

Copyright is owned by the Author of the thesis. Permission is given for a copy to be downloaded by an individual for the purpose of research and private study only. The thesis may not be reproduced elsewhere without the permission of the Author.

e-Process Selection using Decision Making Methods

A thesis presented in partial fulfilment of the requirements for
the degree of

Doctor of Philosophy
in
Information Systems

at Massey University, Palmerston North, New Zealand

Erina Francina Albertyn

2010

E-process selection using decision making methods

Erina Francina Albertyn

Supervisors : Associate Professor R Kaschek
Associate Professor E. Kemp

Department : Information Systems

Degree : Doctor of Philosophy

Keywords

E-Process, Selecting processes, e-Commerce, e-Commerce Information Systems, Case-Based Selection, Development Processes.

Summary

The key objective of this research is to develop a selection methodology that can be used to support and aid the selection of development processes for e-Commerce Information Systems (eCIS) effectively using various decision methods. The selection methodology supports developers in their choice of an e-Commerce Information System Development Process (e-Process) by providing them with a few different decision making methods for choosing between defined e-Processes using a set of quality aspects to compare and evaluate the different options. The methodology also provides historical data of previous selections that can be used to further support their specific choice.

The research was initiated by the fast growing Information Technology environment, where e-Commerce Information Systems is a relatively new development area and developers of these systems may be using new development methods and have difficulty deciding on the best suited process to use when developing new eCIS. These developers also need documentary support for their choices and this research helps them with these decision-making processes.

The e-Process Selection Methodology allows for the comparison of existing development processes as well as the comparison of processes as defined by the developers. Four different decision making methods, the Value-Benefit Method (Weighted Scoring), the Analytical Hierarchy Process, Case-Based Reasoning and a Social Choice method are used to solve the problem of selecting among e-Commerce Development Methodologies.

The Value-Benefit Method, when applied to the selection of an e-Process from a set of e-Processes, uses multiple quality aspects. Values are assigned to each aspect for each of the e-Processes by experts. The importance of each of the aspects, to the eCIS, is defined in terms of weights. The selected e-Process is the one with the highest score when the values and weights are multiplied and then summed.

The Analytic Hierarchy Process is used to quantify a selection of quality aspects and then these are used to evaluate alternative e-Processes and thus determining the best matching solution to the problem. This process provides for the ranking and determining of the relative worth of each of the quality aspects.

Case-Based Reasoning requires the capturing of the resulting knowledge of previous cases, in a knowledge base, in order to make a decision. The case database is built in such a way that the concrete factual knowledge of previous individual cases that were solved previously is stored and can be used in the decision process. Case-based reasoning is used to determine the best choices. This allows the user to either use the selection methodology or the case base database to resolve their problems or both.

Social Choice Methods are based on voting processes. Individuals vote for their preferences from a set of e-Processes. The results are aggregated to obtain a final result that indicates which e-Process is the preferred one.

The e-Process Selection Methodology is demonstrated and validated by the development of a prototype tool. This tool can be used to select the most suitable solution for a case at hand.

The thesis includes the factors that motivated the research and the process that was followed. The e-Process Selection Methodology is summarised as well as the strengths and weaknesses discussed. The contribution to knowledge is explained and future developments are proposed. To conclude, the lessons learnt and reinforced are considered.

Dedication:

I would like to dedicate this thesis to my father, Albertus (Arrie) Schutte, who passed away in October 2007.

Acknowledgements

I would like to sincerely thank my husband, Pieter Albertyn, for his love and support and my daughters, Riani and Carin Albertyn for caring.

I should like to express my sincere thanks and appreciation to my supervisor, Associate Professor Roland Kaschek for his support and expert guidance. I would also like to thank Associate Professor Elizabeth Kemp for her willingness to support and guide me during the finalisation of the thesis.

I am also greatly indebted to the University of Massey and my employer, the Eastern Institute of Technology for their monetary support that made it possible to present the results of this research at a number of International Conferences.

Frina Albertyn
July 2010
Napier, New Zealand

TABLE OF CONTENTS

	Page
Copyright	xiv
Publications and this thesis	xv
Chapter 1 Introduction	1
PART A INTRODUCTORY CONCEPTS	7
Chapter 2 PROBLEM EXPOSITION	9
2.1 Introduction	9
2.2 Research methodology	13
2.3 E-Process Selection Methodology Description	19
2.4 Objective Statement	23
2.5 Importance of the Research	24
2.6 Limitations of the research	28
2.7 Summary	28
Chapter 3 INVESTIGATIVE INITIAL CASE STUDY	29
3.1 Introduction	29
3.2 Investigative research	29
3.2.1 ISD process usage	29
3.3 Conclusion	34
Chapter 4 QUALITY ASPECTS	35
4.1 Introduction	35
4.2 High level quality aspects	39
4.3 Second level quality aspects	39
4.4 Conclusion	41
Chapter 5 e-PROCESS ALTERNATIVES OVERVIEW	42
5.1 Introduction	42
5.2 E-Commerce Information Systems	43
5.3 eCIS Development Processes	44
5.4 Framework for Comparison	46
5.5 Rational Unified Process	47
5.6 Agile methods and extreme programming	50
5.7 Open source development process	52
5.8 User Profiling and Storyboarding	53
5.8.1 User Profiling	54
5.8.2 Storyboarding	56
5.9 Comparison of RUP, AM(XP), OSS and SBUP	57

5.10 Summary	65
PART B DECISION MAKING METHODOLOGY	67
Chapter 6 DECISION MAKING	69
6.1 Introduction	69
6.2 Conceptualisation of Decision Making	69
6.3 Quantitative and qualitative decision rules	71
6.4 Purpose of Decision Making	72
6.5 Types of decisions and software options	73
6.6 Decision analysis and Decision Making Methods	75
6.7 Multi-criteria decision making	78
6.8 Decision making aspects for this research	81
6.9 Conclusions	84
Chapter 7 DECISION MAKING APPROACH – THE META-MODEL	85
7.1 Introduction	85
7.2 Value Benefit Analysis (Weighted score) (VBA)	86
7.2.1 Un-weighted 0-1 factor method	86
7.2.2 Weighted scoring method	87
7.2.3 VBA e-Process Ontology	89
7.2.4 e-Process VBA Selection Methodology	90
7.2.5 e-Process Selection VBA Meta-Model	91
7.3 Analytic Hierarchy Process	93
7.3.1 An AHP example	95
7.3.2 AHP e-Process Selection Model	96
7.3.3 Quality aspect ranking and e-Process alternative ranking	97
7.3.4 Scoring and first recommendation	99
7.3.5 Sensitivity Analysis	99
7.3.6 e-Process Selection AHP Meta-Model	101
7.4 Case-base Reasoning Approach	103
7.4.1 CBR technique used	103
7.4.2 CBR e-Process Selection	106
7.4.3 CBR e-Process Vocabulary	106
7.4.4 CBR e-Process Selection and the similarity measure	106
7.4.5 CBR e-Process Selection Case Base	107
7.4.6 e-Process and solution transformation	108
7.4.7 e-Process Selection CBR Meta-Model	108
7.5 Social Choice Method	109
7.5.1 Social Choice Theory	110
7.5.2 SCM quality aspects used	112

7.5.3 The SCM e-Process Selection Methodology	112
7.5.4 e-Process Selection SCM Meta-Model	113
7.6 Integrated e-Process Selection Meta-Model	114
7.7 Conclusion	114
Chapter 8 DECISION MAKING METHODS – RECOMMENDED USAGE	116
8.1 Introduction	116
8.2 Decision model usage comparison	118
8.2.1 Value Benefit Analysis	118
8.2.2 Analytic Hierarchy Process	119
8.2.3 Case-Base Reasoning	121
8.2.4 A Social Choice Method	121
8.3 Conclusion	122
PART C APPLYING DECISION MAKING METHODS	123
Chapter 9 APPLYING THE DIFFERENT DECISION MAKING METHODS to e-PROCESS SELECTION	125
9.1 Introduction	125
9.2 Applying e-Process Selection using Value Benefit Analysis	127
9.2.1 The sample problem (Club eCIS)	127
9.2.2 Applying VBA	129
9.3 Applying e-Process selection using AHP	131
9.3.1 The sample problem (Quilting eCIS)	131
9.3.2 Applying AHP	133
9.4 Applying e-Process selection using Case-Based Reasoning	137
9.4.1 The sample problem (Packaging eCIS)	137
9.4.2 Applying CBR	138
9.5 Applying e-Process Selection using a Social Choice Method	141
9.5.1 The sample problem (Kayak eCIS)	141
9.5.2 Applying SCM	142
9.6 Validation	144
9.6.1 Cross Case analysis	144
9.6.2 Validation of VBA	145
9.6.3 Validation of AHP – applying sensitivity analysis	148
9.6.4 Validation of CBR	151
9.6.5 Validation of SCM	154
9.6.6 Analysis of results obtained by applying DMM	155
9.6.7 Feasibility of using the e-Process selection Methodology	158
9.7 Final Conclusions	164

PART D PROTOTYPE DEVELOPMENT	165
Chapter 10 PROTOTYPE DEVELOPMENT DOCUMENTATION	167
10.1 Introduction	167
10.2 Requirement Specification	168
10.3 Use Case Diagrams	169
10.4 Activity Diagrams	172
10.5 Decision Analysis	176
10.6 Three-tier System Software Architecture	179
10.7 Database Schema	182
10.8 System Implementation	184
Chapter 11 PROTOTYPE TOOL	185
11.1 Introduction	185
11.2 Phase 1 – focus on VBA	185
11.3 Phase 2 – focus on AHP	188
11.4 Phase 3 – focus on CBR	191
11.5 Phase 4 – focus on SCM and other admin functions	193
11.6 Conclusions	195
PART E CONCLUSION AND FUTURE WORK	197
Chapter 12 CONCLUSION AND FUTURE WORK	199
12.1 Introduction	199
12.2 Assessment of objectives achieved	199
12.3 Summary of the Research Process	200
12.4 Strengths and Challenges of the e-Process Methodology	201
12.5 Contribution to Knowledge	203
12.6 Future development	204
APPENDIX A: Publications by author on e-Process Selection	205
APPENDIX B: Technology Specifications	207
APPENDIX C: Legends of different types of figures used in thesis	208
APPENDIX D: Methodological Survey into Web Development Usage	210
References	213
Glossary of terms	226
Index of terms	228

List of Figures

Figure 1.1: Thesis Framework	4
Figure 2.1: Framework for discussion of Problem Exposition	9
Figure 2.2: Decision Methods used in e-Process Selection	12
Figure 2.3: Methodology framework	20
Figure 2.4: Comparison of development processes	26
Figure 3.1: Investigative Case Studies Framework	29
Figure 3.2: Structured development process usage	32
Figure 3.3: Specific Methods used by Developers	33
Figure 4.1: Framework for quality aspects	35
Figure 5.1: Framework for e-Process alternatives overview	42
Figure 5.2: IS development abstraction layers	46
Figure 5.3: The RUP dimensions and phases	48
Figure 6.1: Aspects of decision making included in this chapter	70
Figure 6.2: The Relative Value of Decision Support and Decision Automation for Different Types of Decisions (based on Taylor, 2007)	74
Figure 6.3: A taxonomy of MCDA methods (Chen and Hwang, 1991)	81
Figure 6.4: A semantic view of the decision meta-model	83
Figure 7.1 Aspects of a decision making meta-model	85
Figure 7.2 Sample Project Evaluation Form	86
Figure 7.3: Mathematical form of the weighted scoring method	87
Figure 7.4: Example Weight Scoring Models (Mantel, Meredeith, et al., 2005)	88
Figure 7.5: e-Process Selection VBA Meta-Model	92
Figure 7.6: Analytic Hierarchy Process	94
Figure 7.7: Sensitivity diagram example	100
Figure 7.8: e-Process Selection AHP Meta-Model	102
Figure 7.9: CBR process	105
Figure 7.10: e-Process Selection CBR Meta-Model	108
Figure 7.11: e-Process Selection SCM Meta-Model	113
Figure 7.12: Integrated e-Process Selection Meta-Model	114
Figure 8.1: Decision Making Methodology Framework	116
Figure 8.2: Decision making and decision models	117
Figure 9.1: Framework for Value-Benefit Method discussion	126
Figure 9.2: Second-level aspect scores for high level quality aspect: e-P aspects	136
Figure 9.3: Final e-Process Scores and contributions of quality aspects	136
Figure 9.4: Sensitivity diagram for the Quality Concepts	149

Figure 10.1: Framework for eProcSel specification	167
Figure 10.2: eProcSel Use Case Diagram	169
Figure 10.3: VBA Use Case Diagram	170
Figure 10.4: AHP Use Case Diagram	170
Figure 10.5: CBR Use Case Diagram	171
Figure 10.6: SCM Use Case Diagram	171
Figure 10.7: eProcSel Activity Diagram	172
Figure 10.8: eProcSel VBA Decision Making Method Activity Diagram	173
Figure 10.9: eProcSel AHP Decision Making Method Activity Diagram	174
Figure 10.10: eProcSel CBR Decision Making Method Activity Diagram	175
Figure 10.11: eProcSel SCM Decision Making Method Activity Diagram	176
Figure 10.12: System Software Architecture	179
Figure 11.1: VBA Initial Screen	186
Figure 11.2: VBA High Quality Aspect Entry	186
Figure 11.3: First Phase Suitability Check	187
Figure 11.4: VBA Second Level Quality Aspect Values Entry	187
Figure 11.5: Initial setup of goal, criteria and alternatives	189
Figure 11.6: Rating of alternatives to all criteria (quality aspects)	190
Figure 11.7: Initial Results on e-Process Selection	190
Figure 11.8: Sensitivity Analysis	191
Figure 11.9: Capturing CBR information - determine similar cases	192
Figure 11.10: CBR Similar Result	193
Figure 11.11: Capturing SCM results	194
Figure 11.12: Entering Quality Aspect Information	195
Figure 12.1: Framework for evaluation, conclusions and future work	199

List of Tables

Table 4.1: Hierarchy Structure of Software Product Quality Aspects	37
Table 5.1: Comparison of e-Processes	57
Table 7.1: e-Process Selection VBA Meta-Model Data Dictionary	92
Table 7.2: Pair-wise Comparison Values	95
Table 7.3: Object preferences	95
Table 7.4: Weights on objectives	95
Table 7.5: Salary scores and relative scores	96
Table 7.6: Relative scores per objective	96
Table 7.7: e-Process Selection AHP Meta-Model Data Dictionary	102
Table 7.8: e-Process Selection CBR Meta-Model Data Dictionary	109
Table 7.9: e-Process Selection SCM Meta-Model Data Dictionary	113
Table 7.10: Combined e-Process Selection Meta-Model Data dictionary	115
Table 9.1: Definition of weights and alternative values the quality aspects	130
Table 9.2: Selected experts' e-Process quality aspects scores	134
Table 9.3: High Level Quality Aspects Ranking	135
Table 9.4: Comparing high level quality aspects for two cases with problem	139
Table 9.5: Low level quality assessment results	140
Table 9.6: Scores obtained when applying Borda voting	143
Table 9.7: Comparing high level quality aspects for two cases with problem (2)	146
Table 9.8: Low level quality assessment results	147
Table 9.9: CBR values used in applying VBA	153
Table 9.10: Cost/Benefit Analysis of applying SCM to Kayak eCIS	155
Table 9.11: Comparing the decision making methods	157
Table 9.12: Operational feasibility of the new e-Process Selection Methodology	158
Table 9.13: Cost associated with the development of the methodology	161
Table 9.14: Cast associated with the implementation of the methodology	162
Table 9.15: Cost/benefit analysis	163
Table 10.1: Comparison of candidate solutions	177
Table 10.2: Feasibility Analysis Matrix	178
Table 12.1 Thesis objectives and analysis of results	200

Copyright

Copyright is owned by the Author of the thesis. Permission is given for a copy to be downloaded by an individual for the purpose of research and private study only. The thesis may not be reproduced elsewhere without the permission of the author.

Publications and this thesis

Much of the information contained in this thesis has been published in journals, international conference proceedings, and national conference proceedings or as technical papers. The style in the chapters might in some cases be in the style of the particular publication. Consequently, there is some repetition, particular in the application sections and there are stylistic differences between the chapters.

Some of the published work included other authors. For each of the chapters my input was substantial. The initial design of the research, some of the fieldwork, the analysis and the writing of the publication has been mainly my work. For some of the publications I was, however, assisted by co-authors and would like to thank them for their contributions.

A list of publications by the author on e-Process Selection can be found in appendix A.

Chapter 1

Introduction

E-Commerce can be defined as the trading of products over the internet. The internet provides the facilities to run a flexible and adaptable place for buying and selling goods, electronically, as well as the execution of financial processes. This is called an e-Commerce website or information system. It is becoming more and more important for businesses that want to expand, both globally and locally, to use e-Commerce websites. An e-commerce information system is the software, hardware and other components required to run an e-Commerce website. E-Commerce Information Systems (eCIS) require allowing the exchange of information over the internet in order to facilitate the financing and payment aspects of the business transactions taking place between the client and the company.

Rapid growth has taken place in the use of the Web for commercial purposes. This includes the need for companies to develop e-commerce web sites (Cheung, 1998; Taylor, McWilliams, Forsyth & Wade, 2002; Vidgen, Avison, Wood, & Wood-Harper, 2002). The same challenges and problems that existed with the development of "traditional" Information Systems (IS), also exist when developing e-commerce information systems (eCIS) (Butler, 2003). eCIS therefore require a mix of web development techniques (such as user profiling) with traditional IS development competencies which includes database and program design (Gruhn & Schope, 2002; Vidgen, 2002). eCIS development requires quick project completion, yet demands quality software (Sharma, Sugumaran, & Rajagopalan, 2002).

It is well known that the task of developing information systems is difficult. In order to obtain high quality systems, a very strong focus is placed on the process-oriented aspects of information systems development (Kautz & Nielsen, 2004). If non-suited processes are used the quality, in particular the usefulness of the information system under development, is likely to be unsatisfactory (Sommerville, 2000). According to Butler (2003) many issues in the development of non-Web information systems are also present in the development of Web-information systems.

The use of the Web varies from business to business. E-commerce is a more advanced use of the Web, and capable of enhancing the competitiveness of businesses and of

increasing the efficiency of their operations (Pflughoeft, Ramamurthy, Soofi, Yasai-Ardekani, & Zahedi, 2003).

Creating an e-Commerce Information System also creates the technical foundation for the appreciation of its quality. It is a common belief that the development process used to develop an eCIS impacts on the quality of the resulting systems (Chrissis, Konrad & Shrum, 2006; Pfleeger, Hatton & Howell, 2002; Tian, 2005; Li, Allaeddine & Tian, 2010). Some developers decide not to use a development process which reduces the chances of the resulting eCIS being of a high quality. The developers' choice of a development process might also be wrong, flawed or inappropriate.

According to Brynjolfsson (2003, p. 1): "IT has been the biggest single factor driving the productivity resurgence" and "IT creates value only if it lets users work more effectively". MacCormack, et al. (2003, p. 79) writes that the "... problem is not so much that we lack 'silver bullets' for a project manager's arsenal but that we have such an extensive array of differently coloured bullets that choosing among them has become impossible."

It is therefore recommended that developers select an appropriate development process to follow from some of the available development processes. This might not be an easy task as the decision makers: may not have sufficient knowledge about the software development processes to make a choice, may not have enough knowledge about the quality aspects of the processes they are considering, or they may not have had a selection methodology to help them to make the choice.

There are a number of differences between various development processes such as: the cost of these development processes; the speed of development using a specific development process; how easy and understandable that specific development process is; whether there are prescribed workflows defined for that development process, and many more characteristics. The Rational Unified Process, a well-known development process, is quite involved and can be expensive to acquire and use while development processes which are open source can be quite cheap to use. Agile Development Processes are known for the speed with which development can take place, while some of the other development processes are very thorough and may need more time for development. It is also possible to sometimes combine some of the features of development processes in order to have a development process that suits the requirements for the development of a specific eCIS.

An investigative case study (see chapter three) shows the results of research conducted to determine whether developers still use development processes. The research shows that most developers use development processes. There are a number of reasons why decision support is required in the selection of a suitable development process. These include: to help the people responsible for selecting the e-Process (eCIS development process) to make a choice; to allow the selection process to be documented; to provide the rationale why a specific development process is used; to allow the decision makers to compare their preferred or combined development process with others in order to determine whether they have made a good choice; education and staff training; compliance with standards – e.g. company policies or government might require; comparability – to introduce a method for measuring staff for salary; and maintenance – to allow for technical standardisation.

There are also a number of companies, with their decision makers (developers and managers), who can benefit from getting some support when selecting e-processes. These include:

- The decision makers of a company who want to determine what to specify or contemplate when they ask Bidders to bid for a contract to do eCIS development for their company;
- Decision makers having support when deciding which Bidders' quotation is the most relevant to their problem environment;
- Companies that might want to build up an eCIS development team and want to determine what skills to require from the developers they are going to employ;
- Decision makers of small vendors who want to determine what training to give their developers in order to build up their relevant skill set;
- Medium to large companies or vendors who have choices in the development methods that they can apply;
- Companies that might like to monitor projects which are being developed for them by vendors and;
- Helping decision makers to determine if an e-Process which has already been selected, is an acceptable choice.

This thesis aims to provide these decision makers with a method for selecting eCIS development processes (e-Processes) by using various decision making methods. The selection method uses quality aspects, with a scale for each one; and this thus provides a conceptual framework for quantitatively assessing e-Processes. The decision making methodology as well as each of the decision making methods used are discussed in this thesis. The decision making methods presented for selection of the e-Process are the

value-benefit method, the analytic hierarchy method, case-based reasoning and a social choice method. The thesis concludes with the architecture, development documentation, and resulting e-Process selection tool.

As seen in Figure 1.1, the thesis is divided into five main parts which include Part A, the introductory concepts to the thesis, Part B, which describes the decision making methodology used in the thesis, Part C which describes the application of the selected decision making methods, Part D, which describes the prototype tool and its construction, and Part E, which describes the evaluations and conclusions reached in this research. Part A is discussed in chapters 2 to 5; Part B is covered chapters 6 to 8; Part C by chapter 9, Part D by chapters 10 and 11 and Part E by chapter 12.

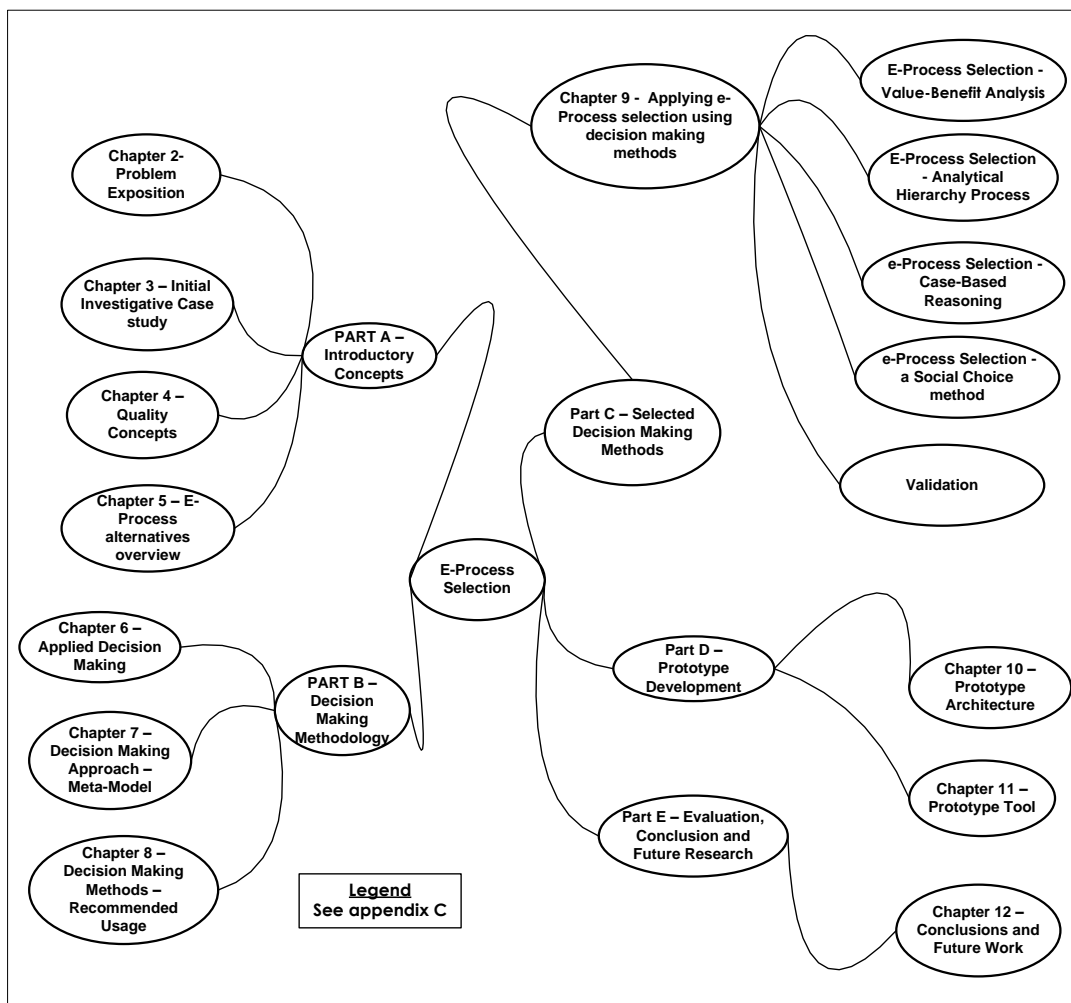


Figure 1.1: Thesis Framework

The second chapter provides detail about the origins of, and the motivation behind, the research that is presented in this thesis. In the e-commerce current environment, a new competitor can appear quickly in the international marketplace and transform the whole way that business is conducted. It is therefore important that information systems development can be rapid, adaptable and flexible. In an e-commerce environment it

is also important to remember that the systems development process is combined with marketing, branding of products and that the communication with the user is of utmost importance (Knight, Steinbach & Keller, 2001). The aim of this chapter is to develop the framework within which the problem of providing support to developers to select an appropriate system development methodology for developing an e-commerce system can be explored. The methodology used for the thesis, the key objectives, the importance and limitations of the research are also discussed.

Chapter three introduces the introductory investigative research done to support the process followed to solve the problem presented in the thesis. Chapter four discusses the quality aspects that are used through the thesis for evaluation of the e-Processes using the different decision methods. In chapter five the e-Process alternatives used through the thesis are briefly discussed and compared.

Chapter six introduces decision making, specifically multi-criteria decision making, which is used to define a number of the e-Process Selection Method options for selection.

Chapter seven discusses the different decision making methods and the methodology used in the thesis, by providing and building the meta-model required for the e-Process selection method. In the last section of this chapter an integrated meta-model for e-Process Selection is presented. This chapter includes the decision making methods and their supporting techniques. The four chosen decision models - the Value benefit method; the Analytic Hierarchy Process; the Case-Based Reasoning process and Social Choice Decision method, are each discussed in detail in this chapter.

In section 7.2 the ontology for development processes of eCIS (e-Processes) based on the Value Benefit Method is discussed. This ontology provides e-Process quality aspects and for each of these a scale culminating in a unifying conceptual framework for quantitatively assessing e-Processes. The selection methodology is proposed and some well-known e-Processes are used for exemplifying the quality aspects.

In section 7.3 the Analytic Hierarchy Process is discussed to show how the generated data can be used to ensure that a sensible selection is made. This section discusses the selection procedure and incorporates some sensitivity analysis.

In section 7.4 the use of Case-Based Reasoning is discussed as a basis for selecting a suitable e-Process for a specific problem domain. In order to use this method the

creation and maintenance of a database with previous cases are firstly outlined. This database is then used to determine whether there are enough similarities, with the new project at hand, to use this case as a recommended solution and prevent having to repeat previous selection processes. The resulting selection is tested in a simulated world and then retained as a reasonable solution in the knowledge base for future reference.

Section 7.5 describes how social choice methods can be used to choose between different e-processes being considered. The first part of this section describes how the social choice theory of the Borda Method can be applied to a generic environment. This method is then applied to the selection of e-Processes.

Finally the decision methods and techniques used in chapter 7 are integrated and form the "toolbox" of the e-Process Selection Methodology.

Chapter eight discusses the recommended usage of the different proposed decision models for e-process selection and then evaluates their strengths and weaknesses.

Chapter nine (part C) applies Value-Benefit Analysis, Analytic Hierarchy Process, Case Based Reasoning Method and a Social Choice Method, identified and discussed in chapter 7, on different real case scenarios. This is followed by a section on the validation processes followed (applying a second decision making method on a specific case study, rich case descriptions, sensitivity analysis or doing a cost/benefit analysis). Also included in this chapter is a cost/benefit analysis to show that there is benefit in applying this e-Process Selection Methodology.

Chapter ten defines the development documentation used for the development of the prototype tool. This documentation includes use case diagrams, activity diagrams, software architecture and the database schema. In chapter eleven the prototype tool development is discussed and some examples provided of the final interfaces developed and used in the prototype.

The last chapter provides a synopsis of the research as well as some further discussion on possible research directions in the future. The factors that motivated the research are identified and discussed. The process that was followed is detailed and a concise summary of the modelling methodology is provided. The strengths and weaknesses of the e-Process Selection Methodology are discussed. The contribution to knowledge is explained and possible future developments are proposed. To conclude, a summary of the lessons learnt and reinforced during the completion of the research are presented.

PART A

INTRODUCTORY CONCEPTS

Chapter 2

PROBLEM EXPOSITION

The research presented in this thesis focuses on using different decision methods in order to make a reasonable, defensible or suitable selection of a development process to use for developing an e-Commerce Information System. The purpose of this introductory chapter is to provide an explanation of the research foundations presented in the thesis. This chapter also explains the methodology used throughout the thesis. The objectives of this study as well as limitations and challenges encountered during this study are introduced.

2.1 Introduction

The term "exposition" is defined in dictionaries as an explanatory statement or account, which is what the author is attempting to achieve in this chapter. This chapter provides detail about the origins of, and the rationale behind the research is presented. This chapter's outline used for discussion can be seen in Figure 2.1.

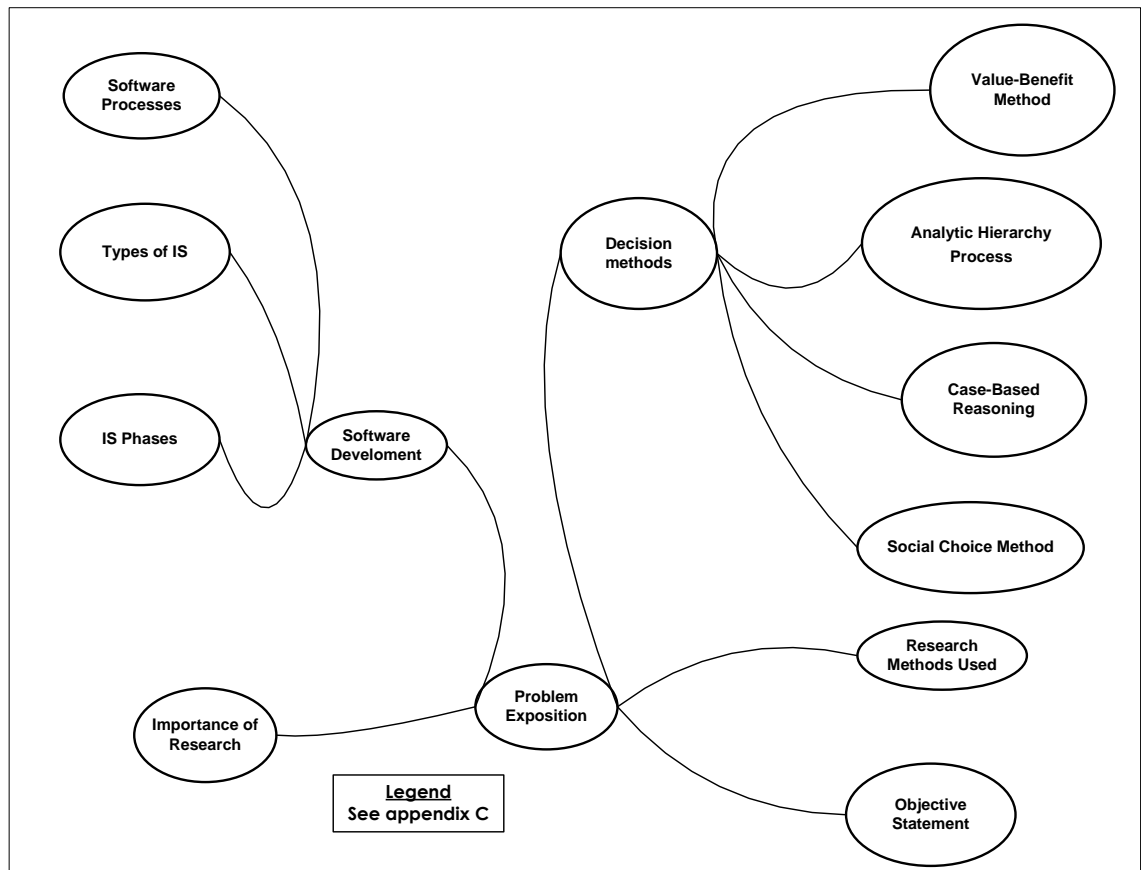


Figure 2.1: Framework for discussion of Problem Exposition.

Technology, especially computer technology, has developed at a rapid tempo. As the performance of computer hardware and software has increased, the demand for more advanced and well-developed quality software has increased. This has meant that more specific and specialised software development processes had to be developed. These software development processes need to consist of an organised strategy and often consist of phases and life-cycles.

An Information System needs to be developed in such a way that the end product is trustworthy (dependable in all circumstances and the user can have confidence in the product) and robust (the product is strongly constructed and capable of performing under different conditions without failing). According to Taguchi (1990), it is important to design products not to fail in the field and reduce defective products in the factory. This can be applied to the development of Information Systems as well. It is therefore important that in order to achieve such a product it is necessary to choose the software development process with care to ensure a quality end product. In this thesis a multifaceted view of quality was adopted in order to compare different development processes in order to choose one that will produce a quality end-product. The quality aspects chosen for comparison are further discussed in chapter 4 of the thesis.

There are many different types of Information Systems being developed. Some of these types are database systems, web application systems including e-Commerce Information Systems (eCIS), transaction processing systems and many more.

Web Information Systems are a special type of Software Information System and E-Commerce Information Systems (eCIS) are a special type of Web Information System. All types of Information Systems use software development processes in their development. In this thesis the term e-Process is used. An e-Process Model (as used in this thesis) is the development process used for the development of an eCIS. Web development, especially eCIS development, requires quick completions of the project, while delivering quality software, (Sharma, Sugumaran & Rajagopalan, 2002).

There exist many development processes and not all of these have the features that are required and will support the development of eCIS. Commercial Web development, as an industry, has been around about 12 years which makes it fairly young when you consider other Information Systems have been around for about half a century. eCIS development is even younger. There are a huge number of websites and eCIS that are built and rebuilt for clients every day, which points to the fact that there might be poor quality work being done (Bauer, 2005).

It is therefore important that a suitable tried, trusted and standardised approach is adopted when developing eCIS. It is however very difficult to find a software development process that suits the development of a specific eCIS. The author has not been able to find any, but according to Bauer (2005) there are not many eCIS, if any, development processes that have been developed exclusively for just that purpose. This will mean that in some instances the developers will need to adopt or adapt existing methodologies (Bauer, 2005).

One should however remember that there are risks involved if the wrong or unsuitable software development processes were to be selected. Some of the risks are that the resulting eCIS might not have the quality that one would expect; and that the development process might not be known to the developers; or that the development process might be too expensive for a small company to be able to afford.

In this thesis the focus was on selecting development processes that are most suited for a specific eCIS development project.

eCIS development, compared to traditional software development is unique on a number of issues which include:

- The time frames for development are shorter for eCIS.
- The experience levels of the developers vary and could be quite limited because of the newness of eCIS development.
- The clients often have a limited understanding of what is possible and a limited view of what they require.
- The technology changes quickly and developers need to be constantly adapting to new versions of software, new technology etc.
- There is a prescribed interface technology that needs to be used.

(Bauer, 2005)

- eCIS involves the buying and selling of goods which implies that money will most probably electronically change hands when the deal is done. This implies extra features such as strong security and links to banking or credit card facilities.
- eCIS is interactive in a web environment and will most probably be linked to a database. eCIS will also include all the usual Information Systems required in businesses such as inventory control, marketing, incoming goods, and sales.
- Consumer protection is no longer relevant. eCIS transactions are not protected under legislation.

- New environment – shopping cart instead of personal inspection. Goods are sold differently – user does not look at or inspect the goods.

This does not mean that these components do not exist in traditional software development – it just means that the limitations are much more pronounced in Web development and specifically eCIS (Bauer, 2005).

The aim of the research presented in this thesis was to provide the developer with a number of decision making methods, each of which allow the developers to select a suitable process for their development based on the characteristics and features of the eCIS that are being developed. The reasons why there is a need for such a support process is:

- Many developers are new to eCIS development and need help in making the decision on which process to use.
- There is also often the need to be able to have documented support that there was a reason for choosing one development process above another. This research will help to understand and thus justify why a specific e-Process was decided on.

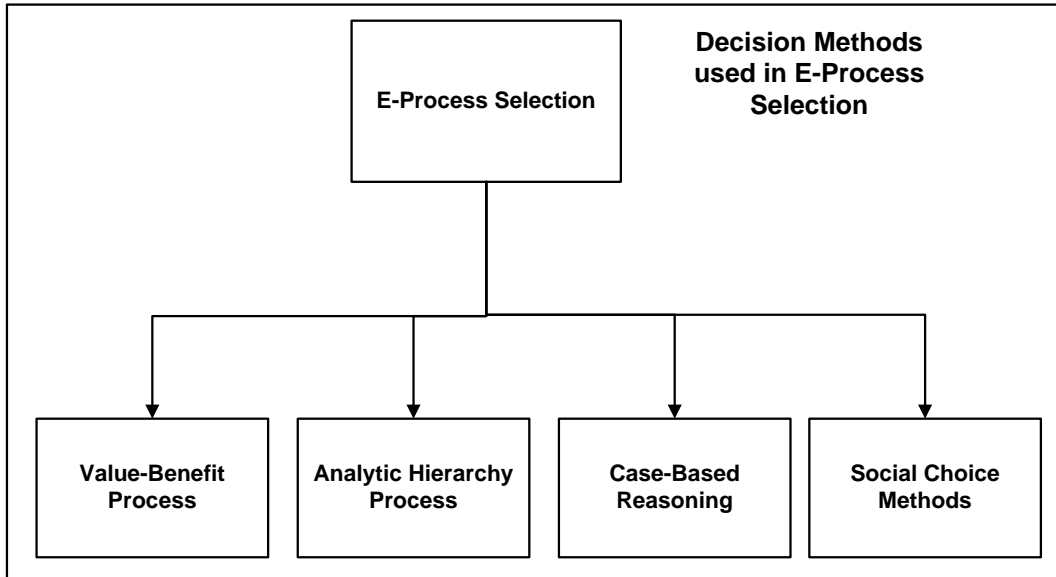


Figure 2.2: Decision Methods Used in e-Process Selection

All the decision making methods presented are based on a number of quality aspects (see chapter 4). The decision models presented in this thesis (see figure 2.2) are value-benefit analysis; analytic hierarchy process; case-based reasoning and a social choice

method.

2.2 Research methodology

“Research involves the application of various different methods and techniques in order to create scientific knowledge by using objective methods and procedures” (Welman & Kruger, 2001, p.2).

According to Birney and Moreland (1998), Welman and Kruger (2001), Yin (2006) and Blaxter, Hughes and Tight (2008) research can be either qualitative or quantitative - with quantitative research being empirical research where the data is in the form of numbers and qualitative research being empirical research where the data is not in the form of numbers.

Quantitative research requires that the reality be measured and verified objectively to test a specific hypothetical understanding. Some of the more common quantitative research methods are experimental research and survey designs.

Qualitative research consists of inquiry into the problem domain; defining a set of assumptions about how knowledge is produced and defining reality (Birley & Moreland, 1998; Burns, 2000; Davidson, 2005; Flick, 2004; Greenfield, 2002; Kumar, 1996; Leary, 2004).

Qualitative research focuses on discovering what people experience and interpret of the real world and how these experiences impact on their actions. Qualitative research includes methods such as ethnographic, critical ethnographic, case studies, participant observation, in-depth interviews, evaluations, historic research and market research (Birley & Moreland, 1998; Burns, 2000; Davidson, 2005; Flick, 2004; Greenfield, 2002; Kumar, 1996; Leary, 2004).

According to Creswell (1994) and Merriam (1988) qualitative researchers focus on the process, rather than outcomes and products; are interested in how people interact with their experiences and world structures; do their own data collection using human interaction; get immersed in the actual setting where the data is collected; try to understand the process by being descriptive; and develop abstractions and theories from the detail obtained.

In Yin (2006) the idea is that different research methods serve different complementary functions and therefore in this thesis multiple methods are used which include the use of literature study, descriptive research with descriptive surveys and case studies as the main research methods and then being able to prove the validity of the end product.

Literature study as part of research means that it was necessary to consult other research conducted on the same topic areas in order to understand the problem at hand and in order to investigate the research area. Looking at the existing research enabled the author of the thesis to critically evaluate what has been produced before in order to identify the significant parts that impact on the research that was being investigated and allow the relationships to be built between the research being done and the existing available work. Literature study helped to place the current research in perspective and provided context for the work undertaken (Flick, 2006).

Reviewing literature for the research helped the researcher to understand what was known about the issue in particular; theories available on this topic; concepts that are being used and concepts that have been disputed; what are the areas of controversies in this field; any unanswered questions and; what still needs to be studied (Flick, 2006).

Descriptive research is when the result of the research is a description of the data, using different formats such as words or pictures or when the statistical relationships are determined by analysis of the data or the data is described. This type of research helps to describe characteristics and behaviours within a specific subject area in a systematic and accurate manner (Leary, 2004).

Different variables needed to be investigated in order to describe the result. If one knows the variables, but it is not possible to predict their action, then this could lead to the use of descriptive survey designs. Descriptive surveys were used to obtain information about what is occurring or exists in a real life situation and determine its status. It seeks to determine the perspectives or experiences, of respondents, on a specified subject in a predetermined structured manner. Descriptive surveys can be used as part of a larger research methodology (Leary, 2004).

Participant observation consists of three phases:

- Descriptive observation is done to provide the researcher with an orientation to the field to be studied. The researcher aims to understand the complexities of the specific area being investigated and the results in a nonspecific description of the area being investigated;
- Focused observation, which focuses on the processes and problems, and are essential to determine the research question and;
- Selective observation, which is focused on finding evidence and examples of other types of practices and processes being used.

(Flick, 2006)

Yin (2006) writes that the case study, as a methodology, has reached a point where it can be seen as being a routine method. The strength of using case studies is its ability to explore in detail the specific scenario within its “real-life” context. Yin (2006) and many other researchers, who have wide experience in case study methodology, have developed robust procedures, which, when followed, are as well developed and tested as any other research methodologies in the scientific field (Tellis, 1997).

Developing a theoretical or practical methodology, when doing case study research, is an iterative process. The process involves the researcher investigating one part of the project and then moving onto the next part – but having to revisit the issues involved in the first part iteratively. In the process there might be cross-case comparisons as well as possibly redefining some of the research questions, before gathering further evidence on additional cases (Eisenhardt, 1989). A number of tactics and methods are used in order to gather the information for the different cases. The aim is to use all of the input in order to construct definitions, measure some of the results and then build a framework for the evaluation of the results (Eisenhardt, 1989, p.546).

The case study research methodology then:

- Creates a new theoretical vision using the literature foundations and the results obtained from the different cases;
- Create constructs which are measurable and hypothesis that can be proven true or false and have been verified during the theory building process;
- Deliver a resultant theory which is empirically valid. This is achieved because the theory building process is tied to evidence from the case applications.

(Eisenhardt, 1989, p. 547).

One of the possible risks of using case study research is the intensive use of empirical evidence can possibly lead to a theory which is too complex and tries to capture all the different options. This may lead to a theory which is has too much detail and not enough perspective (Eisenhardt, 1989). Eisenhardt (1989) further states that it is important when using case study as a research strategy, to understand the dynamics present in one or more settings and on various levels of analysis.

Eisenhardt (1989) discusses a framework with eight steps for building the theory from case study research, outline in the following:

Getting started: It is necessary to define the initial research question, when building a theory from case study research. This is done in order to not be overwhelmed with the amount of data available and to provide focus to the research and the types of case scenarios to be investigated (Eisenhardt, 1989).

Selecting cases: Another important feature of case study research is the selection of the cases. These cases are used in order to do the hypothesis testing and will help to form the theory at hand. In this thesis the cases were selected from the industry based problems available at the time of research and the relevant decision making method applied to the problem at hand in order to validate the theory (see part C in the thesis) (Eisenhardt, 1989).

Crafting instruments and protocols and entering the field: Theory building researchers combine different data collection methods such as interviews, questionnaires, observation and they are not confined to only these choices (Eisenhardt, 1989). Another feature of theory building using case studies is the fact the data collection and data analysis overlap. This can be seen in part C of this thesis where the researcher built the theory over a number of years and was therefore unable to use the same case study for all the decision making methods used in this thesis (see Part C of the thesis for further explanation) (Eisenhardt, 1989).

Analysing the data: In order to analyse the results obtained it was necessary to do within-case analysis. This involved detailed case study write-ups for each of the cases (Eisenhardt, 1989, p. 540). According to Eisenhardt (1989) these write-ups are usually simple descriptions of the results obtained, but form the basis to generate the insight required for the theory and the resulting validation of the theory. The validation (as seen in Part C of this thesis) was based on the comparison of the results obtained from the theory and the feedback as obtained from the different developers in the actual

problem domain. It is necessary to combine within-case analysis with the search for cross-case patterns. Within-case analysis typically involves detailed descriptions of the cases and sites and searching for cross-case patterns involve evaluating the data in different ways (Eisenhardt, 1989).

Shaping the hypothesis: The main idea of case study research is to iterate in such a way that a theory is developed that fits the data. The theory is applied to cases and refined and the theory extended to help to create an internal validity (Eisenhardt, 1989).

Enfolding literature: The literature was used to compare conflict that may occur as well as compare the research to similar findings (Eisenhardt, 1989).

Reaching closure: Iterate by adding cases until the improvements to the theory are minimal (Eisenhardt, 1989).

In Eisenhardt (1989) it is stated that there is no generally accepted set of guidelines for the assessment of theory building from case study research. The assessment is rather based on whether the concepts investigated, the framework being developed or the propositions that emerge from the process can be classified as “good theory”, with this being seen as theory which is developed with caution, is testable and is logically understandable (Eisenhardt, 1989).

In conclusion, the strengths of theory building using case study research are that it is a novel way to approach the problem; it is testable and has empirical validity. It is also well suited for research in areas where the current theory is new and inadequate (Eisenhardt, 1989).

It is however important to also understand the limitations of theory building using case study research. The intensive use of empirical data can lead to the development of a theory which is too complex and which tries to capture everything. The researcher using this method lacked the quantitative measurements such as regression analysis and observation over multiple studies. Another possible limitation of this methodology is that a bottom up approach is used. This means that the general theory is built by using the specific data. The risk here was that the researcher is not able to raise the generality of the theory (Eisenhardt, 1989). Eisenhardt (1989) suggests that to build the theory it is necessary to use multiple case studies – an accumulation of theory building and theory-testing of the empirical case studies.

Validity is the best possible approximation to the truth for any specific scenario or a proposition that has been made. Validity is the measure that is being used in order to lead to valid conclusions or the fact that specific samples enable valid inferences (Trochim, 2006). A methodological proposition provides the building block for the substantive conclusion that one wants to achieve. Most research has a measure of observation and measurement involved and these are influenced by the circumstances in which these occur. The research needs to draw conclusions about the quality of the observations and measurements. These conclusions provide the substance of the research that was done and thus the validity of the research that is being undertaken (Trochim, 2006).

In order to measure validity it is important to define variables to be used in the test that is going to be used when measuring. Validity information gives an indication of how well the test used measures that specific area for those specific circumstances and groups (Burns, 2000). Validity therefore relates to what is being measured by the test being performed and whether this test is valid for its purposes (Burns, 2000).

There are five types of validity namely content validity, predictive validity, concurrent validity, construct validity and face validity (Burns, 2000; Trochim, 2006). Content validity is whether the content is representative of the whole and whether the sample is adequate. This is determined by expert judgement. This is a non-statistical type of validity. Predictive validity is the desire to predict performance on some other criterion, using assessments or some other technique. The criterion measurement should be reliable. Concurrent validity differs from predictive validity in terms of time only. Concurrent validity has to do with the now. What is occurring now? Construct validity has to do with the explanation of the aspects of human behaviour, refers to all the evidence gathered and determines whether a particular operation of a construct is representative of what is intended by the theoretical account of the construct being measured (Burns, 2000; Trochim, 2006). Content validity therefore is dependent on the theory and uses this to determine if a test is assessing all domains of a specific criterion. This assumes a causal relationship and determines whether the program reflected the construct well and that the measure used reflected the idea of the construct measure. Face validity is concerned with whether the test being conducted appear to measure what was intended to be measured using the examination of the different items (Burns, 2000; Trochim, 2006), while construct validity, in contrast to face validity, is the degree to which the measures used by the researchers relate to the abstract construct being investigated (Smith & Glass, 1987; Yin, 1994; Yin, 2006). The construct, which would have been defined by the researcher, needed to match the definitions of the concept in the

literature. These literature concepts would have to be well established and would need to be seen as having authority in the area being investigated (Dey, 1993).

Another method of facilitating validation of the results obtained is to use triangulation. Triangulation is the cross verification from more than two sources. One can apply and combine a number of research methodologies when studying the same outcomes (Benbasat, et al. 1987; Yin, 1994). This method can be employed in both quantitative (validation) and qualitative (inquiry) studies. The credibility of qualitative analyses can therefore be founded using a method-appropriate strategy (Benbasat, et al. 1987; Yin, 1994). The researchers can aim to limit and overcome the weaknesses, problems and biases that can occur when a single method is used (Benbasat, et al. 1987; Yin, 1994).

Denzin (2006) identified four types of triangulation: Data triangulation which involves time, space and people; investigator triangulation which is when there are a number of researchers involved; theory triangulation is when more than one theory is used to interpret the phenomenon; and methodological triangulation is when more than one method is used to gather data such as interviews, observations, questionnaires and document gathering. The specific types of triangulation that were used in the thesis are methodological triangulation and data triangulation to a lesser degree (Benbasat, et al. 1987; Yin, 1994).

This next section will briefly describe and provide a framework for the e-Process Selection Methodology used in this thesis.

2.3 E-Process Selection Methodology Description

Following on from the previous paragraph, the research methodology used in this thesis is mostly based on building a theory based on case studies. The research, however, used other research methods as well, such as literature review, descriptive surveys and different types of validation.

Figure 2.3 shows an overview of the methodology followed in this thesis. Firstly the research in the thesis has a strong theoretical foundation. Multi-criteria decision making methods are covered in the literature and are used to define a method for selecting e-Processes. The discussion of the different decision making methods has a strong theoretical basis which helps to define the foundations of selection using that specific method. The theory building in this thesis used a case study methodology.

Case studies present stories of real life situations. There are a number of researchers that have used case study research successfully. Three of the most important methodological articles on using the case study method in the IS field are those by Benbasat et al. (1987), Dubé and Paré (2003) and Lee (1989). An article by Markus (1983) is one of the most cited empirical examples of case study research in Information Systems (Myers, 1997).

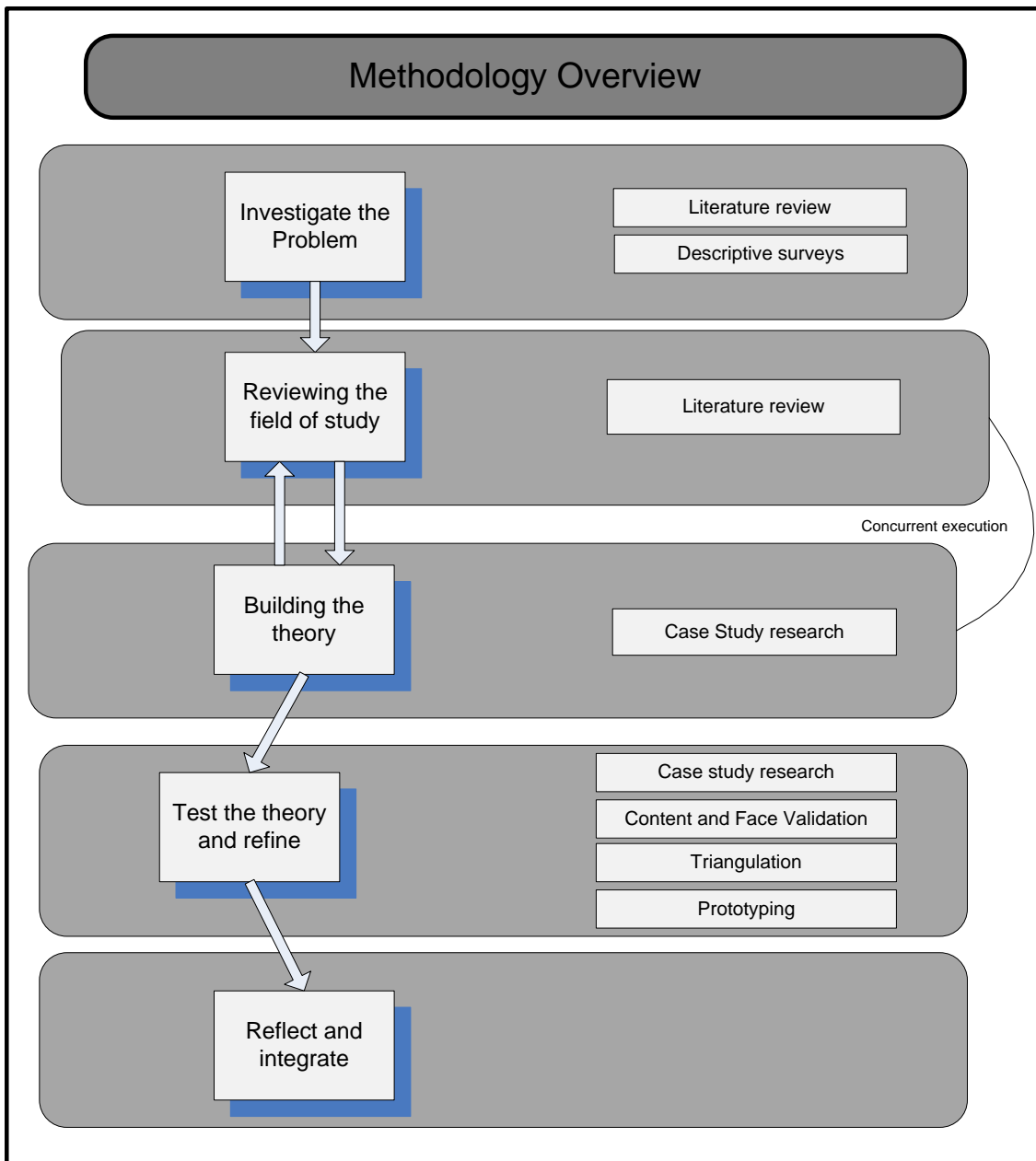


Figure 2.3: Methodology Framework

Case study as a research methodology was selected because of real world constraints. The case studies were available to be used, but the author of the thesis did not have access to the developers for long periods of time and could not interfere with the

development process. Gilbert (1981) writes that this methodology allows a participant-observer to keep a low-profile. Marshall and Rossman (1980) state that in order to understand human behaviour it is necessary to understand the framework within which humans operate and field study research supports the researcher in the exploration of the different processes and identifying the meanings of events taking place.

Case study methodology is often used in the IS environment to accommodate the applied nature of research in this field. A case study methodology allows for the numerical focus in building the theory - which is required here. Using case study research allows the testing of the theory and has empirical validity. The author of the thesis developed a theory which is new and not previously developed. According to Eisenhardt (1989), case study research is well suited for this scenario. It is therefore credible to use this methodology for this research, as it was a standard phenomenon that was being researched and thus allowed the results to be studied.

As stated in the previous section, there are some limitations in using a case study methodology. The author of the thesis tried to overcome these limitations by using multiple case studies and by keeping the theory simple. An attempt was also made to gather as much input as possible from the developers and to adjust the theory accordingly. With the use of multiple case studies and the input from the developers it was possible to prove the credibility of the results. A single case study would have been a better solution, but the research was conducted over a long period of time and for each of the decision making methods, a case study was identified and applied.

Case study research was applied when the four different decision making methods used in the selection of an e-Process were applied in chapter 9. A three phase approach was followed for each of the case studies used:

- In phase 1, the researcher became involved with the problem environment and took an open approach - used interviews, document analysis and observation to gather the necessary data.
- In the second phase the data collected was applied by using one of the decision making methods on the developed theory. Using the lessons learnt for each of these applications the theory was further refined by making changes to the specific decision making method where required or using the lessons learnt to develop the methodology for the next decision making method.
- Phase 3 consisted of interviewing the developers and collecting their feedback. The information-gathering methods used included mostly observation, interviews, and document analysis (Benbasat, et al. 1987; Denzin, 2006; Yin,

1994). In the thesis each of the 4 case studies used briefly defines the background of the problem environment, the process followed in order to execute the specific decision method, and the outcome of each execution applied on a specific case eCIS.

The different case studies are used to determine their similarities, strengths, differences and when to use a specific decision method as well as to refine the theory. The research here is based on modifications of the methodology used for case study research by Eisenhardt (1989), Tellis (1997) and Yin (2006):

1. Design the case study protocol.
 - a. Identify the case study.
 - b. Identify the players.
 - c. Decide on the decision making method to be used.
2. Conduct the case study.
 - a. Prepare the data collection process.
 - b. Collect the data.
 - c. Interview the players.
3. Analyse the data
 - a. Apply the decision making method.
 - b. Discuss the results with the development team.
4. Develop conclusions and analyse the implications.
 - a. Validate the results by:
 - i. Comparison of feedback and results.
 - ii. Applying sensitivity analysis and weak-point analysis where applicable.
 - iii. Applying a second method where possible.
 - iv. Providing rich descriptions where available.
 - v. Doing a cost/benefit analysis to determine the cost involved with the use of one of the decision making methods.
 - b. Do a cost/benefit analysis on the decision making methodology.
 - c. Write up conclusions.

Results from 4a above, shows the validation of the decision making method done in this thesis included the application of different methods on the cases; validation of the decision making methods by discussing the outcome with the developers; providing some rich descriptions and applying sensitivity and weak-point analysis. The validity used in this thesis is a mixture of content and face validity.

The application of the decision making method to a realistic problem in a New Zealand environment appears to have measured what it was suppose to measure (Leary, 2004). Face validity involves the judgment of the researcher or the research participants and is thus not a technical or statistical procedure (Leary, 2004).

Content validity, in contrast to face validity, ensures that the measure includes an adequate and representative list of items that can be used in the concept and is dependent on the theory. By using quality aspects which are grounded in a standard and applying the values for the quality aspects to the decision making methods for the different cases, it was possible to define the construct well and show that the measure used reflected the idea of the construct measurement (Burns, 2000; Trochim, 2006; Sekaran, 1992).

Methodological triangulation was employed; the results were obtained plus the feedback was analysed. Results were replicated where possible (using more than one decision making method to select an e-Process for each of the case studies). Rich descriptions to describe the feedback from the developers were analysed and sensitivity analysis was applied to the results where applicable (Benbasat, et al. 1987; Yin, 1994). As the views of a variety of developers were collected working on different problems, there was limited data triangulation as space and people were involved (Fielding & Fielding, 1986).

Finally prototyping was used, whereby a new systems or programs were constructed, tested and evaluated. The research contained in this thesis includes the description of the developed prototype for e-Process selection in order to allow further development in the future.

2.4 Objective Statement

The primary objective of this thesis was to assist developers of an e-Commerce Information Systems (eCIS) select a suitable process based on a systematic application of the appropriate decision making methods from a set of known alternatives.

The primary objective can be divided into the following sub-objectives:

- Identifying whether development processes are actively used and how development processes are used in an industry based web development environment.
- Applying a set of multi-criteria decision making methods on the selection process and for each performing the following three activities:

- Develop a formal meta-model and then integrate this into one the e-Process selection meta-model;
- Apply the method to a specific case study in collaboration with a real software company and;
- Implement the method in a prototype software tool.

The successful achievement of this is that every decision making method will be formalised and then integrated into a common decision making meta-model as well as tested on an industry based real- world eCIS while providing a prototype tool for support.

2.5 Importance of the Research

The rigor required to develop an eCIS cannot be compared to the rigor required in the development process in order to send someone into space but it is still important to select a suitable development process in order to provide a quality end-product.

According to Amber (2006), if you want to succeed at software development you need to choose a suitable software development method. It is important to spend time and effort in order to choose the right development method for that specific eCIS. There are several reasons why it is important to do so:

- Different technologies and/or projects require different techniques. Some of the object-oriented methods might be more suitable for projects using object-oriented technologies; some of the data-oriented methods might be more suited for data-oriented applications; and some development processes might be more suited for the development of a specific eCIS.
- Development of eCIS is done with a development team. The members of this team are unique individuals with different backgrounds, skills and experience. The team members have different ways of working and solving problems. In order to take this into account the most suitable e-process is required.
- Every eCIS-project will be developed by a unique team. These teams consist of individuals, and will require a unique way for them to work in order to maximize their potential.
- The eCIS has external needs which will be unique for each project. The eCIS might need to conform to government or business regulations. Some of the eCIS being developed depend on suppliers, technology vendors or software development outsourcers, and therefore must tune the development process to reflect these requirements.

- eCIS projects are not all the same. Different eCIS projects require different approaches because each eCIS has different priorities and goals.
- Development methods vary and have different strengths and weaknesses. It is important to determine what works best for that particular eCIS.

(Amber, 2006)

- The e-Process selection might need to take place in order to help motivate a specific choice to the users or owners of the eCIS by the developers. The developers (vendors in this case) might need to convince the owners of an eCIS that a specific e-Process is suitable.
- A specific selection method may identify specific qualification needs of the developing team. It might be necessary to train the developers in the use of a specific e-Process.

It might therefore be important to follow the right development process. It is unrealistic to expect every project team developing eCIS to use the same e-Process. Different developers will create different types of products and in different ways. To develop eCIS is complex and varied and as a result you need to investigate several e-Processes before you select one or a combination of a few if you wish to succeed (Amber, 2006).

There are basically four different categories, according to Amber (2006), into which development processes can be classified:

- No Process: the developers' code and fix as they develop. There is no plan, estimations or schedule and chaos is often the result.
- Well-defined Rigor: Development processes are well-defined with detailed procedures and the rules are expected to be strictly adhered to. The "waterfall methodology" can be seen as an example of this approach.
- Iterative approaches: The different procedures are applied in an iterative manner. Small subsystems are tackled and software is delivered on an incremental basis in short development and implementation cycles. The Rational Unified Process (RUP) is an example of this approach (Larman, 2002; Medvidovic, Rosenblum, Redmiles & Robbin, 2002)).
- Agile approaches: This development process is people orientated. Change is allowed. These development processes are defined on a high level and can be a collection of the same type of practices or philosophies. Agile Unified Process and Extreme programming are examples of these types of development processes (AM, 2001; Ambler, 2002; Beck & Kent, 2001).

The different development approaches have appeal to different people and will most probably suit in different circumstances. It is seldom recommended that no process be used as this can lead to failed projects. Each of the approaches, however, has specific types of projects; specific types of teams and specific types of environments where they will be recommended for use.

There are multiple development processes available for software development. These include the main areas such as agile development, incremental and iterative development, evolutionary development and extreme development. Selecting the most suited process for the problem at hand is not easy.

According to Amber (2006) the different development processes have different levels of rigor and adherence to methodology structures; refer to Figure 2.4.

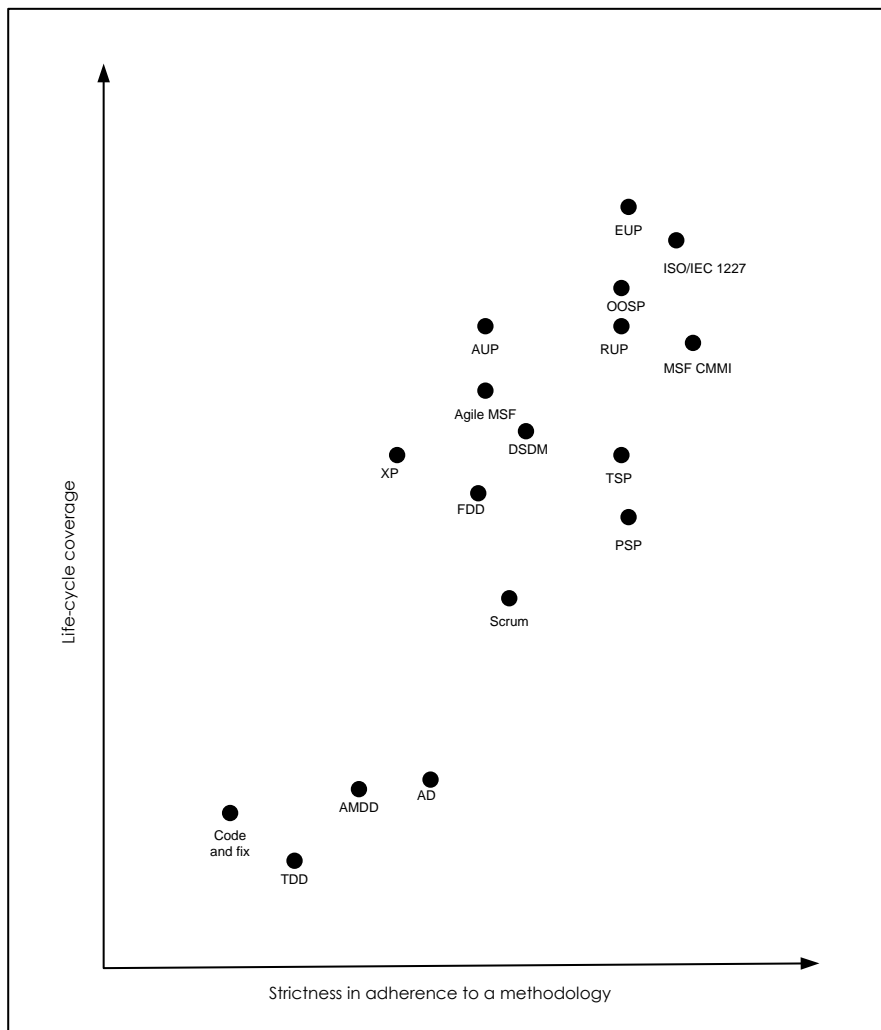


Figure 2.4 Comparisons of development processes. (based on Amber, 2006)

Some of the specific development processes are: Agile Model Driven Development

(AMDD); Agile MSF (Microsoft Solutions Framework); Agile Unified Process (AUP); Data-Driven Approach Dynamic System Development Method (DSDM); Enterprise Unified Process (EUP); Extreme Programming (XP); Feature-Driven Development (FDD); Glacial Methodology; ICONIX; ISO/IEC 12207; MSF for CMMI (Microsoft Solutions Framework for Capability Maturity Model Integrated); Object Oriented Software Process (OOSP); Personal Software Process (PSP); Rational Unified Process (RUP); Scrum; Team Software Process (TSP); Test Driven Development (TDD); Open Sourcing; User-Profiling and Storyboarding (Amber, 2006). This list is not exhaustive, but does identify some of the development methods that appear to be in reasonably common use.

It is not easy to identify and document a suitable e-Process for the development of a specific eCIS but this research endeavoured to support developers in this process by providing them with some tools.

There has been some research into different approaches to development process selection. A number of authors have developed frameworks for the effective selection of software development processes, see Woo, Mikusauskas, Bartlett and Law (2006), Vazquez (1994), Ramsin and Paige (2008) and Knight, Steinbach and Keller (2001). The focus in these papers is on providing frameworks for the different development processes, which can then be used for selection.

- In Ramsin and Paige (2008, p.3.2) it is stated that: "There is a proliferation of methodologies, and it is difficult to choose from among them". This paper continues to discuss different object oriented software development methodologies in terms of their core philosophies, processes, and internal activities. This paper aimed to make the selection process easier by understanding the development process.
- Woo, et. al. (2006) developed an adoption framework for system development methodologies in order to support users in their selection process. The framework uses five main strategic perspectives in the framework, namely industry perspective, organisation perspective, project perspective, product perspective and developer perspective. The author of this thesis used similar factors in the evaluation process; see quality aspects in chapter four.
- In Vazquez (1994) a set of heuristic rules were defined to help in the selection of suitable methodologies for development of information systems.
- A method of eCIS development process selection is briefly targeted in Knight et al. (2001).

The author of the thesis could not, however, find much research on selection of development processes for e-Commerce Information Systems.

2.6 Limitations of the research

One of the main limitations of the research undertaken for this thesis was that it was not possible to apply the four decision making methods on one case. The reason for this was that each of the decision making methods was developed over a period of years and then applied to the case that was available at that time. The cases were not selected to fit the specific decision making method, but rather the case that was available at the required time. The author had contact with industry through the supervision of final year degree students who did eCIS development for industry. These were the eCIS that the author of the thesis had available for use in the research.

It was also not possible to trial each of the four decision making methods on several cases, which would have been the ideal situation, because of industry contacts' time limitations.

There is a large selection of e-processes to choose from. Just reading up or listening to the advice of others is insufficient to determine whether a given method is a good candidate for the eCIS development because the type of eCIS should also be taken into account. The research in this thesis aimed to support the decision makers in their choice. One of the limitations in doing this, however, was the difficulty of deciding which of the e-processes to considered for selection. The thesis used four development processes as examples, but there are a many options to choose from.

2.7 Summary

This chapter introduced the research and included an introduction to the basic problem of deciding on a suitable e-Process to use for eCIS development. The chapter also briefly discussed the methodology used in the research, objectives of what was to be achieved with the research, the importance of the research and some of the limitations of the research.

The rest of the thesis focuses on the e-Process Selection Process as developed by the author. The next chapter discusses some initial research, followed by two chapters which define the quality aspects and the components e-Process alternatives, which is required for this e-Process Selection Process.

Chapter 3

Investigative Initial Case Study

This chapter reports on the findings from the initial investigation into the development processes used in the local New Zealand industry. This chapter goes on to discuss the conclusions that can be drawn on these findings.

3.1 Introduction

This chapter is based on an investigative published study that was done at the start of the thesis research. The research reported here was undertaken in order to investigate the state of eCIS and web development in companies that are involved with this task.

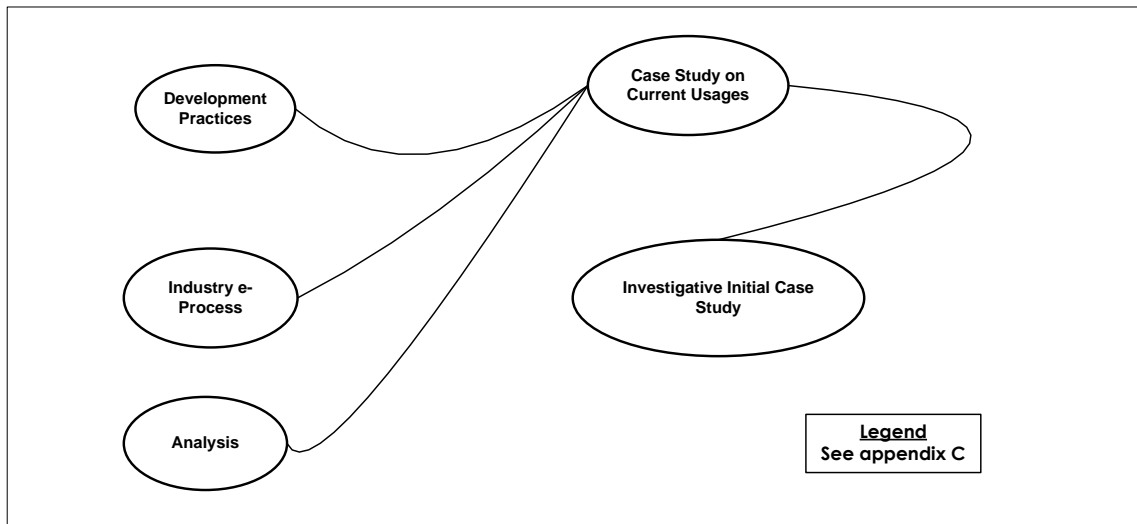


Figure 3.1: Investigative Case Studies Framework

The chapter focuses on the use of software development processes and the tools used in businesses. Figure 3.1 provides a framework for the discussion in this chapter.

3.2 Investigative research

The research presented in this section is based on publications from Albertyn (2004a) and Albertyn (2005a).

3.2.1 Information system development (ISD) process usage

Not many studies are available that provide information about the software development methods and tools used by software engineers in New Zealand. In Groves, Nickson, Reeve, Reeves and Utting (2000) as well as Phillips, Kemp and

Hedderley (2005), surveys were conducted to determine some of the practices of software engineers in New Zealand. The conclusions of these studies were that there was a wide range of types of organisations engaged in software development in New Zealand. These companies are involved with the development of a wide range of applications. One of the most significant conclusions from these studies was that the size of the development group determines the depth and level of the development methods used (Phillips, et. al., 2005). The paper published by Phillips, et. al (2005), which was published after this investigative case study, also widened the focus to include software development practices adopted by different companies. This study found few associations between the size of the organisation, the type of company, the length of the project or the type of applications being developed (Phillips, et. al., 2005). It was however found that developers see the need for integrated tools and feel strongly that the issue of development processes which are adopted is of great importance (Phillips, et. al., 2005).

In the investigative research, presented here, it was decided to employ questionnaires to survey the respondents (Sekaran, 1992; Trueman, 1976; Eisenhardt, 1989). Mainly a quantitative method was adopted with the use of closed questions; however there were a few open-ended questions which the results of can be quantitative in nature. This method was used in order to reach a wider target group than would have been possible using interviews. This method of research also takes less time to administer and allows for more comprehensive analysis of the data than would have been possible using interviews (Sekaran, 1992; Trueman, 1976; Whitten, et al., 2007). In order to determine who needs to fill in the questionnaire a sampling strategy was selected. There are two distinctive types of sampling designs, namely probability sampling and non-probability sampling. When using probability sampling every element in the population as a whole is known and has an equal chance of being selected for the sample. In non-probability sampling, the different elements do not have a known or predetermined chance of being selected for the sample (Cavana, Delahaye & Sekaran, 2001; Sekaran, 1992). The non-probability sampling method called purposive sampling was used. In this type of sampling the researcher decides who is the most suited to completing the questionnaire (Cavana, et. al., 2001; Sekaran, 1992). For this investigative case study it was decided to use a NZ based purposive sample because it was convenient to do so and the subjects chosen to complete the sample were in the best position to provide the information required. This is a viable method because the information required can be obtained from a very specific group. It is however important to remember that this type of sampling is not a very reliable method in terms of generalisation of the information obtained (Cavana, et. al., 2001).

Once the research objectives were defined for the initial study, it was used as a basis for the development of the questions used in the questionnaire. The main reason for this research was to find out how practitioners actually apply and adopt software development processes in their work. This study is not comprehensive and further research in the future will be required to understand the environment better. The study reported in this section involved sending out questionnaires to 55 different practitioners via email. The developers were randomly selected from a directory of business contacts in the computing sector, which has been built up over a number of years in the author's department.

The focus of the questions in the questionnaire was on whether software development companies use development methods and tools and the type of development methods and tools used by the developers in their specific organisations. The questionnaire was made up of seven questions, and appears in full in Appendix D. It contains a mix of general questions, open-ended, multi-choice and ranked questions. This questionnaire was piloted with some local software developers and modified accordingly. The questionnaire gathered general background information, information on the types of development taking place in the company, the size of the respondents work environment, whether team development takes place in that environment, the general size of the development teams, the types of development processes used in that company, the type of development tools in use in that company and a classification of the type of company that the respondent works for. The purpose of the survey was not to determine the full extent of software development usage, but rather whether there is substantial use of development methods and tools in the local environment. The respondents were allowed to expand on some of their answers and many took the opportunity to further elaborate.

This focus of this survey, completed by web developers, was to determine the usage of web ISD processes. This section reports on the findings of this small survey of 31 respondents conducted in March 2004 amongst 55 web developers on their usage of web IS processes. The respondents that completed the surveys were mainly developers from a number of different companies. These respondents completed the survey as individuals, and not necessarily as representatives of their respective companies.

Fifty percent of the developers surveyed had been involved in the development of web pages, whilst 28% indicated the development of large interactive websites and 22%

developed e-commerce sites - of this latter group, 9% used predefined software to do the development.

The respondents indicated that they were working for a range of companies and did development for a range of areas that included the health sector, government, SMEs, tertiary sector and large companies. Some of these companies are software development specific, while the business focus of the others are in a range of different areas.

The work environment of developers included single person development (37%), private company (29%), small-to-medium sized business or SMEs (21%) and larger business (13%). The research presented in this thesis focused mainly on small-to-medium sized enterprises in the New Zealand environment. In most countries the SME sector makes an enormous contribution. In New Zealand, the 350,000 or so SMEs make up more than 99% of all businesses and is responsible for more or less 60% of employment. The SME sector broadly covers micro-enterprises (fewer than 5 staff), small enterprises (6-49 staff) and medium enterprises (50-100 staff) (Ministry of Economic Development, 2007).

A fifth of the people did their development in teams of between 4 and 10 people, whilst two-fifths did their development as part of a team of up to three people. Two-fifths of the developers did individual development.

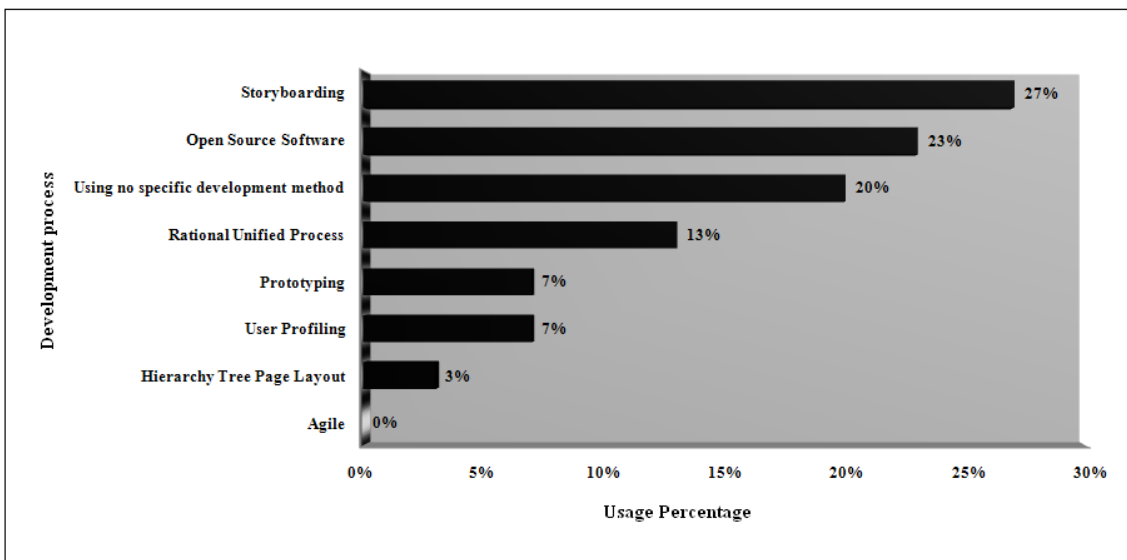


Figure 3.2: Structured development process usage

As seen in Figure 3.2, 20% of the developers did not use a structured development process; whereas 27% used storyboarding, 23% used open source software, 13% used

the rational unified process, 7% used prototyping and 7% use user profiling. Only 3% of the developers used a hierarchy tree layout and no one used agile modelling. There were however three of the respondents that commented on the fact that they would strongly consider using Agile as a development process in the future. This was one of the reasons why Agile is included as one of the development processes used as examples in this thesis, refer to chapter 5.

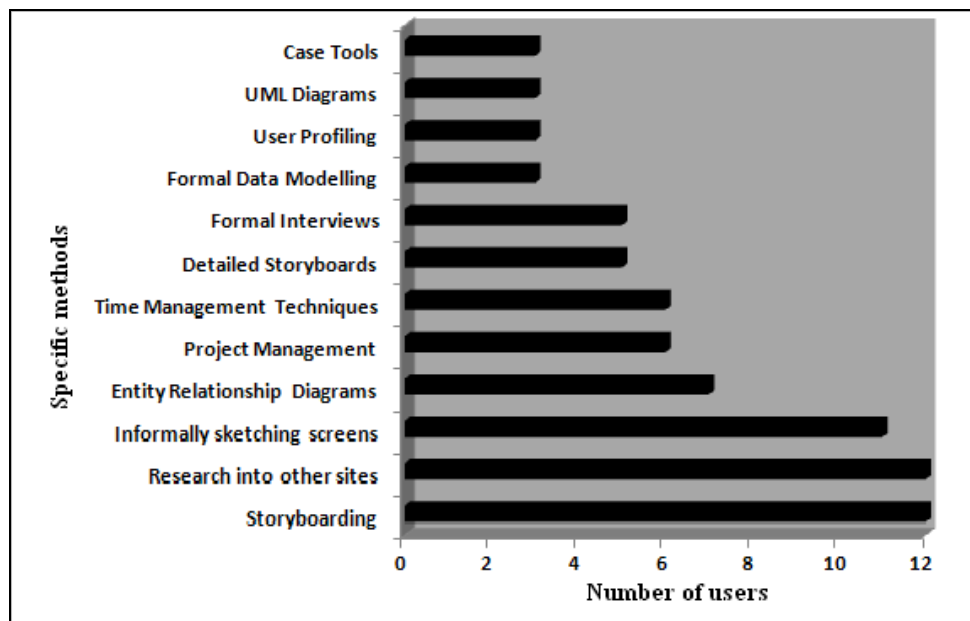


Figure 3.3: Specific Methods used by Developers

Respondents were asked to indicate all the development methods they had used for web ISD (See results in Figure 3.3). Upper most in their responses was storyboarding, sketching screens and research into other sites. The larger teams/projects included data modelling and entity-relationship diagrams as well as project management and time management techniques in their list of modelling methods used. User profiling, UML diagrams and case tools are used by some of the developers, but not extensively.

The feedback received from some of the larger development companies indicated that there is a need to use a formal web ISD process, especially when developing larger sites. Elementary sites require at least extensive prototyping. The developers further commented that for complicated sites, that require extensive database or e-commerce work, there is a definite need to do complete analysis, scoping and documentation before any development work. Functional requirements need to be defined and agreed upon with the user. Some of the companies set up a staging server that allows the client access to the site at will before formal testing and going live.

As seen in this section, there is a definite need for developers of web sites and e-commerce sites to use a formal ISD process. The need exists to provide developers with support in making the decision. The development process used impacts on the quality of the resulting system. It is a difficult task to ensure that the formal web ISD processes have evolved sufficiently to meet the developers' requirements for web ISD processes. Further research into this is required.

3.3 Conclusion

A major aim when developing a new eCIS is to achieve a good quality product. The research in this chapter showed that developers used and need development processes for Web development and eCIS development.

Formal development processes were seen by respondents as necessary for web development. The feedback from the eCIS developers specifically, also indicated the need to use formal development processes. Some of the respondents in this group mentioned that they were not always sure which e-Process to use.

This thesis aims to support in the selection process for e-Processes. In the next chapter the quality aspects used in the thesis research is presented and in the chapter thereafter the four e-Process alternatives to be used as examples in this thesis are overviewed.

Chapter 4

Quality Aspects

This chapter introduces the quality aspects used in the research as one of the components required for selection of e-Processes. It first introduces quality aspects in general and then continues by first describing the high level quality aspects and then the second level quality aspects used in this thesis.

4.1 Introduction

E-Commerce Information Systems development processes can be considered as systems. If this is the case then the aspects that apply to the quality of systems can, to some extent, apply to development processes. Depending on exactly how the term “information system” is defined, humans are part of these or at least are closely related, see Bernus and Schmidt (1998); Hirschheim, Klein, & Lyytinen (1995) for respective definitions. Consequently forming concepts of the quality of development processes and methodologies need to be approached with care. According to Knight et al. (2001) the quality aspects to be considered are organisation, project, and team.

This chapter focuses on the quality aspects that are required as components for e-Process Selection as specified in the presented research. The layout for this chapter can be seen in Figure 4.1.

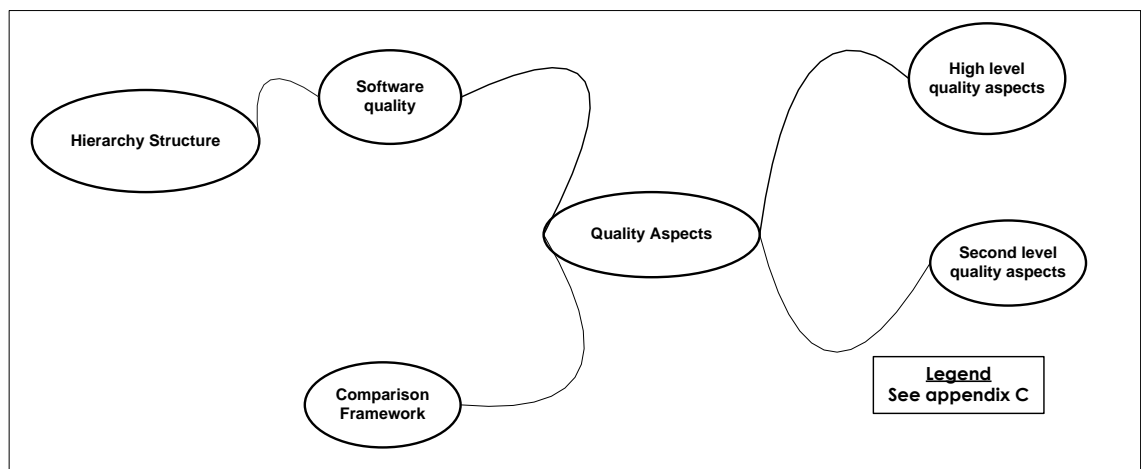


Figure 4.1 Framework for quality aspects

A taxonomy regarding various classification features of methodologies for workflow systems development is targeted in Al-Humaidan and Rossiter (2000). As mentioned in Chapter Two, a method of eCIS development process selection is briefly discussed in Knight, et al. (2001). The selection approach taken relies on the identification of situation patterns best supported with a particular process. This qualitative approach may become inconclusive as several patterns might apply to a lesser or higher degree. To cope with such situations we introduce quantification as the guiding idea for the process selection.

The quantification of quality aspects is based on software quality. The relationship between object oriented software development methods and software quality factors and metrics have been discussed in Biricik, Buharali and Kalipsiz (2003). According to Sommerville (2000) any quality plan should set out a number of desired qualities.

Defining and measuring the quality of systems can be very difficult. The process of improving the quality of already existing systems is also not an easy task. Quality of a system, based on the work of the international standardisation organisation ISO (ISO-standard, 1991), could be defined as the degree to which the consistency of that system makes it suitable for its intended (or unintended) use. This definition involves the somewhat vague concept of "system consistency". The use of systems may have quite a lot of different aspects to consider. Consequently measuring and maintaining system quality according to this definition is not really practical. (Prioritised) lists of so-called quality aspects (see Barbacci, et al. (1995)), might be much more practical to use. These may be more or less aggregated, for example, focusing on correctness and efficiency (Schach, 2002).

Quality related discussions in information systems development often avoid quantification. They therefore rely on development processes and methodologies. These were put into the focus of the quality debate after the United States Department of Defense realised, about 30 years ago, that quality improvement is possible by managing the development process properly and applying suitable development methodologies. The Software Engineering Institute (SEI) was set up at the Carnegie Mellon University in Pittsburgh and the Capability Maturity Model (CMM) was developed to encourage companies to improve their development processes, (Schach, 2002; Whitten, 2000). Development processes were supposed to be used, applied in standardised form, and improved. The degree of standardisation and sophistication of applying a development process was used as an indicator of the system development capacity.

The task of developing information systems is known to be difficult. A very strong focus is being placed on process-oriented aspects of information systems development for obtaining high quality systems (Kautz, et al., 2004). If non-suited processes are used the quality, in particular the utility, of the information system under development is likely to be unsatisfactory (Sommerville, 1996). According to Butler (2003) many issues in the development of non-Web information systems are also present in the development of Web-information systems.

Jayaswal and Patton (2006) introduces a list of quality aspects, which forms the initial basis for our list of quality aspects that will be used for e-Process selection.

Hierarchy Structure of Software Product Quality aspects						
Software Quality						
Usability				Maintainability		
Portability	Reliability	Efficiency	Human Engineering	Testability	Understandability	Modification Capacity
Hardware Independence	Self containment Correctness Completeness Integrity Freedom from conflict	Measurability Equipment efficiency Accessibility	Integrity Accessi-bility Communi-cative-ness	Measurability Accessibility Communica-tiveness Self-descrip-tiveness Structure	Self-descrip-tiveness Structure Simplicity Clarity	Structure Expanda-bility

Table 4.1: Hierarchy Structure of Software Product Quality Aspects

The quality of an information system is often measured by using the ISO9001 quality standards as set and maintained by the International Organisation for Standardization (Bamford & Deibler, 2003; Dalglish, 2005). These standards form a formal quality policy for the management of IS, business and marketing plans and customer needs. All levels in the company need to understand and follow these policies. These standards form the basis for all decisions made regarding quality systems; recorded data; the auditing processes followed for measuring conformance and effectiveness (Bamford & Deibler, 2003; Dalglish, 2005). What is important for the research in this thesis is that the ISO 9001

standards also provide a framework for the staged development of new information systems and the testing required for each phase of the development. The standards are used for performance reviews and to help determine the quality of the end product. They also help to document procedures followed (Bamford & Deibler, 2003; Dalgleish, 2005).

There is an advantage to using quality aspects for the selection process as they focus quality discussions on system usage. No list of quality aspects for information systems, in general, will ever be complete. As eCIS and the use thereof are better understood, it might be an advantage to reflect on this new knowledge in refined or extended terms in order to discuss systems quality. It is also important to remember that with respect to a given development task some of the quality aspects may be important while other are less or not at all important.

The quality of e-Processes is multi-faceted (as is the case for information systems in general (Ghezzi, Jazyeri and Mandrioli, 2004)). Individual quality aspects are often considered as too broad and unspecific and are therefore decomposed into lists of lower level quality aspects.

ISO9001 quality standards and hierarchy structure of software product quality aspects (Jayaswal & Patton, 2006) are used as a basis for defining the quality aspects used in the thesis. The author also exploits the two-level hierarchy in Kaschek et al. (2006) and integrates and extends the approach in Kaschek and Mayr (1998). This work was adapted for selecting from admissible business process modelling methodologies (Kaschek, Pavlov, et al., 2006).

For the selection method the employed quality aspects are irrelevant, i.e., any other set could be chosen as well by allowing some of the quality aspects to be ignored by setting the weights to 0. The outcome of a specific selection, however, is likely to critically depend on the chosen quality aspects. Due to this focus on illustrating the method the author does not argue in particular for the chosen quality aspects.

In the next sections the high level quality aspects and second level quality aspects used in the thesis are introduced.

4.2 High level quality aspects

It is proposed that a structured list of quality aspects is used to choose between different e-Processes. The high level quality aspects, for this research, are defined as the following (loosely based on (Kaschek, et al., 2006)):

- e-Process aspects, i.e. the modelling notions, abstraction concepts, and other aid suggested by the e-Process;
- Quality concepts of the e-Process (its reliability, robustness etc);
- Cost for using the e-Process;
- Domain impact, i.e. the impact of the project domain;
- Usability, i.e. the e-Process aids in developing a high quality eCIS;
- Compatibility of the e-Process with other methodologies, and
- Maturity, i.e. e-Process stability, tool support, documentation etc.

The high level quality aspects provided in this section are decomposed in the next section into second level quality aspects.

4.3 Second Level Quality Aspects:

The second level quality aspects are the following and are also based on Kaschek, et al. (2006):

The quality aspect **e-Process Aspects** is decomposed into:

- Completeness, i.e. the degree to which the e-Process provides means of expression (such as modelling notions, abstraction concepts, patterns, and anti-patterns) that enable the eCIS developer to effectively and efficiently solve development tasks within the domain of application of the e-Process.
- Understandability, i.e. the degree to which one easily understands the e-Process.
- Visibility, i.e. the degree to which the defined activities of the e-Process results in clear outcomes and effective and efficient project management is enabled.
- Supportability, i.e. the degree to which CASE tools are accessible that aid in using the e-Process.
- Maintainability, i.e. the degree to which the e-Process enables managing requirements in particular with respect to change requests after project commencement.

The quality aspect **Quality Aspects** is decomposed into:

- Readability, i.e. the degree to which the notation prescribed for use in the e-Process is easy to read.

- Reliability, i.e. the degree to which the e-Process is designed in such a way that errors in the development process are avoided or identified and fixed prior to system deployment.
- Robustness, i.e. the degree to which the e-Process continues to aid developers in case of unexpected events occurring.

The quality aspect **Cost** is decomposed into:

- Development Budget, i.e. the degree to which a tight development budget does not tend to reduce the applicability of the e-Process;
- Running Costs, i.e. the degree to which the eCIS developed with the e-Process will run according to allowed budget.

The quality aspect **Domain Impact** is decomposed into:

- Infrastructure, i.e. the degree to which the technical environment of the enterprise affects the applicability of the e-Process.
- Enterprise Culture, i.e. the degree to which matching the e-Process to the enterprise culture does not reduce the applicability of this e-Process.
- Technology, i.e. the degree to which matching the e-Process to all other technology being used in the enterprise does not reduce the applicability of this e-Process.
- Geographic Interaction, i.e. the degree to which the globalisation of the enterprise is necessary for the e-Process being applied.
- IT Strategy, i.e. the degree to which the enterprise's system development standards (specifically eCIS development) do not reduce the applicability of this e-Process.
- Business Strategy, i.e. the degree to which the enterprise's business strategy does not reduce the applicability of this e-Process.
- Team Experience, i.e. the degree to which the involvement of the team with previous eCIS development affects the applicability of this e-Process.
- Domain Knowledge, i.e. the degree to which the domain knowledge of the development team affects the applicability of this e-Process.
- E-Process Knowledge, i.e. the degree to which the knowledge of the e-Process of the development team affects the applicability of this e-Process.
- Development Time, i.e. the degree to which the development time influences the applicability of this e-Process.

The quality aspect **Usability** is decomposed into:

- Functionality, i.e. the degree to which the e-Process specifies the required development artifacts (such as requirements, design, implementation, testing, etc.) and instructs how to create and use them.

- Manageability, i.e. the degree to which the e-Process aids managing projects effectively (including planning, tracking, and risk management etc.).
- Quality assurance, i.e. the degree to which the e-Process aids developers in following the principles, requirements, and recommendations of the Total Quality Management announced at ISO 9000:2000, ISO 9001 and ISO 9004.
- Adjustability, i.e. the degree to which the e-Process can be adjusted to meet the specific needs of the eCIS project in the most effective way possible.

The quality aspect **Compatibility** is decomposed into:

- Exchangeability, i.e. the degree to which the artifacts of the given e-Process can be exchanged between the tools implementing this e-Process.
- Mappability, i.e. the degree to which the artifacts of the given e-Process can be mapped into the artifacts of other e-Processes.

The quality aspect **Maturity** is decomposed into:

- Stability, i.e. the degree to which the e-Process has been proven, i.e., its standing.
- Tool support, i.e. the degree of availability of the tools supporting this e-Process (such as version control and document/workflow management systems) and the quality of these tools.
- Documentation, i.e. the degree to which adequate documentation is available.

4.4 Conclusion

The high level quality aspects and second level quality aspects presented in this chapter will be used in chapter 7 and Part C of the thesis as components for the selection of e-Processes by applying different decision methods.

Chapter 5

E-PROCESS ALTERNATIVES OVERVIEW

This chapter discusses the different e-Process alternatives used in the research. The chapter first introduces e-Commerce Information Systems, and then describes the four e-Process alternatives. The chapter concludes with a comparison of the four e-Processes.

5.1 Introduction

Simply stated, the purpose of this chapter is to introduce eCIS and discuss the different e-Processes used in the research into e-Process Selection. This chapter includes a comparison of these e-Processes, which are used as alternatives in the selection process. The e-Processes discussed here are used as examples and there are many other e-Processes that could be used in the e-Process Selection.

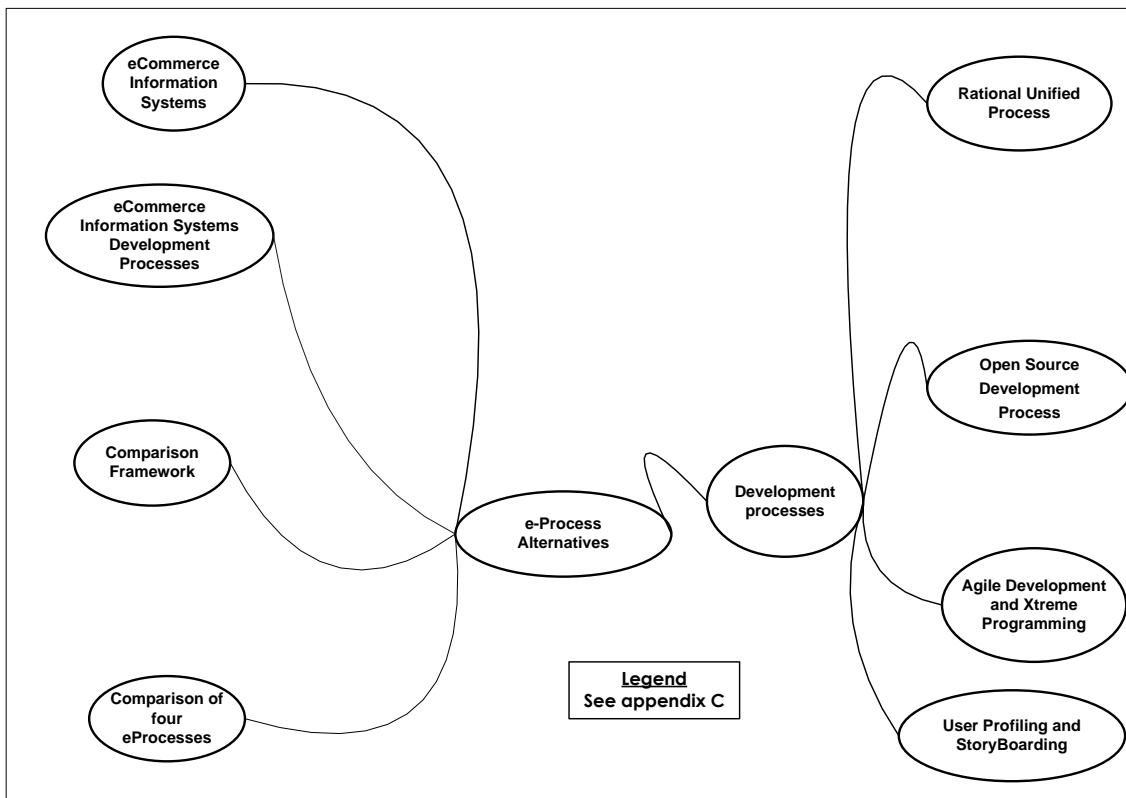


Figure 5.1: Framework for the e-Process alternatives overview

The e-Process selection method was developed with a specific type of Information System Development in mind, namely e-Commerce Information Systems (eCIS). In the research conducted for this thesis four different e-Processes were used in all the case

studies. This chapter briefly describes each and then compares them. Figure 5.1 briefly summarises the topics covered in this chapter.

5.2 E-Commerce Information Systems

E-Commerce is doing business transactions via telecommunications networks. The customer and the merchant are in different geographical places, without prior contact between the parties, when the transaction takes place (Mattila, 1998). This includes viewing goods on sale over the internet, selecting goods to buy, and processing or using different payment methods (Gruhn & Schope, 2002). Software developers need to develop applications quickly which results in a lack of process discipline and procedural guidance. This highlights the need for a framework to support practitioners' development efforts (Antón, et al., 2001). It is important to ensure high productivity and excellent quality from a selected e-Commerce Development Processes (e-Process) for an e-Commerce Information System (eCIS) (MacCormack, et al., 2003; Mansar, et al., 2005).

The efficiency of eCIS development (producing quality eCIS based on an efficient e-Process) can be controlled by quantification of the development methodologies (Dumke & Foltin, 1996). Comparison can be achieved by using a list of features and addressing quality issues such as efficiency, maintainability, portability etc (Dumke & Foltin, 1996).

Software development and specifically e-commerce development can either be easy or complex and are sometimes developed without using a defined methodology. This can potentially lead to failure of the implemented Information System and can be catastrophic for the business (Astels, Miller & Novak, 2002; Kaschek, Schewe, Wallace & Matthews, 2004).

E-commerce Information Systems require different approaches than the classical system development processes as well as other types of resources. The impact from this is that knowledge of processes and methodologies is not sufficient and some support is required (Janssen & Steen, 2003). Noticeable differences between the methodologies required for a normal information system development and an e-commerce system development are:

- Software used for an e-commerce site will pertain to e-commerce software and the e-commerce site will be published on the Internet.

- In e-commerce systems, the most important ingredient is the customer, which cannot be identified before they, the customer decides to access the e-commerce site.

When developing an e-commerce site the focus will be on two parts namely the front-end interface which is used to communicate with the user and the back-end, which provides the capabilities necessary to capture and process the customers' orders, control the inventory, and process the product distribution. The next section identifies the e-Processes that are going to be used as alternatives.

5.3 eCIS Development Processes

Traditional software engineering provides a number of frameworks that support developers (see section 2.5) when selecting development processes for Information Systems. One such is Vazquez (94) which states that there are a number of factors when selecting a methodology for the development of the project. Some of these are the complexity of the system; implementation environment e.g. object oriented; the level of knowledge and experience of the staff that will develop the system; the amount of resources (software libraries or repositories) available and; the required performance of the system (All of these were incorporated in the quality aspects being investigated as discussed in chapter 4). Modern systems may have multiple architectures and languages, and therefore the use of several methodologies is appropriate (Phillips, et al., 2005).

As stated in chapter 1, there are not many e-Processes that are specific to eCIS, but it can be seen that many of the existing development processes, traditional or more recently developed can be applied in an eCIS development environment. For this reason a number of representative e-Processes were identified for use in our decision models in order to enable proof of concept. The development processes chosen do not present all groups of development processes, but was chosen to supply a spectrum of choice. The idea with the methodology being researched is that the e-processes to be investigated for comparison and selection purposes, be added to the list by the developers at a later stage as required.

In the thesis four different e-Processes are used to illustrate the different decision making methods. These e-Processes are:

- Rational Unified Process (RUP);

- Open source development process – the “bazaar” approach (Raymond, 2001) (OSS);
- Agile and extreme programming eCIS development process (AX);
- Development process using storyboarding and user profiling (SBUP).

The reasons for choosing these four development processes are the following:

- The Rational Unified Process was chosen because it is used widely in industry as a formal development process (Kruchten, 2001; RUP, 2002) and was also strongly supported as a choice for web development in the initial case study – refer to chapter 3. This development process has extensive support and is well documented.
- Agile software development is seen as a group of software development methodologies which include Extreme Programming (XP), Scrum, DSDM and Agile Unified Process (AUP) (Astels, et al., 2002). This group of development processes is based on iterative development and the solutions are developed by the use of cross-functional teams (Astels, et al., 2002). Extreme programming is a type of agile software development (Astels, et al., 2002). Using Extreme programming is very successful because the emphasis is on customer satisfaction and the developers focus on communication, simplicity, feedback, respect, and courage. Agile modeling is a practice based methodology used to effectively model and document software systems in a lightweight manner using its values, principles and practices – see section 5.6 (Ambler, 2000; Astels, et al., 2002). As extreme programming is sometimes seen as not having sufficient design and documentation the strengths of Agile modeling was combined with the strengths of extreme programming for this thesis. Agile and extreme programming was chosen because discussion with developers showed that many developers would like to use this development process. Their reasons included the speed of development, flexibility and ease of use which might potentially make this a future e-Process (AM, 2001, Ambler, 2000, Astels, et al., 2002).
- An Open Source Software Development Process used by the open source community was chosen to include a development process that is free, re-distributable and can accommodate small or large team development (Feller & Fitzgerald, 2000) as well as being strongly supported in the initial case study (refer to chapter 3).

- Storyboarding and user profiling was included in the list of development processes because it includes more eCIS specific techniques and seems to be widely used by web and eCIS developers either as individual processes or as a combined development process and had fairly strong support in the initial case study (refer chapter 3).

All of these chosen development process can be applied as e-Processes, can be applied to front-end development as well as back-end development, hopefully provide a spread and variety of development processes for assessment, can be legally supported as a choice and will allow for different work cultures to be accommodated.

This section of the chapter investigates the differences and similarities between these e-Processes. In the next paragraph an overview will be given of the Abstraction Layer Model (ALM), which will be used here as a framework for comparison.

5.4 Framework for Comparison

In order to compare different development processes, a framework for comparison needs to be identified. IS analysis and design is often based on the Abstraction Layer Model (ALM) seen in Figure 5.2.

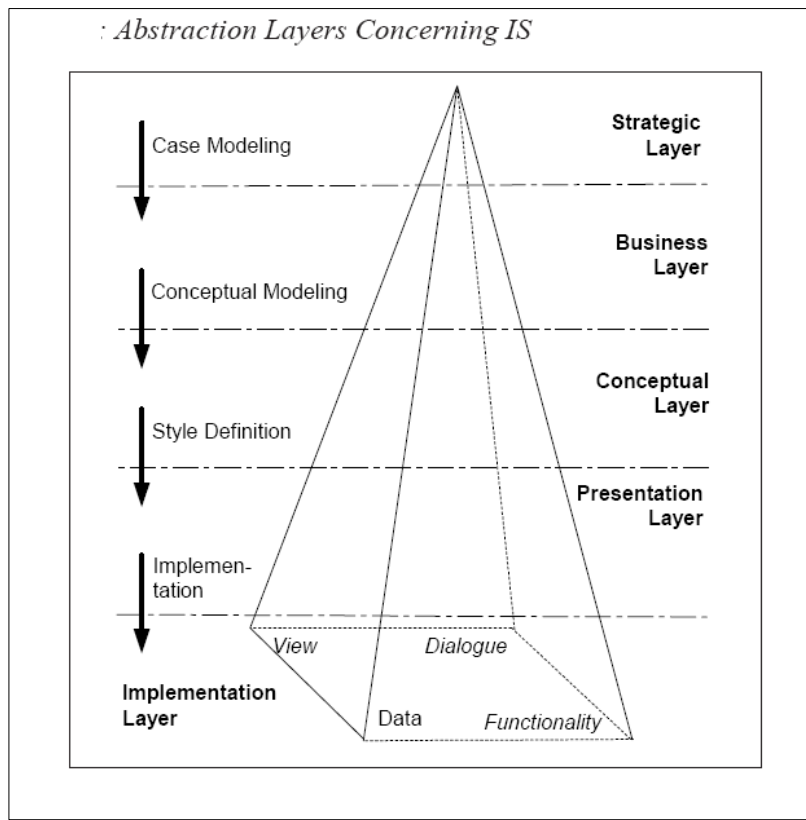


Figure 5.2: IS development abstraction layers

(Kaschek, Schewe, et.al., 2004)

ALM introduces five layers of abstraction for identifying the activities and tasks occurring during IS development. These five layers are from top to bottom:

- Strategic layer: In this layer the system purpose and expected customers are identified;
- Business layer: The business processes (input-output requirements) required for the IS are addresses;
- Conceptual layer: The conceptual model of the IS is developed using the input-output requirement identified in the business layer;
- Presentation layer: This layer allocates access channels to system resources.
- Implementation layer: This layer addresses all the different implementation issues.

In the next sections the different development processes are briefly described before the comparison will be done loosely based on the ALM model as well as the high level quality aspects and second level quality aspects as described in the previous chapter.

5.5 Rational Unified Process

The Rational Unified Process has emerged as a popular development process for building object-oriented systems. RUP is an iterative process (Larman, 2002; Medvidovic, Rosenblum, Redmiles & Robbin, 2002). The RUP is aimed at guiding the development process and using software engineering processes. RUP has a well-defined structure and uses an object-oriented approach. RUP provides the whole development environment using UML (Unified Modelling Language) as a basis. The development of RUP was started in the 1990's.

As seen in Figure 5.3 on the next page, the two dimensions of the RUP are organised as follows: the dynamic aspect (horizontal) expresses cycles, phases, iterations, and milestones; the static aspect (vertical) expresses activities, disciplines, artifacts, and roles (Bloomberg, 2000; Graham, 2001; Kruchten, 2001). RUP is divided into 4 phases namely inception, elaboration, construction and transition. A software product is developed using a number of incremental iterations. The Rational Unified Process involves five different views of the systems architecture namely Use-case view, Logical view, Implementation view, Process view and Deployment view (Reed, 2002).

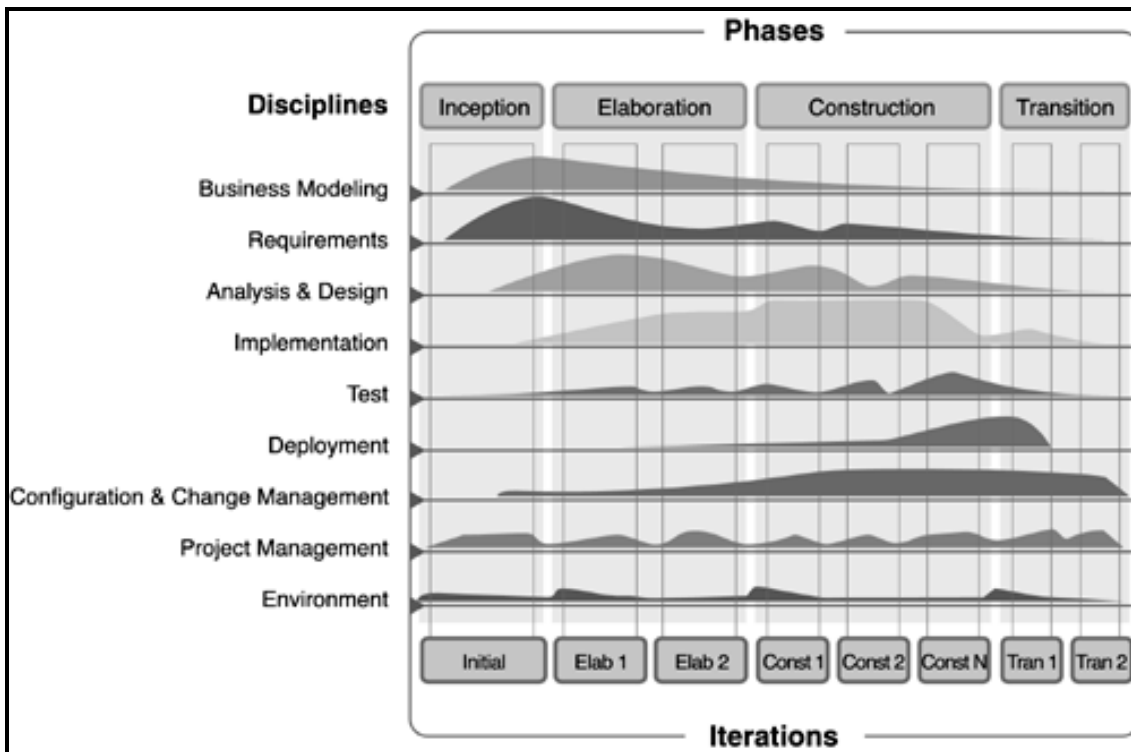


Figure 5.3: The RUP dimensions and phases (Kroll & Kruchten, 2003)

The task of developing a system using the Rational Unified Process can be seen as the following:

Inception:

Envision the project scope, vision and business case. Determine whether the stakeholders agree on the vision of the project and whether the vision feasible. The artifacts produced will be:

- Vision and Business Case: Describes the basic goals, constraints and provides a summary.
- Use-Case model: Functional requirements and non-functional requirements.
- Extra requirements
- Glossary with key domain terminology
- Risk List and Risk Management plan: business, technical, resource, schedule risks and possible resolutions
- Prototypes and proofs: clarify vision and validate technical ideas
- Iteration plan: Plan first iteration of elaboration.
- Phase plan and Software development plan: Plan the tool, people, education and other resource requirements.
- Development Case: Describe and customise steps of process for this application.

Requirements are the capabilities and conditions to which a project must conform to please the client. It should be functional, usable, reliable, have the necessary performance and be supported. A use case is used to specify the requirements.

Elaboration:

The following artifacts will be produced in this phase:

- Domain model: domain concepts visualised and static information model of domain entities.
- Design model: Diagrams to describe the logical design (class diagrams, object iteration diagrams, package diagrams).
- Software architecture document: Key architecture issues.
- Data model: Database schemas and mapping strategies between object and non-object representations.
- Test model: What and how?
- Implementation model: Actual model: source code, executables.
- Develop use-case storyboards and user interface prototypes, which describe the user interfaces, paths of navigation and usability models.
- Draw system sequence diagrams in different iterations.
- Define and name events and operations.

Construction:

Elaboration ends when the high risk factors have been resolved. The design aspects have been solved. Build the product and start developing user guides and online help.

Transition

The system is now ready for operational deployment. The developer must complete user guides and develop training materials. Data must be converted for use in the life system. Market and implement the system.

(Larman, 2002; Medvidovic, et al., 2002; Reed, 2002)

Some authors believe that the Rational Unified Process has been over-developed, but the supporters of RUP believe that this development process dominates the market place (Burt, 2001; Hesse, 2002; Reed, 2002; RUP, 2002; Schewe, 2000; Sundsted, 2001).

The next section will briefly discuss the Agile development process.

5.6 Agile methods and extreme programming

Agile modelling (AM) is a practice-based methodology for modelling and documenting software-based systems. The Agile Alliance (AM, 2001) promotes interaction and individuals over tools and processes; working software over extensive documentation; customer collaboration over contract negotiation, and response to change over sticking to the plan. AM focuses on a portion of the whole development process and needs to be used with Extreme Programming. The idea is to start with Extreme Programming and incorporate the Agile Process into it (AM, 2001; Ambler, 2002; Beck & Kent, 2001).

The values of AM are communication, courage, feedback, humility and simplicity. The principles of AM include model with a purpose, assume simplicity, embrace change, incremental change, multiple models, quality work, rapid feedback, software production is the goal, model with a purpose, know your tools and models and maximise stakeholder investment.

Some of the best practices for AM are stakeholder need to actively take part, collective ownership, create a number of models in parallel, apply the right artefacts, depict models simply, iterative processes, prove with code and apply standards (Ambler, 2002).

The core practices of AM are to:

- Have active stakeholder participation in the whole development process.
- Apply the relevant models and artefacts to the correct application.
- Everybody owns the whole project and is allowed to work on any of the parts.
- Promote quality assurance and develop tests before developing the software.
- Develop several models in parallel.
- Keep requirements, models etc. as simple as possible.
- Place developed models on a wall where they will be visible to the whole development team.
- Change focus to another part of the project if you get stuck on anything.
- Model small portions at a time.
- Communicate your ideas to others and get their input.
- Prove ideas with code.
- Use basic tools such as a whiteboard and basic drawing tools.
- Apply standards to the development process.

(AM, 2001)

Business people and software developers often see the traditional software development methods as too slow and therefore prefer to use the faster Agile type of software development processes (Astels, et al., 2002).

The phases used for AM are the following:

- Conceptualize the system.
 - o Create a vision of the system.
 - o Write the necessary user stories
 - o Develop the acceptance tests to be used.
 - o Find a basic solution to the problem.
 - o Check that the best solution has been used.
- Plan the new system.
 - o Do the estimations for the project.
 - o Plan when the different phases will be released.
 - o Develop an iteration schedule.
 - o Do some tactical planning.
- Develop the system.
 - o Program the different parts in pairs.
 - o Do continuous testing according to a test framework.
 - o Design using the Agile values, principles and practices.
 - o Develop the code and refactor.
 - o Integrate on a daily basis.
- Deliver the system.
 - o Deliver the system into production.

(Ambler, 2000; Astels, et al., 2002)

Extreme modeling is a type of Agile software development method which aims to combine the advantages of methodologies based on the Unified Modelling Language with the advantage of Extreme Programming (XP). The best practices of UML are combined with the flexibility of developing and testing XP code. The philosophy of XP is to invest just enough effort to understand what it is that you intend to build and then build to see whether your design is right (Ambler, 2000; Boger, 2002; Astels, et al., 2002). Extreme programming embraces the following 10 best practices (of which 7 are included in the Agile Manifesto): modular, iterative, incremental, time bounded, parsimonious (require a minimal number of activities to mitigate risks and achieve goals), adaptive, convergent (worthwhile risks are actively dealt with), people orientated, collaborative and complementing (Astels, et al., 2002). In this thesis the

values, principles and practices of Agile software development are combined with extreme programming as a development method.

Developers are often responsible for a company's decision to adopt AM (XP) as a development environment. It is important to convince managers that this development process has merit and depth.

5.7 Open source development process

The Open Source Software (OSS) development approach is based on the idea of using software released under a license as defined by the Open Source Initiative. The Open Source Initiative (OSI) is a non-profit corporation that manages and promotes Open Source Software. This software is free, re-distributable and can accommodate unlimited users and usage. The source code is available and can be modified to suit the requirements for the development process (Feller & Fitzgerald, 2000; OSI, 2003).

The following principles are involved with the OSS development approach: programmers and developers have the freedom to innovate and modify code as required. Developers are potentially a large number of volunteers. The programmer has the ability to distribute modifications to software to others free. A good developer knows what code needs to be created new and what code can be re-used. It is important that the developer responds to user requests very quickly (Mockus, Fielding, & Herbsleb, 2002).

The following steps have been identified:

- Define roles and responsibilities;
- Identify work to be done;
- Assign and perform the development work;
- Prerelease testing;
- Inspections and;
- Managing releases.

(Mockus, Fielding, & Herbsleb, 2002)

Open source software development is an idea which has reached maturity. It is being used by the commercial world more and more (OSI, 2003).

5.8 User Profiling and Storyboarding

According to Binemann-Zdanowicz, Kaschek, Schewe and Thalheim (2004) it is important when developing an eCIS that the modelling usage processes require an expressive semantic model to develop a conceptual framework for the eCIS. According to Binemann-Zdanowicz, et al. (2004) and Schewe and Thalheim (2001) this can be achieved by using user profiling together with storyboarding. In Schewe, K-D, Kaschek, R., Matthwes, C. and Wallace, C. (2002) it is suggested that the storyboarding be combined with scenarios to further help with the specification and that the user profiling is applied by using user dimensions to capture the various aspects of the characteristics of the users. This combined usage of storyboarding and user profiling in the area of on-line loan systems was applied in Schewe, et al. (2002). The next two sections briefly discuss these two techniques and their application as development process.

5.8.1 User Profiling

Web searches and e-commerce web sites usually return results without any regard to the concepts in which the user is interested. One of the emerging ways to do eCIS development is using user profiling to determine the users' likes and dislikes in order to develop a site that will be personal and acceptable to the customer (Callum, et al., 2007). User profiling can make a huge difference to an eCIS site when, instead of everyone seeing the screen, the content of each page can be automatically tailored to fit the surfing interests of each user. Profiling can be done automatically using non-invasive approaches such as commercial software and services, or the information can be collected by asking the users to specify their interest explicitly (Callum, et al., 2007).

When determining a user profile it is important to determine the type of user first. The other aspects that need to be investigated include a description of the scenario of the eCIS to be developed, a description of the character of the eCIS as well as the customer, quality aspects of what the customer expects from the eCIS, the reasons that the customer will have to visit the eCIS, what the customer is interested in, what the customer is looking for, the web experience and expectations of the customer and a list of all tasks which will each describe specific goals that user wishes to achieve (Cartwright, 2008; Sigmer_Technologies, 2008).

There are two reasons for doing user modelling and profiling:

- Differences in individual users' needs, and

- The diversity between different groups of user.

(Razmerita, et al., 2003)

Building user models and doing user modelling forms an integral part of making the eCIS personal and ensuring that the eCIS contains feature integration. These aspects allow the development of an advanced eCIS (Razmerita, et al., 2003).

According to Australian Government ICT strategy (2004) the concept of user profiling requires the capturing of the needs, goals, values, expectations and habits of users into well-defined user groups. The different categories of data that need to be captured includes: demographics (such as age, gender, occupation, and education), web behaviour (such aspects as online usage habits, experience, and ability), user needs analysis (including issues, concerns, and information requirements), user goals and expectations (including how they currently satisfy these needs and accomplish the goals, barriers or problems, including perceived and real problems in accomplishing the stated goals) and diversity (the special needs that affect users).

Using the user needs analysis helps the developer to better understand the users' issues and concerns, the typical tasks that the user performs, the users' information requirements, how frequently the user will perform common tasks, and any constraints that may exists. When developing the new eCIS it is important to ensure that the information needs of the target audience are captured as part of the development process to help tailor the general eCIS portal for the various specific user groups (Australian Government ICT strategy, 2004).

The types of user goals that need to be investigated when doing user profiling includes their experiences and feelings when using eCIS, the general outcomes and benefits acquired by the user when they use or are going to use the eCIS. These outcomes include components such as whether they saved time or money, efficiency of the eCIS, quality of the eCIS and whether their problems will be solved. Finally the practical result (did they buy and pay for goods, etc.) of using the eCIS should be investigated (Australian Government ICT strategy, 2004).

The information necessary for the user profile is often acquired by the use of a questionnaire. Other options include observation or interactive methods such as interviews, behavioural observation or using focus groups (Australian Government ICT strategy, 2004).

The information that is important for building the user profile includes the users' experience level in the use of computers, the Internet and e-Commerce sites, the preferences that the user might have, and the market segment that the eCIS will be targeted at. The market segment can possibly be classified in terms of the age of the users, the gender of the users, educational levels and possibly the type of work they might be doing. It is also important to capture the possible physical and/or cognitive aspects of the user that might have an influence on their patterns of usage. The user requirements variations can only be taken into account if the developer group has identified the issues. By executing a user profile and capturing the relevant data a basis is formed which allows for usability scenario development (Australian Government ICT strategy, 2004).

The aim of defining a user scenario is to understand and document the circumstances in which the user would be likely to choose the particular eCIS. The ideal is to build an eCIS in such a way that the user is satisfied with their experience on the eCIS pages and that the contract between the buyer and seller is executed successfully (Australian Government ICT strategy, 2004).

A user profile scenario should have the following requirements:

- Should be designed to include user group issues and unique requirements,
- Should include the high-level goals from the user's perspective,
- Scenario should be reasonable and realistic, and
- Should not use keywords that will actually be used in the eCIS.

(Australian Government ICT strategy, 2004).

When developing the user profile scenario the following steps should be followed:

- First identify the nature of the eCIS goals to be identified e.g. key metrics and qualitative data about the user group.
- Identify the right tasks concerning the information requirements.
- Determine all eCIS features that require testing e.g. main navigational elements, transactional features, new subsites or pages, and new or redesigned features, as well as any problems that may occur.
- Phrase tasks as directed or non-directed scenarios using site features.
- Determine sequence of tasks.
- Check wording of tasks to ensure that user will not be led or misled.

(Australian Government ICT strategy, 2004).

When developing scenarios that can be tested, the users need to show that they can move through numerous pages and navigate to the required eCIS or through the

pages of the eCIS to show the navigational usability of the sites (Australian Government ICT strategy, 2004).

User Profiling and storyboarding have been combined in this thesis as a single e-Process. It is important to understand the user and build their profile taking their preferences into account, as seen above. It is also important to develop an eCIS which is user-friendly and logical, using storyboarding as described below which is the visualisation of what the eCIS is going to look like.

5.8.2 Storyboarding

Developers use storyboards, which are graphically organised pictures (Stevens, 2008), to pre-visualise what they are trying to design for the eCIS. These pictures are sequences of images organised in such a way that the story is visibly sketched. Storyboarding, which was originally used to design the layout for cartoons for the Walt Disney group (Storyboard, 2008), has recently evolved into a useful eCIS development tool. This technique is used to visually plan the eCIS as well as to provide a visual method to communicate with the user. According to Stevens (2008) the objective of doing a storyboard is to identify the parameters of the eCIS while using only the available resources and time. It also allows the developer to organise and focus the eCIS site while defining the medium to be used for the eCIS site.

The client side of the eCIS is usually a combination of different mediums such as video, text, still photos, audio and graphics. The way that these pages interact is presented in a nonlinear format in the storyboard (Stevens, 2008). The process to follow in the development of the storyboard includes dividing the story into its logical, nonlinear parts and then dividing the content of the story among the media that will be used – video, photos, audio, to be included with each photo and interactive animation (Stevens, 2008).

Next a map is developed for the eCIS, either a location map or a layered map. The associated text that is required for the eCIS is also identified and built into the storyboard. It is important to ensure that the different media types complement each other, remembering that the story is linear and the user will be providing the input into the order and way the different pages will be controlled. The storyboard is a rough sketch of the final eCIS and not too much effort should go into making it perfect (Stevens, 2008). A storyboard helps to identify missed aspects of the eCIS being developed and provide some order in the non-linear product that is being developed (Stevens, 2008).

5.9 Comparison of RUP, AM(XP), OSS and SBUP

RUP	AM(XP)	OSS	SBUP
High Level Quality Aspect: e-Process Aspects			
Second Level Quality Aspects: Completeness			
Detailed development phases have to be laid down. An iterative lifecycle is followed and risk-driven development is undertaken. Can easily be used for eCIS Development.	AM is not a complete software process, but focuses on effective modelling of requirements and documentation and should be combined with a process such as XP. Can be applied to eCIS development.	No explicit system-level design or detailed design. No project plan, schedule or a list of the deliverables. Has been used for eCIS development.	This process follows the rules for user profiling and modelling to build a user model and the rules for storyboarding in order to develop the layout and order of the eCIS. In an eCIS development scenario where this is used it is probable that the database will already exist.
Second Level Quality Aspect: Understandability			
Detailed user manuals available but very detailed and extensive.	Easy to understand and learn.	Can be difficult to identify aspects required for development.	Easy to use and apply to eCIS environment.
Second Level Quality Aspect: Visibility			
Results in clear outcomes. Effective and efficient project management aspect is available.	Outcome not necessarily clear and project management not built into this process.	Application of this process will determine the clearness of the outcome.	Can result in clear outcomes. No project management available as part of the process.
Second Level Quality Aspect: Supportability			
Extensive support.	Support from other developers.	Support from other OSS developers over internet.	Not much support specifically specified.

Table 5.1 Comparison of e-Processes

RUP	AM(XP)	OSS	SBUP
Second Level Quality Aspect: Maintainability			
Change requests easily managed.	Change requests can be accommodated.	Change request not always easy to accommodate.	Change requests not always easy to accommodate.
High Level Quality Aspect: Quality Aspects			
Second Level Quality Aspects: Completeness and reliability.			
Notation is easy to use once the developer knows the process. Errors easy to avoid or follow up. Extensive support when unexpected occurs.	Relatively easy to use the notation available. Smaller implementations mean better control of errors.	Easy to use once OSS process has been clearly defined. Peer support for error fixing.	Very easy to use notation. Error not always easy to monitor. Not much support for unexpected.
Second Level Quality Aspects: Robustness (Testing).			
The whole development environment is based on the use of UML (Unified Modelling Language) as a basis.	Tests developed before coding. Test everything. Test before and after you refactor as well as when a new task is implemented. The following types of testing should all be included in the deployment plan: acceptance testing, performance and quality testing.	This development consists of continuous testing.	No formal testing procedures are recommended or included.

Table 5.1 Comparison of e-Processes (continue)

RUP	AM(XP)	OSS	SBUP
High Level Quality Aspect: Cost			
Second Level Quality Aspect: Development Budget and Running costs			
Expensive for both development work and running the resulting system. Once off initial costs and upgrades.	Can be developed reasonably inexpensive.	Inexpensive for both development and running the resulting system.	Can be developed reasonably inexpensive.
High Level Quality Aspect: Domain Impact			
Second Level Quality Aspect: Team experience and Domain Knowledge			
Large development teams are created for the development of large systems. All have access to the repository.	Develop software in pairs. The development team should be prepared to take on some of the following roles: Coach; Tracker; Facilitator and Architect. The development team is responsible for estimations, consequences of decisions, software development process and the releases of the software.	Developers are potentially a large number of volunteers. The developers using OSS can be assured of the following rights: make copies of existing programs and distribute the copies; access source code and; make improvements to programs.	All types of eCIS can be accommodated. The development approach will most probably be incremental. The development team has a joint responsibility for development in Joint Application Sessions.

Table 5.1 Comparison of e-Processes (continue)

RUP	AM(XP)	OSS	SBUP
Second Level Quality Aspect: Enterprise Culture and Business Strategy			
End-users see the developed parts quickly and the users are engaged to provide feedback for adaptation, so that the end product can meet the needs of the stakeholders.	Customer decides scope, priority and release content. Customers form an integral part of the development process. User roles are storytellers, acceptors, resource providers, planners and the person that champions the system.	Users play a large role as testers and documenters as well as defining new requirements promptly.	The user is involved in both the user profiling exercise as well as giving input into the development of the storyboard.
Second Level Quality Aspect: IT Strategy (Type of System)			
All types of eCIS systems.	Large eCIS systems will have to be subdivided into smaller systems to be developed by AM(XP).	Infra-structural, multi-user and all types of eCIS.	All types of eCIS can be accommodated. The development approach will most probably be incremental.
Second Level Quality Aspect: Infrastructure and Technology			
Usually used for larger projects. There are many developers in the development team and all have access to a central repository.	Usually used for larger projects. There are many developers in the development team and all have access to a central repository.	Usually used for larger projects. There are many developers in the development team and all have access to a central repository.	Usually used for larger projects. There are many developers in the development team and all have access to a central repository.

Table 5.1 Comparison of e-Processes (continue)

RUP	AM(XP)	OSS	SBUP
High Level Quality Aspect: Usability			
Second Level Quality Aspect: Functionality			
The Rational Unified process has the following phases: inception, elaboration, construction and transition.	<p>Conceptualize the system: Create a vision of the system, write user stories, acceptance tests, find and check solution.</p> <p>Plan new system: Estimations, plan releases, iteration schedule, tactical planning.</p> <p>Develop the system. (Pair programming, continuous testing, use Agile design values, principles, practices, develop code and refactor; integrate daily)</p> <p>Deliver the system.</p>	Define roles and responsibilities; identify work to be done; assign and perform the development work; pre-release testing; inspections; managing releases.	The rules and standards of development as defined for user profiling and storyboarding are followed.
Second Level Quality Aspect: Manageability			
Delivery of small releases, but transition phase consists of beta tests and then full deployment.	The delivery of small releases is made to the customer.	New parts of the project are released frequently.	The user is involved with the whole process and development will take part in increments. Brainstorming.

Table 5.1 Comparison of e-Processes (continue)

RUP	AM(XP)	OSS	SBUP
Second Level Quality Aspect: Quality Assurance			
Develop the software iteratively, manage the requirements, use component-based architecture, visually model the software, verify software quality and control changes to software.	Work with the customers; Use metaphors to describe difficult processes; plan; keep all meetings short; test first; keep it simple; develop programs in pairs; code to standards; everybody owns the software; integrate the newly developed code continuously; improve the internal structure of the code; release early and often; don't work too long hours; be adaptive and open to change.	Freedom to innovate and modify as required. The development process provides the ability to distribute modifications to software to others free. Know what to create and what to re-use. There can be rapid responses to user requests.	Determine the type of user as well as a description of the scenario of the eCIS to be developed, a description of the character of the eCIS and the customer, customer expectations, reasons for using eCIS, customer interest and needs, description of goals to be achieved. The objective of storyboarding is to identify the parameters of the eCIS while using only the available resources and time. It allows the developer to organise and focus the eCIS site while defining the medium to be used for the eCIS site with the help of the user.
High Level Quality Aspect: Compatibility			
Second Level Quality Aspect: Exchangeability and Map-ability			
Artifacts have a high level of exchangeability.	Can map parts of projects onto other e-Processes quite easily.	Can map and exchange artifacts easily.	Less designed for easy exchange and mapping than other processes.

Table 5.1 Comparison of e-Processes (continue)

RUP	AM(XP)	OSS	SBUP
High Level Quality Aspect: Maturity			
Second Level Quality Aspect: Stability and tool support			
Business Modelling: Domain Model Partial artifacts in each iteration Requirements: Specify requirements and constraints (terms, attributes and validation) Use-case models Vision Statement Supplementary Specifications Glossary Design: Design model Software architecture. Project Management: Software development plan. Test: Test plan (with acceptance tests from requirements) Environment: Development case.	AM(XP) uses models such as case models, class models, data models and user interface models. Develop different models in parallel.	Process model needs to be developed.	All types of eCIS can be accommodated. The development approach will most probably be incremental. A customer user profile and model is built with user profiling and with storyboarding. The storyboard is built with the help of the user to reflect the whole eCIS.

Table 5.1 Comparison of e-Processes (continue)

RUP	AM(XP)	OSS	SBUP
Second Level Quality Aspect: Documentation (UML modelling)			
The whole development environment is based on the use of UML (Unified Modelling Language) as a basis.	UML provides a number of diagrams. XP has found 3 of these diagrams useful to help find solutions to problems, namely the class diagram; sequence diagram and state diagram.	UML not used as a standard.	The development process does not use UML diagrams as a prescribed part of the process.
Second Level Quality Aspect: Tool Support (Case Tools)			
Full software support for the whole development process.	Agile enhances other software processes, but agile does not have its own Case Tools.	Browsers, editors, query servers, authoring servers and distribution servers are required. A developers' Email list is also available. 3GLs. The OSS Application area is more complex.	Software is available for fully automated user profiling as well as software specific to make the storyboarding process easier. However, it is recommended that storyboarding be done on, for example, a white board with the application team brainstorming.
A few possible reference site/s.			
www.therationaledge.com www.rational.com	www.agilealliance.org www.agilemodeling.com www.extrememodelling.org	www.opensource.org	http://multimedia.journalism.berkeley.edu/tutorials/reporting/starttofinish/storyboarding http://www.webdevelopers-journal.com/articles/user_profiling_diy.html

Table 5.1 Comparison of e-Processes (continue)

The above comparison information was sourced from the following references:
(AM, 2001; Astels, et al., 2002; Bloomberg, 2000; Feller & Fitzgerald, 2000; Hogarth, n.d.;
Kruchten, 2001; Larman, 2002; Mockus, et al., 2002; Perens, 1999; Reed, 2002; RUP, 2002)

5.10 Summary

This chapter introduced four different e-Processes that will be used in the rest of the thesis to demonstrate e-Process Selection. These four e-Processes are not the only e-Processes that can be used, but are used in the thesis as examples.

In summary, the Rational Unified Process was chosen because it is widely used in industry as a formal development process; Agile software development represents a number of development processes and discussions with developers indicated that they would like to use this development process; Open Source Software Development was chosen to include a development process which is free and re-distributable and Storyboarding and User Profiling was included in the list of 4 possible e-Processes because it is more eCIS specific and seems to be widely used as an individual or as a combined development process.

PART B

DECISION MAKING METHODOLOGY

Chapter 6

DECISION MAKING

This chapter introduces decision making and focuses on multi-criteria decision making. Multi-criteria decision making uses multiple criteria in order to identify and select one solution to a problem from a selection of possible solutions. This chapter introduces and briefly discusses decision theory and how it is used in a decision making environment. The chapter then continues with those aspects that are required for defining the decision making approach in the following chapters.

6.1 Introduction

It is important in today's business environment to make decisions that are of a high standard, and this can be a difficult and complex process. There are different components required for any specific solution to a problem. These components include resources such as technology, material, development, and human resources which tend to be expensive. The decision maker is responsible for deciding which solution will be selected. Today's market is very competitive, which means it is important to make fast decisions, which are based on well-motivated reasons and provide the right outcome (Pomerol & Barba-Romero, 2000, p. 2; Trueman, 1976). The decision maker will most probably require some support in this decision making process.

It is difficult to specify the components that will define how to make decisions that result in the best possible outcome, but it is important to take a number of different aspects into consideration when making decisions.

6.2 Conceptualisation of Decision Making

Decision analysis is when the decision maker or group of decision makers decide between different choices of action in an environment which is uncertain. According to Doumpos and Zopounidis (2002, p. 3) and Triantaphyllou (2000) there are four different kinds of analysis that are required to be performed when considering a discrete decision making problem. This analysis includes:

- Identifying the best alternative/s from a set of alternatives,
- constructing a ranking of the alternatives from best to worse option,
- classifying or sorting the alternatives into predefined homogenous groups, or
- identifying the major features of the alternatives and then defining their description based on these features.

Analysis of the alternatives provides support to decision makers in their decision making roles. The first three types of decision making processes (choice, ranking, classification) are based on the evaluation of the alternatives in order to make a selection. Both choice and ranking are based on relative judgements, where the alternatives are compared – usually pair-wise to achieve the result (Doumpos & Zopounidis, 2002, p. 3). The classification problem however, is based on absolute judgements, where each alternative is assigned to a specific group on the basis of some predefined rule, which does not depend on the alternatives (Doumpos & Zopounidis, 2002, p. 5).

For the purposes of this chapter, Figure 6.1 is being used as a basis for discussing the important aspects that are required in the thesis.

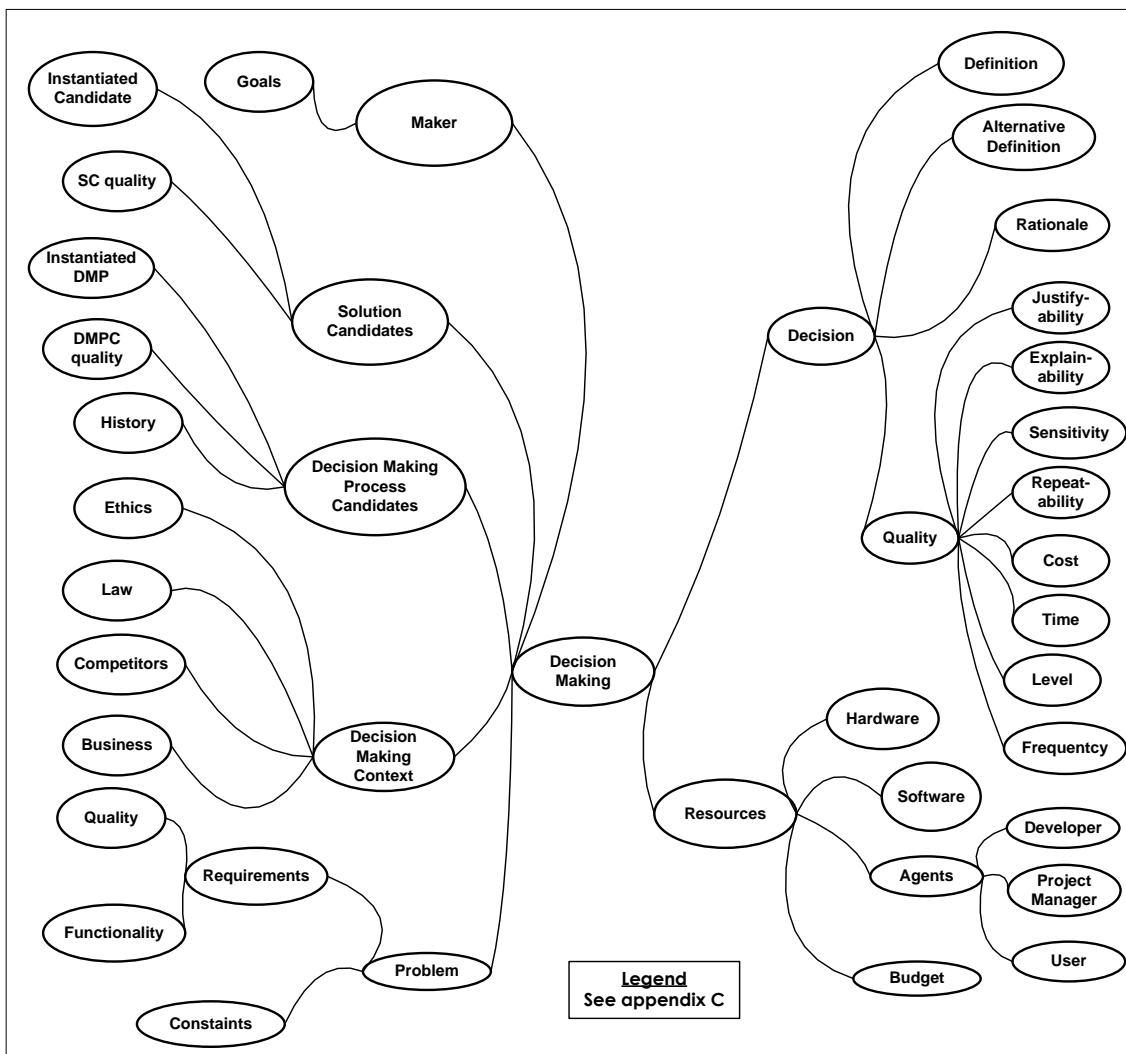


Figure 6.1: Aspects of decision making included in this chapter

6.3 Quantitative and qualitative decision rules

A decision can be seen in the following framework: If S is a set of states and X is a set of potential consequences of decisions made then the act can be seen as the mapping f from S to X , where for each $s \in S$ the act f produces a result $f(s) \in X$. Sometimes the different acts are ranked in order to support the decision maker in the decision making process. This is often done using a specific, often numerical ranking method (Riccia, Dubois, Kruse and Lenz, 2003, p. 5).

There are a number of factors that influence decision making and situations calling for these decisions to be made are never the same and differ from case to case. The factors impacting on the decision include (which include some as stated by French & Xu (2005)):

- The context in which the problem is being defined including the quality aspects of the problem, e.g. how well-defined it is;
- What we know about the problem domain;
- The number of solutions we need to choose from.
- The people involved with the decision making process including their background and knowledge about the problem domain;
- How many people need to be accommodated;
- Who is responsible for which part of the problem solution;
- The people who are accountable for the decision being made;
- What type of decision needs to be made;
- The knowledge and quality aspects of the decision makers, which include how intelligent and knowledgeable they are;
- Their risk profile and whether they can accept a measure of uncertainty;
- Their pre-knowledge and experience in the problem domain;
- Available resources and;
- Applying the defined constraints.

In the decision making process there are a number of factors that have to be considered. These factors help to determine the quality of the candidate solutions. They include:

- The people involved with the decision making process are the decision makers, experts and different problem domain stakeholders. Not all of these people will necessarily be involved from day one, so it is necessary to determine when and

on which level they will be involved. It is also necessary to define the roles that each of these people are going to play in the decision making process.

- The methodology that the people listed prefer or believe in. If a specific method has always been preferred in the decision making process that is most probably going to be the preferred method in this process.
- How long this process can take – does the decision need to be made in one meeting or do the decision makers have months to consider their choice.
- The level of detail required in documenting the decision process can play a role in the decision making process adopted. The level of involvement or impartiality of the decision makers needs to be factored into the decision making process (French & Xu, 2005).
- The resources required to run the process as well as available resources that will impact on the final decision taken.

Other factors to be considered when deciding on how to approach the decision making process include aspects such as: The structure of the problem (solution candidates), which in this research includes the quality aspects, to be included in the decision making process applied; how different weights and values used in the decision making process will be gathered and applied in the problem environment; and how the data will be presented and whether sensitivity analysis will be applied (French & Xu, 2005).

All of these aspects listed above will be further investigated in the application to the problem domain in the rest of this document.

6.4 Purpose of Decision Making

Decision making is the process of finding an option or alternative out of a set of eligible solutions or alternatives to a problem, which will solve the problem. Business requires that there be some motivation or documented proof of why specific solutions were selected from a set of alternative solutions. It is therefore important that for this thesis it is stated that the rationale behind decision making is found in the process of making defensible decisions which can be justified.

There are a number of concepts that are important to decision making. These include: ambiguity, decision, action to be taken after a decision has been made, available choices, the actual problem to be solved, the knowledge of the decision makers, the preferences of the people that will have to operated in the decision environment, and

consequences of the decision made – who/what will it affect and does one need to justify why a specific decision was made – orally and documented (White, 2006, p.1).

There are two reasons why a specific decision making process should be followed:

- Firstly, it is important to ensure that the process is documented. Documentation helps to support the decisions made, should management require reasons for specific decisions made. Management is concerned with strategic decisions and expenditures made. Structured decision making processes will help with the justification of why specific decisions were made.
- The second reason is that the decision maker, without a strong background in eCIS development, e-Processes and/or decision making methods, requires some support in making these decisions. Following a structured process will provide these people with the required support.

6.5 Types of decisions and software options

As seen in Figure 6.2 there are three main types of decisions that can be taken.

These decisions are:

- **Strategic decisions:** Major decisions that can impact on how the business is run. Cost plays a major role in strategic decisions. These decisions are usually long term, at an organisational level and are required to achieve major business goals. In this thesis the decision making model to be used for the selection process can be seen as a strategic decision. The inclusion or exclusion of different experts in determining the values to be used can also be seen as a strategic decision. Even the selection of a specific e-Process can in some instances be seen as a strategic process.
- **Tactical Decisions:** These are decisions made on the direction that the business should be taking. This is the grey area between strategic and operational business decisions and focuses on medium level goals. The level of documentation required for documenting the process of decision making can be seen as a tactical decision in this research.
- **Operational Decisions:** These are day to day business decisions. These decisions are short term and on a task level. Minor goals need to be achieved. Operational decisions include the business model used in the process; selling or

contracting the decision making to user and; which base machines to use to run the software.

(Taylor, 2007)

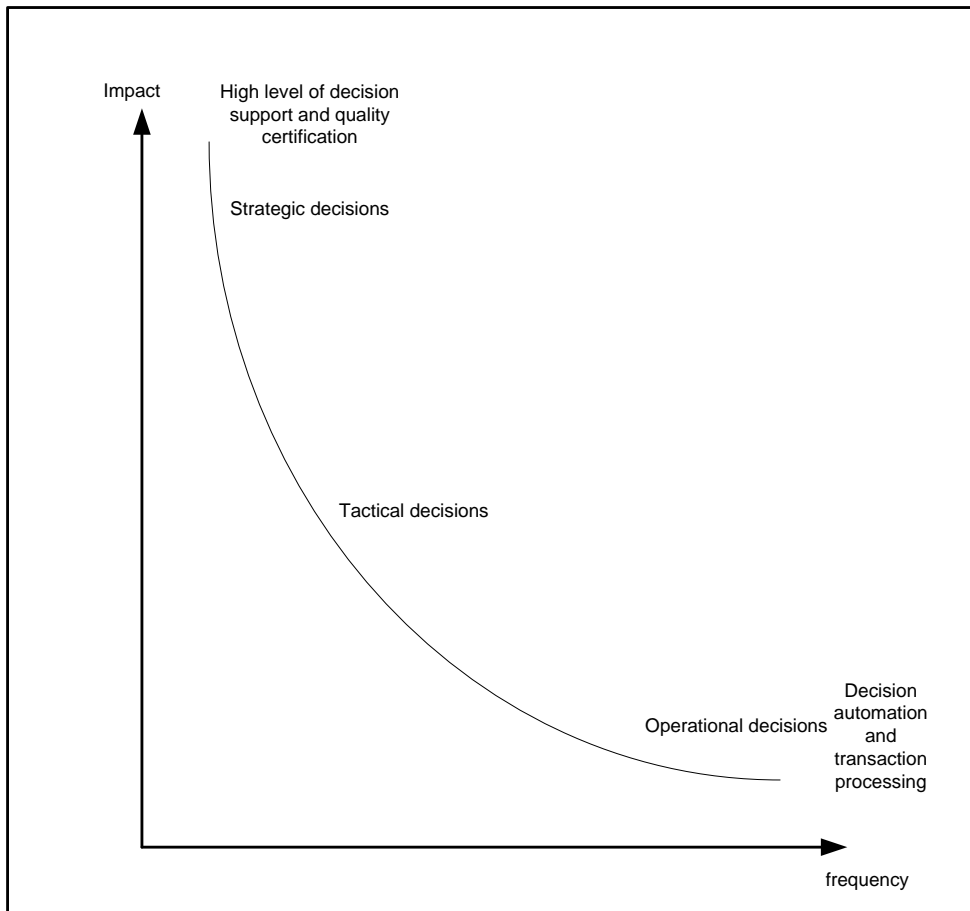


Figure 6.2: The Relative Value of Decision Support and Decision Automation for Different Types of Decisions (based on Taylor, 2007)

Moving up or down in the decision making process is required depending on the level of the problem complexity. The e-Process Selection presented in this thesis could be classified as a tactical decision; it is therefore important that software support for decision makers as researched and defined, and some decision automation are developed to support their decision processes. This is especially relevant for small-to-medium sized enterprises in New Zealand where there are not much support available for decision making. The focus in this thesis is mainly on these types of businesses.

There are a number of software packages available that support decision analysis. French and Xu (2005) identified that a survey in 2004 listed 45 different packages for decision analysis. All the packages were described as versatile, user-friendly and with

the capability to accommodate fairly substantial problems (French & Xu, 2005). As each of these packages have their own view of the world and the problem to be solved, the author of the thesis focuses on the development of a decision making prototype that select an e-Process for a specific eCIS using different decision making methods.

There are a number of software packages available that support the different schools of thought. These include multi-value decision analysis (MVDA) (Keeney, 1992) and analytic hierarchy process (AHP) (Saaty, 1990a), which will be covered in this thesis.

6.6 Decision analysis and Decision Making Methods

Decision analysis is when the decision maker or group of decision makers decide between different choices of action in an environment which is uncertain. A set of alternatives is usually available and one of these needs to be decided upon.

The theory of decision analysis has been developed to help with the process of making a choice between a set of pre-agreed upon alternatives. In order to make the decision the information available about the different alternatives is used to determine the choice.

The quality of the available information can be anything from subjective interpretations to scientifically proven data, from being sure about the outcome of the decision (deterministic information) to being uncertain about the outcome and using probabilities and fuzzy logic. Different methods and techniques are therefore required to assist in processing the wide variety of available information (both type and quality of information) to support the people making the choice (ISDS, 2004).

In order to understand the decision making process, it is necessary to understand the underlying concepts. Some of the definitions of some of the general terms are as follows:

- Decision making is the process of deciding on what is important by making choices between different alternatives and then reaching a specific conclusion (Bouyssou, Marchant, Pirlot, Tsoukas & Vincke, 2006; MSNEncarta, 2007).
- Decision theory is the study of outcomes for decisions: the study of the best possible outcomes for decisions made under varying conditions (Bouyssou, et al., 2006).
- A decision is the action of deciding (a contest, controversy, question, etc.), settlement or determination (Oxford_English_Dictionary, 2007).
- A decision is also the process of making up of one's mind on any point or on a course of action; a resolution, determination (Oxford_English_Dictionary, 2007).

- A decision is the commitment to irrevocably allocate valuable resources as well as a choice or judgement and with a firmness of conviction where a result is arrived at by the judges (Bird, 2003; Borda count, 2007).

Decision making can be described as the cognitive process leading to the selection of a course of action among a set of possible candidate solutions. Every decision making process produces a final choice. It can be an action or an opinion. It begins when we need to do something but do not know what. Therefore, decision making is a reasoning process which can be rational or irrational; it can be based on explicit assumptions or tacit assumptions (Saaty, 1990b).

Making decisions can be seen as identifying possible alternative solutions to a problem and then using a decision making process to determine which of these alternative solutions will provide a possible solution to solving the problem at hand. Making decisions is not a rational process with specific answers as to why specific choices were made. It is however, a defensible and defined process to follow in order to support the decision making process and be able to justify the choices made (Gunther, 2008).

The science of making decisions is complex and covers a wide range of topics. It involves theoretical as well as practical aspects. People are confronted with different problems on a daily basis – in their own lives as well as in the corporate environment. Choices need to be made between different alternatives based on a problem that need to be solved (Doumpos & Zopounidis, 2002, p. 40; Pomerol & Barba-Romero, 2000, p. 8).

If the decision making method used is organised and the decision making is done systematically then there is an excellent chance of making good decisions. This will reduce the risk of making bad decisions having used insufficient data and insufficient analysis of the results. There is a wide range of variables that can influence the decision. If a well-defined decision making process is followed which identifies good alternatives and evaluates these well, then a quality result can be obtained (Saaty, 1990b).

According to Saaty (1990b), the following steps need to be taken when making decisions:

- Structure the problem with a model that identifies the main elements and the relationships between these elements.
- Elicit judgements from experts that will reflect their knowledge, skills, feeling or emotions.

- Represent or measure these judgements in terms of meaningful numbers.
- Use these numbers to determine priorities. (If using a hierarchical structure then use these number to set priorities within the hierarchy).
- Calculate the result using a specific decision making method.
- If required do a sensitivity analysis in order to make necessary changes in judgement.

As published in *Pensees* (1660), Blaise Pascal discussed wagers, probabilities and decision making. Since this early publication a number of decision making methods have been researched and described. In 1939 Abraham Wald introduced much of the grounding of modern decision theory. This included statistical hypothesis testing, estimation theory, loss functions, risk functions, admissible decision rules, Bayes decision rules and minimax decision rules (Hanson, 1994).

Bayesian interpretation probability is an extension of logic which allows for reasoning about uncertain statements. Bayesian probability sees the concept of probability as a measure of a specific state of knowledge (Bernardo, 2006). The rules of Bayesian statistics can be justified by rationality and consistency or it can also be a personal belief (Bernardo, 2006).

Frank Ramsey (1931) and others introduced subjective probability, which is an extension on expected utility theory. This theory allowed for actual human decision making behaviour under risk (Ramsey, 1931). Much of the research was undertaken in the fields of economics and behavioural humanities. Some of the later research was done in the field of game theory (Kelly, 2003) and socio-cognitive engineering (Gadomski, 2003).

In this research the focus is on the formal decision making methods. These include cost-benefit analysis, multi-criteria decision analysis, decision trees and many more. It is difficult to decide from the wide range of decision making methods, which is the most appropriate, as the choice depends on the problem to be solved or may sometimes to some extent depend on the decision making method that the decision maker is the most comfortable with (Bouyssou, et al. 2006). In summary there are many decision making methods available, which means that the decider needs to select one of these methods.

As Multi-Criteria Decision Analysis (MCDA) or Multi-Criteria Decision Making (MCDM) is one of the most well known branches of decision making (Triantaphyllou, 2000), for the purposes of this thesis the author has focused on multi-criteria decision making methods.

6.7 Multi-criteria decision making

The wide variety of different types of decision problems and the quality of the information available at the time of making decisions identifies the need for methods and techniques that can assist the processing of the available information as well as ultimately lead to better decision making (Unit-For-Sustainable-Development-and-Environment-Organization-of-American-States, 2007).

The quality of the decision made depends on the quality of the information available on which the decision is based. Different methods and techniques are required to assist and cope with the variety and diversity of the types of projects and the quality of the information available about the problem domain. Ultimately, using a structured multi-criteria decision making process will help to support making better decisions.

Powerful mathematical programming tools have emerged using multiple objectives to assist in the process of searching for decisions which best satisfy a multitude of conflicting objectives, and there are a number of distinct methodologies for multi-criteria decision making problems available. These methodologies can be classified in a number of ways, such as form of model (e.g. linear, non-linear, stochastic), quality aspects of the decision space (e.g. finite or infinite), or solution process (e.g. prior specification of preferences or interactive) (Pomerol & Barba-Romero, 2000; Unit-For-Sustainable-Development-and-Environment-Organization-of-American-States, 2007).

In 1896, Pareto first defined a basis for solving decision problems using multiple criteria (Doumpos & Zopounidis, 2002, p. 40; Pareto, 1896). Multi-criteria analysis was defined and formulated more thoroughly in 1960 as: Multi-criteria analysis is the study of solving a problem by choosing one of the alternative solutions to a problem based on a set of multiple criteria associated with each solution.

At a mathematical conference in The Hague, Netherlands in 1970 a section was devoted to multi-criteria analysis. A number of publications resulted in the next four years on this topic (Pomerol & Barba-Romero, 2000, p. 3). Since then multi-criteria decision making has been the topic of much research. Multi-criteria decision making can be seen as the field of activity in which practical applications as well as information technology are dominant. The focus in this process is in providing practical ideas and methods with depth that can solve the problems associated with decision making (Pomerol & Barba-Romero, 2000, p. 6; Bouyssou, et al. 2006; Triantaphyllou, 2000).

In order to understand the different solutions available to solve a problem it is necessary to briefly explore problems. It is difficult to classify decision making problems. One solution is to define only two categories of decision making problems. These are:

Discrete problems: Discrete problems are problems that contain and involve the evaluation of a discrete set of alternatives. Each alternative problem solution is classified in terms of a set of aspects. These aspects form the basis for deciding between the different choices available.

Continuous problems: Continuous problems are problems that contain and involve the evaluation of an infinite set of possible alternative solutions. When there is an infinite set of solutions the answer to problem solution will lie in a region rather than with a specific solution (Doumpos & Zopounidis, 2002, p. 1; Triantaphyllou, 2000).

All of the decision making methods that will be used in the research in this thesis are discrete.

Multi-criteria decision making uses multiple criteria in order to identify a solution to a problem choosing one solution from a selection of possible solutions. "Multi-Criteria Decision Making (MCDM) is the study of methods and procedures by which concerns about multiple conflicting criteria can be formally incorporated into the management planning process" from the International Society of Multiple Criteria Decision Making (Steuer, 2008, unpaginated).

There are different names given to this study area including Multi-Criteria Decision Analysis (MCDA), Multi-Dimensions Decision-Making (MDDM) and Multi-Attributes Decision Making (MADM). According to Triantaphyllou (2000) and other authors MCDA is divided into multi-objective decision making (MODM) and multi-attribute decision making (MADM) but all of these refer to mean the same class of decision making methods (MCDA).

Multi-Criteria Decision Analysis (MCDA) or Multi-Criteria Decision Making (MCDM), is defined as being a discipline aimed at supporting decision makers who are faced with making numerous and conflicting evaluations between numerous different solutions to a problem. MCDA aims at highlighting these conflicts and deriving a way to come to a compromise in a transparent process (Doumpos & Zopounidis, 2002, p. 39; Multi-criteria decision analysis, 2008; Triantaphyllou, 2000). MCDA was developed because it was found that resolving complex real-life decision problems cannot be performed using uni-dimensional approaches (Doumpos & Zopounidis, 2002, p. 40).

In MCDA the existence of a decision maker is assumed (Pomerol & Barba-Romero, 2000, p. 17). The individual (or group of) decision makers will investigate different choices of action in an uncertain environment in order to determine the best course of action. The theory of decision analysis can be seen as the support provided to the decision maker when making a choice between different pre-defined or pre-specified alternatives. The decision making process is reliant on a set of quality aspects that provides information about the alternative options available (Unit-For-Sustainable-Development-and-Environment-Organization-of-American-States, 2007).

The decision making process can be made based on deterministic information (certainty about decision outcomes) or the decision making process can be based on information that has uncertain outcomes and may be represented by fuzzy logic and probabilities. The quality of this information can enable the decision maker to make clear scientific determined decisions or decisions based on different interpretations (Unit-For-Sustainable-Development-and-Environment-Org.-of-American-States, 2007).

There are many MCDA methods available in the literature, each with its own characteristics. One methods of classifying MCDA methods is according to the type of data they use. Some of these may be deterministic; some may be stochastic or fuzzy. Often these data type can however be used in combination (Triantaphyllou, 2000).

Another method of classification can be according to whether a single decision maker is involved or whether a group of decision makers may be involved (Triantaphyllou, 2000).

A taxonomy of MCDA methods – adopted from Chen and Hwang (1991) - can be seen in figure 6.3, on the next page. Some of the more widely used methods are the Weighted Scoring Method (or Value-Benefit Analysis – see chapter 7), the analytic hierarchy process (AHP), the weighted product model (WPM), the ELECTRE and the TOPSIS method (Triantaphyllou, 2000).

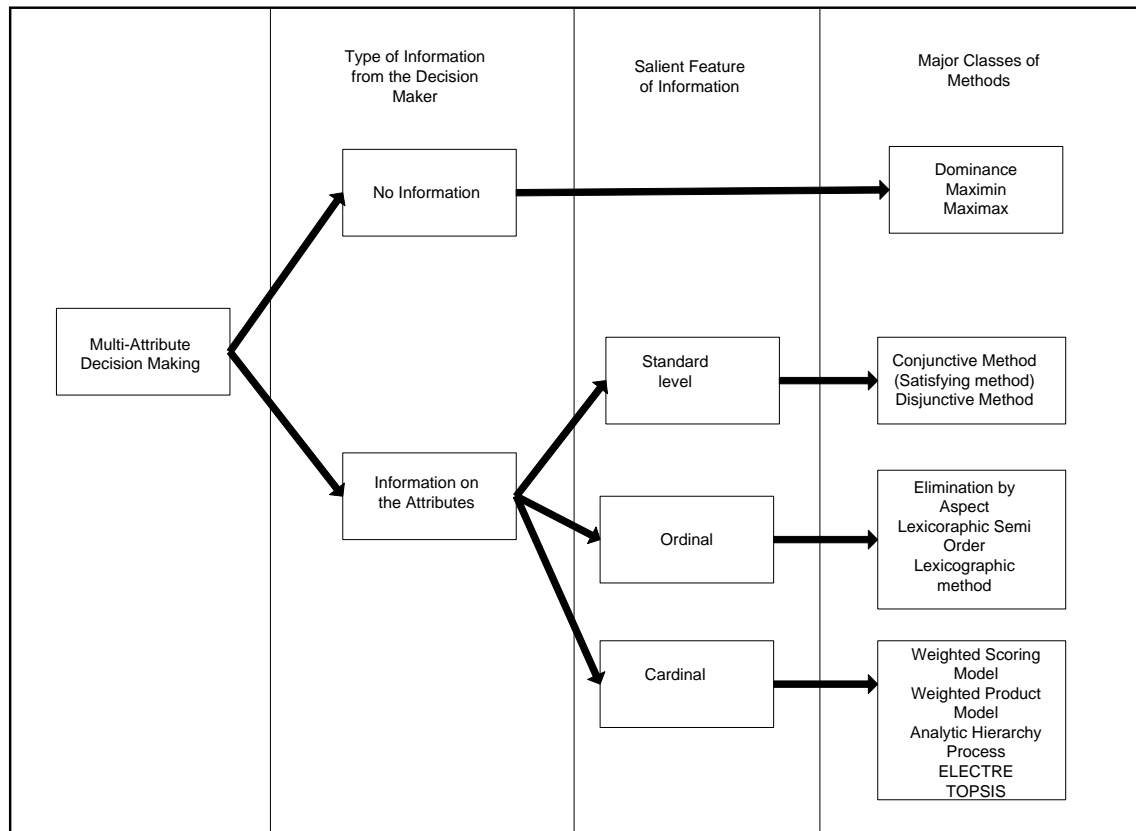


Figure 6.3 A Taxonomy of MCDA Methods (Chen and Hwang, 1991)

6.8 Decision making aspects for this research

The focus of this research is, for a given problem (eCIS) to be developed, and with a given set of possible candidate e-Processes, to select the e-Process that is most likely going to provide a high quality eCIS and allow documented support for selecting that specific selected e-Process. The eCIS to be developed will have requirements which include quality requirements and functional requirements that need to be incorporated into the final product.

The development of the eCIS will also have defined constraints that will determine aspects such as time and money available for development of the eCIS. Resources available for development will include the hardware, software and agents (people). The agents include the developer, the project manager, the sponsor and eventual users of the eCIS. Firstly the agents will most probably be the people who will need to either be responsible for making the decisions or be responsible for justifying why specific decisions were made.

All of the aspects mentioned will play a role in the way that possible candidate e-Processes will be selected. Many of these will be used as quality aspects in evaluating the different e-Processes for selection.

In this thesis a decision needs to be made on:

- Firstly, which of a number of decision making processes to use, in order to make a decision (selection) between the different candidate e-Processes available?
- And then secondly, most importantly, which of a number of e-Processes to select in order to develop a specific eCIS.

As stated before, MCDA is one of the most well-known branches of decision making (Triantaphyllou, 2000), the author therefore decided to choose different decision making methods from this group. The research in this thesis focuses on multi-criteria decision models in order to select a suitable e-Process. To place this decision-making process into context, a dependable, well-defined decision making model is followed when selecting an e-Process for eCIS development. The rigor and quality of the decision making process will help to define and strengthen the selection made. Businesses compete in the market place for a market share. An eCIS which is of a high quality and dependable will make the business as competitor stronger as the eCIS will hopefully help to increase their sales, visibility, marketability and client base.

There are a number of non-functional requirements that will need to be taken into consideration when selecting the e-Process. These will include issues such as: **privacy** for the people who will be using the eCIS; **security** of the information being captured and stored; and **cultural background** of the users, business and developers. All of these will need to be taken into consideration in the decision making process. eCIS is international and as such the fact that it is accessible from different countries will play a role in the development of the eCIS. The **laws of the different countries** will need to be considered and taken into account in the whole process (Sommerville, 2000).

There are different dimensions in the decision making process and also a number of different meta-models that can be considered in the decision making process. In this thesis there are two decisions to be made. The first one is that there are multiple MCDA proven methods available that can be used in the process of selecting a suitable e-Process. In this thesis four of these MCDAs have been researched as being suitable in this selection process. Each of these decision making processes candidates has its own quality and strengths and weaknesses. A guideline on how to decide between the four

MCDAs in this thesis, in order to obtain the instantiated decision making method, is further explored in chapters 7 and 9.

The second decision that needs to be taken by the decision makers is which of a set of e-Processes (solution candidates) will be suitable to develop their specific eCIS. Each of the solution candidates will have its own quality aspects to distinguish it from other candidates. One of these solution candidates is instantiated and recommended as a suitable e-Process.

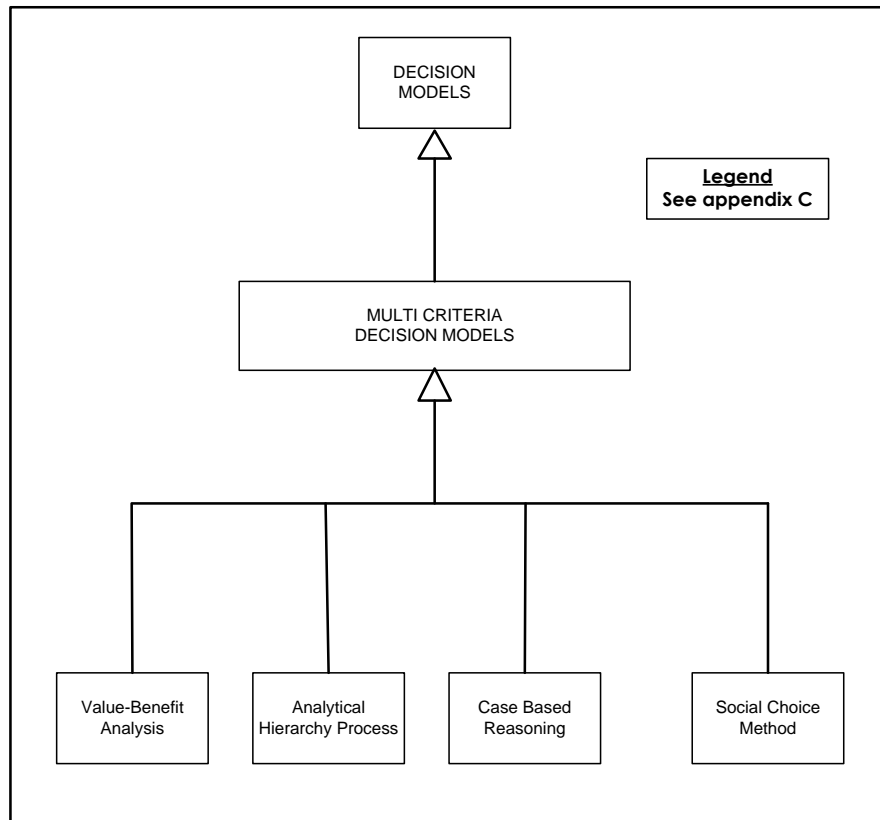


Figure 6.4: A semantic view of the decision meta-model

The research presented in this thesis is all Multi-Criteria Decision Analysis. Figure 6.4 models basic views which are presented of the decision making process. In this UML semantic model each class is considered an element in UML specialisation/generalisation relationship between for example the class of all decisions and the class of multi-criteria decisions.

The multi-criteria decision class has a number of subclasses that are important in this thesis, namely the Value-Benefit Analysis Decision Making Class, the Analytic Hierarchy Process Decision Making Class, the Case-Based Reasoning Decision Making Class and the Social Choice Decision Making Class. The first three can all be associated with a

number of possible e-Processes and each of these is also associated with a number of quality aspects that will be used in the evaluation process. These components are explored in the next chapter when the meta-model is developed. The Case-Based Class is associated with a number of historical cases that will be contained in a knowledge base. The meta-models for all of these Multi-criteria decision models will be developed in the next chapter.

6.9 Conclusions

Decision making is an important research field. The development of eCIS is also important and necessary and requires an e-Process. Decision makers need to make good and defensible decisions about which e-Process to select when a new eCIS is being developed. The decisions made have an impact on how well the business will perform in the future. Different resources are required in the development process with the implication that capital is required for the development process. It is easier to defend decisions if they are well documented and based on thorough study of the available options.

There are a number of multi-criteria decision making processes available. The decision making in this thesis consists of a first level and a second level process. The first level process is when the decision making method is selected and the second level when the development process is selected, with the end result being the instantiation of each.

The next chapter will describe the decision making approach followed and will include the integrated meta-model used in this research and chapter 8 focuses on the decision making methodology that will support decision makers in the process of deciding which decision method listed to use.

Chapter 7

Decision Making Approach – the Meta-Model

This chapter introduces the different decision making methods, their application and then the requirements of a meta-model for each that will support the e-Process Selection Process. The chapter concludes with an integrated meta-model for e-Process selection.

7.1 Introduction

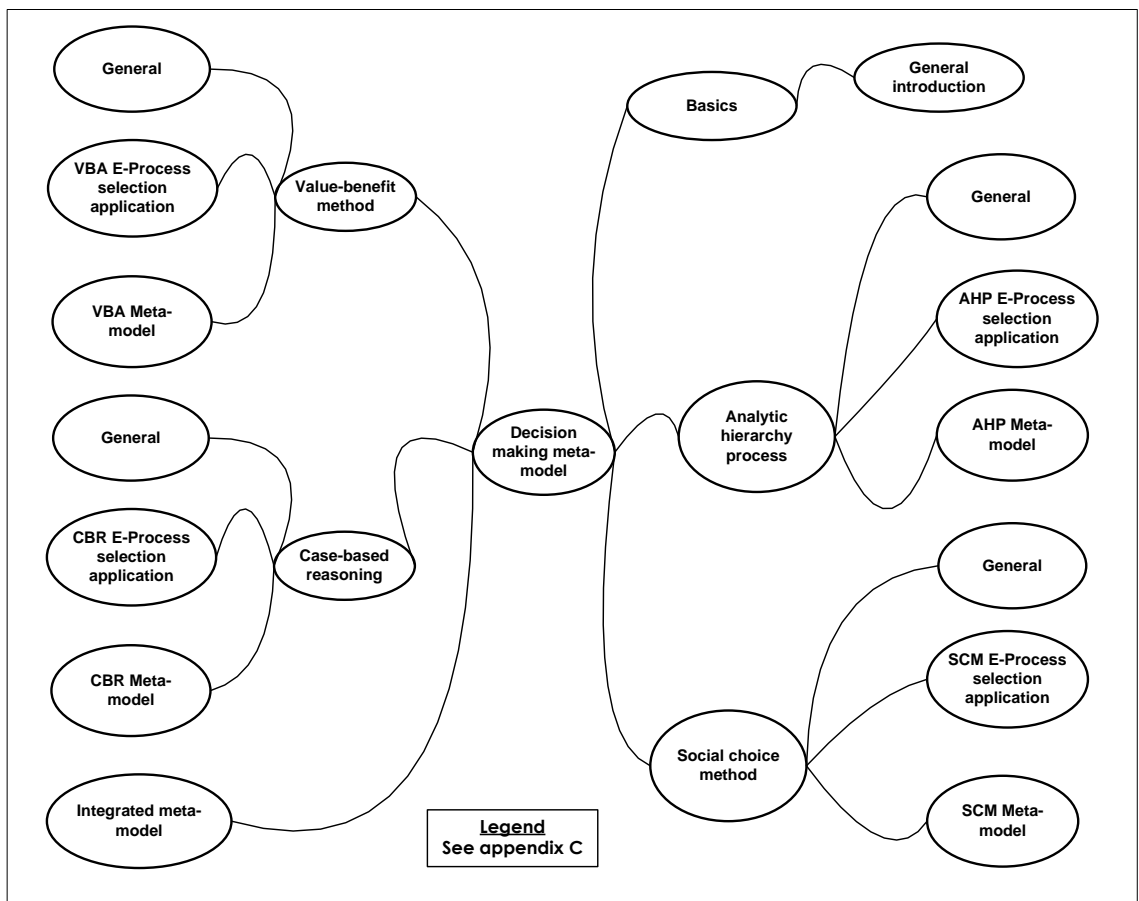


Figure 7.1 Aspects of a decision making meta-model

Chapter 6, in this thesis, introduced and discussed decision making. This chapter continues that discussion by focusing on the meta-model used for each of the MCDM methods trialled in this thesis. Different decision making processes are investigated and trialled in order to help the decision making when selecting a specific e-Process.

The decision making processes that are being used are the Value Benefit Method (VBA), the Analytic Hierarchy Process (AHP), Case-Based Reasoning (CBR) and Social Choice Methods (SCM). Figure 7.1, on the previous page, gives a brief overview of the topics covered in this chapter. The decision making processes are briefly introduced in this chapter and a meta-model presented for each.

7.2 Value Benefit Analysis (Weighted score) (VBA)

Value Benefit Analysis is one of the multi-criteria decision methods (Mantel, et al., 2005). It is a method that allows the selection of one option, in our case one e-Process, from a set of options based on an evaluation process. This procedure does not focus on cost as a criterion, but rather specifies the identification of a number of criteria (see chapter 4 – quality aspects) that can be used for evaluating and comparing the different options against one another. The application of Value Benefit Analysis is also called the weighted scoring method.

7.2.1 Unweighted 0-1 factor method

Project Name: _____		
Criteria rater: _____		
Date: _____		
Criteria	Qualifies	Does not qualify
Has a potential market share	X	
No new technical expertise required		X
No training required	X	
Rate of return more than 15%	X	
Impact on company image with customers		X
Impact on company image with competitors	X	
Total:	4	2

Figure 7.2 Sample Project Evaluation Form

Scoring methods were developed originally to overcome some of the biases of other selection methods (Mantel, et al., 2005). A simple scoring method developed is the unweighted 0-1 factor method, which lists multiple criteria of significant interest to the people making the decision on what to select. The selection committee is given a list of the chosen criteria and a portfolio of each of the potential projects. This committee

then checks off those criteria that the specific project fulfils. A sample form for the 0-1 Value Benefit Selection Method can be seen in Figure 7.2 on the previous page. The "winning" project is the project that has the largest number of check marks (Mantel, et al., 2005).

7.2.2 Weighted scoring method

In the selection process discussed above, it was, however, found that often not all criteria are of equal importance. Various projects may satisfy each criterion to different levels. This led to the development of a more sophisticated method of selection called the weighted scoring method (Clemens, 2002; Mantel, et al., 2005).

The weighted scoring method is a disciplined subjective approach, using multiple criteria, in selecting a rationally favoured choice from a field of candidate choices (Clemens, 2002).

Sum for a specific project is:

$$S_j = \sum_{i=1}^h S_{ij}W_j$$

s_{ij} = the score of the i th project on the j th criterion

w_j = the weight of the j th criterion

i = 1 to the number of projects and

j = 1 to number of criterion

Figure 7.3: Mathematical form of the weighted scoring method

See Figure 7.3, for the mathematical formula for the weighted scoring method. In this method a set of $C = \{c_1 \dots c_n\}$ criteria is identified for use to consider and select between different software projects – in our case development processes. Too few criteria (< 3) may not work effectively. The importance of each of the criteria is defined in terms of weights (w). The sum of all the weights over all the n criteria is usually set to 1.0 or 10. It is suggested that the number of criteria used in the evaluation be restricted for ease of use and to exclude those criteria that only marginally contribute to the final selection (Clemens, 2002; Mantel, et al., 2005). The value assigned to each weight can be determined in a number of ways. These include the results of surveys, group

composite beliefs such as the average of values between a group of members and the subjective belief of an individual (Mantel, et al., 2005).

In order to choose between the different projects it is also necessary to assign a score s_{ij} to each of the projects (i) for each of the criterion (j). Usually a 5 point score will be used (Mantel, et al., 2005). Excellent performance of a specific criterion will produce a top score of five while three is reserved for an average performance and one for a poor performance of that specific criterion. This 5 point scale can be used for qualitative as well as quantitative data thus making it possible to use both criteria which can easily be measured (hard data) as well as soft data such as the fit with the organisation's goals, comfort, ease of use and others (Mantel, et al., 2005). Each score is multiplied by its corresponding criterion weight and then for each process these values are summed to determine a total weighted score.

An example of a basic application of this method can be seen in Figure 7.4.

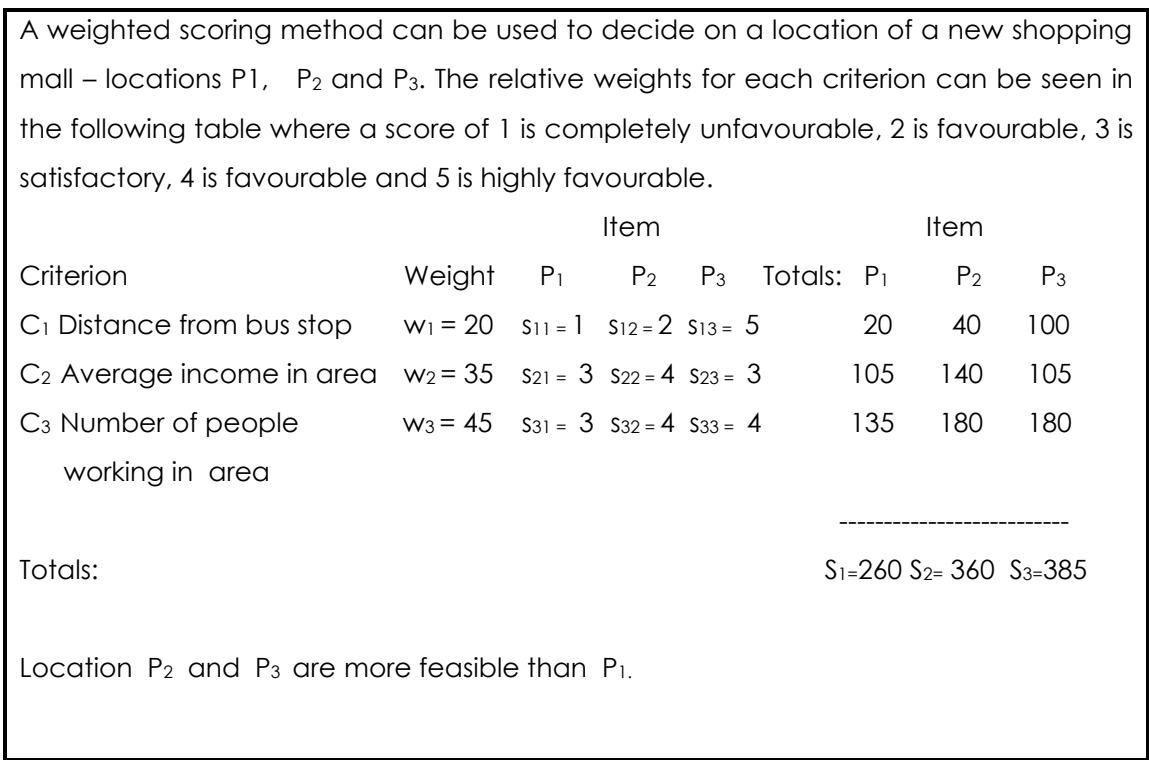


Figure 7.4: Example Weighted Scoring Method (Mantel, Meredeith, et.al., 2005)

The VBA e-process ontology and specific mathematics will be further explored in the next section while a case study is presented in chapter 9 of this thesis.

7.2.3 VBA e-Process ontology

The research contained in this section is based on parts of the following publications; see Albertyn and Kaschek (2005c), Albertyn (2005d) and Albertyn and Kaschek (2004b). Similar to a number of researchers in the area (Fettke & Loos, 2003; Opdahl, et al., 2001; Wand, et al., 1999), following Tom Gruber, this thesis considers ontology as a specification of a conceptualisation that is shared by a number of people. For the purpose proposed here, i.e., assessing e-Processes, it is best that this specification takes the form of a list of concepts (including definitions of these) that enables the characterising of the e-Processes. The quality aspects discussed in chapter 4 are proposed.

Once the associated scales have been assigned to the e-Process quality aspects then e-Processes can be conceptualised as points in a number space, the dimension of which is the number of quality aspects assumed. In order to scale up or drill down respectively, the dimensions in this space are aggregated or decomposed. Considering super- and subspaces in this way allows for the inclusion of further quality aspects if that should be required. Also, quality aspects might be blinded out for particular purposes such as managerial discussions regarding e-Process selection. This means that certain quality aspects may not or must not affect the selection process in which case the weight factors of these quality aspects are simply set to zero.

At the highest level of abstraction high level quality aspects are used to describe e-Processes. With these high level quality aspects the aim is to address e-Process aspects which can be assumed to be important when choosing one of them for a particular development task. Refining the quality aspects, grouping or naming them differently, or adding some of those not considered, may impact a particular choice being made for a given project. However, this would not invalidate this particular selection method.

The e-Process selection methodology associates with each e-Process profile a vector of numbers and for each quality aspect a weighting factor. This will enable the quantitative comparison of the e-Processes and allow the one with largest weighted sum to be selected as the chosen one. To further improve the selection process, weak-point analysis will also be used, see Böhm and Wenger (1996) for more detail regarding this method, to analyse different types of improvements to the e-Process selected that would be reasonable with respect to a development task at hand. Both of these methodological ideas, however, presuppose that for each of the quality aspects a numerical scale is available.

7.2.4 E-Process VBA Selection Methodology

This section defines a meta-heuristic that is based on the idea of patterns. An e-Process pattern here is considered as a triplet (context, problem, e-Process) where it is assumed that the e-Process is an acceptable solution to the problem in the given context. For each e-Process a score is introduced to measure how well it suits as a problem solution in a given context. The meta-heuristic can be defined as:

```

LOOP
    CHOSE an e-Process from the list
    APPLY the e-Process
    ASSESS its success
    UPDATE the e-Process score
ENDLOOP

```

To formulate this initial heuristics some conventions are introduced. Let k represent any number of high level quality aspects d_1, \dots, d_k , m represents the second level quality aspects c_1, \dots, c_m and n represents e-Processes P_1, \dots, P_n . Then each e-Process can be represented by an m -tuple of numbers between 0 and 1. For each high level quality aspect $d_j, j \in \{1, \dots, m\}$ or second level quality aspect $c_i, i \in \{1, \dots, m\}$, and each process $P_j, j \in \{1, \dots, n\}$, an enterprise staff, who is an expert in the field, is asked to determine the weight $w(1), \dots, w(m)$ of the quality aspects $c(1), \dots, c(m)$, the weight of high level quality aspect d_1, \dots, d_k as well as the performance $p(1,j), \dots, p(m,j)$ of process P_j with respect to quality aspects $c(1), \dots, c(m)$ respectively. The numbers $w(i)$ are chosen such that $\sum w(i) = 1$ and $0 \leq w(i) \leq 1, \forall i \in \{1, \dots, m\}$. The initial heuristics is then:

1. $C := \emptyset$
2. Chose $J \subseteq \{1, \dots, n\}$ such that for $j \in J$ the number $S(j) = \sum_{i \in \{1, \dots, m\}} d(i) * w(i) * p(i,j)$ is maximal and define $C := J$, observe to chose J maximal. Set $C := C \cup J$.
3. For all $j \in \{1, \dots, n\} \setminus J, k \in \{1, \dots, m\}$ perform a sensitivity analysis, i.e., Calculate $S(k,j) = \sum_{i \in \{1, \dots, m\} \setminus \{k\}} w(i) * p(i,j)$
Chose sets W, P such that $w(k) \in W, p(k, j) \in P$ and determine $T(k,j) = \max\{w * p \mid w \in W, p \in P\}$. It will often be convenient to chose W and P such that $W = \{c_w + h_w * \Delta_w \mid h_w \in \{0, \dots, r_w\}\}, P = \{c_p + h_p * \Delta_p \mid h_p \in \{0, \dots, r_p\}\}$.
If $S(k, j) + T(k, j) > S(j)$, then the values w, p for which the maximum $T(k, j)$ was achieved need to be investigated. If these are reasonable and acceptable, then redefine $C := C \cup \{k\}$.
4. Do weak point analyses for each candidate in $j \in C$, i.e., determine those quality aspects with high impact (weight higher than for, e.g. 70 % of the

quality aspects) and low performance (performance lower than for, e.g. 70 % of the quality aspects).

- a. For each of the weak points consider the performance assessment and weight. If one of these should be corrected then do so.
 - b. If weak points remain after a. then either $C := C \setminus \{j\}$ or replace P_j by an improved version Q scoring no less than P_j , and assess it.
5. If the weak point analysis in 4 does not change anything then chose among the candidates in C according to a predefined strategy. Otherwise go back to 3.

Note that the definition of weak point used here is somewhat arbitrary in that the thresholds of 70 % are not justified. One can thus attempt to tune the heuristics changing these values. In particular the threshold values could be chosen differently. In the next section the Value Benefit Method meta-model is presented.

7.2.5 E-Process Selection VBA Meta-Model

In Figure 7.5 and Table 7.1, on the following page, the different entities required to define the meta-model for Value-Benefit Analysis Method and their descriptions are stored.

These quality aspects include identification of all the Candidate e-Processes that are going to be used to calculate the “winner” using VBA; capturing all the quality aspects, both the high level aspects as well as the second level aspects. Each high level quality aspect may have 0 to many second level quality aspects associated with it; the experts (assessors) identify the scores for each of the e-Processes and their associated quality aspects (This is then semi-permanently stored in the system. The project will, for each of the quality aspects, have a one-to-many association with the quality scores. After applying the VBA calculations (see the case study in chapter 9), weak-point analysis as well as sensitivity analysis is performed on the results to determine whether there might be a better match. This information will also be captured and forms part of the meta-model.

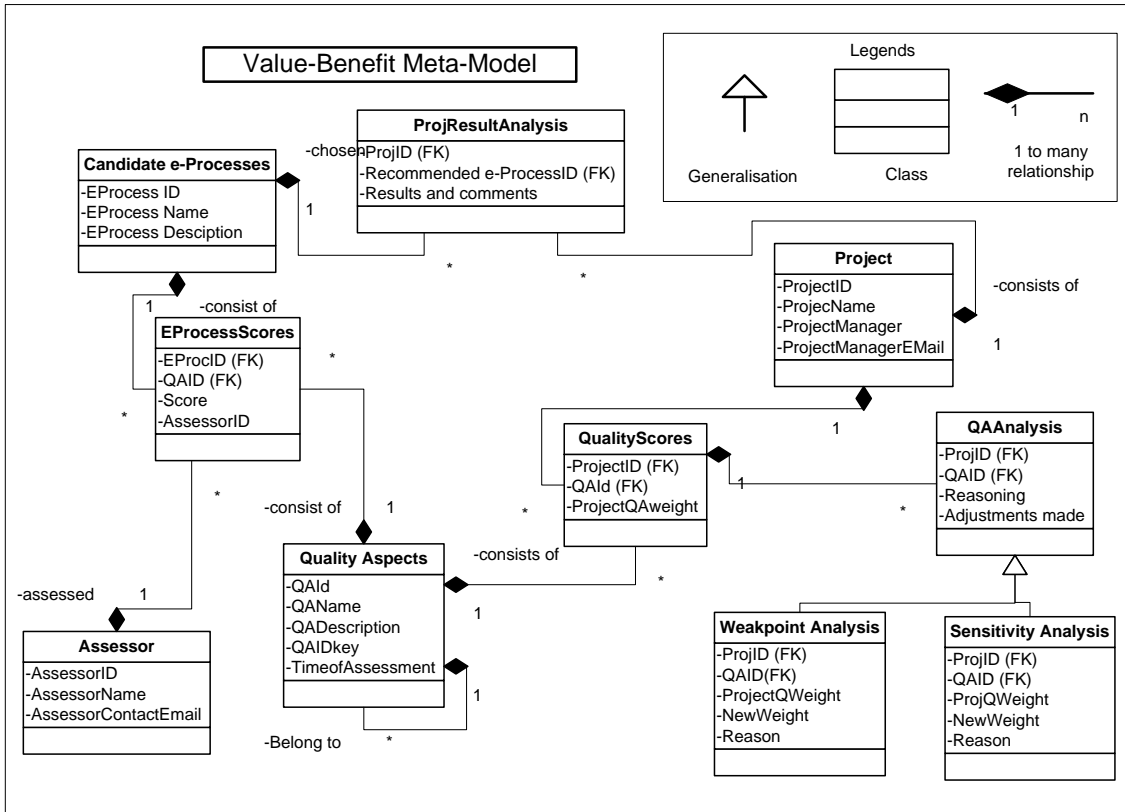


Figure 7.5: e-Process Selection VBA Meta-Model

e-Process Selection VBA Meta-Model Data Dictionary		
Name	Description	Type
Candidate e-Processes	The e-Processes being taken into consideration.	Class
eProcessScores	The score for each e-Process in terms of each quality aspect as estimated by experts.	Class
Assessor	The expert of a specific e-Process' estimating eProcessScores.	Class
Quality Aspects	A list of all quality aspects including the high level quality aspects and the second level quality aspects.	Class
Quality Scores	The weight assigned to a specific quality aspect for a specific score.	Class
ProjResultsAnalysis	The result of the selection process.	Class
Project	Detail about the project for which an e-Process is required.	Class
QAAnalysis	Capturing of reasons for adjustments made to results.	Class
Sensitivity Analysis	The results obtained applying SA.	Class
Weakpoint Analysis	The results obtained applying WA.	Class

Table 7.1: e-Process Selection VBA Meta-Model Data Dictionary

It is difficult to appreciate the relationship between the input into a mathematical model and the model dependant output variables. In order to understand this relationship it is necessary to understand the behaviour of the model when changes are made to the input values. Sensitivity Analysis is one way of investigating this. Sensitivity Analysis (SA) is the study of how the variation in the output of a model (numerical or otherwise) can be apportioned, qualitatively or quantitatively, to different sources of variation (Saltelli et al., 2007). In other words, sensibility analysis is the understanding of what happens to the output in your model when the input is changed. This helps you to understand the intrinsic value of your model. In the decision models in this thesis which involve many input variables sensitivity analysis is in some instance being used as an essential ingredient of the definition of the e-Process selection model and quality assurance of the deliverable.

The sensitivity analysis done here is of the type where the decision model is executed a number of times for different combinations of input factors and determining whether this has a significant impact on the result. By executing sensitivity analysis on the result from the e-Process Selection Process a better understanding can be achieved of the type of contribution made by the different factors and system performance in general. Sometimes no visible change might be the result, in which case we have improved the certainty of the selection.

Weak point analysis is used to optimise the selection process by identifying weak points in the calculation process, by removing those that are either very high or very low, from the calculation. These points are identified and then removed from the calculation in order to determine whether one or two of the factors used in the equation may have biased the result.

7.3 Analytic Hierarchy Process

The Analytic Hierarchy Process (AHP) is a decision making model that supports complex decision making processes. This process was developed by Saaty (1990a) and has been used extensively since.

Figure 7.6, on the next page, is used to quantify the different components of a problem and then to use these to evaluate alternative solutions to the problem and determine the best matching solution to the problem. Using AHP has the advantage of allowing quantitative as well as qualitative considerations when evaluating alternative solutions (Jayaswal & Patton, 2006). AHP is powerful in that it allows the ranking of the elements

and also provides the relative worth of each of the elements (Jayaswal & Patton, 2006). According to Jayaswal and Patton (2006), there are multiple quality aspects that play a role in the software development process. Applying AHP using multiple quality aspects can be difficult. One option to manage the complexity is to minimise the number of the quality aspects used for evaluation of the development processes and then to use AHP as a decision tool in this complex multi-criteria decision making process.

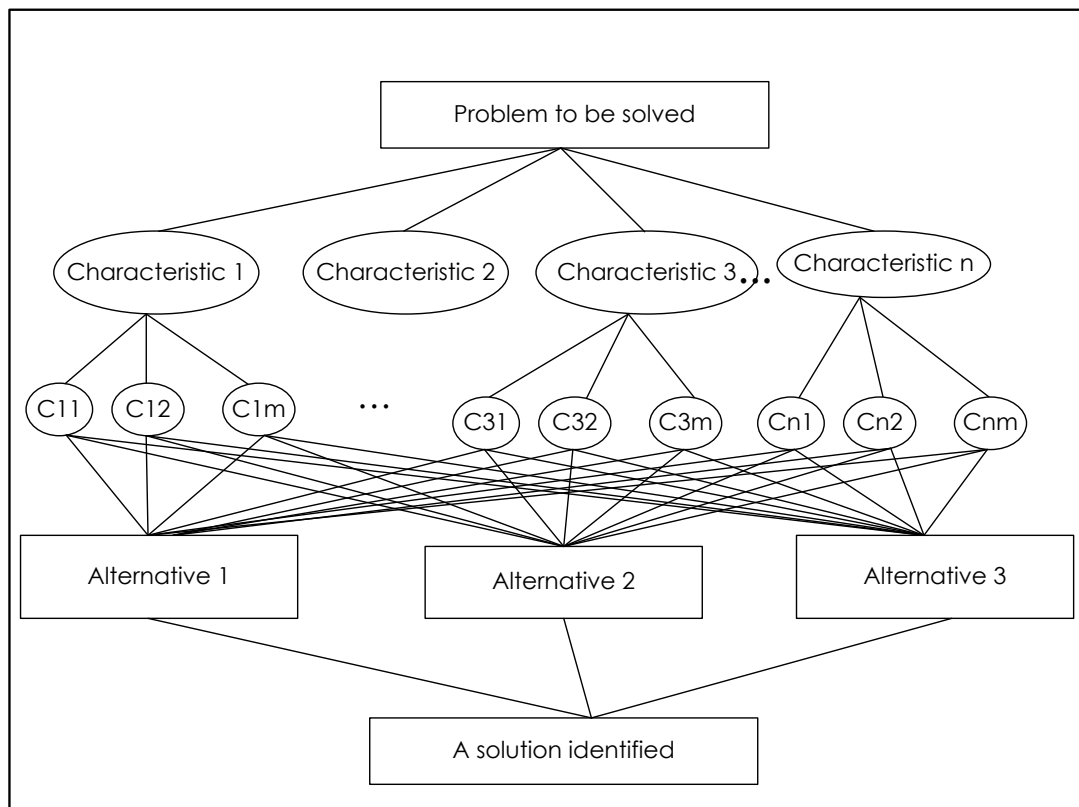


Figure 7.6: Analytic Hierarchy Process

The AHP decision model is executed as follows:

- Identify the quality aspects that are going to be used to describe the problem.
- Identify the different alternative solutions to the problem.
- For each quality aspect and each pair of alternative solutions the developers give their preferences.
- Developers also need to rank the different quality aspects in terms of their significance. $C1 > C4 > C2$ etc.
- The matrices of preferences are then evaluated and a score calculated.

7.3.1 An AHP Example

Suppose a decision needs to be made on which of a number of job offers to accept (AHP example, 2008). Suppose the job offers come from ABC Computing (A), Software Expert (B) and Create IS (C). Suppose the person who needs to decide on a job offer knows that the following factors are important to him/her: Location, salary and type of job.

Step 1: Decide on the relative importance of each of the objectives (named i and j) and rank them using the following scale:

$A_{i,j}$ (Scale for comparing object i to objective j)	Description of the comparison
1	Both objectives of equal importance
3	i is weakly more important than j .
5	i is strongly more important than j .
7	i is more strongly more important than j .
9	i is absolutely more important than j .

Table 7.2 Pair-wise comparison values

In the table $a_{i,i}=1$ and if $a_{i,j} = k$ then $a_{j,i} = 1/k$;

Assume this person's preferences can be stated as follows:

	Location	Salary	Type of work
Location	1	1/7	1/3
Salary	7	1	2
Type of work	3	1/2	1

Table 7.3 Object Preferences

In order to calculate the overall weight, a number between 0 and 1, each entry is divided by the sum of the column it appears in.

	Location	Salary	Type of work	Average
Location	0.09	0.09	0.10	0.09
Salary	0.64	0.61	0.60	0.62
Type of work	0.27	0.30	0.30	0.29
				$\Sigma = 1$

Table 7.4 Weights on Objectives

Step 2: Evaluate all the jobs on each of the objectives and normalise. For instance if we evaluate all jobs in terms of salary we could end up with the following table:

Salary:	A	B	C	Normalised			Avg
				A	B	C	
ABC Computing (A)	1	1/7	1/4	0.08	0.10	0.06	0.08
Software Expert (B)	7	1	3	0.58	0.67	0.70	0.65
Create IS (C)	4	1/3	1	0.33	0.23	0.24	0.27
							$\Sigma = 1$

Table 7.5 Salary scores and relative salary scores

Table 7.5 shows that the salary value for job B is 65%, 8% for job A and 27% for job C. If a similar process is followed for location and type of work the results can be seen in Table 7.6.

	A	B	C
Location	0.25	0.49	0.26
Salary	0.08	0.65	0.27
Type of work	0.40	0.30	0.30

Table 7.6 Relative scores per objective

A total value for each job can now be calculated:

$$\text{Job A: } 0.25 \times 0.09 + 0.08 \times 0.62 + 0.4 \times 0.29 = 0.19$$

$$\text{Job B: } 0.49 \times 0.09 + 0.65 \times 0.62 + 0.30 \times 0.29 = 0.53$$

$$\text{Job C: } 0.26 \times 0.09 + 0.27 \times 0.62 + 0.30 \times 0.29 = 0.28$$

The value for job A is thus 19%, 53% for job B and 30% for job C – which makes job B the recommended selected job. This process and specific mathematics of AHP will be further explored in the next section and applied to a case in Part C.

7.3.2 AHP e-Process Selection Model

The research presented in this section follows parts of the research as published in Kaschek, Albertyn, Shekhovtsov and Zlatkin (2007). AHP is a very well-known and successfully applied method for multi-criteria decision making. Obviously the e-Process selection task is an instance of that decision making task. This method thus is justifiable. According to Ahituv and Neumann (1990) Simon's model of decision making has the three stages intelligence, design, and choice. The focus here is on choice, because

only existing software processes are considered. The simplified supposition is then that the decision maker identifies some existing e-Processes and the choice is made by quantitative assessment used to identify the top-scorers and then selecting one of these.

As discussed in chapter 4, there are a number of quality aspects which are considered essential for any method for e-Process selection. Once again these high level quality aspects and second level quality aspects are used to demonstrate the use of the analytic hierarchy process in the selection of an e-Process.

The recommendation here, is that the expertise required for this selection task is provided by humans in the role of either an e-Process expert or system quality expert. The AHP enables the blending of the experts' expertise to select the e-Processes best suited for the problem at hand. Choosing the experts is not necessarily trivial or cheap. The recommendation is that the qualification profiles of the designated developers are considered. If that profile indicates sufficient e-Process and system quality expertise then the recommendation is to use these developers as experts. Also recommended is that any individual with skills for both expert roles, be used – if available. Expert selection needs to consider areas like affiliation, area of competence, standing, availability, price, etc. According to Nureg-1150 (1989) expert selection should consider demonstrated experience, expert versatility, expert group diversity, and expert cooperation.

The AHP-based selection follows the following steps for the purposes of this research:

- Quality aspect ranking,
- e-Process ranking
- Scoring and first recommendation.
- Sensitivity analysis on the data
- Final recommendation

7.3.3 Quality aspect ranking and e-Process alternative ranking

When applying AHP, there is a hierarchy structure. In this AHP e-Process selection:

- The goal is seen as the eCIS to be developed and it is placed at the root of the hierarchy structure;
- The high level quality aspects, as identified in chapter 4, is on the second level of the hierarchy structure;
- Each of these high level quality aspect branches has associated second-level quality aspects - this is the third level of the hierarchy structure.

- The bottom level of this hierarchy structure is the set of e-Process alternatives, as introduced in chapter 5.

Ranking occurs when e-Process experts pair-wise compare the selected quality aspects as well as rank each of the e-Process alternatives in terms of the second level quality aspects.

The quality aspects are sorted, as discussed in chapter 4, in terms of high level quality aspects and second level quality aspects. The pair-wise quality aspect comparison needs to occur within each branch of the hierarchy structure.

For the purposes of comparing two aspects x and y where $x, y \in \{RUP, AM(XP), OSS, SBUP\}$ or $x, y \in \{\text{all quality aspects}\}$ then define a predicate $\partial_i(x, y)$ for $i \in \{1, 3, 5, 7, 9\}$ to indicate equally preferred, moderately preferred, strongly preferred, very strongly preferred, extremely preferred, respectively.

Let H be the high level quality aspects and S be the second-level quality aspects respectively. With $s \in S$ and $h \in H$, let X_h be the set of second-level quality aspect into which s has been decomposed with S the disjoint union of all X_h . Let $q \in Q = S \cup H$.

In order to execute quality aspects ranking, the system quality experts apply the pair-wise comparison technique for assessing the relative importance of the quality aspects two times. Apply the ranking on the high level quality aspects first and then their decompositions and thus obtain weights for each.

In order to do the e-Process ranking, E is defined as the set of e-Process experts, and X a set of e-Processes. It is now possible to denote that expert e judges $\partial_i(x, y) = \text{TRUE}$ with respect to quality aspect s and define $\alpha: S \times E \times X \times X \rightarrow \{\partial, 1/\partial \mid \partial \in \{1, 3, 5, 7, 9\}\}$ for a predicate $t(e, \partial, x, y, s)$ and thus define: $\alpha(s, e, x, y) = \partial$ for $t(e, \partial, x, y, s)$ and $\alpha(s, e, x, y) = 1/\partial$ for $t(e, \partial, y, x, s)$. This mapping is called a comparison mapping. In terms of $x=y$, we can define $\alpha(s, e, x, x) = 1$.

Similar to Kaschek(2007), let be $q \in Q$ a quality aspect, $e \in E$ an expert, and for brevity $X = \{1, \dots, n\}$, then for the restriction $A(q, e)$ of α to $X \times X$ holds $A(q, e)(i, j) = \alpha(q, e, i, j)$, for all i, j . This can be presented in a matrix. The elements of the matrix are the results of all pair-wise e-Process comparisons (Saaty, 1990a; Saaty, 1990b):

$$A(q, e) = \begin{pmatrix} a(q,e,1,1) & \dots & a(q,e,1,n) \\ \dots & \dots & \dots \\ a(q,e,n,1) & \dots & a(q,e,n,n) \end{pmatrix}$$

The matrix can be normalised and then eigenvectors can be determined for these matrixes. Solving eigenvectors consist of squaring the matrix repeatedly, each time calculating the row sums and then normalising them. Continue with this process until the difference between successive iterations is small, where small is defined beforehand.

Using software designed by Raimo P. Hämmäläinen (2008), it is possible to employ a version of AHP, which allows for pair-wise comparisons in a consistent pair-wise comparison matrix.

7.3.4 Scoring and first recommendation

Using the mathematics from Forman and Selly (2001) to calculate AHP and as defined in Kaschek (2007); it is now possible to normalise and then calculate the first set of "winners". For this then use the maximum value $f^*_q = \max \{f_{q,x} \mid x \in X\}$ for normalisation, as denoted above, with the q-score $w_{qf_{q,x}} / f^*_q$ of x as $p_{q,x}$. The score p_x of $x \in X$ is then defined as $\sum_{q \in Q^2} p_{q,x} / \sum_{x \in X} \sum_{q \in Q^2} p_{q,x}$. This then produces the set of e-Processes best suited for the problem at hand and can be defined as the set $\{x \in X \mid p_x = \max \{p_y \mid y \in X\}\}$.

7.3.5 Sensitivity analysis

Sensitivity analysis is used to identify possible poor decisions made with data input. This technique is used to identify the quality aspects and e-Processes which will, with the final ranking be the most sensitive to changes in the data. The pair-wise comparisons for both of the e-Processes and the quality aspects can be biased by the experts' subjective views and errors they might make. Sensitivity analysis is done, after the initial execution of AHP, on each e-Process quality aspect by creating a sensitivity diagram.

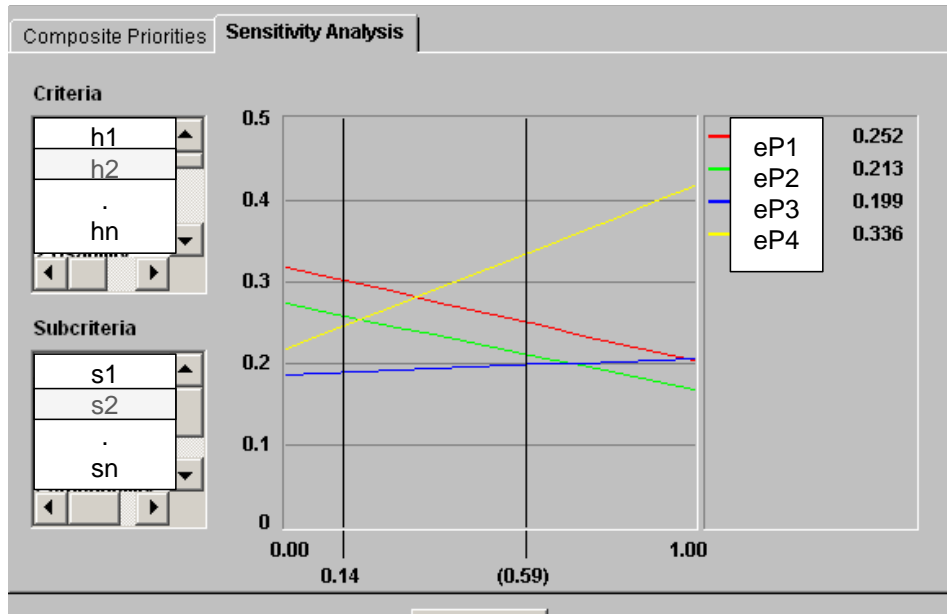


Figure 7.7: Sensitivity diagram example

The idea when doing sensitivity analysis is to perform this analysis on all the quality aspects by letting one quality aspect at a time vary in small steps in an interval I . As can be seen in Figure 7.7, all the e-Processes are linearly represented. h_2 is the diagram indicates the high level quality aspect chosen, with s_1 to s_n all the related second level aspects. In the diagram s_2 is the selected second level quality aspect. This is the quality aspect that is depicted in the sensitivity diagram. eP_1 to eP_4 represents the chosen alternatives – our case possible e-Processes. Dragging the vertical line horizontally signifies changing the quality aspect scores of the admissible e-Processes. Using this approach, then at an intersection of any of the depicted sensitivity diagrams, there can be a so-called rank reversal – another winner might be indicated.

By changing the relative importance of one of the quality aspects, there might be an impact on the overall performance of the e-Processes. By increasing/decreasing a quality aspect, there might be the impact of ending up with a different “winning” e-Process. By dragging the bar across one sensitivity diagram, there are corresponding changes to all the sensitivity diagrams (Jayaswal & Patton, 2006).

As a result of the sensitivity analysis of the whole set of quality aspects and e-Processes using the set of all possible sensitivity diagrams), the following sensitivity measures (Triantaphyllou & Sanchez, 1997) and used in Kaschek (2007) can be obtained and presented to the users of this method:

1. *Quality aspect criticality degree*. This measure is calculated for every quality aspect. The criticality degree is the minimal percentage of change of the rating for

any rank reversal to occur for any of the quality aspects. The criticality degree can be calculated using the distance between the vertical line in the sensitivity diagram and the closest intersection of any e-Process functions' graphs. In Figure 7.7, it is the distance to the intersection of the functions of eP1 and eP4. With not intersection it is an indication that no variations of the quality aspect's rating will affect the ranking of e-Processes. Ranking quality aspects according to this degree lets the user observe the differences in their sensitivity to the data variations.

2. *Percent-any critical quality aspect*. Find the quality aspect with the smallest criticality degree among all the quality aspects. This quality aspect can be considered as the most sensitive to the data variations. If no such aspect is found the e-Processes ranking can be considered stable.
3. *Percent-top critical quality aspect*. This is the quality aspect with the smallest relative change for any rank reversal to occur among the top-scoring e-Processes. This quality aspect can be considered as the most important for our selection task, because the variations of its weight most likely affect the selection outcome. If no such aspect is found, the selection outcome can be considered stable.
4. *E-Process criticality degree*. This is the smallest aspect-related criticality degree for the e-Process. Ranking e-Processes according to this degree lets the user observe the differences in strength of their AHP-produced ranking.
5. *Most critical e-Processes*. These are the e-Processes with the smallest criticality degree. Each most-critical e-Process is among the e-Processes for which the ranking is most unclear and weak. In fact, the ranking of the most critical e-Processes should be investigated further. If no most critical e-Process can be found the AHP-produced ranking can be seen as being stable.

7.3.6 E-Process Selection AHP Meta-Model

In Figure 7.8 and Table 7.7, over the page, the meta-model aspects for AHP are presented.

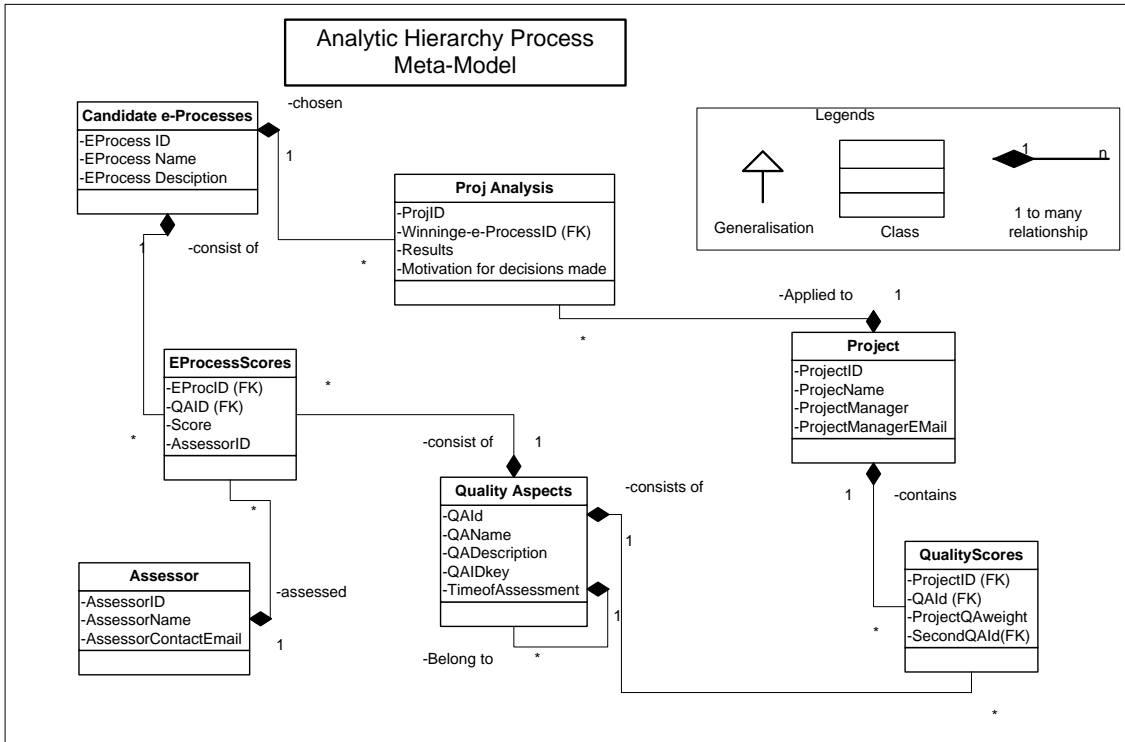


Figure 7.8: e-Process Selection AHP Meta-Model

e-Process Selection AHP Meta-Model Data Dictionary		
Name	Description	Type
Candidate e-Processes	The e-Processes being taken into consideration.	Class
eProcessScores	The score for each e-Process in terms of each quality aspect as estimated by experts.	Class
Assessor	The expert of a specific e-Process' estimating eProcessScores.	Class
Quality Aspects	A list of all quality aspects including the high level aspects and the second level quality aspects.	Class
Quality Scores	The weight assigned to a specific quality aspect for a specific score.	Class
ProjResultsAnalysis	The result of the selection process.	Class
Project	Detail about the project for which an e-Process is required.	Class

Table 7.7: e-Process Selection AHP Meta-Model Data Dictionary

When applying the Analytic Hierarchy Process the Candidate e-Processes which is going to be used to calculate the “winner” using AHP needs to be identified; next the quality aspects need to be identified and captured. Each high level quality aspect may have 0 to many second level quality aspects associated with it – to exclude a whole group of quality aspects set the top-level aspect to 0; the experts (assessors) identify the

scores for each of the e-Processes and their associated quality aspects (This is then semi-permanently stored in the system) – similar to VBA; the project will for each of the quality aspects have a one to many association with the quality scores. The quality scores will have associations with other quality scores in order to accommodate the pair-wise comparison of AHP. After applying the AHP calculations, reasons and motivation for choices are captured in the analysis class.

The application of AHP to a case study, is discussed further in part C, specifically how it is used to evaluate the different e-Processes in order to recommend one of the solutions.

7.4 Case Based Reasoning Approach

Case Based Reasoning (CBR) is the approach where new problems are solved by comparing them to a number of historical cases with solutions and then choosing the “best” previous case which is nearest to this new case. The solution of the historic case then becomes, after modification, the solution of the new case. The research published in this section is based on the following publications; see Albertyn and Kaschek (2007) and Albertyn (2007).

A number of research papers have been published on methodology selection for different domains such as frameworks for comparing object-oriented modelling (OOM) tools (Kaschek & Mayr, 1996) and analysis methods (Kaschek & Mayr, 1998). They built two detailed lists of quality aspects for these selection method frameworks. Applying Case-Based Reasoning (CBR) in different domains (finance, medical, etc.) involving decision-making has been successful (Macaya et al., 2002). Access to relevant case histories of different problems reduces the requirement for problem analysis by reuse of the old problem solutions (Cunningham & Bonzano, 1999). The next section discusses the CBR technique.

7.4.1 Case Based Reasoning Technique Used

CBR is a computer technique, which combines the development of an actual experiences knowledge base with the simulation of human reasoning. This means, search for similar situations in the past and re-use those experiences (Leake, 1996; Mansar et al., 2003). When a specific problem needs to be solved, similar actual occurrences are retrieved and adapted from the knowledge base. CBR is inductive (based on measuring case similarity), rather than deductive (based on logic and consistency) (Cunningham & Bonzano, 1999).

The four basic steps (also known as the four Rs) of a CBR process are (Cunningham & Bonzano, 1999; Watson & Marir, 1994), namely:

- Retrieve: When a eCIS project has been specified for which an e-Process needs to be found then retrieve a number of cases from the case database that are relevant to the problem at hand. These “old” cases will have information concerning the project that was solved, the method/s used to derive a credible e-Process and then the actual e-Process selected.
- Re-use: Once a case has been identified as being the nearest to the project at hand then adapt this old solution to fit the new project.
- Revise: Once the solution has been identified it is necessary to simulate the new solution in the real world and adapt or revise if required to fit the new project.
- Retain: Store the revised project, parameters and solution in the case database for future use.

When using CBR, the knowledge base will contain representations of a number of previous cases. A case is a contextual piece of knowledge representing a previous experience. It consists of two parts, the case-content, and the case-context. The latter tells when to use the case-content. The case-content, additionally to a case description, contains the solution applied to the problem represented by the case, and the outcome of applying the solution (Watson & Marir, 1994).

Figure 7.9, on the next page, indicates the different aspects involved in the CBR cycle. Initially the cases are structured and stored in the knowledge base (also called case base). A new case that needs to be solved then gets compared to these previous cases stored in the knowledge base. Heuristic methods are used to retrieve a number of similar cases from the knowledge base. These are then compared to the new case to determine the best fit. A solution is then determined for reuse, based on one or more of these similar cases – if necessary the historical case will be revised and adapted to be a better fit. This solution will be applied to see whether it is suitable. As soon as the solution has been validated then this solution becomes a new case in the knowledge base.

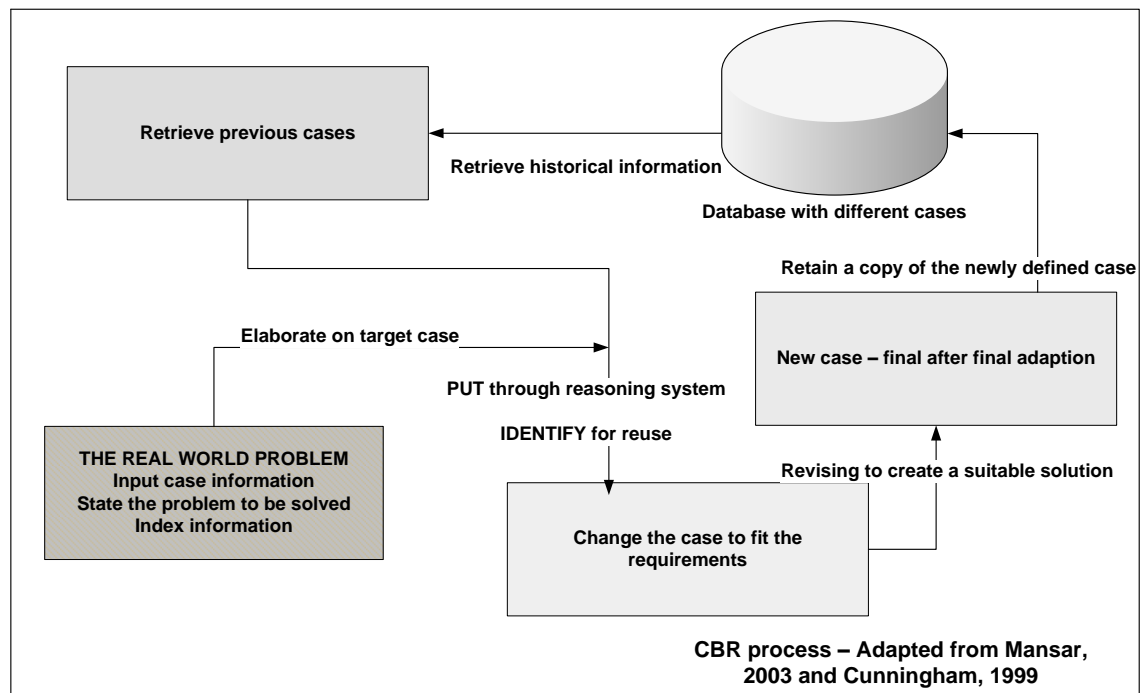


Figure 7.9 CBR process (Mansar et al., 2003)

When building the knowledge base a number of methods can be used to organise, retrieve, utilise and index the past cases (Harrison, 1997). According to Cunningham (Cunningham & Bonzano, 1999) the tasks involved with building up the knowledge base are identification of the “real world” problem and representation of the key components thereof in the knowledge base.

Next it is important to develop the inference mechanism that describes the causal interactions involved in deriving solutions. The inference mechanism is implemented using the knowledge base with the cases of solved problems and a mechanism to retrieve and adapt these cases (Cunningham & Bonzano, 1999).

Retrieving information from the knowledge base consists of:

- start with a full/partial problem description
- identify a set of relevant problem descriptors
- check for similarities with elements in the knowledge base
- return a set of sufficiently similar cases
- select the best matching case

(Cunningham & Bonzano, 1999; Harrison, 1997)

Richter (1998) has identified four different ways, called knowledge containers, by which knowledge can be represented in the knowledge base, namely:

- The vocabulary used,

- The similarity measure,
- The case base, and
- The solution transformation.

The CBR community has widely accepted these as a natural organisation of knowledge.

7.4.2 CBR E-Process Selection

It is assumed that previous cases are used in order to simplify the decision process when selecting a suitable E-Process for a specific eCIS. Case-based reasoning methods are based on the use and outcomes of past experience in order to solve a new problem (Armengol, et al., 2004). A knowledge base system applies its reasoning ability through the explicit representation and use of the knowledge from a specific domain (Diaz-Agudo & Gonzalez-Calero, 2000). The knowledge containers are going to be discussed in the next four paragraphs in terms of case vocabulary knowledge, the use of similarity measures, case knowledge and retrieval knowledge.

7.4.3 CBR E-Process Vocabulary

It is recommended that when developing the knowledge base that the case vocabulary knowledge be standardised. The question that needs to be answered is whether the optimal case representation has been determined for a particular domain? It is also important to decide whether all the cases should be represented in the same way in the knowledge base.

The information captured from applying the other e-Process selection decision making methods is captured in a knowledge base – see integrated meta-model in section 7.6.

7.4.4 CBR E-Process selection and the similarity measure

A task, that needs to be executed as part of CBR, is the similarity measure. This entails that the most similar (or relevant) case/s stored in the knowledge base be identified in order to solve a particular problem, in our case to identify the most suitable e-Process to use to develop a specific eCIS. Currently our case base is still quite small and therefore all cases will be evaluated for similarity.

According to (Coyle et al., 2004) the similarity between a query Q and a case C is defined as the sum of the similarities of its constituent features multiplied by the relevant weights:

$$Sim(Q, C) = \sum_{f \in F} w_f * \sigma_f(q_f, c_f).$$

In this equation w_f is the constituent feature weight, of the similarity measure applied to feature f of Q and C , and F the set of all features. The weights are seen as feature attributes. The similarity measures obviously are more complex. Coyle et al. (2004) use three different kinds of feature similarity measures. These are (1) the exact similarity measure, i.e., the similarity score is 1 if the feature values are equal and is 0 otherwise; (2) difference based similarity measure, i.e., the similarity score depends on the difference of the numerical feature values but not necessarily is 0 for non-equal feature values; and (3) complex similarities, i.e., all other similarity measures.

Using the difference based similarity measure essentially turns a Case Based Reasoning problem into a version of Value-Benefit Analysis. In this method Social Choice Method (see next section) also plays a role. The author believes that the weaknesses of VBA (i.e. that it often is very hard to score items on a scale according to a number of features) can be overcome by the incorporation of SCM method parts that only rely on ranking items.

In selecting an E-Process both measure kinds (1) or (2) could be used. We are going to use (2), i.e., the difference based similarity measure. Currently the values of weights and feature similarities are obtained from the developers who were asked to assess these variables quantitatively. In future the plan is to account for the well-known critique of this approach by using SCM or AHP to obtain scores based on rankings provided by developers. In order to fully apply these methods, a software system dedicated to aid humans in applying these methods, needs to be developed.

7.4.5 CBR e-Process Selection Case Base

The quality of e-Processes can be seen as being multi-faceted, as is often experienced for other complex entities such as information systems (Ghezzi et al., 2004), but individual quality aspects are often considered as too broad and unspecific and are therefore decomposed into lists of second level quality aspects. We exploit a hierarchical approach to define the eProcess by using a two level system of e-Process quality aspects – see chapter 4 (these quality aspects are used here for CBR).

Using both the weighted scoring method in section 7.2 and the analytical selection method in section 7.3, a case knowledge base has been developed – see the meta-

model in section 7.6. Our Case Based, used here, has a number of levels that are used in the comparison. Firstly, there is information about each of the projects/cases being stored. Then for each of the cases we have group values (high level quality aspects) that group a number of quality aspects (second level quality aspects). There are weights assigned to each of the quality aspects.

7.4.6 e-Process and solution transformation

In order to determine the best match, information about the eCIS is entered. This information is used in order to determine which of the cases stored in the Case Base, is the best possible solution for the problem. Once solution/solutions have been identified, the eCIS at hand need to be transformed and becomes one of the cases being investigated. The application of Case-Based Reasoning on a case study will be discussed in Part C of the thesis.

7.4.7 e-Process Selection CBR Meta-Model

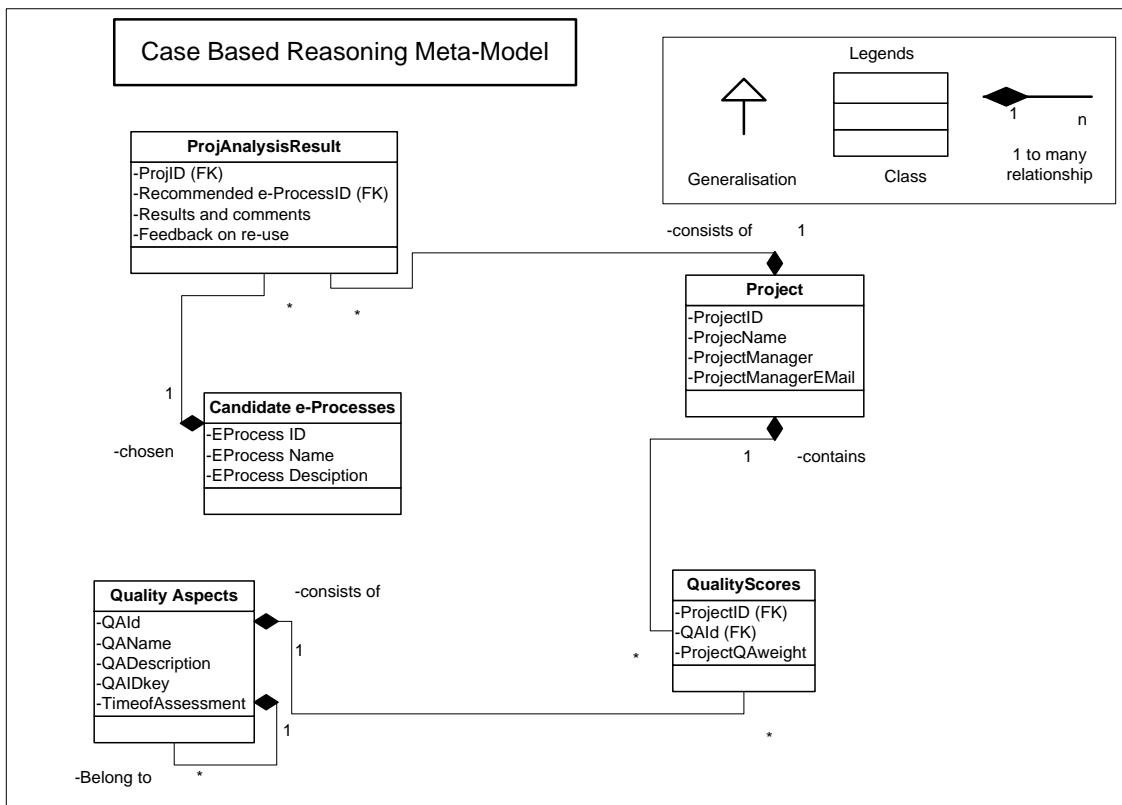


Figure 7.10: e-Process Selection CBR Meta-Model

In Figure 7.10 and Table 7.8, over the page, the meta-model aspects for CBR are presented.

When applying the Case Based Reasoning Process, the requirements of the new eCIS need to be defined. These are then matched with those in the knowledge base that appear to be near matches. The matches are limited to the best 4. Next the four best matches are analysed to determine the best match. The nearest match is adapted to fit the requirements, which forms the solution to the new problem. The solution is then stored in the knowledge base and recommended as the solution to which e-Process to use.

e-Process Selection CBR Meta-Model Data Dictionary		
Name	Description	Type
Candidate e-Processes	The e-Processes being taken into consideration.	Class
Quality Aspects	A list of all quality aspects including the top-level aspects and the primary quality aspects.	Class
Quality Scores	The weight assigned to a specific Quality aspect for a specific score.	Class
ProjAnalysisResults	The result of the selection process.	Class
Project	Detail about the project for which an e-Process is required.	Class

Table 7.8: e-Process Selection CBR Meta-Model Data Dictionary

In part C, CBR is discussed further, specifically how it is used to evaluate the different e-Processes, in order to recommend one of the solutions.

7.5 Social Choice Method (SCM)

The social choice method is based on voting for individual preferences from a list of candidate solutions and then aggregating the results to obtain a result that defines the collective preference (Pomerol & Barba-Romero, 2000, p. 124).

The first recorded use of a voting method was used in the voting of the Roman senate around the year 105 (Gaertner, 2001; Borda count, 2007). Condorcet formulated the voting paradox and in Arrow's book of 1951 the modern form of the theory was defined (Arrow, 1963).

There exist various social choice methods by using different acceptance criteria such as assigning decreasing points to consecutive positions and then ranking the alternatives with this point systems (Borda), Simple Majority of vote, Maximum (Score the alternatives with the worst margin they achieve and rank these according to scores), choose the

ordering with minimal distance to all rankings in the profile, where the distance is defined as the number of different pair-wise relations (Kemeny method)(Bernroider & Mitlohner, 2007). The research in this thesis uses one of these. The detail of this method is discussed later in this section.

When applying the Social Choice Method then, first of all, identify the voters who are going to vote for the "winning" e-Process. Next, the requirements of the new eCIS as well as those e-Processes that are going to be evaluated are defined. Next, the voters vote for their preferred e-Process. The most voted is deemed the "winner" and put forward as the solution to which e-Process to use.

People have been using voting to decide between different choices for a long time. The social choice theory studies what different people prefer from a list of options and then aggregates the results in such a manner that a result can be obtained by defining the collective preference (Pomerol & Barba-Romero, 2000, p. 106). The social choice method is one of the ways that voting can be applied in order to make a decision. Social choice uses the mapping of the preferences of individuals determined by an ordered list of alternatives in order to select one of the alternatives. Different methods can be used e.g. Plurality, Borda and Pair-wise comparison which is similar to AHP (Bernroider & Mitlohner, 2007). In this section the focus is on using the Borda method.

The first recorded use of a voting method similar to the Borda method was used in the voting of the Roman senate around the year 105 (Gaertner, 2001; Borda count, 2007). The more modern format of the Borda method was by Ramon Llull (1232 – 1315) who wrote the manuscripts *Ars notandi*, *Ars electionis* and *Alia ars electionis* and *Blanquema* (Gaertner, 2001). He is seen as the first person to document the Borda count and Condorcet criterion. The method devised by Jean-Charles de Borda in 1770 was used to elect members for the French Academy of Sciences. He published this method in *Memoire sur les elections au scrutiny* in the *Histoire de l'Academie Royale des Sciences* in Paris.

7.5.1 Social choice theory

According to Bernroider and Mitlohner (2007) the problem of social choice can be defined in the following manner:

A set of voters n provide different rankings for the m alternatives, which results in a profile of alternatives a, b, c . If there are three voters then the rankings might be $a \succ b \succ c$, $b \succ c \succ a$, $b \succ c \succ a$. The problem requires that an aggregate ranking is

found $x \succ y \succ z$ such that the preferences of the voters are expressed in the aggregate ranking.

In social choice problems it can be assumed that the profile consists of strict orderings – but similar to Bernroider and Mitlohner (2007) differences are allowed here. There are also a number of aggregate rules and one of the most important ones is the Condorcet criterion. This criterion states that if there is an alternative x that beats all other alternatives in pair-wise comparisons then this is a winner (Bernroider & Mitlohner, 2007). The aggregate rule should not allow a representation of weak order of alternatives or contain cycles.

Further in this voting process dimensions or attributes can be considered. The evaluation of the alternatives is now used to determine the n rankings of alternatives for the m dimensions (Bernroider & Mitlohner, 2007).

Below find some of the major scoring procedures:

- **Simple Majority (SM):** This is a well known procedure based on margins.
 - A positive margin means that x wins against y in pair-wise comparison and results in $x \succ y$ in the aggregate relation.
 - A negative margin means that y wins against x in pair-wise comparison and results in $y \succ x$ in the aggregate relation.
 - A zero margin means indifference and results in $y = x$ in the aggregate relation.

This rule can result in cycles and thus limits the use of this rule in practical problems.

- **Maximin (MM):** This procedure scores the alternatives with the worst margin that they achieve and then ranks them according to these scores.
- **Copeland (CO):** This procedure scores the alternatives with the sum over the signs of the margins they achieve and ranks them according to those scores.
- **Borda (BO):** This procedure assigns decreasing points to consecutive positions.
- **Kemeny (KE):** This procedure chooses the ordering with minimal distances to all rankings in the profile, where the distance is defined as the number of different pair-wise relations. (Bernroider & Mitlohner, 2007)

The plan is to use one of the Borda procedures, therefore, this section continues to discuss some of the Borda aspects. Borda, one of the social choice methods, initially starts by ranking each alternative solution for a given criteria by giving the preferred alternative a value of 1, the second alternative a value of 2, the third one a value of 3 and so on. The social choice, or aggregate pre-order, is calculated by calculating the

total value for each of the alternative solutions. This approach means that the winner is the solution with the least points. In chapter 12 the application of this approach will be modified in order to make the winner the e-Process alternative with the most points (Pomerol & Barba-Romero, 2000, p. 106).

According to Pomerol and Barba-Romero (2000, p.106) m integer alternatives are chosen such that $k_1 > k_2 > k_3 > \dots > k_m \geq 0$. These are called the Borda coefficients. For each of the different criterion j the alternatives are ranked according to a complete pre-order. These rankings are called r_{ij} (where alternative i is associated with the pre-order associated with criterion j). Borda voting, according to Pomerol & Barba-Romero (2000, p.106), is thus for a given n complete preorders \succsim_j (denotes strict preference or indifference) on m alternatives A_1, A_2, \dots, A_m it is the procedure which, for a given alternative A_i , consists of taking the sum of the votes (or $\sum_{k \neq i} v_{ik}$) that it obtains in all possible duels of A_i versus A_k . The alternatives are then ranked in order of the number of votes.

When using Borda all the personal preferences are weighed and the highest scoring one is the winner. Even though Borda can be seen as the peoples' choice the scoring method uses cardinal utilities and disregards personal preferences. Borda focuses on the complete reference profile (Riedl, n.d.). One problem that can be identified with majority voting is that only the top preferences are taken into account (Riedl, n.d.). Another method that can be taken into account is the adjusted Borda method (Black, 1958). This method basically consists of counting the votes for and the votes against a specific candidate and calculating the difference.

7.5.2 SCM Quality Aspects used

As discussed earlier in the thesis, it can be stated that the quality of e-Processes is multi-faceted (as is the case for information systems in general) (Ghezzi et al., 2004). In this section the high level quality aspects and second level quality aspects as discussed in chapter 4 is once again used.

7.5.3 The SCM e-Process Selection Methodology

As explained above, when applying the Borda rule, decreasing points are assigned to consecutive positions, such that in our case 3 points is assigned to the first place, 2 points to the second place, 1 point to the third place and zero to the fourth. In this instance the aim is to once again choose between the e-Process quality aspects given above. The e-Processes, as discussed in chapter 5, are used.

7.5.4 e-Process Selection SCM Meta-Model

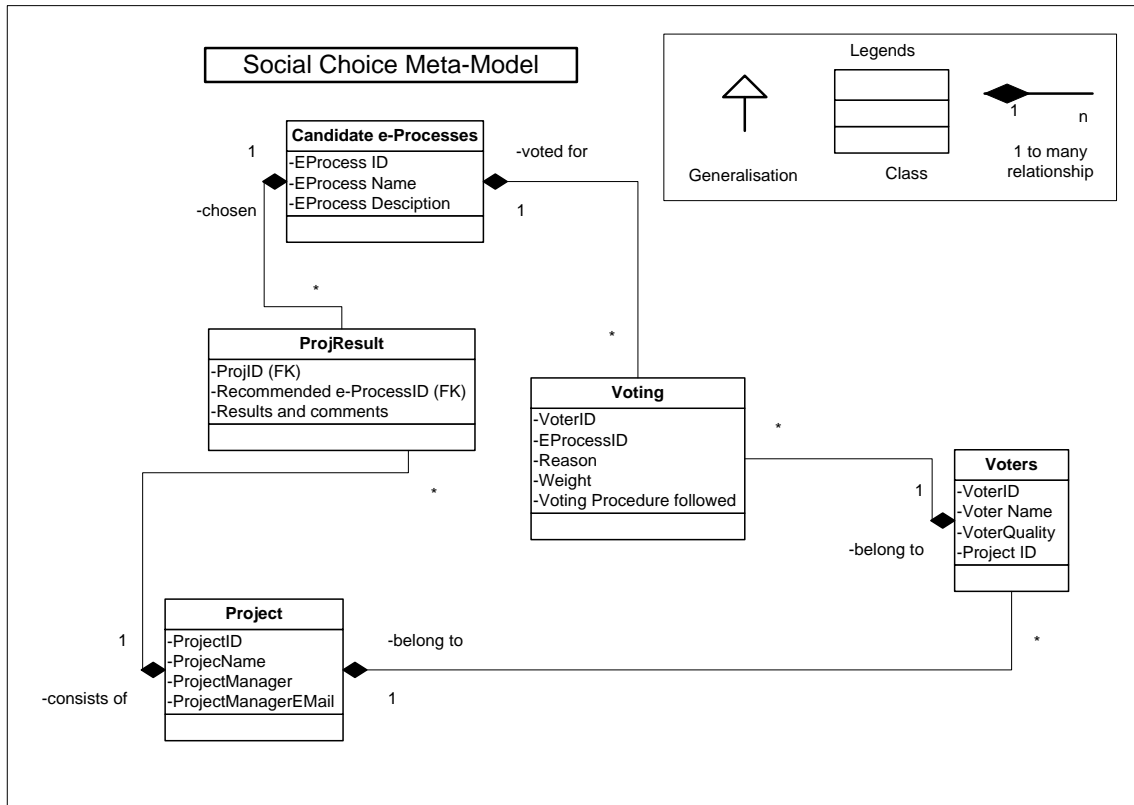


Figure 7.11: e-Process Selection SCM Meta-Model

e-Process Selection SCM Meta-Model Data Dictionary		
Name	Description	Type
Candidate e-Processes	The e-Processes being taken into consideration.	Class
Voting	A list of all votes obtained.	Class
Voters	Detail about all voters participating in the process.	Class
ProjAnalysisResults	The result of the selection process.	Class
Project	Detail about the project for which an e-Process is required.	Class

Table 7.9: e-Process Selection SCM Meta-Model Data Dictionary

The developers will be asked to vote on each of the high level aspects used for evaluating the environment as well as which of the e-Processes they prefer to use for the development process. This information, is then evaluated in order to determine which one will be best suited. In Figure 7.11 and Table 7.9, the meta-model aspects for Social Choice method is presented. In part C, SCM is applied to a case study.

7.6 Integrated Meta-Model

In Figure 7.12 and Table 7.10, the meta-model aspects, for a combined view of all four decision models are presented.

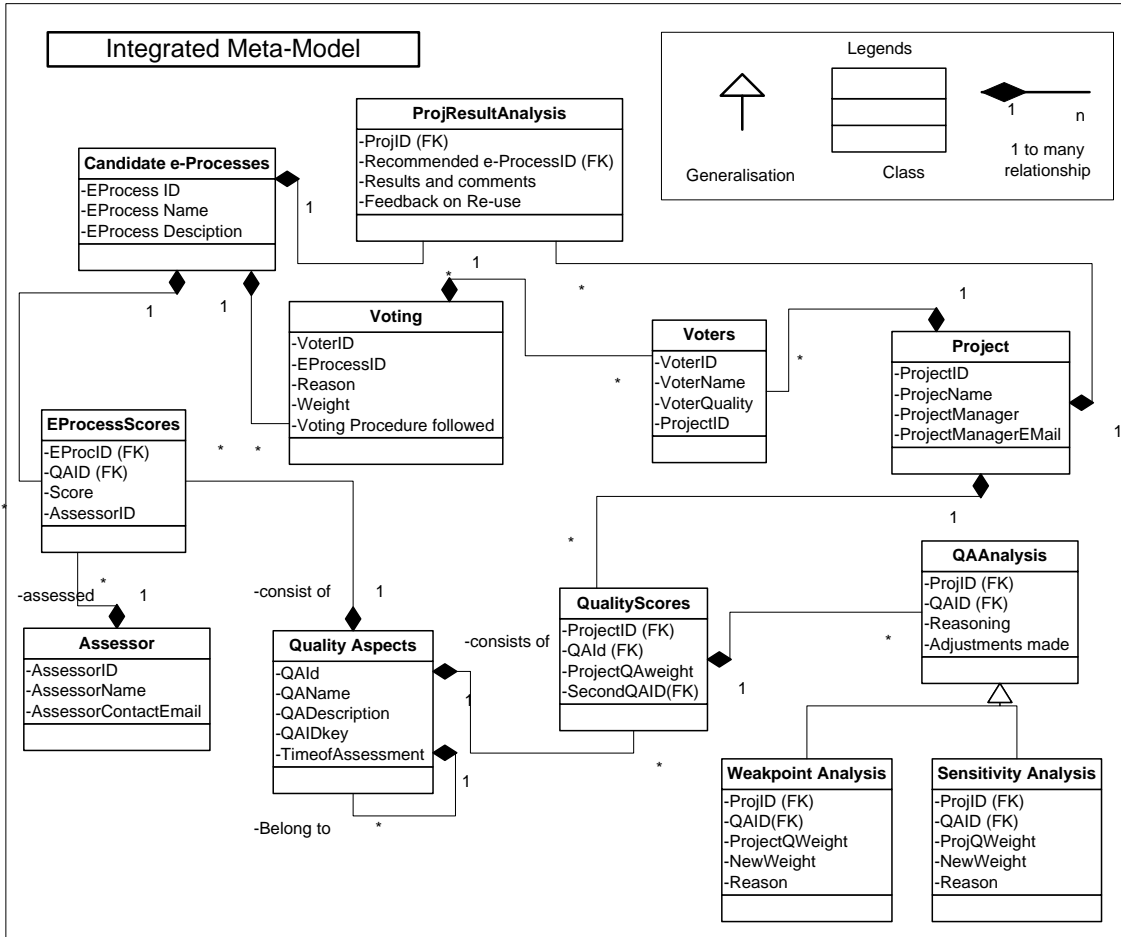


Figure 7.12: Integrated e-Process Selection Meta-Model

When applying any of these decision methods the resulting meta-model should be able to accommodate all of these. In part D, the prototype model uses this meta-model to define the data base schema required.

7.7 Conclusion

This chapter described the concepts required for the research contained in the thesis. The decision making methods described in this chapter forms the basis of the methodology used in this study. When a new eCIS is required, each of the decision making methods can potentially be used for e-Process selection.

The next chapter introduces a usage recommendation for the decision making methods and then each of these methods are applied and validated in chapter 9 of the thesis. In Part D of the thesis, the integrated e-Process Selection Meta-Model developed here, forms the basis for the prototype tool.

Integrated e-Process Selection Meta-Model Data Dictionary		
Name	Description	Type
Candidate e-Processes	The e-Processes being taken into consideration.	Class
eProcessScores	The score for each e-Process in terms of each quality aspect as estimated by experts.	Class
Assessor	The expert of a specific e-Process' estimating eProcessScores.	Class
Quality Aspects	A list of all quality aspects including the high level aspects and the second level quality aspects.	Class
Quality Scores	The weight assigned to a specific quality aspect for a specific score.	Class
ProjResultsAnalysis	The result of the selection process.	Class
Project	Detail about the project for which an e-Process is required.	Class
QAAAnalysis	Capturing of reasons for adjustments made to results.	Class
Sensitivity Analysis	The results obtained applying WA.	Class
Weakpoint Analysis	The results obtained applying WA.	Class
Voting	A list of all votes obtained.	Class
Voters	Detail about all voters participating in the process.	Class

Table 7.10: Integrated e-Process Selection Meta-Model Data dictionary

Chapter 8

Decision Making Methodology

Recommended Usage

This chapter evaluates the four decision methods presented in this thesis in order to provide guidelines to decide which will be the most suitable for use. The four methods are the Value Benefit Analysis Method, the Analytic Hierarchy Analysis method, the Case Based Reasoning method and a Social Choice Method.

8.1 Introduction

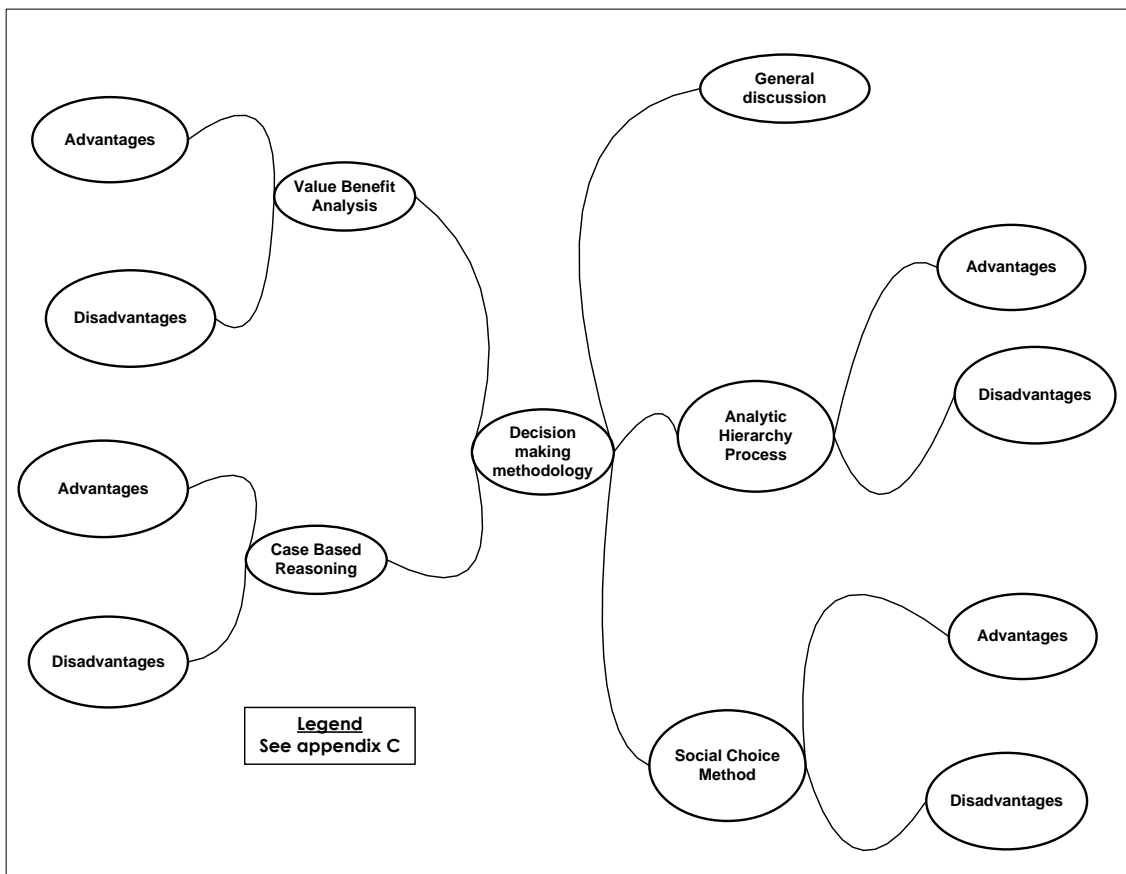


Figure 8.1: Decision Making Methodology Framework

Good decision making is an essential skill for deciding which development method to use for developing an eCIS. If the developer can learn how to make timely, well-reasoned and well-considered decisions then the resulting eCIS has a better chance of success. However, if the developer were to make a poor decision when selecting an e-Process, then the whole eCIS runs the risk of failure.

As seen in Figure 8.1, on the previous page, the discussion in this chapter will cover the decision making models that have been presented that could be used for selecting a specific e-Process, and then provide the documented motivation for the selection of that specific e-Process. This thesis aims to help the developer map out the likely consequences of making specific decisions by taking a number of quality factors into account and using a specific decision model. This chapter will provide some support in deciding which of these decision models to use for which specific scenarios. Sometimes it can be obvious which of the four decision methods are most suited and sometimes it is not a clear-cut decision.

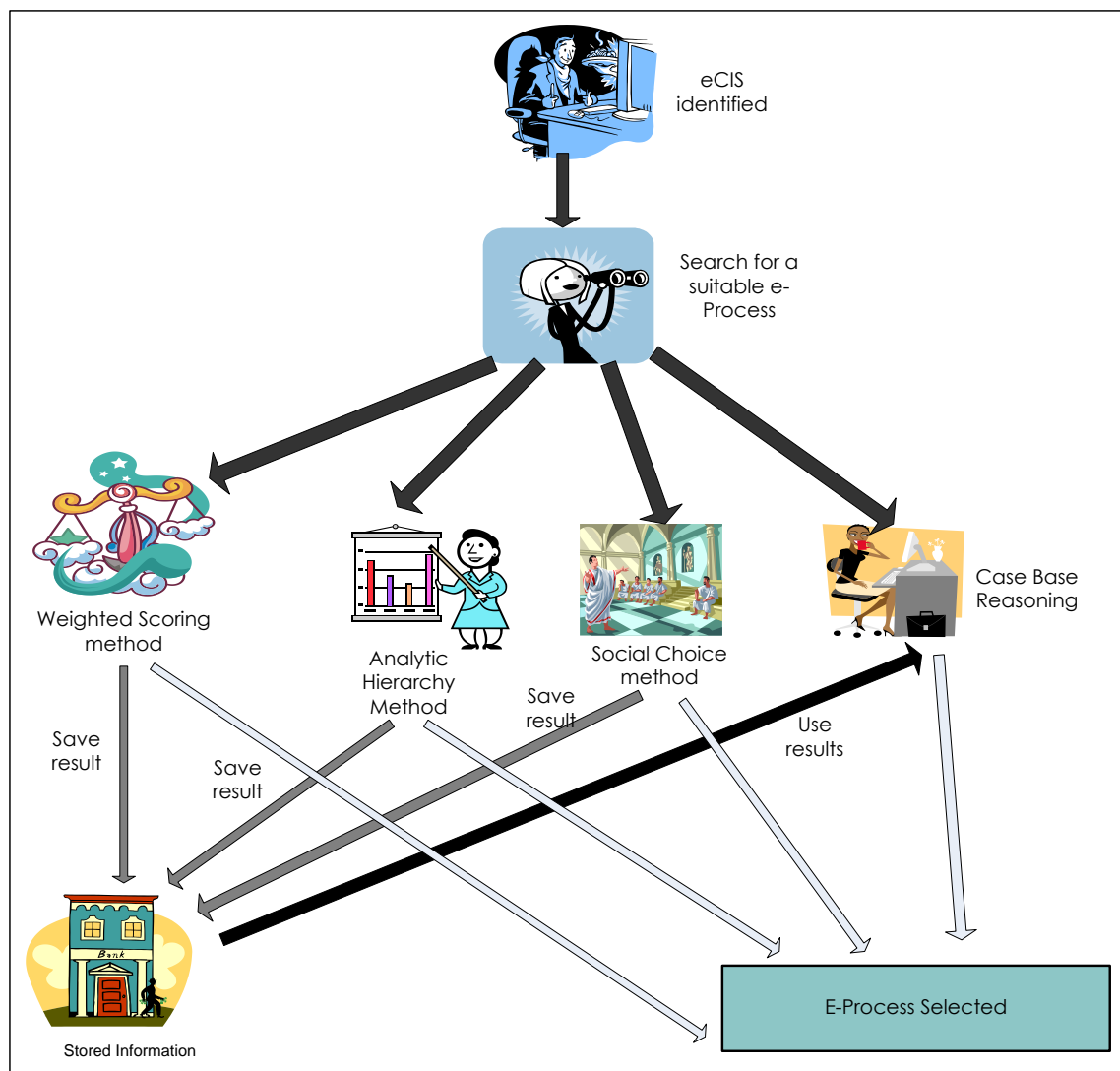


Figure 8.2: Decision making and decision models

In Figure 8.2 the four decision models used for decision making are shown. The process starts when a new eCIS is required and a suitable e-Process is required. This chapter will

provide basic support to decide which of the four decision making methods to use in which instances (see chapter 9 for the practical observations).

When a decision making method has to be selected, a function f can be applied to the set of decision making alternatives $\{A_1, \dots, A_n\}$ in order to determine True or False for each of these. This function will be a combination of decision making quality aspects and the purpose of the decision making method. The decision making quality aspects will include the following:

- Cost to execute or use this specific decision making method (Macdonald & Cheng, 1997).
- The main purpose of this decision method (Harris, 2009).
- Dependability of the decision making method (Harris, 2009).
- The speed with which a result can or needs to be obtained (Harris, 2009).
- Workflows of the decision making method (Doumpos & Zopounidis, 2002).
- Understandability of the decision making method (Serrano, Calero, Sahraoui & Piattini, 2008)

The next section will provide the different aspects that need to be taken into account when deciding which decision making method to use.

8.2 Decision model usage comparison

There are a large number of different decision models that are available. In this thesis four of these are used to illustrate support provided to decision makers when selecting suitable e-Processes for a specific eCIS. In the next four sections each of these decision models will be briefly described in order to explain when a particular decision model will be preferable.

8.2.1 Value Benefit Analysis

This method, like any other method, has its limitations and is not optimal in all situations. The weighted scoring system is used in this thesis in order to evaluate and quantify a wide range of quality criteria in order to select a suitable e-process for eCIS development. The individual 'evaluation criteria' are scored and weighted to determine an overall 'winner' score (Clemens, 2002; Mantel, et al., 2005).

Applying Value Benefit Analysis (VBA) is relatively easy. If there is a definite “winning e-Process” this method is ideal to use. The potential developers are involved in this process and are required to help with the evaluation process.

VBA does not focus on the monetary side of the decision in order to reach a conclusion and it does not take into consideration the impact of the different weights selected and should be combined with a sensitivity analysis to determine the impact of the different values selected for the different quality criteria used (Mantel, et al., 2005).

This method is similar to the Analytic Hierarchy Process except that you can take more criteria into account for use in the evaluation process. The calculations are however more basic and do not cross-reference the criteria to the same extent that the Analytic Hierarchy Process does.

Based on the quality aspects for decision making, identified earlier in the chapter, and the characteristics of VBA (Clemens, 2002; Mantel, et al., 2005), it can be stated that VBA:

- Is relatively cost effective to execute;
- Is well suited for the purpose of selecting an e-Process;
- Is dependable if the developers responsible for specifying the weights for each quality aspect of the eCIS know their environment well. This process may take a long time and slow down the selection process;
- Has a well-defined workflow process which is easy to follow and;
- Is easy to understand and execute.

The VBA method is recommended for use when the developers know the development environment and have no specific preference of a specific e-Process to use. This method will also be the preferred method above the Case Based Reasoning method if this specific eCIS is substantially different from any of those stored in the knowledge base of historical cases (see chapter 9 for further discussion of this method).

8.2.2 Analytic Hierarchy Process

The Analytic Hierarchy Process (AHP) is used to evaluate both the subjective and the objective evaluation measures. This process reduces the decision making bias of the developers by allowing the process to check the consistency of both the evaluation quality aspects and the alternative e-Processes. The AHP is based on the idea of decomposing a decision making problem in such a way that the multiple quality

aspects used as decision criteria can be organised into a hierarchy of criteria (Jayaswal & Patton, 2006).

Using the quality aspects for decision making, identified earlier in the chapter, and the characteristics of AHP (Jayaswal & Patton, 2006), it can be stated that AHP:

- Is relatively cost effective to execute if the number of quality aspects used for evaluation is limited and a tool is available;
- Is suitable for the purpose of selecting an e-Process. Once again, the method is dependable if the developers responsible for specifying the weights for each quality aspect of the eCIS know the scenario well;
- Can take a long time, especially if there are a large number of quality aspects to evaluate or many e-Process alternatives to select from. With a suitable tool this process can be automated and provide an answer quickly;
- Is well-defined in terms of its workflow process and;
- Is not very easy to understand the logic, but using automated support (AHP software applications) is easy to execute.

Additional advantages of using this method include:

- The main benefit of the AHP is that it provides a systematic approach that has been validated.
- Using this formal decision method minimises problems that might occur, such as defective planning, lack of focus of the development team, or excessive expenses that might impact on how the decision team decides on an e-Process to use.
- AHP should be the preferred decision method in those cases where the decision team want to limit the number of criteria (quality aspects) used for the decision process. AHP is such that all the aspects are compared to all the other aspects and using too many quality aspects with this decision method will not be feasible.
- AHP allows the developer to choose the level that they want to consolidate on without losing any important information in the process. It is, however, often necessary to group some quality aspects together in order to reduce the amount of work that the developers are asked to do. This will ensure that there is not too much information that is too detailed and not well structured.

This decision making method is discussed further in chapter 9 where it is applied to a case study.

8.2.3 Case-Base Reasoning

Re-use is considered to be an important solution to many of the problems that are encountered in software development. One of the ways to incorporate re-use in this decision process is by keeping track of historical information and using this captured knowledge when making a choice (Mansar, et al., 2003; Watson & Marir, 1994).

When the eCIS required, is quite a standard system then it will be preferable to check whether there is a eCIS with a selected e-Process that is similar to this one stored in the knowledge base. With a large number of previous historical cases stored there is a good chance that similar decisions have been required previously. In this case it is worthwhile to determine how similar the two eCIS are and then to use the same solution as the historical case for the new case (Mansar, et al., 2003; Watson & Marir, 1994).

Evaluating the quality aspects for decision making, identified earlier in the chapter, and the characteristics of CBR (Cunningham & Bonzano, 1999; Leake, 1996; Mansar, et al., 2003), it can be stated that CBR:

- Can be expensive if a large knowledge base needs to be maintained. It is, however, relatively cost effective to use as not much input is required from the development team;
- Is dependable if the information in the knowledge base is extensive and complete and can thus provide an extensive list of possible previous solutions to find a best match;
- Is relatively quick to execute with the right software support and is well-defined and;
- Is relatively easy to understand and execute in terms of its workflow process.

Additional advantages are that this method is very easy to execute if the new eCIS has a number of similar eCIS in the knowledge base and it is easy to use for standard types of eCIS. CBR is applied to a case study in chapter 9 and the application discussed.

8.2.4 A Social Choice Method

Voting has been used as a selection method for a very long time. One person, one vote is well-known voting method, but as discussed in previous chapters there are other methods such as the Borda count and Condorcet methods (Pomerol & Barba-Romero,

2000). When developers use the social choice method, it is required that they make their choice from the actual feasible set of e-Process alternatives given the actual preference profile. Their choices have to be justified by sound reasoning and actual preferences.

This method will be used when there are developers in the team that can explain the use of the different e-Processes and the specific eCIS is relative easy to develop.

Once again evaluating the quality aspects for decision making, earlier in the chapter, and the characteristics of SCM (Bernroeder & Mitlohner, 2007; Pomerol & Barba-Romero, 2000), it can be stated for SCM:

- The cost of using Social Choice Methods is inexpensive and has an added benefit as the development team has the opportunity to work together and discuss some of the eCIS development issues.
- This method is as dependable as the team partaking in this method. If these people are knowledgeable then their decisions or votes will most probably be of a higher quality.
- Social Choice Methods have been used extensively and are well-defined. This method is very easy to use and apply.

This method is applied to a case study in chapter 9 and further conclusions are discussed there.

8.3 Conclusion

The objective of the selection process is to evaluate the set of selected e-Processes by applying the evaluation aspects and then being able to document the reasons for selecting a specific e-Process.

As there are a large number of quality aspects being used for evaluation it is important that the weights be assigned to each of these aspects. Both the Value Benefit Analysis method and the Analytic Hierarchy Process method, focus on the developers being able to actually assign weights to these quality aspects. In the case of the AHP method these developers should also be able to decide not to use specific quality aspects.

Developers could use any of the four suggested decision models when selecting e-Processes. There are, however, for each new eCIS to be developed, indicators that will make one of the decision models more preferable than another.

PART C

APPLYING DECISION MAKING METHODS

Chapter 9

APPLYING THE DIFFERENT DECISION MAKING METHODS TO E-PROCESS SELECTION

This chapter applies e-Process selection using value benefit analysis; analytic hierarchy process; case based reasoning and social choice method on a series of cases to demonstrate applicability of each of these approaches.

9.1 Introduction

During the development of the research for the thesis, four different case studies were used to trial the different decision making methods. These case studies were available because of eCIS that were being developed in the Hawkes Bay area in New Zealand and the author of the thesis had access to the developers and the development process. Most of the development taking place in Hawkes Bay is quite small scale, therefore the author did not have access to larger projects in order to trial the decision making methods on larger applications.

From a validation perspective it would have been better to apply each of the decision making methods to the same case study. This was however, not possible, as the research was developed over 6 years. The application to each of the sample problems occurred during the development for the specific decision making method that was being researched. The validation of these methods was done by discussing and evaluating the results with the development team as well as applying, where possible, the case study to an alternative decision making method or applying sensitivity analysis.

For the first case study used for the application of the VBA decision making method, further validation was carried out by applying sensitivity analysis, weak-point analysis and applying the CBR decision making method using the data obtained during the application of the VBA method as well as interviewing the developers.

For the second case study used for the application of the AHP decision making method, further validation was done by applying sensitivity analysis to the results obtained as well as interviewing the developers.

For the third case study, the application of the CBR decision making method, enough data was obtained during the initial application of CBR to the case study, to also apply

the VBA decision making method to this case study for further validation. The developers were also interviewed in this case.

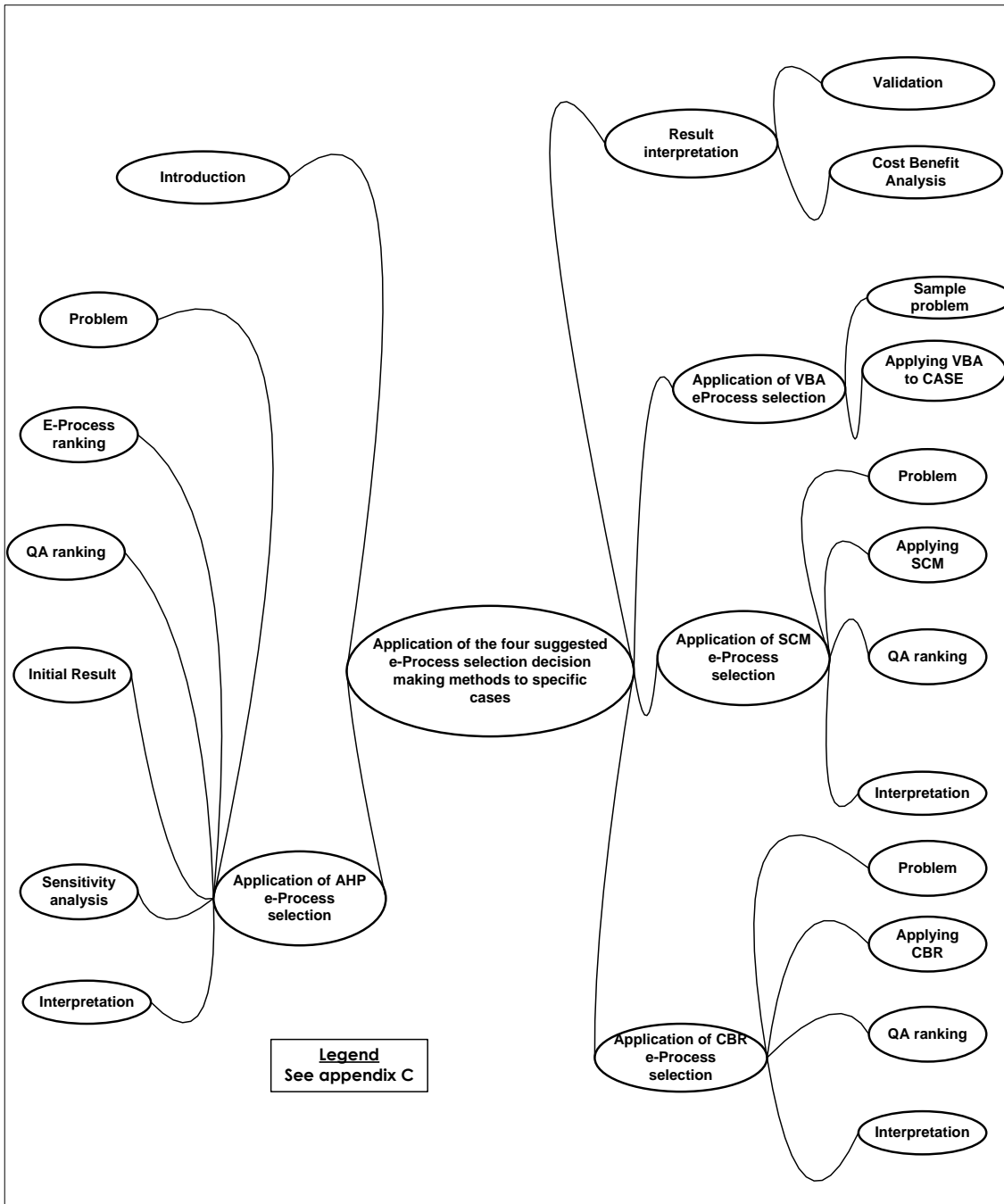


Figure 9.1 Applying the different decision making methods to e-Process selection

For the case study used to apply the SCM decision making method, the data required and gathered at the time was not sufficient to apply any of the other decision making methods, but further cost-benefit analysis was done on the case study and the developers interviewed.

As seen in Figure 9.1, this chapter applies each decision making method to a different case scenario and then validates each of these methods.

9.2 Applying e-Process Selection to Value Benefit Analysis

This section is the first of the four sections where the different decision making methods used for e-Process Selection are applied. This section focuses on the Value Benefit Analysis method. The research contained in this section is based on parts of the following publications; see Albertyn and Kaschek (2005c), Albertyn (2005d) and Albertyn and Kaschek (2004b).

Before VBA was applied to the case study it was necessary to approach experts on each of the e-Processes in order to capture their expertise on each of the e-Processes. These experts are also called assessors in this thesis – refer to section 7.2.5. Six different people with expertise in specific e-Processes were approached in 2003/4 in order to capture their expertise.

The assessors, each an expert on a specific e-Process or in two cases on more than one of the e-Processes, identified the scores for the different e-Processes used in the thesis and their associated quality aspects. This information was captured and is used in a semi-permanent capacity in the e-Process Selection Methodology.

During the research period this captured information was further refined a few times when more input was received from additional experts. See table 9.1, the last four columns, for examples of the numbers obtained from the experts (assessors).

9.2.1 The sample problem (Club eCIS)

In 2004, the author of the thesis was allowed access to the developers and development of a new eCIS for a local Athletic club. The local Athletic Club had contracted one of the local software development companies to develop an eCIS. At that stage the author of the thesis had contact with this development company.

Description of the eCIS:

The eCIS to be developed had to allow new club members to join the club online and allow existing members to edit their information. The management of the club wanted to be able to do maintenance on and processing of the data when required.

The eCIS had to allow the members to register for the different local events and associated sub-events as well as allow the members to order the t-shirts and other paraphernalia associated with each of these events. The eCIS also had to provide links to other athletic events and clubs.

The eCIS had to contain an event calendar, descriptions of the different events as well as course descriptions. The club's training schedule and plans had to be available to the members. The member had to be able to look at times achieved for time trials in their training sessions. The club wanted to be able to use this eCIS as a communications venue – to let the members know of events that are occurring. Reunions will also be organised from time to time and the club wants to keep track of their “old” members. The development company planned to eventually expand and sell this eCIS to other athletic clubs nationwide if the eCIS was of a high quality.

The member information and event as well as sub-event information was at that point stored in a spreadsheet. The selling of the goods was not tracked extensively, but as this part of the business was expanding the idea was to keep better track of this information with the new eCIS.

The Developers

The software development company had three developers of which two were very experienced developers and one graduate with flair for C# and ASP programming. The development was taking place in a small office environment. This company had a number of contracts that they were working on and did not plan to focus on this eCIS exclusively. The company wanted to start using a more formalised development environment and had been investigating the possibility of using the Rational Unified Process.

The initial discussions about the methodology

The author of the thesis discussed the methodology with each of the developers. None of the three people had any strong preferences about which methodology to use. Their comments included remarks such as: “Scribbles on paper is fine by me”, “Rational is neat – used it at varsity” and the manager's comment about “have started to look at RUP – can you help us with some input about what is the best to use”.

9.2.2 Applying VBA

Just after the initial meeting with the Athletic Club Management, the author met with the development team to capture their input for the initial exercise to test the e-Process Selection Theory using VBA. This meeting took place at the developers' offices. At that stage the prototype tool was not available but the theory applied to VBA had been developed. The quality aspects were also available with the values assigned to each of the development processes – see discussion on assessors in section 9.2.

The author of the thesis spent three hours with two of the developers discussing their input for the quality aspects. The meaning of each of the high level quality aspects was first explained and the developers input for the eCIS discussed and after the two developers agreed – it was written down. Next the low level quality aspects were first explained and then their values were also captured.

Capturing the information was a straight-forward exercise and did not take much time. It was easy to get agreement from the developers on the values to be used. This e-Process Selection method is quite easy to apply and use.

The results of this discussion are captured in Table 9.1, on the next page. The initial values for each of the quality aspects were captured as a value out of 10 and then normalised.

The author then applied the theory to the values and the following calculation was done on these values:

e-Process (n) =

$$\sum (\text{normalised HLQA} * \text{normalised QA} * \text{normalised e-Process QA}(n)) * 10000$$

which produced the following results:

$$\text{RUP} = 51.891$$

$$\text{AM(XP)} = 43.638$$

$$\text{OSS} = 41.013$$

$$\text{SBUP} = 44.218$$

The winner for this particular application was thus the Rational Unified Process.

High Level Quality Aspects	Quality Aspect(QA)	Value (1-10)	Weight of QA	RUP	AM(XP)	OSS	SBUP
e-Process aspects 0.20	Completeness	4	0.030	0.046	0.031	0.019	0.026
	Understandability	8	0.060	0.039	0.037	0.025	0.032
	Visibility	5	0.037	0.059	0.025	0.031	0.039
	Supportability	2	0.015	0.053	0.043	0.031	0.026
	Maintainability	3	0.022	0.053	0.037	0.031	0.026
Quality concepts 0.30	Readability	4	0.030	0.026	0.031	0.037	0.026
	Reliability	9	0.067	0.046	0.043	0.031	0.039
	Robustness	7	0.052	0.046	0.031	0.031	0.032
Cost 0.10	Development Budget	9	0.067	0.013	0.025	0.037	0.032
	Running Costs	4	0.030	0.007	0.025	0.037	0.032
Domain impact 0.04	Infrastructure	2	0.015	0.033	0.025	0.031	0.026
	Enterprise Culture	5	0.037	0.026	0.049	0.049	0.058
	Technology	4	0.030	0.039	0.031	0.043	0.039
	Geographic Interaction	3	0.022	0.033	0.043	0.037	0.032
	IT Strategy	7	0.052	0.020	0.025	0.031	0.032
	Business Strategy	6	0.045	0.013	0.019	0.037	0.032
	Team Experience	5	0.037	0.013	0.049	0.056	0.032
	Domain Knowledge	3	0.022	0.020	0.037	0.043	0.039
	E-Process knowledge	4	0.030	0.026	0.037	0.031	0.026
	Development Time	6	0.045	0.013	0.049	0.043	0.045
Usability 0.14	Functionality	5	0.037	0.026	0.031	0.025	0.032
	Manageability	4	0.030	0.020	0.031	0.031	0.039
	Quality assurance	3	0.022	0.046	0.031	0.031	0.039
	Adjustability	2	0.015	0.053	0.049	0.043	0.039
Compatibility 0.03	Exchangeability	6	0.045	0.046	0.037	0.037	0.045
	Map ability	5	0.037	0.039	0.031	0.037	0.026
Maturity 0.19	Stability	4	0.030	0.053	0.031	0.025	0.039
	Tool support	3	0.022	0.046	0.025	0.031	0.032
	Documentation	2	0.015	0.046	0.043	0.031	0.039

Table 9.1: Definition of weights and alternative values for the quality aspects

This result was further explored by applying first **sensitivity analysis** with one example providing results such as:

$$\begin{aligned} \text{RUP} &= 50.2 \\ \text{AM(XP)} &= 43.01 \\ \text{OSS} &= 40.02 \\ \text{SBUP} &= 43.12 \end{aligned}$$

These results were not significantly different from the first results and thus the first results were accepted as true. Next **weak-point analysis** was done. This exercise also did not result in significant changes to the results.

The software development company decided to use the Rational Unified Process in the development as they wanted to use this process in future development of eCIS. The results are discussed with the developers in section 9.6.2 where further validation of this VBA decision making method is also explored.

The case study used for VBA was not specifically selected to be used with VBA but turned out to be well suited for a VBA application. Two of the developers were experienced developers and were able to provide valid and dependable input for the quality aspects (see chapter 8 for usage recommendations). They found it easy to use and follow.

9.3 Applying e-Process selection using the AHP

The Quilting eCIS case study, discussed below, is used to demonstrate the applicability of the AHP approach to e-Process selection. The research discussed in this section follows parts of the research as published in Kaschek, Albertyn, Shekhovtsov and Zlatkin (2007).

Extending the discussion in section 7.3, the method presented in section 7.3 has been applied to the example problem. The reason for doing so is to investigate how the analytic hierarchy process can be applied in a "real" scenario and what kinds of problems can occur in such a case scenario.

9.3.1 The sample problem (Quilting eCIS)

In 2006, the author of the thesis was involved with the development of an eCIS for a small Napier-based quilting company in New Zealand. This allowed her access to the developers and the development of the eCIS.

Description of the eCIS

The Quilting Company has asked a small software vendor to set up an e-Commerce site, as they lack any IT-knowledge. Quilting is a needlework process in which layers of material are attached to each other with continuous stitches, either by hand or with specialist machines to make a quilt. This quilting company sells quilting machines, patterns as well as finished products. There is a number of heavy duty quilting machines in the shop – new and those used in their quilting work. The company can also order machines for those people who want to set up their own quilting company. There are also a large number of patterns for sale. The new eCIS will need to manage the sale of patterns, quilts – either finished products or special orders and as the company has the New Zealand contract to sell these quilting machines – be able to manage the sale of these machines. The company also wanted to be able to manage their client base and to be able to send out marketing material when required.

The Developers

The software vendor, contracted for the eCIS development, had used the Rational Unified Process in the development of previous eCIS, had some experience in storyboarding and user profiling and some of their previous eCIS had been developed using Agile methods.

At that time the company had been in IT development for three years, had two full-time staff and used part-time staff when required.

The software vendor often used final year Information Technology students to support their development. Data was gathered from the software vendor developers by interviewing them concerning the relevant quality aspects.

The initial discussions about the methodology

The author of the thesis interviewed the two full-time staff and also asked the student for her views and input. One of the full-time staff members had extensive experience in the use of RUP and said “I would be very glad if this experiment shows that RUP is the recommended process”. The other full-time staff member did not have any specific preferences, but was quite willing to trial the e-Process Selection Methodology and said – “at least the decision making has been taken out of our hands”. The student did not have any specific thoughts, but as she had been using storyboarding in her courses she said “I hope it comes out with a recommendation for storyboarding and user profiling”.

9.3.2 Applying AHP

The author met with the development team to capture the input required to apply AHP. This meeting took place in the offices of the software vendor. Working through the process took about 4 hours and 3 developers were involved. A further 2 hours was spent by the author of the thesis to process the data on the software.

E-Process ranking

The high level quality aspects as well as the second level aspects, as discussed in chapter 4, were applied in this case study. The e-Process alternatives, as specified in chapter five, were used in the selection process.

Firstly, these e-Process alternatives were evaluated against each second-level quality aspect by the experts. Table 9.2 shows a subset of the assessments (expert-supplied comparison values $a(q,1, x, y)$. The normalised assessments $f_{q,x}$ is shown in the table).

The e-Processes used are all well known and were acceptable to the software vendor in this case study. No explicit expert selection was carried out as it was determined that the members of the developing team had enough knowledge to be able to come up with a set of the expert values.

System quality aspect ranking

In the interview, the eCIS developers scored the importance of the quality aspects to the problem. Using Hipre 3.3 (see Hämäläinen, R.P. (2008)) the following data were captured:

- Table 9.3 shows all the high level quality aspects and how each of these is ranked against the other high level quality aspects.
- Figure 9.2 provides an example of second level ratings. It shows, as an example, one of the processes, one of the high level quality aspects and the ranking of each of its second level quality aspects against all the others.

Quality Aspect	e-Process assessment			
	RUP	OSS	AM(XP)	SBUP
Completeness	0.60	0.12	0.20	0.09
Understandability	0.17	0.50	0.17	0.17
Visibility	0.58	0.12	0.19	0.12
Supportability	0.60	0.12	0.20	0.09
Maintainability	0.19	0.19	0.56	0.06
Readability	0.54	0.18	0.18	0.11
Reliability	0.19	0.19	0.56	0.06
Robustness	0.19	0.56	0.19	0.06
Development Budget	0.13	0.38	0.38	0.13
Running Costs	0.13	0.38	0.13	0.38
Infrastructure	0.40	0.13	0.40	0.08
Enterprise Culture	0.50	0.17	0.17	0.17
Technology	0.54	0.18	0.18	0.11
Geographic Interaction	0.10	0.50	0.30	0.10
IT Strategy	0.58	0.12	0.19	0.12
Business Strategy	0.54	0.18	0.18	0.11
Team Experience	0.40	0.13	0.40	0.08
Domain Knowledge	0.54	0.18	0.18	0.11
E-Process knowledge	0.19	0.03	0.58	0.19
Development Time	0.60	0.12	0.20	0.09
Functionality	0.60	0.12	0.20	0.09
Manageability	0.60	0.12	0.20	0.09
Quality assurance	0.55	0.18	0.18	0.08
Adjustability	0.60	0.12	0.20	0.09
Exchangeability	0.38	0.13	0.13	0.38
Mappability	0.10	0.30	0.30	0.30
Stability	0.54	0.11	0.18	0.18
Tool support	0.17	0.50	0.17	0.17
Documentation	0.40	0.14	0.40	0.06

Table 9.2: Selected experts' e-Process quality aspects ranking

	A	B	C	D	E	F	G
A	1.00	0.91	1.20	1.30	1.20	1.30	0.63
B	1.10	1.00	0.91	0.83	0.83	0.71	1.80
C	0.83	1.10	1.00	1.20	0.83	0.56	1.60
D	0.77	1.20	0.83	1.00	1.20	0.63	0.63
E	0.83	1.20	1.20	0.83	1.00	0.91	0.91
F	0.77	1.40	1.80	1.60	0.91	1.00	0.91
G	1.60	0.56	0.63	1.60	1.10	1.10	1.00
A=e-Process aspects B=Quality concepts C=Cost D=Domain impact E= Usability F=Compatibility and G=Maturity							

Table 9.3: High Level Quality Aspects Ranking

First recommendation

The final calculations were performed to incorporate the expert values, quality aspects hierarchy, and other data necessary for selection. The final e-Process scores are shown in Figure 9.3, on the next page, with RUP (0.3), OSS (2.8), SBUP (0.25) and AM(XP) (0.19).

As RUP is the e-Process with highest score it is the recommended alternative for the project. Although the example was illustrative, the result obtained here was discussed with the developers – see result discussion in section 9.6.3.

Conclusion

AHP has already been used for architecture selection (Zhu et al., 2005; Al-Naeem et al., 2005; Svahnberg et al., 2003). E-Process Selection is another application for applying AHP. Validation of this decision making method is done later in this chapter. The sensitivity analysis done on the data and the views obtained from the interviews conducted at the conclusion of the project are included in section 9.6.3.

The case study used for AHP was not specifically selected to be used with AHP. It did however turn out to be well suited for an AHP application but will most probably have been suited for a VBA application as well, as the eCIS was relatively easy to develop. Even using SCM will have been a good choice as the developers were quite experienced and the eCIS was small (see chapter 8 for usage recommendations). The developers found the process of providing the relevant input easy because of their previous experience but found the process tedious.

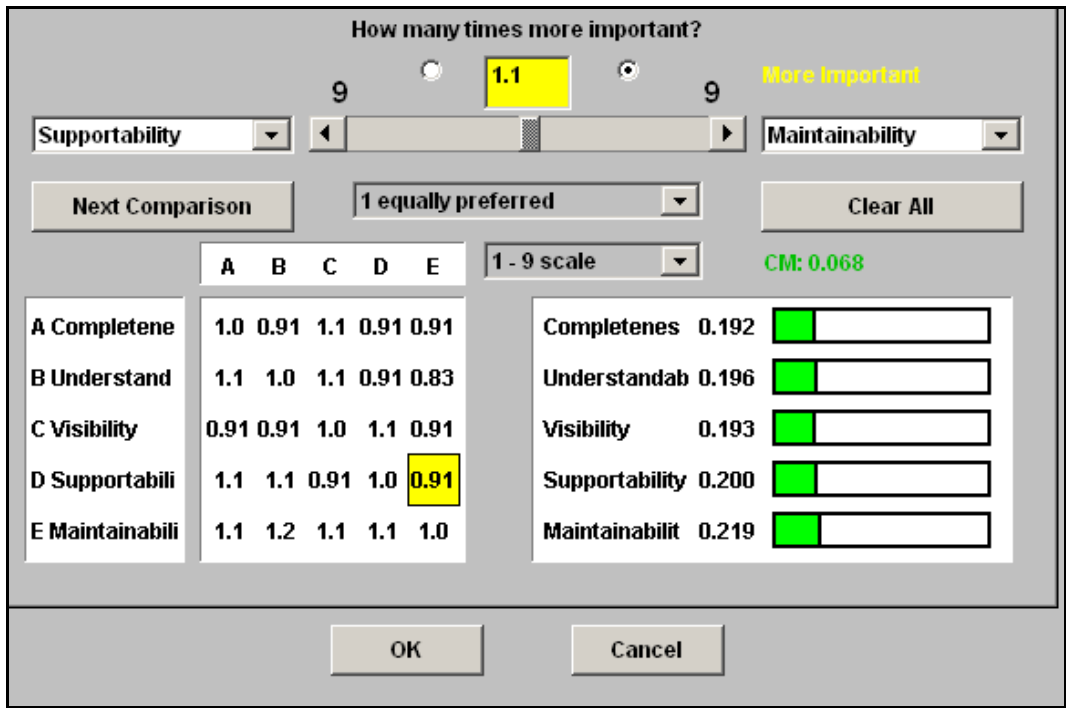


Figure 9.2: Second-level aspect scores for high level quality aspect: e-Process aspects

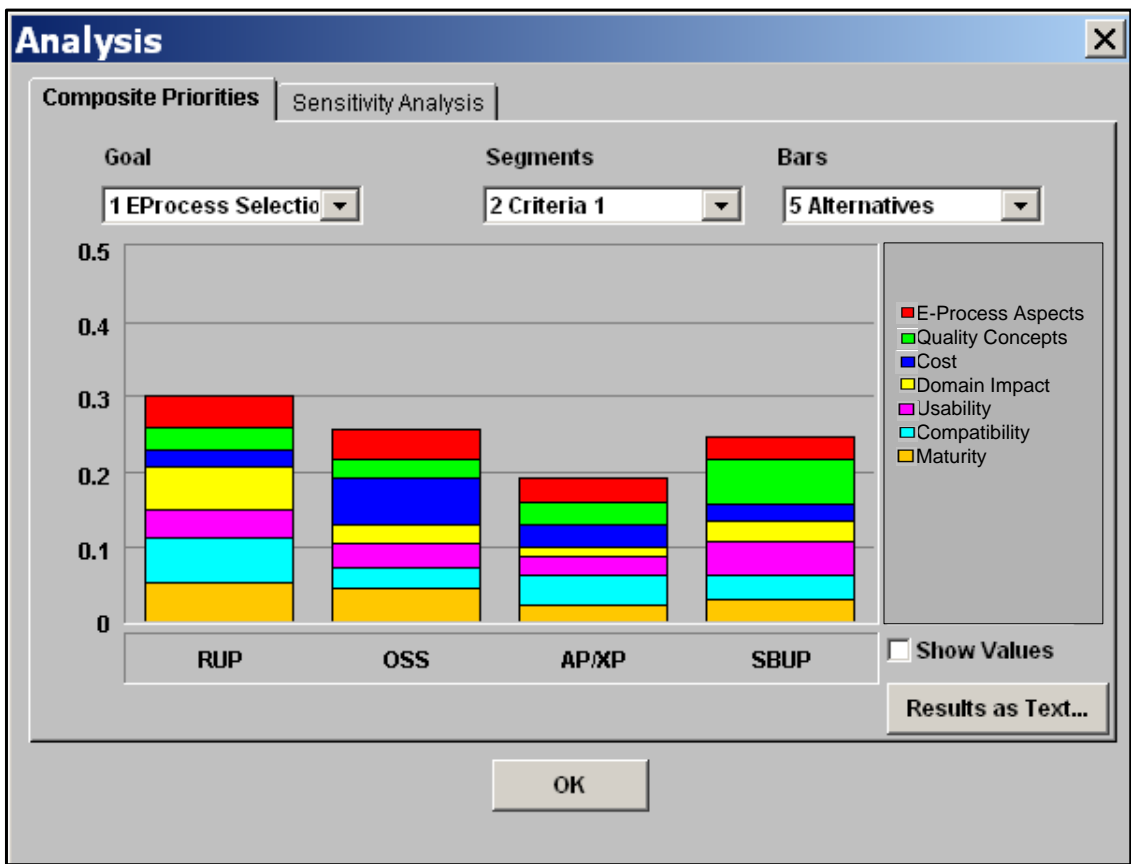


Figure 9.3: Final e-Process Scores and contributions of quality aspects

9.4 Applying e-Process Selection using CBR

This section applies Case Based Reasoning as a decision making method for selecting a suitable e-Processes to a case study. Aspects of the research published in this section were included in the following publications; see Albertyn and Kaschek (2007) and Albertyn (2007).

The heuristic, as discussed in section 7.4, is applied to a sample problem to demonstrate how it works. The aim is to determine a best-suited e-Process available for the development of the eCIS for the problem by using historical information and applying Case Base Reasoning. First the problem is briefly described and then the selection method is explained in more detail.

9.4.1 The sample problem (Packaging eCIS)

In 2006, a New Zealand packaging company asked their internal Information Systems department to set up an e-Commerce site. The author of the thesis was at that stage supervising a student doing a project with the company and was allowed access to the developers and some of the potential users of the eCIS.

Description of the eCIS

This New Zealand Company specialises in cardboard packaging. Their customers are based nationwide and are mostly large manufacturers. The company wanted the eCIS to allow their customers to place orders online. The orders were to be mostly business-to-business and much of the business takes place with small-to-medium sized fruit and vegetable growers.

The requirements stated that the product detail were to be available online. Invoicing was going to be done through the eCIS. Included in the eCIS were also user registration and user login; the facility to be able to update their detail online.

The product search options were to include: searching by existing product or being able to develop new options using existing material options.

The eCIS was required to have an online shopping cart allowing creating new products, adding new products to the list and deleting products from the order. Also included in the eCIS was the function to allow the customer to merge orders and change shopping carts as orders. It was also required in the eCIS specification to easily place an order

and then tracing the order - both on the customer side and the manufacturing business' side.

Some of the required options on the packaging company side included listing existing orders, tracking material in stock, ordering new material as required, listing information about those orders assigned to specific production lines, providing supervisors with information and producing line information, allowing for the updating of individual order status and being able to insert new packaging products.

The Developers

Their Information System department did at that stage run their own website and maintained the Information Systems required for production in the plant. The IT personnel consisted of 5 people, 3 of whom had development experience. Only one of the three developers was experienced in website/eCIS development; all three developers had been using development processes to support most of their development. All three developers would be involved in the eCIS development process. One student from the local tertiary Institute of Technology was also involved with the development process.

The Initial discussion about the methodology

The author of the thesis discussed the e-Process Selection Methodology with the developers as a group. Initially the developers were not very sure whether the e-Process Selection will work. They were very apprehensive that the exercise will take a long time and in their words "waste their time". They were however willing to become part of the process. What they appreciated about this first meeting was that they were able "to discuss the development process to be used" and "at least our preferences are being heard" were some of the comments made during this meeting. The group did however appreciate the consequences of choosing the wrong development process and the financial implications that this might have for the company.

9.4.2 Applying CBR

The author met with the development team to capture their initial input required for applying CBR. This meeting took place in the offices at the production plant. The information obtained from this meeting – after about 2 hours of discussion was applied on the cases available in the knowledge base.

As the case base was still relatively small, all the stored cases are evaluated for similarity. Only two cases were used to demonstrate the process. These cases from the Case Base

were the nearest similar to the problem at hand. In the first Case Base example the resulting e-Process recommended was the Rational Unified Process and in the second Case Base example the resulting e-Process recommended was Storyboarding and User Profiling .

In Table 9.4 the values provided by the developers of both the problem at hand, as well as the developer information for the two historical eCIS developments are shown. The Case Base included historical information on e-Process choices for eCIS using one of the four e-Processes, as defined in chapter 5. The high level quality aspects and second level quality aspects, discussed in chapter 4, were used. From the results in this table the “winning” e-Process could be determined for the problem case. The winner from this table was case 1 where the resulting e-Process recommended was the Rational Unified Process.

High Level Quality Aspect	Weight of quality aspect	Example of problem	Case Base example 1	Similarity value (Case 1)	Case Base example 2	Similarity value (Case 2)
	A	B	C	$A \times B-C $	D	$A \times B-D $
e-Process aspects	0.20	0.300	0.100	0.040	0.143	0.031
Quality concepts	0.22	0.048	0.148	0.022	0.213	0.036
Cost	0.3	0.150	0.250	0.030	0.251	0.030
Domain impact	0.1	0.143	0.040	0.010	0.204	0.006
Usability	0.04	0.238	0.338	0.004	0.123	0.005
Compatibility	0.05	0.048	0.050	0.000	0.022	0.001
Maturity	0.09	0.073	0.074	0.000	0.062	0.001
				0.106		0.110

Table 9.4: Comparing high level quality aspects for two cases with the problem

Although the table shows only 2 historical cases, the process allows for the whole Case Base to be investigated and then the winning 4 cases to be further analysed. These two cases could now be taken to the next step. As can be seen in Table 9.5, when a detailed analysis was executed on the second level quality aspects - case 1 was again the winner, but case 2 was not too far behind. For simplicity a weight of 1 was used for all low level quality aspects in this example.

Quality Aspect(QA)	Weight of QA	Example problem	Case Base example 1	Similarity (Case 1)	Case Base example 2	Similarity (Case 2)
Completeness	1	0.6	0.12	0.48	0.2	0.4
Understandability	1	0.17	0.5	0.33	0.3	0.13
Visibility	1	0.58	0.12	0.46	0.19	0.39
Supportability	1	0.6	0.12	0.48	0.2	0.4
Maintainability	1	0.19	0.19	0	0.86	0.67
Readability	1	0.54	0.18	0.36	0.18	0.36
Reliability	1	0.19	0.19	0	0.76	0.57
Robustness	1	0.19	0.56	0.37	0.19	0
Development Budget	1	0.13	0.38	0.25	0.38	0.25
Running Costs	1	0.13	0.38	0.25	0.13	0
Infrastructure	1	0.4	0.13	0.27	0.8	0.4
Enterprise Culture	1	0.5	0.17	0.33	0.17	0.33
Technology	1	0.54	0.18	0.36	0.08	0.46
Geographic Interaction	1	0.1	0.5	0.4	0.3	0.2
IT Strategy	1	0.58	0.12	0.46	0.19	0.39
Business Strategy	1	0.54	0.18	0.36	0.18	0.36
Team Experience	1	0.4	0.13	0.27	0.4	0
Domain Knowledge	1	0.54	0.18	0.36	0.18	0.36
E-Process knowledge	1	0.19	0.03	0.16	0.88	0.69
Development Time	1	0.6	0.12	0.48	0.2	0.4
Functionality	1	0.6	0.12	0.48	0.1	0.5
Manageability	1	0.6	0.12	0.48	0.2	0.4
Quality assurance	1	0.55	0.18	0.37	0.18	0.37
Adjustability	1	0.6	0.12	0.48	0.2	0.4
Exchangeability	1	0.38	0.13	0.25	0.13	0.25
Map ability	1	0.1	0.3	0.2	0.8	0.7
Stability	1	0.54	0.11	0.43	0.18	0.36
Tool support	1	0.17	0.5	0.33	0.17	0
Documentation	1	0.4	0.14	0.26	0.4	0
				9.71	9.74	

Table 9.5: Low level quality assessment results

The recommendation for the problem at hand was therefore to use the Rational Unified Process, but storyboarding and user profiling was a strong alternative. In a real problem scenario both these results (and even more) will be provided with the feedback on how the application went in order to recommend a best suited e-Process. More detail about the actual problem, before using the quality aspects will most probably be used.

In this section Case Based Reasoning was used for selecting a most suitable e-Process from a set of admissible e-Processes by using previous information in order to make the selection. However, as only a small number of case studies had been captured, further expansion of the knowledge base is required.

Section 9.6.4 will further discuss the feedback on the use of the e-Process Selection Methodology obtained from interviews with the developers as well as further validation by applying another DMM to this case study.

The case study was not chosen specifically for use with CBR. It did however turn out to be well suited as there were 2 eCIS in the case base that was similar enough to the case study to be suitable for comparison. The developers found it easy to provide the necessary input and seemed to have confidence in the outcome because other similar eCIS have used the same e-Process.

9.5 Applying e-Process selection using the social choice method

The social choice method is illustrated in this section by the use of an example case study. The theory discussed in chapter 7 is applied to this case study.

9.5.1 The sample problem (Kayak eCIS)

In 2006, the author of the thesis was involved with the development of a new eCIS for the Sea Kayak and Surf Boarding Shop. This allowed her access to the developers of the eCIS during this development period.

Description of the eCIS

The case study used here was the development of an eCIS for a Sea Kayak and Surf Boarding Shop. The shop wanted more exposure and wanted to develop better marketing strategies. The shop had a number of different products that could be sold over the internet. They wanted to use the eCIS both as a marketing tool as well as to allow their customers to place online orders. One of the products that they sold was a

specific brand of goods and this was linked to the worldwide distribution of this specific brand of surf board.

The eCIS had to mostly focus on this specific product. The final eCIS had to have the technical product description updates such as which type of epoxy was used, the use of different fiberglass types and many more technical aspects. The final eCIS development included videos of development processes used in the developing this surf board and some of the other product; videos of the use of some of the products and; testimonials from different satisfied customers. The eCIS also had to provide the information about alternative distributors as well as had to list other accessories that were available.

The Developers

A group of developers were asked to develop a new eCIS to promote this shop. There were two developers in the team. These two developers did not have much experience in developing eCIS, but both did have degrees in Computing. Two of the business owners were also skilled in eCIS development and were part of the development process.

The initial discussions about the e-Process Selection Methodology

The author of the thesis discussed the methodology with the two developers and had some informal discussions with the two managers. None of these people had any strong preferences about which methodology to use.

The managers had used a range of development methods before but they did not have a preference for any particular development method. The managers felt by discussing the options and then voting will be quite "democratic and fair".

The two developers had some knowledge of development processes through their studies, but had not used any development process extensively before. Their comments included remarks such as: "I don't know, what I don't know – but isn't it great to be able to discuss the issues and then have a say in the outcome", "yes we think the e-Process Selection Methodology can work" and "great to work together as a team".

9.5.2 Applying SCM

The Borda voting method, presented in section 7.5, was applied to an example problem. The author met with the development team to lead them through the voting process of applying SCM. This meeting took place in the Kayak shop. The managers,

who were quite knowledgeable about eCIS development, led the discussion on the 4 e-Processes and explained to the 2 developers the advantages and disadvantage of each e-Process. This discussion and voting process took 45 minutes.

When applying the social choice method the demands for producing and processing data were limited considerably in relation to some of the other methods discussed. There was no requirement to weight the attributes or quality aspects being taken into account and there was no requirement for value judgements.

Table 9.6 depicts the scores obtained when the developers were asked to vote on each of the quality aspects in terms of the e-Processes listed. The numbers were entered into an Excel spreadsheet. Each developer had a 4, 3, 2 and 1 to apply to each high level quality aspect which was summed in the excel spreadsheet. The table contains the final results.

The result obtained in the table showed the Agile and extreme programming eCIS development process as the resulting winner. The recommendation was thus to use this e-Process. The developers used this e-Process, but combined it with storyboarding and user profiling (SBUP) to develop the resulting eCIS.

E-Process	RUP	OSS	AM(XP)	SBUP
Criteria				
e-Process Aspects	5	3	7	5
Quality Aspects	8	2	4	6
Costs	2	4	6	8
Domain Impact	8	5	5	2
Usability	2	4	8	6
Compatibility	2	6	6	6
Maturity	8	4	6	2
TOTAL SCORES	35	28	42	35

Table 9.6: Scores obtained when applying Borda voting

The results obtained from applying SCM, were discussed with the developers after they completed their eCIS development. The consensus was that Agile and extreme programming will also have worked well for the development even though they elected to use storyboarding and user profiling. Even though some of the developers felt during the application of SCM that the Rational Unified Process was an option, this was not the case after they had finished their development. The developers felt that if

forced this development process would have had some benefit, it would not have been the best solution to their problem. Overall the developers agreed that the results from applying SCM gave a good indication of what they would have preferred to use at the time.

It could be concluded that the social choice – Borda method was easy to apply to any case scenario and minimal software support was required. As more significant case studies are applied it would be possible to improve the application of this decision method. It might even be possible to trial some of the other social choice voting methods in the future.

Kayak eCIS was not specifically selected to be used with SCM because of its characteristics, but was the case study available at the time of application. It was however quite suitable as the two managers were very knowledgeable and talked through the different e-Processes with the two less experienced developers before the voting process. The eCIS was also quite small and suitable for SCM application (see chapter 8 for usage recommendations).

9.6 Validation

A cross case analysis, see section 9.6.1, was done to ensure validity of the e-Process Selection Methodology. Sensitivity analysis was applied to ensure the validity of the results obtained from some of the decision making methods used. In section 9.7.1 a feasibility study on the e-Process Selection Methodology was executed. The interviews conducted were also included using rich text descriptions.

Using all of these methods allowed for methodological triangulation – see section 2.2, which is the use of more than one approach to gather data (Bensabat, et al., 1987; Yin, 1994). Applying a number of research approaches helped to ensure the credibility of the results.

9.6.1 Cross Case analysis

Eisenhardt (1989) states that cross-case analysis should be used to search for patterns. Eisenhardt (1989) believes that the researcher should be forced to go beyond the initial impressions using structured and diverse views to investigate the data. This will hopefully result in achieving an improved, accurate and reliable theory.

Eisenhardt (1989) suggested three different tactics. These are:

- 1) Select categories and look for within-group similarities while studying the intergroup differences,
- 2) Select pairs of cases and list the similarities and differences between each pair, and
- 3) Divide the data by data source to exploit "unique insights possible from different types of data collection" (Eisenhardt 1989, pp. 540-541).

The first tactic, as suggested above, was followed when the meta-model was developed in chapter 7 and used in the application on each of the case studies. The focus for the cross case analysis was however on tactic 2 which was followed by applying a further decision making model on the specific case study used for VBA and CBR and doing sensitivity analysis for AHP. The third tactic was applied by analysing the results and the feedback received from the developers.

By following these tactics, according to Eisenhardt(1989), the researcher is forced to analyse further and not just accept the first impressions. Following these tactics improve the likelihood of developing an accurate and reliable theory which fits the data.

9.6.2 Validation of VBA

Validation of VBA was done by mainly reviewing the results obtained with the development team as well as applying another DMM to the same case study to be able to compare the results obtained and to prove that the results were not just a match by coincidence.

Discussion of the results obtained from applying VBA to the Club eCIS case study:

The result was discussed with the developers towards the end of the eCIS development process. The team enjoyed using RUP and were quite keen to keep on using this process. As none of them had very strong feelings regarding the use of a particular development process they could not comment on whether it would have been better to use any of the other processes. The developers felt that storyboarding and user profiling could also have worked for the eCIS development process. They did however make comments like:

- "It is good not to have to choose."
- "We did not know where to start in the selection process."

- "I think we did a good job when we applied this process which could have been different if we used another development process – but I believe the future will tell".
- "At least we have a recommendation".
- "It was a neat exercise to talk about these things for a change".

The developers did not feel that they were in a position to comment on the outcome, but they were appreciative of the support provided when they had to make a selection.

Validation of VBA by applying a different DMM to the Club eCIS:

In order to further validate the VBA process used in section 9.2 above, the author now uses the Club eCIS case study with the same data, and applies the CBR decision making method. Only two cases are used to demonstrate this process as the case base is still quite small and for simplicity purposes. Table 9.7 shows only 2 historical cases, but the process allows for the whole Case Base to be investigated and then the winning 4 cases can be further analysed, refer to discussion in section 9.4 on CBR.

For Case Base example 1 the resulting e-Process recommended was the Rational Unified Process and for the Case Base example 2 the recommended e-Process was Story Boarding and User Profiling – see Table 9.7. The table shows the values provided by the developers for both the problem at hand and the two historical eCIS developments. This table also shows the weights that experts provided for each of the high level quality aspects. These two cases can now be taken to the next step. As can be seen in Table 9.8, case 1 is currently clearly the winner.

High Level Quality Aspect	Weight of quality aspect	Example problem	Case Base example 1	Similarity value (Case 1)	Case Base example 2	Similarity value (Case 2)
	A	B	C	$A \times B-C $	D	$A \times B-D $
e-Process aspects	0.16	0.20	0.21	0.0016	0.14	0.0096
Quality concepts	0.15	0.30	0.29	0.0015	0.21	0.0135
Cost	0.10	0.10	0.15	0.0050	0.25	0.015
Domain impact	0.34	0.04	0.04	0.0000	0.20	0.0544
Usability	0.10	0.14	0.10	0.0040	0.12	0.002
Compatibility	0.08	0.03	0.04	0.0008	0.02	0.0008
Maturity	0.07	0.19	0.17	0.0014	0.06	0.0091
				0.0143		0.1044

Table 9.7: Comparing high level quality aspects for two cases with the problem

Quality Aspect(QA)	Weight of QA	Example problem	Case Base example 1	Similarity (Case 1)	Case Base example 2	Similarity (Case 2)
Completeness	1	0.030	0.11	0.08	0.1	0.07
Understandability	1	0.060	0.51	0.45	0.18	0.12
Visibility	1	0.037	0.1	0.063	0.4	0.363
Supportability	1	0.015	0.12	0.105	0.6	0.585
Maintainability	1	0.022	0.17	0.148	0.4	0.378
Readability	1	0.030	0.18	0.15	0.18	0.15
Reliability	1	0.067	0.19	0.123	0.56	0.493
Robustness	1	0.052	0.56	0.508	0.19	0.138
Development Budget	1	0.067	0.38	0.313	0.38	0.313
Running Costs	1	0.030	0.38	0.35	0.13	0.1
Infrastructure	1	0.015	0.13	0.115	0.4	0.385
Enterprise Culture	1	0.037	0.17	0.133	0.17	0.133
Technology	1	0.030	0.18	0.15	0.18	0.15
Geographic Interaction	1	0.022	0.5	0.478	0.3	0.278
IT Strategy	1	0.052	0.12	0.068	0.19	0.138
Business Strategy	1	0.045	0.18	0.135	0.18	0.135
Team Experience	1	0.037	0.13	0.093	0.4	0.363
Domain Knowledge	1	0.022	0.18	0.158	0.18	0.158
E-Process knowledge	1	0.030	0.03	0	0.58	0.55
Development Time	1	0.045	0.12	0.075	0.2	0.155
Functionality	1	0.037	0.12	0.083	0.2	0.163
Manageability	1	0.030	0.12	0.09	0.2	0.17
Quality assurance	1	0.022	0.18	0.158	0.18	0.158
Adjustability	1	0.015	0.12	0.105	0.2	0.185
Exchangeability	1	0.045	0.13	0.085	0.13	0.085
Map ability	1	0.037	0.3	0.263	0.3	0.263
Stability	1	0.030	0.11	0.08	0.18	0.15
Tool support	1	0.022	0.01	0.012	0.17	0.148
Documentation	1	0.015	0.14	0.125	0.4	0.385
4.696						6.862

Table 9.8: Low level quality assessment results

For simplicity a weight of 1 was used for all low level quality aspects in this example. This table also shows that when a detailed analysis was executed on the second level quality aspects example case 1 was again identified as the winner, with the lowest score showing that the problem scenario is the most similar to this case. This supports the result obtained from using VBA, where the Rational Unified Process was also indicated as the “winner”.

Using VBA was relatively inexpensive although it takes some time to work through the required quality aspects in comparison to CBR and SCM. The author of the thesis found from applying all the decision making methods and observation of the end results that applying VBA is cheaper than AHP and CBR to use. It was also found that, with the use of sensitivity analysis and weak-point analysis, the results seem to be reliable and dependable and the workflows are well defined. The feedback from the developers indicated that it was very easy to understand this method.

9.6.3 Validation of AHP

Validation of AHP was mainly done by applying sensitivity analysis to the results as well as reviewing the results obtained with the development team.

Applying sensitivity analysis

Continuing with the Quilting eCIS case study – sensitivity analysis is now applied. The sensitivity diagrams were created for the case study with HIPRE v. 1.22 available at <http://www.hipre.hut.fi>. Rank reversals occurred and were expected, mainly for quality aspects, i.e., “Quality Concepts”, “Cost”, and “Usability”. It was found, however, that rank reversals occurred also for the second-level quality aspects “Geographic Interaction” and “E-Process Knowledge”. The first-level percent-top critical quality aspect was “Cost”.

Figure 9.4 contains a sensitivity diagram showing rank reversal that occurred in our case study. The diagram shows that changing the “Quality Concepts”-score from the current value (about 0.14) to approximately 0.35 implies a rank reversal (i.e. SBUP will go on top of RUP). Knowledge about rank reversals may be important because for small changes in the quality aspect score they indicate caution is advisable regarding the e-Process assessment and reconsideration being an option. Our finding is remarkable because Table 9.5 shows that “Quality Concepts” is not a top-scoring high level quality aspect.

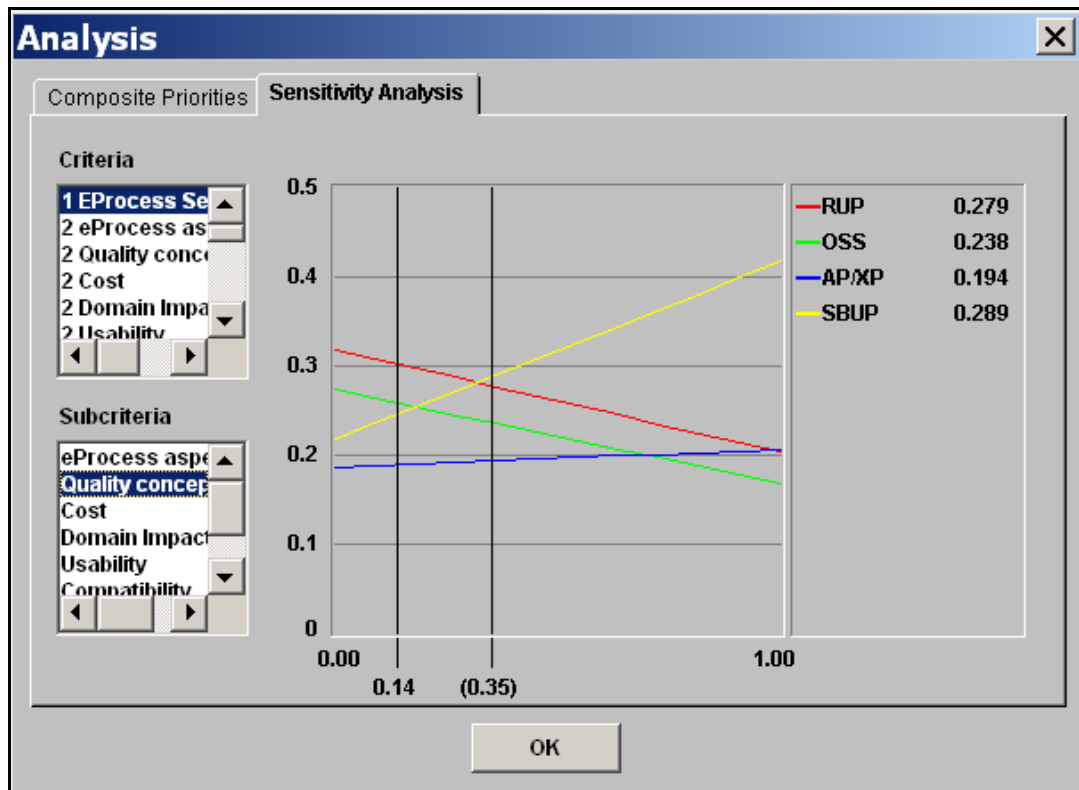


Figure 9.4: Sensitivity diagram for the Quality Concepts

AHP results interpretation

The developing company used the RUP to develop the abovementioned eCIS. This decision was not made based on the outcome of our research.

The reasons for using RUP were explored further by interviewing the developers. Their explanations included the following reasons:

- Developers have used other software processes on previous occasions, but came to the conclusion that their users wanted to have visible results sooner. The users needed to think about what they wanted included in the eCIS and wanted to see some outcomes before committing to further requirements. (This might indicate that the developer should have considered some of the other processes.)
- The fact that the user was able to visually see progress helped to form their specific requirements. It was also easy for the user to understand use case diagrams.
- It is also very important to their users to have budgets and project plans in place that are strictly controlled as well as having the ability to change the scope of the project if required.
- It was also felt that, as RUP is a strong process, integration of external developers would be simplified.

- Using RUP also allow for sub parts of the project to be developed quickly and then be able to show the user results quicker.
- It is easier to track iterative development.

The developers felt that as they had the expertise available in their company, they would be able to do a good job of the development. Limiting their use of artifacts provided them with a quality eCIS. Managing user requirements and providing visible sub-parts (using component architecture) were also results from using RUP. The developer had to manage changes strictly and limit requests for new requirements, but developing iteratively, focusing on pieces at a time, made the process easier.

This method is economically viable because its more complex parts are only applied if that is required. Even then, at least for smaller companies, applying explicit meta-development reasoning is an economical burden. Applying AHP to e-Process selection provides benefits that strengthen the justification to use it.

- First, in a virtual company context or for strategically allied companies it can be used for increasing resource utilisation.
- Second, the use of our method may pay off for larger (real or virtual) vendors that need to have a more diversified workforce anyway.
- Third, using this method may be beneficial even when the resulting recommendation is not followed, as one can get an impression of what kind of trend regarding the qualification of the workforce is emerging. This information can be exploited in staff education programs.
- Fourth, this method can also play a role in risk management, as a software vendor may decide not to sign a contract if our method suggests using a software process regarding the use of which the designated developers are not experienced and if these developers are not really confident about doing a project at hand with their preferred software process.

Some of the observations by the developers about the use of the e-Process Selection Methodology included:

- "Using the e-Process Selection Methodology will allow us to discuss our preferences and the complexities of the eCIS".
- "It will provide us with documentation on our choices".
- "It gives us a workable answer".

One staff member was not so sure whether the e-Process Selection Methodology will work, but conceded that even just talking about the problem helped them to clarify their thoughts.

This method was found to be the most expensive to use as it required extra time to implement. This was also the method that developers found the most difficult to understand, use and quite confusing at times. The results however showed that this method is very dependable and reliable and the application of sensitivity analysis ensures dependable results. This decision making method is widely used in other areas.

9.6.4 Validation of CBR

Validation of CBR was done by mainly reviewing the results obtained with the development team as well as applying another DMM to the same case study to be able to compare the results obtained.

Discussion of the results obtained from applying CBR to the case study Packaging eCIS

The results were discussed with the development team halfway through their development process. This was the last time the author of the thesis could gain access to the developers. The discussions about the results from applying CBR showed that the developers had a preference for using either storyboarding and user profiling or the Rational Unified Process. They did however confirm that the Rational Unified Process worked in the eventual development of the project even though they did not decide to use this process based on the recommendation from this study.

The developers felt that they learnt a lot using this development process and were quite pleased that the choice was made. Some of the comments, about the use of the e-Process Selection Methodology, made by the team included:

- "I am glad there are other projects out there that are similar to ours".
- "I really enjoyed using this development process and I am glad your system gave us this option."

The manager felt that there was benefit in discussing the options and having some support in selecting an e-Process and having some documentation on how the selection was made. He did say that in future he would be willing to use the e-Process Selection Methodology again – but would prefer the author of the thesis to steer them through the process.

The author of the thesis did not discuss the results of the further validation with the developing team, as the last access to the developers was halfway through the development of the eCIS.

Validation of CBR by applying a different DMM to the case study Packaging eCIS:

In order to validate the results obtained from applying CBR to the case problem – Packaging eCIS, the same data was used (see Table 9.9) and the VBA method was applied:

e-Process (n) =

$$\sum (\text{normalised HLQA} * \text{normalised QA} * \text{normalised e-Process QA}(n)) * 1000$$

producing the following results:

$$\text{RUP} = 43.08$$

$$\text{AM(XP)} = 37.06$$

$$\text{OSS} = 35.41$$

$$\text{SBUP} = 36.28$$

Using VBA, the result also came out as the Rational Unified Process being the recommended process. Storyboarding, which was a strong contender in the application of CBR, is however not the second recommended e-Process. A larger case base will most probably help to achieve more reliable data.

It was found that CBR was one of the less expensive decision making methods to use in the field as it did not take a long time to capture the quality aspect information. The focus here is on High Level Quality Aspects and previous cases. It will most probably be more expensive method to use if a large Case Base needs to be maintained. With a large Case Base – eCIS which is similar to cases in the Case Base will most probably provide very accurate results and be very quick to apply. Developers made comments about how easy it was to use this method.

High Level Quality Aspects	Quality Aspect(QA)	Weight of QA	RUP	AM(XP)	OSS	SBUP
e-Process aspects 0.20	Completeness	0.60	0.046	0.031	0.019	0.026
	Understandability	0.17	0.039	0.037	0.025	0.032
	Visibility	0.58	0.059	0.025	0.031	0.039
	Supportability	0.60	0.053	0.043	0.031	0.026
	Maintainability	0.19	0.053	0.037	0.031	0.026
Quality concepts 0.22	Readability	0.54	0.026	0.031	0.037	0.026
	Reliability	0.19	0.046	0.043	0.031	0.039
	Robustness	0.19	0.046	0.031	0.031	0.032
Cost 0.3	Development Budget	0.13	0.013	0.025	0.037	0.032
	Running Costs	0.13	0.007	0.025	0.037	0.032
Domain impact 0.1	Infrastructure	0.40	0.033	0.025	0.031	0.026
	Enterprise Culture	0.50	0.026	0.049	0.049	0.058
	Technology	0.54	0.039	0.031	0.043	0.039
	Geographic Interaction	0.10	0.033	0.043	0.037	0.032
	IT Strategy	0.58	0.020	0.025	0.031	0.032
	Business Strategy	0.54	0.013	0.019	0.037	0.032
	Team Experience	0.40	0.013	0.049	0.056	0.032
	Domain Knowledge	0.54	0.020	0.037	0.043	0.039
	E-Process knowledge	0.19	0.026	0.037	0.031	0.026
	Development Time	0.60	0.013	0.049	0.043	0.045
Usability 0.04	Functionality	0.60	0.026	0.031	0.025	0.032
	Manageability	0.60	0.020	0.031	0.031	0.039
	Quality assurance	0.55	0.046	0.031	0.031	0.039
	Adjustability	0.60	0.053	0.049	0.043	0.039
Compatibility 0.05	Exchangeability	0.38	0.046	0.037	0.037	0.045
	Map ability	0.10	0.039	0.031	0.037	0.026
Maturity 0.09	Stability	0.54	0.053	0.031	0.025	0.039
	Tool support	0.17	0.046	0.025	0.031	0.032
	Documentation	0.40	0.046	0.043	0.031	0.039

Table 9.9: CBR values used in applying VBA

9.6.5 Validation of SCM

Validation of SCM was done by mainly reviewing the results obtained with the development team as well as doing a cost benefit analysis on the case study to determine whether it has been cost effective to use the e-Process Selection Methodology.

Discussion of the results obtained from applying SCM to the case study Kayak eCIS

As SCM is very easy to apply and limited information is required to apply this method, this method was not further validated by a second application of a decision making method. Kayak eCIS is quite small and not much information was gathered at the time. If the case study was larger and more substantial data was available and gathered at the time of execution it would have been possible to further apply this case study to the any of the other decision making methods.

Follow up interviews were however conducted and some of the comments made by the developers included:

- “That was easy”.
- “Voting is fun and at least we discussed the problem”.
- “I think I always want to do it this way”.

To implement this method was found to be inexpensive and “fun”. It took minimal effort to implement and had input and discussion from the whole development team. There can however be a danger that the voting process will not work if the developers are inexperienced and lack knowledge about development. Therefore the skills of the team will determine the credibility of the resulting outcome. Developers loved this method as they are familiar with voting.

Cost/Benefit Analysis: Using SCM e-Process Selection for the Kayak eCIS

In order to determine whether using the e-Process Selection Methodology was beneficial and cost effective a cost benefit analysis was done on the application of SCM to the Kayak eCIS. This also formed further validation for the SCM e-Process Selection.

Applying the SCM e-Process Selection Methodology cost:

Interviewing Manager A – 30 minutes (\$80/hour)	\$ 40
Interviewing Manager B – 60 minutes (\$80/hour)	\$ 80
Interviewing 2 x developers 90 minutes (\$30/hour)	\$ 90

Applying SCM - 4 people (average if \$50/hour/person)	\$400
(This included general discussions of the issues involved)	
General costs – room, facilities etc.	\$ 50
TOTAL COST FOR APPLICATION:	\$660
Annual future costs associated with the use of the e-Process Selection Methodology:	\$ 0
Annual Benefits of using the e-Process Selection Methodology	\$1200
(Benefits were derived from taking 10% of the projected sales increase which can be predicted from an increased presence on the Internet).	

Cost/Benefit Analysis:

See Table 9.10 for the calculation of the Cost/Benefit Analysis.

Costs	Year 0	Year 1	Year 2	Year 3
Application Cost	NZ\$ 660			
Operational Cost		NZ\$ 0	NZ\$ 0	NZ\$ 0
Discount factors at 12%	1	0.893	0.797	0.712
Time adjusted costs	NZ\$ 660	NZ\$ 0	NZ\$ 0	NZ\$ 0
Cumulative Costs	NZ\$ 660	NZ\$ 660	NZ\$ 660	NZ\$ 660
Benefits derived				
implementing a quality eCIS	NZ\$ 0	NZ\$1200	NZ\$1200	NZ\$1200
Discount factors at 12%	1	0.893	0.797	0.712
Adj. Benefit SMEs	NZ\$ 0	NZ\$ 1 072	NZ\$ 956	NZ\$ 854
Cumulative Benefit SME	NZ\$ 0	NZ\$ 1 072	NZ\$ 2028	NZ\$ 2 882
Cumulative Cost/Benefit SMEs	-NZ\$ 660	NZ\$ 412	NZ\$ 1 368	NZ\$ 2 220

Table 9.10: Cost/Benefit Analysis of applying SCM to Kayak eCIS

As seen in Table 9.10 using the SCM e-Process Selection Methodology will show dividends within the first year after the implementation of the eCIS.

9.6.6 Analysis of results obtained by applying DMM

As said at the start of the chapter it would have been better if the same case study could have been used throughout. This however was not possible due to the research being conducted over a number of years and each decision making method being

applied to a case at the time of development. The type of case studies used was determined by the case study available at the time of testing and the developers being available for the research at that time. The ideal situation would have been to use the decision making method that fits a specific case the best. These areas of difference can be further applied on the prototype in the future.

In section 9.2 the VBA method was applied and the results were discussed with the developers. CBR was then applied to the same case in section 9.6.2 and similar results were obtained. Applying the VBA method produced an appropriate outcome. There were no problems working through this methodology. The developers were very familiar with the quality aspects to be rated and had no problem working through these to supply the data necessary for assessment.

In section 9.3 the AHP method was applied, the results were analysed and further sensitivity analysis applied on the results in section 9.6.3. The results were also discussed with the developers who were supportive of the results obtained. This decision making method produced appropriate results. It is important to note that to apply sensitivity analysis and the data takes longer, but allows for more accurate outcomes.

In section 9.4 the CBR method was applied and the results were discussed with the developers. VBA was then applied to the same case and similar results were obtained in section 9.6.4. The case study used here was a larger project. Using the knowledge base to determine the main contenders is quick and easy to use – but more data in the knowledge base is required.

In section 9.5 the SCM method was applied and the developers discussed and then decided on their chosen e-Process. This decision making method is very easy and quick to use and the developers is involved through the whole process.

Using the actual results obtained from the application of the research, a comparison was done on the different decision making methods after each has been validated and tested (see table 9.11, on the next page). The comparison is based on: The experiences of the author after applying and validating each DMM, and the quality aspects identified from the literature in section 8.1. Included in the table is also the actual time spent using each DMM which can be used in a detailed cost/benefit analysis. The author of the thesis does not have sufficient information at this exploratory stage to develop a quantitative comparison for DMM usage and this will be done at a later stage.

	VBA	AHP	CBR	SCM
Cost	Less expensive than AHP and CBR to use.	Expensive to use. Requires extra time to implement.	Cheaper – depending on the available Case Base size. To store case base can be expensive.	This methods is very cost effective to use as it takes little time and effort to implement.
Purpose of DMM	Applying this method and combining with sensitivity analysis provides quality outcomes.	DMM has been used widely in other areas and the outcomes have been proven. Applying sensitivity analysis ensures more accurate results.	Focus on High Level Quality Aspects and previous cases.	Focus on voting process. Quick, easy to use, major developer contribution. Knowledge of developers will determine the strength of the outcome.
Dependability	Results from application of VBA provide reasonably dependable results.	This method has proven to be the most dependable of the four methods trialled.	Depended on type of cases stored. With a large Case Base the results should be accurate.	Not too dependable or reliable – and skills of developers plays a major role.
Speed of results obtained	This method takes time to apply.	Takes a long time to apply in comparison to the other methods.	Quick to apply.	Quick to implement.
Actual time spent on case study to apply e-Process Selection	2 developers x 3 hours = 6 hours	3 developers x 4 hours + 2 for processing = 14 hours	2 developers x 2 hours = 4 hours	4 developers x 45minutes = 3 hours
Workflows	VBA has well defined workflows. The quality aspects (first and second level) are used thoroughly with this method.	The QA (first and second level) have more impact in this method. Workflow well structured.	Workflow well defined. Sensible method and should be used if the eCIS is similar to many other eCIS.	Familiar method for developers. Easy to follow the workflow.
Understand-ability	Is easy to understand	Logic not very visible to the lay person. Can be confusing to the developers.	Easy to use.	Very easy to apply.

Table 9.11 Comparing the decision making methods

9.6.7 Feasibility of using the e-Process Selection Methodology

Further interpretation of the results was conducted by doing a feasibility study on the implementation of the e-Process Selection Methodology. This feasibility study looked at Operational feasibility, Technical feasibility, Schedule feasibility, Economic feasibility and Legal feasibility, but the main focus was on the economic feasibility. "Economic feasibility is the measure used to measure the cost-effectiveness of a project or solution. This is often called cost-benefit analysis." (Whitten, et. al., 2007)

A feasibility study was carried out to determine whether it will be feasible to use the e-Process Selection Methodology when deciding on a specific eCIS. This was done in order to determine whether there will be benefits from using this methodology and whether the costs associated with the use of this methodology might be too high to be beneficial.

Operational Feasibility:

See Table 9.12 for the operational feasibility of the new e-Process Selection Methodology.

Question	
Will the proposed decision making system be able to document the decision process?	Yes
Will the proposed decision making system be easy to use?	Yes
Will the proposed decision making system provide adequate controls to guarantee the accuracy of the final decision?	More information will be available on the decision made.
Will the proposed decision making system help decision makers to investigate different options?	Yes
Will the proposed decision making system ensure a better eCIS based on better e-Process Selection?	To be proven with extensive use,

Table 9.12 Operational feasibility of the new e-Process Selection Methodology

Will there be any resistance expected from developers, decision makers and managers?

New decision makers and developers who want some support when selecting "new" development processes will benefit from the use of this e-Process Selection methodology. The effect that the use of the methodology will have on the work being carried on is that by using the fully developed tool – it will take between 1 and 5 people

– depending on the scale of the eCIS being developed – each 2 hours of evaluation time. It can be noted that developers might in any case have spent a large amount of time discussing different development process options and there might be resistance from developers who have been using different development processes over a number of years.

Technical Feasibility:

This type of feasibility test determines whether the proposed e-Process Selection Methodology is practical in terms of the technology which is available and the expertise which is required to develop and finally use this implemented methodology.

Is the appropriate hardware and software available to develop and support the proposed methodology?

In terms of use of this methodology – decision makers and developers will need to meet for a few hours and discuss the variable required to make a decision, but this task is not too difficult and is quite achievable. Initially no hardware or software will be required.

It might be necessary to develop a prototype tool to support the developers in the future (This has been initiated – see chapters 10 and 11 in this thesis). In order to achieve this, software and hardware is available in terms of web technology. In terms of support - new e-Processes can be added to the database and the expert knowledge can be added to the knowledge base. This will need to be monitored and managed, but this will be a task that a system administrator or similar can easily achieve. In terms of hardware and software requirements – the tool can be implemented and maintained on any website. The knowledge base and database can potentially grow, but the size of the growth can easily be accommodated on a basic sized web storage area.

Is the necessary expertise available to develop the e-Process Selection Methodology?

There is extensive research into decision making methods and a number of e-Processes are available. A meta-model for integrated four methods was presented in chapter 7 of this thesis. The author of this thesis has been researching this topic for a number of years. The expertise is therefore available.

The future prototype tool can possibly be developed using C#, ASP and Microsoft SQL. The author of the thesis has the skills to initiate such a prototype tool (see chapter 10 and 11).

Is this methodology going to require technical expertise from the users in terms of administration tasks required and in terms of the use of the methodology?

In terms of the use of the methodology, as stated above, decision makers and developers will need to meet for a few hours and discuss the variable required to make a decision, but this task is not too difficult and is quite achievable. The users will then need to enter the results from the discussions and maybe discuss the final recommendations before usage.

Schedule Feasibility:

This feasibility test determine whether the time frame available to complete the project is reasonable or not.

Is the time required to determine the data required to use the methodology reasonable?

This task will take 1 – 40 hours and 1 - 8 developers to gather the necessary data – depending on the size of the eCIS being developed. In a New Zealand context – many of the eCIS are developed in less than 8 months. In International terms this is quite small eCIS.

Usability analysis:

It is imperative that the e-Process selection be completed within the shortest time possible, without incurring too much cost in the selection process and that the final recommended e-Process is on a level where the resulting eCIS is of a high quality and that this quality is not compromised in any way. This is achievable as shown in this chapter.

Legal and Risk Feasibility:

This test of feasibility examines all the legal issues that may affect the selection using the e-Process Selection Methodology, as well as the identification of some of the possible risks associated with the use of this methodology. It is required that none of the legal issues are violated and that the risks are identified and stringently monitored and managed. Some of the legal issues and possible risks are:

- No material be used that might infringe on copyright law – the methodology suggested in this thesis do not have any copyright issues associated with it.
- No pirated software will be used in the execution of e-Process selection.
- It will be important to build in strong database security in the final tool to secure any sensitive company information that may be stored. In most cases the sensitive material being stored will be minimal.

- That no liability be in place based on any e-Process recommendation that was identified by the methodology. The final decision is still the decision makers' responsibility.
- That the copyright issues of the tool be clearly defined before full implementation.

Assumption: Only one decision making method will be used when applying the e-Process Selection Methodology.

Economical Feasibility:

This test of feasibility is performed to measure the cost effectiveness of using the e-Process selection methodology. In order to investigate this cost-benefit analysis will be done on the application of e-Process Selection based on a small to medium sized eCIS. The estimated cost of development and implementation can be seen in Table 9.13.

Development and implementation cost (Developer/s)	NZ\$ 1 500
Software	NZ\$ 580
Setting up a web site	NZ\$ 120/Year
<ul style="list-style-type: none"> • 100 GB Web Space • 100 GB Web Traffic • Host 2 Websites on Windows 2008 • Unlimited Domain & Pointer • 4 MySQL & 2 MS SQL 2008 DB. • Shared ASP.NET Hosting 	
Total Development Costs:	+/-NZ\$ 2 200

Table 9.13 Cost associated with the development of the e-Process Selection Methodology

Will it be economically feasible to use this system?

The cost of using the e-Process Selection Methodology can be seen in Table 9.14.

Small to medium sized system:	NZ\$ 600
A team of 3 people brainstorming for 4 hours to identify the figures required to apply the e-Process Selection Methodology (Developers/Decision makers/Users) 3 x 4 x \$50 (Based on using AHP which is the DMM that takes the most time to apply)	
Larger sized system:	NZ\$ 6000
A team of 6 people brainstorming for 20 hours to identify the figures required to apply the e-Process Selection Methodology (Developers/Decision makers/Users) 6 x 20 x \$50 (Based on using AHP which is the DMM that takes the most time to apply)	
Once only Operational Costs:	+/-NZ\$ 4000
Annual web site licensing fee	NZ\$ 120/Year
Annual Maintenance cost	NZ\$ 1000/Year
Annual operational and maintenance cost	NZ\$ 1120/Year

Table 9.14 Cost associated with the implementation of the e-Process Selection Methodology

Table 9.15 shows the cost/benefit analysis of using the e-Process Selection Methodology. The estimated Nett Present Value of using the selection methodology for a small to medium size company will be NZ\$3555 in the fourth year and for a larger company NZ\$10 082. Benefits were derived from using a small percentage of the business' increasing sales and turnover revenue based on a quality eCIS achieved by making a good decision on which e-Process to use. Even by using only a small percentage of the increased sales it can be concluded that for a small to medium sized eCIS the payback period is 3 years and for a larger eCIS the payback period is less than 2 years. The disadvantages of using the wrong e-Process will be substantial and cannot be measured.

Cost/Benefit Analysis					
Costs	Year 0	Year 1	Year 2	Year 3	Year 4
Development Cost	NZ\$ 2200				
Operational Cost (Staff time)		NZ\$ 5120	NZ\$ 1120	NZ\$ 1120	NZ\$ 1120
Discount factors at 12%	1	0.893	0.797	0.712	0.636
Time adjusted costs	NZ\$ 2200	NZ\$ 4572	NZ\$ 893	NZ\$ 797	NZ\$712
Cumulative Costs	NZ\$ 2200	NZ\$ 6772	NZ\$ 7665	NZ\$ 8 462	NZ\$ 9 174
Benefits derived from a small to medium sized eCIS					
Benefits derived from a small to medium sized eCIS	NZ\$ 0	NZ\$ 2 558	NZ\$3 600	NZ\$ 5 000	NZ\$ 6 300
Discount factors at 12%	1	0.893	0.797	0.712	0.636
Adj. Benefit SMEs	NZ\$ 0	NZ\$ 2 284	NZ\$ 2 869	NZ\$ 3 569	NZ\$4 007
Cumulative Benefit SME	NZ\$ 0	NZ\$ 2 284	NZ\$ 5 153	NZ\$ 8 722	NZ\$ 12 729
Cumulative Cost/Benefit SMEs	-NZ\$ 2 200	-NZ\$ 4 488	-NZ\$ 2 512	NZ\$ 1 260	NZ\$ 3 555
Benefits derived from a larger eCIS					
Benefits derived from a larger eCIS	NZ\$ 0	NZ\$ 4 000	NZ\$ 5 432	NZ\$ 6 722	NZ\$ 10 329
Adj. Benefits Larger Company	NZ\$ 0	NZ\$ 3 572	NZ\$ 4 329	NZ\$ 4 786	NZ\$ 6 569
Cumulative Benefit Large Company	NZ\$ 0	NZ\$ 3 572	NZ\$ 7 901	NZ\$12 687	NZ\$ 19 256
Cumulative Cost/Benefit Large Company	-NZ\$ 2 200	-NZ\$ 3 200	NZ\$ 236	NZ\$ 4 225	NZ\$ 10 082

Table 9.15 Cost/benefit analysis

9.7 Final Conclusions

The validation of the e-process selection methodology in this thesis was done by applying each of the decision making processes to a case study. This process was carried out over a number of years and it was not possible to use the same case study to apply all the decision making methods to the same case study. The results of each of the case study applications were either further validated by applying another decision making method to the same data (see section 9.2 and section 9.4) or by applying sensitivity analysis and/or weak-point analysis to the results (see section 9.2 and 9.3). Using sensitivity analysis and weak-point analysis it was possible to ensure that none of the quality aspects had an undue influence on the final recommendation. Even though RUP was not used based on the recommendation made from the e-Process Selection in section 9.3 using the DMM validated the use of RUP because the developers gained through applying the e-Process Selection. Further it can be stated that although the outcome of applying e-Process Selection was not necessary the determining factor in the developers decision on which e-Process to use, the result was a close match to their choice (see section 9.3).

A cost/benefit analysis was carried out on the SCM method case study (see section 9.5) to show that it was cost effective to use. The results were validated by discussing the results with the development team to get their feedback and in all cases some merit could be seen in using this selection methodology.

It can also be seen from section 9.7.1 that the return of investment of using a suitable e-Process is substantial while choosing the wrong e-Process could possibly have an impact on the final resulting eCIS.

In order to further improve the benefits of using e-Process Selection, it is advised that an e-Process Selection Tool be developed to support the developers and to simplify the application of e-Process selection. Part D presents the prototype design documentation to allow for future development and improvement as well as examples of the prototype tool to allow for refinement in future research.

PART D

PROTOTYPE DEVELOPMENT

Chapter 10

PROTOTYPE DEVELOPMENT DOCUMENTATION

e-Process Selection Tool (eProcSel)

This chapter provides the analysis and design documentation for eProcSel.

10.1 Introduction

This chapter contains the analysis and design documentation required to develop the prototype tool for e-Process Selection. This selection tool prototype is called eProcSel. The prototype itself is presented in chapter 11. The integrated meta-model for eProcess Selection can be found in chapter seven. This was used to develop the layout of the database for eProcSel. Descriptions of the functionality of the different decision making methods can be found in chapter 7 with the application of the methods chapter 9. These chapters were used to develop the documentation contained in this chapter. This chapter will follow the framework seen in Figure 10.1.

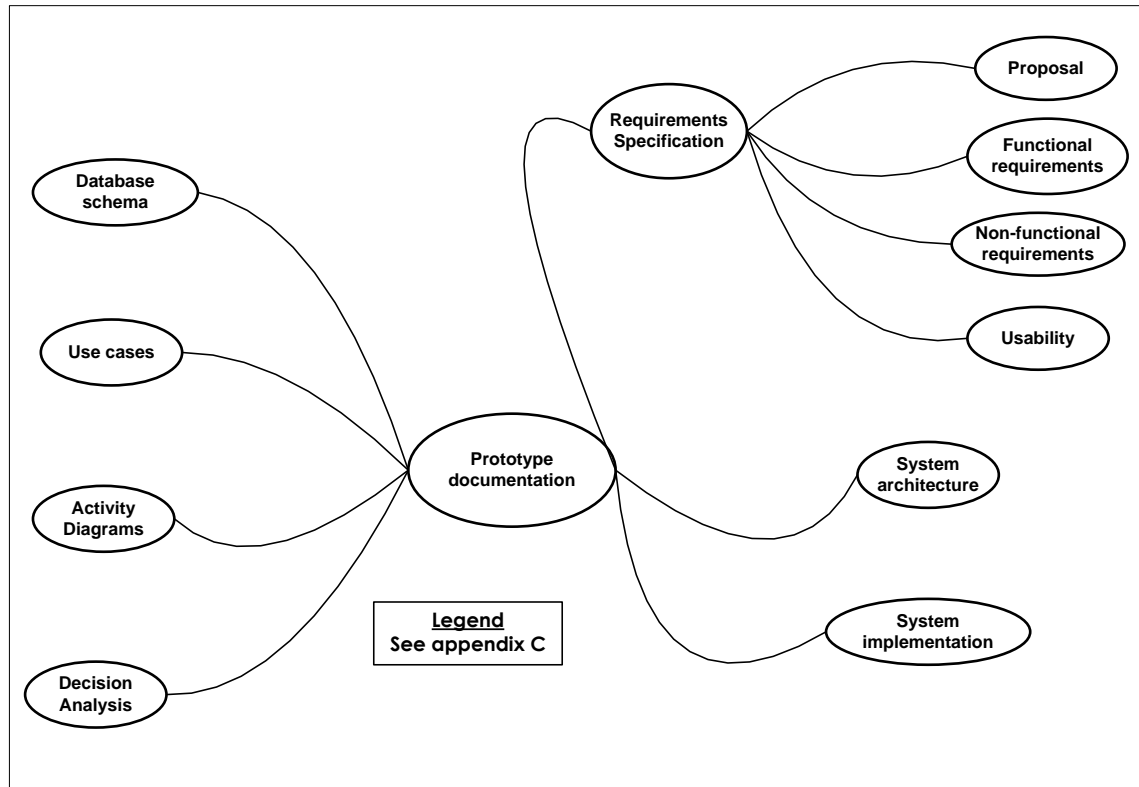


Figure 10.1: Framework for eProcSel specification

The rest of the chapter provides the reader with detail about eProcSel.

10.2 Requirement Specification

The proposal was to develop an e-Process selection tool to prove that the concept was feasible. The tool needed to be automated and easy to use. The tool had to be suitable for use by anyone that wanted assistance with selecting e-processes using one of the four proposed decision making methods.

Requirements

Functional requirements

User:

- EProcSel assesses whether a user is eligible to use the tool.
- EProcSel allows the user to choose the decision making options to be used.
- EProcSel displays the quality aspects (high level and second level) information and allows the user to assign a value to these criteria which are then applied.
- EProcSel provides a view of historical information.
- A recommendation on which eProcess to use.
- Allows for Project and Feedback information to be entered

Developer side:

- Process high level or second level quality aspects (Add and edit)
- Process e-processes (restricted in the initial model to four see chapter three)

Expert side:

- Process e-Process QA values

Non-Functional requirements

Performance criteria (response times etc.)

Anticipated volumes of data: Limited volumes of data will be accommodated.

Security considerations: Only valid user should use eProcSel – login and security.

Keep other QA as identified in chapter 4 in mind.

Usability requirements

Characteristic of the users:

- Identify characteristic of users - managers or decision making IT personnel that have to decide on which e-process to use for eCIS development.
- Identify characteristics of experts and developers/administrators,

Tasks and goals users undertake: The user will have to know their specific setup before running eProcSel – need to know the commitment of company in terms of development process, available budget, manpower commitment, skills-sets, etc.

Situation factors that may arise during system use

If the criteria to be used keep growing it may be worth allowing the user to store their input and continue using the tool at a later stage.

Acceptance criteria, which the user will judge the delivered system

10.3 Use case diagrams

Using the descriptions as defined in chapter 7 and Part C the use case diagrams for eProcSel can be found in:

- Figure 10.2: eProcSel Use Case Diagram
- Figure 10.3: VBA Use Case Diagram
- Figure 10.4: AHP Use Case Diagram
- Figure 10.5: CBR Use Case Diagram
- Figure 10.6: SCM Use Case Diagram

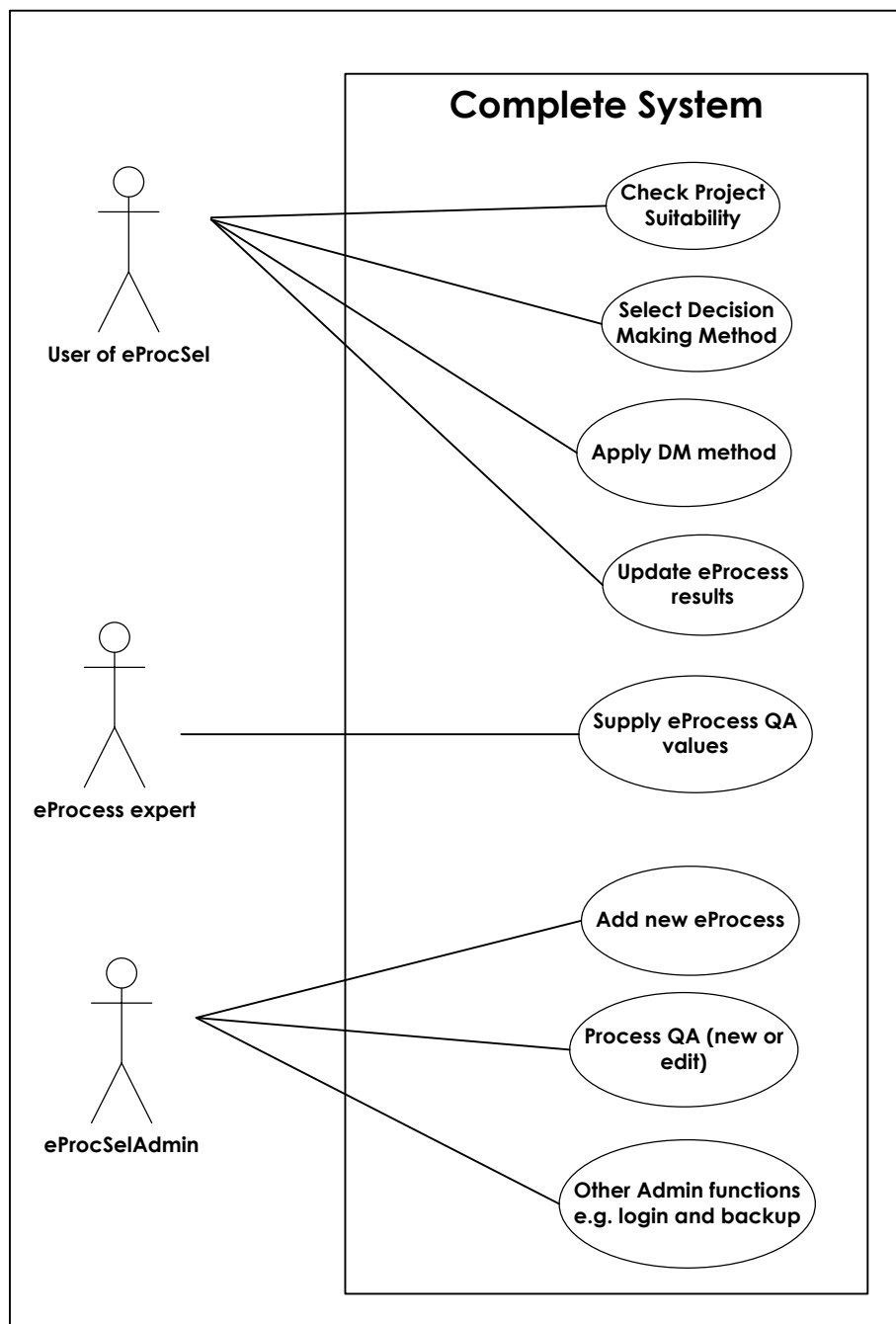


Figure 10.2: eProcSel Use Case Diagram

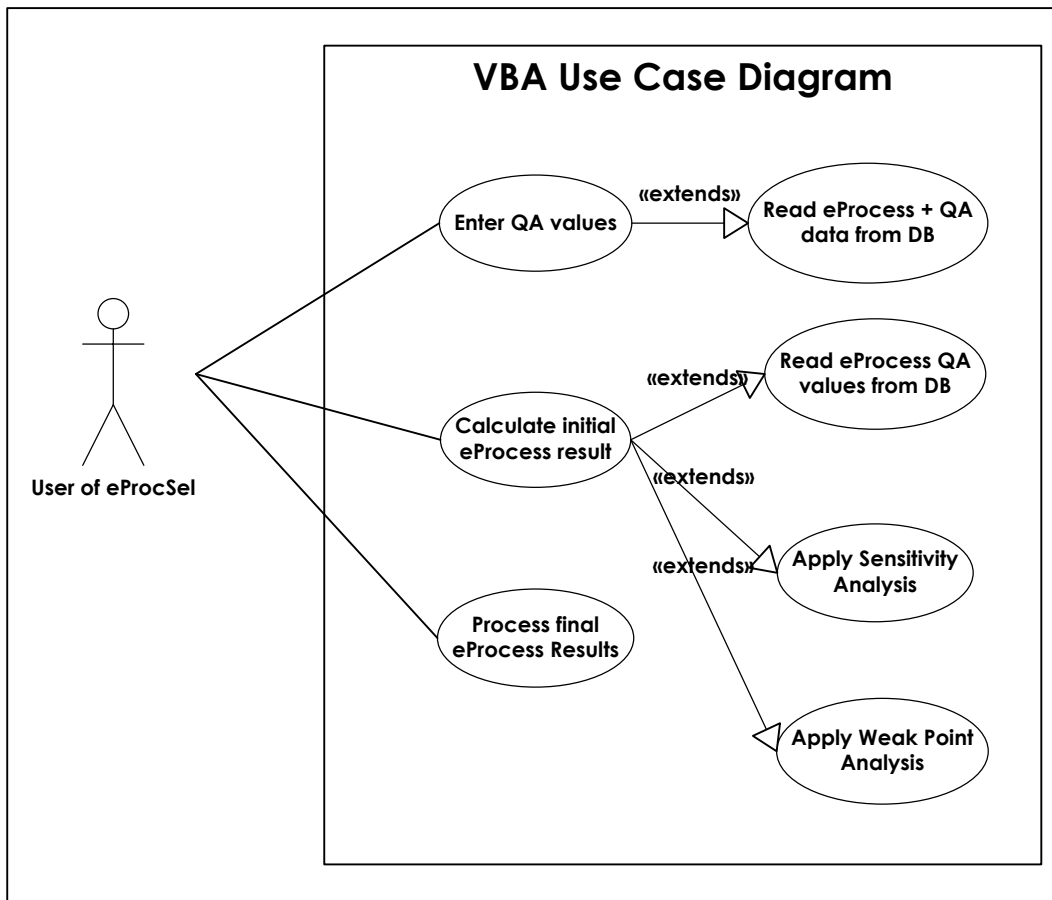


Figure 10.3: VBA Use Case Diagram

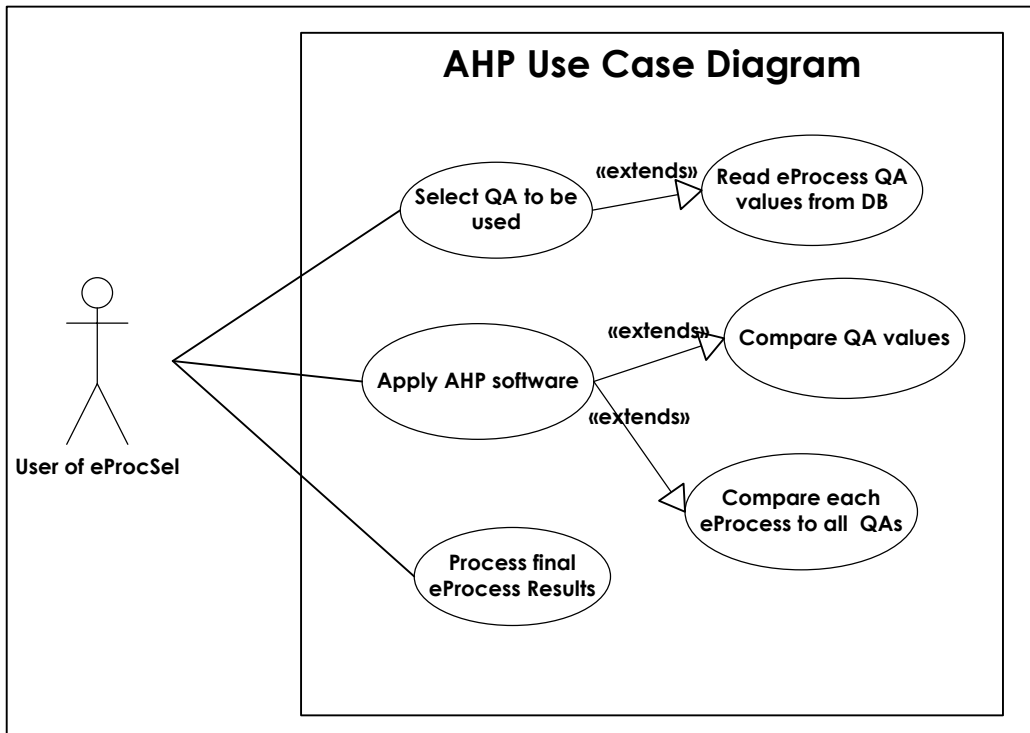


Figure 10.4: AHP Use Case Diagram

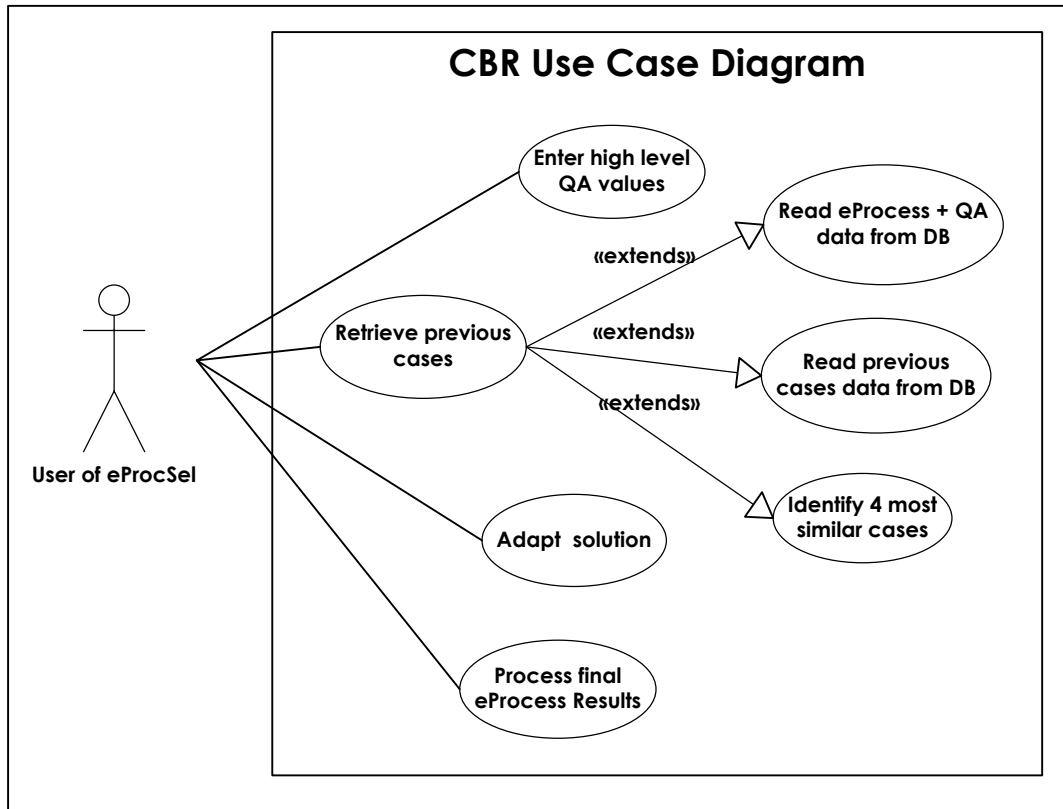


Figure 10.5: CBR Use Case Diagram

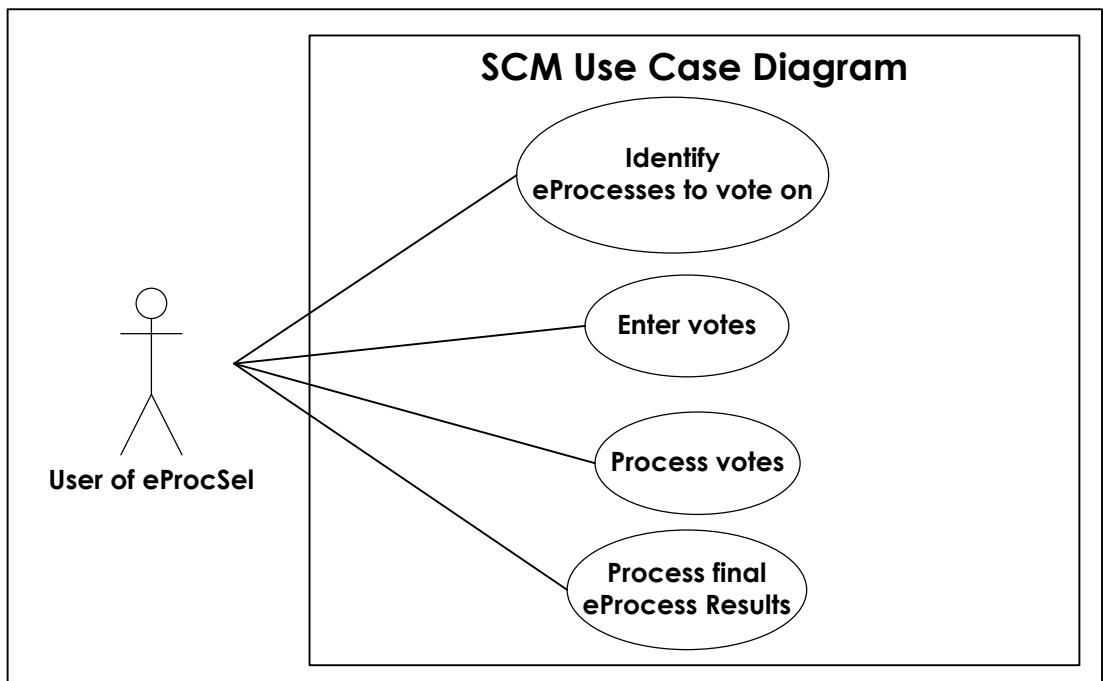


Figure 10.6: SCM Use Case Diagram

10.4 Activity Diagrams

For the basic activities of eProcSel, see:

- Figure 10.7: eProcSel Activity Diagram
- Figure 10.8: EProcSel VBA Decision Making Method Activity Diagram
- Figure 10.9: EProcSel AHP Decision Making Method Activity Diagram
- Figure 10.10: EProcSel CBR Decision Making Method Activity Diagram
- Figure 10.11: EProcSel SCM Decision Making Method Activity Diagram

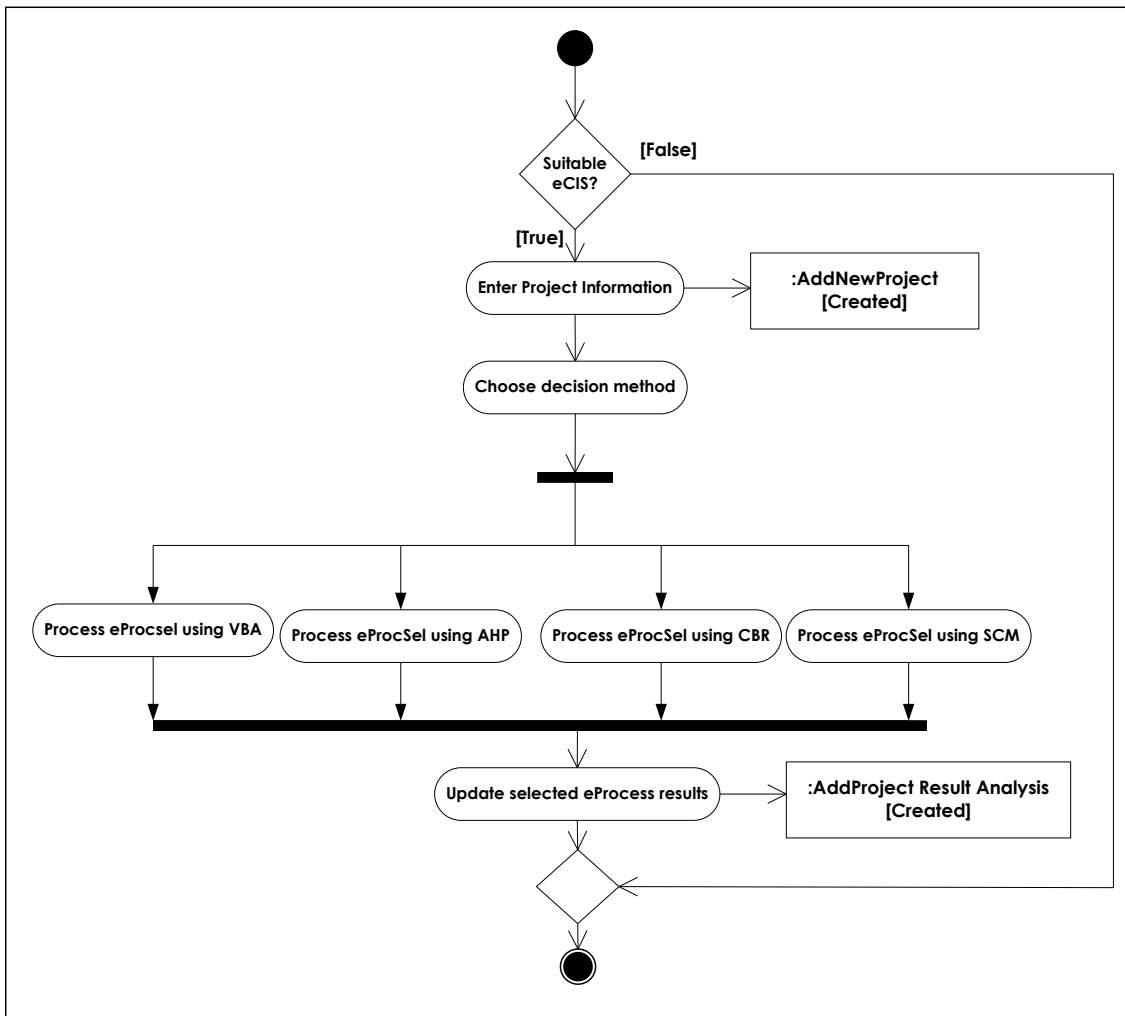


Figure 10.7: eProcSel Activity Diagram

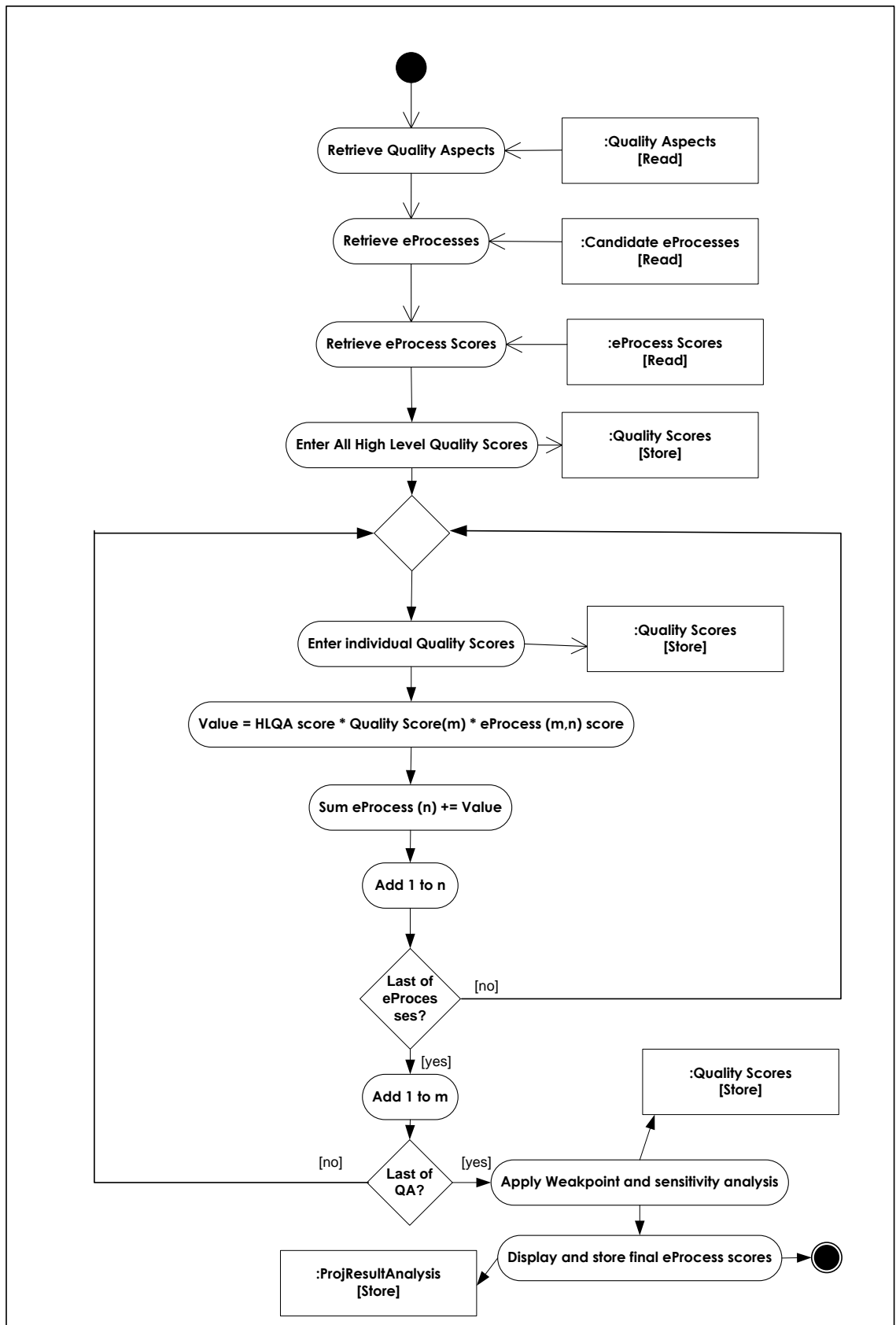


Figure 10.8: EProcSel VBA Decision Making Method Activity Diagram

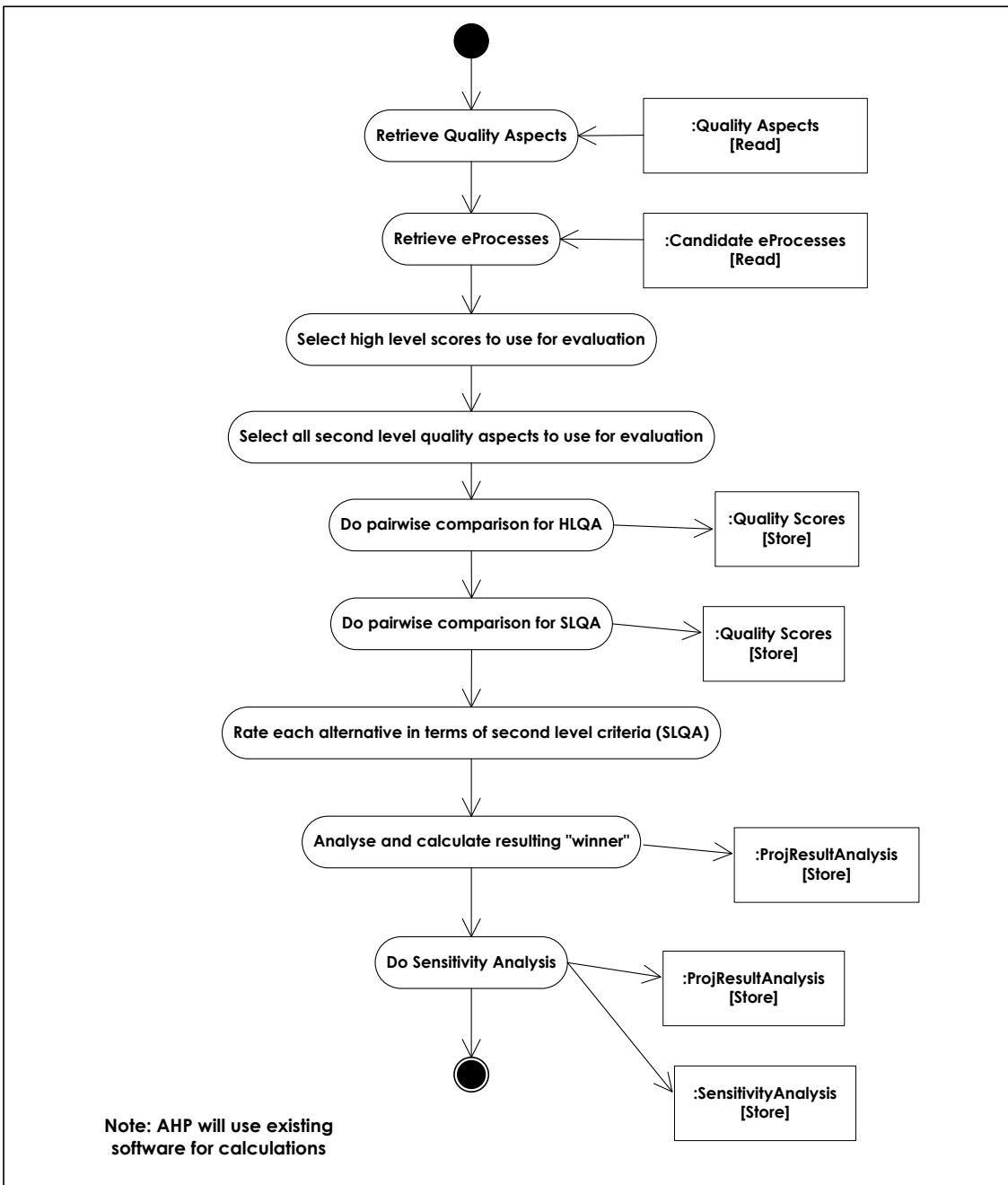


Figure 10.9: EProcSel AHP Decision Making Method Activity Diagram

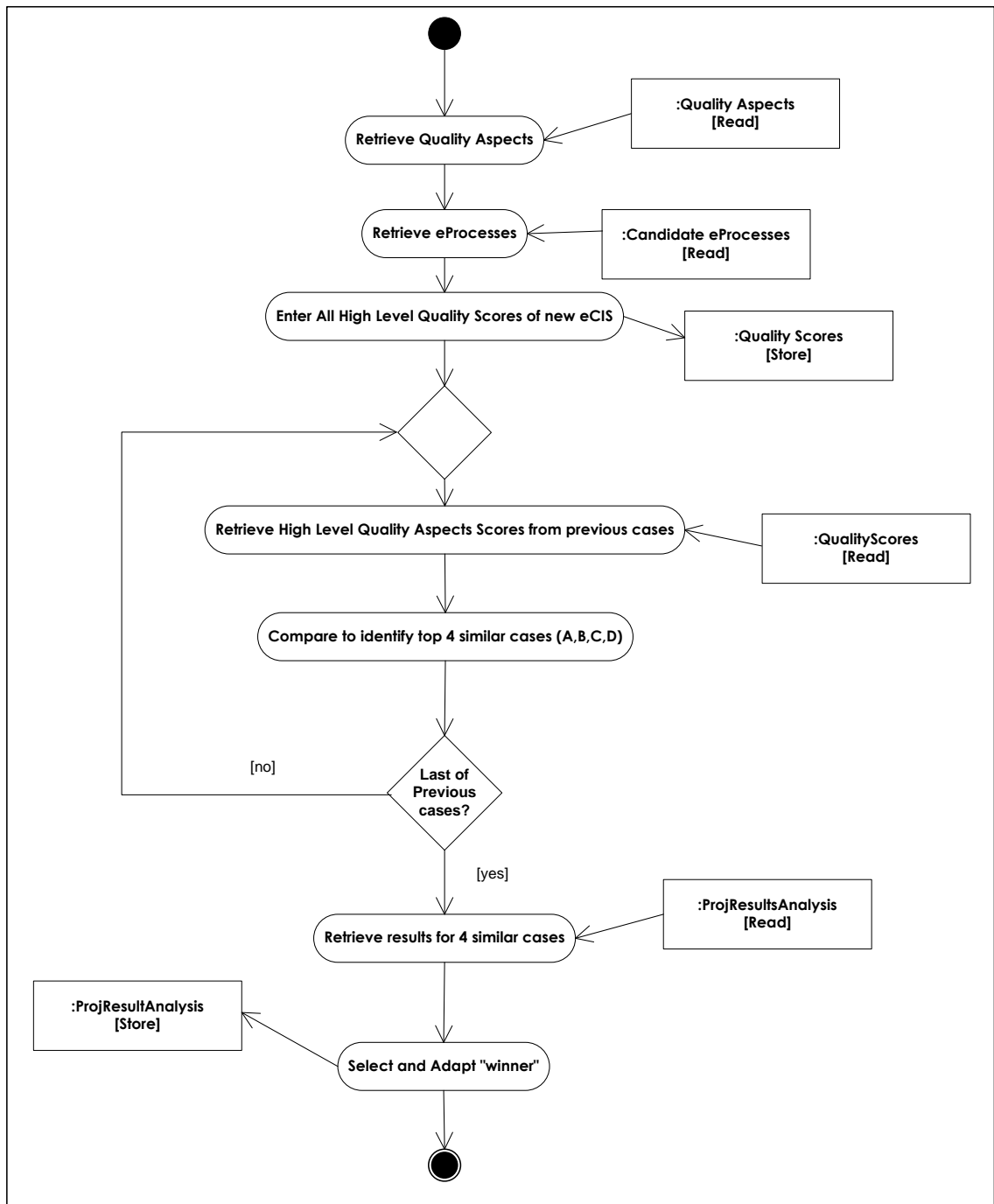


Figure 10.10: EProcSel CBR Decision Making Method Activity Diagram

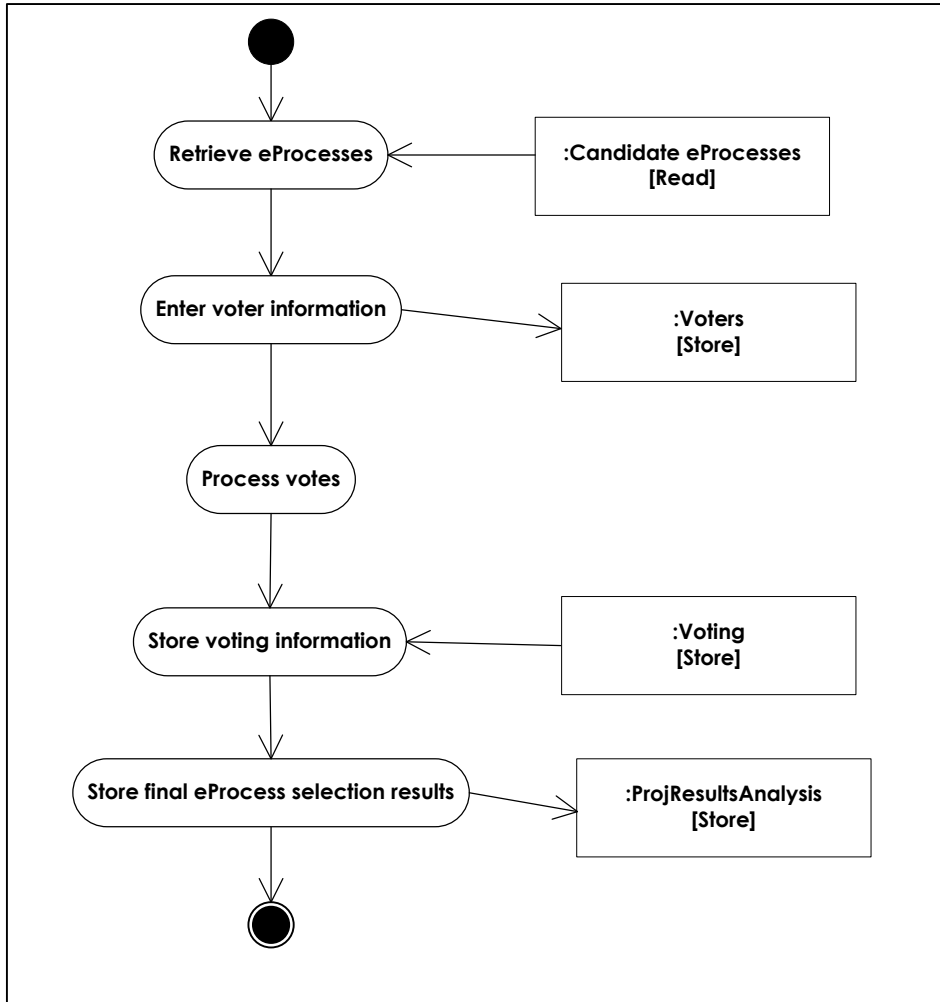


Figure 10.11: EProcSel SCM Decision Making Method Activity Diagram

10.5 Decision Analysis

In this section the different candidate solutions were evaluated in order to determine the best suited candidate solution. (See Table 10.1, on the next page).

Characteristics	Solution Candidate 1	Solution Candidate 2	Solution Candidate 3
Portion of System Computerised	Develop complete eProcSel.	Develop VBA and CBR, incorporate use of choice results software for AHP and provide instructions for use of SCM.	Same as candidate 2.
Benefits/Constraint	No 30-day use or pay license fee restrictions on own development.	Restriction on use of sourced AHP software.	Restriction on use of sourced AHP software.
Servers and Workstations	Single PC	Website – support for ASP and MS SQL	Website – support for PHP and MySQL.
Software Tools Needed	Use Visual Studio – Visual Basic for development and run on flat files.	Use Visual Studio – ASP, C# and Microsoft SQL for development.	Open Sources options: PHP and MySQL
Method of Data Processing	Immediate transaction processing.	Online Web	Online Web
Input and Output Devices and Implications	Keyboard and screen.	Same as 1.	Same as 1.
Storage Devices and Implications	Flat files	MS SQL database	MySQL database.

Table 10.1: Comparison of candidate solutions (Template from Whitten, et al., 2007)

See Table 10.2, on the next page, where, based the calculations in this table, the recommendation was to develop candidate 2.

Feasibility Criteria	Wt.	Candidate 1	Candidate 2	Candidate 3
Operational Feasibility Functionality and .Political.	40%	System can be adjusted to fit requirements, but AHP programming is time consuming. Work well on single PC – difficult to make widely available. Can be individually distributed which will have support of management and organisations. Score: 65	Microsft environment has extensive development support. Easy to distribute over the Web. Will have support of users, management and organisations. Score: 95	Open sourced development needs to be sourced and implemented. Easy to distribute over the Web. Will have support of users, management and organisations. Web sites do not always support PHP and MySQL. Score: 80
Technical Feasibility Technology and Expertise.	30%	Visual Basic and Visual Studio environment easy to develop. Flat files easy to set up. Technical expertise available for development. VB not a well-supported development environment. Score: 70	Visual Studio 2005 supports ASP, C# and MS SQL development. Developer needs to refine knowledge. Score: 75	PHP and MySQL are open source software. Easy to obtain – sometimes tricky to install. Developer needs to refine knowledge. Score: 65
Economic Feasibility	20%	Software available but of lesser benefit. Score: 70	Software available and supported extensively in industry currently. Score: 60	Cost effective, but of lesser benefit. Score: 80
Schedule Feasibility	10%	2 months Score: 75	2 months Score: 75	2 months Score: 75
Ranking:	100%	68.5	80.0	75.0

Table 10.2 Feasibility Analysis Matrix (Template from Whitten, et al., 2007)

10.6 Three-tier System Software Architecture

The architecture of the e-Process Selection Tool consisted of the client side and server side connected via the Internet or similar.

Presentation layer: This is the different forms contained in eProcSel and provides the user interface. These forms is used for the user input and displaying the results on the screen and the programme uses this layer to get and set the data back and forth. These forms contain the controls like textbox, labels, command buttons etc.

Business layer with the property layer: This is a class where the functions get the data from the different forms and after verification passes the information through to the data access layer. The business layer has a sub-layer where the properties are set or get from the forms.

Data layer: This is a class which receives data from the business layer and then communicates with the MS SQL database or sent the data from the MS SQL database to the business layer.

The architecture of the system can be seen in Figure 12.12.

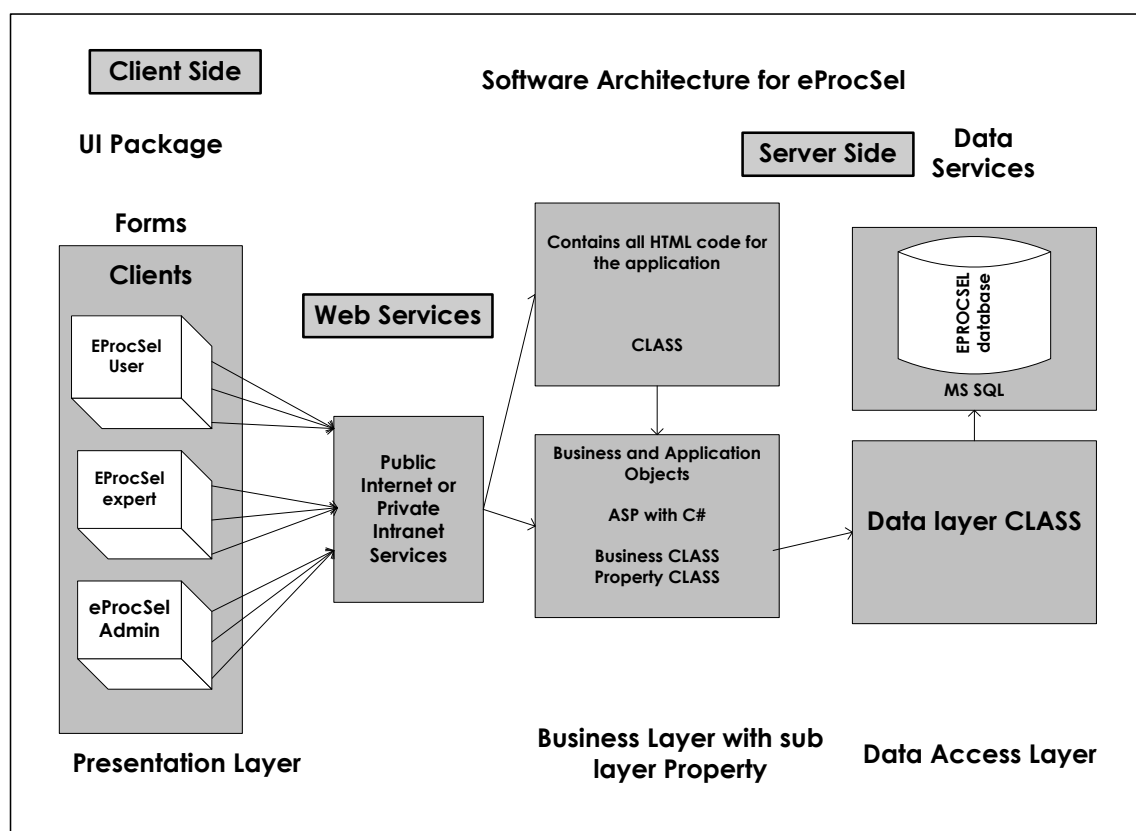


Figure 10.12: System Software Architecture

Some of the initial code for retrieving candidate eProcess from the database follows (initially developed using (Bean_software, 2007)):

Presentation Layer:

```
private void DataGrid1_SelectedIndexChanged(object sender, System.EventArgs e)
{
    // Instantiate the object of the Property layer
    clseProcess objproperty=new clseProcess();

    // Instantiate the object of the business layer
    clseProcessInfo objbs=new clseProcessInfo();

    // Object of the dataset: receive the data sent by the business layer
    DataSet ds=new DataSet();

    // place value in the property using the object of the property layer
    objproperty.id=int.Parse(DataGrid1.SelectedItem.Cells[1].Text.ToString());

    // Call a function from the business layer and passing the object of the property layer
    // which will carry the ID from the database.
    ds=objbs.GetAlleProcessIDWise(objproperty);

    // Data returned by the above function into the dataset - populate through
    // the presentation layet.
    txtEProcessId.Text=ds.Tables[0].Rows[0][0].ToString();
    txtEProcessname.Text=ds.Tables[0].Rows[0][1].ToString();
    txtEProcessDescriptionAddress.Text=ds.Tables[0].Rows[0][2].ToString();
}
```

Property Layer

```
public class clseProcess
// Using the object of the property layer -set or get the data to or from the properties.
{
    // Class for Candidate eProcess Table
    private string _id;
    private string _Name;
    private string _Desc;

    public int id
    {
        // Property to set or get the value into _id variable

        get{return _id;}
        set{_id=value;}
    }

    public string Name
```

```

{
    // Defining the Name Variable for I/O
    get{return _Name;}
    set{_Name=value;}
}

public string Description
{
    // Defining the Description Variable for I/O
    get{return _Descr;}
    set{_Descr=value;}
}
}

```

Business Layer:

"Obj" is the object of the clseProcess class has been defined in the property layer. This function is receiving the property object and passing it to the datalayer class

```

public class clseProcessInfo
// this is the function of the business layer which accepts the data
//from the application layer and passes it to the data layer.
{
    public DataSet GetAllProcessIDWise(objproperty);
    {
        DataSet ds=new DataSet();
        ds=objdt.getdata_dtIDWise(obj);    // Calling of Data layer function
        return ds;
    }
}

```

Data Layer

// Data layer function - receives data from the business layer and performing the required operation into the database

```

public class clseProcessData // Data layer class
{
    public DataSet getdata_dtIDWise(clseProcess obj)
        // object of property layer class
    {
        DataSet ds;
        string sql;
        sql="select * from eProcess where eProcessId="+obj.id+" order by eProcessId";
        ds=new DataSet();
        // datalayer function - accepts tre sql query and
        // performs the corresponding operation
        ds=objdt.ExecuteSql(sql);
        return ds;
    }
}

```


10.7 Database Schema

Using the Integrated Meta-Model from chapter 7, the following schema for the database was developed:

Candidate_eProcesses

EProcess_Id	varchar (4)	NOT NULL
EProcess_name	varchar (30)	NOT NULL
EProcess_description	varchar (50)	NOT NULL

Primary Key - EProcess_Id

EProcess_Scores:

EProcess_Id	varchar (4)	NOT NULL	Foreign Key
Quality_Aspects_Id	varchar (4)	NOT NULL	Foreign Key
Score	int	NOT NULL	
Assessor_Id	varchar (4)	NOT NULL	Foreign Key

Primary Key - EProcess_Id + Quality_Aspects_Id

Assessor

Assessor_Id	varchar (4)	NOT NULL
Assessor_name	varchar (30)	NOT NULL
Assessor_email	varchar (20)	NOT NULL

Primary Key - Assessor_Id

Voters

Voter_Id	varchar (4)	NOT NULL	
Voter_name	varchar (30)	NOT NULL	
Voter-Quality	int	NOT NULL	
Project_Id	varchar (4)	NOT NULL	Foreign Key

Primary Key – Voter_Id

Project

Project_Id	varchar (4)	NOT NULL
Project_name	varchar (30)	NOT NULL
Project_manager	varchar (30)	NOT NULL
Project_manager_email	varchar (20)	NOT NULL

Primary Key - Project_Id

Voting

Voter_Id	varchar (4)	NOT NULL	Foreign Key
EProcess_Id	varchar (4)	NOT NULL	Foreign Key
Reason	varchar (100)	NOT NULL	
Feedback	varchar (100)	NOT NULL	
Primary Key - Voter_Id + EProcess_Id			

Quality_Aspects

Quality_Aspects_Id	varchar (4)	NOT NULL	
Quality_Aspects_Name	varchar (30)	NOT NULL	
Quality_Aspects_Description	varchar (100)	NOT NULL	
Quality_Aspects_High_Level_ID			
	varchar (4)	NOT NULL	Foreign Key
Time_of_Assessment	decimal	NOT NULL	
Primary Key - Quality_Aspects_Id			

Quality_Scores

Project_Id	varchar (4)	NOT NULL	Foreign Key
Quality_Aspects_Id	varchar (4)	NOT NULL	Foreign Key
ProjectQualityWeight	decimal	NOT NULL	
Second_Quality_Aspects_Id	varchar (4)	NOT NULL	Foreign Key
Primary Key - Quality_Aspects_Id + Project_Id			

QProject_Result_Analysis

Project_Id	varchar (4)	NOT NULL	Foreign Key
Recommended_EProcess_Id	varchar (4)	NOT NULL	Foreign Key
Results_and_Comments	varchar (150)		
Feedback_on_reuse	varchar (150)		
Primary Key - Project_Id + Recommended_EProcess_Id			

Quality_Aspects_Analysis

Project_Id	varchar (4)	NOT NULL	Foreign Key
Quality_Aspects_Id	varchar (4)	NOT NULL	Foreign Key
Reasoning	varchar (150)		
Adjustments_made	varchar (150)		
Primary Key - Quality_Aspects_Id + Project_Id			

WeakPoint_Analysis

Project_Id	varchar (4)	NOT NULL	Foreign Key
Quality_Aspects_Id	varchar (4)	NOT NULL	Foreign Key
ProjectQualityWeight	decimal	NOT NULL	
NewQualityWeight	decimal	NOT NULL	
Reasoning	varchar (150)		
Primary Key - Quality_Aspects_Id + Project_Id			

Sensitivity_Analysis

Project_Id	varchar (4)	NOT NULL	Foreign Key
Quality_Aspects_Id	varchar (4)	NOT NULL	Foreign Key
ProjectQualityWeight	decimal	NOT NULL	
NewQualityWeight	decimal	NOT NULL	
Reasoning	varchar (150)		
Primary Key - Quality_Aspects_Id + Project_Id			

10.8 System Implementation

The basic implementation screens of the prototype tool can be found in chapter 11.

Chapter 11

PROTOTYPE TOOL

This chapter focuses on the implementation of the prototype tool.

11.1 Introduction

eProcSel was developed in phases. These phases are:

- Phase one of the development of eProcSel consisted of the development of the system to use the Value Benefit Analysis method for decision making – see chapter 7.2, 9 and 10 for the detail about VBA.
- Phase two of the development of the prototype consisted of using existing software for the Analytic Hierarchy Process for decision making to trial the process – see chapter 7.3, 9 and 10 for the detail on AHP.
- Phase three of the development of the prototype consisted of adapting phase one to store information that could be used for Case Based Reasoning and then to implement the software required for CBR – see chapter 7.4, 9 and 10 for the detail on CBR.
- Phase four of the development of the prototype consisted of implementing SCM by providing an explanation on how to execute this process and then to capture the resulting information – see chapter 7.5, 9 and 10 for the detail on SCM.

11.2 Phase 1 – focus on VBA

The first phase of the development prototype was based on the weighted scoring method (VBA method)

Examples of parts of the application of the VBA method can be seen in:

- Figure 11.1
- Figure 11.2
- Figure 11.3
- Figure 11.4

The initial screen contains a menu with all the options available in the tool. These options include the option to check whether this specific project qualifies for using the prototype tool; the application of the tool as well as a section which is used for the

maintenance of the database. The application of the tool also consists of two main sections namely one to assign or decide on the importance of each of the high level quality aspects in relation to the development of this eCIS and then once that has been decided, to provide the data for the individual importance of each of the second level quality aspects.

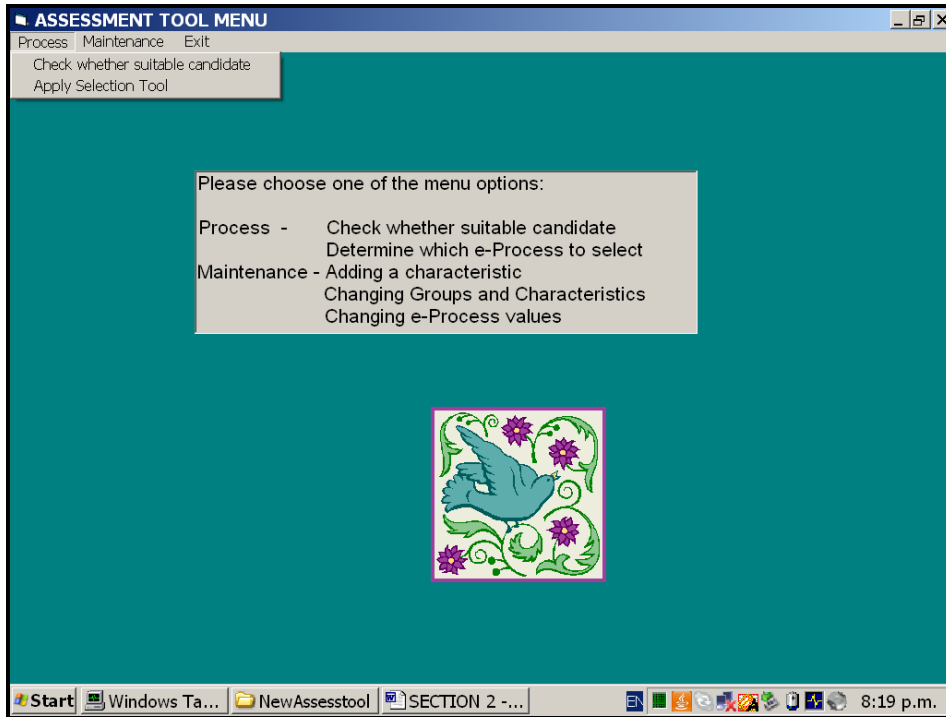


Figure 11.1: VBA Initial Screen

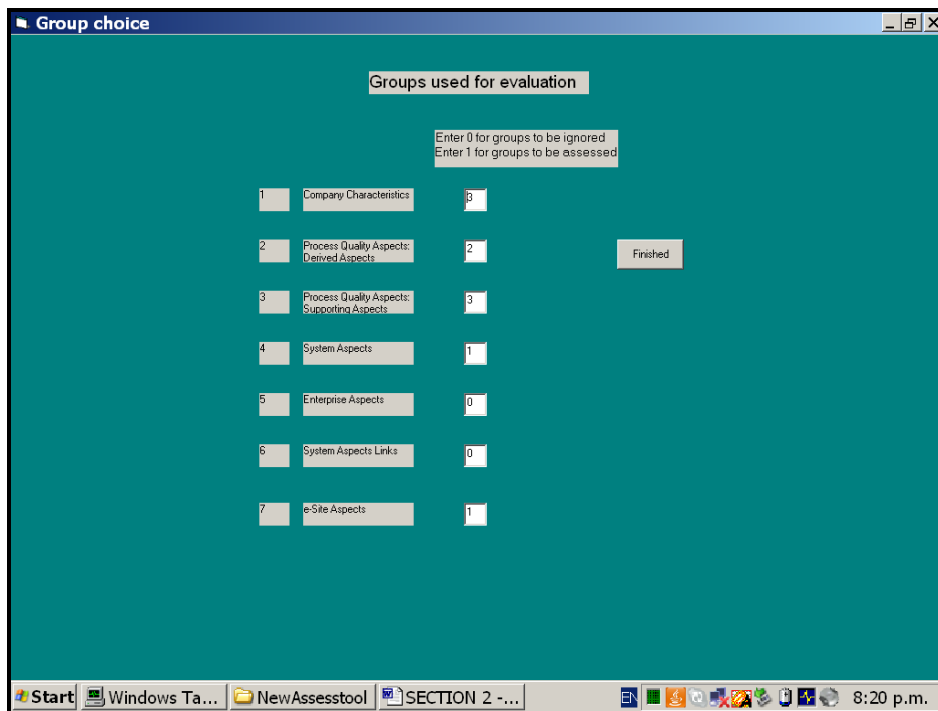


Figure 11.2: VBA High Level Quality Aspect Entry

Are you a suitable candidate to use this tool - Please complete the following questions:

Please tick the option that best describes your business:

Small-to-medium sized business
 Large Business
 Corporation

Are you planning to develop a new e-commerce site?

Yes
 No

Please tick the option that best describes the development you want to do:

B2C
 B2B
 Other development

Tick one of the following options : (ERP = Enterprise Resource planning software)

J.D.Edwards
 PeopleSoft Inc
 SAP AG
 Oracle
 Other ERP
 No ERP

Please tick the options that your company currently supports on an internal corporate system.

Sales system
 Payroll System
 Inventory System
 A/ R System
 Billing System
 No Formal System

Return to menu Evaluate

Figure 11.3: First Phase Suitability Check

e-Process Assessment

Assessing characteristics to determine which e-Process will best suite your needs

Company Characteristics

ITStrategy

Rate the importance of this characteristic : (1 = Low, 5 = High)

1
 2
 3
 4
 5

IT strategy: The degree to which the e-process matches the company's IT Strategy.

Start Windows Ta... NewAssesstool SECTION 2 - ... 8:26 p.m.

Figure 11.4: VBA Second Level Quality Aspect Values Entry

11.3 Phase 2: focus on AHP

A number of software tools have been developed that can be used to apply to the selection of an e-Process using the analytic hierarchy process (AHP). A number of different AHP tools were trialled. These included Web Hipre version 1.22 (<http://www.hipre.hut.fi/>) and ChoiceResults (<http://choiceresults.download-141-28160.programsbase.com/>). Both these applications provide functionality for the modelling and the analysing of complex decisions and the final results.

In the end it was decided to use Web Hipre (Hämäläinen, 2008), as it was applied in earlier research undertaken. The quality aspects were agreed upon by the developers using brainstorming – most of the quality aspects from chapter 4 were included for the testing of the tool. The number of quality aspects used, however, needs to be limited otherwise the number of pair-wise comparisons becomes too large to manage and that will most probably be recommended for future applications of e-Process selection using AHP.

Next these high level quality aspects are then pair-wise evaluated which results in the decision weights. Then each of the groups (branch from a high level quality aspect) of second level quality aspects is pair-wise evaluated. The alternatives are evaluated against each second level quality aspect to determine the results. These evaluations are then processed – see section 7.3 for an explanation of the calculations – to produce the results.

The results are usually graphically presented. These are then evaluated by analysing the different factors used for evaluation and then determining the sensitivity of those factors to the decision – see also section 7.3 for sensitivity analysis.

Some of the resulting graphs can be seen in:

- Figure 11.5
- Figure 11.6
- Figure 11.7
- Figure 11.8

The results can be stored in the database see previous chapter for database schema.

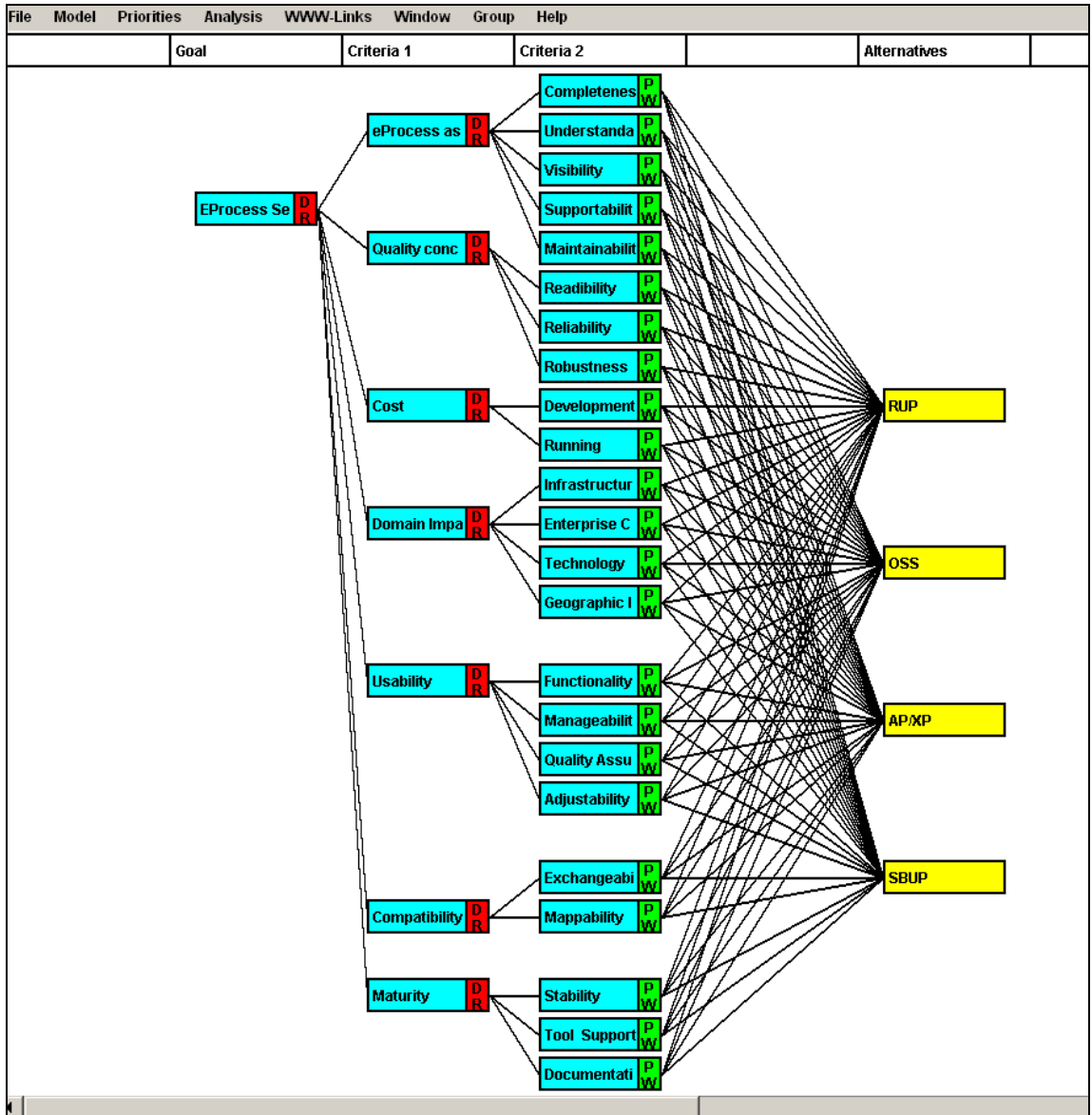


Figure 11.5 Initial setup of goal, criteria and alternatives

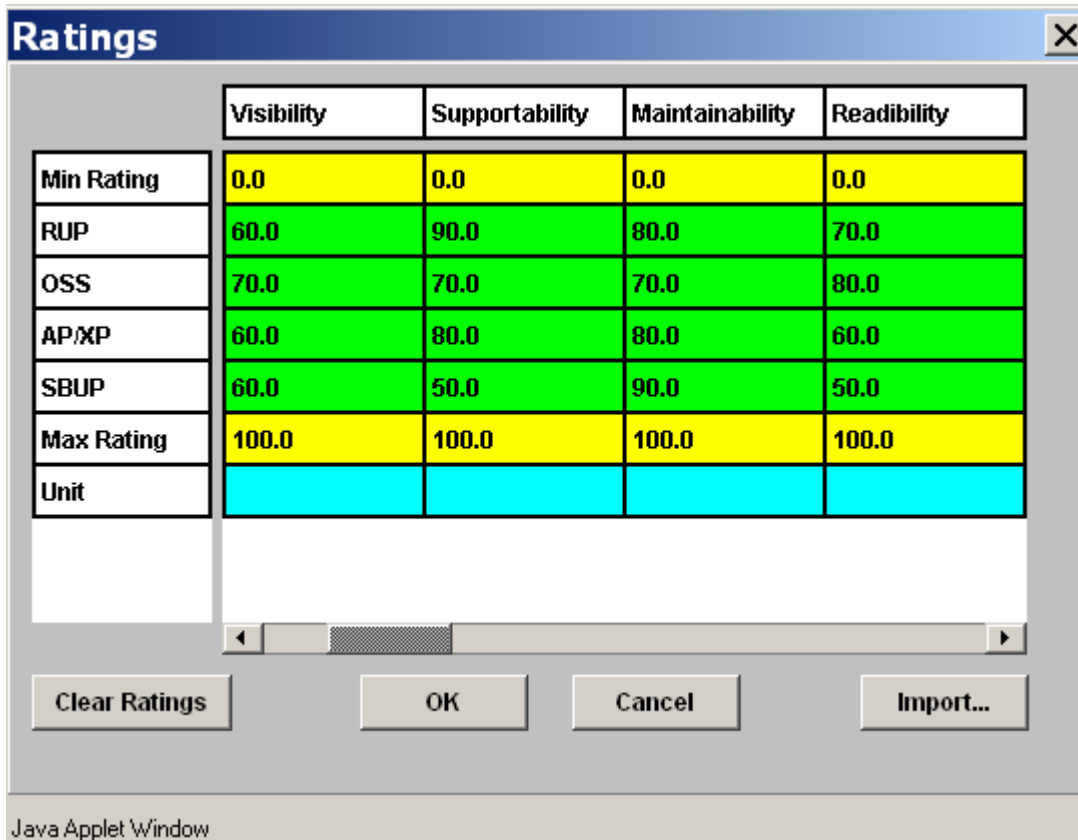


Figure 11.6: Rating of alternatives to all criteria (quality aspects)

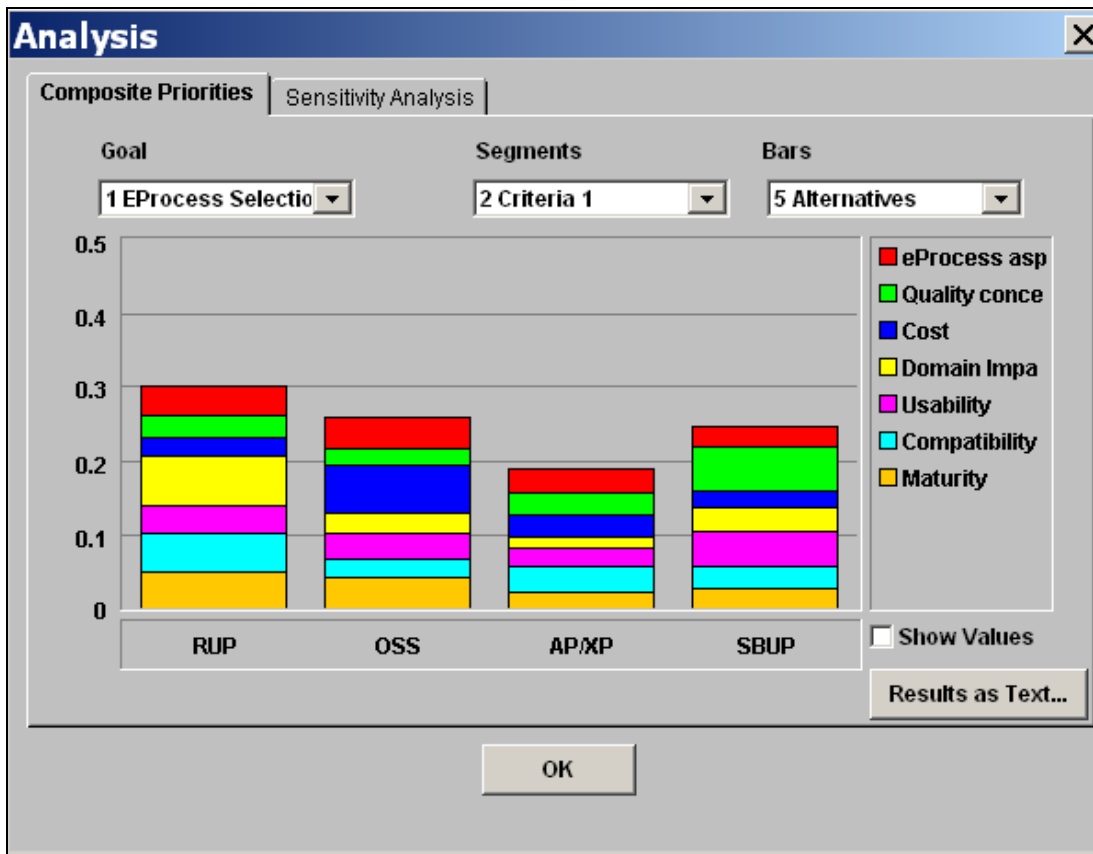


Figure 11.7: Initial Results on e-Process Selection

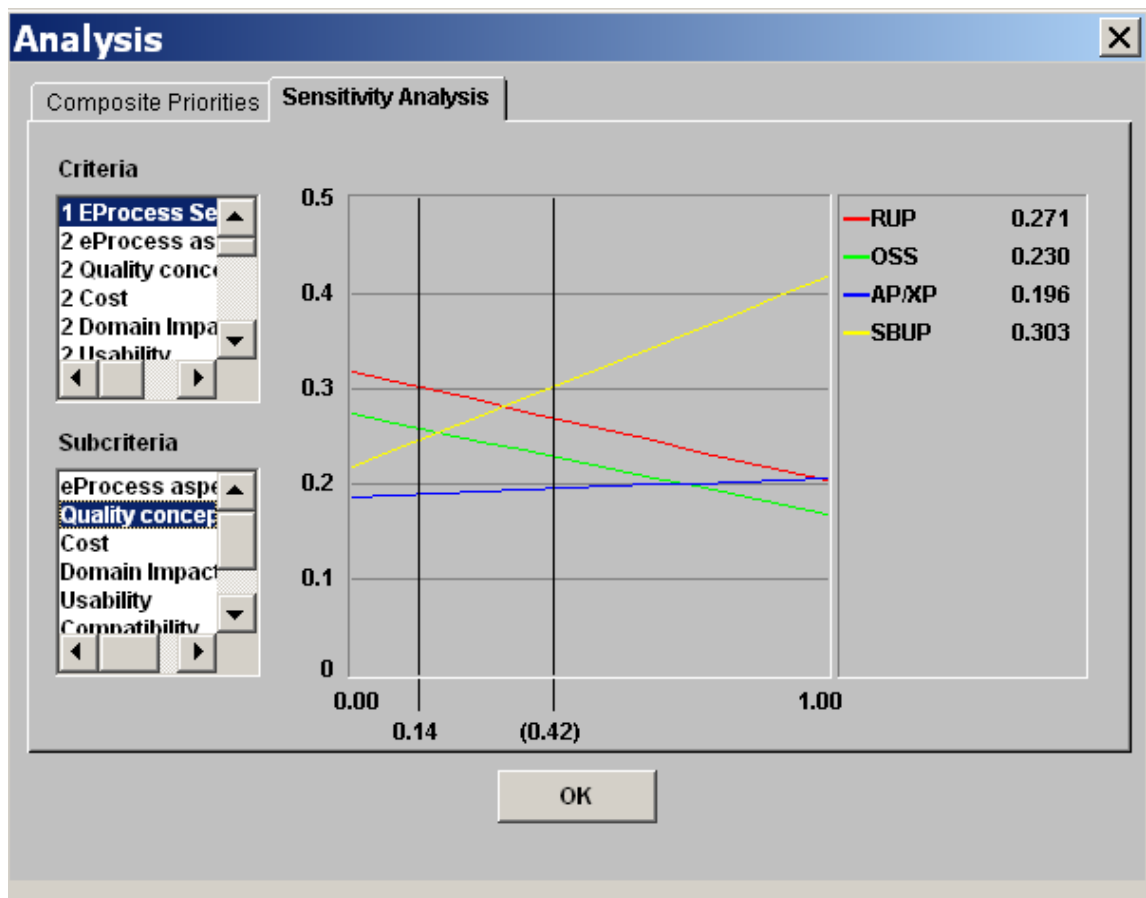


Figure 11.8: Sensitivity Analysis

11.4 Phase 3: focus on CBR

When applying CBR, the first step is to capture the information that will be used to determine similar cases. One of the screens used for the function can be seen in Figure 11.9.

The similar cases are then displayed and the user chooses their preference. See figure 11.10.

http://localhost:1617/WebSite10/Default.aspx

Google G Go Bookmarks 223 blocked Check

Untitled Page

Case Based Reasoning

High level quality aspects

The high level quality aspects are used for finding similar cases

e-Process aspects	<input type="text"/>
Quality concepts	<input type="text"/>
Domain impact	<input type="text"/>
Usability	<input type="text"/>
Compatibility	<input type="text"/>
Maturity	<input type="text"/>

Figure 11.9: Capturing the CBR information for determining similar cases

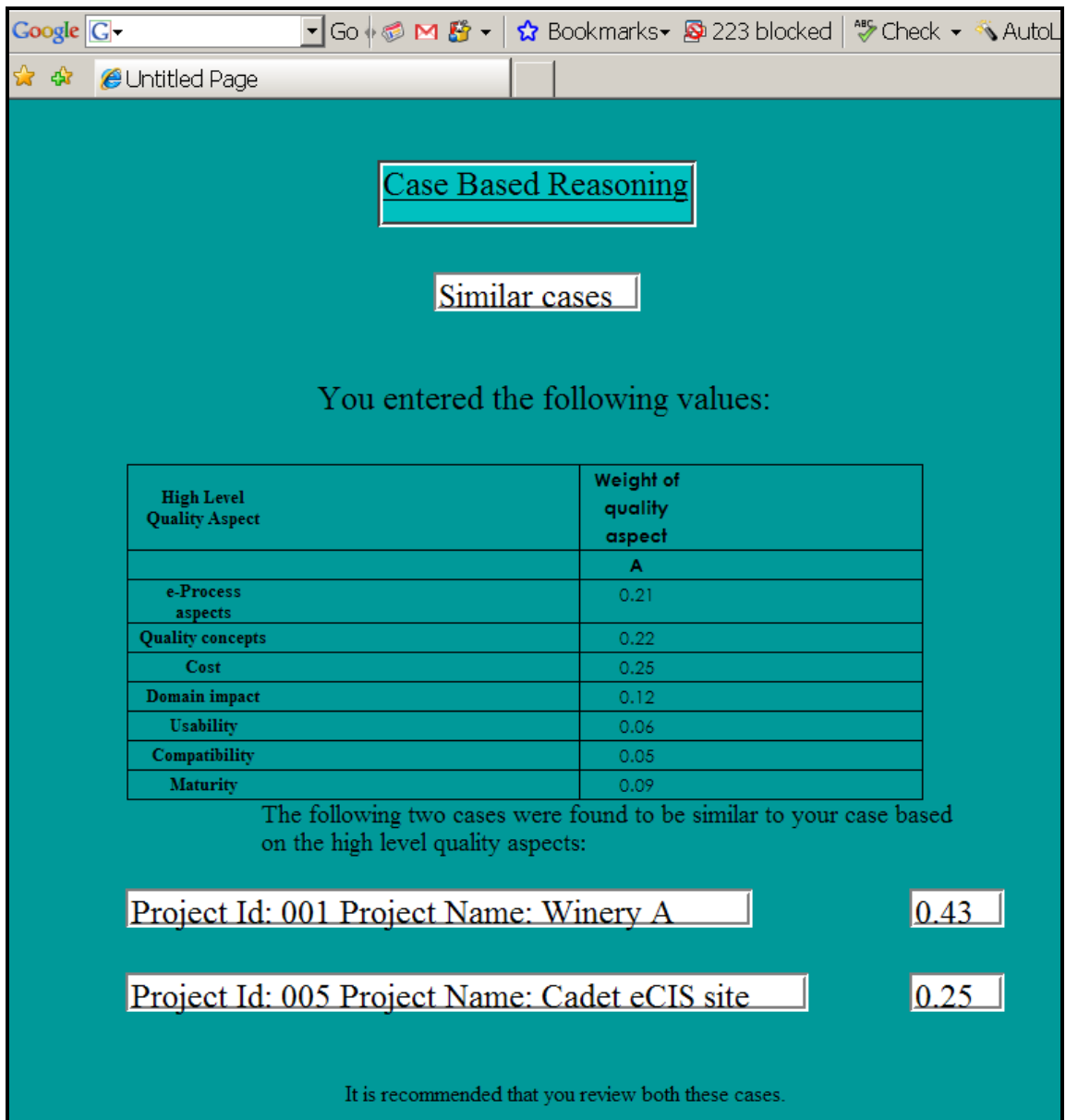


Figure 11.10 CBR Similar result

11.5 Phase 4: focus on SCM and other admin functions

The final phase of the prototype tool provided the instructions for executing SCM and capturing the results obtained from these exercises. See figure 11.11 for an example of the entry of results.

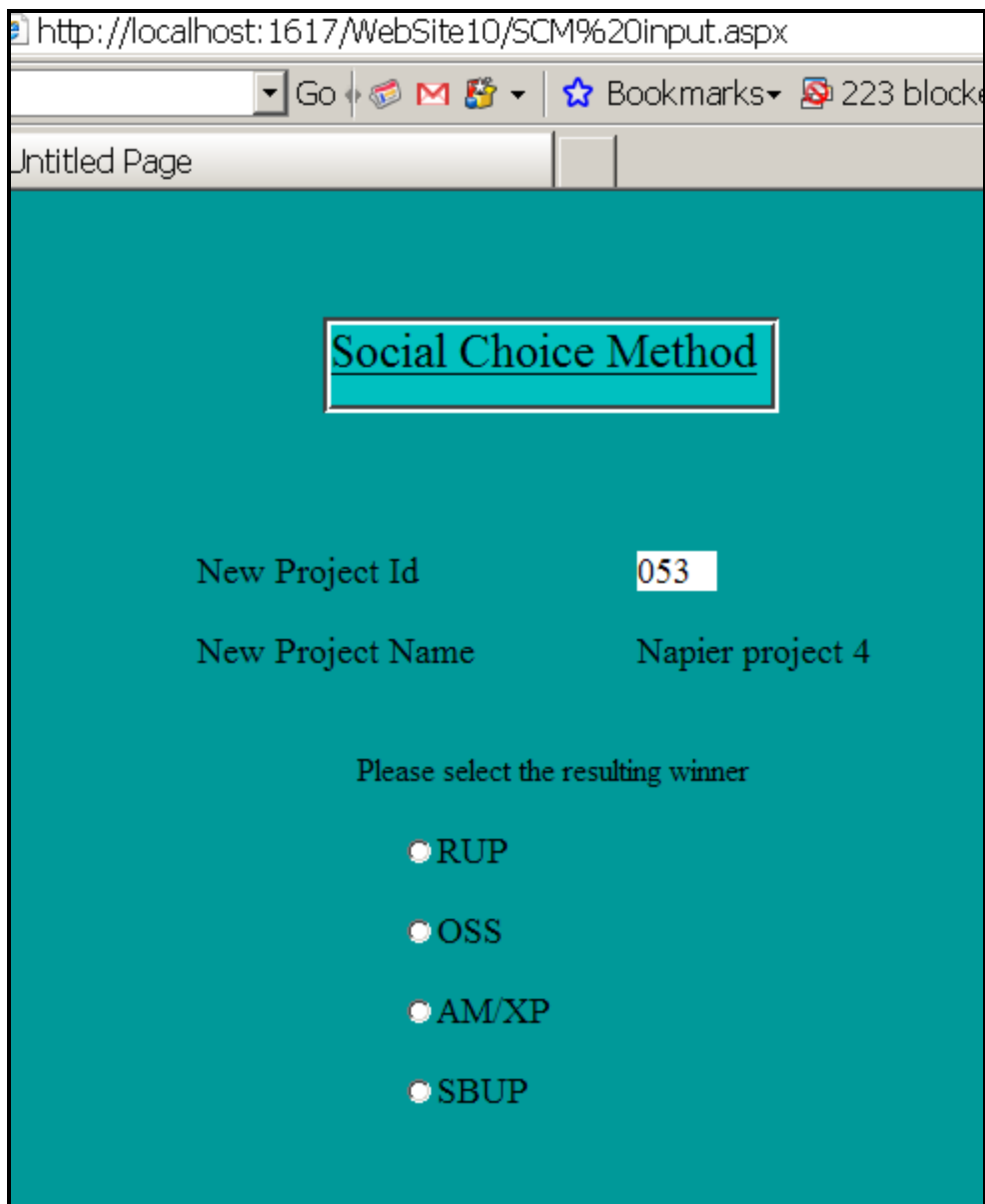


Figure 11.11 Capturing SCM results

This phase also incorporated some of the administrative functions. An admin function screen shots can be seen in Figure 11.12, on the next page.

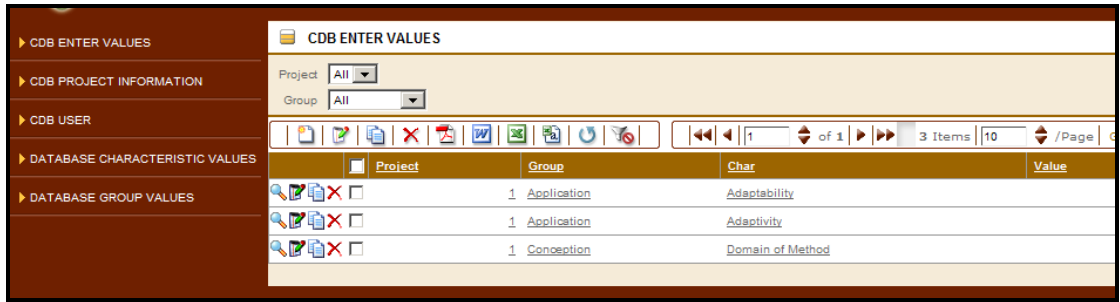


Figure 11.12: Entering Quality Aspect Information

11.6 Conclusions

The prototype tool is limited in its functionality at this stage, but it provides for some insight into the possibilities of such a tool. Although a number of different parts and versions of the e-Process selection prototype tool have been developed and used, further refinement is necessary and more feedback is required on the ease of use and viability of the tool before this prototype tool will be commercially viable.

PART E

CONCLUSIONS AND FUTURE WORK

Chapter 12

CONCLUSION AND FUTURE WORK

This is the final chapter in the thesis. Conclusions that can be drawn are summarised and an evaluation is done on what has been achieved. Also included in this chapter is some of the future development that could be explored.

12.1 Introduction

The framework for the discussion in this chapter can be seen in Figure 12.1. This chapter summarises some of the aspects from the thesis, highlights some of the strengths and weaknesses of the thesis and then discusses possible future work that could be considered.

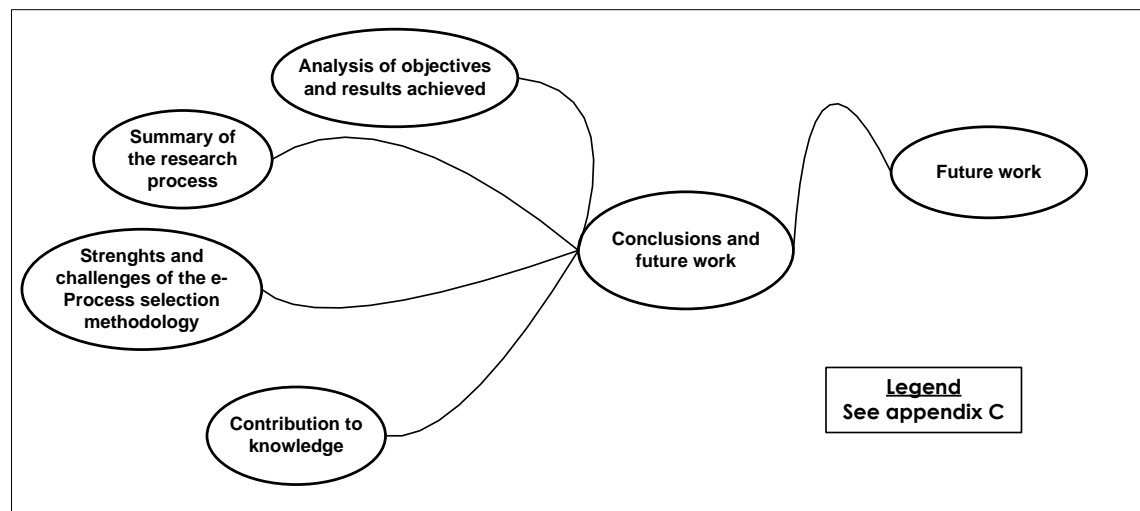


Figure 12.1: Framework for evaluation, conclusions and future work

12.2 Assessment of objectives achievement

See Table 12.1 for a discussion of the achievement of the objectives set for this thesis. A well-grounded methodology was followed to address some realistic concerns in industry. The methodology included a descriptive survey. Case study research was also done and included cross-case comparisons, the use of rich descriptions, content and face validation, triangulation and prototyping (refer to Figure 2.3 in chapter 2).

Objective	Result
<p>Primary: The primary objective of this thesis was to assist developers of eCIS to select a suitable process based on a systematic application of the appropriate decision making methods from a set of known alternatives.</p>	<p>Although this objective was only tested on 4 decision making methods and 4 possible e-Processes – chapter 7 and chapter 9 showed that this is a plausible way to select e-Processes.</p>
Sub-Objectives	Results
<p>Identifying whether development processes are actively used and how development processes are used in an industry based web development environment.</p>	<p>The research has shown that developers are using development processes, but are often unsure of which development processes to use for eCIS.</p>
<p>Applying a set of multi-criteria decision making methods on the selection process and for each performing the following three activities:</p> <ul style="list-style-type: none"> • Develop a formal meta-model and then integrate this into one the e-Process selection meta-model; • Apply the method to a specific case study in collaboration with a real software company and; • Implement the method in a prototype software tool. 	<p>The research on multi-criteria decision making methods to select a suitable e-Process has been achieved by:</p> <ul style="list-style-type: none"> • The development of a formal integrated meta-model for e-Process selection see chapter 7. This integrated model has been published and presented to the research community and also used as a basis to develop a prototype tool. • The developed theory has been applied to four different real case scenarios. • A prototype software tool has been developed.

Table 12.1 Thesis objectives and analysis of results

12.3 Summary of the Research Process

The research in this thesis was based on creating a model for e-Process Selection and has a strong theoretical foundation.

Four e-Process alternatives were used in the research and included a comparison of these alternatives. The quality aspects used were also identified and discussed. The

investigative initial case study researched the use of development processes by industry.

Four decision making methods, covered in the literature, were used in the e-Process Selection Methodology to select e-Processes. Each of the decision making method descriptions included in this thesis have a strong theoretical basis which helped to define the foundations of selection for using that specific method. Also included is an integrated meta-model for e-Process selection based on the four decision making methods. Each of the decision making methods was also applied to a case scenario. The case study used for each decision making method briefly defined the background of the problem environment, the process followed in order to execute the specific decision method, and the outcome of each execution applied on a specific case eCIS. Each of the case studies was also further validated.

Finally the methodology used included design-demonstration, where new systems or programs are constructed and tested. The research contained in this thesis included the development of a prototype for e-Process Selection Decision Making Software.

12.4 Strengths and Challenges of the e-Process Methodology

The research undertaken for this thesis focused on small-to-medium sized enterprises in the New Zealand environment. The value to these SME's using the e-Process Selection Methodology is that it allows the developer to debate the different issues around selecting a suitable development process - these issues include the speed with which the eCIS needs to be developed, the unknown profile of the client and the fact that eCIS usually work with the collection of money. It also allows the developers and other people involved with the development to discuss and think about the quality aspects of the resulting eCIS and the quality aspects required for the development process. Using the e-Process Selection Methodology helps the developers and decision makers to identify a development process and to accept the recommended development process. The Cost/Benefit analysis in chapter 9 showed that using the e-Process Selection Methodology was not expensive and did not require much time to implement.

A further strength of the research was that a number of different options have been presented and tested that can support developers in making a decision on which e-Process to use. The developers had found the e-Process Selection Methodology useful and helpful (see comments in chapter 9) and felt that they had confidence in the recommendations. A conclusion that could be drawn from the validation done by

applying more than one DMM on the same case study in chapter 9 was that the results were not a match by coincidence. Even if the developers were to use more than one e-Process (Phillips, et al., 2005) then the process of discussing the e-Process options and thinking about the choices as a team has many advantages.

The decision for all of these decision models rests on a defined set of quality aspects. Developers using the research in this thesis to support their choice will have documented proof of how a specific choice was made. This e-Process selection process is supported by a prototype tool.

The case study results were from applying different decision making methods to small sized problems. The approach has, however, been tried in four realistic situations with developers from a variety of backgrounds (face validity). The methodology proposed in the research is strongly grounded in decision making and quality standards literature (content validity).

Some of the challenges that had to be investigated included – Challenge 1: Whether the results were credible? Challenge 2: Whether an e-Process was a winner by accident and whether it is always the same winner? Challenge 3: Whether one needs to apply sensitivity analysis in all instances?

The credibility of the research and answers to the challenges mentioned above relied on methodological and data triangulation – which included multiple applications of DMM on the data obtained and analysis of the feedback from the developers.

In answer to challenge 1 - the two comparative exercises that the author of the research was able to complete showed similar results and follow up interviews with the developers indicated that they supported the recommended e-Processes. In answer to challenge two – the two comparative exercises that the author of the research was able to complete indicated that the “winner” was the same in each of these cases. If there is more than one strong candidate – will it matter? No. The methodology helps developers to make a decision and motivate their decision. In all cases the developers were happy with the results and glad for the support in the decision making process. In terms of challenge 3 – sensitivity analysis was conducted for VBA and AHP and allowed the possibility of one or more quality aspects having a substantial impact on the results to be investigated. Sensitivity analysis would have been applied to most cases captured in the knowledge base for CBR – so further sensitivity analysis was not required for this

DMM. In SCM the developers discussed their choices before voting and further sensitivity analysis was not required here.

Further challenges that could have been processed in more detail during the implementation of the e-Process selection methodology included:

- Validation on each decision model used could be explored in much more detail by following the development process and results over a period of time.
- More examples or applications on each of the decision models could have been implemented to further validate and evaluate the process.
- A better and more detailed prototype tool could possibly be developed which could then be implemented and used extensively in order to provide further evaluation of the process as well as ensuring that the knowledge base in the Case-Based Reasoning Decision model has more supporting data.

12.5 Contribution to Knowledge

In chapter two the problem of e-Process selection was described and the methodology used in the research was discussed.

The contribution in chapter three was the initial research undertaken into what type of development processes developers use. This allowed the researcher insight into what the current status quo in industry is and helped with the identification of the e-Processes used for this initial research.

In chapter four the quality aspects used for e-Process selection was identified and described – both high level quality aspects as well as second level quality aspects. This was followed in chapter five by a description of the different e-Process alternatives used. This chapter's contribution to knowledge included a comparison, based on the quality aspects in chapter 4, of the different e-Process alternatives used in the thesis.

In chapter 7, each of the decision making methods were discussed in terms of their general application and then specifically the theory of their application on e-Process selection. A meta-model was developed for each of these decision making methods. In the last part of that chapter, an integrated meta-model was developed, that can be further refined to capture the knowledge required in order to make quality decisions when selecting e-Processes using any of the decision making methods.

Contribution to knowledge in chapter eight consisted of usage recommendations for the different decision making methods and included a comparison of the DMM. This

was further expanded in chapter 9 – see Table 9.10 - by a comparison on the decision making methods based on the actual results.

Chapter 9 discussed the application of the different decision making methods used for e-Process selection. Although the different decision models have been applied on different problem scenarios for a long time this is the first time that these four decision making methods have been used to support the selection of a suitable e-Process. This contributes to knowledge by helping developers in their endeavours to make informed choices. The feasibility analysis in the chapter also showed that it is relatively inexpensive to use the selection methodology.

In chapters 10 and 11 the prototype tool was introduced and although further development was required the concept has been explored and validated.

12.6 Future development

Future research will need to further analyse quality aspects to be used in the evaluation and even whether more second level quality aspects could be incorporated for more accurate results.

Expanding and developing alternative decision making methods could also be researched, including the study of knowledge acquisition, namely, multiple classification ripple-down rules (MCRDR) (Richards & Compton, 2000).

Future research could also include making the e-Process Selection prototype tool available to developers on the internet for future use. This could then be followed up with in-depth interviews. It will be beneficial to measure the results after a while to further determine the strengths and weaknesses of using the e-Process Selection Methodology.

Making the tool available over the internet will mean that a large number of developers could potentially be targeted and their feedback incorporated into improving the methodology. As it is necessary to expand the knowledge base to such a size that enough information is available to make reasonable assumptions when selecting an e-Process using CBR – making the tool available to industry will allow for expansion of the knowledge base.

It will also be beneficial to further expand the e-Process list to include more potential (maybe currently not developed yet) e-Processes.

Appendix A

Publications by author on e-Process Selection

Albertyn, F. (2008). Decision Making and e-Process Selection. 21st Annual Conference of the National Advisory Committee on Computing Qualifications (NACCQ 2008), Auckland, New Zealand. Samuel Mann and Mike Lopez (Eds).

Albertyn, F. & Kaschek, R. (2007). *A hybrid method for E-Process selection*. 18th Australasian Conference on Information Systems (ACIS 2007). Toowoomba, Australia. (<http://www.acis2007.usq.edu.au/>).

Albertyn F. (2007). *E-Process Selection using Case-Based Reasoning Techniques*. IADIS international conference E-Society 2007. Lisbon, Portugal. (<http://www.esociety-conf.org/>) part of the IADIS Multi Conference on Computer Science and Information Systems (MCCSIS 2007) Lisbon, Portugal, 3 - 8 July 2007 (<http://www.mccsis.org>) .

Albertyn, F., Kaschek, R., Shekhovsov, V., & Zlatkin, S. (2007). *Formal Approach to e-Process Selection*. Technical Report Number: 3/2007. Massey University. Palmerston North.

Kaschek, R., Albertyn, F., Shekhovtsov, V.A., Zlatkin, S. (2007) *An e-Process Selection Model*. Twenty-Sixth International Conference on Conceptual Modelling - ER 2007 - Tutorials, Posters, Panels and Industrial Contributions, Auckland, New Zealand. 5 - 7 November 2007. *Conferences in Research and Practice in Information Technology*, Vol. 83. John Grundy, Sven Hartmann, Alberto H. F. Laender, Leszek Maciaszek and John F. Roddick, Eds.

Albertyn, F., F. Fritz et al. (2006). Towards Engineering Clinical Pathways. *The Journal of Internet Electronic Commerce Research* 2005 – Volume 12: p. 207 – 224, ISSN 1598 - 1983.

Albertyn, F. (2005). *Ontology for the Selection of e-Processes*. *Web Information Systems Engineering - WISE 2005 Workshops*, New York, eds. G. Y. Dean M., Jun W., Kaschek R., Krishnawasmy S., Pan Z., Sheng M. 1 - 10.

Albertyn, F. & Kaschek, R. (2005). *eCIS Development: Refining E-Process Selection*. IADIS International Conference - e-Society, Qawra, Malta, P. K. a. M. McPherson, 233 – 240, ISBN 972-8939-03-5.

Albertyn, E.F. & Kaschek, R.H. (2005). Deriving an assistant architecture from a tool architecture: an e-Process Study. *Perspectives of Intelligent Systems' Assistance (PISA) Technical Report*. Number: 2/2005. Department of Information Systems. Massey University. ISSN 1175-1738.

Albertyn, F. (2005). Exploring *E-Commerce Development: Lessons learnt from a Case Study*, Annual Conference of the National Advisory Committee on Computing Qualifications, Tauranga 10 – 13 July 2005, 123-126.

Albertyn, F. (2005). To use or not to use: Investigating web methodologies, *JACIT- New Zealand Journal of Applied Computing and Information Technology*, Vol. 9, No. 1; June 2005; pp. 8-13.

Albertyn, F., Fritz, F., Nguyen, P., Zlatkin, S., Tennant, A. & Kaschek, R. (2005) Towards Engineering Clinical Pathways. Proceedings of the KIECA (Korea Internet E-Commerce Association) International conference 2005 on Trade, Investment, Logistics and e-Buz, Kunsan, The Republic of Korea, September 30-October 2 2005, pp. 241 - 250

Albertyn, F. (2004), To use or not to use: Investigating web methodologies, *Seventeenth Annual Conference of the National Advisory Committee on Computing Qualifications*, Christchurch, 201 – 204.

Albertyn, F. & Kaschek, R. (2004). Ontology Based E-Process Selection. *The Sixth International Conference on Information Integration and Web-based Applications & Services (iiWAS2004)*, Jakarta, Indonesia, Austrian Computer Society, 143 – 152. eds. G. Kotsis, S.Bressan, D. Taniar and I. K. Ibrahim. ISBN. 3-85403-183-01.

Albertyn, F. & Zlatkin, S. (2004). *The process of developing a business processes assembler"*, ISTA 2004: 3rd International Conference on Information Systems Technology and Its Applications, Salt Lake City, Utah, USA July 15-17, 165 – 176, ISSN 1617-5468.

Albertyn, F. (2004). E-Process Selection Methodology. *Doctoral Consortium ACIS2004 (Australian Conference on Information Systems)* Hobart, Tasmania.

Appendix B

Technology Specifications

IBM ThinkPad laptop:

Intel(R) Pentium® M
Processor 1.70 GHz
798 MHz 512 MB of RAM

Operating System:

Microsoft Windows XP Professional Version 2002 Service Pack 2

Software for thesis production:

Internet Explorer
Microsoft Office Word 2003
Microsoft Office Visio 2003
Microsoft Office Excel 2003
EndNote 9

Software used for prototype development:

Visual Basic 6
Microsoft Access 2003
PHP and MySQL
Visual Studio 2005
Using C#
Using Web page development ASP
Choice Results

Word page setup:

Font type:

Century Gothic 10.




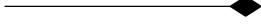
Page setup:

Pages are printed on both sides. One and a half line spacing is applied, with margins of 4 cm on the inner side (to allow for the binding), 2.5 cm at top and bottom and 2 cm on the outer side. Font size is 10 points for normal text.


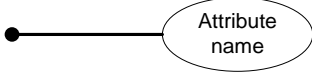


Appendix C

Legends for different types of figures used in thesis

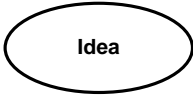
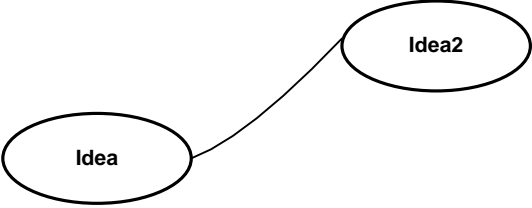
UML Class diagrams:

UML Class Diagram Legends	
<p>Boxes</p> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <p style="text-align: center; margin: 0;">Voting</p> <hr style="border: 0.5px solid black;"/> <p style="margin: 0;">-VoterID -EProcessID -Reason -Weight -Voting Procedure followed</p> </div>	<p>Boxes represent classes of objects. The class names appear at the top of the box. Attributes, if shown, appear below the class name. The class name and attributes are separated by a line. The third part represents the different operations or methods of that class.</p>
<p>Lines</p> <div style="margin: 5px 0;"> <p style="text-align: center;">0..* 1</p>  </div>	<p>Lines represent possible relationships between objects of two classes. Objects of the class on one end of the line can be "associated" of the class on the other end of the line.</p>
<p>Filled diamonds</p> <div style="margin: 5px 0;"> <p style="text-align: center;">0..* 1</p>  </div>	<p>Filled diamonds on the end of a line indicate containment by value. Objects of the class on the other end of the line are part of one and only one object of the class the diamond touches</p>
<p>Generalisation</p> <div style="text-align: center; margin: 5px 0;">  </div>	<p>Open triangle on the end of a line indicates this is a super class. Objects at other end of the line can be thought of as subgroups.</p>
<p>Cardinality</p> <div style="margin: 5px 0;"> <p style="text-align: center;">0..* 1</p>  </div>	<p>1 : one and only one 0..1 : zero or one 0..n : zero or more 1..n : one or more</p>

Entity Relationship Diagrams:

Entity Relationship Diagram Legends	
<p>Entity</p> 	<p>An entity represents a collection of objects or things in the real world</p>
<p>Attributes</p> 	<p>Attributes express the properties of the entities.</p>
<p>Relationships</p> 	<p>Relationships describe the association between entities.</p>
<p>Cardinality</p> 	<p>Cardinality is of three types: one-to-one, one-to-many, many-to-many</p>

Mind maps:

Mindmap-description and Diagram Legends	
<p>Mindmapping can be used to help maximise your thought process and easily visualise them. Mindmapping is the process that helps to map your logical way of thinking and link the different ideas to be discussed together.</p>	
	<p>This represents the idea.</p>
	<p>The line links two ideas and shows how the thought processes can be linked</p>

Appendix D

Methodological Survey into Web Development Usage

Contact detail (optional)

Name: _____ Company/Organisation (optional): _____

Contact email (optional): _____

1. Mark the types of development you are involved in:

- Developing web pages
- Developing large interactive websites
- Developing e-commerce sites
- Setting up e-commerce sites using predefined software
- None of the above

2. How will you classify your work environment?

- Single person development
- Private small company
- Small-to-medium sized business
- Larger company

3. Do you usually do the development as part of a team?

Yes No

4. If yes to 3, what is the size of an average development team in your environment?

- 2 – 3 three people
- 4 – 10 people
- More than 10

5. Do you use a structured development process?

Yes No

6. Mark the development process/es that you use:

- Rational Unified Software
- Open source software
- Agile modelling
- Storyboarding
- User profiling
- Other

7. If you marked Other in 6, please expand/list:

8. Please mark any of the following that you use:

- | | |
|---|---|
| <input type="checkbox"/> Formal data modelling | <input type="checkbox"/> Time management techniques |
| <input type="checkbox"/> Entity relationship diagrams | <input type="checkbox"/> Case Tools |
| <input type="checkbox"/> User profiling | <input type="checkbox"/> Storyboarding |
| <input type="checkbox"/> Detailed storyboards | <input type="checkbox"/> Agile Methods |
| <input type="checkbox"/> UML diagrams | <input type="checkbox"/> Formal interviews |
| <input type="checkbox"/> Project management | <input type="checkbox"/> Research into other sites |

9. Types of companies you have done development work for?

- | | |
|---|---|
| <input type="checkbox"/> Private small company | <input type="checkbox"/> Primary or High School |
| <input type="checkbox"/> Health sector | <input type="checkbox"/> Tertiary Institutions |
| <input type="checkbox"/> Small-to-medium sized business | <input type="checkbox"/> Government |
| <input type="checkbox"/> Larger company | <input type="checkbox"/> Other – Specify: _____ |

Any further comments that you think might be relevant to the research:

References:

- Aczel, J. & Saaty T.L. (1983). Procedures for Synthesizing Ratio Judgments. *Journal of Mathematical Psychology*, 27, 93-102.
- Ahituv, N. & Neumann, S. (1990). *Principles of information systems for management* (3rd ed.) Wm. C. Brown Publishers.
- Albertyn, F. & Kaschek, R. (2007). *A hybrid method for E-Process selection*. 18th Australasian Conference on Information Systems (ACIS 2007). Toowoomba, Australia. (<http://www.acis2007.usq.edu.au/>).
- Albertyn F. (2007). *E-Process Selection using Case-Based Reasoning Techniques*. IADIS international conference E-Society 2007. Lisbon, Portugal. (<http://www.esociety-conf.org/>) part of the IADIS Multi Conference on Computer Science and Information Systems (MCCSIS 2007) Lisbon, Portugal, 3 - 8 July 2007 (<http://www.mccsis.org>).
- Albertyn, F. (2005a). To use or not to use: Investigating web methodologies, *JACIT- New Zealand Journal of Applied Computing and Information Technology*, Vol. 9, No. 1; June 2005; 8-13.
- Albertyn, F. (2005b). *Exploring E-Commerce Development: Lessons learnt from a Case Study*, Annual Conference of the National Advisory Committee on Computing Qualifications, Tauranga, 10 – 13 July 2005, 123-126.
- Albertyn, F. & Kaschek, R. (2005c). *eCIS Development: Refining E-Process Selection*. IADIS International Conference - e-Society, Qawra, Malta, P. K. a. M. McPherson, 233 – 240, ISBN 972-8939-03-5.
- Albertyn, F. (2005d). *Ontology for the Selection of e-Processes*. Web Information Systems Engineering - WISE 2005 Workshops, New York, eds. G. Y. Dean M., Jun W., Kaschek R., Krishnawasmy S., Pan Z., Sheng M. 1 - 10.
- Albertyn, F. (2004a), *To use or not to use: Investigating web methodologies*, Seventeenth Annual Conference of the National Advisory Committee on Computing Qualifications, Christchurch, 201 – 204.
- Albertyn, F. & Kaschek, R. (2004b). *Ontology Based E-Process Selection*. The Sixth International Conference on Information Integration and Web-based Applications & Services (iiWAS2004), Jakarta, Indonesia, Austrian Computer Society, 143 – 152. eds. G. Kotsis, S.Bressan, D. Taniar and I. K. Ibrahim. ISBN. 3-85403-183-01.
- Al-Humaidan, F. & Rossiter, B. N. (2000). *A taxonomy and Evaluation for System Analysis Methodologies in a Workflow Context: SSADM, UML, Unified Process, SSM and Organization Process Modelling*. Retrieved October, 2003 from www.cs.ncl.ac.uk/research/pubs/trs/papers/751.pdf.

Al-Naeem, T., Gorton, I., Babar, M.A., Rabhi, F. & Benatallah, B. (2005). *A Quality-Driven Systematic Approach for Architecting Distributed Software Applications*. Proceedings of the 27th International Conference on Software Engineering (ICSE), St. Louis, USA.

AM (2001). *The Agile Modelling site*. Retrieved in September 2003 from www.agilemodeling.com.

Ambler, S. W. (2000). *Extreme modeling - Design is fundamental to the XP process, regardless of what the name may imply*.

Ambler, S. W. (2002). *Agile Modeling (AM) An overview*. Ronin International. Retrieved from www.ronin-intl.com.

Amber, S. W. (2006) *Choose the Right Software Method for the Job*. Retrieved June, 2006 from <http://www.agiledata.org/essays/differentStrategies.html#-Comparing Methodologies>.

Analytic Hierarchy Process (AHP) example (2008) Retrieved October, 2008 from <http://courses.washington.edu/samcrs/ahpexample.pdf>.

Antón, A. I., Carter, R. A., Earp, J. B., & Williams, L. A. (2001) *EPRAM: Evolutionary Prototyping Risk Analysis & Mitigation (e-Commerce Software Development Process Document)* [Electronic Version]. Software Requirements Engineering Lab, North Carolina State University, Department of Computer Science, Technical Report #TR-2001-08 Retrieved in 2004 from www.csc.ncsu.edu/faculty/anton/pubs/epramTechReport.pdf.

Armengol, E., Ontañón, S., & Plaza, E. (2004). *Explaining Similarity in CBR*. Paper presented at the ECCBR'04, Madrid, Spain.

Arrow, K. J. (1963). *Social Choice and Individual Values*. ISBN 0-300-01364-7.

Astels, D. Miller, G. & Novak, M. (2002). *A practical guide to eXtreme programming*. Upper Saddle River, NJ, USA. Prentice Hall.

Australian_Government ICT_strategy (2004) *User Profiling and Testing Toolkit*. Retrieved in January, 2004 from www.agimo.gov.au/publications/2004/06/toolkit/user/concepts_

Barbacci, M., Longstaff, T. M., Klein, M.H., & Weinstock, C.B. (1995). *Software Quality Attributes*. Carnegie Mellon University: Pittsburgh, Pennsylvania. 1-53.

Bamford, R., Deibler, W. (2003). *ISO 9001: 2000 for Software and Systems Providers: An Engineering Approach* (1st ed.). CRC-Press.

Bauer, M. (2005) *Successful Web Development Methodologies*. Retrieved from <http://www.sitepoint.com/article/successful-development/1>.

Baxter, L., Hughes, C. and Tight, M. (2006). *How to Research*. Maidenhead, Open University Press.

Bean Software (2007). *Three Tier Architecture in ASP.NET*. Retrieved in October 2007 from <http://www.beansoftware.com/ASP.NET-Tutorials/Three-Tier-Architecture.aspx>

Benbasat, I., Goldstein, D.K. and Mead, K. (1987), The case research strategy in studies of information systems, *MIS Quarterly*, September 1987, 369-386.

Bernardo, J.M. (2006). A Bayesian Mathematical Statistics Primer. International Conference for Teaching Statistics (ICOTS 7). Brazil.

Bernroider, E. W. N., & Mitlohner, J. (2007). *Resolving Indifferences in Multiple Attribute Business Information System Selection with Social Choice Methods*. 6th International Conference ISTA 2007, Kharkiv, Ukraine.

Bernus, P. & Schmidt, G. (1998). *Architecture of Information Systems*. Handbook on Architectures and Information Systems. Eds. P. Bernus, K. Mertins and G. Schmidt, Springer Verlag, Berlin et al. P. 1-9.

Binemann-Zdanowicz, A. Kaschek, R., Schewe, K-D and Thalheim, B. (2004) Context-Aware Web Information Systems. *First Asia-Pacific Conference on Conceptual Modelling (APCCM 2004)*, Dunedin, New Zealand. Conferences in Research and Practice in Information Technology, Vol. 31. Sven Hartmann, John Roddick, (Eds).

Bird, S.J. (2003) Allocating resources in a global community. *Science and Engineering Ethics*. Volume 9 No. 3 Springer.

Birney, G. and N. Moreland (1999). *A practical guide to academic research*. London, Kogan page limited.

Black, D. (1958). *The theory of committees and elections*. Boston: Kluwer.

Bloomberg, J. (2000). *Building E-businesses with the Rational Unified Process*.

Biricik, G., Buharali, A. and Kalipsiz, O. (2003). *The Impact of Object Oriented Development Methods on Software Quality*. Impact of Software Process on Quality Workshop Proceedings, Bilkent Üniversitesi, Ankara, 6 Haziran. Available from <http://www.cs.bilkent.edu.tr/improq03/Papers/Biricik.pdf>.

Boger, M. b. M. (2002). *Welcome to xTremeModeling.org*, XM - Extreme Modeling. Retrieved on March, 2004 from www.extrememodeling.org.

Böhm, R. & Wenger, S. (1996). *Methoden und Techniken der System-Entwicklung*. Zürich, vdf Hochschulverlag AG and der ETH. 2nd. ed.

Borda count. (2007). In *Wikipedia, The Free Encyclopedia*. Retrieved March 2008, from http://en.wikipedia.org/w/index.php?title=Borda_count&oldid=344535169.

Bouyssou, D., Marchant, T., Pirlot, M., Tsoukas, A., and Vincke, P. (2006) *Evaluation and decision models with multiple criteria – Stepping stones for the analyst*. Springer International Series.

Brynjolfsson, E. (2003). *The IT Productivity Gap*. Optimize, July 2003.

Burns, R.B. (2000). *Introduction to research methods*. 4th ed. Sage Publications. London.

Burt, J. (2001). *Rational spreads wealth of process tools with RUP*. eWeek 12/3/2001 18(47): 24.

Butler, T. (2003). *An institutional perspective in developing and implementing intranet- and internet-based information systems*. *Info Systems Journal* 2003(13): 209-231.

Cartwright, D. (2008). DIY User Profiling. *Web Developer's Journal*. Retrieved in February 2008 from [www.webdevelopersjournal.com/articles /user_profiling_diy.html](http://www.webdevelopersjournal.com/articles/user_profiling_diy.html).

Cavana, R.Y., Delahay, B.L. & Sekaran, U. (2001) *Applied Business Research: Qualitative and quantitative methods*. John Wiley and Sons, Australia.

Challam, V., Gauch, S., & Chandramouli, A. (2007). *Contextual Search Using Ontology-Based User Profiles*. Paper presented at the Conference RIAO2007, Pittsburgh PA, U.S.A.

Chen, S.J. and Hwang, C.L. (1991) Fuzzy Multiple Attribute Decision Making: Methods and Applications. *Lecture Notes in Economics and Mathematical Systems*. No. 375, Springer-Verlag, Berlin, Germany.

Cheung, W. (1998). The use of the world wide web for commercial purposes. *Industrial Management and Data systems*, 1998(3/4), 172-178.

Chrissis, M.B., Konrad, M. and Shrum, S. (2006) *CMMI: Guidelines for Process Integration and Product Improvement*, second ed. Addison-Wesley/Pearson Education, Boston, MA.

Clemens, P. L. (2002). *Weighted Scoring Decision Making*. Jacobs Sverdrup.

Coyle, L., Doyle, D., & Cunningham, P. (2004). *Representing similarity for CBR in XML*. Paper presented at the Advances in case-based reasoning ECCBR 2004 : European conference on case-based reasoning No7, Madrid, ESPAGNE. ISBN 3-540-22882-9.

Creswell, J. W. (1994). *Research design: Qualitative & quantitative approaches*. Thousand Oaks, CA: Sage Publications.

CS-TR 751, Department of Computing Science, University of Newcastle (2000). Retrieved in March 2007 from [www.cs.ncl.ac.uk/research/pubs/trs/papers /751.pdf](http://www.cs.ncl.ac.uk/research/pubs/trs/papers/751.pdf).

Cunningham, P. and Bonzano A. (1999). *Knowledge engineering in a real world casebased reasoning application*. Technical Report TCD-CS-1999-36, Trinity College, Dublin.

Dalgleish, S. (2005). Probing the Limits: ISO 9001 Proves Ineffective. *Quality Magazine* April 1, 2005.

Davidson, J.E. (2005). *Evaluation Methodology Basis: The nuts and bolts of sound evaluation*. Sage Publications. California. ISBN 0-7619-2929-0.

Denzin, N. (2006). *Sociological Methods: A Sourcebook*. Aldine Transaction. ISBN 9780-202308401. (5th edition).

Dey, I (1993) *Qualitative data analysis: a user-friendly guide for social scientists*. London: New York, NY : Routledge.

Diaz-Agudo, B. and Gonzalez-Calero P. A. (2000). *An architecture for Knowledge Intensive CBR systems*. Retrieved in April 2005 from www.idi.ntnu.no/~agnar/tdt55/2000_ewcbr_belen-2.pdf.

Doumpos, M., & Zopounidis, C. (2002). *Multicriteria Decision Aid Classification Methods*. Dordrecht: Kluwer Academic Publishers. 1-4020-0805-8.

Dubé, L., and Paré, G. (2003). Rigor in Information Systems Positivist Case Research: Current Practices, Trends, and Recommendations, *MIS Quarterly* (27:4) 2003, pp 597-636.

Dumke, R. R. and Foltin E. (1996). *Metrics-based Evaluation of Object-Oriented Software Development Methods*. Retrieved in March 2007 from ivs.cs.uni-magdemurg.de/sw-eng/agruppe/forshung/paper/ooevel.pdf.

Eisenhardt, K. M. (1989). Building Theories from Case Study Research. *Academy of Management. The Academy of Management Review* 14(4): 532-550.

Elmasri, R. and Navathe, S.B. (1989). *Fundamentals of Database Systems*. California. The Benjamin/Cummings Publishing Company, Inc. 0-8053-0244-1.

Feller, J. and Fitzgerald, B. (2000). *A framework analysis of the open source software development paradigm*. Proceedings of the 21st Annual International Conference on Information Systems (ICIS 2000), Brisbane, Australia.

Fettke, P. and Loos, P. (2003). *Ontological evaluation of reference models using the Bunge-Wand-Weber model*. Ninth Americas Conference on Information Systems.

Fielding, N and Fielding, J. (1986) *Linking Data: the articulation of qualitative and quantitative methods in social research*. Sage, London and Beverly Hills.

Flick, U. (2006). *An introduction to Qualitative research*. 3rd. Ed. Sage Publications.

Forman, E. H., & Selly, M. A. (2001). *Decision By Objectives: How to convince others you are right*. World Scientific Press. Retrieved on 3 March 2008 from <http://mdm.gwu.edu/forman/DBO.pdf>.

French, S., & Xu, D.L. (2005). Comparison Study of Multi-attribute Decision Analytic Software. *Journal of Multi-Criteria Decision Analysis*, 13, pp. 65-80.

Gadomski, A.M. (2003) *Socio-Cognitive Engineering Foundations and Applications: From Humans to Nations*. *Proceedings of the Socio-Cognitive Engineering Foundations and Applications Conference*. Rome.

Gaertner, W. (2001). *Domain conditions in social choice theory*. Cambridge, UK: Press syndicate of the University of Cambridge. 0-521-79102-2.

Ghezzi, C., Jazyeri, M., & Mandrioli, D. (2004). *Software Qualities and Principles*. Chapter 101 of *Tucker, A.B. (editor), Computer Science Handbook*. Chapman & Hall/CRC.

Gilbert, V.K. (1981). *The Case Study as a Research Methodology: Difficulties and Advantages of Integrating the Positivistic, Phenomenological and Grounded Theory Approaches*. Paper presented at the *Annual Meeting of the Canadian Association for the Study of Educational Administration* (Halifax, Nova Scotia, Canada, June 1-4, 1981). Retrieved on 20 January 2010 from http://www.eric.ed.gov/ERICWebPortal/custom/portlets/recordDetails/detailmini.jsp?_nfpb=true&_ERICExtSearch_SearchValue_0=ED218750&ERICExtSearch_SearchType_0=no&accno=ED218750

Graham, T. (2001). *Introducing the Rational Unified Process*, IEEE Computer Society. Retrieved August 2002 from [www.computer.org/software/bookshelf /2001/5501bks_2.htm](http://www.computer.org/software/bookshelf/2001/5501bks_2.htm).

Greenfield, T. (ed.) (2002). *Research Methods for Postgraduates*. 2nd. Ed. Arnold Publishers.

Groves, L., Nickson, R., Reeve, G. Reeves, S. & Utting, M. (2000). *A survey of software requirements specification practices in the New Zealand software industry*. Proceedings ASWEC 2000: Australian Software Engineering Conference 2000. Grant D.D. (ed.)IEEE Computer Society. Pp.189-201.

Gruhn, V., & Schope, L. (2002). Software processes for the development of electronic commerce systems. *Information and Software Technology*, 44(2002), 891 – 901.

Gunther, R. E. (2008). *The Truth about making Smart Decisions*. F.T.Press. 0-13-235463-2. p. 208. (Accessed through Safari Books May 2008).

Hämäläinen, R.P. (2008) *Web Hibre – Global Decision Support Software*. Copyright 1998-2007 Systems Analysis Laboratory, Helsinki University of Technology. All rights reserved. Accessed on web in October 2008 at <http://www.hipre.hut.fi/>.

Hansson, S.O. (1994). *Decision Theory – A brief Introduction*. Retrieved on 20 January 2010 from www.infra.kth.se/~soh/decisiontheory.pdf.

Harris, R. (2009) *Introduction to decision making*. Virtual Salt. Retrieved on 20 January 2010 from <http://www.virtualsalt.com/crebook5.htm>.

Harrison, I. (1997). *Case Based Reasoning*. Inductive Solutions. Retrieved in August 2006 from <http://www.aiai.ed.ac.uk/links/cbr.html>.

Hesse, W. (2002). *DinoSaur Meets Archeaeopteryx? Seven Thesis on Rational's Unified Process (RUP)*.

Hirschheim, R., Klein, H. K., & Lyytinen, K. *Information systems development and data modelling, conceptual and philosophical foundations* (Vol. 1995): Cambridge University Press; Cambridge.

Hogarth, M. *A framework for Open Development of Terminologies*. Retrieved in September 2005 from [mycin.ucdvis.edu/presentations /opendevterminologies.ppt](http://mycin.ucdvis.edu/presentations/opendevterminologies.ppt).

ISO-Standard. (1991). *Information Technology - Software product evaluation - Quality characteristics and guidelines for their use: International Standard 9126. 1991 (E)*. SO/IEC Copyright Office,Case Postal 56,CH-1211, Geneva, Switzerland.

Janssen, W. & Steen, M. (2003). *Rapid Service Development: an integral approach to e-business engineering*, Telematic Institute. Retrieved November 2003 from doc.telin.nl/dscgi/ds.py/Get/File-9189/jansse.pd.

Jayaswal, B. K., & Patton, P. C. (2006). *Design for trustworthy software*. Tools, techniques and methodology of developing robust software. United States: Prentice Hall Pearson education.

Kaschek, R. & Mayr, H.C. (1996). *A Characterization of OOA Tools*. Proceedings of the Fourth International Symposium on Assessment of Software Tools (SAST '96).

Kaschek, R., & Mayr, H. C. (1998). Characteristics of Object Oriented Modeling Methods. *EMISA Forum*, 1998(2), 10-39.

Kaschek, R., Schewe, K.-D., Wallace, C., & Mathews, C. (2004). Story Boarding for Web Based Information Systems. Chapter 1. In *Taniar D., Wenny Rahayu J.: Web Information Systems*. (Vol. 2004): IDEA Group Publishing. Hershey, PA.

Kaschek, R., Pavlov, R., Shekhovtsov, V. A., & Zlatkin, S. (2006). *Towards Selecting among Business Process Modeling Methodologies*. 9th International Conference on Business Information Systems, BIS 2006, Klagenfurt, Austria.

Kaschek, R., Albertyn, F., Shekhovtsov, V.A., Zlatkin, S. (2007) An e-Process Selection Model. *Twenty-Sixth International Conference on Conceptual Modeling - ER 2007 - Tutorials, Posters, Panels and Industrial Contributions*, Auckland, New Zealand. 5 - 7 November 2007. *Conferences in Research and Practice in Information Technology*, Vol. 83. John Grundy, Sven Hartmann, Alberto H. F. Laender, Leszek Maciaszek and John F. Roddick, Eds.

Kautz, K., & Nielsen, P. A. (2004). Understanding the implementation of software process improvement innovations in software organisations. *Info Systems Journal*, 2004(14), 3-22.

Keeney, R. L. (1992). *Value-Focused Thinking: A path to creative decision making*. Cambridge, M.A.: Harvard University Press.

Kelly, A. (2003). *Decision Making using Game Theory – An introduction for managers*. Cambridge University Press, Cambridge.

Kent, B., Beedle, M., van Bennekum, A, Cockburn, A, et al. (2001). *Manifesto for Agile Software Development*. Retrieved in August 2005 from <http://www.agilemanifesto.org/>.

Knight, L. V., Steinbach, T. A., & Kellen, V. (2001) *System Development Methodologies for Web-Enabled E-Business: A Customization Paradigm*. Retrieved in February 2003 from www.kellen.net/SysDev.htm.

Kumar, R. (1996). *Research Methodology – A step by step guide for beginners*. Addison-Wesley-Longman. ISBN. 0-582-80125-7.

Kruchten, P. (2001). *What is the Rational Unified Process?, The Rational Edge*. Retrieved in July 2004 from www.therationaledge.com/content/jan_01/f_rup_pk.html.

Kroll, P., Philippe Kruchten (2003). *Rational Unified Process Made Easy: A Practitioner's Guide to the RUP*. Addison Wesley Professional 13. 978-0-321-16609-8.

Larman, C. (2002). *Applying UML and Patterns*. Upper Saddle River, Prentice Hall PTR.

Leake, D. B. (1996). *CBR in context: the present and future. Case Based reasoning experiences - Lessons and future experiences*. D. Leake. Cambridge, MIT Press.

Leake, D. B. (1996). *What Case-Based Reasoning Really Means: Now and the Future*. Retrieved in May 2004 from <http://www.cis.ufl.edu/~ddd/FLAIRS/FLAIRS-96/leake.htm>.

- Leary, M.R. (2004). *Introduction to behavioural research methods*. 4th ed. Boston, MA : Pearson/Allyn and Bacon, 2003, c2004.
- Lee, A.S. (1989). A Scientific Methodology for MIS Case Studies. *MIS Quarterly* (13:1), 1989, pp. 33-52.
- Li, Z., Allaeddine, N. and Tian, J. (2010) Multi-faceted quality and defect measurement for web software and source contents. *Journal of Systems and Software*. Volume 83, Issue 1, p.18 – 28. Elsevier.
- Macaya, D., J. Melendez et al. (2002). *Case Based approach for supervision. Application to PID controllers*. 15th-IFAC World Congress, Barcelona
- MacCormack, A. M., Kemerer, C. F., Cusumano, M., & Crandall, B. (2003). Trade-offs between Productivity and Quality in Selecting Software Development Practices. *IEEE Software*, September/October 2003, 78 – 85.
- Macdonald, N.J. & Cheng, D.C.B. (1997) *Basic finance for Marketeers*. FAO of the United Nations. Rome. ISBN 92-5-103965-8. Retrieved on 10 October 2009 from <http://www.fao.org/docrep/W4343E/w4343e06.htm>.
- Mansar, S. L., Marir, F., & Reijers, H. (2003) Case-Based Reasoning as a Technique for Knowledge Management in Business Process Redesign *Academic Conferences Limited 2003* Retrieved in June 2003 from www.ejkm.com.
- Mansar, S. L., Reijers, H. A., & Ounnar, F. (2005). *BPR Implementation: a Decision-Making Strategy*. Proceedings of the Workshop on Business Process Design - 2005.
- Mantel, S. J., J. R. Meredith, et al. (2005). *Core Concepts: Project Management in Practice*, John Wiley & Sons.
- Marshall, C., & Rossman, G. (1980). *Designing qualitative research*. Newbury Park, CA: Sage.
- Mattila, S. (1998). *Secure Electronic Transaction Terms*. Retrieved in September 2004 from http://www.its.bldrdoc.gov/projects/devglossary/_e-commerce.html.
- Medvidovic, N., Rosenblum, D. S., Redmiles, D. F., & Robbin, J. E. (2002). Modeling software architectures in the Unified Modeling Language. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 11 (1), 2-57. 1049-331X.
- Merriam, S. B. (1988). *Case study research in education: A qualitative approach*. San Francisco: Jossey-Bass.
- Ministry of Economic Development. (2007). *SMEs in New Zealand: Structure and Dynamics*. Retrieved in March 2009 from sme_centre.massey.ac.nz/files/SMEs_Structure_&_Dynamics_07.pdf.
- Mockus, A., Fielding, R.T. & Herbsleb, J.D. (2002). Two case studies of open source development: Apache and Mozilla. *ACM Transactions on Software Engineering and Methodology (TOSEM)* 11(3): 309-346.

MSNEncarta. (2007). Dictionary. Retrieved October 2007 from <http://encarta.msn.com/encnet/features/dictionary/dictionaryhome.aspx>.

Multi-criteria decision analysis. (2008). In *Wikipedia, The Free Encyclopedia*. Retrieved January 2008 from http://en.wikipedia.org/wiki/Multi-criteria_decision_analysis.

Myers, M.D. (ed) (1997) *Qualitative Research in Information Systems: References on Case Study Research*. MSIQ discovery. Retrieved on 20 January 2010 from <http://www.qual.auckland.ac.nz/case.aspx#Lee,%20A.S.%20A%20Scientific>

Noghini, V. D. (2004). A Simplified Variant of the Analytic Hierarchy Processes Based on a Nonlinear Scalarizing Function. *Computational Mathematics and Mathematical Physics*, Vol. 44(No. 7), 1194-1202.

NUREG-1150. (1989). *Reactor Risk Reference Document*. U.S. Nuclear Regulatory Commission, Washington D.C.

Opdahl A.L., Henderson-Sellers B. & Barbier F. (2001) Ontological analysis of whole-part relationships in OO-models. *Information and Software Technology*, 2001. 43(2001): 387 - 399.

OSI (2003). *Open Source Initiative*. OSI News. Retrieved November 2003 from www.opensource.org.

Oxford_English_Dictionary (2007). Retrieved October 2007 from <http://dictionary.oed.com>.

Pareto, V. (1896). *Cours d'Economie Politique*. Lausanne

Pascal, B. (1660) *Pensees*. Translated by Trotter, W.F. Retrieved on 20 January 2010 from www.ccel.org/ccel/pascal/pensees.toc.html.

Perens, B. (1999). *Open Sources: Voices from the Open Source Revolution*, O'Reilly online catalog. Retrieved in October 2003 from www.oreilly.com/catalog/opensources/book/perens.html.

Pfleeger, S.L., Hatton, L. and Howell, C.C. (2002). *Solid Software*. Prentice Hall, Upper Saddle River, New Jersey.

Pflughoeft, K. A., Ramamurthy, K., Soofi, E. S., Yasai-Ardekani, M., & Zahedi, F. (2003). Multiple Conceptualizations of Small Business Web Use and Benefit. *Decision Sciences*, 34(3): 467 – 512.

Phillips, C., Kemp, E.A. & Hedderley, D. (2005). Software Development methods and tools: a New Zealand study. *Australasian Journal of Information Systems (AJIS)*. Volume 12 No. 2: 21 - 49.

Pomerol, J.-C., & Barba-Romero, S. (2000). *Multicriterion Decision in Management: Principles and Practice*. Massachusetts: Kluwer Academic Publishers. 0-7923-7756-7.

Ramsey, F.P. (1931). *Truth and probability*. Chapter 7 in *The Foundations of Mathematics and other logical Essays*.

Ramsin, R & Paige, R.F. (2008) Process-Centered Review of Object Oriented Software Development Methodologies. *ACM Computing Surveys*. Vol. 40, No. 1, Article 3. New York. 0360-0300/2008/02-ART3. Retrieve in October 2008 from <http://DOI.acm.org/10.1145/1322432.1322435>.

Raymond, E. (2001). *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*. O'Reilly, Sebastopol.

Razmerita, L., Angehrn, A., & Maedche, A. (2003). *Ontology-based User Modeling for Knowledge Management Systems*. Retrieved in May 2003 from www.calt.insead.edu/Project/OntoLogging/documents/2003-UM-ontology_based_user_modeling_for_Knowledge_Management_Systems.pdf.

Reed, P. R. J. (2002). *Developing Applications with Java and UML*. Indianapolis, Addison-Wesley.

Riccia, G. D., Dubois, D., Kruse, R., & Lenz, H.-J. (2003). *Planning based on decision theory*. New York: Springer-Verlag Wien. 3-211-40756-1.

Richter, M. M. (1998). *Introduction - the basic concepts of CBR. Case-Based Reasoning Technology: From Foundations to Applications*. M. Bartsch-Sporl and S. Wess, Springer-Verlag. Vol. 1400.

Richardts, D. & Compton, P. (2000) *Generalising Ripple-Down Rules*. University of NSW, Sydney.

Riedl, A. (2004). *Voting and Social Choice* (Chapter 4) Retrieved in April 2004 from www1.fee.uva.nl/creed/pdfiles/moulinch4elferink.pdf.

RUP (2002). *Rational Rose: Product Information*, Rational the software development company. Retrieved in May 2004 from www.rational.com/products/rose/prodinfo.jsp.

Saaty, T. L. (1990a). *Multicriteria Decision Making: The Analytic Hierarchy Process: Planning, Priority Setting, Resource Allocation*. RWS Publications. Pittsburgh.

Saaty, T.L. (1990b) How to make a decision: The Analytic Hierarchy Process. *European Journal of Operational research*. Retrieved in Octpber 2008 from <http://sigma.poligran.edu.co/politecnico/apoyo/Decisiones/curso/Interfaces.pdf>.

Saltelli, A., Ratto, M., Tarantola, S. & Campolongo, F. (2005) *Sensitivity Analysis for Chemical Models*, *Chemical Reviews*, 105(7): 2811 – 2828.

Schach, S.R. (2002), *Object-Oriented and Classical Software Engineering*. McGraw Hill.

Schewe, K. D. (2000). *UML: A modern dinosaur? A critical analysis of the Unified Modelling Language*. 10th European-Japanese Conf. on Information Modelling and Knowledge Bases, Saariselka/Finland.

Schewe, K.D. and Thalheim, B. (2001). Modelling interaction and media objects. *Advances in Conceptual Modelling*. Métais, E. (Ed.) vol. 1959 of LNCS. Springer-Verlag, Berlin. pp. 313-324.

- Schewe, K-D, Kaschek, R., Matthews, C. and Wallace, C. (2002). Modelling web-based banking systems: Story Boarding and user profiling. In *Proceedings of the Workshop on Conceptual Modelling Approaches to e-Commerce*. Mayr, . and Van den Heuvel, W.J. (Eds.) Springer-Verlag.
- Sekaran, S. (1992). *Research methods for business: a skill-building approach*. New York: Wiley.
- Serrano, M.A., Calero, C. Sahraoui, H.A. & Piattini, M. (2008). Empirical studies to assess the understandability of data warehouse schemas using structural metrics. *Software Quality Journal*. Volume 16 no. 1. Springer Verlag.
- Sharma, S., Sugumaran, V., & Rajagopalan, B. (2002). A framework for creating hybrid-open source software communities. *Information Systems Journal*, 12(1): 7.
- Sigmer_Technologies (2008) Website Development User Profiling. Retrieved in January 2008 from www.sigmer.com/products_services/user_profile_report.php.
- Smith, M. & Glass, G. (1987) *Research and Evaluation in education and the Social Sciences*. New Jersey: Prentice Hall
- Sommerville, I. (2000). *Software Engineering*. USA, Addison Wesley.
- Steuer, R. E., & Miettinen, K. (2008). International Society on Multiple Criteria Decision Making [Electronic Version]. Retrieved in February 2008 from <http://www.terry.uga.edu/mcdm/>.
- Stevens, J. (2008). *Multimedia storytelling* [Electronic Version]. Retrieved in May 2008 from <http://multimedia.journalism.berkeley.edu/tutorials/reporting/starttofinish/storyboarding/>.
- Storyboard. (2008). In *Wikipedia, The Free Encyclopedia*. Retrieved March 2008, from <http://en.wikipedia.org/wiki/Storyboard>.
- Sundsted, T. E. (2001). Robust process gives developers boost. *InfoWorld* 6/18/2001 23(25): 66.
- Svahnberg, M., Wohlin, C., Lundberg, L., & Mattsson, M. (2003). A Quality-Driven Decision-support Method for Identifying Software Architecture Candidates. *International Journal of Software Engineering and Knowledge Engineering*, vol. 13, no. 5, 2003: 547–573.
- Taylor, M. J., McWilliams, J., Forsyth, H., & Wade, S. (2002). Methodologies and website development: a survey of practice. *Information and Software Technology*, 44(2002): 381 – 391.
- Taguchi, G., & Clausing, D. (1990). *Robust Quality*. Harvard Business Review, 1990: 68.
- Taylor, J., & Raden, N. (2007). *Smart (Enough) Systems: How to Deliver Competitive Advantage by Automating the Decisions Hidden in Your Business*, Prentice-Hall. 0-13-234796-2. (accessed through Safari Books).

Tellis, W. (1997). Application of a Case Study Methodology. *The Qualitative Report*. 3(3): 1-17.

Tian, J. (2005). *Software Quality Engineering: Testing, Quality Assurance, and Quantifiable Improvement*. John Wiley & Sons Inc. and IEEE CS Press, Hoboken, New Jersey.

Triantaphyllou, E. (2000). *Multi-criteria decision making methods: A Comparative study*. Kluwer Academic Publishers.

Triantaphyllou, E., Sanchez, A. (1997). A Sensitivity Analysis Approach for some Deterministic Multi-Criteria Decision Making Methods, *Decision Sciences*, 28(1) 1997: 151-194.

Trochim, W.M.K. (2006) *Research Methods Knowledge Base*. 4th ed. Pearson Publications.

Trueman, R. E. (1976). *An Introduction to quantitative methods for decision making* (Second edition ed.): Library of Congress Cataloging in Publication Data. 0-03-018391-x.

Unit-For-Sustainable-Development-and-Environment-Organization-of-American-States (2007) *Multi-Criteria / Dimensions in Decision-Making*. Retrieved in July 2007 from <http://www.oas.org/dsd/idsd/methodologies/criteria.htm>.

Vazquez, F. (1994) Selecting a Software Development Process. *Proceedings of the conference on TRI-Ada '94*. ACM 0-89791-529-1/92/001-0209.

Vidgen, R. (2002). Constructing a web information system development methodology. *Info Systems Journal* 2002(12): 247-261.

Vidgen, R., Avison, D., Wood, B., & Wood-Harper, T. (2002). *Developing Web Information Systems*: Butterworth-Heinemann. 0-7506-57634.

Wand, Y. Storey, V.C. and Weber, R. (1999). An ontological analysis of the relationship construct in conceptual modelling. *ACM Transactions on Database Systems*. 24(4): 494 - 528.

Watson, I. and F. Marir (1994). *Case-Based Reasoning: A Review*. Retrieved in July 2006 from <http://www.surveying.salford.ac.uk/ai-cbr-mirror/classroom/cbr-review.html>.

Welman, J. C. and S. J. Kruger (2001). *Research Methodology*, Oxford University Press.

White, D. J. (2006 - Paperback Copyright 1969). *Decision Theory*. Aldine Transaction - Division of Transaction Publishers. 0-202-30898-7.

Whitten, J.L., Bentley, L.D. and Dittman, K.C., (2007) *System analysis and design for the Global Enterprise*. McGraw-Hill. 978-0-07-110766-2. 7th edition.

Woo, F, Mikusaukas, R, Bartlett, D and Law, R (2006) A Framework for the effective adoption of Software Development Methodologies. *ACM SE'06*. Melbourne, Florida, USA: 198 – 202. 1-59593-315-8/06/0004.

Yin, R. (1994). *Case study research: Design and methods* (2nd ed.). Beverly Hills, CA: Sage Publishing.

Yin, R. K. (2006). Case study methods. Handbook of Complementary Methods in Education Research. G. C. Judith L. Green, Patricia B. Elmore Routledge: 896

Zhu, L., Aurum, A., Gorton, I. and Jeffery, R. (2005). Trade-off and Sensitivity Analysis in Software Architecture Evaluation Using Analytic Hierarchy Process. *Software Quality Journal*, 13(4) 2005: 357-375.

Glossary of terms

Analytic Hierarchy Process (AHP) is a decision making technique used to deal with complex decisions. This process was developed by T.L Saaty in the 1970's and is based on mathematic and human psychology. It has been extensively studied and expanded on since. AHP provides a detailed framework for structuring a problem, for representing and quantifying its elements, for relating those elements to overall goals, and for evaluating alternative solutions.

Case Based Reasoning (CBR) can be seen as the process of solving new problems based on the solutions of similar past problems. A case database is built up of similar problems and their solutions. A solution to the current problem is then identified by finding the previous solution that has been the most similar.

Decision Making is a process, usually cognitive which leads to a selection of a course of action among several alternatives.

Decision Models can be seen as the conceptualization of the relationship of the various components that are relevant in decision-making and planning.

Fuzzy logic can be seen as the processing of fuzzy set theory dealing with reasoning that is approximate rather than precisely deduced from classical predicate logic. It can be thought of as the application side of fuzzy set theory dealing with well thought out real world expert values for a complex problem

Knowledge Acquisition is the collection, analysis, modelling and validation of knowledge in order to manage the knowledge.

Multi-Criteria Decision Making (MCDM) is the study of methods and procedures by which concerns about multiple conflicting criteria can be formally incorporated into the management planning process

Open Source Development Process (methodology) is a new lightweight paradigm for software development which uses frameworks developed by volunteers as a basis for development. It may provide processes, practices, and communities that develop open source software for free.

Rational Unified Process is a comprehensive framework which allows the user to use industry-tested practices for software information system development as well as provide the tools for effective project management.

Ripple-Down (RDR) is an incremental knowledge acquisition methodology, which allows one to capture the knowledge from expert users in a knowledge base. The knowledge is captured as a set of rules in the knowledge base and then uses decision trees to get to the answer.

Sensitivity Analysis (SA) is the study of how the variation in the output of a model (numerical or otherwise) can be apportioned, qualitatively or quantitatively, to different sources of variation.

Value Benefit Analysis (VBA), also called the weighted scoring method is a decision method used to select a solution from a number of candidate solutions by using weights and calculating a "winner".

Weak-point Analysis is the removal of the very high or very low factors in the decision calculation that might have adversely influenced the result.

Index of terms

Agile 2, 25, 32, 45, 50-53, 61, 132
Agile Modelling (AM) 32
Analytical Hierarchy Process 5, 12, 86, 93 - 96
Application 6, 10, 13, 15, 21, 30, 39, 129, 141, 142, 154, 155
Borda (BO) 6, 110, 111, 143
Case Based Reasoning (CBR), 6, 103 - 106
Construction, 4, 47, 49, 61
Continuous Problems, 79
Copeland (CO), 111
Decision 69, 73, 75, 81
Decision Makers, 2, 28, 69 - 79
Decision Making 67 – 78, 156, 159, 167, 173, 201
Decision Making Methods 3, 75, 116, 125, 157
Development Processes 5, 10, 11, 23 - 28
Discrete Problems 79
eCIS Development Processes 3, 44
E-Commerce, 11, 23, 43
E-Commerce Development 43
E-Commerce Information Systems 1, 10, 23, 27, 35, 43 - 45
Elaboration 47 - 49
E-Process Selection VBA meta-Model 92
E-Process Selection AHP Meta-Model 101
E-Process Selection CBR Meta-Model 109
E-Process Selection SCM Meta-Model 113
Exposition 9
Inception 47, 48, 61
Integrated Meta-Model 5, 84, 114
Iterative Approaches 25
Kemeny (KE), 110 - 111
Maximin (MM) 111
Methodology 67, 91, 112
Multi-Criteria Decision Analysis (MCDA) 77, 79 - 81
Multi-Criteria Decision Making 84, 94, 96
Open Source Software (OSS) 32, 45, 52, 178
Operational Decisions 73

Prototype Tool 185 - 193
Quality Aspects 35 - 41
Rational Unified Process 47 - 50
Requirement Specification 168
Research Methodology 13 - 22
Selection Methodology 2, 5, 6, 19, 90, 112, 127, 132, 138, 141, 142, 150, 158
Sensitivity Analysis (SA) 99, 164, 188, 191
Similarity Measure 106 - 107
Simple Majority (SM) 109, 111
Social Choice Method (SCM) 110 - 115
Software Development Processes 2, 10, 27, 29, 31, 51
Storyboarding 53 - 56
Strategic Decisions 73
System Architecture (also software architecture) 6, 49, 63, 179
Tactical Decisions 73
Testing 77, 156, 188
Transition 47, 49, 61
User Profiling 53 - 56
User profiling and Storyboarding 53 - 56
Value Benefit Analysis 86
Waterfall Model 26
Web Information Systems 1, 10, 37
Weighted Scoring Method 86 - 90