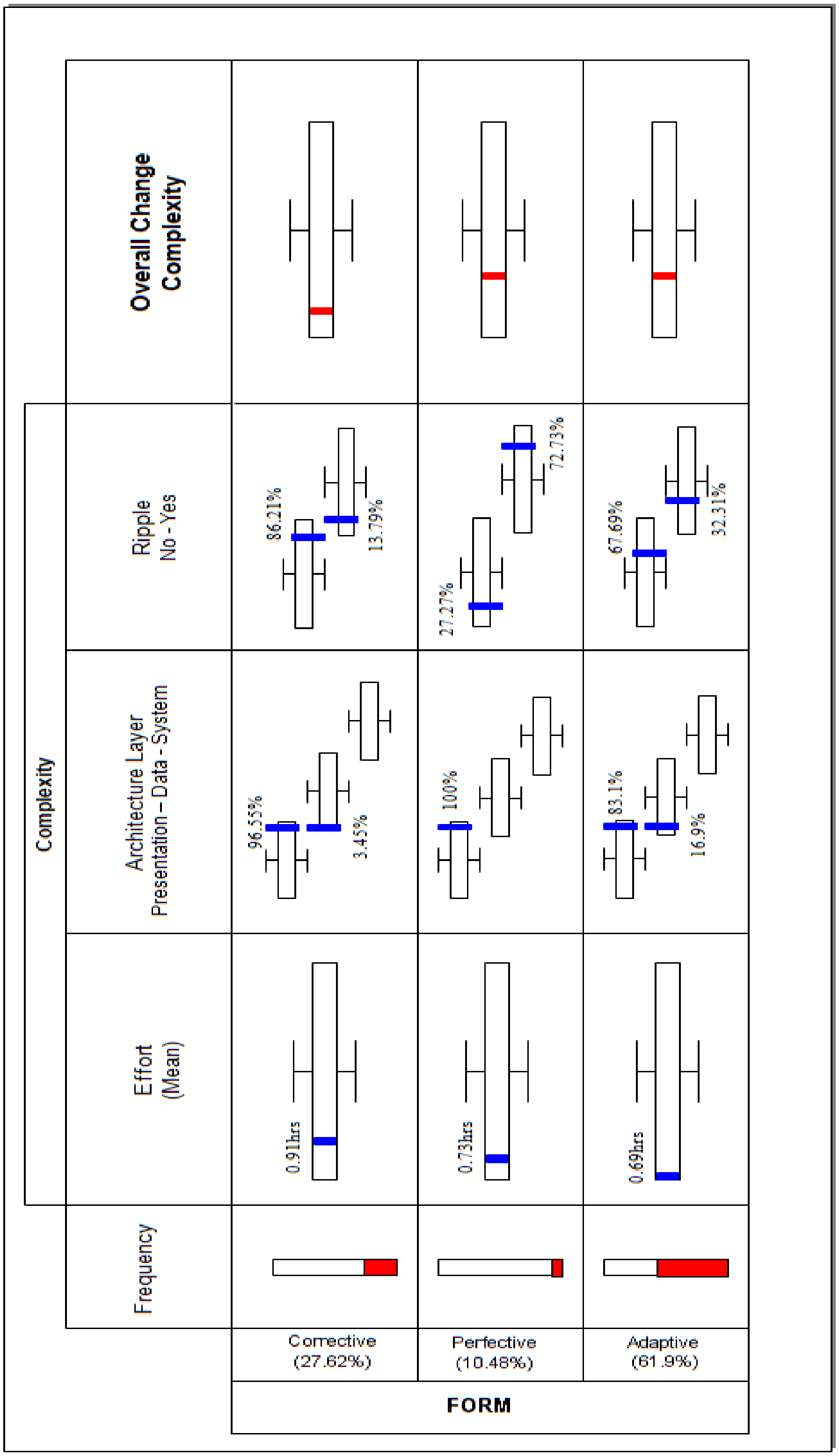

Figure 6.1 – AoC group category: TUI Content change profile

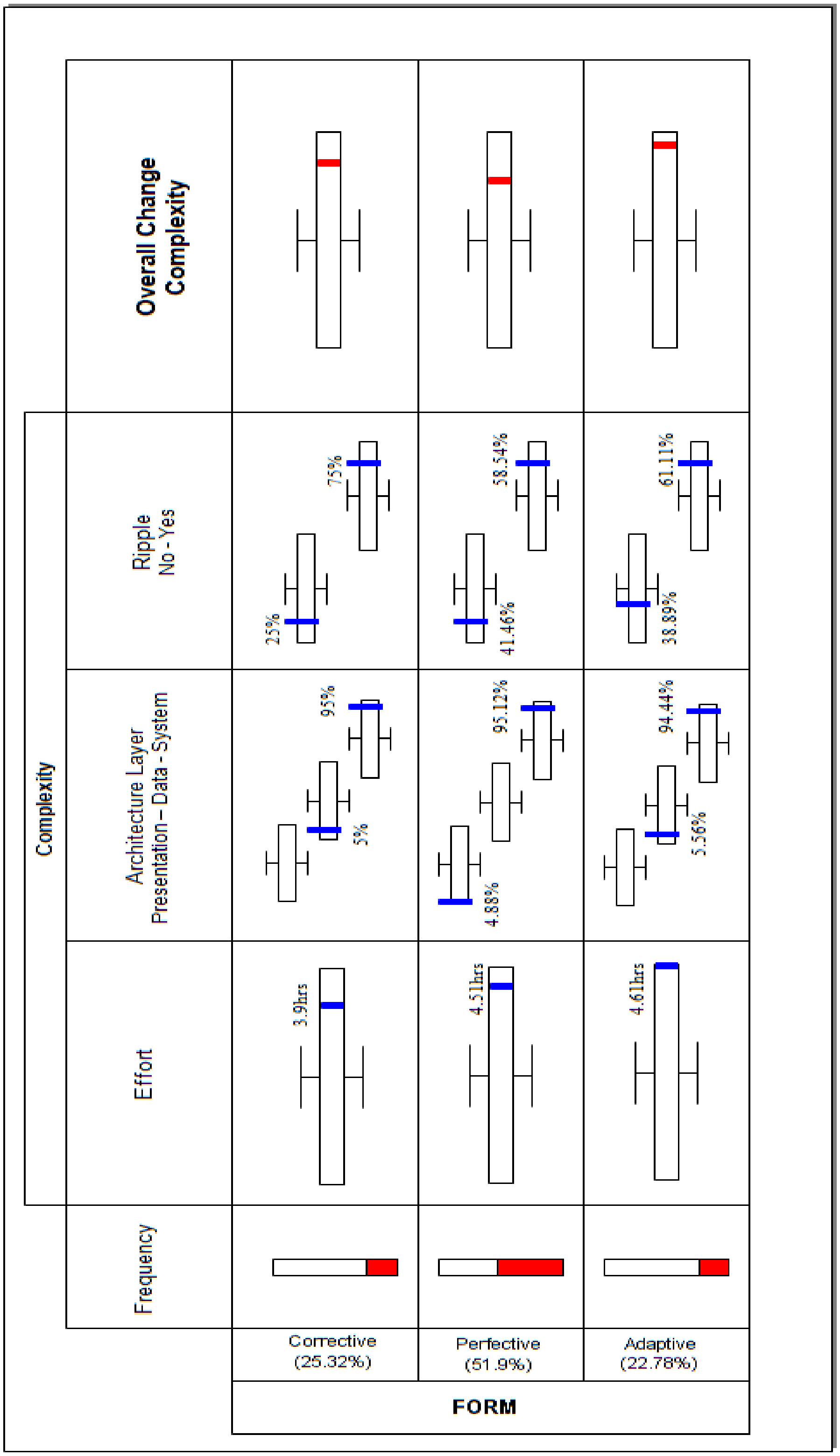

Figure 6.2 – AoC group category: Function change profile

6.3 How can maintenance change be facilitated in the WIS context?

This section conceptualises the process needed to address the second research question that of ‘How can WIS maintenance change be better facilitated?’. The literature review in Chapter 2 indicated several propositions that were conceived of in order to address the issues of handling continual change. The objective of these approaches, aims to afford sustainable and adaptable software systems that can operate in dynamic and often unpredictable environments. Specifically, Chapter 2 distinguishes between the panaceas of agile development methods, change management techniques and the inherent flexibility concept. These approaches are discussed in relation to the problematic issue of coping with the maintenance iceberg. While the latter two approaches (development methods and change management) are important aspects to be considered within any software system initiative, the former (flexibility) technique was the adopted solution within this thesis. This establishes the reasoning behind the sub-research question of ‘How can flexibility facilitate WIS maintenance changes?’. The sub-research question specifically identifies inherent flexibility as a solution and the following section selects and applies the flexibility concept to the identified AoC categories discussed above and in Chapter 5. The next section provides an abridged view of the IS flexibility concept and subsequently an appropriate framework was applied within this part of the thesis.

6.3.1 The IS Flexibility Concept and Frameworks

As was asserted in Chapter 2, an alternative approach to managing and dealing with continual change is the concept of IS flexibility. In particular Section 2.4.3 identified the dichotomous nature of the IS flexibility concept into process and structural (inherent) types. This research focuses on the inherent type of flexibility as a solution to facilitate change. The perspective taken considers IS flexibility as a potential solution to address the problems of continual change and the associated maintenance overhaul of the system lifecycle. The IS flexibility concept is a desired design characteristic to incorporate into software components in order to handle maintenance change in a more efficient manner. It is proposed that a tangible reduction in the effort, cost and time parameters would be the resultant affect of introducing such adaptability measures. The caveat to the flexibility phenomenon is that it is a complex, multi-faceted, diverse and diffuse concept that is open to interpretation. A 100% flexible system (zero maintenance) can also be an unwanted not to mention an unrealistic goal. Indeed as Hanseth et al. (1996) suggests there is a fine balance between the need to provide standardisation as well as introducing flexibility to anticipate and accommodate future changes. The trade-off when considering flexibility is not to detrimentally impact standardisation processes that already provide efficiency gains, whilst simultaneously introducing adaptive flexibility mechanisms.

For the flexibility concept to be a viable solution to IS's, it requires a targeted well defined application as well as seamless integration with the development methodology. However, the realisation of the flexibility method into a generic framework has yet to be defined and adopted in the IS realm. The disparity in frameworks often results in ambiguity of its practical application and use within IS. Current frameworks are differentiated on the basis of their operationalisation, application, measurement and context of use (Section 2.4.3). The determinant factor to the introduction of the flexibility concept is to ascertain where it is needed within the IS before it can be actively applied (Land, 1982; Fitzgerald, 1988, 1990; Boogaard, 1994). This thesis as its primary objective has isolated where flexibility needs to be applied by analysing the nature and characteristics of changes that occur within a WIS context. A pre-emptive flexibility mechanism was now required to reciprocally map to these existing change 'hot-spots' in order to realise the benefit of change facilitation.

In reviewing the IS literature (Appendix A), three applicable flexibility frameworks from various sources were found (Knoll and Javenpaa, 1994; Victor, 1995; Nelson and Nelson, 1997; Nelson and Ghods, 1998; Bryd and Turner, 2000). Collectively they establish the taxonomy of the flexibility concept into distinct dimensions. They demarcate and present the flexibility concept as dimensions to suggest the necessary predicates to be considered when using flexibility in IS's. As will be shown below the flexibility framework as conceived and proposed by Knoll and Jarvenpaa (1994) was used and applied in a systematic way to the identified AoC for WIS. Their framework provides a comprehensive depiction of the flexibility concept under the parameters of flexibility in use, functionality and modifications. These parameters encompass the structural flexibility type attributes from all the available frameworks reviewed (Appendix A), however they do not include the process flexibility dimensions. Nevertheless, the framework provides a detailed perception of the flexibility concept that is granulated at the necessary level and focus to align with the detailed analysis of change studied in the remits of this thesis.

A practical means of managing and embedding the flexibility concept into IS's is often disregarded. Thus a formalised approach is required if it is to be introduced into any IS in a controlled method. The 'matrix of change' framework is described below and was customised to manage, control and apply the flexibility dimensions to the specific WIS AoC. This investigation now progresses into addressing the second research question by using the change profile models established in Section 6.2.1 for the group categories of TUI Content and Function. The implications from the derived change profile models indicate where the benefits of flexibility could be realised. This allows for a targeted application of the flexibility dimensions within a WIS context.

6.3.2 The Matrix of Change

The matrix of change was used as a tool by Brynjolfsson et al. (1997) for the purposes of business process reengineering. It presented a way to capture connections between existing and target practices from an organisational change management perspective. The matrix of change provided a unique, useful guideline for the overall change management process for the purposes of managers. Its intentions are to support and anticipate the complex interrelationships surrounding business change. Brynjolfsson et al. (1997) designed the matrix of change to involve four steps: 1. Managers determine which business practices matter most for their business objectives. 2. The matrix highlights interactions among these practices and possible transition difficulties from one set of practices to another. 3. It encourages stakeholders to provide feedback on proposed changes. 4. It reveals process interactions that can provide guidelines for the pace, sequence, feasibility and location of change. The matrix of change as envisaged by Brynjolfsson et al. (1997) is displayed in Appendix G.

The ability of the matrix of change to correlate two sets of relatively diffuse parameters relating to, business processes as well as its ability depict the interaction between these parameters led the researcher to adopt a similar technique. This customised version of the matrix of change was able to provide a succinct and coherent visual model to analyse change and flexibility. More precisely this customised framework systematically maps WIS change (as identified by the change profile models) to target flexibility dimensions (Section 6.3.2.1). The visual model of a flexibility measurement scheme is used as the mechanism to show the relationship and activities between WIS change and flexibility in a concise and succinct manor.

6.3.2.1 Operationalisation of the Flexibility Matrix of Change (FMoC)

The flexibility measurement scheme will henceforth be referred to as the ‘flexibility matrix of change’ (FMoC). The overall purpose of the FMoC was to act as a management and control mechanism to identify and map flexibility dimensions to the identified WIS AoC group categories. The FMoC framework was designed to incorporate the parameter of the change profile models (Figure 6.1 and 6.2) for the two most significant AoC within WIS. The dimensions of flexibility which form the second parameter within the FMoC are derived from the flexibility framework prescribed by Knoll and Javenpaa (1994) (Figure 6.3). It was envisaged that the fulfilment of the FMoC framework will indicate the degree and extent of the required flexibility need within that particular change area.

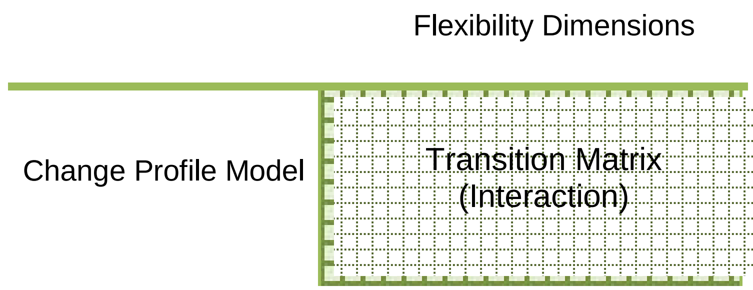

Figure 6.3 – FMoC high level view

The section below delineates the operationalisation of the FMoC framework into three distinct steps:

1. The input and utilisation of the change profile model.
2. The selection and input of the chosen flexibility dimension based framework.
3. The interaction and mapping procedure between the change profile to the targeted dimensions of the flexibility framework.

Step 1 – Change Profile Input

The first step involved in the input process of the FMoC framework involves the recognition of the existing areas of the system that require change. This research developed a process whereby key change categories emerged from the WIS's investigated (Chapter 4). The change categories and its dependencies were analysed and reported in order to ascertain its characteristic nature (Chapter 5). The characteristics were determined in accordance with the analysis constructs of this research, namely Context, Complexity and the Form of the change. These characteristics are displayed graphically as change profile models for the group categories of TUI Content and Function (Figures 6.1 and 6.2). The change profile models for the selected AoC should be listed on the horizontal axis of the FMoC. Figure 6.5 (page 209) displays the selected group categories of TUI Content and Function placed on the horizontal axis.

Step 2 – Inputting the Flexibility Framework and its Dimensions

The next step in producing the FMoC was the input of the selected flexibility framework and its associated dimensions. The preceding section identified the available IS flexibility frameworks and the perspectives taken to define, measure and apply it. For the purposes of this research the selected framework developed by Knoll and Javenpaa (1994) was used. The flexibility framework is de-segmented into three broad areas by the authors and was then further decomposed into twenty individual dimensions (Table 2.1 and Appendix A). The flexibility framework thus forms the vertical axis of the FMoC as target dimensions that the intended AoC category may require.

The FMoC offers the ability to select and use applicable flexibility frameworks within the context of use and the granularity at which the WIS is considered. The AoC are filtered and analysed through the lens of each one of the flexibility dimensions. Figure 6.5 shows the flexibility framework dimensions inputted into the vertical axis of the FMoC.

Step 3 – Transition Matrix: Relationships of Flexibility to Change

Lastly, the juxtaposition between the horizontal and vertical axis represents the transition matrix. The transition matrix shows the corresponding interaction and fit between the selected dimensions of flexibility with the specific attribute of the AoC change. The flexibility dimensions are mapped to the AoC being analysed on the basis of its suitability to facilitate this change or a certain characteristic of this change (Figure 6.5). The fulfilment of the transition matrix was based on the following four options:

1. Flexibility dimension required.
2. Flexibility dimension not required (Inherent flexibility may already exist OR is not needed).
3. Flexibility dimension not applicable in this instance.

4. Relevant change parameter data unavailable.

6.3.3 Flexibility Dimensions and Justifications

Each dimension of the flexibility framework was selected on the basis of context and an understanding of the characteristics of the AoC group category. Presented below are the justifications given for the group categories of TUI Content and Function. The explanations supplement the transition matrix values in Figure 6.5.

TUI content AoC group category

The change profile for the TUI Content AoC shows that the effort to implement these types of changes is Low suggesting that tacit flexibility was evident in facilitating these types of changes. Nevertheless the scope to increase flexibility further has been recognised and will be described below.

Robustness:
In the case of robustness this dimension was not needed irrespective of the Form of the change. It requires the opposite to robustness (i.e. some aspect of pliability) in order to accommodate the changes more efficiently.

Scalability:
Perfective Form could be a candidate for scalability however due to the low frequency and Low-Medium overall change complexity for these types of changes; time/ cost implications would most likely outweigh potential benefits gained by increasing scalability. Adaptive Forms do need a degree of scalability as they are occurring at a significantly higher frequency and have a Low-Medium overall change complexity.

Slack:
In context of the three month WIS maintenance observation period not enough of Perfective types of changes were occurring (low frequency), however if the study

was conducted over a greater time period it may warrant the need for Slack in this area. Although Corrective type changes are highly frequent, it does not warrant Slack because the overall change complexity is Low. Changes of the Adaptive Forms have a high occurrence and require a degree of Slack for variations in TUI Content modifications.

Commensurability:

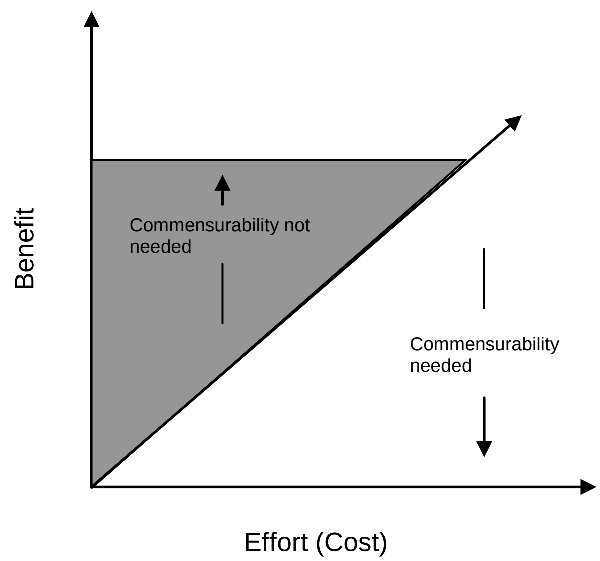

Figure 6.4 – Flexibility dimension of commensurability

Although the researcher collected substantial amount of data in relation to the effort (cost) involved for the change categories, no data was collected pertaining to the benefit achieved by inducing TUI Content changes from the end clients perspective. Due to this it was not possible to comment on the commensurability flexibility dimension. The above diagram (Figure 6.4) shows how commensurability is measured. As recognised in the literature the cost to benefit ratio of building in flexibility is often a judgment call of senior IS managers. This is because both cost and benefit are difficult to quantify in finite terms (Boogaard, 1994).

Feedback Sensitive - Goal Adjusting:

This type of flexibility was not required as TUI Content does not require intelligence in automating feedback and goal adjusting from external influences.

Just-In-Time Adjusting:

Based on the effort for the group category as a whole (and when considering by Form) the TUI Content changes can be made relatively quickly which indicates that this flexibility may already be inherent.

Polyadjustable:

Due to the overall complexity for each Form being relatively low across the whole group category and the inherent nature of TUI Content changes, it would not be beneficial to have multiple means of executing these types of changes.

Self-Adjusting:

Due to the overall complexity for each Form being in the Low range, the self-adjusting capability is unnecessary. However, in the instances where complexity is in the medium-high range (e.g. multi-lingual TUI Content) it may be warranted.

Trialability:

Low effort indicates that TUI Content changes can be implemented (i.e. tested and developed) quickly, hence no need for trialability. As a result of the inherent nature of web page structures, it is quick to test/ trial/ prototype changes to assess the immediate effect. When considering Perfective Forms of change the increased ripple and relatively high complexity of these changes suggest that a greater degree of trialability is required.

Concurrency:

TUI Content changes do not require simultaneous modifications.

Connectivity – Modularity – Multiple Forms – Reusability – Versatility:

As a result of the inherent web presentation components of HTML-tagged documents and FRAME PAGE hierarchical structures, the flexibility dimensions of Connectivity, Modularity, Multiple Forms, Reusability and Versatility have already been accommodated by flexible solutions. There is evidence to suggest that these types of flexibility are already inherent for TUI Content changes.

Responsiveness:

Due to the inherent cashing mechanisms of web technology i.e. local browser cache and proxy server cache this dimension of flexibility appears to be inherent for the TUI Content changes.

Spatial Decoupling – Transparency:

The inherent HTML-tagging mechanism enables the implementation of the TUI Content changes, however it requires a relatively high level of sophistication. In the case for both Perfective and Corrective Forms of change the effort involved was in the Medium-Low range. This suggests that there was further scope (through increasing flexibility in - Spatial Decoupling and Transparency) to lower the level of sophistication (increasing the ease), thus facilitating TUI Content changes further with these dimensions of flexibility. The definition of both dimensions propose technologies whose use is easy to infer based on its similarity to other structures and processes in the task domain. The similar structures and processes that are designed within the scope of the system should be more intuitive as well as usable. The resultant effect of such measures should inherently reduce the time and effort involved in introducing similar changes when they arise.

Temporal Decoupling:

Inherent flexibility in terms of this dimension was achieved by the use of the CMS as well as the inherent nature of the WIS design that allowed online modifications (HTML) to this AoC. These technologies allow the WIS to be flexible enough to accommodate real time changes for the TUI Content AoC without disruption to the operational environment. This was compelling evidence for the existence of tacit temporal decoupling flexibility for this AoC.

Functions AoC group category

Robustness:

Based on the finding that a sizeable number (22.78%) of Adaptive changes are occurring for this AoC, this suggests that the Adaptive changes of the Function AoC are unable to cope with pressures induced from the external environment. This leads to the conclusion that the degree of robustness should be increased. The functions by their very nature relates to the components of the WIS that are unlikely to change much during the lifespan of the system (for example computational processing). Therefore pliability was not a usual type of flexibility that would be required for this AoC and for the reasons stated for the Adaptive changes, Perfective changes should also include an increased level of robustness.

Scalability:

When considering the scalability dimension of flexibility for the group category of Function it was better suited to discuss the differentiation of functions on the basis of Architecture Layer as well as the frequency of occurrence. It was assumed more scalability is required on the Data Layer > System Layer > Presentation Layer. However, in this instance for the generic group category of Function the vast majority of elements are affecting the System Layer. In the context of WIS it was not a primary concern for functions that effect the System layer to necessarily be scalable therefore this type of flexibility would not facilitate any of the changes of this AoC.

Slack:

Inherent slack would decrease the effort and complexity involved in implementing changes of this AoC. In all cases of Form the effort and overall complexity was in the high range, thus requiring a degree of slack in all instances. When considering Perfective and Adaptive Forms of change, building in slack from the outset should be able to accommodate growth and diversification of this AoC.

Commensurability:

The same argument for the group category of TUI Content is presented for the Functions group category.

Feedback Sensitive – Goal Adjusting – Self Adjusting:

When considering Adaptive and Perfective Forms of change, the system should be intelligent enough to have Feedback Sensitive, Goal Adjusting and Self Adjusting mechanisms, in order to reduce the effort involved in response to new or environmentally induced changes. Irrespective of form the vast majority of changes are effecting the System Layer. All three types of flexibility need to be increased, other considerations regarding the flexibility solution was that the majority of these changes are on the System Layer and induce ripple.

Just-In-Time Adjusting:

The common trend found across the whole group category was that the Effort rating is High. By increasing Just-In-Time Adjusting flexibility for function type changes, this would result in a lower effort required to implement Function changes. These types of changes could then be implemented closer to the time of the systems operational use.

Polyadjustable:

The common trend found across the whole group category was that the overall complexity for Functions is in the High region. Thus the polyadjustable dimension of flexibility is required in all instances irrespective of Form, in order to provide multiple means to resolve the problem.

Trialability:

The group category of Functions displays high effort and high overall complexity for each Form of change. When considering the trialability dimension of flexibility the ripple effects need to be taken into account specifically. The results indicate that a significant amount of ripple was taking place for each Form. This warrants the need to be able to test/ trial/ prototype the change effects more rigorously across the system.

Concurrency:

Inherently within a web environment you must cater for multiple people making similar queries at the same time. This dimension of flexibility becomes more important with an increasing user base. The degree to which it is applied is thus proportional to the total user base size.

When specifically considering System and Data Layers of the application architecture, this dimension of flexibility should be intrinsically adhered to. The results borne from this research indicates that most Function type changes, irrespective of Form types, are positioned upon the System Layer. This suggests that a further degree of concurrency may be required. Since an accurate sizing of the total user base is unknown, more information will be required before making a judgement on whether this flexibility dimension needs to be enhanced.

Connectivity:

The only inference that can be made for the Function AoC and this dimension of flexibility is that tacit connectivity appears to exist based on the finding of the significant ripple effects caused by this change. However in order to judge whether this type of flexibility should be increased more precise change data would be required.

Modularity – Reusability – Spatial Decoupling – Transparency – Versatility:

The separation and distinction of functional processes into modular activities facilitates the change process in a system. In becoming more modular the system can provide the added value of having repeatable functions that can be reused in similar tasks (inheritance). The increased organisation of the system structure will also provide more spatial decoupling, transparency and versatility. An example of this is the concept of object oriented (OO) technology. The results indicate that the effort and overall complexity of the group category Function irrespective of Form is high. Incorporating an OO language ‘appropriately’ (.NET, C++) into the WIS design would facilitate Function AoC and would be a means to increase flexibility in all of the five dimensions. This flexibility solution also targets the System Layer, further assisting with the facilitation, however, it is noted that the OO solution and the increased flexibility that it can potentially provide, is contingent on its appropriate and skilled implementation.

Multiple Forms:
This type of flexibility would not reduce the effort or overall complexity for this group category, therefore it was not required. Furthermore, introduction of this type of flexibility would introduce unnecessary redundancy into the WIS.

Responsiveness:
Functions pertaining to data entry and retrieval are the largest candidate for this kind of flexibility. However, bottle-necks in the system can occur at any level of the Architecture Layer. The responsiveness dimension of flexibility needs to be incorporated at the System Layer as the change profile suggests that this layer is predominantly the one that relates to function changes.

Temporal Decoupling:
Within the maintenance activity periodic down times were present in order to incorporate the new system releases (observed in all three project cases). The system was not flexible enough to accommodate real time changes and thus the operational use was compromised. It is thus beneficial to increase temporal decoupling to ultimately result in the ability to make real time changes within an operational environment (i.e. system still in use while amendments are being made).

6.4 Flexibility Implications and Inferences

The FMoC framework allows the researcher to answer the second research question that of how can WIS maintenance be facilitated. The analysis of the flexibility dimensions required to optimise and enhance the maintenance change process for two specific areas of WIS maintenance change have been detailed and depicted in Figure 6.5. This section summates the broader implications and inferences that can be drawn regarding flexibility in a WIS context.

6.4.1 Inherent Technological Flexibility

The FMoC framework can be used to identify where and what types of flexibility are required, and in the case that flexibility is not required to assess whether some form of tacit flexibility may already exist. The tacit flexibility may exist as a result of the inherent nature of a technology or some design features that have already been built into system to resolve the change in that area more efficiently. Within web technology tacit flexibility appears to be inherent in its presentational components; one facet of this component being the TUI Content AoC. Analysis conducted in this chapter suggests that due to WIS attributes such as HTML-tagged documents and Page/ Frame hierarchal structure design features, the flexibility dimensions of Connectivity, Modularity, Multiple Forms, Reusability and Versatility are addressed to a large degree within this AoC. The observations that TUI Content type changes were resolved with relatively low effort and complexity further support this view.

In addition to these prominent inherent flexibility types that appear to exist in WIS, the analysis also suggests that other forms of flexibility are also present to some degree. The flexibility dimensions of Just-in-Time Adjusting, Transparency, Spatial Decoupling, Modularity, Reusability, Versatility and Multiple Forms appear to have been incorporated to an extent. The incorporation of a CMS may have contributed to increasing the effects of these flexibility dimensions. This technology allows the facilitation of TUI Content changes to be accomplished in an easier manor using a medium that requires less sophistication and proves to be a more efficient resource overall. As a consequence of incorporating a CMS into the WIS design the effort and complexity of implementing these changes will decrease further as this type of technology matures.

When considering the AoC of Function, the analysis suggests that there may be inherent modes of flexibility occurring; however the results from the FMoC overwhelmingly support the notion that there is considerable scope to increase flexibility across almost all dimensions studied. Figure 6.5 shows that apart from Scalability and Multiple Forms the other eighteen dimensions of flexibility are required. This view is supplemented by the finding that the effort and overall complexity to resolve functional type changes is High. It could be the case that greater granularity with respect to the Function AoC change profile (nature of change) is required to give a more focused, concise and accurate representation of the flexibility dimensions that can facilitate these types of change. Although the Functions AoC was developed using a bottom-up categorisation approach it encompasses a diverse variety of changes. Therefore an increase in granularity of the Function AoC could result in less generalisations relating to the flexibility inferences. Further work will be required to segregate this relatively course AoC.

6.4.2 Cost Effectiveness of Flexibility Solutions

The cost effectiveness issues relating to the implementation of flexibility solutions have two antagonistic forces at play. On the one hand there is the view that facilitating the change area via flexibility will innately reduce the effort and complexity involved to resolve the change. However, the opposing view asserts, if the cost of building in a flexibility solution into the system outweighs the cost saving benefit made to the organisation, then overall the solution is not cost effective and should not be implemented. The optimisation process of any type of IS thus needs to take into account the surrounding issues of durability and reusability of that system. Also a realistic appreciation of the systems lifespan and move away from 'short-sighted' 'cheap-hacks' with respect to maintenance solutions, is required. Flexibility solutions should always be carefully considered and its benefits viewed in-situ of the longevity of the system. This research does not have data to support the issue of commensurability, relationship of benefit to effort/ cost of implementing change. Due to the lack of this measure the cost-effectiveness of flexibility solutions cannot be reported on as part of this research study. Further work to highlight the cost effectiveness of implementing flexibility solutions is required and can be achieved by the operationalisation of the FMoC framework within the scope of an action research exercise.

6.4.3 Implications for the Maintenance Process

This research has highlighted the holistic trend that most changes are of the Adaptive Form in WIS's. Statistically the proportion of Adaptive changes that have been found in the WIS context differs significantly from that reported in the traditional IS context (Section 5.5.1). Within the IS maintenance literature Perfective Forms of change are the dominant type. The observation that Perfective type changes are diminished when considering the WIS maintenance phase does not necessarily indicate that new requirements are not being implemented. It would appear that WIS's allow the implementation of new requirements without the need of 'traditional development'. This indicates that WIS's are inherently more flexible (agile, malleable, versatile, adaptable, elastic, manoeuvrable and pliable (Siddiqui, 2002)) than their traditional IS predecessors. The implication for the maintenance processes, resources, management, technology and the initial development lifecycle within WIS is that it should be more orientated to facilitating Adaptive Forms of change. Flexibility mechanisms should thus be designed to cope with this shift of emphasis occurring within the WIS maintenance realm.

Various authors have commented that maintenance should not only be viewed as replacing 'as-is' components (Boogarrd, 1994). Section 5.5.1 elaborates on the finding that three-quarters of all IS maintenance change is in the form of enhancements (sum of Adaptive and Perfective), which indicates that the redesign of systems has always played a significant role in the maintenance phase. This study suggests that the ratio of changes attributed to enhancements in a WIS context occur at a similar rate. However, as mentioned a key difference was the ratio of the Adaptive type changes, which are shown to be more significant within a WIS context. Thus, a shift of perception is required when considering WIS maintenance to be viewed as a more adaptive phase of the system lifecycle. This re-conceptualisation of this phase is not necessarily a new concept, however the emphasis on the adaptive and evolutionary aspects are even more pertinent within a WIS context. This leads to the notion that maintenance in a WIS context is much more analogous to a traditional development phase that aims to constantly address adaption. Following this logic through directs the researcher to viewing WIS's as organic constantly evolving systems (Paul, 1994). This theme is further developed in the next chapter.

6.4.4 Implications for the FMoC

The FMoC framework offers a concise and coherent mechanism in order to focus, control and manage change with respect to flexibility. It can afford itself as a useful tool to key IS project personnel such as system architects, project mangers as well as lead developers. The framework provides a guiding facility to map interactions between the distinct WIS change characteristics to the relevant dimensions of flexibility in an intuitive manner. The FMoC framework indicates targeted and focused types of flexibility that are required to increase efficiency and optimise the change process. For example, within the case study context of this investigation, the FMoC (Figure 6.5) firstly indicates the dominant Form of change that needs to be considered is type Adaptive for the TUI Content AoC. Secondly, in order to further reduce the effort involved in resolving Adaptive Forms of changes in the maintenance phase it can be facilitated by incorporating the following flexibility dimensions: Scalability, Slack, Spatial Decoupling, Temporal Decoupling and Transparency. Furthermore the proposed flexibility solution should primarily take into account if there are instances of ripple effects taking place. However, the majority of changes for this AoC did not experience much ripple effects overall. The solution should address changes predominantly affecting the presentation layer of the system architecture. Lastly, considerations for the proposed solution should account for the overall complexity changes being in the Medium-Low range as well as the effort being in the Low range.

When considering the design and development of WIS's, the challenge is to reduce maintenance overheads and to build more sustainable software systems, 'designing for change'. By utilising the output from the FMoC as a data point of reference, development teams (i.e. system architects, project managers, developers etcetera) can design and build systems with flexibility from the outset. It has been proposed that the development lifecycle should be inclusive of Futures Analysis/ Flexibility Analysis stages to formally address the issue of designing for flexibility (Section 2.4.2). The process, models and frameworks developed within this research can supplement the suggested analysis stage within the lifecycle. Designing flexibility solutions early on within the development lifecycle can ultimately reduce the 'maintenance iceberg'.

6.5 Conclusion

The objective of this chapter was to build upon the findings and analysis of Chapter 5 to answer research questions one and two. In order to explore and best present the WIS maintenance change characteristics observed, a graphical profiling model was designed and utilised (change profile model). In addition to answering research question one, the change profile models also provided the input into the 'flexibility matrix of change' (FMoC) that can be used as a management and control tool. This allowed the researcher to map specific flexibility dimensions to the key AoC that emerged during the course of this research. The result of this process allows the researcher to answer the second research question of how flexibility can be used to optimise and enhance the maintenance change process within the WIS context. Lastly, wider implications regarding inherent technological flexibility, cost effectiveness of flexibility solutions, the maintenance process and inferences for the FMoC were explored and presented. These themes will be further explored in the following chapter where conclusions and implications are discussed.

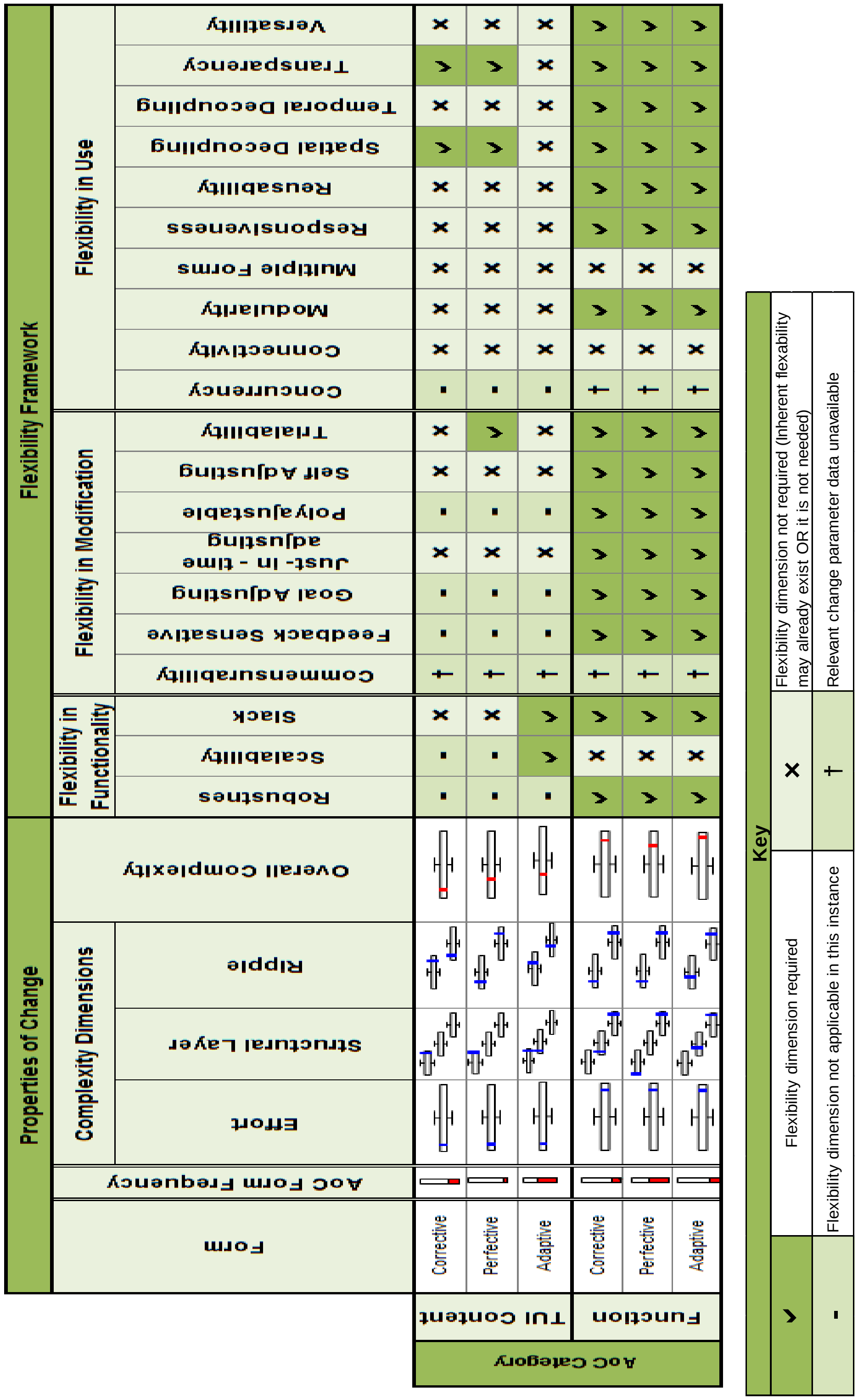

Figure 6.5 – The Flexibility Matrix of Change (FMoC)

Chapter 7 ▪ Conclusions and Implications

7.1 Introduction

The objective of this chapter is to provide a synopsis of the research that has culminated in this thesis. Specifically this thesis aims to investigate the nature of change in WIS for the purposes of isolating how flexibility could facilitate the maintenance phase of the SDLC. The research framework used, incorporated a case study approach and a emergent categorisation process that adopted the techniques established by Yin (2003) and Miles and Huberman (1994), respectively. The framework was essential to collate and analyse the empirical data relating to WIS change requests that was used to produce conclusions and findings.

This chapter is comprised of two main sections. The first section summarises each chapter of the thesis and its findings relating to the research questions established in Chapter 1 (Section 7.2). The second section discusses the implications, contributions and the further research that emanates from this study. The wider implication section attempts to extrapolate the empirical evidence and analysis into broader discussions on the topics of maintenance, the SDLC and the concept of flexibility (Section 7.3). The research contributions are organised into five themes that collectively summarise and contribute to the body of knowledge (Section 7.4).

The further research section suggests four potential streams of work that can be conducted to supplement and build upon the conclusions and findings of this thesis (Section 7.5).

7.2 Summation of the Research

Chapter 1 *'Overview'* provided the context and reasoning for the proposed research questions. The overarching rationale of the thesis was positioned to address the issue of adaptability in context of dynamic and often unpredictable business environments. It was suggested that both internal and external organisational change are problematic and difficult to integrate into existing IS's. The chapter outlines two key research questions (Section 1.3), that aim to investigate how change facilitation and the resultant effects to increase adaptability using the flexibility concept can placate the maintenance phase dilemma's. The research focus on a web orientated environment was explained and identified as an area that has been somewhat neglected in terms of what is known about WIS maintenance. The chapter also presents the objectives and an overview of the thesis.

Chapter 2 '*The Maintenance Quandary: A Literature Review'* explores and examines the issues presented in Chapter 1 in greater depth. In particular the traditional perspective of IS maintenance is reviewed in conjunction with the problems and issues that it presents to the sustainability of WIS's. The characterisation of maintenance change in terms of its methodologies, approaches and techniques provides a comprehension of how it has been viewed and investigated to date. The subject and new era of WIS was described as it forms the context of this thesis. The WIS literature indicated a lack of in-depth considerations of its maintenance activity and the need for more research by academics to better understand the phenomena was required. There is an active debate amongst practitioners and academics in the similarity and differences between IS's and WIS's, this chapter studies the perspectives in-situ of the maintenance phase activities. This provides the justification, motivation and rationale for conducting this timely research investigation. The concept of flexibility within the IS realm was introduced as a potential solution and was used in Chapter 6 for the creation of a qualitative flexibility measurement framework.

Chapter 3 *'Research Design: Building a Research Framework to Investigate Maintenance change in Web Information Systems'* focuses on designing a research framework that would be adequate to answer the research questions sufficiently. Research Framework A was developed from the axiom that maintenance change occurring in WIS's could be studied from an IS perspective. The pre-existing IS classification of the 'areas of change' according to Fitzgerald et al. (1999) was selected from the literature and served as an appropriate analysis metric. This conjecture was specifically used as the basis to Research Framework A in order to address the issue and active debate on the conformity of IS's and WIS's. Research Framework A adopted a case study (Yin, 2003) which was deemed a suitable strategy when considering the exploratory nature of the research questions posed; *'What is the nature of change in WIS maintenance?'* and *'How can WIS maintenance change be better facilitated?'*. The first stage of the research framework entailed the data collection of change requests using documentation, interview and observation techniques (Yin, 2003). The second stage of Research Framework A was a categorisation process whereby change requests were filtered on the basis of the categories as defined by Fitzgerald et al. (1999) to attain what areas of the WIS were being impacted. The purpose of the second stage was to answer the sub-research question of *'What are the areas of the software system most susceptible to change?'*. The categorisation process was validated by an independent researcher carrying out the same activities as that of the researcher. However, due to the observation that a more granular approach would be required, Research Framework A was suspended. A change in research direction was required and subsequently a new research framework was designed.

Research Framework A was designed using a top-down approach which was founded on the basis of the existing areas of change categories. Alternatively

Research Framework B used a emergent and bottom-up approach to organically allow areas of change and subsequent group categories to emerge. Research Framework A inferred that a comprehensive understanding of WIS maintenance change was required. The consequence of this was two-fold, firstly that the new sub-question of *'What are the characteristics of change?'* was added to research question one. Secondly, that the research emphasis was shifted from being predominantly concerned with the answering of research question two to now focusing on providing empirical evidence to substantiate research question one. A deeper understanding of WIS maintenance change was to be achieved by the addition of further analysis metrics (constructs of form, complexity and context). The resultant triangulated analysis framework was designed based on each construct and its associated dimensions. Additionally, Research Framework B was designed to enhance the case study strategy by incorporating embedded units of analysis (Yin, 2003) in order to create a larger data pool (each unit is equated to a WIS project - four units in total). Based on the collection of documentary, interview and observation evidence, the researcher would be able to gain an holistic perspective of each specific area of change. The researcher's perspective of the identified area of change was then pattern-matched and clustered (Miles and Huberman, 1994) to create change group categories. A saturation point with respect to new categories emerging was determined by using a control unit of analysis (a forth project case). A new categorisation process based on a flow chart mechanism was designed for the validation process. The chapter concludes by presenting the process and logical flow of the research frameworks.

Chapter 4 *'Operationalisation of the Research Framework'* discussed the operation of both Research Frameworks A and B. The chapter begins by describing the case study used which provided the data for both research frameworks. The chapter commences with an account of the data collection process as prescribed by Research Framework A, which entailed the researcher collecting change requests that emanated from various projects and collating them into a coherent dataset document. The categorisation process conducted (the second stage of the research framework) was exemplified and explained how change requests were classified and filtered into Fitzgerald et al.'s (1999) areas of change categories. The results produced out of this activity revealed that almost half of change requests fell into Fitzgerald et al.'s (1999) Data category. This divergence from Fitzgerald et al.'s (1999) results prompted the researcher to refine the category into sub-categories using Miles and Huberman's (1994) pattern-matching and clustering technique. Two significant sub-categories emerged namely Content and Graphical which were distinguishable WIS areas of change in their own right. The research framework could not proceed as it became apparent that greater granularity was required. The process however enabled a initial distinction to be made between IS and WIS maintenance change based upon the significantly different results of both studies. Three issues arose from the initial conduction of Research Framework A: firstly, that the research perspective had to change direction, secondly, that the research framework had to be revised and thirdly, that a greater degree of analysis of each change request was required.

The latter part of Chapter 4 describes the operationalisation of Research Framework B and how change requests were collected from the P1, P2 and P3 projects. The data collected from these projects were consolidated to form the dataset repository which would be used for analysis in Chapter 5. A emergent process using the bottom-up approach was used to identify impacted areas of change and the formulation of change group categories. The impacted WIS areas of change determined by the researcher were examined using the pattern-matching and clustering technique (Miles and Huberman, 1994) in order to identify and classify new change group categories. The WIS areas of change are further classified in accordance with the analysis constructs as defined in Chapter 3. The chapter concludes with stating that 425 individual WIS maintenance change requests were analysed and that twelve areas of change group categories emerged. At this stage of the research the sub-research question of *'What are the areas of the software system most susceptible to change?'* was partially answered.

Chapter 5 '*Results and Analysis of Web Information Systems Maintenance Change'* demonstrates the analytical examination process undertaken on the dataset repository. The chapter presents quantitative results that are extrapolated from the aggregated dataset repository and is used to substantiate the sub-question *'What are the characteristics of change?'* of research question one. The initial section of the chapter presents the analysis framework established in Chapter 3 and the interrelationships to be investigated between construct dimensions. Detailed analysis was conducted on the group categories of TUI Content and Functions, as they represented the top two most prominent areas of change of the WIS's studied. The analytical process was shown for the group category of TUI Content in its entirety and a synopsis of results provided for the Functions group category. The aim of the detailed analysis was to explain trends and anomalies presented by the dataset repository. The contexts of individual projects were used as explanations in some instances to the occurrence of particular trends or anomalies.

The results produced from the analytical work were discussed and evaluated in the latter half of the chapter, in order to provide a detailed answer to research question one, that of *'What is the nature of change in WIS maintenance?'*. The conclusions that were drawn were based on the dimensions of the constructs of Form, Complexity and their associated interrelationships. In addition a WIS qualitative model was developed in order to consolidate the interrelationships derived from the analysis constructs for each area of change. In conclusion, the researcher was afforded with an enriched perception and understanding of the characteristic nature of the WIS maintenance phenomenon.

Chapter 6 '*Maintenance Change Facilitation: Designing a Conceptual Flexibility Framework'*, this chapter proceeds by delineating the thesis into two sections based on answering the research questions one and two, respectively (Section 1.3). The first question of *'What is the nature of change in WIS maintenance?'* is fulfilled by directing the discussion in relation to the two group categories of TUI Content and Functions that were used in the previous chapter for the purposes of detailed analysis. The aim of this section was to abstract out of the detailed findings and provide a succinct means to present the data. Thus the data provided by the change constructs and their associated dimensions for the group categories (TUI Content and Functions) were summated and assimilated into profile models. The profile models offer a concise graphical representation of the analysis showing the nature of change in WIS maintenance. The models were required as a prerequisite to answering research question two, that of *'How can WIS maintenance change be better facilitated?'*. The sub-research question of *'How can flexibility facilitate WIS maintenance changes?'* was firstly addressed by the selection of a literature based flexibility framework developed by Knoll and Javenpaa (1994). Secondly, a qualitative flexibility measurement framework was formulated through the design of the FMoC (Flexibility Matrix of Change) tool. The design and development of this framework was based on the matrix of change (Brynjolfsson et al., 1997). Both the profile models and the selected flexibility framework provided the inputs into the FMoC, where a mapping was then undertaken to deduce the flexibility dimensions required. The chapter concludes by providing implications and inferences regarding flexibility that are based on the results produced from the FMoC for the two group categories of TUI Content and Functions.

7.3 The Wider Implications

The aim of this section is to consolidate the findings of this research and assess the impact and implications for the wider IS community. The wider implications are presented as three areas of discussion, namely implications for maintenance, implications on the software development lifecycle and lastly flexibility as a panacea.

Discussion 1 – Implications for WIS Maintenance

Some authors support the view that maintenance as defined as being the servicing of pre-existing components (i.e. maintaining the software system as-is), is in fact a misnomer (Martin and McClure, 1983; Boogaard, 1994). Another predisposition about IS maintenance is that it is viewed as a corrective phase. Boogaard (1994) asserts that IS maintenance should be thought of as a process where the design of the IS is actually changed. This research indicated that WIS is characterised by a greater degree of Adaptive forms of change that don't fundamentally change the system dimensions but rather replace existing components (for example TUI Content). With WIS's it was noted the replacement of pre-existing system components is a key attribute and thus the maintenance perception in the traditional sense of the word is held true. This was a interesting finding as this thesis supports the view that maintenance should be considered as more of a as-is activity which is contrary to the opinion of authors such as Boogaard (1994). It is suggested that this difference may be attributed to the context of WIS. The ubiquitous user base and interactive functionalities that WIS's can offer compared to IS's, suggest that the scope of influencing environmental factors that apply to WIS's are greatly increased. This research showed that enhancements (Adaptive and Perfective forms) to WIS's are occurring at a considerable rate and are likely to increase with the lifespan of the WIS. In summary this research indicates that WIS's have to incorporate two facets of maintenance that of:

- Maintaining existing components to a greater degree.
- Maintaining to expand system design capabilities to a lesser degree.

Discussion 2 – Implications on the Software Development Lifecycle

As mentioned in Section 6.4 the evolutionary software activities within the maintenance phase of WIS must be considered (Yeh, 1990). Traditional systems development suffers from several inherent problems. One such problem is its inability to cope with changing needs and requirements of the system pre and post development. Traditional system development methods suggest that the maintenance phase is the concluding task of the software lifecycle. Yet 70% of system development efforts are engaged in conducting maintenance work for the system, causing the widely acknowledged maintenance iceberg phenomenon (Martin and McClure, 1983). The attitude towards traditional systems development has been tailored to creating IS's and maintaining them up until the point that the maintenance of such systems become too unwieldy and not cost effective enough to support, at which point organisations implement a new IS solution. In the era of WIS's it is becoming evident that unless the organisation or business case behind the WIS is no longer viable these types of systems do not necessarily reach a termination point. Inherent structures and technologies that underpins WIS's allow for the constant evolution or assimilation of WIS's into larger more complex WIS's. These systems have the ability to be evolutionary and therefore maintenance should not be considered as the concluding stage to development but rather as an on-going expansion phase to develop and adapt the system. A new perspective is required that aims to incorporate the processes of WIS maintenance and its constant evolution.

This perspective supports the view that maintenance should be regarded as more of an extension phase. A living or evolutionary perspective is proposed to explain the changing and growing forms of software systems. The diagram below (Figure 7.1) attempts to exemplify the different lifecycle models that are proposed for WIS compared to the existing IS lifecycle. The model incorporates a different view on maintenance when compared to traditional IS. The IS lifespan goes through periods of being commissioned (conceived), developed, maintained and decommissioned (terminated). This continual commissioning and decommissioning of IS's is a costly and unavoidable situation that many organisations find themselves in. This is partly due to the fact that the financial aspect of the maintenance burden becomes greater than that required to create a new IS. Contrary to this process the proposed WIS extension lifecycle has an initial construction phase of commissioning and development (conceived) after which maintenance is considered as extensions to the existing infrastructure. It is during this extension phase where the WIS is incrementally metamorphosed until the point of termination, the extension phase can be regarded as a continuation of the build phase. The extension phase can be segregated into two discrete activities, that of evolution and substitution. It is during the activity of evolution whereby the WIS scope is enhanced as it acclimatises to new demands and additions to the system. The substitution activity refers to replacement of system objects/ components that require updating or modification in order to remain viable from a business perspective. In either instance Flexibility

Analysis (Fitzgerald, 1990) should be a consistent and constant activity within the lifecycle to allow the continuous analysis of change and the need for flexibility.

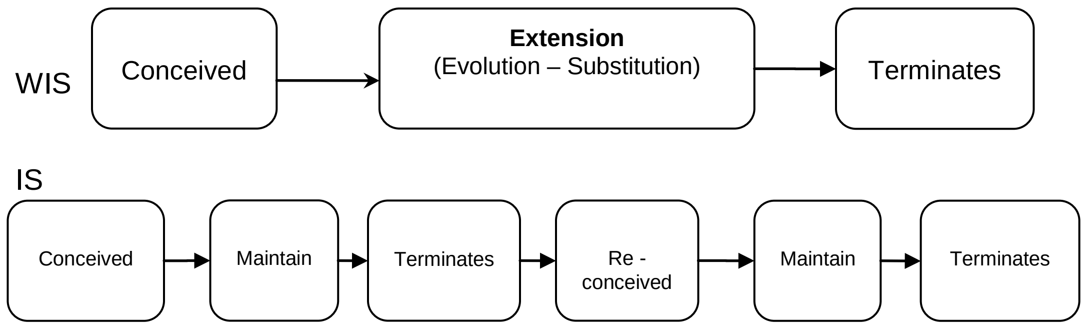

Figure 7.1 – Proposed WIS lifecycle in comparison to the IS lifecycle

Discussion 3 – Flexibility as the Panacea

The findings and implications on flexibility from this thesis have been discussed at length in Section 6.4. This thesis clearly demonstrates that there is scope to increase built-in flexibility within WIS. However, it is also recognised that WIS already utilises and incorporates a significant number of mechanisms, components and design features that have already inherently increased the inherent flexibility within these systems. Perhaps as a result of the organic evolving nature of WIS's, innovation in this area is growing apace. The proliferation of web 2.0 and service orientated architectures is based upon principles that aspire to increase flexibility. Indeed these latest innovations build upon the progress of 'first generation web technologies' that have utilised object orientation programming languages and distributed computing concepts to make great strides in making WIS more adaptable than their IS predecessors. Peering into the near future it can be seen that the evolution of the high bandwidth GRID and the distributed processing advantages that this will result in, will be another great leap forward in the progression of adaptable WIS's. Services that are offered through WIS's provide organisations with the ability to be more flexible and agile in their business models, an example of this has been the recent success of organisations such as salesforce.com. This WIS allows the financial accounting needs of the client organisation to be serviced by the salesforce.com third party, with the only technology requirement on the end client being to have access to a web browser. The ability to customise the application to incorporate the workflows of the end client is the only other prerequisite required. This is a far cry from the days of traditional IS's which would require a team of developers, project managers, business analyst, testers, implementers and a team of maintenance professionals to offer organisations the same level of service from their IS. Thus this service based, cloud computing model (The Economist, 2010) can provide organisations with business/ organisational flexibility, by reducing the constraints of time, cost and procuring a physical location. The rise of internet computing further diminishes the constraints listed above on organisations which are being facilitated by the vast data centres and server farms as well as the fast fibre optic networks that can service them (The Economist, 2008).

The benefits of flexibility in many respects is a double edged sword and cannot be regarded as the panacea to all of IS's ills. The flexibility solutions mentioned in Section 6.4 and 6.5 as well as above, would no doubt improve efficiency and facilitate the adaptability, however these gains do come at a price. Whenever flexibility solutions are proposed, one of the key considerations for most organisations will be the cost effectiveness of the solution. The analytical evidence provided in Chapter 5 clearly suggests that there is scope to introduce flexibility to facilitate the maintenance changes pertaining to the TUI Content area of change. This could have been achieved by enhancing the content management flexibility mechanism that was developed by the software vendor. However, if such a solution resulted in the project costing significantly more or meant that the project delivery would have been delayed, then it is more than likely that flexibility would have to be sacrificed for time and cost purposes. In the same example the software vendor could take it upon themselves to develop such flexibility mechanisms at a personal cost to their enterprise. The pay back from such an investment could only be realised after the gains achieved from multiple project deliveries. Thus the economy of scale argument is a key consideration when deciding upon flexibility solutions. The time horizon aspect of whether flexibility solutions should be utilised is also a principal concern, particularly when considering flexibility in-situ of facilitating maintenance change. Organisations must assess the costs related to the total maintenance over the lifespan of the system against the cost overhead to implement a flexibility solution within the initial development phase. The balance between these two opposing factors will ultimately determine whether flexibility benefits or hinders the organisation. Flexibility in some regards is also the antithesis of the standardisation process. The standardisation of communication protocols as well as other elements was essential and pivotal in making the Internet and WIS what they are today. Standardisation can result in the constraining of technology and process innovation for flexibility solutions. Again, a trade-off between the efficiency and stability gains provided by standardisation and the adaptability gains provided by flexibility are to be balanced.

7.4 The Research Contributions

The implications on WIS maintenance that have emerged from this thesis are examined and presented in this section. The contributions stem from the analysis of the literature review on the topic of maintenance change in Chapter 2, the bespoke approach and data collection explained in Chapters 3 and 4, the analysis of the empirical evidence surrounding the nature of change presented in Chapter 5, and the flexibility findings that have been elaborated on in Chapter 6. The larger themes presented below are elaborated further with more specific contributions.

Theme 1 - A Research Method to Categorise Change

A classification mechanism was developed within this thesis in order to address the first objective of research question one. Based on the pattern-matching and clustering principles of Miles and Huberman (1994), a emergent approach to identify areas of change and group categories were derived. A bespoke categorisation workflow process was also created for the purposes of validation. This research method allows for the emergence of categories from a large and varied data source and can be applied to various contexts as well as providing a means to build upon existing categorisations other than that described in this thesis.

Theme 2 – Detailed Analysis that depicts the nature of WIS Maintenance Change based on empirical evidence

As noted by Taylor et al. (2002) ‘there appears to be few actual case studies in academic or professional literature regarding the overall process of developing a company website and even a fewer regarding the maintenance of company websites’. This thesis provides extensive empirical evidence pertaining to changes that occur in the maintenance phase of WIS’s. In particular this research has contributed to existing WIS maintenance literature by conducting a case study that included multiple units of analysis. Generally, the degree of research into the maintenance phase is greatly diminished when compared to the body of work that has accumulated in the realm of IS predating the web era. Whether this is a result of the industry failing to distinguish between differences in WIS and IS is a debatable point, however this research has found there is a disparity in the characteristics of both. Based on the rationale described in Section 3.4, changes in WIS maintenance were recorded and analysed on the basis of the area of change and the analysis constructs. Furthermore the interrelationships between the dimensions of the analysis constructs were examined and investigated to reveal an in-depth understanding of the nature of WIS maintenance change. The empirical evidence within this thesis has also enabled a comparative analysis between maintenance in the IS and WIS domains to be made. In addition, the results of this analysis allowed the researcher to quantify the characteristics of WIS maintenance change and thus contribute to knowledge in this relatively neglected area within the WIS realm.

Theme 3 - Profiling Model of Change group categories

Based on the analytical evidence, individual profile models for the most significant areas of change group categories were developed. These were created to graphically represent and consolidate the sizeable and complex statistical data relating to each group category. The model facilitates the comprehension of the characteristics of WIS maintenance change. As stated in Section 6.3.1 this model in itself allows project managers, developers, change managers, system architects and business owners to quantify the resources required when planning for the maintenance phase. The model can also be extrapolated to identify where flexibility is required in order to facilitate change through the software system lifecycle. This profile modelling technique is context and granularity independent when considering the diffuse subject of change and can be reused in various situations other than that of this thesis.

Theme 4 - A Qualitative Flexibility Measurement Framework

The flexibility concept is a major facilitating factor when considering change within WIS. One of the major contributions of this thesis is the attempt to study flexibility in relation to this domain. The flexibility framework devised by Knoll and Jarvenpaa (1994) was assimilated into the flexibility measurement framework designed in this thesis (FMoC). The FMoC tool is a practical means of utilising the theoretical dimensions of Knoll and Javenpaa (1994). The tool has been designed to incorporate the profile models of group categories derived from the empirical data as well as the selected flexibility framework to demonstrate exactly to what degree and which types of flexibility are required to facilitate maintenance change. The two core inputs and the resultant flexibility outputs collectively make the FMoC an extremely useful utility for organisations to manage, control and plan for change within their IS's.

Theme 5 – The Distinction between IS and WIS Maintenance Change

This research established twelve areas of change group categories for WIS that showed a divergence from the established five IS categories (Fitzgerald et al., 1999) when considering maintenance change. Clearly a difference existed when considering the TUI Content group category in WIS and the reciprocal Data area of change category within IS. This difference may be attributed to the more pronounced presentational ‘published’ aspect of WIS’s. The Adaptive form of change is also significantly different across the two domains, being more prevalent in WIS (Section 5.5.1.1). This suggests that WIS’s need to be more reactive to environmental conditions where the ubiquitous user base and interactive functionalities inherent to WIS are more exaggerated. It is noteworthy to mention the difference between both domains in the amount of changes related to the Perfective form.

7.5 Direction for Further Research

The following section of the thesis provides some indications for further research. In reviewing the entire spectrum of the research process it becomes evident that Ph.D. investigations cannot engage all subject matters at the same meticulous level, thus assumptions and arbitrary choices need to be made to avoid a state of ‘research chaos’. It is imperative to narrow and focus the researcher’s scope on particular issues in order to successfully complete the study; however it is noteworthy to mention that the areas for further research clearly become apparent. During the research process, further work was required to gain an in-depth understanding of research question one as a consequence of conducting Research Framework A. This exemplifies the need for further research to be conducted during the process itself that can shift the orientation and emphasis of the desired investigation.

This section is broken down into specific further research suggestions that encompass relevant points that relate to the subject.

Further Research 1 – Expansion of the Dataset

In order to further validate the findings of this thesis the research techniques described in Section 3.4 should be applied to a varied range of WIS's. The thesis analysed four projects that were confined to one software vendor and predominantly orientated towards the financial services sector. The need to conduct this further research is in order to firstly validate that the areas of change are consistent even though the context of the WIS's studied may be different to this research. This can be achieved through a case study or a survey based research approach. Secondly, the observation period in the maintenance phase should be increased to ensure that the trends recognised in this study are consistent. As explained in Section 6.4 more granularity may be required to further investigate the Functions group category in order to better identify flexibility attributes required to facilitate these types of changes. This can be achieved by extending the pattern matching and clustering technique specifically to the Functions group category.

Further Research 2 – More Comprehensive Change Request details

Although this thesis accounted for the complexity of each change request collected, the priority criteria was not defined by the client, developer or researcher. Having this as an additional metric may provide some additional insights into the change management process and change generally. The change request analysis can be further supplemented by attaining additional information directly from the end client in the form of questionnaires and interviews, as this study focused on the software vendor's view. Further work in this area should also consider the differentiation of change requests raised in the development and maintenance phases of the lifecycle.

Further Research 3 – Developing Profile Models for Change group categories

This thesis identified twelve areas of change group categories within WIS. For the purposes of providing an insight into the analytical processes (Chapter 5) and in the interest of keeping the thesis to a manageable and succinct level, only the top two most frequently occurring areas of change group categories were selected for detailed analysis and profile modelling. The profile models in their own right, are of significant value to key IT/ IS personnel and as such the profile model should be extended to each of the areas of change group categories. Although the overall complexity measure within the profile models were based on empirical data, this measure is very much a qualitative representation. It has been recognised that a more accurate quantitative measure of overall complexity can be formulated using an algorithmic foundation.

Further Research 4 – The Operationalisation of the FMoC

The FMoC tool developed in Chapter 6 has created a direct relationship between the area of change group categories prevalent in WIS and a comprehensive view of the flexibility criteria that would facilitate these changes. The challenge still remains to create flexibility solutions in the practical IT domain that are able to realise effective flexibility solutions. A vast array of processes and tools are available to IT developers that can be applied to attain the desired effect. It can be supposed that the FMoC can be further customised to create a relationship between the types of flexibility required and a technical solution. The resultant output of the FMoC (Figure 6.5) should be applied and validated in an action research study.

7.5.1 Limitations and Reflections of the Research

All research has some form of limitations and in reflection this section identifies some restrictive factors of this particular research. These factors are either inherent in the research approach whilst others relate to contextual issues that the research had to address during the course of the investigation. These limitations are dichotomised into the following categories, firstly, research context – the limitations of the research area and secondly, research process – the limitations of the research framework, method and approach.

Research Context

In terms of the context of the research, the following constraints can be identified:

- The study was conducted at one organisation, so the findings cannot be readily generalised. However, this research required an in-depth study in order to firstly clearly identify the phenomenon and secondly to address the call for case study research to be conducted.
- The constructs of analysis used within this research provides a framework by which WIS maintenance changes can be analysed. However, it is realised that other parameters from the literature can be adopted in order to provide alternative perspectives. For example the maintenance types used in this research was adopted from Swanson and Lientz (1988), however more granular classifications could have been adopted from the literature.
- Due to the vast volumes of data collected, a spreadsheet mechanism was eventually utilised as a means to sort and organise as well as collate all the necessary information. A more sophisticated mechanism such as a database could thus be adopted in order to provide varied and intricate data relationships.
- The use of frequency analysis was used to indicate the results borne from the research. This was done in order to indicate and present the data in a clear fashion in an otherwise limited area of research. Software packages such as SPSS and MatLAB can be used to show more mathematical relationships between data entities.
- Further comparative analysis can also be conducted. A comparison between traditional IS and WIS maintenance changes would present interesting findings in relation to the evolving nature of IS's. Also the changes that occur pre and post development and how changes are propagated throughout the lifecycle would provide additional information on the nature of WIS maintenance changes.

- Additional change request data was required in order to fulfil the FMoC. An organisational perspective from key stakeholders will provide additional cost benefit information.

Research Process

A number of issues were noted (Chapter 4), that included the nature of the research method itself, as it was deduced that a more emergent bottom-up research method was required as a consequence of using a pre-existing categorisation model. The research evolved and the perspective taken on the investigation required a paradigm shift in order to enhance the research contribution. This is usually considered as the progressive nature of research and a key ingredient of any research process in order to narrow the focus of the thesis. The following research process limitations are identified:

- The data was mainly collected via documentations and interviews from the IT supplier. A wider perspective encapsulating other key stakeholders such as client views would aid in analysing cost benefit issues.
- The research approach can be repeated with the same constructs, tools and methods in order to extend, refute or confirm emerged categories from this research and for the validity of the research approach itself. This will also aid in building and creating further patterns and sequences of data relating to the nature of WIS maintenance changes.

7.6 Obiter Dictum

This thesis recognises that a paradigm shift is currently underway regarding the technologies available within WIS. A proportional change of mentality by IT professionals will also be required to embrace and to flourish in the new era of evolutionary systems. Technology and the industry are acknowledging the need for flexibility and to some extent they have already achieved inherent flexibility in the current and forthcoming WIS's. The challenge is to reconceptualise the management process and techniques as well as to educate the IS/ IT specialists, enabling them to align their thinking to the dynamics of the evolving nature of WIS's. As the industry matures further into this century we can only infer that the proliferation and complexity of WIS's will increase, the objective being to avoid some of the chasms and traps of the IS field. This is a cultural move away from the ubiquitous traditional manufacturing/ engineering mindset, however unless it occurs rapidly the WIS maintenance iceberg is likely to grow to mammoth proportions.

List of References

Ackoff, R. L. (1967), 'Management Misinformation Systems', *Management Science*, Vol. 14, No. 4, pp. B147-B156.

Alatalo, T. (2001), 'Lessons learned from putting a Web Engineering approach into practice', In: *Proceedings of the 24th Information Systems Research Seminar in Scandinavia (IRIS)*, Ulvik in Hardanger, Norway, August 11-14. IRIS.

Allen, B. R. and Boynton, A. C. (1991), 'Information Architecture: In Search of Efficient Flexibility', *MIS Quarterly*, December, pp. 435-445.

Ambler, S. W. (1995), *The Object Primer: The Application Developer's Guide to Object-Orientation*, New York, USA, SIGS Books.

Ansoff, H. I. (1968), *Corporate Strategy*, Middlesex, UK, Penguin.

Ansoff, H. I. (1984), *Implementing Strategic Management*, Englewood Cliffs, New Jersey, Prentice-Hall, Inc.

Aoyama, M. (1998), 'Web-Based Agile Software Development', *IEEE Software*, Vol. 15, No. 6, pp. 56-65.

Argyris, C. (1997), 'Organizational Learning and Management Information Systems', *Accounting Organization and Society*, Vol. 2, No.2, pp. 113-123.

Arnold, R. S. and Bohner, S. A. (1993), 'Impact Analysis - Towards a Framework for Comparison', In: *Proceedings of the Conference on Software Maintenance (CSM)*, Montreal, Quebec, September 27-30. Washington, DC, IEEE Computer Society, pp. 292-301.

Ashby, W. R. (1956), *An Introduction to Cybernetics*, London, UK, Chapman and Hall.

Austin, R. (2000), ‘The Paradox of E-Business’, *Cutter Consortium, Business Technology Trends and Impacts Advisory Service*, No. 1, November, pp. 1-4.

Avison, D. (1997), ‘The Search for the Discipline of Information Systems’, In: McKenzie, G. and Usher R. (eds.), *Understanding Social Research, Perspectives on Methodology and Practice*, UK, Falmer Press. pp. 87-100.

Avison, D. and Fitzgerald, G. (1995), *Information Systems Development: Methodologies, Techniques and Tools*, Second Edition, Maidenhead, McGraw-Hill.

Avison, D. and Fitzgerald, G. (2003), *Information Systems Development: Methodologies, Techniques and Tools*, Third Edition, Maidenhead, McGraw-Hill.

Avison, D. E. and Wood-Harper, A. T. (1990), *Multiview: An Exploration in Information Systems Development*, Oxford, UK, Blackwell Scientific Publications.

Avison, D. E., Fitzgerald, G. and Wood-Harper, A. T. (1988), ‘Information Systems Development: A Tool Kit is Not Enough’, *The Computer Journal*, Vol. 31, No. 4, pp. 379-380.

Avison, D. E., Powell, P. L., Keen, J. H. and Ward, S. (1995), ‘Addressing the need for Flexibility in Information Systems’, *Journal of Management Systems*, Vol. 7, No. 2, pp. 43-60.

Bahli, B. and Tullio, D. D. (2003), ‘Web Engineering: An Assessment of Empirical Research’, *Communications of the Association for Information Systems (CAIS)*, Vol. 12, pp. 203-222.

Banker, R. D., Datar, S. M. and Kemerer, C. F. (1991), ‘A Model to evaluate variables impacting the Productivity of Software Maintenance Projects’, *Management Science*, Vol. 37, No. 1, pp. 1-18.

Banker, R. D., Datar, S. M., Kemerer, C. F. and Zweig, D. (1993), ‘Software Complexity and Maintenance Costs’, *Communications of the ACM*, Vol. 36, No. 11, pp. 81-94.

Banker, R. D., Davis, G. B. and Slaughter, S. A. (1998), ‘Software Development Practices, Software Complexity, and Software Maintenance Performance: A Field Study’, *Management Science*, Vol. 44, No. 4, pp. 433-450.

Barry, C. (2004), ‘Web-Based Information Systems - Innovation or Re-Spun Emperor's Clothing’, In: Fitzgerald, B. & Wynn, E. (eds.), *IT Innovation for Adaptability and Competitiveness. IFIP TC8/WG 8.6 Seventh Working Conference on IT Innovation for Adaptability and Competitiveness*, Leixlip, Ireland, May 30-June 2. Boston, Kluwer Academic Publishers, pp. 441-458.

Barry, C. and Lang, M. (2003), ‘A Comparison of “Traditional” and Multimedia Information Systems Development Practices’, *Information & Software Technology*, Vol. 45, No. 4, pp. 217-227.

Basili, V. and Perricone, B. (1984), ‘Software Errors and Complexity: An Empirical Investigation’, *Communications of the ACM*, Vol. 27, No. 1, pp. 42-52.

Basili, V. R. (1980), 'Data Collection, Validation, and Analysis', In: *Tutorial on Models and Metrics for Software Management and Engineering*, Los Alamitos, California, IEEE Computer Society Press., pp. 310-313.

Baskerville, R. and Lee, A. (1999), 'Distinctions Among Different Types of Generalizing in Information Systems Research', In: Ngwenyama, O., Introna, L., Myers, M. and DeGross, J. I. (eds.), *New Information Technologies in Organizational Processes: Field Studies and Theoretical Reflections on the Future of Work*, New York, Kluwer Academic Publishers. pp. 49-65.

Baskerville, R. and Pries-Heje, J. (1999), 'Grounded Action Research: A Method For Understanding IT in Practice', *Accounting, Management and Information Technology*, Vol. 9, pp. 1-23.

Baskerville, R., Pries-Heje, J. and Venable, J. (2009), 'Soft Design Science Methodology', In: *Proceedings of the 4th international Conference on Design Science Research in information Systems and Technology*, Philadelphia, Pennsylvania, May 07 - 08. New York, NY, ACM, pp. 1-11.

Baskerville, R., Travis, J., and Truex, D. (1992), 'Systems Without Method: The Impact of Computer Supported Technologies on Information Systems Development', In: Kendall, K., Lyytinen, K. and DeGross, J. (eds.), *The Impact of Computer Supported Technologies on Information Systems Development*, North-Holland, Amsterdam, Elsevier Science Publishers B. V.

Beck, K. (2000), 'Emergent Control in Extreme Programming', *Cutter IT Journal: The Journal of Information Technology Management*, Vol. 13, No. 11, pp. 22-25.

Beck, K. and Fowler, M. (2001), *Planning eXtreme Programming*, Reading, Massachusetts, Addison-Wesley Publishing, Inc.

Behrsin, M., Mason, G. and Sharpe, T. (1994), *Reshaping IT for Business Flexibility: The IT Architecture as a Common Language for Dealing with Change*, New York, NY, McGraw-Hill.

Benbasat, I. and Zmud, R. W. (1999), 'Empirical Research in Information Systems: The Practice of Relevance', *MIS Quarterly*, Vol. 23, No. 1, pp. 3-16.

Benbasat, I., Goldstein, D. K. and Mead, M. (1987), 'The Case Research Strategy in Studies of Information Systems', *MIS Quarterly*, Vol. 11, No. 3, pp. 369-386.

Bengtsson, P. and Bosch, J. (1999), 'Architecture Level Prediction of Software Maintenance', In: *Proceedings of the Third European Conference on Software Maintenance and Reengineering (CSMR)*, Amsterdam, The Netherlands, March 03-05. Washington, DC, IEEE Computer Society, pp. 139.

Bengtsson, P., Lassing, N., Bosch, J. and van Vliet, H. (2004), 'Architecture-level modifiability analysis (ALMA)', *Journal of Systems and Software*, Vol. 69, No. 1-2, pp. 129-147.

Bennett, K. H. (1991), 'Automated Support of Software Maintenance', *Information and Software Technology*, Vol. 33, No. 1, pp. 74-85.

Bennett, K. H. and Rajlich, V. T. (2000), 'Software Maintenance and Evolution: A Roadmap', In: *Proceedings of the International Conference on the Future of Software Engineering (ICSE)*, Limerick, Ireland, June 04-11. New York, NY, ACM, pp. 73-87.

Berard, E. V. (1996), *Basic Object Orientated Concepts*, [Online document], http//www.toa.com/oobasics.htm, The Object Technology Inc.

Berkman, J. (1997), 'Web Design and Maintenance', In: *Proceedings of the 25th Annual ACM SIGUCCS Conference on User Services: Are you ready?*, Monterey, California, USA, November 09-12, New York, USA, ACM, pp. 23-27.

Berners-Lee, T. and Fischetti, M. (2008), *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by its Inventor*, New York, HarperCollins.

Bhatt, P., Shroff, G. and Misra, A. K. (2004), 'Dynamics of Software Maintenance', *SIGSOFT Software Engineering Notes*, Vol. 29, No. 5, pp. 1-5.

Bieber, M. and Isakowitz, T. (1995), 'Designing Hypermedia', *Communications of the ACM*, Vol. 38, No. 8, pp. 27-29.

Bieber, M., Conger, S., Ives, B., Janko, W. and O'Keefe, B. (1998), 'What's so Different about the World Wide Web anyway?', In: *Proceedings of the Nineteenth International Conference on Information Systems (ICIS)*, Helsinki, Finland, December 13-16, Helsinki. Finland, ACIS, pp. 415-419.

Bijker, W., Hughes, T. and Pinch, T. (1987), *The Social Construction of Technological Systems*, Cambridge, MA, MIT Press.

Black, S. E. (2001), ‘Computing Ripple Effect for Software Maintenance’, *Journal of Software Maintenance: Research and Practice*, Vol. 13, No. 4, pp. 263-279.

Boardman, J. and Sauser, B. (2007), ‘Systems of Systems - the meaning of of’, *Symposium on Complex Systems Engineering*, Santa Monica, January 11-12. California, LA, California State University.

Boehm, B. (2002), ‘Get Ready for Agile Methods, with Care’, *IEEE Computer*, Vol. 35, No. 1, January, pp. 64-69.

Boehm, B., Egyed A., Kwan, J., Port, D., Shah, A. and Madachy, R. (1998), ‘Using the Win-Win Spiral Model: A Case Study’, *IEEE Computer*, Vol. 31, No. 7, pp. 33-44.

Bohner, S. A. and Arnold, R. S. (eds.) (1996), *Software Change Impact Analysis*, Los Alamitos, CA, USA, IEEE Computer Society Press.

Bohner, S. A. (1996), ‘Impact Analysis in the Software Change Process: A year 2000 Perspective’, In: *Proceedings of the International Conference on Software Maintenance (ICSM)*, Monterey, CA, November 04-08. Washington, DC, IEEE Computer Society, pp. 42-51.

Bohner, S. A. (2002), ‘Software Change Impacts - An Evolving Perspective’, In: *Proceedings of the International Conference on Software Maintenance (ICSM)*, Montréal, Canada, October 03-06. Washington, DC, IEEE Computer Society, pp. 263-272.

Boland, R. (1985), 'Phenomenology: A Preferred Approach to Research in Information Systems', In: Mumford, E., Hirschheim, R. A., Frizgerald, G. and Woodharper, T. (eds.), *Research Methods in Information Systems*, North Holland, Amsterdam, North-Holland Publishing Co., pp. 193-201.

Boldyreff, C. and Tonella, P. (2004), 'Web Site Evolution', *Journal of Software Maintenance and Evolution: Research and Practice*, Vol. 16, No. 1-2, pp. 01-04.

Boogaard, M. (1994), *Defusing the Software Crisis: Information Systems Flexibility Through Data Independence*, Amsterdam, Holland, Tinbergen Institute Research Series. No. 79.

Boogaard, M. and Spoor, E. R. K. (1993), *Information Systems Flexibility: A Conceptual Framework*, Holland, Serie research memoranda, 1993-1959.

Bowker, G. C. and Star, S. L. (2000), *Sorting Things Out: Classification and its Consequences*, Cambridge, MA, The MIT Press.

Boynton, A. C. (1993), 'Achieving Dynamic Stability through Information Technology', *California Management Review*, Vol. 24, pp. 58-77.

Brereton, P., Budgen, D. and Hamilton, G. (1998), 'Hypertext: The Next maintenance Mountain', *IEEE Computer*, Vol. 31, No. 12, pp. 49-55.

Bryman, A. (1989), *Research Methods and Organisation Studies*, London, Uwin Hyman.

Brynjolfsson, E., Renshaw, A. A. and Van Alstyne, M. (1997), 'The Matrix of Change', *Sloan Management Review*, Vol. 38, No. 2, pp. 37-55.

BSI Corporate (2006), 'Software engineering, Software life cycle processes, Maintenance', *BSIBS ISO/IEC* 14764:2006.

Buckley, J., Mens, T., Zenger, M., Rashid, A. and Kniesel, G. (2005), 'Towards a taxonomy of software change', *Journal of Software Maintenance and Evolution: Research and Practice*, Vol. 17, pp. 309-332.

Burgess, S., Bingley, S. and Sellitto, C. (2006), 'Changes to the Websites of Micro Businesses: The Need to Look Closer', In: *Proceedings of the 29th Institute for Small Business and Entrepreneurship (ISBE) Conference: International Entrepreneurship*, Cardiff, Wales, 31 October - 2 November. London, ISBE , pp. 1-12.

Bushcman, F., Meunier, R., Rohnert, H ., Sommerlad, P. and Stal, M. (1999), *Pattern - Oriented Software Architecture: A System of Patterns*, Chichester, UK, Wiley.

Byrd, T. A. and Turner, D. E. (2000), 'Measuring the Flexibility of Information Technology Infrastructure: Exploratory analysis of a Construct', *Journal of Management Information Systems*, Vol. 17, No.1, pp. 167-208.

Callon, M. (1991), 'Techno-economic networks and irreversibility', In: Law, J. (eds), *A Sociology of Monsters: Essays on Power, Technology and Domination, Sociological Review Monograph*, London, Routledge.

Callon, M. (1992), 'The dynamics of techno-economic networks', In: Coombs, R., Saviotti, P. and Walsh, V. (eds.), *Technological Change and Company Strategies: Economic and Sociological Perspectives*, London, Academic Press.

Callon, M. and Bowker, G. (1993), ‘Is Science a Public Good?’, *Science, Technology & Human Values*, Vol. 19, No. 4, pp. 395-424.

Candell, O., Karim, R. and Söderholm, P. (2009), ‘eMaintenance Information logistics for maintenance support’, *Robotics and Computer-Integrated Manufacturing*, Vol. 25, No. 6, pp. 937-944.

Canning, R. G. (1972), ‘That Maintenance Iceberg’, *EDP Analyzer*, Vol. 10, No. 10, pp. 1-14.

Canning, R. G. (1981), ‘Easing the Software Maintenance Burden’, *EDP Analyzer*, Vol. 19, No. 8, pp. 1-14.

Cartwright, M. (2000), ‘Empirical Perspectives on Maintaining Web Systems: A Short Review’, In: *Proceedings of the 6th International IEEE Workshop on Empirical Studies of Software Maintenance (WESS)*, San Jose, California, October 14. Los Alamitos CA, IEEE Computer Society Press, pp. 05.

Cash, J. T. and Lawrence, P. (1989), *The Information Systems Research Challenge: Qualitative Research Methods*, Boston, Harvard Business School Research Colloquin.

Cavaye, A. L. M. (1996), ‘Case Study Research: A Multi-faceted Research Approach for IS’, *Information Systems Journal*, Vol. 6, No. 3, pp. 227-242.

Ceri, S., Fraternali, P. and Bongio, A. (2000), ‘Web Modeling Language (WebML): a modelling language for designing Web sites’, *Computer Networks: The International Journal of Computer and Telecommunications Networking*, Vol. 33, No.1-6, pp. 137-157.

Chapin, N. (2000), ‘Software Maintenance Types - A Fresh View’, In: *Proceedings of the International Conference on Software Maintenance (ICSM)*, San Jose, California, USA, October 11-14. Washington, DC, IEEE Computer Society, pp. 247-252.

Chapin, N., Hale, J. E., Kham, K. M., Ramil, J. F. and Tan, W. (2001), ‘Types of Software Evolution and Software Maintenance’, *Journal of Software Maintenance: Research and Practice*, Vol. 13, No. 1, pp. 3-30.

Chaumun, M. A., Kabaili, H., Keller, R. K. and Lustman, F. (1999), ‘A Change Impact Model for Changeability Assessment in Object-Oriented Software Systems’, In: *Proceedings of the 3rd European Conference on Software Maintenance and Reengineering (CSMR)*, Amsterdam, Netherlands, March 03-05. Washington, DC, IEEE Computer Society, pp. 130-138.

Chaumun, M. A., Kabaili, H., Keller, R. K. and Lustman, F. (2002), ‘A Change Impact Model for Changeability Assessment in Object-Oriented Software Systems’, *Science of Computer Programming*, Vol. 45, No. 2-3, pp. 155-174.

Chen, C. and Chen, P. (2009), ‘A Holistic Approach to Managing Software Change Impact’, *Journal of Systems and Software*, Vol. 82, No. 12, pp. 2051-2067.

Cherinka, R., Miller, R. and Smith C. (2005), ‘Beyond Web Services: Toward On-Demand, Complex, adaptive Environments’, In: *Proceedings of the IEEE International Conference on Web Services (ICWS)*, Orlando, FL, July 11-15. Los Alamitos, CA , IEEE Computer Society, pp. 815-816.

Ciborra, C. U. (1999), ‘A Theory of Information Systems Based on Improvisation’, In: Currie, W. L. and Galliers, B. (eds.), *Rethinking Management Information Systems*, Oxford, Oxford University Press, pp. 136-155.

Clemons, E. K. (1991), ‘Special section: Strategic and Competitive Information Systems’, *Journal of Management Information Systems*, Vol. 8, No. 2, pp. 7-9.

Clemons, E. K., Row, M. C. and Thatcher, M. E. (1995), ‘An Integrative Framework for Identifying and Managing risks Associated with Large Scale Reengineering Efforts’, In: *Proceedings of the 28th Hawaii International Conference on System Sciences (HICSS)*, January 04-07, Washington, DC, IEEE Computer Society, pp. 960-969.

Coad, P. and Youdon, E. (1991a), *Object - Orientated Analysis*, Second Edition, New Jersey, USA, Yourdon Press.

Coad, P. and Youdon, E. (1991b), *Object - Orientated Design*, New Jersey, USA, Yourdon Press.

Cockburn, A. (2000), ‘Balancing Lightness with Sufficiency’, *Cutter IT Journal: The Journal of Information Technology Management*, Vol. 13, No. 11, pp. 26-33.

Cockburn, A. (2002), ‘Agile Software Development joins the 'would-be' Crowd’, *Cutter IT Journal: The Journal of Information Technology Management*, Vol. 15, No. 1, January, pp. 6-12.

Collofello, J. S. (1989), ‘Improving Software Maintenance Skills in an Industrial Environment’, In: *Proceedings of the SEI Conference on Software Engineering Education (CSEE)*, Pittsburgh, Pennsylvania, USA, July 18-21, London, Springer-Verlag, pp. 26-36.

Cooper, R. B. and Zmud, R. W. (1990), 'Information technology implementation research: A technological diffusion approach', *Management Science*, Vol. 36, pp. 123-139.

Corbato, F. J. (1991), 'On Building Systems That will Fail', *Communications of the ACM*, Vol. 34, No. 9, pp. 72-81.

Crowe, A. (1992), 'Integration is Not Synonymous with Flexibility', *International Journal of Operations and Product Management*, Vol. 12, No. 10, pp. 26-33.

Currie, W. (2000), *The Global Information Society*, Chichester, John Wiley & Sons, Inc.

Curtis, B., Sheppard, S. B., Milliman, P., Borst, M. A. and Love, T. (1998), 'Measuring the Psychological Complexity of Software Maintenance Tasks with the Halstead and McCabe Metrics', *IEEE Transactions on Software Engineering*, Vol. 5, No. 2, pp. 96-104.

Cusumano, M. A. and Yoffie, D. B. (1998), *Competing on Internet Time/ Lessons from Netscape and Its Battle with Microsoft*, New York, The Free Press.

Cusumano, M. A. and Yoffie, D. B. (1999a), 'Software Development on Internet Time', *IEEE Computer*, Vol. 32, No. 10, pp. 60-69.

Cusumano, M. A. and Yoffie, D. B. (1999b), 'What Netscape learned from cross-platform Software Development', *Communications of the ACM*, Vol. 42, No.10, pp. 72-78.

De Troyer, O. M. F. and Leune, C. J. (1998), 'WSDM: a user centered design method for Web sites', *Computer Networks and ISDN Systems*, Vol. 30, No. 1-7, pp. 85-94.

DeSanctis, G. (1993), ‘Social Construction of Communication Technology’, *Academy of Management Review*, Vol. 36, No. 5, pp. 921-950.

Deshpande, Y., Murugesan, S., Ginige, A., Hansen, S., Schwabe, D., Gaedke, M. and White, B. (2002), ‘Web Engineering’, *Journal of Web Engineering*, Vol. 1, No. 1, pp. 003-017.

Dishaw, M. T. and Strong, D. M. (1998), ‘Assessing Software Maintenance Tool Utilization using Task-technology Fit and Fitness-for-Use Models’, *Journal of Software Maintenance: Research and Practice*, Vol. 10, pp. 151-179.

Doherty, N., Coombs, C. and Loan-Clarke, J. (2006), ‘A re-conceptualization of the interpretive flexibility of information technologies: redressing the balance between the social and the technical’, *European Journal of Information Systems*, Vol. 15, pp. 569–582.

Doke, E. R. and Swanson, N. E. (1991), ‘Software Maintenance Revisited: A Product Life Cycle Perspective’, *Information Executive*, Winter, pp. 8-11.

D'Souza, D. and Willis, A. C. (1999), *Objects, Components and Frameworks with UML: The CATALYSIS Approach*, Reading, MA, USA, Addison-Wesley.

Dugmore, J. and Taylor, S. (2008), *ITILo V3 and ISO/IEC 20000 For IT Service Management*, Alignment, White Paper, March.

Eardley, A., Avison, D. and Powell, P. (1997), ‘Developing Information Systems to Support Flexible Strategy’, *Journal of Organizational Computing and Electronic Commerce*, Vol. 7, No 1, pp. 57-77.

Earl, M. (1988), 'IT and Strategic Advantage: A Framework of Frameworks', In: Earl, M. (eds.), *Information Management: The Strategic Dimension*. Oxford, Clarendon Press, pp. 34-53.

Eisenhardt, K. M. (1989), 'Building Theories from Case Study Research', *Academy of Management Review*, Vol. 14, No. 4, pp. 532-550.

Eppink, D. J. (1975), *Flexibility: A Necessary Complement to Planning*, The Netherlands, Free University of Amsterdam.

Eppink, D. J. (1978), 'Planning for Strategic Flexibility', *Long-Range Planning*, Vol. 11, No. 4.

Eschenfelder, K. R. and Sawyer, S. (2001), 'Web Information Systems Management: Proactive or Reactive Emergence?', In: Russo, N. L., Fitzgerald, B. and DeGross, N. (eds.), *Realigning Research and Practice in IS Development: The Social and Organizational Perspective (Proceedings of the IFIP 8.2 Conference)*, Boston, MA, Kluwer Academic Publishers, pp. 163-181.

Fagrell, H. and Ljungstrand, P. (1998), 'Make and Agent and You Shall Find', In: *Proceedings of the 21st Information Systems Research Seminar In Scandinavia*, Sæby Søbad, Denmark, August 8-11. Göteborg, Sweden, IRIS Association, pp. 197-206.

Feagin, J., Orum, A. and Sjoberg, G. (1991), *A Case for Case Study*, Chapel Hill, NC, University of North Carolina Press.

Feldman, *M. 1995. Strategies for interpreting Qualitative Data*, Newbury Park, California, Sage Publications.

Fitzgerald, B. (1997), *Methodology-in-Action: The Nature of Usage of System Development Methodologies in Practice*, Unpublished PhD. Thesis, London, Birkbeck University of London.

Fitzgerald, G. (1988), 'Information Systems Development for Changing Environments: Flexibility Analysis', In: *Proceedings of the 1st European Conference on Information Technology for Organisational Systems: Concepts for Increased Competitiveness*, Athens, May. Amsterdam, Elsevier Science, pp. 587-593.

Fitzgerald, G. (1990), 'Achieving Flexible Information Systems: The Case for Improved Analysis', *Journal of Information Technology*, Vol. 5, No. 1, pp. 5-11.

Fitzgerald, G., Philippides, A. and Probert, S. (1998), 'Maintenance, Enhancement and Flexibility in Information Systems Development', In: *Proceedings of the 7th International Conference on Information Systems Development (ISD)*, Bled, Slovenia.

Fitzgerald, G., Philippides, A. and Probert, S. (1999), 'Information Systems Development, Maintenance and Enhancement: findings from a UK study', *International Journal of Information Management*, Vol. 19, No. 4, pp. 319-328.

Fowler, M. (2000), 'Put Your Process on a Diet', *Software Development*, Vol. 8, No. 12.

Fowler, M. (2003), *Patterns of Enterprise Application Architecture*, Reading, MA, Addison-Wesley Signature Series.

Frazelle, E. H. (1986), 'Flexibility: A Strategic Response in Changing Times', *Industrial Engineering*, March, pp. 17-20.

Friedlein, A. (2003), *Maintaining and Evolving Successful Commercial Web Sites: Managing Change, Content, Customer Relationships, and Site Measurement*, San Francisco, CA, Morgan Kaufmann Publishers.

Fyson, M. J. and Boldyreff, C. (1998), 'Using Application Understanding to support Impact Analysis', *Journal of Software Maintenance: Research and Practice*, Vol. 10, No. 2, pp. 93-110.

Gable, G. (1994), 'Integrating case study and survey research methods: an example in information systems', *European Journal of Information Systems*, Vol. 3, No. 2, pp. 112-126.

Galbraith, J. (1977), *Organization Design*, Reading, MA , Addison-Wesley Publishing, Inc.

Galliers, R. D. (1987), 'Information Systems and Technology Planning in a Competitive Strategy Framework', *Information Management*, Vol. 14, No. 7, pp. 27-38.

Galliers, R. D. (1993), 'Towards a Flexible Information Architecture: Integrating Business Strategies, Information Systems Strategies and Business Process Redesign', *Journal of Information Systems*, Vol. 3, No. 3, pp. 199-213.

Galliers, R. D. (1991), *Choosing Appropriate Information Systems Research Approaches: A Revised Taxonomy*, Amsterdam, North Holland, Elsevier Science. pp. 327-345.

Galliers, R. D. (1992), *Information System Research: Issues, Methods, and Practical Guidelines*, Oxford, Blackwell Scientific Publications.

Galliers, R. D. and Land, F. F (1987), ‘Choosing Appropriate Information Systems Research Methodologies’, *Communications of the ACM*, Vol. 30, No. 11, pp. 900-902.

Garzotto, F., Mainetti, L. and Paolini, P. (1995), ‘Hypermedia Design, Analysis, and Evaluation issues’, *Communications of the ACM*, Vol. 38, No. 8, pp. 74-86.

Gellersen, H., Wicke, R. and Gaedke, M. (1997), ‘Web Composition: an object-oriented support system for the Web engineering lifecycle’, In: *Proceedings of the 6th International Conference on World Wide Web*, Santa Clara, California, USA, April 7-11. Essex, UK, Elsevier Science Publishers Ltd., pp. 1429-1437.

Gill, G. K. and Kemerer, C. F. (1991), ‘Cyclomatic Complexity Density and Software Maintenance Productivity’, *IEEE Transactions on Software Engineering*, Vol. 17, No. 12, pp. 1284-1288.

Ginige, A. (1998), ‘Web Engineering: Methodologies for Developing Large and Maintainable Web Based Information Systems’, In: *Proceedings of the IEEE International Conference on Networking the India and the World CNIW’98*, Ahmeddabad, India, December 1998. Los Alamitos, CA , IEEE Computer Society.

Ginige, A. (2002), ‘Web Engineering: Managing the Complexity of Web Systems Development’, In: *Proceedings of the Software Engineering and Knowledge Engineering Conference (SEKE)*, Ischia, Italy, July 15-19. New York, USA, ACM, Vol. 27, pp. 721-729.

Ginige, A. and Murugesan, S. (2001a), ‘Web Engineering: An Introduction’, *IEEE Multimedia*, Vol. 8, No. 1, pp. 14-18.

Ginige, A. and Murugesan, S. (2001b), 'The Essence of Web Engineering - Managing the Diversity and Complexity of Web Application Development', *IEEE Multimedia*, Vol. 8, No. 2, pp. 22-25.

Ginige, A. and Murugesan, S. (2001c), 'Web Engineering: A Methodology for Developing Scalable, Maintainable Web Applications', *Cutter IT Journal: The Journal of Information Technology Management*, Vol. 14, No. 7, July, pp. 24-35.

Glass, R. L. (2001), 'Agile Versus Traditional: Make Love, Not War!', *Cutter IT Journal: The Journal of Information Technology Management*, Vol. 14, No. 12, pp. 12-18.

Gode, D. K., Barua, A. and Mukhopadhyay T. (1990), 'On the Economics of the Software replacement problem', In: *Proceedings of the 11th International Conference on Information Systems (ICIS)*, Copenhagen, Denmark, December 16-19. ICIS, pp. 159-170.

Golden, W. and Powell, P. (2000), 'Towards a Definition of Flexibility: In Search of the Holy Grail?', *Omega: The International Journal of Management Science*, Vol. 28, pp. 373-384.

Golden, W. and Powell, P. (2004), 'Inter-organisational Information Systems as enablers of Organisational Flexibility', *Technology Analysis and Strategic Management*, Vol. 16, No.3, pp. 299-325.

Gómez, J. and Cachero, C. (2003), 'OO-H Method: Extending UML to Model Web Interfaces', In: van Bommel, P. (eds.), *Information Modelling for Internet Applications*, Hershey, PA, IGI Publishing, pp. 144-173.

Gómez, J., Cachero, C. and Pastor, O. (2000), 'Extending a Conceptual Modelling Approach to Web Application Design', In: *Proceedings of the 12th International Conference on Advanced information Systems Engineering*, Stockholm, Sweden, June 05 - 09. London, Springer-Verlag, pp. 79-93.

Graham, I. (1995), *Object - Orientated Methods*, Second Edition, Addison-Wesley Publishing Company Inc.

Gremillion, L. L. (1984), 'Determinants of Program Repair Maintenance Requirements', *Communications of the ACM*, Vol. 27, No. 8, pp. 826-832.

Grisogono, A. M. (2007), 'The Success or Failure of Adaptation', *Symposium on Complex Systems Engineering*, Santa Monica, January 11-12. California, LA, California State University, pp. 1-6.

Gross, N., Giacquinta, J. B. and Bernstein, M. (1971), *Implementing Organizational Innovations: a sociological analysis of planned educational change*, New York, Basic Books.

Guba, E. G. and Lincoln, Y. S. (1981), *Effective Evaluation: Improving Responsive and Naturalistic Approaches*, San Francisco, CA, Jossey-Bass.

Guimaraes, T. (1983), 'Managing Application Program Maintenance Expenditures', *Communications of the ACM*, Vol. 26, No. 10, pp. 739-746.

Hamill, J. (1997), 'The Internet and International Marketing', *International Marketing Review*, Vol. 15, No. 5, pp. 300-323.

Haney, F. M. (1972), 'Module Connection Analysis: A Tool for Scheduling Software Debugging Activities', In: *Proceedings of the Fall Joint Computer Conference, Part I (AFIPS)*, Anaheim, California, December 5-7. New York, NY, ACM, pp. 173-179.

Hanseth, O., Monteiro, E. and Hatling, M. (1996), 'Developing Information Infrastructure: The tension between Standardization and Flexibility', *Science, Technology and Human Values*, Vol. 21, No. 4, pp. 407-426.

Harrison, W., Magel, K., Kluczny, R. and DeKock, A (1982), 'Applying Software Complexity Metrics to Program Maintenance', *IEEE Computer*, Vol. 15, No. 9, pp. 65-79.

Hart, B. L. J. (1964), *Dynamic Systems Design: company control for the computer era*, London Business Publications Limited.

Hawrysh, S. P. and Ruprecht, J. (2000), 'Its Like Deja Vu All Over Again', *Cutter IT Journal: The Journal of Information Technology Management*, Vol. 13, No. 11, pp. 4-12.

He, L. and Carver, J. (2009), 'Modifiability measurement from a task complexity perspective: A feasibility study', In: *Proceedings of the 3rd International Symposium on Empirical Software Engineering and Measurement (ESEM)*, Lake Buena Vista, Florida, USA , October 15-16. Washington, DC, IEEE Computer Society, pp. 430-434.

Hees, R. N. van, and Monhemius, W. (1970), *Produkiebesturing en Voorraadbeheer*, Theoretische Achtergronden, Deventer Kluwer.

Henderson, J. C. and Venkatraman, N. (1993), 'Strategic Alignment: Leveraging Information Technology for Transforming Organizations', *IBM Systems Journal*, Vol. 32, No. 1, pp. 4-16.

Howcroft, D., Mitev, N. And Wilson, M. (2004), 'What We May Learn from the Social Shaping of Technology Approach', In: Mingers, J. and Willcocks, L. (eds.), *Social theory and Philosophy for Information Systems*, London, John Wiley & Sons Ltd, pp. 329-371.

Iansiti, M. And MacCormack, A. D. (1997), 'Developing Products on Internet Time', *Harvard Business Review,* Vol. 75, No. 5.

IEEE (1990), 'IEEE Standard Glossary of Software Engineering Terminology', *IEEE Std 610.12-1990*, IEEESTD.1990.101064, New York, NY, IEEE, pp. 1.

IEEE (1998), 'IEEE Standard for Software Maintenance', *IEEE Std 1219-1998*, IEEESTD.1998.88278, New York, NY, IEEE.

IEEE (2006), 'International Standard - ISO/IEC 14764 IEEE Std 14764-2006 Software Engineering and Software Life Cycle Processes and Maintenance', *ISO/IEC 14764:2006 (E) IEEE Std 14764-2006*, Revision of IEEE Std 1219-1998, IEEESTD.2006.235774, New York, NY, IEEE, pp. 1-46.

Isakowitz, T., Bieber, M. and Vitali, F. (1998), 'Web Information Systems', *Communications of the ACM*, Vol. 41, No. 7, pp. 78-80.

Ittner, C. D. and Kogut, B. (1995), 'How Control Systems can Support Organizational Flexibility', In: Bowman, E. and Kogut B. (eds.), *Redesigning The Firm*. New York, USA, Oxford University Press, pp. 155-180.

Ives, B. and Learmonth, G. P. (1984), 'The Information System as a Competitive Weapon', *Communications of the ACM*, Vol. 27, No. 12, pp. 1193-1201.

Jeffries, R., Anderson, R. and Hendrickson, C. (2001), *Extreme Programming Installed*, Reading, Massachusetts, Addison-Wesley Publishing, Inc.

Jones, C. (1989), 'Software Enhancement Modelling', *Journal of Software Maintenance: Research and Practice*, Vol. 1, No. 2, pp. 91-100.

Joshi, A. and Singh, M. P. (1999), 'Multiagent Systems on the Net', *Communications of the ACM*, Vol. 42, No. 3, pp. 38-40.

Kafura, D. and Reddy, G. R. (1987), 'The Use of Software Complexity Metrics in Software Maintenance', *IEEE Transactions on Software Engineering*, Vol. 13, No. 3, pp. 335-343.

Kajko-Mattsson, M. (2004), 'Evolution and Maintenance of Web Service Applications', In: *Proceedings of the 20th IEEE International Conference on Software Maintenance (ICSM)*, Chicago Illinois, USA, September 11-17. Los Alamitos CA, IEEE Computer Society Press, pp. 492-493

Kanellis, P. (1996), *Information Systems and Business Fit in Dynamic Environments*. Unpublished Ph.D. Thesis, London, Brunel University.

Kaplan, B. and Duchon, D. (1988), 'Combining qualitative and quantitative methods information systems research: a case study', *Management Information Systems Quarterly*, Vol. 12, No. 4, pp. 571-586.

Katz, D. and Kahn, R. L. (1978), *The Social Psychology of Organizations*, Second Edition, New York, NY, John Wiley & Sons, Inc.

Kemerer, C. F. (1995), ‘Software complexity and software maintenance: A survey of empirical research’, *Annals of Software Engineering*, Vol. 1, No. 1, pp. 1-22.

Kemerer, C. F. and Slaughter, S. A. (1997), ‘Determinants of Software Maintenance Profiles: an Empirical Investigation’, *Journal of Software Maintenance and Evolution: Research and Practice*, Vol. 9, No. 4, pp. 235–251.

Ketler, K. and Turban, E. (1992), ‘Productivity Improvements in Software Maintenance’, *International Journal of Information Management*, Vol. 12, No. 1, pp. 70-82.

Kirovski, D., Drinic, M. and Potkonjak, M. (2001), ‘Hypermedia-aided Design’, In: *Proceedings of the 38th Annual Design Automation Conference (DAC)*, Las Vegas, Nevada, USA, June 18-22. New York, NY, ACM, pp. 407-412.

Kitchenham, B. A., Travassos, G. H., von Mayrhauser, A., Niessink, F., Schneidewind, N. F., Singer, J., Takada, S., Vehvilainen, R. and Yang, H. (1999), ‘Towards an Ontology of Software Maintenance’, *Journal of Software Maintenance: Research and Practice*, Vol. 11, No. 6, pp. 365-389.

Klein, H. K. (1999), ‘Knowledge and Methods in Research: for Beginnings to the Future’, In: Ngwenyama, O., Introna, L. D., Myers, M.D. and Gross, J. I. (eds.), *New Information Technologies in Organizational Processes: Field Studies and Theoretical Reflections on the Future of Work*, Boston, MA, Kluwer Academic Publishers.

Klein, H.K. and Myers, M. (2001), 'A Classification Scheme for Interpretive Research in Information Systems', In: Trauth, E. M. (eds.), *Qualitative Research in IS: Issues and Trends*, Hershey, PA, IGI Publishing., pp. 218-239.

Knoll, K. and Jarvenpaa, S. L. (1994), 'Information Technology Alignment or "Fit" in Highly Turbulent Environments: The Concept of Flexibility', In: *Proceedings of ACM SIGCPR '94 Conference*, Alexandria, VA, USA, March 24-26. New York, NY, ACM, pp. 1-14.

Koskinen, J., Salminen, A. and Paakki, J. (2004), 'Hypertext Support for the Information needs of Software Maintainers', *Journal of Software Maintenance and Evolution: Research and Practice*, Vol. 16, No. 3, pp. 187-215.

Kozaczynski, W. and Booch, G. (1998), 'Guest Editors Introduction: Component-Based Software Engineering', *IEEE Software*, Vol. 15, No. 5, pp. 34-36.

Krulwich, B. (1997), 'Automating the Internet: agents as user surrogates', *IEEE Internet Computing*, Vol. 1, No. 4, pp. 34-38.

Kung, D., Suchak, N., Gao, J., Hsia, P., Toyoshima, Y. and Chen, C. (1994), 'On Object State Testing', In: *Proceedings of the 18th Annual International Conference on Computer Software and Applications (COMPSAC)*, Taipei, Taiwan, November 9-11. Washington, DC, IEEE Computer Society, pp. 222-227.

Lambert, D. D. and Peppard, J. (1993), 'Information Technology and New Organizational Forms: Destination but no road map?', *Journal of Strategic Information Systems*, Vol. 2, No 3, pp. 108-205.

Land, F. (1982), ‘Adapting to Changing User Requirements’, *Information and Management*, Vol. 5, pp. 59-75.

Lang, M. (2001), ‘Issues and Challenges in the Development of Hypermedia Information Systems’, In: *Proceedings of 11th Annual Business Information Technology Conference (BIT 2001)*, Manchester, UK, October 30-31.

Lang, M. and Fitzgerald, B. (2005), ‘Hypermedia Systems Development Practices: A Survey’, *IEEE Software*, Vol. 22, No. 2, pp. 68-75.

Lang, M. and Fitzgerald, B. (2006), ‘New Branches, Old Roots: A Study of Methods and Techniques in Web/Hypermedia Systems Design’, *Information Systems Management*, Vol. 23, No. 3, pp. 62-74.

Lavery, J., Boldyreff, C., Ling, B. and Allison, C. (2004), ‘Modelling the Evolution of Legacy Systems to Web-based Systems’, *Journal of Software Maintenance and Evolution: Research and Practice*, Vol. 16, No. 1-2, pp. 05-30.

Lawrence, P. and Dyer, D. (1981), *Toward a Theory of Organizational and Industrial Adaptation*, Graduate School of Business Administration, Harvard University, USA, HBS-80-57.

Layzell, P. (2001), ‘Addressing the software evolution crisis through a service-oriented view of software: A roadmap for software engineering and maintenance research’, In: *Proceedings of the IEEE International Conference on Software Maintenance (ICSM)*, Florence, Italy, November 6-10. Washington, DC, IEEE Computer Society, pp. 5.

Lee, A. S. (1989), ‘A Scientific Methodology for MIS Case Studies’, *MIS Quarterly*, Vol. 13, No. 1, pp. 33-52.

Lee, A. S. and Baskerville, R. L. (2003), 'Generalizing Generalizability in Information Systems Research', *Information Systems Research*, Vol. 14 , No .3.

Lee, M. L. (1998), *Change Impact Analysis of Object-Oriented Software*, Unpublished PhD. Thesis, Virginia, George Mason University.

Lee, M., Offutt, A. J. and Alexander, R. T. (2000), 'Algorithmic Analysis of the Impacts of Changes to Object-Oriented Software', In*: Proceedings of the 34th International Conference on Technology of Object-Oriented Languages and Systems (TOOLS)*, Santa Barbara, CA, 30 July - August 03. Washington, DC, IEEE Computer Society, pp. 61-70.

Lee, T. W. (1999), *Using Qualitative Methods in Organizational Research*, Thousand Oaks, CA, Sage.

Lehman, M. and Belady, L. (1985), *Program Evolution Processes of Software Change*, London, UK, Academic Press.

Lehman, M. M. (1998), 'FEAST-Feedback, Evolution and Software Technology', In: *Proceedings of the 6th European Workshop on Software Process Technology (EWSPT)*, Weybridge, UK, September 16-18, London, Springer-Verlag, pp. 150

Li, L. and Offutt, A. J. (1996), 'Algorithmic Analysis of the Impact of Changes to Object-Oriented Software', In: *Proceedings of the International Conference on Software Maintenance (ICSM)*, Monterey, CA, November 04-08. Washington, DC, IEEE Computer Society, pp. 171-184.

Lieberman, H. (1995), 'Letizia: an agent that assists web browsing', In: *Proceedings of the 14th International Joint Conference on Artificial intelligence - Volume 1*, Montreal, Quebec, Canada, August 20-25. San Francisco, CA, Morgan Kaufmann Publishers, pp. 924-929.

Lientz, B. P. and Swanson, E. B. (1980), *Software Maintenance Management*, Reading, Massachusetts, Addison-Wesley Publishing, Inc.

Lientz, B. P., Swanson, E. B. and Tompkins, G. E. (1978), 'Characteristics of Application Software Maintenance', *Communications of the ACM*, Vol. 21, No. 6, pp. 466-471.

Ligezinski, P. (1988), 'Zero-Maintenance Systems', *Second place winner of the International Paper Competition of the Society of Information Management*.

Lincoin, Y. and Guba, E. (1985), *Naturalistic inquiry*, Beverly Hills, CA , Sage.

Linos, P., Ososanya, E. and Natarajan, H. (2001), 'Maintenance Support for Web Sites: A Case Study', In: *Proceedings of the 3rd International IEEE Workshop on Web Site Evolution (WSE)*, Florence, Italy, November 6-10. Los Alamitos CA, IEEE Computer Society Press, pp. 70-77.

Lipnack and Stamps (1997), *Virtual Teams: Reaching Across Space, Time, and Organizations with Technology*, New York, NY, John Wiley & Sons, Inc.

Liu, C. and Arnett, K. P. (2000), 'Exploring the Factors Associated with Web site Success in the Context of Electronic Commerce', *Information & Management*, Vol. 38, No. 1, pp. 23-33.

Livschitz, V. (2005), ‘Conquering the Crisis in Software Engineering’, In: *Proceedings of the 29th Annual International Conference on Computer Software and Applications Conference (COMPSAC)*, Edinburgh, UK, July 26-28. Washington, DC, IEEE Computer Society, Vol. 2, pp. 369.

Longworth, G. (1985), *Designing Systems for Change*, Manchester, England, NCC Publications.

Lowe, D. (1999), ‘Web Development Methodologies: Help or Hindrance?’, *WebNet Journal*, Vol. 1, No. 3, pp. 9-10 & 17.

Lowe, D. and Henderson-Sellers, B. (2001), ‘OPEN to Change: Addressing Web Development Process Differences’, *Cutter IT Journal: The Journal of Information Technology Management*, Vol. 14, No. 7, July, pp. 11-17.

Löwgren, J. and Stolterman, E. (1999), ‘Methods & Tools: Design Methodology and Design Practice’, *ACM Interactions*, Vol. 6, No. 1, pp. 13-20.

Lucas, H. C. Jr and Olson, M. (1994), ‘The Impact of Information Technology on Organizational Flexibility’, *Journal of Organizational Computing*, Vol. 4, No. 2, pp. 155-176.

Maes, P. (1994), ‘Agents that reduce work and Information overload’, *Communications of the ACM*, Vol. 37, No. 7, pp. 30-40.

Maglio, P. P. and Barrett, R. (1997), ‘How to build modelling agents to support web searchers’, In: *Proceedings of the 6th International Conference on User Modelling*, Chia Laguna, Sardinia, June 02-05. Springer, pp. 05-16.

Maier, M. W. (1996), *Architecting Principles for Systems-of-Systems*, [online], Needham, MA: Infoed, Available from: http://www.infoed.com/Open/PAPERS/systems.htm (Accessed last on 26 February 2010).

Maier, M. W. (1998), 'Architecting Principles for Systems-of-Systems', *Systems Engineering*, Vol. 1, No. 4, pp. 267-284.

Maier, M. W. (2007), 'Dimensions of Complexity other than "Complexity"', *Symposium on Complex Systems Engineering*, Santa Monica, January 11-12. California, LA, California State University.

Martin, J. and Martin, L. (2001), 'Web Site Maintenance with Software-Engineering Tools', In: *Proceedings of the 3rd International IEEE Workshop on Web Site Evolution (WSE)*, Florence, Italy, November 6-10. Los Alamitos CA, IEEE Computer Society Press, pp. 126-131.

Martin, J. and McClure, C. (1983), *Software Maintenance: The Problem and its Solutions*, Englewood Cliffs, New Jersey, Prentice-Hall, Inc.

Martin, R. J., and Osborne, W. M. (1983), *Guidance on Software Maintenance*, NBS, Special Publication, 500-106, U.S. Department of Commerce, National Bureau of Standards, Washington, D.C.

McCabe, T. J. (1976), 'A Complexity Measure', *IEEE Transactions on Software Engineering*, Vol. SE-2, No. 4, pp. 308-320.

McClure, C. (1989), *CASE is Software Automation*, Englewood Cliffs, New Jersey, USA, Prentice-Hall, Inc.

Mendo, F.A. and Fitzgerald, G. (2005), ‘A Multidimensional Framework for SME E-Business Progression’, *Journal of Enterprise Information Management*, Vol. 18, No. 6, pp. 678-696.

Merton, R. K., Fiske, M. and Kendall, P. L. (1990), *The Focused Interview: A Manual of Problems and Procedures*, Second Edition, New York , The Free Press.

Miles, M. B. and Huberman, A. M. (1994), *Qualitative Data Analysis: An Expanded Sourcebook*, Second Edition, Newbry Park, CA, Sage Publications, Inc.

Mintzberg, H. (1979), ‘An Emerging Strategy of 'Direct' Research’, *Administrative Science Quarterly*, Vol. 24, pp. 580-589.

Mintzberg, H. (1979), *The Structuring of Organization*, Englewood Cliffs, New Jersey, Prentice-Hall.

Mintzberg, H. (1981), ‘Organization Design: Fashion or Fit?’, *Harvard Business Review*, Vol. 59, No. 1, pp. 103.

Mintzberg, H. (1983), *Structure in Fives, Designing Effective Organizations*, Englewood Cliffs, New Jersey, Prentice-Hall.

Mintzberg, H. (1994), ‘The Fall and Rise of Strategic Planning’, *Harvard Business Review*, January-February, pp. 107-114.

Mintzberg, H. and Quinn, J. B. (1996), *The Strategy Process: Concepts, Contexts, Cases*, Englewood Cliffs, New Jersey, Prentice-Hall, Inc.

Misa, T. J. (1992), ‘Controversy and closure in technological change: constructing ‘steel’’, In: Bijker, W. E. and Law, J. (eds.), *Shaping Technology/Building Society: Studies in Sociotechnical Change*, Cambridge, MA, MIT Press, pp. 109-39.

Misa, T. J. (1994), ‘Retrieving Sociotechnical Change from Technological Determinism’, In: Smith, M. R. and Marx, L. (eds.), *Does Technology Drive History? The Dilemma of Technological Determinism*, Cambridge, MA, MIT Press, pp. 115-142.

Moitra, D. and Ganesh, J. (2005), ‘Web Services and Flexible Business Processes: Towards the Adaptive Enterprise’, *Information and Management*, Vol. 42, No. 7, pp. 921-933.

Morey, N. C. and Luthans, F. (1984), ‘An Emic Perspective and Ethnoscience Methods for Organizational Research’, *Academy of Management Review*, Vol. 9, No. 1, pp. 27-36.

Moss, L. T. (2001), ‘Business Intelligence Methodologies: Agile with Rigor?’, *Cutter IT Journal: The Journal of Information Technology Management*, Vol. 14, No. 12, pp. 19-26.

Munford, E. (1983), *Designing the Human Systems - The ETHICS Method*, The Manchester Business School, Manchester, UK.

Murugesan, S. and Deshpande, Y. (1999), ‘ICSE'99 workshop on Web Engineering’, In: *Proceedings of the 21st International Conference on Software Engineering (ICSE)*, Los Angeles, California May 16-22. New York, NY, ACM, pp. 693-694.

Murugesan, S. and Deshpande, Y. (2000), 'Second ICSE workshop on Web Engineering', In: *Proceedings of the 22nd International Conference on Software Engineering (ICSE)*, Limerick, Ireland June 04-11. New York, NY, ACM, pp. 794-795.

Murugesan, S. and Deshpande, Y. (2002), 'Meeting the Challenges of Web Application development: The Web Engineering Approach', In: *Proceedings of the 24th International Conference on Software Engineering (ICSE)*, Orlando, Florida May 19-25. New York, NY, ACM, pp. 687-688.

Murugesan, S., Deshpande, Y., Hansen, S. and Ginige, A. (2001), 'Web Engineering: A New Discipline for Development of Web-Based Systems'. In: Murugesan, S. and Deshpande, Y. (eds.), *Web Engineering 2000: Lecture Notes in Computer Science 2016 (LNCS)*, Berlin, Heidelberg, Springer-Verlag. pp. 3-13.

Myers, G. J. (1975), *Reliable Software Through Composite Design*, New York, Petrocelli/ Charter.

Myers, M. D. (1997), 'Qualitative Research in Information Systems', *MIS Quarterly*, Vol. 21, No. 2, pp. 241-242.

Navarro, A., Sierra, J. L., Fernandez-Vamayor, F. and Fernandez-Manjon, B. (2005), 'A First Step Towards the Web Engineering Body of Knowledge', In: Lowe, D. and Gaedke, M. (eds.), *Web Engineering: 5th International Conference, ICWE 2005*, Sydney, Australia, July 27-29, Proceedings, Lecture Notes in Computer Science 3579 (LNCS), Berlin, Heidelberg, Springer-Verlag, pp. 585-587.

Nelson, K. M. and Ghods, M. (1998), 'Measuring Technology Flexibility', *European Journal of Information Systems*, Vol. 7, No. 4, pp. 232-240.

Nelson, K. M. and Nelson, H. J. (1997), 'Technology Flexibility: Conceptualization, Validation and Measurement', In: *Proceedings of the 30th Hawaii international Conference on System Sciences (HICSS): Information System Track-Organizational Systems and Technology - Volume 3*, January 03-06, Washington, DC, IEEE Computer Society, pp. 76-87.

Nikora, A. P. and Munson, J. C. (1998), 'Determining Fault Insertion Rates for Evolving Software Systems', In: *Proceedings of the 9th International Symposium on Software Reliability Engineering*, Paderborn, Germany, November 04-07. Washington, DC, IEEE Computer Society, pp. 306.

Northrop, L. (2008), *Ultra-Large-Scale Systems: Scale Changes Everything*, [online], Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, Available from: http://www.sei.cmu.edu/library/assets/northropkeynote.pdf (Accessed last on 26 February 2010).

Northrop, L., Feiler, P., Gabriel, R. P., Goodenough, J., Linger, R., Longstaff, T., Kazman, R., Klein, M., Schmidt, D., Sullian, K. and Wallnau, K. (2006), *Ultra-Large-Scale Systems: The Software Challenge of the Future*, Pittsburgh, PA, Software Engineering Institute, Carnegie Mellon University.

Orfali, R., Harkey, D. and Edwards, J. (1999), *Client/ Server Survival Guide*, Third Edition, New York, John Wiley & Sons, Inc.

Orlikowski, W. J. (2000), 'Using Technology and Constituting Structures: A Practice Lens for Studying Technology in Organizations', *Organization Science*, Vol. 11, No. 4, pp. 404-428.

Orlikowski, W. J. and Baroudi, J. J. (1991), ‘Studying Information Technology in Organizations: Research Approaches and Assumptions’, *Information Systems Research*, Vol. 2, pp. 1-28.

Oxford Corpus (2008), *Compact Oxford dictionary 11th revised edition*, Oxford, Oxford University Press.

Palmer, J. D. (1999), ‘Systems Integration’, In: Sage, A. P. and Rouse, W. B. (eds.), *Handbook of Systems Engineering and Management*, New York, John Wiley & Sons, Inc., pp. 483-518.

Paul, R. J. (1993), ‘Dead Paradigms for Living Systems’, In: *Proceedings of the 1st European Conference on Information Systems (ECIS)*, Henley-on-Thames, March 29-30. Birmingham, The Operational Research Society, pp. 250-255.

Paul, R. J. (1994), ‘Why Users Cannot 'Get What They Want'’, *International Journal of Manufacturing System Design*, Vol. 1, No. 4, pp. 389-394.

Paul, R. J. (1995), ‘An O.R. View of Systems Development’, In: Lawrence, M. and Wilsdon, C. (eds.), *Operational Research Tutorial Papers*, UK, Operational Research Society, pp. 46-56.

Pettigrew, A. (1985), *The Awakening Giant – Continuity and Change at ICI*, Oxford, Basil Blackwell.

Pfleeger, S. L. and Bohner, S. A. (1990), ‘A Framework for Software Maintenance Metrics’, In: *Proceeding of the Conference on Software Maintenance (CSM)*, San Diego, CA, USA, November 26-29. New York, IEEE Computer Society Press, pp. 320-327.

Polo, M., Piattini, M. and Ruiz, F. (2002a), 'Integrating Outsourcing in the Maintenance Process', *Information Technology and Management*, Vol. 3, No. 3, pp. 247-269.

Polo, M., Piattini, M. and Ruiz, F. (2002b), 'Using a Qualitative Research Method for building a Software Maintenance Methodology', *Software: Practice & Experience*, Vol. 32, No. 13, pp. 1239-1260.

Polo, M., Piattini, M., Ruiz, F. and Calero, C. (1999a), 'Roles in the Maintenance Process', *ACM SIGSOFT Software Engineering Notes*, Vol. 24, No. 4, pp. 84-86.

Polo, M., Piattini, M., Ruiz, F. and Calero, C. (1999b), 'MANTEMA: A Complete Rigorous Methodology for Supporting Maintenance Based On the ISO/IEC 12207 Standard', In: *Proceedings of the 3rd European Conference on Software Maintenance and Reengineering (CSMR)*, Amsterdam, Netherlands, March 03-05. Washington, DC, IEEE Computer Society, pp. 178-181.

Polo, M., Piattini, M., Ruiz, F. and Calero, C. (1999c), 'MANTEMA: A Software Maintenance Methodology Based on the ISO/IEC 12207 Standard', In: *Proceedings of the 4th IEEE international Symposium and Forum on Software Engineering Standards (ISESS)*, Curitiba, Brazil, May 17-21. Washington, DC, IEEE Computer Society, pp. 76-81.

Potier, D., Albin, J. L., Ferreol, R. and Bilodeau, A. (1982), 'Experiments with Computer Software Complexity and Reliability', In: *Proceedings of the 6th International Conference on Software Engineering*, Tokyo, Japan, September 13-16. Los Alamitos, CA, IEEE Computer Society Press, pp. 94-103.

Pressman, R. S. (1994), *Software Engineering: A Practitioner's Approach*, European Edition, Maidenhead, UK, McGraw-Hill.

Pressman, R. S. (1998), ‘Can Internet-Based Applications be Engineered?’, *IEEE Software*, Vol. 15, No. 5, pp. 104-110.

Pressman, R. S. (2001), ‘What a Tangled Web we Weave’, *IEEE Software*, Vol. 18, No. 1, pp. 18-21.

Prevelakis, V. (1998), ‘A Framework for the Management of Large WWW Sites’, In: *Proceedings of the International Network Conference (INC)*, Plymouth, United Kingdom, July 6-9.

Quinn, J. B. (1980), ‘Managing Strategic Change’, *Sloan Management Review*, Vol. 21, No. 4, pp. 3-20.

Quinn, R. E. (1996), *Deep Change: Discovering the leader within*, San Francisco, Jossey-Bass.

Ramanujan, S. and Cooper, R. B. (1994), ‘A Human Information Processing Perspective on Software Maintenance’, *Omega*, Vol. 22, No. 2, pp. 185-203.

Reil, A. J. (1996), *Object - Oriented Design Heurristics*, Addison-Wesley Publishing, Inc.

Reutter III, J. (1981), ‘Maintenance is a Management Problem and a Programmer's Opportunity’, In: *Proceedings of the National Computer Conference (AFIPS '81)*, Chicago, Illinois, May 04-07. New York, NY, ACM, pp. 343-347.

Rhenman, E. (1973), *Organization Theory for Long - Range Planning*, London, UK, John Wiley & Sons, Inc.

Ricca, F. and Tonella, P. (2006), ‘Detecting Anomaly and Failure in Web Applications’, *IEEE Multimedia*, Vol. 13, No. 2, pp. 44-51.

Rising, L. and Janoff, N. S. (2000), ‘The Scrum Software Development Process for Small Teams’, *IEEE Software*, Vol. 17, No. 4, July, pp. 26-32.

Rochester, J. (1990), ‘Change management in information systems‘, *I/S Analyzer*, Vol. 28, pp.1-16.

Rochester, J. B. (1989),’ Building More Flexible Systems’, *I/S Analyzer*, Vol. 27, No 10, pp. 1-12.

Rosenhead, J. (1989), *Relational Analysis for a Problematic World: Problem Structuring Methods for Complexity*, Uncertainty and Conflict, NY, USA, John Wiley & Sons, Inc.

Rossi, G., Schwabe, D. and Garrido, A. (1997), ‘Design reuse in Hypermedia Applications Development’, In: *Proceedings of the 8th ACM Conference on Hypertext*, Southampton, United Kingdom, April 06-11. New York, NY, ACM, pp. 57-66.

Rubart, J. (2008), ‘Hypermedia Design Patterns’, In: *Proceedings of the 19th ACM Conference on Hypertext and Hypermedia (HT)*, Pittsburgh, PA, USA, June 19-21. New York, NY, ACM, pp. 243-244.

Rubin, H. J. and Rubin, I. S. (1995), *Qualitative interviewing: The Art of Hearing Data*, Thousand Oaks, CA., Sage.

Rumbaugh, J., Blaha, M., Premerani, W., Eddy, and Loresen, W. (1991), *Object - Oriented Modelling and Designs*, Englewood Cliffs, USA, Prentice-Hall, Inc.

Russo, N. and Graham, B. (1999), 'A first step in developing a Web application design methodology: understanding the environment', In: Wood-Harper, A. T., Jayaratma, N., Wood, J. R. G. (eds), *Methodologies for Developing and Managing Emerging Technology based Information Systems, Proceedings of the 6th International Conference on Information Systems Methodology (BCS)*, Salford University, Manchester. London, Springer, pp. 24-33.

Salmela, H., Lederer, A. and Reponen, T. (2000), 'Information Systems Planning in a turbulent Environment', *European Journal of Information Systems*, Vol. 9, No. 1, pp. 3-15.

Sarantakos, S. (2005), *Social Research*, Hampshire, Pelgrave Macmillan.

Schach, S. R. and Tomer, A (2000), 'A Maintenance-Oriented Approach to Software Construction', *Journal of Software Maintenance: Research and Practice*, Vol. 12, pp. 25-45.

Scharl, A. (2000), *Evolutionary Web Development*, London, Springer-Verlag.

Schneidewind, N. F. (1987), 'The State of Software Maintenance', *IEEE Transactions on Software Engineering*, Vol. 13, No. 3, pp. 303-310.

Schwaber, K. (2000), 'Against a Sea of Troubles: Scrum Software Development', *Cutter IT Journal: The Journal of Information Technology Management*, Vol. 13, No. 11, pp. 34-39.

Scott, B. W. (1965), *Long-Range Planning in American Industry*, New York, USA, American Management Association.

Siddiqui, F. A. (2002), *Understanding & Measuring Systems Flexibility: A Case of Object-Oriented Development Environment*, Unpublished PhD. Thesis, London, Brunel University.

Silverman, D. (2001), *Interpreting Qualitative Data: Methods for Analysing Talk, Text and Interaction*, Second Edition, London, Sage Publications Ltd.

Smith, D. and Lewis, G. (2008), ‘Systems of Systems: New Challenges for Maintenance and Evolution’, In: *Proceedings of the 24th IEEE International Conference on Software Maintenance (ICSM) FoSM*, Beijing, China, September 28 - October 4. Washington, DC, IEEE Computer Society, pp. 149-157.

Smith, N. L. (1990), ‘Cautions on The use of Investigative Case Studies in Meta-Evaluation’, *Evaluation and Program Planning*, Vol. 13, pp. 373-378.

Soong, N. L. (1977), ‘A Program Stability Measure’, In: *Proceedings of the ACM Annual Conference*, Seattle, Washington, October, 17-19. New York, NY, ACM, pp. 163-173.

Sørensen, C., Macklin, D. and Beaumont, T. (2001), ‘Navigating the World Wide Web: Bookmark Maintenance Architectures’, *Interacting with Computers*, Vol. 13, No. 3, pp. 375-400.

Stigler, G. J. (1939), ‘Production and Distribution in the Short Run’, *Journal of Political Economy*, Vol. 47, pp. 305-327.

Strauss, A. and Corbin, J. (1998), *Basic Qualitative Research*, Thousand Oaks, CA, Sage Publications.

Stroud, D. (1998), *Internet Strategies: A Corporate Guide to Exploiting the Internet*, Basingstoke, Macmillan.

Suarez, F. F., Cusumano, M. A. and Fine, C. H. (1995), 'An empirical Study of Flexibility in Manufacturing', *Sloan Management Review*, Fall, pp. 25-32.

Sutherland, J. (2001), 'Agile Can Scale: Inventing and Reinventing SCRUM in Five Companies', *Cutter IT Journal: The Journal of Information Technology Management*, Vol. 14, No. 12, pp. 5-11.

Swanson, E. B. (1976), 'The Dimensions of Maintenance', In: *Proceedings of the 2nd International Conference on Software Engineering*, San Francisco, California, USA, October 13-15. Los Alamitos, CA, IEEE Computer Society Press, pp. 492-497.

Swanson, E. B. (1999), 'IS "Maintainability": Should It Reduce the Maintenance Effort?', *The DATA BASE for advances in Information Systems*, Vol. 30, No. 1, pp. 65-76.

Swanson, E. B. and Beath, C. M. (1989), *Maintaining Information Systems in Organizations*, Chichester, John Wiley & Sons, Ltd.

Swanson, E. B. and Beath, C. M. (1990), 'Departmentalization in Software Development and Maintenance', *Communications of the ACM*, Vol. 33, No. 6, pp. 658-667.

Taber, C. and Fowler, M. (2000), 'An Iteration in the Life of an XP Project', *Cutter IT Journal: The Journal of Information Technology Management*, Vol. 13, No. 11, pp. 13-21.

Takang, A. A. and Grubb, P. A. (2003), *Software Maintenance: Concepts and Practice*, Second Edition, Singapore, World Scientific Publishing Co. Pte. Ltd.

Taylor, M. J., McWilliam, J., Sheehan, J. F. and Mulhaney, A. (2002), ‘Maintenance Issues in the Website Development Process’, *Journal of Software Maintenance and Evolution: Research and Practice*, Vol. 14, No. 2, pp. 109-122.

Taylor, M. J., Moynihan, E. P. and Wood-Harper, A. T. (1997), ‘Knowledge for Software Maintenance’, *Journal of Information Technology*, Vol. 12, No. 2, pp. 155-166.

The Economist (2008), ‘Where the cloud meets the ground: Data centres are quickly evolving into service factories’, *The Economist Newspaper Limited*, Switzerland, Ringier Print Zofingen, October 23rd.

The Economist (2010), ‘A different game: Information is transforming traditional businesses’, *The Economist Newspaper Limited*, Switzerland, Ringier Print Zofingen, February 25th.

Thomke, S. and Reinersten, D. (1998), ‘Agile Product Development: Managing Development Flexibility in Uncertain Environments’, *California Management Review*, Vol. 40, No. 1, pp. 8-30.

Tonella, P. and Ricca, F. (2004), ‘Statistical Testing of Web Applications’, *Journal of Software Maintenance and Evolution: Research and Practice*, Vol. 16, No. 1-2, pp.103-127.

Trapnell, F. (1995), ‘Again, that maintenance “iceberg”’, *Journal of Software Maintenance: Research and Practice*, Vol. 7, No. 5, pp. 381.

Truex, D. and Baskerville, R. (1998), 'Deep Structure or Emergence Theory: Contrasting Theoretical Foundations for Information Systems Development', *Information Systems Journal*, Vol. 8, No. 2, pp. 99-118.

Truex, D. P., Baskerville, R. and Klein, H. K. (1999), 'Growing Systems in an Emergent Organization', *Communications of the ACM*, Vol. 42, No. 8, pp. 117-123.

Truex, D., Baskerville, R. and Travis, J. (2000), 'Amethodical Systems Development: The Deferred Meaning of Systems Development Methods, Accounting', *Management and Information Technologies*, Vol. 10, pp. 53-79.

Tudor, D. J. and Tudor, I. J. (1997), *Systems Analysis and Designs: A Comparison of Structured*, Methods, Basingstoke, UK, Macmillan Press Ltd.

Turau, V. (1998), 'What Practices are being Adopted on the Web?', *IEEE Computer*, Vol. 31, No. 5, pp.106-108.

Turver, R. J. and Munro, M. (1994), 'Early Impact Analysis Technique for Software Maintenance', *Journal of Software Maintenance*, Vol. 6, No. 1, pp. 35-52.

Upton, D. M. (1994), 'The Management of Manufacturing Flexibility', *California Management Review*, Vol. 36, No 2, pp. 72-89.

Vander Blonk, H. (2003), 'Writing case Studies in Information Systems Research', *Journal of Information Technology*, Vol. 8, pp. 45-52.

Veld, J. in't (1978), *Analyze van Organisatieproblemen*, Elsevier.

Veldwijk, R. J. (1993), *Introspective Systems Design: Exploring the Self-Referentail Capabilities of the Relational Model*, Unpublished Ph.D. Thesis, Saxon Hogeschool, Enschede, The Netherlands.

Veldwijk, R. J., Boogaard, M. and Spoor, E. R. K. (1994), 'Assessing the Software Crisis: Why Information Systems are beyond control', *Information Sciences*, Vol. 81, No. 1-2, pp. 103-116.

Vessey, I. and Weber, R. (1983), 'Some Factors affecting Program Repair Maintenance: An Empirical Study', *Communications of the ACM*, Vol. 26, No. 2, pp. 128-134.

Victor, K. M. (1995), *Software System Flexibility: Conceptualization and Measurement*, Unpublished PhD. Thesis, Austin, The University of Texas.

Vidgen, R. (2002), 'What's so Different about Developing Web-Based Information Systems?', In: *Proceedings of the 10th European Conference on Information Systems, Information Systems and the Future of the Digital Economy (ECIS)*, Gdansk, Poland, June 6-8. ECIS, pp. 262-271.

Walsham, G. (1993), *Interpreting Information Systems in Organisations*, London, John Wiley & Sons, Inc.

Warren, P., Boldyreff, C. and Munro, M. (1999), 'The Evolution of Websites', In: *Proceedings of the 7th International Workshop on Program Comprehension*, Pittsburgh, PA, May 05-07. Los Alamitos, CA, IEEE Computer Society, pp.178-185.

Watson, H. and Wood-Harper, T. (1996), 'Deconstruction contexts in interpreting methodology', *Journal of Information Technology*, Vol.11, pp. 59-70.

Weick, K. E. (1984), ‘Theoretical Assumptions and Research Methodology’, In: McFarlan, F. W. (eds.), *The information systems research Challenge*, Boston, MA, Harvard Business School Press, pp. 111-132.

Weiser, M. (1982), ‘Programmers use Slices when Debugging’, *Communications of the ACM*, Vol. 25, No. 7, pp. 446-452.

White. B. E. (2005), ‘A Complementary Approach to Enterprise Systems Engineering’, In: *Proceedings of the 8th Annual Systems Engineering Conference, National Defence Industrial Association (NDIA)*, San Diego, CA, October 24-27. Ft. Belvoir, VA, Defence Technical Information Centre (DTIC).

Xu, F., Uszkoreit, H. and Li, H. (2007), ‘A Seed-driven Bottom-up Machine Learning Framework for Extracting Relations of Various Complexity’, In: *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL)*, Prague, Czech Republic, June 23-30. Stroudsburg, PA, USA, Association for Computational Linguistics, pp. 584–591.

Yau, S. S. and Collofello, J. S. (1980), ‘Some Stability Measures for Software Maintenance’, *IEEE Transactions on Software Engineering*, Vol. 6, No. 6, pp. 545-552.

Yau, S. S., Collofello, J. S. and McGregor, T. M. (1978), ‘Ripple Effect Analysis of Software Maintenance’, In: *Proceedings of the IEEE Computer Society's 2nd International Conference on Computer Software and Applications (COMPSAC)*, Chicago, USA, Los Alamitos, CA, IEEE Computer Society Press, pp. 60-65.

Yeh, R. T. (1990), ‘Software Requirements: A Management Perspective’, In: Thayer, R. H. and Dorfman, M. (eds.), *System and Software Requirements Engineering*, Los Alamitos: IEEE Computer Society Press.

Yin, R. K. (2003), *Case Study Research: Design and Methods*, Third Edition, Thousand Oaks, CA, Sage Publications, Inc.

Yourdon, E. (1997), *Rise and Resurrection of the American Programmer*, Upper Saddle River, New Jersey, Prentice-Hall, Inc.

Yu, A. G. (2009), ‘Software Crisis, What Software Crisis?’, In: *Proceedings of the International Conference on Management and Service Science (MASS)*, Beijing, China, September 20-22. Washington, DC, IEEE Computer Society, pp.1-4.

Zafiris, P. A., Georgantis, N. P., Kalamaras, G. E., Christodoulou, S. P. and Papaatheodorou, T. S. (2001), ‘A Practitioner's Approach to Evolving and Remodeling Large-Scale WWW Sites’, In: *Proceedings of the 34th Annual Hawaii International Conference on System Sciences (HICSS)*, Hawaii Jan 3-6. Los Alamitos, CA, IEEE Computer Society, pp. 10.

Appendices

Appendix A: Flexibility Attributes

Land, F. (1982). "Adapting to Changing User Requirements." Information & Management **5**: 59-75.

Feature/ Characteristic:	Stage:	Description:
Life span & Extent of flexibility		This should be considered for the system.
Modelling		Present (and maybe future too) set of user requirements.
Futures Analysis		Predicting the changes that could occur over the expected life of the system and which could have an impact on the system in question.
	Stage 1	Discover the kinds of change that may have an impact on the system under consideration.
	Stage 2	Design team to assess the kind of future that the system will have to face.
	Stage 3	Design team to test system to identify features that are sensitive to any change in requirements. (Logic or traffic).
	Stage 4	Design team to consider output from stages 2 & 3, therefore planning the life of a system. Also to see which scenarios have to be coped with.

Knoll, K. and S. L. Jarvenpaa (1994). Information Technology Alignment or "Fit" in highly turbulent environments: The concept of Flexibility. Proceedings of the 1994 computer research on reinventing IS: managing information technology in changing organizations (ACM), Alexandria, VA USA.

- Flexibility types: Organisation flexibility; Business Process flexibility.
- Evans (1991) terms: adaptability, agility, corrigibility, elasticity, hedging, liquidity, malleability, plasticity, resilience, robustness, versatility.

Discipline:	Dimension:	Description:
Information Technology (platform) (Keen, 1991)	Reach	Represents the locations to which a platform can link
	Range	Represents the connectivity of information across systems.
Manufacturing (Sethi and Sethi, (1990) **Time** & **Range characteristics** for all Time refers to the speed at which the change can be made. Range refers to the amount or number of changes that can be made.	Mix	The extent of product variety
	Changeover	Major design changes, which can be accommodated
	Modification	Minor design changes, which can be accommodated
	Volume	Amount of change in aggregate production levels
	Rerouting	The extent to which re-routing can occur, and the variety of parts for which it occurs
	Material	Handling unexpected variations in inputs
	Flexibility responsiveness	Changes to types, ranges, and times of the six dimensions.
Strategic Management	Yielding to pressure	**Intention** and **Timing** are important **characteristics**.
	Capacity for new situations	
	Susceptibility to modification	

Flexibility Categories	Flexibility in Functionality (input)	Flexibility in Use (output)	Flexibility in Modification (process)
Flex Domain	Input quality and quantity	Outcome	Process
Source of Pressure	Changes in inputs due to malfunctions or environmental variability	Changes in goals	Feedback that adjustments are necessary
Method	Yield to pressure temporarily or permanently	Participate in new relationships for expanded capabilities	Interpret feedback and apply effort to modification
Result	Operate well in many different environments	Ready and able to precipitate a new state of affairs (i.e. new use)	Capable of rapid adjustments with minimal effort

Type:	Dimensions:	Definition:
Flexibility in Functionality	Robustness	Technology with the necessary cohesion to withstand a variety of pressures.
	Scalability	Technology that can be increased or decreased by small increments to accommodate load levelling.
	Slack	Technology whose excess capacity is able to withstand increasing loads without any modification.
Flexibility in modification	Commensurability	Size of change effort is proportional to benefits of change
	Feedback Sensitive	Technology that can be adjusted based on feedback.
	Goal Adjusting	Technology that can change internal goals based on external or internal feedback.
	Just-in-Time Adjusting	Technology that can be adjusted close to the time of its use.
	Polyadjustable	Technology that can be adjusted by a variety of mechanisms.
	Self-Adjusting	Technology that has all required adjustment capabilities within its boundaries.
	Trialability	Ease of demonstrating a technology's features through prototyping.

Flexibility in Use	Concurrency	Technology that appears to give simultaneous access because it can manage two or more rapidly fluctuating relationships.
	Connectivity	Technology that has similar processes and structural features at its boundary allowing exchanges of information between parts.
	Modularity	Self-contained technology that fits with other basic parts for more complex subsystems. The technology can be reused for the same task or a different task.
	Multiple Forms	Technology capable of producing essentially the same information using various representation systems.
	Responsiveness	Technology that provides feedback of the kind and speed that is acceptable to the user.
	Reusability	Technology that can repeat its functions for use in similar task.
	Spatial Decoupling	Technology whose use is easy to infer based on its similarity to other structures and processes in the task domain.
	Temporal Decoupling	Technology that can exchange information without regard to start or stop times, or rate of functioning.
	Transparency	Technology whose use is easy to infer based on its similarity to other structures and processes in the task domain.
	Versatility	Technology that can incorporate other technologies for expanded capabilities.

Above two tables taken from paper.

Boogaard, M. and E. R. K. Spoor (1993). "Information Systems Flexibility: A Conceptual Framework." Serie research memoranda: 1993-1959.

- Rochester (1989) flexibility from three points of view: Business view, user's view, and IS department view.
- Stimulus-response paradigm.

The below table shows a conceptual framework of flexibility where past and present developments in this field can be classified and compared as far as their contribution to flexibility is concerned.

Question:	Feature:	Description:
What is the origin?	Agent	This feature indicates whether the mobilisation of the potential arises from within the IS itself (***internal***) or from an outside source (***external***).
Where is it applied?	Process	The process feature refers to the application area of the flexibility potential. ***Use, development and operations.***
What does it affect?	Recipient	Recipient feature involves the implications of an alteration on the elements of the IS. Sagasti (1970) – Two aspects that can be affected by change i.e. structure and function. ***Architecture, database structure and function.***
How is it applied?	Method	Method feature emphasis the procedure that is followed when the flexibility potential is mobilised. ***Parameters, library and development***.
When is it applied?	Inception	Indicates the moment of mobilisation of the flexibility potential. Time of disturbance (stimulus), time of counter measure (response) and time of perception. ***Reactive or Anticipative.***

Kanellis, P., R. J. Paul, et al. (1996). An Archetype for Reseaching Information Systems Flexibility. Proceeding of The Fifth International Conference on Information Systems Development, Gdansk - Poland.

Characteristics of change:	Responses to change:
Frequency	**Insulate** themselves from change
Amplitude	**Resist** the change/ or attempt to remove the original source of the change
Predictability	**Accept** the change and adapt accordingly
	Ignore the change

Fitzgerald, G. and F. A. Siddiqui (2002). "Business Process Reengineering and Flexibility: A Case for Unification." The International Journal of Flexible Manufacturing Systems **14**: 73-86.

- Flexibility in management, operations management, and information systems.
- Types of flexibility: Strategic flexibility; Business flexibility.

Discipline:	Flexibility Type:	Description:
Manufacturing (Browne, Dubois, Sethi and Stecke (1984))	Machine flexibility	Is the ease of making the machine changes required to produce a given set of part types.
	Process flexibility	Is the ability to produce a given set of part types, each possibly using different materials, in different ways.
	Product flexibility	Is the ability to changeover to produce a new product economically and quickly.
	Routing flexibility	Is the ability to handle breakdowns and continue to produce the given set of parts. Also can be applied to performance.
	Volume flexibility	Is the ability of a manufacturing system to produce products economically at different production volumes
	Expansion flexibility	Is the capacity of a manufacturing system to expand easily and modularly
	Operational flexibility	Is the ability to interchange the ordering of several operations for each product type
	Production flexibility	Is the universe of product types that a manufacturing system can produce

Mandelbaum and Brill (1989)	Action flexibility	Defined as the capacity for taking new action to meet new circumstances
	State flexibility	If the system has the capacity to continue functioning effectively despite changes in circumstances in the short term.
Architecture	Flexibility	Is the variety of possible uses or rearrangements that can be made without major physical change
	Adaptability	Is the process of enabling a facility to be used for a purpose or use other than the original intention.
IS-IT (Rochester (1989)	Elements of flexibility (below)	
	Business View	The capability to build or adapt IS in response to corporate changes
	User view	Essentially means intuitive and adaptable user interfaces.
	IS development view	Which means portability, connectivity and maintainability.
Organisational Strategy and design	Internal	Management's capability to adapt to the demands of the environment
	External	Management's capability to influence the environment so that the firm becomes less vulnerable to environmental changes.
	Steady-state	Static procedures to optimise the firm's performance when the nature and levels of throughput remain relatively stable over time. Efficiency e.g. overheads.
	Operational	Routine capabilities that are based on current structure or goals of the organisation. Volume and mix of activities. Internal and external distinction can be applied here.
	Structural	Consists of managerial capabilities for adapting the organisation structure and its decision and communication processes, to suit changing conditions in an evolutionary way. Internal and external can also be applied here.
	Strategic	Consists of managerial capabilities related to the goals of the organisation or the environment. Internal and external applied here.

Nelson, K. M. and M. Ghods (1998). "Measuring technology flexibility." European Journal of Information Systems **7**(4): 232-240.

- Flexibility a key characteristic in both **technology** and **business processes**.

Flexibility type:	Determinants:	Description:
Structural flexibility		Is the capability of the design and organisation of a technology to be successfully adapted to business process changes.
	Modularity	The degree of formal design separation within a technology
	Change Acceptance	The degree to which a technology contains built-in capacity for change
	Consistency	The degree to which data and components are integrated consistently across a technology
Process flexibility		Is the ability of people to make changes to the technology using management processes that support business process changes.
	Rate of Response	The degree to which changes can be made to a technology in a timely manner.
	Expertise	The degree to which up-to-date knowledge about the operations and maintenance of a technology exists and is communicated.
	Co-ordination of Action	The degree to which the technology maintenance and user organisations operate according to the requirements of each other and total organisation.

Golden, W. and P. Powell (2000). "Towards a definition of flexibility: in search of the Holy Grail?" Omega The International Journal of Management Science **28**(4): 373-384.

- Types of flexibility: Organizational flexibility; Strategic flexibility.

Dimension:	Description:	Measure:
Temporal	How long it takes an organization to adapt Flex → Operational (Short term); Tactical (medium term); Strategic (long term).	Degree of fit within time limit → **Efficiency** Length of time needed → **Responsiveness**
Range	The number of options that an organisation has open to it for change that was foreseen and the number of options that it has available to react to unforeseen change. Type I: flex that relates to the concept of risk and involves planning for foreseeable events. Type II: flex that relates to uncertainty (unforeseeable changes in the environment)	Foreseen → **Versatility** Unforeseen → **Robustness**
Intention	Whether the organisation is being proactive or reactive. Offensive: take control of change in the environment. Defensive: organisations react to changes after they have occurred (minimise impact).	Intention is situation specific.

Focus	Flex gained internally to the organisation or by managing external relationships with trading partners. Flexibility is created. ***Strategic flex***: internal (manufacturing, employee, organisational structure) and external (suppliers, alliances, multinational operations (***network*** forms of organisations)).	Focus is situation specific.

Fitzgerald, G. (1990). "Achieving flexible information systems: the case for improved analysis." Journal of Information Technology **5**: 5-11.

Change categories:	System components changed:	Flexibility Analysis:
Environmental	Hardware	FA team
Technical	Software	Short, medium or long term
Organizational	Interfaces	Identify potential change
	Data	FA fed into design
	Processing	

Veryard, R. (June 2001). Business Flexibility - A Framework for Understanding, Planning and Achieving Flexibility, CBDi Forum.

- Similar terms to flexibility: Responsiveness, Adaptability; Malleability, Agility; Robustness; Generality.
- **Context** and **scope** important for flexibility.
- Context used: **Architecture → *Layers (Brand). Coupling.***

Flexibility focus (levels):	**Description:**	**Flexibility:**	**Inflexibility:**
Technical	Allows an IS to cope with, and indeed to leverage, ongoing changes in the technology base.	Uncertainty about the future.	Legacy systems; Problematic systems; Systems Integration; Test pack; Other people; Accounting systems.
Strategic	Allows a system to keep pace with frequent changes to the configuration of the business.	Ability to accommodate future changes.	
Operational	Allows a system to cope with unforeseen circumstances in the day-to-date operations of the business.	Uncertainty about the present.	

Nelson, K. M. and J. G. Cooprider (2001). <u>The relationship of software system flexibility to software system and team performance</u>. Twenty-Second International Conference on Information Systems, New Orleans, Louisiana, U.S.A.

- Types of flexibility: Software systems flexibility; Business process flexibility

Dimension:	**Performance measures:**
Structural	**System performance** as a measure of product performance**.**
Process	**Team performance** as a measure of the process of maintaining the software system.

Avison, D. E., P. L. Powell, et al. (1995). "Addressing the need for flexibility in information systems." Journal of Management Systems **7**(2): 43-60.

Maneuvers that provide Strategic flexibility:	**Description:**	**Future requirements techniques:**	**Methods that address flexibility:**
Pre-emptive	Creating options, inflicting surprise or seizing initiatives	**Future Analysis** (Land, 1987) // **Flexibility Analysis** Changes and the future.	Ethics, Prototyping, Multiview
Protective	Insuring against losses, hedging, or creating buffers against adverse conditions	**Risk Analysis** (risk engineering) Helps to manage uncertainty and its effects. SCERT (Synergistic Contingency Evaluation and Review technique) →Method. Could be modified to make more applicable to computer-based systems.	
Corrective	The ability to recover from adverse situations and learn from mistakes	**Lateral Thinking**	
Exploitive	Capitalising on opportunities and consolidating advantages		

Byrd, T. A. and D. E. Turner (2000). "Measuring the Flexibility of Information Technology Infrastructure: Exploratory Analysis of a Construct." Journal of Management Information Systems **17**(1): 167-208.

- **Technical** and **Human** IT infrastructures.

Dimensions of It infrastructure flexibility.
First 4 of below → technical IT infrastructure
Last 4 of below → human IT infrastructure

Dimensions:	Description:
IT connectivity	The ability of any technology component to attach to any of the other components inside and outside the organisational environment.
Application functionality	Relates to the ability to add, modify, and remove the modules of software applications with little or no widespread effect on the applications collectively.
IT compatibility	Is the ability to share any type of information across any technology component.
Data transparency	Defined as the free retrieval and flow of data between authorised personnel in an organisation or between organisations regardless of location.
Technology management	Pertains to the organisations ability to deploy IT in the most effective possible manner in support of the business strategies.
Business knowledge	Relates to the ability of IT personnel to understand the business processes they are to support and to apply the appropriate technical solution to a given business problem.
Management knowledge	Refer to the importance of IT personnel having skills and knowledge to assume roles outside there area of training or original competencies.
Technical skills	A set of measures of technical capabilities.

The dimensions were narrowed down to the following: (Two for technical and one for human infrastructure)

Dimension:	Description:
Integration	Transparent access IT connectivity & IT compatibility
Modularity	Application functionality & database transparency
IT personnel flexibility	Technology management, Business knowledge, Management knowledge & Technical skills

Allen, B. R. and A. C. Boynton (1991). "Information Architecture: In Search of Efficient Flexibility." MIS Quarterly: 435-445.

Architecture Strategies for incorporating Flexibility.
The below strategies are limited in their sole use and what is required is an optimisation between the two architects.

Strategies	Description:
Low Road	• IS technology and its management are dispersed widely throughout the firm • Empowerment? – maybe • Distributed systems/networks • IS becomes the responsibility of every operating manager from subsidiaries down to plants • In-order for it to work the following units must be addressed: - **Inter-linked communications** - **Full access to information** - **Data exchange conventions**
High Road	• Core IS activities of the business are centralised • Senior IS executive role expanded • IS infrastructure built around the central precepts of corporate-wide networks

Nelson, K. M., H. J. Nelson, et al. (1997). Technology Flexibility: Conceptualization, Validation, and Measurement. Proceedings of the Hawaii international conference on system sciences, Hawaii, USA : IEEE Institute of Electrical and Electronics.

Technology Flexibility (TF) – The ability to adapt to both incremental and revolutionary change in the business or business process with minimal penalty to current time, effort, cost, or performance.

Flexibility types: Technology Flexibility; Business Process Flexibility; Strategic organisational flexibility

Discipline:	Flexibility Dimension:	Description
Information Technology (Knoll and Jarvenpaa)	Flexibility in Use	Incremental as well as revolutionary change capacity
	Flexibility in Functionality	Incremental change
	Flexibility in modification	Incremental change
Behavioural Psychology (Scott)	Degree of response variability	**Response variability** is defined by the degree of diversity in reactions shown by a particular person under normal, everyday conditions, which is the type of conditions under which **incremental change** occurs.
	Responsiveness	To environmental pressures to change. Capacity to accommodate **dramatic change**.
Manufacturing Flexibility (Garud and Kotha)	Social (Systems approach)	Job design, management organisation, work-team structure, selection and training, compensation and appraisal.
	Technical (Systems approach)	Manufacturing equipment
Organisational (Evans)	Ex ante	Anticipates change before it happens **Agility & Versatility**: **Offensive** characteristics that provide a repertoire for dealing with novel or unexpected situations.

		Robustness & Hedging: **Defensive** characteristics that seek to deflect or avoid the unexpected.
	Ex post	Incremental in nature **Liquidity & Elasticity**: **Offensive** that allows continual change with the environment. **Corrigibility & Resilience**: **Defensive** that aids a system in recuperating or returning to functionality after a change.

Dimension:	Definition:	Determinants:	Definition:
Structural Flexibility	Is the capability of the design and organisation of a technology to be successfully adapted to business process changes	Modularity Change Acceptance Consistency	The degree of formal design separation within a technology application. The degree to which technology contains built-in capacity for change. The degree to which data and components are integrated consistently across a technology.
Process Flexibility	Is the ability of people to make changes to the technology using management processes that support business process changes	Rate of Response Expertise	The degree to which changes can be made to a technology in a timely manner. The degree to which up-to-date knowledge about the operation and

			maintenance of a technology exists and is committed.
		Co-ordination of Action	The degree to which the technology maintenance and user organisations operate according to the requirements of each other and the total organisation.

Appendix B1-B5: Examples of documents collected from the case study

Appendix B1: Written reports

T[redacted] Confirmation (SM) 29-[redacted]

	General Stuff	Fixed?
1.	We agreed that when all the textual changes had been completed in the site, I would find any cases where the bottom line of text has been 'cut' horizontally and looks bad. There may well be none … but this is just a reminder that I have yet to do this.	Sarah to do.
2.	This note is just to remind Sarah that acknowledgement emails are yet to be tested. Sanjay will advise when this will be possible.	Sarah to do
3.	If you don't fill out a submit form correctly the 'please ensure all fields with an * are filled in' message is displayed. This is OK, but the data that you already keyed into the form should NOT be cleared. At the moment, you must key it all again which is really annoying. This would be enough to put me off submitting the info request form.	**Costed.** **Client confirmed that they want it done.**
4.	This is what was agreed for the new links on the l.h.s. of each page : a. Links on the l.h.s. are all 'downwards' until you reach a page that has no further 'downwards' options. These pages should have 'sideways' links instead (the only exception to this is the various products/applications pages which all contain many l.h.s. links already). b. All these extra 'downwards' and 'sideways' links added to the l.h.s. will be headed with the words 'other options include:' in the same small blue typeface. This will avoid these new links being confused with pre-existing links that open a pop-up page. However, we need to take care not to put the 'other options include:' heading above the various pre-existing links to pop-up pages (e.g. the 'Implementation Plan' link on the Services/Consultancy page) in this region. c. Where there are pre-existing links, the new links will be added underneath them, with a small gap separating them.	Proposal dropped. There will be a dividing gap on these pages.

	Product/Platforms	Fixed?
1.	In the third level nav, the options listed should be changed as follows : Client Server (was Server) RDBMS (no change needed) ASP/Hosted (was Work Force ASP) NB: this will also affect the new l.h.s. links at both second and third level nav.	**COSTED** **Client has confirmed that they want changes.**

Appendix B2: Meeting minutes (excerpt)

Software Changes outstanding at 14-[redacted]

NB: latest comments are in bold/red

	Description
1.	Adding PDFs via the Admin Area 29/04: Problem with adding PDFs via the admin area. Server seems to be adding an underscore character to spaces in filenames – with the result that the PDF then isn't found when the associated link is clicked in the site (and the home page is displayed instead). Work-around is to name PDFs without spaces, or with underscores in place of spaces. Bruno to talk to SOL. **14/05 : need an update.**
2.	Viewing the site using Mozilla 29/04: Using Mozilla 0.97 most links in the client zone don't work. Non-urgent. **14/05: need an update.**
3.	Client Zone>Documentation>Helpcards **14/05: Long-standing problem with opening helpcard PDFs. I think this boils down to the strange fact that the helpcard PDFs take MUCH longer to open than the manuals PDFs. This is only strange because the manuals PDFs are about 2-3 times larger documents. Hmmm. Makes no sense to me. I will try to re-build the PDFs using Distiller and see if I can fix the problem this way. I'd just like to leave this 'issue' open for the time being.f**
	NEW STUFF added 14/[redacted] :
4.	Client Zone>Open Forum **14/05: Go to the forums. There are 2. There is a "jump to forum" dropdown list which doesn't always work for me. In both cases it shows "Test Forum" as its current contents.** **If I am in "Test Forum" then I can choose "Software Forum" and it works.** **If I am in "Software Forum" I can't use it to navigate to the "Test Forum"** **I speculate that the reason is that you have to change the contents of the box (ie change the current selection) in order for it to think it has anything to do. In this case the true error is that whatever forum you are in (one of these 2 or any future additional one) the page must always be drawn with the current forum selected in the dropdown list so that any change will cause navigation to a different forum correctly.**
5.	Contact Us and Client Zone>Contact US – Map **14/05: There are a couple of spelling mistakes on the 'small' map.** **1). Change 'Hatsfield Road' to 'Hatfield Road'.** **2). Change 'Upper Marlboro' to 'Upper Marlborough'.**

Appendix B3: Change request email

On 23/[redacted] 10:58 am, "[redacted], Lorraine" <LMC@[redacted].com> wrote:

Chris

Further to our telephone conversation, I would like to confirm the following:

The dropdown menus have had a small amount of vertical space added to stop the text crashing together. This now works on normal and 125% font sizes. The sub-menus now draw to the right, if the right hand side of the page is reached and draw up if the wave is reached. Both you and Jonathan explained to me that reducing the font size to 75% so that our unusually large fonts would fit within the menu would mean that on the majority of settings the text would appear far too small. I agree that obviously this is not desirable.

Upon further discussion with you, we have agreed that a compromise solution may be to increase the height and width of the drop down menus by a few pixels so that there is some extra space between the lines when viewing with a small (standard) text setting. This will mean that when using a large text setting (125%) there will not be an overlap. I attach a screenshot of the site on the small text setting, as it appears to us. The line spacing on this screenshot is not excessive as far as we can see, so adding so extra space would not necessarily pose a huge problem for us. This was what we meant at the meeting on Monday.

You then said that this would mean that the menus would be too long and go down beyond the Flash movie at the bottom. It had already been agreed that the menus would be set up so that they would automatically go back up if they were to come into contact with the Flash movie. This is a necessary feature in any case to accommodate all settings. You agreed to do this. (Similarly, they should automatically go to the left if they come into contact with the edge of the window on the right. This was also agreed previously.)

We have added Netscape 7 to the list for Macs and PCs. I was fully aware of Netscape 6 but had omitted it from the list by mistake. When undertaking cross browser testing an important point has to be borne in mind. The site you are testing has to fully work, functionality and layout, in the most common browsers. When testing in browsers such as Netscape 4.6 or IE4, which have very small percentages of the market and are very old, the desired result is that the site delivers the appropriate content and that major elements of functionality, like drop-down menus, function. The sites need to "degrade gracefully" in non-standard browsers.

During the meeting, I was given a cross browser testing Matrix and invited to enquire whether further items could be added, if I considered that they might be appropriate.

I requested that when testing the site on PCs, Netscape version 7 and any later version between 4 and 7 could be added to this as naturally it needs to be compatible with the latest versions. You replied that as far as you knew that the latest version was 4.6 and then version 7 must have been very new. You also conceded that it should have been your job to keep up to date with this. I have since taken another visit to the Netscape site which I think may prove useful for you at http://wp.netscape.com/download/archive.html. As you will be able to see since version 4.6, the following versions have been released: 4.7 and 4.8 6.1 6.2. I made the point that although the majority of users may be using a less current version, the new versions will become more popular in the near future and that our website must be built to make allowances for future trends as far as is possible.

http://[redacted].about.com/cs/statistics/
http://www.[redacted].com/browsers/browsers_stats.asp

Further to our conversation, I would also be interested to know the source of your statistics on internet browser usage. It is important to bear in mind that although based in London, we have a large number of clients in Europe, the US and Asia, so this must be taken into account.
Browser use what is called a rendering engine. This is the part of the programme that reads the page code then place "draws" the page in the browser window. Netscape 4.6, for example, has an appalling rendering engine and it takes a long time for it to draw anything to the screen. The browsers that you are referring to have a quirk in their rendering engine. The page itself has instructions to the place the contents at the very top left, x=0 & y=0. The engine in this case is drawing the page with the default margin x=5 & y =5 and then deciding to interpret the code go 0,0. This is not a bug but a quirk in the browser and any site with a 0,0 margin will do this.

I have noticed that on several of the firm's PCs, that there is a white space which appears at the top for a second or so when the page first downloads and then disappears. It has been noted by me and Neil
[redacted] in IT that this has an effect of making the page jolt upwards when it first
> downloads. The screenshot of this is included in the document attached as
> this may be helpful. You replied that this type of thing happens with a
> number of websites. I asked whether you could send me some URL examples of
> this. You then said that you would prefer to examine this first and then
> get back to me on my query. This would be much appreciated by both John
> [redacted] and me.
>
> (I would like to reiterate, in case Jonathan also misunderstood, that Neil
> was not present in our meeting on Monday. It was Jonathan [redacted], our IT
> Director, and Lucky [redacted] ,our web developer, who were present. I just
> want to make this clear as you made the point that Neil was quiet during the
> meeting - this was because he was not there :)
>
> I hope that everything is clear. Please feel free to give me a call to
> discuss any points and I apologise if any information in this email is not
> correct. I am no technical expert and would be happy to stand corrected. I
> will continue to flag any potential problems when I become aware of them so
> that we can move forward with the project in the most efficient manner
> possible. Many of our partners are extremely vigilant and will undoubtedly
> pick up on many of the items I am flagging. It is therefore obviously
> preferable for both you and us that these are resolved at this stage to
> avoid potentially far more time-consuming changes taking place once the site
> has been built and launched.
>
> I appreciate that we are still viewing work in progress. It would be helpful
> therefore if you would clarify at what stage I should be flagging
> inconsistencies in design etc.
>
> I thank Jonathan for the email he sent last night. We will take a look at
> this and get back to you with our comments on it shortly.
>

> I would like to thank you and Jonathan for all your efforts in producing the
> site to our satisfaction. John and I look forward to receiving your
> response.
>
> Kind regards
>
> Lorraine
>
> <<Docs_2084471_1.DOC>>
>
>
> *******************
> Please read these warnings and requirements:
> This e-mail transmission is strictly confidential and intended solely for the
> addressee. It may contain privileged and confidential information and if you
> are not the intended recipient, you must not copy, distribute or take any
> action in reliance upon it. If you have received this e-mail in error, please
> notify the sender or Administrator@[redacted].com and delete the e-mail
> transmission immediately.
> Viruses: Although we have taken steps to ensure that this e-mail and
> attachments are free from any virus, we advise that in keeping with good
> computing practice the recipient should ensure they are actually virus free.
> Security Warning: Please note that this e-mail has been created in the
> knowledge that internet e-mail is not a 100% secure communications medium. We
> advise that you understand this lack of security and take any necessary
> measures when e-mailing us.
> [redacted] reserve the right to read any e-mail or attachment
> entering or leaving its systems from any source without prior notice.
> A list of partners is available at www.[redacted].com
>
> [redacted], 35 [redacted] Street, London EC[redacted] 2[redacted]
> Tel: +44(0)207 [redacted] Fax: +44(0)207 [redacted] CDE: 823
>
> [redacted] is regulated by the [redacted]. Equity Incentives
> Limited, an incorporated legal practice wholly owned by [redacted], is regulated by the [redacted] and authorised by the Financial
> Services Authority. Equity Incentives Limited trades under its own name and as
> [redacted]
> ***
>
>

Dear Lorraine

Thanks for your helpful notes – and in response to the issues you raised:

- Drop down menus – all issues completed as stated in our email last night - as asked could you please tell us what the new sized menu boxes look like on your system

- Cross browser testing – new matrix attached to incorporate Netscape 6 & 7.

By quoting 4”x” on the matrix means we cover all versions of a particular system.
Quite agree that site should be compliant with future trends as much as possible.

- Browser usage stats are in fact “global” and ours came from 3 different sources – I am afraid nobody is in a position to guarantee their accuracy but their similarity would suggest that they are as near the mark as possible.

Please view the following for verification --------------

- The “jolting” on the screen is--------------

OUTSTANDING ISSUES FROM OUR EMAIL LAST NIGHT:

- L/H menus - the latest testsite attached shows a, static, sample L/H menu – could we please have your approval and confirmation that it works with your bigger fonts before we build all the L/H menus.

- Screen changed slightly so that wave is fully visible on 860.

- “White corner” on menu no longer an issue as the menu boxes move left rather than overlap the window to the right.

DESIGN APPROVAL

Do please come back to us soonest on all the above and on the points raised in my email last night. I would like to be able to sign off design issues by the end of today – so very keen to have all your comments to the latest testsite (attached). Once the design is approved we will then be able to start testing across the various platforms – a phase which is likely to throw up a few different development issues....

Would it be wise for you to come over tomorrow and go through the site in person?

Best regards

J[redacted]

Appendix B4: Change request form (CRF)

PROJECT	
APPLICATION NAME : PROJECT MANAGER : LICENSE REF. :	Firewall Infrastructure Mihoubi [redacted] -
SERVER :	
PLATEFORM : OPERATING SYSTEM :	CheckPoint SecurePlatform R62
BUSINESS JUSTIFICATION :	
BUSINESS AREA : DEPARTMENT :	IT Network and IT Security

EXPLANATION

After all [redacted] issue has been solved, we need to put back the [redacted] boxes into the [redacted] environment, to see if the behaviour is now corrected.

The plan to do this is :
- Configure the hosts files on the 2 [redacted] LIVE to direct like bellow :
  172.20.3.21 gva01-dmz-ea01
  172.20.3.22 gva01-dmz-ea01

Then reboot the two [redacted] (no service impact since the two boxes are not restarted at the same time.

The plan to test is to generate some flows through [redacted] (including login steps) and validate by analysing the firewall logs that the traffic is well passing through [redacted].

In case of problem, the roll back procedure consists to configure back the hosts files to point the [redacted] servername to the real server (instead of pointing to the [redacted] Web Security Appliance).

Additionnaly to the standard criteria for change requests in G[redacted], this Change must also be approved by the project team : Carl [redacted], Mei [redacted] and James [redacted].

NAME : Mihoubi [redacted] SIGNATURE :

FOLLOWING AREAS HAVE BEEN PROPERLY ADDRESSED (PLEASE FILL WITH Y, N OR N/A)

TESTING	: Yes	DOCUMENTATION UPDATE:	Yes
BACK-UP & MONITORING	: Yes	BCP UPDATE:	
IMPACT & RISK ASSESSMENT	: Yes	RIPPLE:	Yes (See doc no. 427)

ISM or SAM
NAME & SIGNATURE

IT Manager
NAME & SIGNATURE

Appendix B5: Issue Tracker

W fo : Testing Model — **7th December**
Version 3.1

Issue Hunter

No.	Issue Description	Changed	Tested	Verified
1	General: DO NOT REMOVE SOFTWARE TOUR	Y	Y	
1.1	All pop-ups should have a heading at the top of the viewable space.			
1.2	Some dynamic pop-up text is not registering paragraph breaks.			
1.3	Complaint that you can't move from box to box on forms with tab key.	?I can	Client	
1.4	Remove text limitations on text fields to allow user to scroll horiziontally through text.			
1.5	Need a generic form for submission confirmation screens.			
1.6	All links on rhs should be orange with underline.			
1.7	Filesizes: KB should be reported as 'k'. MB should be reported as 'Mb'.			
1.8	Need to test acknowledgement mails. Please divert to test@bestandco.com			
1.9	All acknowledgement mails should include date and time of the submission.			
2	Home Page: If you resize your browser window so that rhs is tight to picture... The	Y	Y	
	'Why workforce' box gets cut off.	Y	Y	
2.1	Leaf image for Client Zone should refresh to image box.	Y		
2.2	Navigation. The second level nav for News doesn't go away when you move from	Y	Y	
	News to Client Zone. They suggest a 2nd level Nav option of 'Login'.	Y	Client	
2.3	The products secondary nav. Is displayed in the top left behind the Workforce logo.	Y		
3	Why Workforce?: Page heading is wrong. Should say 'Why do people buy Workforce?'	Y	Y	
4	Contact Us: Address lines in wrong order.	Y	Y	
4.1	Helpdesk phone and email should not be on this version, only on the one accessed via	Y	Y	
	the Client Zone	Y	Y	
4.2	Largescale Map. Some changes to map needed:	Y	Y	
	Show Kings X Thameslink not St. Pancras as London Station.	Y	Y	
	The M1 should be shown.	Y	Y	
	A1(M) and M1 come inside the M25 closer to Central London	Y	Y	
	Show Junction numbers from M1, A1(M) and M25.	Y	Y	
	Change scale so that more of St. Albans area at the top left can be seen.	Y	Y	
5	Directions: Links on lhs of page should say 'directions from M25', 'directions from M1' and	Y	Y	
	'directions from A1(M)'.	Y	Y	
5.1	The linked windows should be similarly entitled.	Y	Y	
6	Software Tour: Move comments about mandatory fields into previous paragraph to	Y	Y	
	save space.	Y	Y	
6.1	Email field should not be mandatory.	Y	Y	
6.2	If fields in form need to be the same width, they should be longer to accommodate	Y	Y	
	a decent sized company name, and email address.	Y	Y	
6.3	At end of page, the question, 'How did you find the website?' should actually say 'How	Y	Y	
	did you find this website?'	Y	Y	
6.4	The listbox below 'How did you find this website?' should be closer. It looks like it	Y	Y	
	doesn't belong to the question.	Y	Y	
6.5	Include a 'please select' option in the listbox. The form should be rejected unless	Y	Y	
	another choice is selected.	Y	Y	
6.6	Enquiry: 'Is it possible to have the cursor ready in the first box when page loads?'	NO	Client	
6.7	Form should now be submitted to webresponse@workforcehr.com	Y		
6.8	On screen message should say: 'Thanks for your interest. We'll send the demo	Y		
	CD you have requested straight away.'	Y		
7	Information Request: Repeat changes (6.*, I.e. repeat all changes for Software Tour page)	Y		
7.1	The checkbox under 'would you like a copy of of our Software Tour?' should be closer.	Y	Y	
7.2	The explanatory text associated with the Software Tour checkbox is missing. SM has.	Y	Y	
7.3	The checkbox options under 'Which products are you interested in?' should be closer.	Y	Y	
7.4	No question mark at the end of 'Which products are you interested in?' is missing.	Y	Y	
7.5	The web recruitment checkbox is missing from the list of products. (above)			
8	Testimonials: Links should be sorted alphabetically by Client Name.	Y	Y	
9	Vacancies: First para, third sentence. Change'...any of the following positions' to	Y	Y	
	'...any of the positions listed here'.	Y	Y	
9.1	Submit CV:'Tell us about yourself' should not be in orange.	Y	Y	
9.2	To save space put the 'mandatory fields' bit in the previous para.	Y	Y	
9.3	Email field not mandatory.	Y	Y	
9.4	Home address should be a multi line address box.	Y	Y	
9.5	Home address and attach CV fields not mandatory.	Y	Y	
9.6	**There is a submission error.**			

Appendix B6: Change request dataset

Change request dataset table examples for Research Framework A.

No.	Description
1	Adding PDFs via the Admin Area Problem with adding PDFs via the admin area. Server seems to be adding an underscore character to spaces in filenames – with the result that the PDF then isn't found when the associated link is clicked in the site (and the home page is displayed instead). Work-around is to name PDFs without spaces, or with underscores in place of spaces. Bruno to talk to SOL.
2	Viewing the site using Mozilla Using Mozilla 0.97 most links in the client zone don't work.
3	Addition of Links to What's New, Noticeboard and News & Events pages Create a link via the admin area. When the link is clicked in the site, the page is refreshed into the small pop-up page. The linked page should open in the main site's browser window (replacing the page that was displayed when the link was clicked). Not sure if the small pop-up page should also be closed. For now, can we try leaving the small pop-up page open for the user to close when they are ready. Once I've seen it working it will be clear whether we need to auto-close it or not.
4	Addition of Links to What's New, Noticeboard and News & Events pages Regardless of the page URL for the link set up in the admin area, the home page is refreshed.
5	Prices PDF The new 'Training Price List' link is missing from the l.h.s. of the main Services>Training page. It should be immediately below the 'In-house or on-site' link.
6	Client Zone>Services>Interfaces Can you remove the final para of text on this page. The one beginning "To find out more about Interfacing Workforce …".
7	Opening PDFs Several clients have reported 'problems' with opening PDFs on the site (e.g. the PDF won't open) which we believe are really due to problems with the speed of their own PCs, their Internet connection etc. One way to mitigate this perceived problem would be to display a 'progress' message, or preferably a small and unobtrusive icon, whilst PDFs are

	loading. Would this be possible?
8	Problem with Forum Open the Forum. On the ‘All Forums’ page the date and time of the last posting is reported. In fact, the software is not picking up the most recent posting (which was today, 19th June) but is reporting a posting on the previous day at the last posting. Perhaps the software is looking at the TIME rather than the DATE?
9	email link should only underline the word ‘email’ not the words ‘email us with your suggestions’ (as now).
10	To save space, move the ‘If you would like to add to this list ….’ para to the bottom of the page, underneath the links.
11	To save space, move the ‘View Links by Category’ and listbox to the l.h.s of the page. Move the listbox closer to the title.
12	There should be no need for the ‘view’ button next to the ‘view links by category’ listobox. It should be possible to set the listbox up so that it automatically refreshes as soon as a different value is chosen.
13	It should be possible to assign up to 4 categories to each link in the Admin area. Currently, you can only choose 1 category for each link.
14	Links should be sorted alphabetically, by site name.
15	To save space, links should be displayed using a small point size and a simple colour highlight to denote the clickable URL (as example given in spec document).
16	To save space, only use a half space between each link.
17	Links should display the URL (not the site name) plus the description (as example given in spec document).

Table B.6 – Change requests dataset examples

Appendix C: Dataset repository

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
WFO	WFM1	Scrolling facility restricted	Bespoke Navigation Scroll Bar	Bespoke scroll bar	Scrolling device	Adaptive	Presentation	Yes	6 instances of same element on interface (GUI)	4	High	Object or entity of GUI interface control mechanism
WFO	WFM2	Up-date of On-screen message display	Text Heading	Acknowledgment output function	Function	Corrective	System	Yes	3 instances of same element on interface (GUI)	3	High	Acknowledgment process status. Business process
WFO	WFM3	Remove text	Character text content	Text content edit	TUI content	Adaptive	Presentation	No	None	0.5	Low	Symbols and characters included within text
WFO	WFM4	New mouse-over picture refresh	Mouse-over function	Roll-over picture refresh function	Function	Adaptive	System	Yes	New images for mouse over function	3	High	Graphic up-date link function
WFO	WFM5a	Replace incorrect numbers	Numeric content	Numeric content	TUI content	Corrective	Presentation	No	None	1	Medium- Low	
WFO	WFM5b	Numeric format consistency	Numeric content	Format numeric content	TUI content	Adaptive	Presentation	No	None	1	Medium- Low	Template for tel/fax numbers
WFO	WFM6a	Change text of link	Link Text Change	Text navigation link	Textual navigation hyperlink	Adaptive	Presentation	No	None	0.5	Low	Style sheet
WFO	WFM6b	Change Text	Text Change	Text content	TUI content	Adaptive	Presentation	No	None	1	Medium- Low	
WFO	WFM7	Ordering of story data elements	Content Sort Function	Story sort Function	Function	Adaptive	System	No	None	3	High	Style sheet template
WFO	WFM8	Secondary Client Navigation. Required outside client zone	Client view of navigation Uniformity of group view	Client zone secondary navigation function	Navigation function	Adaptive	System	Yes	Presentation layer Secondary Client Navigation. To be included for relevant user type	4	High	Add new link to style sheet
WFO	WFM9	Menu display not up-dated when required to show relevant content	Display function of list box menu	Option Indicator function	Function	Adaptive	System	No	None	3	High	Client admin function to change content. Content management system to handle text changes.
WFO	WFM10	Revisions to the process of enquiring	Submit form process	Business Function - Enquiry process	Function	Adaptive	System	No	None	2	Medium- High	Up-dated annually
WFO	WFM11	Change Text	Text Change	Text Content edit	TUI content	Adaptive	Presentation	No	None	1	Medium- Low	Client admin function to change content. Content management system to handle PDF changes. Control to website administrator

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
WFO	WFM12	Download of up-grade files	File download	Download file function	Function	Adaptive	System	Yes	Data elements to be re-named	5	High	Client admin function to change content. Content management system to handle PDF changes. Control to website administrator. PDF management facility within CMS to cope with PDF changes
WFO	WFM13	Change text of link	Text Link content	Text navigation link edit	Textual navigation hyperlink	Corrective	Presentation	No	None	0.5	Low	Client admin function to change content. Content management system to handle text changes.
WFO	WFM14	Remove text	Text content deletion	Text content edit	TUI content	Corrective	Presentation	No	None	1	Medium- Low	
WFO	WFM15	Text prompt missing	Text content	Text content label prompt	TUI content	Corrective	Presentation	Yes	2 instances of same element on interface	2	Medium- High	
WFO	WFM16	Error message when secondary pages accessed via Navigation. Box	Navigational link control (Navigation box)	Navigational link control function	Navigation function	Corrective	System	No	None	2	Medium- High	
WFO	WFM17	Unable to submit training request	Training submit form	Submit form function	Function	Corrective	System	No	None	2	Medium- High	
WFO	WFM18	Order of links	Navigation links	Text navigation link format	Textual navigation hyperlink	Adaptive	Presentation	Yes	Style consistency	2	Medium- High	
WFO	WFM19	Image link error	Navigation image link	Image navigation link	Multimedia hyperlink	Corrective	Presentation	No	None	1	Medium- Low	
WFO	WFM20	e-mail inquiry response confirmations	Acknowledgment e-mails	Acknowledgment e-mails function	Function	Corrective	System	Yes	11 instances of same element type	6	High	
WFO	WFM21a	Unwanted date/time stamp in heading of story	date/time function	Date/time stamp function	Function	Corrective	System	Yes	Pop-up's (secondary window) and 2 instances	2	Medium- High	
WFO	WFM21b	Admin area crash while editing press release and PDF's	Press release and PDF's edit function	Admin edit function	Admin function	Corrective	System	Yes	Check link to relevant PDF	7	High	Alteration
WFO	WFM22a	Text change	Text content	Text content edit	TUI content	Adaptive	Presentation	No	None	1	Medium- Low	Alteration

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
WFO	WFM22a(i)	Line space required between text	Line break	Text content format	TUI content format	Adaptive	Presentation	No	None	0.5	Low	Alteration
WFO	WFM22b	List-box title format and delete character content	Text content	Text content format	TUI content format	Adaptive	Presentation	Yes	Global view	2	Medium- High	Alteration
WFO	WFM22b(i)	Line space required between text	Text content	Text content format	TUI content format	Adaptive	Presentation	No	None	0.5	Low	Alteration
WFO	WFM22c	Inactive link	Navigation text Link	Navigation text link	Textual navigation hyperlink	Corrective	System	No	None	1	Medium- Low	Alteration
WFO	WFM22d	Preferred date & any date options in list box display	Date sort list-box function	Date list-box function	Function	Perfective	System	Yes	Submit form	5	High	Alteration
WFO	WFM22d(i)	Scrolling effect distorted	Text box scrolling	Form scrolling	Scrolling device	Corrective	Presentation	No	None	2	Medium- High	Alteration
WFO	WFM22d(ii)	Multiple date selection required for submit form	Date selection function	Date selection function	Function	Perfective	System	No	None	3	High	Alteration
WFO	WFM23	Text change	Text content	Text content edit	TUI content	Adaptive	Presentation	No	None	1	Medium- Low	Alteration
WFO	WFM24	File download	File name generation function	File name function	Function	Corrective	System	Yes	Data admin	5	High	Alteration
WFO	WFM25	E-mail acknowledgement process	e-mailing process	e-mail acknowledgement function	Function	Corrective	System	Yes	4 instances	6	High	Alteration
WFO	WFM26	Add function of PDF files to admin area	Add facility of PDF's	PDF Admin add function	Admin function	Perfective	System	Yes	3 instances	5	High	Alteration
WFO	WFM27	Update PDF data	Database entry	Update field	Database data	Adaptive	Data	No	None	3	High	Alteration
WFO	WFM28	PDF filename change	PDF filename change	PDF file name function	Function	Corrective	System	Yes	Change text link file names to new PDF filenames	3	High	Alteration

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
WFO	WFM29	Link errors using Mozilla OS	Link errors using Mozilla OS	Operating System compatibility	Function	Perfective	System	No	None	5	High	Alteration
WFO	WFM30	Update PDF data	Database entry	Update field	Database data	Adaptive	Data	Yes	Several instances	2	Medium- High	Alteration
WFO	WFM31	Dropdown list for forum inactive	Dropdown list for forum inactive	Forum dropdown list function	Function	Corrective	System	Yes	Text content & navigation	4	High	Alteration
WFO	WFM32a	Text change	Text change	Text content edit	TUI content	Corrective	Presentation	No	None	1	Medium- Low	Alteration
WFO	WFM32b	Text change	Text change	Text content edit	TUI content	Corrective	Presentation	No	None	1	Medium- Low	Alteration
WFO	WFM33a	Create new navigation text links	Create new navigation text links	Text navigation link	Textual navigation hyperlink	Perfective	System	Yes	Link to three pages within main site	2	Medium- High	New req
WFO	WFM33b	Refresh page into small pop-up page	Refresh page into small pop-up page	Secondary window pop-up function	Function	Perfective	System	Yes	Navigation path	4	High	Alteration
WFO	WFM34	Incorrect link path to homepage	Incorrect link path to homepage	Navigation link path	Textual navigation hyperlink	Corrective	System	Yes	Link to three pages within main site	1	Medium- Low	
WFO	WFM35	New navigational text link	New navigational text link	Text navigation link	Textual navigation hyperlink	Perfective	System	Yes	Link to three pages within main site	2	Medium- High	New Req
WFO	WFM36	Remove text from page	Remove text from page	Text content edit	TUI content	Adaptive	Presentation	No	None	1	Medium- Low	
WFO	WFM37	Display 'progress' message or small icon for downloading PDF's	Display 'progress' message or small icon for downloading PDF's	Download progress function	Function	Perfective	System	Yes	Progress interface object	6	High	New Req
WFO	WFM38	Forum not picking up most recent posting	Forum not picking up most recent posting	Forum sort function	Function	Corrective	System	Yes	Date and time stamp	8	High	Alteration
WFO	WFM39	Text link should only underline particular word	Text link should only underline particular word	Text navigation link format	Textual navigation hyperlink	Adaptive	Presentation	No	None	1	Medium- Low	Alteration

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
WFO	WFM40	Move paragraph to bottom of page	Move paragraph to bottom of page	Text content format	TUI content format	Adaptive	Presentation	Yes	Page alignment	2	Medium- High	Alteration
WFO	WFM41	Move title and list-box to left hand side of page	Move title and list-box to left hand side of page	Page content format	Style issue	Adaptive	Presentation	Yes	Page alignment	2	Medium- High	Alteration
WFO	WFM42a	Remove 'view' button from list-box	Remove 'view' button from list-box	View button function edit	Function	Perfective	System	Yes	Disable function	2	Medium- High	Different currency
WFO	WFM42b	Automate list-box refresh ability when a different value is chosen	Automate list-box refresh ability when a different value is chosen	List-box refresh function	Function	Perfective	System	No	None	5	High	To data file
WFO	WFM43	Ability to assign 4 categories to each link in the admin area	Ability to assign 4 categories to each link in the admin area	Admin link creation function	Admin function	Perfective	System	No	None	4	High	Pres as text label change
WFO	WFM44	Links to be sorted alphabetically	Links to be sorted alphabetically	Link sort function	Function	Perfective	System	No	None	2	Medium- High	
WFO	WFM45	Links displayed in small point size and simple colour	Links displayed in small point size and simple colour	Text navigation link format	Textual navigation hyperlink	Perfective	Presentation	Yes	Site uniformity	1	Medium- Low	If size changed then image scrolling with need to be implemented
WFO	WFM46	Half space between each link	Half space between each link	Page style format	Style issue	Perfective	Presentation	No	None	0.5	Low	STYLE
WFO	WFM47	Links to display URL & description	Links to display URL & description	Link description function	Function	Adaptive	System	No	None	3	High	
WFO	WFM48	Visual cue for scroll bar not visible	Visual cue for scroll bar not visible	Visual cue image	Image	Corrective	Presentation	Yes	All scroll bar devices	3	High	
WFO	WFM49a	Scroll bar difficult to use	Scroll bar difficult to use	Scroll bar dimensions	Scrolling device	Adaptive	Presentation	Yes	All scroll bar devices	3	High	
WFO	WFM49b	Visual problem for active and non-active regions	Visual problem for active and non-active regions	Mouse-over icon function	Function	Adaptive	System	Yes	Active and non-active regions of site	4	High	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
WFO	WFM50	Bottom line of text cut horizontally	Bottom line of text cut horizontally	Page dimensions	Style issue	Corrective	Presentation	Yes	Page alignment	2	Medium- High	
WFO	WFM51	Right-hand margin in pop-up pages too tight	Right-hand margin in pop-up pages too tight	Secondary window dimensions	Style issue	Adaptive	Presentation	No	None	1	Medium- Low	
WFO	WFM52	Paragraph breaks in text added via the admin area for display in pop-up pages	Paragraph breaks in text added via the admin area for display in pop-up pages	Paragraph break admin function	Admin function	Perfective	System	No	None	2	Medium- High	No changes done on the CMS. All the relevant web pages
WFO	WFM53	All fields in submit form throughout site should be scrollable sideways	All fields in submit form throughout site should be scrollable sideways	Submit form navigation scroll bar function	Navigation function	Perfective	System	No	None	2	Medium- High	
WFO	WFM54	On-screen message display after form submission consistency	On-screen message display after form submission consistency	Text content format	TUI content format	Adaptive	Presentation	Yes	Acknowledgement style consistency	1	Medium- Low	Style sheet
WFO	WFM55	Back button must be pressed twice in IE.5	Back button must be pressed twice in IE.5	Browser compatibility function	Function	Perfective	System	No	None	3	High	
WFO	WFM56	File size should be reported to two significant figures	File size should be reported to two significant figures	File size function	Function	Perfective	System	No	None	1	Medium- Low	
WFO	WFM57	The second and subsequent time you view a page loading should be quicker	The second and subsequent time you view a page loading should be quicker	Page refresh function	Function	Perfective	System	No	None	12	High	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
WFO	WFM58	Print function to output the contents of all associated pop-up's as well as text on the main page	Print function to output the contents of all associated pop-up's as well as text on the main page	Print function	Function	Perfective	System	Yes	All product pages & Why WF page	6	High	
WFO	WFM59	Text headings in wrong format	Text headings in wrong format	Text image format	image	Corrective	Presentation	Yes	Global site consistency	1	Medium- Low	Style sheet template
WFO	WFM60	Cursor prompt for submit form fields	Cursor prompt for submit form fields	Cursor prompt function submit form fields	Function	Perfective	System	No	None	2	Medium- High	On-going
WFO	WFM61	Submit form to retain data inputted and prompted for un-fulfilled data items	Submit form to retain data inputted and prompted for un-fulfilled data items	Submit form field function	Function	Perfective	System	No	None	2	Medium- High	COST factor
WFO	WFM62a	Sideways links instead of downwards for pages that do not have any further 'downwards' options	Sideways links instead of downwards for pages that do not have any further 'downwards' options	Text navigation link display	Textual navigation hyperlink	Perfective	Presentation	Yes	All pages except products/application pages	3	High	
WFO	WFM62b	New header for navigation links	New header for navigation links	Text header image	image	Adaptive	Presentation	No	None	2	Medium- High	Flexibility in terms of adding new content to page sections
WFO	WFM62c	New links to be underneath existing links with a small gap separating them	New links to be underneath existing links with a small gap separating them	Text navigation link creation function	Navigation function	Adaptive	Presentation	Yes	Page alignment	1	Medium- Low	
WFO	WFM63	Align navigation levels 2 & 3 link box's	Align navigation levels 2 & 3 link box's	Link box image format	image	Adaptive	Presentation	No	None	2	Medium- High	
WFO	WFM64	Incorrect indicator of path in client zone	Incorrect indicator of path in client zone	Text navigation link indicator of path	Textual navigation hyperlink	Corrective	Presentation	Yes	Traceability of current position and path	2	Medium- High	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
WFO	WFM65	Add Homepage text link to top of site	Add Homepage text link to top of site	Text navigation link	Textual navigation hyperlink	Perfective	System	Yes	Page alignment	1	Medium- Low	Add new link to style sheet
WFO	WFM66	Graphic device display differences in dark/bold	Graphic device display differences in dark/bold	Image format	image	Corrective	Presentation	Yes	Graphic (image) device style uniformity	1	Medium- Low	
WFO	WFM67	Image refresh when hovering over text navigation links at top of site	Image refresh when hovering over text navigation links at top of site	Image refresh function	Function	Adaptive	System	Yes	Mouse-over function	3	High	
WFO	WFM68	Secondary title to join main title	Secondary title to join main title	Text content format	TUI content format	Adaptive	Presentation	No	None	1	Medium- Low	
WFO	WFM69	Delete text paragraph from page	Delete text paragraph from page	Text content edit	TUI content	Adaptive	Presentation	No	None	0.5	Low	
WFO	WFM70	Missing checkbox on information request page	Missing checkbox on information request page	Check box input function	Function	Corrective	System	Yes	Page alignment & check-box label	2	Medium- High	
WFO	WFM71	Incorrect message display after form submission	Incorrect message display after form submission	Text content edit	TUI content	Corrective	Presentation	No	None	2	Medium- High	
WFO	WFM72	Remove text from above submit form	Remove text from above submit form	Text content edit	TUI content	Adaptive	Presentation	No	None	1	Medium- Low	
WFO	WFM73	Text content missing from top of page	Text content missing from top of page	Text content	TUI content	Corrective	Presentation	Yes	Page alignment	1	Medium- Low	
WFO	WFM74	Error message when trying to add new testimonials via admin area	Error message when trying to add new testimonials via admin area	PDF file manager admin function	Admin function	Corrective	System	Yes	Viewing PDF	7	High	
WFO	WFM75	Incorrect message display after form submission	Incorrect message display after form submission	Text content edit	TUI content	Corrective	Presentation	No	None	2	Medium- High	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
WFO	WFM76	Text content modifications	Text content modifications	Text content edit	TUI content	Corrective	Presentation	Yes	Navigational link within text content	1	Medium- Low	
WFO	WFM77	Incorrect text displayed in pop-up page (secondary window)	Incorrect text displayed in pop-up page (secondary window)	Text content edit	TUI content	Corrective	Presentation	Yes	Secondary window output	2	Medium- High	
WFO	WFM78	Underline navigation links	Underline navigation links	Text navigation link format	Textual navigation hyperlink	Adaptive	Presentation	Yes	3 instances	0.5	Low	
WFO	WFM79	Screen grab missing from page	Screen grab missing from page	Screen grab function	Function	Corrective	System	No	None	4	High	
WFO	WFM80	Remove text character from navigation link	Remove text character from navigation link	Text navigation content edit	Textual navigation hyperlink	Corrective	Presentation	No	None	1	Medium- Low	
WFO	WFM81	Incorrect text content on pop-up page	Incorrect text content on pop-up page	Text content edit	TUI content	Corrective	Presentation	No	None	1	Medium- Low	
WFO	WFM82	New extra link to open in separate browser window	New extra link to open in separate browser window	Text navigation link	Textual navigation hyperlink	Perfective	System	Yes	Page alignment - requires space & Secondary window display	3	High	
WFO	WFM83	Text indicator of path incorrect	Text indicator of path incorrect	Text navigation link indicator of path	Textual navigation hyperlink	Corrective	Presentation	Yes	Traceability of current position and path	2	Medium- High	
WFO	WFM84	Missing screen grab image from page	Missing screen grab image from page	Screen grab Image	image	Corrective	Presentation	No	None	1	Medium- Low	
WFO	WFM85	New navigation device - add 4 new links to open in pop-up pages	New navigation device - add 4 new links to open in pop-up pages	Navigation list box device function	Navigation function	Perfective	System	Yes	Page alignment & 4 text navigation link paths	3	High	
WFO	WFM86	New text to include in page content	New text to include in page content	Text content	TUI content	Adaptive	Presentation	No	None	1	Medium- Low	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
WFO	WFM87	Remove text paragraph from page	Remove text paragraph from page	Text content edit	TUI content	Adaptive	Presentation	No	None	0.5	Low	
WFO	WFM88	Missing screen grab image from page	Missing screen grab image from page	Screen grab Image	image	Corrective	Presentation	No	None	1	Medium- Low	
WFO	WFM89	Remove text from navigation menu	Remove text from navigation menu	Text content edit	TUI content	Adaptive	Presentation	No	None	0.5	Low	
WFO	WFM90	Page headings missing from 3rd level pages	Page headings missing from 3rd level pages	Text image	image	Corrective	Presentation	No	None	0.5	Low	
WFO	WFM91	Missing screen grab image from 4 page	Missing screen grab image from 4 page	Screen grab Image	image	Corrective	Presentation	Yes	4 instances	1	Medium- Low	
WFO	WFM92	Incorrect text bullet list for page	Incorrect text bullet list for page	Text content edit	TUI content	Corrective	Presentation	No	None	0.5	Low	
WFO	WFM93	Remove word from four paragraphs	Remove word from four paragraphs	Text content edit	TUI content	Adaptive	Presentation	No	None	0.5	Low	
WFO	WFM94	Navigation text link style uniformity	Navigation text link style uniformity	Navigation text format	Textual navigation hyperlink	Corrective	Presentation	No	None	1	Medium- Low	
WFO	WFM95	Navigation text link style uniformity	Navigation text link style uniformity	Navigation text format	Textual navigation hyperlink	Corrective	Presentation	No	None	1	Medium- Low	
WFO	WFM96	Incorrect image displayed when screen grab clicked	Incorrect image displayed when screen grab clicked	Screen grab function	Function	Corrective	System	No	None	2	Medium- High	
WFO	WFM97	Missing sideways text links	Missing sideways text links	Text navigation link	Textual navigation hyperlink	Corrective	System	No	None	2	Medium- High	
WFO	WFM98	Change third level navigation menu list	Change third level navigation menu list	Text navigation link edit	Textual navigation hyperlink	Perfective	Presentation	Yes	New links at 2nd & 3rd level navigation	2	Medium- High	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
WFO	WFM99	Line spacing between text paragraphs	Line spacing between text paragraphs	Text content format	TUI content format	Adaptive	Presentation	Yes	Page alignment	0.5	Low	
WFO	WFM100	Missing text content	Missing text content	Text content	TUI content	Corrective	Presentation	No	None	0.5	Low	
WFO	WFM101a	New text to be added at end of page	New text to be added at end of page	Text content	TUI content	Adaptive	Presentation	No	None	0.5	Low	
WFO	WFM101b	Two text navigation links to point to PDF file & html page	Two text navigation links to point to PDF file & html page	Text navigation link path	Textual navigation hyperlink	Perfective	System	Yes	Link path to data file and second link to html page	2	Medium- High	
WFO	WFM102	New navigation link	New navigation link	Text navigation link	Textual navigation hyperlink	Perfective	System	Yes	Link path to PDF file & page alignment	2	Medium- High	
WFO	WFM103	Typo in paragraph	Typo in paragraph	Text content edit	TUI content	Corrective	Presentation	No	None	0.5	Low	
WFO	WFM104	New bullet point for page text	New bullet point for page text	Text content edit	TUI content	Adaptive	Presentation	No	None	0.5	Low	
WFO	WFM105a	Remove file size at end of text and include a full-stop	Remove file size at end of text and include a full-stop	Text content edit	TUI content	Adaptive	Presentation	No	None	1	Medium- Low	
WFO	WFM105b	Navigation link should not be in capitals	Navigation link should not be in capitals	Text navigation link format	Textual navigation hyperlink	Corrective	Presentation	No	None	0.5	Low	
WFO	WFM106a	New text to be added at end of paragraph	New text to be added at end of paragraph	Text content	TUI content	Perfective	Presentation	No	None	0.5	Low	
WFO	WFM106b	Two text navigation links to point to PDF file & html page	Two text navigation links to point to PDF file & html page	Text navigation link path	Textual navigation hyperlink	Adaptive	System	Yes	Link path to data file and second link to html page	2	Medium- High	
WFO	WFM107	New navigation link	New navigation link	Text navigation link	Textual navigation hyperlink	Perfective	System	Yes	Link path to PDF file & page alignment	1	Medium- Low	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
WFO	WFM108	Move text headings as clickable links to left hand side & contents to be displayed in pop-up window	Move text headings as clickable links to left hand side & contents to be displayed in pop-up window	Text navigation link	Textual navigation hyperlink	Perfective	System	Yes	Delete text headings & include secondary window function	3	High	
WFO	WFM109	New links on left hand side are displayed alphabetically	New links on left hand side are displayed alphabetically	Text navigation link format	Textual navigation hyperlink	Perfective	Presentation	No	None	1	Medium- Low	
WFO	WFM110	Missing text link	Missing text link	Text navigation link	Textual navigation hyperlink	Corrective	System	Yes	Page alignment	1	Medium- Low	
WFO	WFM111	Pop-up page for text link should use the link title as its heading	Pop-up page for text link should use the link title as its heading	Text header image	image	Corrective	Presentation	No	None	2	Medium- High	
WFO	WFM112	Gap between links	Gap between links	Text navigation link format	Textual navigation hyperlink	Adaptive	Presentation	No	None	1	Medium- Low	
WFO	WFM113	Remove text links	Remove text links	Text navigation link edit	Textual navigation hyperlink	Corrective	Presentation	Yes	Two instances	0.5	Low	
WFO	WFM114	Remove text links	Remove text links	Text navigation link edit	Textual navigation hyperlink	Corrective	Presentation	Yes	Two instances	0.5	Low	
WFO	WFM115	Remove text links	Remove text links	Text navigation link edit	Textual navigation hyperlink	Corrective	Presentation	Yes	Two instances	0.5	Low	
LJO	LJ1a	New text content	New text content	Text label	TUI content	Perfective	Presentation	Yes	Two instances of & page alignment	0.5	Low	
LJO	LJ1b	Add new text in text box	Add new text in text box	Text content	TUI content	Adaptive	Presentation	Yes	Two instances of & page alignment	0.5	Low	Data if CMS
LJO	LJ2a	New text content	New text content	Text box label	TUI content	Perfective	Presentation	Yes	Page alignment	0.5	Low	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
LJO	LJ2c	New content managed section	New content managed section	Text content managed function	Admin function	Perfective	System	Yes	Backend data structure	2	Medium- High	Client admin function to change content. Content management system to handle text changes.
LJO	LJ2b	Add new text in text box	Add new text in text box	Text content	TUI content	Adaptive	Data	Yes	Page alignment	0.5	Low	Data if CMS
LJO	LJ3a	New text box	New text box	Text box content	TUI content	Perfective	Presentation	Yes	Two instances of & page alignment	1	Medium- Low	
LJO	LJ3b	Add new text in text box	Add new text in text box	Text content	TUI content	Adaptive	Presentation	Yes	Two instances of & alignment	0.5	Low	Data if CMS
LJO	LJ4a	Remove bullet point numbering system for text content	Remove bullet point numbering system for text content	Format text content bullet points	TUI content format	Adaptive	Presentation	Yes	Page alignment	1	Medium- Low	
LJO	LJ4b	Align text on left hand side of page with sub-headings	Align text on left hand side of page with sub-headings	Page style format	Style issue	Adaptive	Presentation	No	None	1.5	Medium- High	
LJO	LJ5	Table column to be aligned through each section	Table column to be aligned through each section	Style format for table columns	Style issue	Adaptive	Presentation	Yes	Relevant pages consistency	2	Medium- High	Universal style
LJO	LJ6a	Reduce width of columns in table	Reduce width of columns in table	Style format for table columns	Style issue	Adaptive	Presentation	Yes	All column's	1	Medium- Low	
LJO	LJ6b	Text wrap for table columns	Text wrap for table columns	Text wrap functionality	Function	Adaptive	System	Yes	All columns	5	High	
LJO	LJ7	New facility to delete headings in admin section	New facility to delete headings in admin section	Text header delete function	Admin function	Perfective	System	Yes	Admin function	3	High	New Business requirement.
LJO	LJ8a	Include new section to existing page	Include new section to existing page	Text content	TUI content	Perfective	Presentation	Yes	Page alignment	1	Medium- Low	Up-dated annually
LJO	LJ8b	Include new section to existing page	Include new section to existing page	Text content	TUI content	Perfective	Presentation	Yes	Page alignment	1	Medium- Low	
LJO	LJ9	Text navigational link	Text navigational link	Text navigational link	Textual navigation hyperlink	Perfective	System	Yes	Navigation path	1	Medium- Low	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
LJO	LJ10	Horizontal tab colour format	Horizontal tab colour format	Content style issue	Style issue	Adaptive	Presentation	Yes	Uniformity across site	1	Medium- Low	
LJO	LJ11	Home page title format	Home page title format	Text title content format	TUI content format	Corrective	Presentation	No	None	1	Medium- Low	
LJO	LJ12	Text content display issue	Text content display issue	Text content display	TUI content	Corrective	Presentation	No	None	1	Medium- Low	
LJO	LJ13	Investment calculations	Investment calculations	Database query function	Function	Corrective	Data	Yes	Data function	10	High	
LJO	LJ14a	Format text content	Format text content	Text content edit	TUI content	Corrective	Presentation	No	None	0.5	Low	
LJO	LJ14b	Admin access to text content inactive page	Admin access to text content inactive page	Inactive page through admin access	TUI content	Corrective	Data	No	None	4	High	
LJO	LJ15	Text content edit	Text content edit	Text content edit	TUI content	Corrective	Presentation	No	None	0.25	Low	
LJO	LJ16	Adding new PDF's function	Adding new PDF's function	Admin PDF replacement function	Admin function	Perfective	System	Yes	Procedures description	6	High	Client admin function to change content. Content management system to handle PDF changes. Control to website administrator
LJO	LJ17	Counter function for website hits	Counter function for website hits	Counter function	Function	Perfective	System	No	None	4	High	
LJO	LJ18	Brochure page requires text box for addresses	Brochure page requires text box for addresses	Text box input function	Function	Perfective	Presentation	Yes	Data structure and access	3	High	
LJO	LJ19	Up-date bulletin function in admin pages inactive	Up-date bulletin function in admin pages inactive	Admin content up-date function	Admin function	Corrective	System	No	None	7	High	
LJO	LJ20	Delete text character content	Delete text character content	Text content edit	TUI content	Corrective	Presentation	No	None	0.5	Low	
LJO	LJ21	Add LJ logo	Add LJ logo	Image	Image	Adaptive	Presentation	No	None	1.5	Medium- High	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
LJO	LJ22	Page style format - logo to be moved	Page style format - logo to be moved	Page style format	Style issue	Adaptive	Presentation	No	None	1	Medium- Low	
LJO	LJ23a	Up-date page section with relevant content	Up-date page content	Text content page	TUI content	Adaptive	Presentation	No	None	1.5	Medium- High	
LJO	LJ23b	Navigational text link required	Navigational text link required	Text navigational link	Textual navigation hyperlink	Perfective	System	Yes	Link coordination	1	Medium- Low	
LJO	LJ24a	Interest rate box required on another page	Interest rate box required on another page	Text content	TUI content	Perfective	Presentation	Yes	Page alignment	1	Medium- Low	
LJO	LJ24b	Text deletion	Text deletion	Text content edit	TUI content	Adaptive	Presentation	No	None	0.5	Low	
LJO	LJ25	Interest rates box on another page	Interest rates box on another page	Text content	TUI content	Perfective	Presentation	Yes	Page alignment	0.5	Low	
LJO	LJ26	Remove text	Remove text	Text content edit	TUI content	Adaptive	Presentation	No	None	0.5	Low	
LJO	LJ27	Text amendment	Text amendment	Text content edit	TUI content	Adaptive	Presentation	No	None	0.25	Low	
LJO	LJ28a	Incorrect admin view of text content layout	Incorrect admin view of text content layout	Text content style format	TUI content format	Corrective	Presentation	Yes	Data management function	1.5	Medium- High	
LJO	LJ28b	Owl image to be included	Owl image to be included	Image	Image	Adaptive	Presentation	Yes	Page alignment	1.5	Medium- High	
LJO	LJ29a	Inclusion of logo image	Inclusion of logo image	Image	Image	Adaptive	Presentation	Yes	Page alignment	1.5	Medium- High	
LJO	LJ29b	Move text box down	Move text box down	Interface page layout style	Style issue	Adaptive	Presentation	Yes	Page alignment	2	Medium- High	
LJO	LJ29c	Text content format -Bold	Text content format -Bold	Text content format	TUI content format	Adaptive	Presentation	No	None	1	Medium- Low	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
LJO	LJ29d	Text content re-position on page layout	Text content re-position on page layout	Page layout style	Style issue	Adaptive	Presentation	Yes	Page alignment	2	Medium- High	
LJO	LJ29e	Navigational link from text content	Navigational link from text content	Navigational text link	Textual navigation hyperlink	Adaptive	System	Yes	Navigational path	1	Medium- Low	
LJO	LJ29f	Delete bullet point and content	Delete bullet point and content	Text content edit	TUI content	Adaptive	Presentation	No	None	0.5	Low	
LJO	LJ29g	Text content format -Bold	Text content format -Bold	Text content format	TUI content format	Adaptive	Presentation	Yes	Three instances	0.5	Low	
LJO	LJ29h	Move text boxes lower down page	Move text boxes lower down page	Page layout style format	Style issue	Adaptive	Presentation	Yes	Page alignment	2	Medium- High	
LJO	LJ30a	Change wording of PDF link	Change wording of PDF link	Navigational text link edit	Textual navigation hyperlink	Adaptive	Presentation	Yes	Two instances & navigational link path to data file	1.5	Medium- High	
LJO	LJ30b	Text content inclusion	Text content inclusion	Text content	TUI content	Adaptive	Data	Yes	Page format & alignment	1	Medium- Low	
LJO	LJ30c	Date and price output style	Date and price output style	Numeric content format	TUI content	Adaptive	Presentation	Yes	Global rule	1	Medium- Low	
LJO	LJ30d	Content managed function of price to be included	Content managed function of price to be included	Content management price function	Admin function	Perfective	System	Yes	Admin of content management access	4	High	
LJO	LJ30e	Text amendment	Text amendment	Text content edit	TUI content	Adaptive	Presentation	No	None	0.25	Low	
LJO	LJ31	Text Amendment	Text Amendment	Text content edit	TUI content	Adaptive	Presentation	No	None	0.25	Low	
LJO	LJ32	Inclusion of logo image	Inclusion of logo image	Image	Image	Adaptive	Presentation	Yes	Page alignment	1.5	Medium- High	
LJO	LJ33	Inclusion of logo image	Inclusion of logo image	Image	Image	Adaptive	Presentation	Yes	Page alignment	1.5	Medium- High	
LJO	LJ34a	Image size reduction	Image size reduction	Image format	Image	Adaptive	Presentation	No	None	2	Medium- High	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
LJO	LJ34b	Add logo bottom right hand side of page	Add logo bottom right hand side of page	Image & layout	Image	Adaptive	Presentation	Yes	Page layout & alignment	1	Medium- Low	
LJO	LJ35	Text content navigational link	Text content navigational link	Text content navigational link edit	Textual navigation hyperlink	Adaptive	Presentation	No	None	1	Medium- Low	
LJO	LJ36	New photograph	New photograph	Photograph image	Image	Adaptive	Presentation	Yes	Page alignment	1	Medium- Low	
LJO	LJ37	New photograph	New photograph	Photograph image	Image	Adaptive	Presentation	Yes	Page alignment	1	Medium- Low	
LJO	LJ38	Centre site in browser	Centre site in browser	Page layout style	Style issue	Perfective	Presentation	No	None	3	High	New requirement
LJO	LJ39	Background colour for browser	Background colour for browser	Page colour style format	Style issue	Adaptive	Presentation	Yes	Global uniformity	1	Medium- Low	
LJO	LJ40	Image size change	Image size change	Image format	Image	Adaptive	Presentation	No	None	1.5	Medium- High	
LJO	LJ41	Text content style and contrast issue	Text content style and contrast issue	Text content format	TUI content format	Adaptive	Presentation	Yes	Global site uniformity	1	Medium- Low	
LJO	LJ42	Contact facility for each section to be removed	Contact facility for each section to be removed	Text content edit	TUI content	Adaptive	Presentation	Yes	Page layout & format	1	Medium- Low	
LJO	LJ43	Introduction page review	Introduction page review	Introduction page style review	Style issue	Adaptive	Presentation	No	None	3	High	
LJO	LJ44	Disclaimer section content addition	Disclaimer section content addition	Text content	TUI content	Adaptive	Presentation	Yes	Page alignment	1	Medium- Low	
LJO	LJ45	Why LJ section content addition	Why LJ section content addition	Text content	TUI content	Adaptive	Presentation	Yes	Page alignment	1	Medium- Low	
LJO	LJ46	Company structure content addition	Company structure content addition	Text content	TUI content	Adaptive	Data	Yes	Page alignment & Several instances for each business unit	0.5	Low	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
LJO	LJ47	Published report section - include reports	Published report section - include reports	Text content	TUI content	Adaptive	Data	Yes	Links to PDF files	1	Medium- Low	
LJO	LJ48	Add designed image	Add designed image	Designed image	Image	Adaptive	Presentation	Yes	Page alignment	2	Medium- High	
LJO	LJ49	Offshore accounts section content addition	Offshore accounts section content addition	Text content	TUI content	Adaptive	Data	Yes	Page alignment	1	Medium- Low	
LJO	LJ50	Add designed image	Add designed image	Designed image	Image	Adaptive	Presentation	Yes	Page alignment	2	Medium- High	
LJO	LJ51a	Funds section to include navigational link to PDF file	Funds section to include navigational link to PDF file	Text Navigational link to data source	Textual navigation hyperlink	Adaptive	System	Yes	PDF file storage and path	2	Medium- High	
LJO	LJ51b	Navigational link title	Navigational link title	Text navigational link title format	Textual navigation hyperlink	Adaptive	Presentation	Yes	Navigational path	1	Medium- Low	
LJO	LJ52	Alternative investments section to include navigational link to PDF file	Alternative investments section to include navigational link to PDF file	Text Navigational link to data source	Textual navigation hyperlink	Adaptive	System	Yes	PDF file storage and path	2	Medium- High	
LJO	LJ53	Guernsey section text content addition	Guernsey section text content addition	Text content	TUI content	Adaptive	Data	Yes	Page alignment	1	Medium- Low	
LJO	LJ54	Bahamas section text content addition	Bahamas section text content addition	Text content	TUI content	Adaptive	Data	Yes	Page alignment	1	Medium- Low	
LJO	LJ55	Investment challenge section text content addition	Investment challenge section text content addition	Text content	TUI content	Adaptive	Data	Yes	Page alignment	0.5	Low	
LJO	LJ56	About the game section text content	About the game section text content	Text content edit	TUI content	Adaptive	Data	Yes	Page alignment	0.5	Low	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
LJO	LJ57	Performance & Standings section calculation function	Performance & Standings section - calculation function	Performance & Standings function	Function	Perfective	System	Yes	Output of results to relevant section and format	12	High	
LJO	LJ58	Brochure section text content addition	Brochure section text content addition	Text content	TUI content	Adaptive	Data	Yes	Page alignment	0.5	Low	
LJO	LJ59	Contact us section text content	Contact us section text content	Text content edit	TUI content	Adaptive	Data	Yes	Page alignment	0.5	Low	
LJO	LJ60	Bulletins section to link to relevant bulletins	Bulletins section to link to relevant bulletins	Text navigational link to data files	Textual navigation hyperlink	Adaptive	System	Yes	Data files storage and link path	1	Medium- Low	
LJO	LJ61	Tab feature colour change	Tab feature colour change	Tab feature colour style	Style issue	Adaptive	Presentation	Yes	Global site uniformity	1	Medium- Low	
LJO	LJ62	Investment challenge section - text content edit	Investment challenge section - text content edit	Text content edit	TUI content	Adaptive	Presentation	No	None	0.5	Low	
LJO	LJ63	PDF inclusion to content management system	PDF inclusion to content management system	Data entry function of PDF's via CMS to database	Admin function	Perfective	System	No	None	6	High	Client admin function to change content. Content management system to handle PDF changes. Control to website administrator. PDF management facility within CMS to cope with PDF changes
LJO	LJ64	Image modification	Image modification	Edit image	Image	Adaptive	Presentation	No	None	1.5	Medium- High	
LJO	LJ65	Add new text content	Add new text content	Text content	TUI content	Adaptive	Presentation	Yes	Page alignment	0.5	Low	
LJO	LJ66	Create new page as sample for directors	Create new page as sample for directors	New text content page	TUI content	Adaptive	Presentation	Yes	Page layout & design	1	Medium- Low	
LJO	LJ67	Re-design country access section	Re-design country access section	Re-design page layout & style	Style issue	Adaptive	Presentation	Yes	Across page alignment and content	4	High	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
LJO	LJ68	Text label edit	Text label edit	Text content label format	TUI content format	Corrective	Presentation	No	None	0.5	Low	
LJO	LJ69	Thin white line on page - deletion	Thin white line on page - deletion	Image edit	Image	Corrective	Presentation	No	None	1	Medium- Low	
LJO	LJ70	Text title cropped at bottom	Text title cropped at bottom	Text image format	Image	Corrective	Presentation	No	None	1.5	Medium- High	
LJO	LJ71	Text content edit for consistency	Text content edit for consistency	Text content edit	TUI content	Corrective	Presentation	No	None	0.5	Low	
LJO	LJ72	Incorrect text content on page	Incorrect text content on page	Text content call function	Function	Corrective	System	No	None	2	Medium- High	
LJO	LJ73	Data source error report - no menu viewable	Data source error report - no menu viewable	Database call function	Function	Corrective	System	Yes	Data elements	3	High	
LJO	LJ74	Data source error report - no menu viewable	Data source error report - no menu viewable	Database call function	Function	Corrective	System	Yes	Data elements	3	High	
LJO	LJ75	Incorrect text content on page	Incorrect text content on page	Text content call function	Function	Corrective	System	No	None	2	Medium- High	
LJO	LJ76	Charges section requires database and admin	Charges section requires database and admin	New content managed section & admin function	Admin function	Perfective	System	Yes	Admin access to include new change process	7	High	Client admin function to change content. Content management system to handle text changes.
LJO	LJ77	London interest rates section requires database and admin	London interest rates section requires database and admin	New content managed section & admin function	Admin function	Perfective	System	Yes	Admin access to include new change process	7	High	New req. Duplicated charges code
LJO	LJ78	Text content to be moved to secondary page of section	Text content to be moved to secondary page of section	Page layout style	Style issue	Adaptive	Presentation	Yes	Page layout reformat	2	Medium- High	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
LJO	LJ79	Charges sub-section requires database and admin	Charges sub-section requires database and admin	New content managed section & admin function	Admin function	Perfective	System	Yes	Admin access to include new change process	7	High	New req. Duplicated charges code. CMS modulated function to code with this type of change
LJO	LJ80	London interest rates sub-section requires database and admin	London interest rates sub-section requires database and admin	New content managed section & admin function	Admin function	Perfective	System	Yes	Admin access to include new change process	6	High	New req. Duplicated charges code. CMS modulated function to code with this type of change
LJO	LJ81	Guernsey interest rates section requires database and admin	Guernsey Interest rates section requires database and	New content managed section & admin function	Admin function	Perfective	System	Yes	Admin access to include new change process	7	High	New req. Duplicated charges code. CMS modulated function to code with this type of change
LJO	LJ82	Text in bullet points being cut off	Text in bullet points being cut off	Text content format	TUI content format	Corrective	Presentation	No	None	1	Medium- Low	
LJO	LJ83	Scroll button to be in line with left margin of text	Scroll button to be in line with left margin of text	Page layout style	Style issue	Corrective	Presentation	Yes	Page alignment	3	High	
LJO	LJ84a	Scroll button to be in line with left margin of text	Scroll button to be in line with left margin of text	Page layout style	Style issue	Corrective	Presentation	Yes	Page alignment	3	High	
LJO	LJ84b	Text content to be moved to secondary page of section	Text content to be moved to secondary page of section	Page layout style	Style issue	Adaptive	Presentation	Yes	Page layout reformat	2	Medium- High	
LJO	LJ85a	Scroll button to be in line with left margin of text	Scroll button to be in line with left margin of text	Page layout style	Style issue	Corrective	Presentation	Yes	Page alignment	3	High	
LJO	LJ85b	Text content to be moved to secondary page of section	Text content to be moved to secondary page of section	Page layout style	Style issue	Adaptive	Presentation	Yes	Page layout reformat	2	Medium- High	
LJO	LJ86	Content looks sparse	Content looks sparse	Text content format	TUI content format	Adaptive	Presentation	No	None	1	Medium- Low	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
LJO	LJ87a	Scroll button to be in line with left margin of text	Scroll button to be in line with left margin of text	Page layout style	Style issue	Corrective	Presentation	Yes	Page alignment	3	High	
LJO	LJ87b	Text content to be moved to secondary page of section	Text content to be moved to secondary page of section	Page layout style	Style issue	Adaptive	Presentation	Yes	Page layout reformat	2	Medium- High	
LJO	LJ88a	Scroll button to be in line with left margin of text	Scroll button to be in line with left margin of text	Page layout style	Style issue	Corrective	Presentation	Yes	Page alignment	3	High	
LJO	LJ88b	Text content to be moved to secondary page of section	Text content to be moved to secondary page of section	Page layout style	Style issue	Adaptive	Presentation	Yes	Page layout reformat	2	Medium- High	
LJO	LJ89a	Scroll button to be in line with left margin of text	Scroll button to be in line with left margin of text	Page layout style	Style issue	Corrective	Presentation	Yes	Page alignment	3	High	
LJO	LJ89b	Text content to be moved to secondary page of section	Text content to be moved to secondary page of section	Page layout style	Style issue	Adaptive	Presentation	Yes	Page layout reformat	2	Medium- High	
LJO	LJ90a	Scroll button to be in line with left margin of text	Scroll button to be in line with left margin of text	Page layout style	Style issue	Corrective	Presentation	Yes	Page alignment	3	High	
LJO	LJ90b	Text content to be moved to secondary page of section	Text content to be moved to secondary page of section	Page layout style	Style issue	Adaptive	Presentation	Yes	Page layout reformat	2	Medium- High	
LJO	LJ90c	Scroll button colour change to green	Scroll button colour change to green	Scroll button colour style edit	Style issue	Adaptive	Presentation	No	None	2	Medium- High	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
LJO	LJ91a	Scroll button to be in line with left margin of text	Scroll button to be in line with left margin of text	Page layout style	Style issue	Corrective	Presentation	Yes	Page alignment	2	Medium- High	
LJO	LJ91b	Text content to be moved to secondary page of section	Text content to be moved to secondary page of section	Page layout style	Style issue	Adaptive	Presentation	Yes	Page layout reformat	2	Medium- High	
LJO	LJ91c	Scroll button colour change to green	Scroll button colour change to green	Scroll button colour style edit	Style issue	Adaptive	Presentation	No	None	2	Medium- High	
LJO	LJ92a	Scroll button to be in line with left margin of text	Scroll button to be in line with left margin of text	Page layout style	Style issue	Corrective	Presentation	Yes	Page alignment	2	Medium- High	
LJO	LJ92b	Text content to be moved to secondary page of section	Text content to be moved to secondary page of section	Page layout style	Style issue	Adaptive	Presentation	Yes	Page layout reformat	2	Medium- High	
LJO	LJ92c	Scroll button colour change to green	Scroll button colour change to green	Scroll button colour style edit	Style issue	Adaptive	Presentation	No	None	2	Medium- High	
LJO	LJ93	Cannot enter Admin area - data source error	Cannot enter Admin area - data source error	Admin access function error	Admin function	Corrective	System	Yes	Database access rights and call functions	7	High	
ABN	ABN1	New top navigation	New top navigation	Navigational text links	Textual navigation hyperlink	Perfective	System	Yes	Several top level navigation links	2	Medium- High	
ABN	ABN2a	Questionnaire response page content	Questionnaire response page content	Text content format	TUI content format	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN2b	Questionnaire response validation page functionality	Questionnaire response validation page functionality	Questionnaire functionality	Function	Adaptive	System	Yes	Core functionality that relevant across site	5	High	
ABN	ABN3	Animated graphic roll over function and text output	Animated graphic roll over function and text output	Animated graphic function - rollover	Function	Adaptive	System	Yes	Output relevant text content and highlight feature of selected choice	3	High	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
ABN	ABN4	Database design & call functions	Database design & call functions	Data structure & function	Function	Adaptive	Data	Yes	Call functions and data population	5	High	Database call function
ABN	ABN5	New graph in PDF	Database entry	Update field	Database data	Adaptive	Data	Yes	Query amendments	5	High	
ABN	ABN6	New introduction page required for global site	New introduction page required for global site	Dynamic introduction page function	Function	Perfective	Presentation	Yes	Flash features & content	5	High	
ABN	ABN7	Java-script page error	Java-script page error	Code error report	Function	Corrective	System	Yes	Code system calls	2	Medium- High	
ABN	ABN8	Image alteration	Image alteration	Image edit	Image	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN9	Text Navigational link to PDF files	Text Navigational link to PDF files	Text navigational link	Textual navigation hyperlink	Perfective	System	Yes	Navigational path and PDF data storage	1	Medium- Low	
ABN	ABN10	Change page number	Change page number	Numeric content edit	TUI content	Corrective	Presentation	No	None	1	Medium- Low	
ABN	ABN11	Wealth projection section text and image content	Wealth projection section text and image content	Page content - text and images layout	Style issue	Perfective	Presentation	Yes	Secondary page features and navigational link to	2	Medium- High	
ABN	ABN12	Performance graph images	Performance graph images	Static graph images	Image	Perfective	Presentation	No	None	4	High	
ABN	ABN13	Wealth production graph image production	Wealth production graph image production	Static graph images	Image	Perfective	Presentation	No	None	4	High	
ABN	ABN14	Review business scope of application	Business logic review	Business logic	Business Analysis	Perfective	System	No	None	10	High	Reviewing web application to be integrated with other existing client organisation sub-systems. Comparison with competitors.
ABN	ABN15	Separate content into two pages	Separate content into two pages	Page content format	Style issue	Perfective	Presentation	Yes	Secondary window feature	3	High	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
ABN	ABN16	Graphical text navigational link phase change	Graphical text navigational link phase change	Graphical text navigational link edit	Textual navigation hyperlink	Adaptive	Presentation	Yes	6 instances	1	Medium- Low	
ABN	ABN17	Create questionnaires for fixed product section	Create questionnaires for fixed product section	Questionnaire functionality	Function	Adaptive	System	Yes	Input choice features & content	6	High	
ABN	ABN18	Save sample selection button required	Save sample selection button required	Save button function	Function	Perfective	System	Yes	Data storage and retrieve function	2	Medium- High	
ABN	ABN19	Miss-clicking questionnaire responses error reports	Miss-clicking questionnaire responses error reports	Error reporting function	Function	Corrective	System	Yes	For all questionnaires	6	High	
ABN	ABN20	Remove text content from page	Remove text content from page	Text content edit	TUI content	Adaptive	Presentation	Yes	Page alignment	0.5	Low	
ABN	ABN21	Proceed to contract button required	Proceed to contract button required	Proceed button function	Function	Perfective	System	Yes	Data processing function	3	High	
ABN	ABN22	Add sample portfolio content	Add sample portfolio content	Text content addition	TUI content	Perfective	Presentation	Yes	Page alignment	0.5	Low	
ABN	ABN23	Result output of choice function - 60 possibilities based on selection of pre-defined choices	Result output of choice function - 60 possibilities based on selection of pre-defined choices	Choice function	Function	Perfective	System	Yes	Text and relevant image (graph)	10	High	New Req
ABN	ABN24	Result output of choice function - 6 possibilities based on selection of pre-defined choices	Result output of choice function - 6 possibilities based on selection of pre-defined choices	Automated choice function	Function	Perfective	System	Yes	Text and relevant image (graph)	7	High	New Req

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
ABN	ABN25	New text navigational link to disclaimer	New text navigational link to disclaimer	Text navigational link	Textual navigation hyperlink	Perfective	System	Yes	Navigational path to relevant content	1	Medium- Low	
ABN	ABN26	Introduction flash animation	Introduction flash animation	Animation function	Textual navigation hyperlink	Perfective	System	Yes	Image and graphic files production	3	High	New Req
ABN	ABN27	Questionnaire style change	Questionnaire style change	Page style format	Style issue	Perfective	Presentation	Yes	Questionnaire validation process	4	High	New Req
ABN	ABN28	Add and Save button for questionnaire	Add and Save button for questionnaire	Button function	Function	Perfective	System	Yes	Page alignment & button design elements	2	Medium- High	New Req
ABN	ABN29	New validate button on particular page	New validate button on particular page	Button function	Function	Perfective	System	Yes	Page alignment & button design elements	2	Medium- High	New Req
ABN	ABN30	Roll-over function over graphic to display correct text content	Roll-over function over graphic to display correct text content	Roll-over function	Function	Perfective	System	Yes	Relevant text content	8	High	Alteration
ABN	ABN31	Remove text content from roll-over feature of 'selecting the appropriate service'	Remove text content from roll-over feature of 'selecting the appropriate service'	Remove functional process for particular data roll-over	Function	Adaptive	System	Yes	Remove text content	5	High	Alteration
ABN	ABN32	Increase size of graphic	Increase size of graphic	Image edit	Image	Adaptive	Presentation	No	None	1	Medium- Low	Alteration
ABN	ABN33	Increase size of graphic	Increase size of graphic	Image edit	Image	Adaptive	Presentation	No	None	1	Medium- Low	Alteration
ABN	ABN34	Remove text	Remove text	Text content edit	TUI content	Adaptive	Presentation	No	None	0.5	Low	Alteration
ABN	ABN35	Include new graph	Include new graph	Graph image	Image	Adaptive	Presentation	No	None	1	Medium- Low	Alteration

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
ABN	ABN36	Translation Text error.	Translation Text error.	Text content edit	TUI content	Corrective	Presentation	No	None	0.25	Low	Alteration
ABN	ABN37	Increase text content size	Increase text content size	Text content format	TUI content format	Adaptive	Presentation	No	None	0.5	Low	Alteration
ABN	ABN38	Text graphic content change	Text graphic content change	Text image graphic edit	Image	Adaptive	Presentation	No	None	1	Medium- Low	Alteration
ABN	ABN39	Delete text content	Delete text content	Text content edit	TUI content	Adaptive	Presentation	No	None	0.25	Low	Alteration
ABN	ABN40	Increase pop-up (secondary window) size	Increase pop-up (secondary window) size	Pop-up (secondary window) format	Style issue	Adaptive	System	No	None	2	Medium- High	Alteration
ABN	ABN41	Change title	Change title	Text image content edit	Image	Adaptive	Presentation	No	None	1	Medium- Low	Alteration
ABN	ABN42a	Align arrows	Align arrows	Image format	Image	Adaptive	Presentation	No	None	1	Medium- Low	Alteration
ABN	ABN42b	Increase pop-up (secondary window) size	Increase pop-up (secondary window) size	Pop-up (secondary window) format	Style issue	Adaptive	System	No	None	2	Medium- High	Alteration
ABN	ABN43	Include text content details	Include text content details	Text navigation link edit	Textual navigation hyperlink	Adaptive	Presentation	No	None	1	Medium- Low	Alteration
ABN	ABN44	Page colour style format	Page colour style format	Page style format	Style issue	Adaptive	Presentation	No	None	1.5	Medium- High	Alteration
ABN	ABN45	Error message displayed when submitting questionnaire	Error message displayed when submitting questionnaire	Investment calculation function	Function	Corrective	System	Yes	Output messages	3	High	Alteration
ABN	ABN46	Include text navigational link	Include text navigational link	Text navigational link	Textual navigation hyperlink	Perfective	System	No	None	1	Medium- Low	Alteration
ABN	ABN47	Add text disclaimer content	Add text disclaimer content	Text content	TUI content	Perfective	Presentation	No	None	0.5	Low	Alteration
ABN	ABN48	Add terms and conditions link	Add terms and conditions link	Text navigational link	Textual navigation hyperlink	Perfective	System	No	None	1	Medium- Low	Alteration

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
ABN	ABN49	Clickable clear function required for pop-up (secondary window)	Clickable clear function required for pop-up (secondary window)	Clear' function	Function	Perfective	System	No	None	5	High	New req
ABN	ABN50	Translation Text error.	Translation Text error.	Text content edit	TUI content	Corrective	Presentation	No	None	0.25	Low	Alteration
ABN	ABN51a	Questionnaire print button function	Questionnaire print button function	Print button function	Function	Perfective	System	Yes	Input button & data structure	2	Medium- High	New Req
ABN	ABN51b	Scroll bars facility for questionnaire	Scroll bars facility for questionnaire	Scroll bar	Scrolling device	Perfective	System	No	None	2	Medium- High	New Req
ABN	ABN52	Questionnaire function to change answer & resubmit	Questionnaire function to change answer & resubmit	Questionnaire modify function	Function	Perfective	System	Yes	Data structure data calls	4	High	New Req
ABN	ABN53	Outcome page from questionnaire to have a print button	Outcome page from questionnaire to have a print button	Print button function	Function	Perfective	System	Yes	Print button & position	4	High	New Req
ABN	ABN54	Graphical text change	Graphical text change	Text image content edit	Image	Perfective	Presentation	No	None	1	Medium- Low	Alteration
ABN	ABN55	Risk graph addition as output option for selection	Risk graph addition as output option for selection	Image graph	Image	Perfective	Presentation	Yes	Page alignment	2	Medium- High	Alteration
ABN	ABN56	Graph image size change	Graph image size change	Image graph format	Image	Adaptive	Presentation	No	None	1	Medium- Low	Alteration
ABN	ABN57	Graphical image to output correct pop-up	Graphical image to output correct pop-up	Text output function for image	Function	Perfective	System	No	None	3	High	
ABN	ABN58	Translation Text error.	Translation Text error.	Text content edit	TUI content	Corrective	Presentation	No	None	0.25	Low	
ABN	ABN59	Text content change	Text content change	Text content edit	TUI content	Adaptive	Presentation	No	None	0.25	Low	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
ABN	ABN60	Text navigational link	Text navigational link	Text navigational link	Textual navigation hyperlink	Perfective	System	No	None	1	Medium- Low	
ABN	ABN61	Change details on graph image	Change details on graph image	Image graph edit	Image	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN62	Text colour change	Text colour change	Text content format	TUI content format	Adaptive	Presentation	No	None	0.5	Low	
ABN	ABN63a	Edit graph image	Edit graph image	Image graph edit	Image	Adaptive	Presentation	No	None	2	Medium- High	
ABN	ABN63b	New text content for graph	New text content for graph	Text content	TUI content	Perfective	Presentation	No	None	1	Medium- Low	
ABN	ABN64	Disclaimer to attach itself to printouts	Disclaimer to attach itself to printouts	Printout function	Function	Perfective	System	Yes	Relevant details to capture and printout	3	High	
ABN	ABN65	New table data in PDF	Database entry	Update field	Database data	Perfective	Data	Yes	Query amendments	4	High	
ABN	ABN66	New graphs in PDF	Database entry	Update field	Database data	Perfective	Data	Yes	Query amendments	4	High	
ABN	ABN67	Remove graph image	Remove graph image	Image graph edit	Image	Perfective	Presentation	No	None	1	Medium- Low	
ABN	ABN68	Text content change	Text content change	Text content edit	TUI content	Adaptive	Presentation	No	None	0.25	Low	
ABN	ABN69	Edit image - colour	Edit image - colour	Image graph edit	Image	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN70	Include new graph	Include new graph	Graph image	Image	Adaptive	Presentation	No	None	2	Medium- High	Alteration
ABN	ABN71a	Edit graph image	Edit graph image	Image graph edit	Image	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN71b	Include navigational text links to graph	Include navigational text links to graph	Text navigational link	Textual navigation hyperlink	Perfective	System	No	None	1	Medium- Low	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
ABN	ABN72	Edit graph image	Edit graph image	Image graph edit	Image	Adaptive	Presentation	Yes	36 instances	1	Medium- Low	
ABN	ABN73	Translation Text error.	Translation Text error.	Text content edit	TUI content	Corrective	Presentation	No	None	0.25	Low	
ABN	ABN74	Text navigational link	Text navigational link	Text navigational link	Textual navigation hyperlink	Perfective	System	Yes	Four instances	1	Medium- Low	
ABN	ABN75	Text content change	Text content change	Text content edit	TUI content	Adaptive	Presentation	No	None	0.25	Low	
ABN	ABN76	Graph to be displayed in secondary window and size changed	Graph to be displayed in secondary window and size changed	Display function	Function	Perfective	System	Yes	Edit graph output dimensions	6	High	
ABN	ABN77	Text content change	Text content change	Text content edit	TUI content	Adaptive	Presentation	No	None	0.25	Low	
ABN	ABN78	Graph to be displayed in secondary window	Graph to be displayed in secondary window	Display function	Function	Perfective	System	No	None	6	High	
ABN	ABN79	Text content change	Text content change	Text content edit	TUI content	Adaptive	Presentation	No	None	0.25	Low	
ABN	ABN80	Graph to be displayed in secondary window	Graph to be displayed in secondary window	Display function	Function	Perfective	System	No	None	6	High	
ABN	ABN81	Text content change	Text content change	Text content edit	TUI content	Adaptive	Presentation	No	None	0.25	Low	
ABN	ABN82	Image window to be able to scroll	Image window to be able to scroll	Navigational scroll function	Scrolling device	Perfective	System	No	None	3	High	
ABN	ABN83	Update PDF data	Database entry	Update field	Database data	Adaptive	Data	No	None	4	High	
ABN	ABN84	Update PDF data	Database entry	Update field	Database data	Adaptive	Data	No	None	3	High	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
ABN	ABN85	Update PDF data	Database entry	Update field	Database data	Adaptive	Data	No	None	3	High	
ABN	ABN86	Update PDF data	Database entry	Update field	Database data	Adaptive	Data	No	None	3	High	
ABN	ABN87	Update PDF data	Database entry	Update field	Database data	Adaptive	Data	No	None	3	High	
ABN	ABN88	New line image	New line image	Image	Image	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN89	Questionnaire output text return	Questionnaire output text return	Text content	TUI content	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN90	Questionnaire output text return	Questionnaire output text return	Text content	TUI content	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN91	Questionnaire output text return	Questionnaire output text return	Text content	TUI content	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN92	Incorrect currency	Incorrect currency	Edit text content	TUI content	Corrective	Presentation	No	None	0.25	Low	Different currency
ABN	ABN93	Text navigational links	Text navigational links	Text navigation link	Textual navigation hyperlink	Adaptive	System	No	None	1	Medium- Low	
ABN	ABN94	Questionnaire output text return	Questionnaire output text return	Text content	TUI content	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN95	Format numeric content	Format numeric content	Format numeric content	TUI content	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN96	Create new information page	Create new information page	Web page	Style issue	Adaptive	System	Yes	Page content	3	High	
ABN	ABN97	New graphs navigational links	New graphs navigational links	Navigation text links	Textual navigation hyperlink	Adaptive	System	Yes	6 instances	1	Medium- Low	
ABN	ABN98	Questionnaire output text return	Questionnaire output text return	Text content	TUI content	Adaptive	Presentation	No	None	1	Medium- Low	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
ABN	ABN99	Add new save sample button	Add new save sample button	Save button function	Function	Perfective	System	Yes	Data storage and retrieve function	2	Medium- High	
ABN	ABN100	Miss-clicking responses	Miss-clicking responses	Error handling function	Function	Perfective	System	No	None	3	High	
ABN	ABN101	Translation Text error.	Translation Text error.	Edit text content	TUI content	Corrective	Presentation	No	None	0.25	Low	
ABN	ABN102	Add new button	Add new button	Proceed button function	Function	Perfective	System	Yes	Data storage and retrieve function	2	Medium- High	
ABN	ABN103	Image link	Image link function	Image navigation link	Multimedia hyperlink	Adaptive	System	No	None	1	Medium- Low	
ABN	ABN104	Remove image	Remove image	Edit image	Image	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN105	E-mail response function	E-mail response function	E-mail response function	Function	Adaptive	System	No	None	10	High	
ABN	ABN106	Page print dimensions	Page print dimensions	Print function	Function	Adaptive	System	No	None	3	High	
ABN	ABN107	Page print dimensions	Page print dimensions	Print function	Function	Adaptive	System	No	None	3	High	
ABN	ABN108	New portfolios navigation links	New portfolios navigation links	Text navigation link	Textual navigation hyperlink	Adaptive	System	Yes	Data file path	1	Medium- Low	To data file
ABN	ABN109	Remove numeric content	Remove numeric content	Edit numeric content	TUI content	Adaptive	Presentation	No	None	0.5	Low	
ABN	ABN110	Update PDF data	Database entry	Update field	Database data	Adaptive	Data	No	None	4	High	
ABN	ABN111	Change style of terms and conditions content	Change style of terms and conditions content	Format text content	TUI content format	Adaptive	Presentation	No	None	0.5	Low	
ABN	ABN112	Include graph image in GIF folder	Include graph image in GIF folder	Graph image	Image	Adaptive	Presentation	No	None	2	Medium- High	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
ABN	ABN113	Include graph image in GIF folder	Include graph image in GIF folder	Graph image	Image	Adaptive	Presentation	No	None	2	Medium- High	
ABN	ABN114	Include new text content - disclaimer	Include new text content - disclaimer	Text Content	TUI content	Adaptive	Presentation	No	None	0.5	Low	
ABN	ABN115	Translation Text error.	Translation Text error.	Text content edit	TUI content	Corrective	Presentation	No	None	0.25	Low	
ABN	ABN116a	Change layout style of percentage output box	Change layout style of percentage output box	Format output box	Style issue	Adaptive	Presentation	No	None	4	High	
ABN	ABN116b	Change percentage content to 10	Change percentage content to 10	Edit numeric content	TUI content	Adaptive	Presentation	No	None	0.5	Low	
ABN	ABN117	Reporting page footnote content style change	Reporting page footnote content style change	Format style content	Style issue	Adaptive	Presentation	No	None	2	Medium- High	
ABN	ABN118	Change graph image	Change graph image	Edit graph image	Image	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN119	Change navigation label	Change navigation label	Edit text navigation link	Textual navigation hyperlink	Adaptive	Presentation	No	None	0.5	Low	Pres as text label change
ABN	ABN120	New graph images	New graph images	Graph image	Image	Adaptive	Presentation	No	None	2	Medium- High	
ABN	ABN121	Format text content on site map	Format text content on site map	Format text content	TUI content format	Adaptive	Presentation	No	None	1.5	Medium- High	
ABN	ABN122	Choice of language to define view and print outputs	Choice of language to define view and print outputs	Language function	Function	Adaptive	System	Yes	Output view & print	12	High	
ABN	ABN123	Add product brochure files to link pages	Add product brochure files to link pages	Navigational text link	Textual navigation hyperlink	Adaptive	System	Yes	Link path	2	Medium- High	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
ABN	ABN124	Pie chart image as return from questionnaire answers	Pie chart image as return from questionnaire answers	Graph image	Image	Corrective	Presentation	No	None	4	High	
ABN	ABN125	Edit text content in EOC	Edit text content in EOC	Edit text content	TUI content	Corrective	Presentation	No	None	0.25	Low	
ABN	ABN126	New icons for save, print and save & proceed	New icons for save, print and save & proceed	Image	Image	Adaptive	Presentation	No	None	2	Medium- High	
ABN	ABN127	Title change	Title change	Text content edit	TUI content	Adaptive	Presentation	No	None	0.25	Low	
ABN	ABN128	Make text bigger and more readable	Make text bigger and more readable	Format text content	TUI content format	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN129	Graphs to increase in size due to readability issues	Graphs to increase in size due to readability issues	Image format	Image	Adaptive	Presentation	No	None	3	High	
ABN	ABN130	Slides to increase in size due to readability issues	Slides to increase in size due to readability issues	Format slides	TUI content format	Adaptive	Presentation	No	None	2	Medium- High	
ABN	ABN131	Global site body text size reduction	Global site body text size reduction	Text content format	TUI content format	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN132	Calculation to work out probability	Calculation to work out probability	Probability Calculation function	Function	Perfective	System	Yes	Data source	8	High	
ABN	ABN133	Shorten line length throughout body text	Shorten line length throughout body text	Image format	Image	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN134	Large left margin on site page	Large left margin on site page	Page style format	Style issue	Adaptive	Presentation	No	None	2	Medium- High	
ABN	ABN135	Change text title to upper case	Change text title to upper case	Text content format	TUI content format	Adaptive	Presentation	No	None	0.25	Low	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
ABN	ABN136	Site map text title to be upper case	Site map text title to be upper case	Text content format	TUI content format	Adaptive	Presentation	No	None	0.25	Low	
ABN	ABN137	Disparities in top navigation style	Disparities in top navigation style	Page layout style	Style issue	Adaptive	Presentation	Yes	Global page layout conformance	2	Medium- High	
ABN	ABN138	Move numerical page numbering up page	Move numerical page numbering up page	Page layout style	Style issue	Adaptive	Presentation	Yes	Two instances	3	High	
ABN	ABN139	Prevent scrolling on two particular graph	Prevent scrolling on two particular graph	Image navigation scrolling	Scrolling device	Adaptive	Presentation	Yes	Two instances	1.5	Medium- High	
ABN	ABN140	Graph design does not fit layout of page	Graph design does not fit layout of page	Image edit	Image	Adaptive	Presentation	Yes	Two instances	3	High	
ABN	ABN141	Remove corner bit from flyer image	Remove corner bit from flyer image	Image edit	Image	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN142	Link style for sample portfolios	Link style for sample portfolios	Navigational text link	Textual navigation hyperlink	Corrective	System	Yes	Navigation path to data file	2	Medium- High	
ABN	ABN143	Link style for sample portfolios	Link style for sample portfolios	Navigational text link	Textual navigation hyperlink	Corrective	System	Yes	Navigation path to data file	1	Medium- Low	
ABN	ABN144	Missing navigation links in on page when accessed via site map	Missing navigation links in on page when accessed via site map	Navigational text link	Textual navigation hyperlink	Corrective	System	Yes	Navigation path to data file	1.5	Medium- High	
ABN	ABN145	Investment fund calculation function to have limit	Investment fund calculation function to have limit	Exception handling function	Function	Perfective	System	No	None	8	High	
ABN	ABN146	Add hyphens to menu	Add hyphens to menu	Text content edit	TUI content	Adaptive	Presentation	No	None	1	Medium- Low	

Case Study Project	Change Request I.D	Change Request Detail	Impacted WIS Area (WS supplier view)	Base Unit Area of Change (Researcher)	WIS Change Group Category	Form of Change	Impacted Structural Layer	Ripple Effect	Ripple Effect Details	Effort (Hours)	Effort Rating	Comment/ Notes
ABN	ABN147	Page numbering to be made bigger	Page numbering to be made bigger	Text content format	TUI content format	Adaptive	Presentation	No	None	0.5	Low	
ABN	ABN148	Replace questions 7 & 8 in questionnaire	Replace questions 7 & 8 in questionnaire	Text content edit	TUI content	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN149	Manual questionnaire admin review process	Manual questionnaire admin review process	Review function	Function	Perfective	System	Yes	Database facility for storage	4	High	
ABN	ABN150	Move question 9 in questionnaire to relevant section	Move question 9 in questionnaire to relevant section	Text content format	TUI content format	Adaptive	Presentation	No	None	1	Medium- Low	
ABN	ABN151	Combine 2 introduction pages into 1 page	Combine 2 introduction pages into 1 page	Page content format	TUI content format	Adaptive	Presentation	Yes	Page alignment	1	Medium- Low	
ABN	ABN152	Page replacement	Page replacement	Page content edit	TUI content	Adaptive	Presentation	Yes	Page alignment	1	Medium- Low	
ABN	ABN153	Add two new portfolios	Add two new portfolios	Image	Image	Adaptive	Presentation	No	None	1.5	Medium- High	
ABN	ABN154	Yellow option boxes to be made clickable	Yellow option boxes to be made clickable	Image input function	Function	Perfective	System	No	None	3	High	
ABN	ABN155	Update PDF data	Database entry	Update field	Database data	Adaptive	Data	No	None	4	High	
ABN	ABN156	New wealth projection graph	Graph image	Image	Image	Adaptive	Presentation	No	None	4	High	
ABN	ABN157	Add new links	Navigational links	Navigational text links	Textual navigation hyperlink	Perfective	System	No	None	2	Medium- High	

Appendix D: Cumulative frequency graph

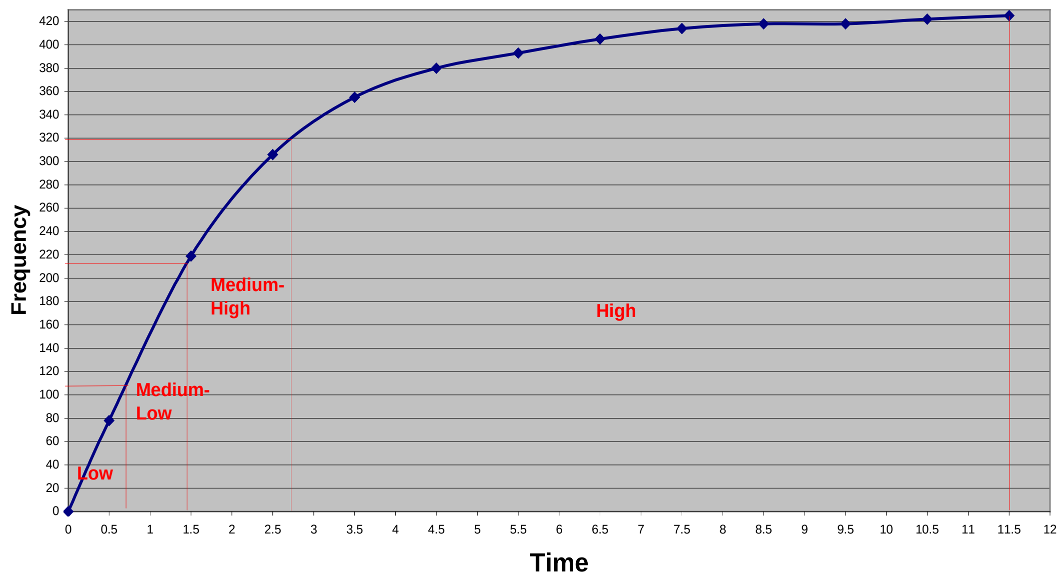

Figure D.1 – Cumulative Frequency Graph

Appendix E1-E3:
Analysis of AoC per project
(Notes)

Appendix E1 Analysis: Project Case 1 (P1) - WFO

The total number of individual change requests collected and analysed over a three month period for the P1 project amounted to the sum of 134. The sample represents a variety of post-development change requests within the boundaries of the maintenance phase of the project lifecycle that were subsequently implemented into the system (WIS). Figure E.1 illustrates a bar graph that summarises the results of the categorisation and clustering process (as described in Chapter 4) into group categories of which a total of eleven categories emerged. These findings indicate that the main aspects of the system experiencing the highest frequency of change are the group categories of Function (25.37%), Textual Navigation Hyperlink (23.88%) and TUI Content (23.13%) that together represent above two-thirds of overall change requests for the P1 project case and secondly are similar in size statistically. The remaining categories collectively represent 27.61% of the total change requests of which 10.45% are relating back to an Image (7.46%) or Style Issue (2.99%), another 7.46% are pertaining to navigation or administration functional processes (Navigation Function – 3.73% and Admin Function – 3.73%). The distributions of the remaining changes are relatively small with the categories of Database Data, Multimedia Hyperlink and Scrolling Device each showing a contribution of less than or equal to 2.24% and lastly 5.22% for the category TUI Content Format. However when considered individually the remaining eight categories did not breach a 10% threshold and thus can be considered as having a minimal influence in terms of change on the system, nevertheless the effort involved with implementing these changes could out-weight its small frequency factor.

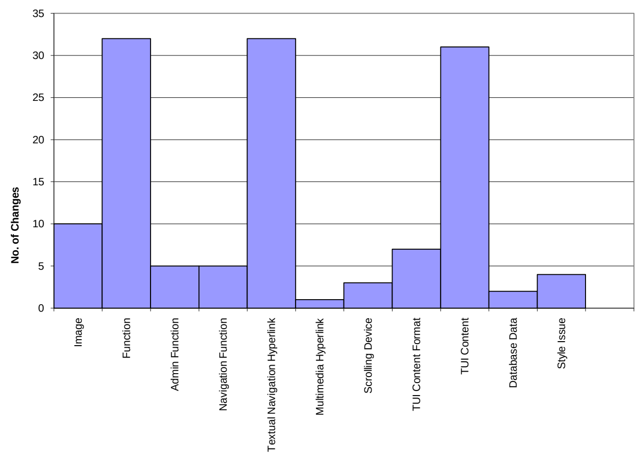

Figure E.1 – Distribution of WFO (P1) group categories

The location of the change requests with reference to the systems architecture layers for the WFO project case are shown as bar-graph totals in Figure E.2. The frequency findings indicate that over half (56.72%) of the change requests influenced the system on the Presentation layer and that a further 41.79% of changes affected the System layer. The remaining 1.49% of changes equally impacted the Data layer. The vast majority of changes within the Presentation layer follow the sequence from the least to most and are accredited to the group categories of Image 13% (10/76), Navigation Text 25% (19/76) and Alphanumeric Content 41% (31/76). Of the remaining nine categories, six of them (namely: Function, Admin Function, Navigation Function, Navigation Image, Data) had a 1% or less of a contribution to the layer with three group categories of Scrolling Device (3/76), Style Issue (4/76) and Text Format (7/76) having a 4%, 5% and 9% affect on the system, respectively. On further investigation of the Application/System layer only two change group categories reflect substantial figures, that of Function 59.3% (32/54) and Navigation Text 24%

(13/54). With the exception of the group categories of Admin Function (9.3% - 5/54) and Navigation Function (7.4% - 4/54) the remaining eight categories had a 0% influence on the Application/System layer. The findings of the architecture layers are shown in Figure E.2 as frequency totals.

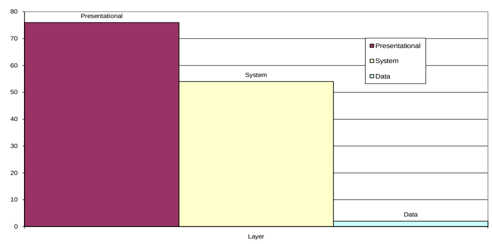

Figure E.2 – Distribution of WFO (P1) change requests across structural layers

The WFO project case illuminated an almost even split between changes that did and did-not experience ripple effects. The findings specifically shows that 47% (63/134) of change request did experience some sort of knock-on effect and that 53% (71/134) of the changes were isolated to a particular part of the system. This tends to indicate that either the changes were specific to a singular feature of the system or that system was designed well enough to separate or even up-date the effects of a change to any other relevant part of the system (through technology or design). Figure 5.6 summarises the relationship in totals of ripple against no ripple on a per group category basis. Of the 47% total that did have a ripple, almost two-thirds (64%) of the sum changes that had a ripple effect were pertaining to the Functions 29% (18/63) and Navigation Text 35% (22/63) group categories. The remaining 17% that had a ripple was distributed amongst eight of the remaining categories with all of them showing a less than 10% influence on the overall system. The group categories of Navigation Image and MISC (miscellaneous) were the only categories that did not experience any ripple effects from the induced changes, although there frequencies were minimal overall. Of the 53% of change requests that did not have a ripple effect 27/71 came from the category Alphanumeric Content, 10/71 was from Navigation Text and 14/71 were relating to the Functions category. Thus over two-thirds (38% from the overall 54%) that did not have a ripple were from the above mentioned categories. The remaining 20/71 that did not have a ripple of all the collective categories were evenly spread against the remaining group categories and were negligible when considered individually. Again all of these categories showed a less than 10% influence on the overall system in terms of not having a ripple. However the Alphanumeric Content category is clearly identified as a category that had a far greater frequency of no ripples to ripples. Also it is interesting to note that with the exception of the group categories of Navigation Text, Navigation Image, Alphanumeric Content the ratio distributions per group category of having and not having a ripple are very similar.

When considering the form of the change Figure 5.7 illustrates the frequency of occurrence in percentages of the traditional category types. The findings show that three out of the five category types depicted the nature of the change requests that arose. Corrective type change requests were 40% (54/134), Perfective types were 22% (30/134) and the remaining category of Adaptive was collectively only 9% (12/134). Just under a three-quarter (74.1% of the 40%) of the Corrective type changes was from the group categories of Function (12/54), Navigation Text (14/54) and Alphanumeric Content (14/54) the remaining group categories when merged together (as they were considered negligible at a individual level) represented 25.9% (14/54) of the corrective change requests. It is however noted here that just under half of WFO project changes had a large amount of changes that were corrective in nature (54/134).

When considering this category as a whole it is noticeable that most of the higher instances (Alphanumeric Content, Text Format and Navigation Text) are from categories relating to some visual or content aspect of the system. The Perfective maintenance type changes fell into two main categories, that of Functions (13/30) and Navigation Text (12/30). The WF project was inherently a complex large site that required functional and navigational additions within the maintenance phase.

Appendix E2 Analysis: Project Case 2 (P2) – ABN

Individual change requests in the LJO project collectively amount to the sum total of 128, which spanned a post-development period of three months. Eight change request categories emerge that fall within the parameters of the maintenance phase of the project lifecycle – these were subsequently assimilated into the system (WIS).

The findings infer that the areas of the system in which the highest frequency of change is occurring, are the group categories of Alpha Numeric Content that constitutes 30% (38/128) and Style Issue constituting 27% (34/128). Together they make up just over a half of all change requests for the LJO case project and are also similar in size statistically. The third highest category at 12% (15/128) is Image. The remaining five categories jointly stand for 32% of the total change requests of which the categories of Function, Navigation Text and Text Format all equate to 7%, while 9% can be attributed to Admin Functions. The category Misc only represents 2% of change. Of the 32% of change, none of the categories therein surpass a 10% threshold and from this it can be deduced that their influence is negligible in terms of change on the system. However, the effort deployed in the implementation of a particular change can perhaps counterbalance or indeed far outweigh its minimal frequency factor.

Analysis of the systems architecture layer for the LJO case, and as seen in the bar graph (Figure 2c), demonstrates that two-thirds of change requests are affecting the system at the Presentation layer 66% (84/128), while a further 14% of changes affected the application on the Systems layer (18/128). The remaining changes affected the Data layers 10% (13/128). The majority of the changes prevalent in the Presentation layer of the WIS resonated from the group categories of Image 18% (15/84), Text Format 11% (9/84), Alphanumeric Content 29% (24/84) and Style Issue 38% (32/84). These group categories are components of WIS GUI's and thus exist in the Presentation layer. Within the Systems layer two main categories of change appear that compose most of the changes, namely - Function 38% (7/18) and

Navigation Text 33% (6/18). No change requests can be seen to have arisen from the categories of Image, Text Format or Style Issue here. There is a noticeable absence of change requests emerging from five categories of Image, Function, Navigation Text, Text Format within this layer. Administrator functionalities often relate back to the CMS such that users are provided with functionality that can be used to manipulate aspects of the web information system. The CMS provided for the LJ system was basic and thus new content managed section and administrator access was required which infers that the business logic was changing and was thus represented at the Business layer. A prominent category within the Data layer was Alpha Numeric Content at 85% (11/13). This can be explainable as the CMS as LJ was limited in its capacity to handle Alphanumeric Content changes. Any textual content that are based within the CMS will ultimately effect the data layer of the application as content is stored as data objects within the CMS database. The Data layer denotes negligible change or no change at all, in all categories with the exception of Alpha Numeric Content, which attributes 85% of change request occurring at this layer (11/13).

A large proportion, specifically 71% (just under three-quarters) of change requests have a ripple effect on the system as opposed to the requests that do not have this effect. The group category of Alpha Numeric Content forms 30% of changes having an effect on other parts of the system, while Style Issue in a similar fashion, is also a significant contributing factor to rippling at 30% again (27/91). The remaining categories amount to 35% of which the categories: Image 10% (9/91), Function 7% (6/91), Admin Function 10% (9/91) and Navigation Text 9% (8/91) had overall ripples. 29% of the total change request did not experience a ripple, and came from the following categories: Alpha Numeric Content forms 32% (12/37), Style Issue 19% (7/37), Image 16% (6/37) and Text Format 14% (5/37). In the category of Misc rippling is wholly absent, not even making a 1% threshold. As can be seen in Figure 3b the category showing the highest non-occurrence of rippling is Alpha Numeric Content at 32% (12/37) and the second highest is Style Issue at 19% (7/37), the third highest being Image at 16% (6/37). The remaining five categories of Function, Admin

Function, Navigation Text, Text Format and Misc all comprise less than 10% of changes that did not induce rippling.

The highest maintenance category was Corrective that amounted to 22% (28/128) of overall changes. This maintenance type was mainly composed of the group categories of Style Issue 32% (9/28), Alphanumeric Content 18% (5/28) and Function 18% (5/28). The remaining four group categories collectively represented 32% of the total change requests for Corrective maintenance types with Navigation Text category having no corrective changes made to it. The next highest maintenance change type was Perfective 10% (13/128). The dominant group category within this maintenance type was Alphanumeric Content 54% (7/13), suggesting that textual additions to the web application was a key feature. 23% of Perfective type changes also occurred within the category group of Function (3/13). The maintenance type of Preventative represented 8% (10/128) of the total number of change requests. All ten instances only affected the category group of Admin Function. Finally the maintenance type of Adaptive had the least number of changes 2% (3/128) however it is worth noting that the only category group it had a impact on was Navigation Text and can be explainable by hyper-links responding to external changes outside the bounds of the system.

Appendix E3 Analysis: Project Case 3 (P3) – LJO

The categories consuming 10% or more changes are Image (20%), Function (23%), Navigation Text (12%) and Alphanumeric Content (18%) representing almost three-quarters of the overall changes studied within the project. The other seven categories collectively constituted to 28% of the total change requests.

The isolation of the change requests were further identified by localising the effect of the change onto a 3-tier system architecture framework. The results indicates that Presentational layer (58%) and the System layer (35%) had the most effects from the change requests implemented into the system. A substantial amount of changes that occurred at the Presentation level fell into the category groups of Image (33/94), Text Format (13/94), Alphanumeric Content (30/94) and Style Issue (9/94). These interface type modifications are often typified as the key components of a WIS and its virtue as a web based application. The composition of the structural layer System was namely from the group categories of Function (34/57) and Navigation Text (16/57). Navigational aspects of WIS's are included here as functional objects as they are a key (and unique) feature within of the logical process in the web paradigm. The Data layers are the lowest structural components influenced by change thus limited effort is required when considering its change capacity. Within the Data layer most of the changes were from the Data category (10/11). The ABN project did not have a implemented CMS and thus all the changes made directly affected this particular layer without any knock-on effects on other structural layers of the system.

A large portion of changes did not have a ripple effect within the ABN project (66% - 108/163). Further investigation into the types of changes that did have ripple showed that Function (26/55) and Navigation Text (11/55) consumed a large portion of the overall 34% (55/163) that did have a ripple. Slightly over two-thirds of the changes that did have a ripple relates back to either a Function or Navigation Text categories. These categories in most cases represent the system features of the application from algorithms computations to navigation structures and components. It is also noted that 10% (6/55) of changes had a ripple for the category Style Issue. Of the 66% (108/163) of changes that did not have a ripple effect just less than two-thirds related to one of the following new elements: Alphanumeric Content (27/108), Text Format (12/108) and Image (30/108) with Image and Alphanumeric Content still having a greater majority. The remaining eight categories within this project collectively represented 36% (39/108) of changes that did not experience a ripple effect.

The three highest traditional maintenance categories of the overall project were: Corrective 10% (16/163), Perfective 31% (50/163) and Other 57% (93/163). The Adaptive category represented 3% of the overall changes with the Preventative category producing no results. The ABN project was implemented under strict deadlines and restrictions thus only core functionality was produced which explains the higher percentages of Corrections, Perfections within the maintenance phase. This also helps to explain why minimal adaptive and no preventative changes took place as timescales did not allow for environmental changes to affect the system or that the capacity for preventative mechanisms would incur greater costs within the maintenance phase as no capacity was in place in the systems architecture e.g. no CMS structure and technologies used were a disabling factor (Technologies also disable the ease of implementing new features). Of the Corrective maintenance changes 50% (8/16) of them were related to the category Alphanumeric Content with most instances requiring some textual amendments been made or textual additions to the content. The remaining eight Corrective change types spanned the remaining group categories with the exception of Navigation Image, Navigation Scroll, Text Format, Data and Style Issue. Within the Perfective maintenance type two main categories emerged as the focus of these types of change, namely Functions 50% (25/50) and Navigation Text 18% (9/50).

Appendix F: Result tables for the Functions AoC group category

Project	Total no. per project	No. of Function change requests/ Total no. of change requests per project
P1	34	25.37% (34/134)
P2	36	22.09% (36/163)
P3	9	7.03% (9/128)

Table F.1 – Distribution of Function AoC across the projects

	Form					
AoC Group Category	Adaptive		Corrective		Perfective	
Function	18	22.78%	20	25.31%	41	51.90%

Table F.2 – Distribution of Form for Functions AoC

	Architecture Layer					
Group Category	Data		Presentation		System	
Function	(2/79)	2.53%	(2/79)	2.53%	(75/79)	94.94%

Table F.3 – Distribution of Function AoC across Structural Layers

Function AoC (per project)	Architecture Layer		
	Data	Presentation	System
P2	(1) 2.94%	(1) 2.94%	(34) 94.44%
P3	(1) 14.29%	(1) 14.29%	(7) 77.78%
P1	0	0	(34) 100%

Table F.4 – Distribution of Function AoC across Architecture Layers and projects

Case Study Project Aggregated				
	Ripple Effect			
Group Category	No		Yes	
Function	(29)	36.71%	(50)	63.29%

Table F.5 – Distribution of Function AoC and ripple effect

Architecture Layer	Ripple Yes	Ripple No
System Layer	(46/75) 61.33%	(29/75) 38.67%

Table F.6 – Distribution of Function AoC across Architecture Layers and ripple effect

Case Study Project Aggregated								
	Effort Rating							
Group Category	High		Medium- High		Medium-Low		Low	
Function	(60)	75.95%	(18)	22.78%	(1)	1.27%		0%

Table F.7 – Distribution of Function AoC and effort rating

Appendix G: The Matrix of Change

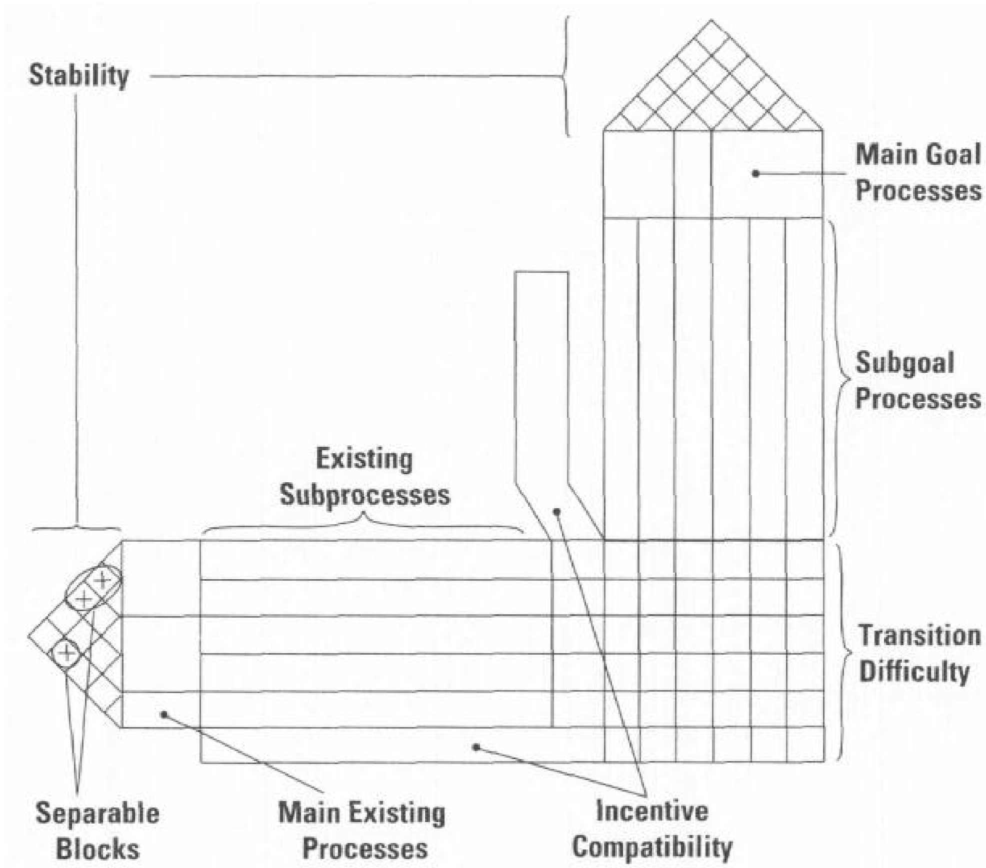

Figure G.1 – Functions of Matrix Graphics

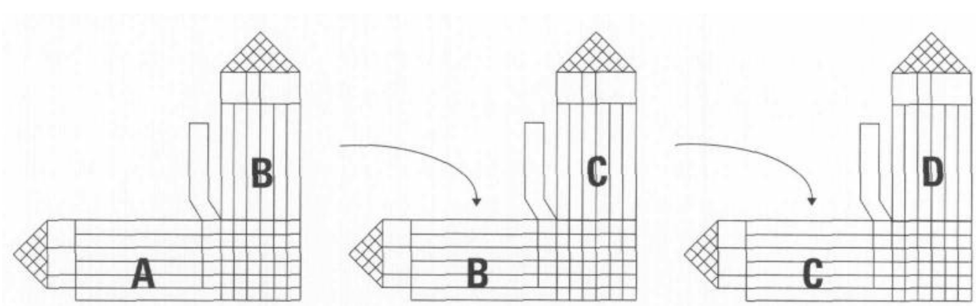

Figure G.2 – Linked Matrices

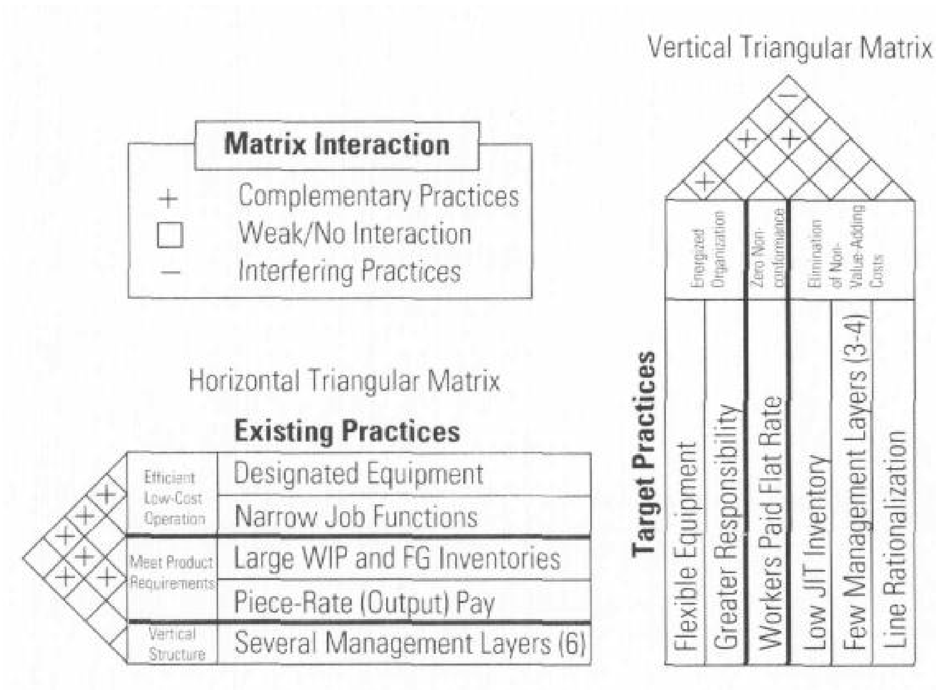

Figure G.3 – Horizontal and Vertical Matrices

Matrix Interaction

+ Complementary Practices
□ Weak/No Interaction
— Interfering Practices

	Target Practices	Energized Organization		Zero Non-conformance	Elimination of Non-Value-Adding Costs		
	Existing Practices	Flexible Equipment	Greater Responsibility	Workers Paid Flat Rate	Low JIT Inventory	Few Management Layers (3-4)	Line Rationalization
Efficient Low-Cost Operation	Designated Equipment	—	—				+
	Narrow Job Functions	—	—			—	+
Meet Product Requirements	Large WIP and FG Inventories	—			—		
	Piece-Rate (Output) Pay		—	—			
Vertical Structure	Several Management Layers (6)		—			—	

Figure G.4 – Transition Matrix

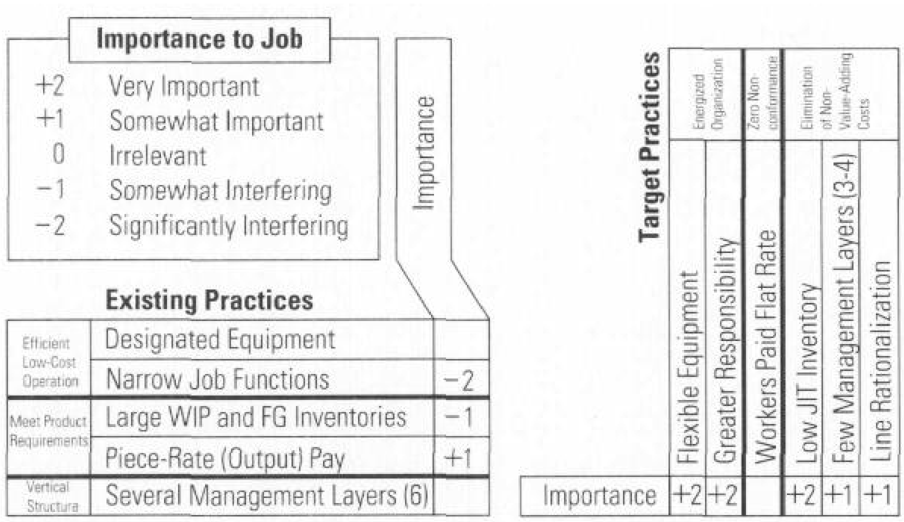

Figure G.5 – Satisfaction Ratings Matrix

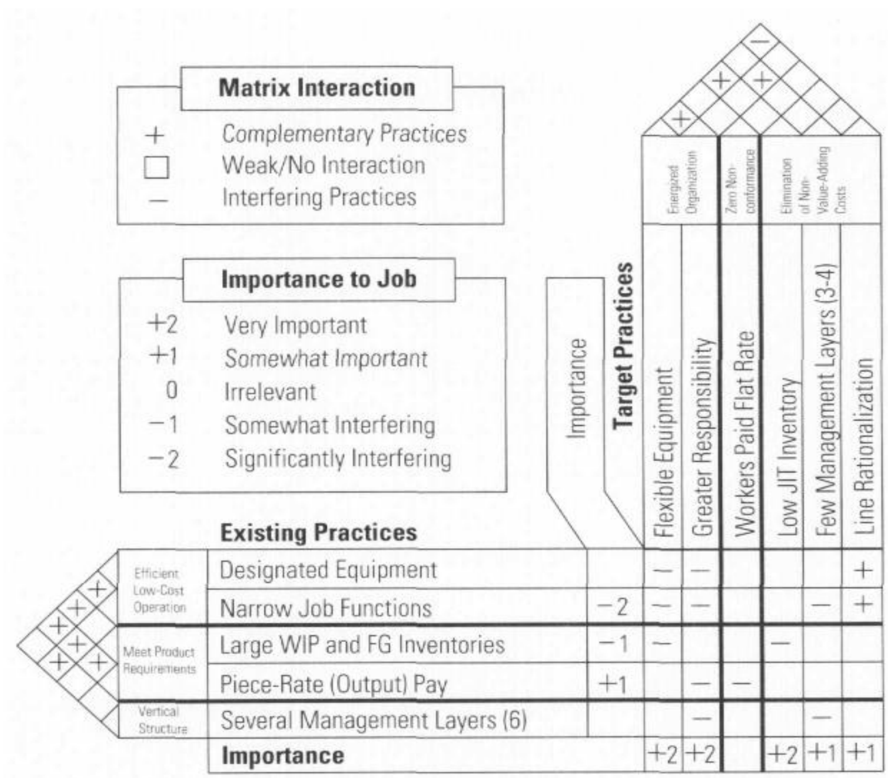

Figure G.6 – The Matrix of Change