Towards an ontology-driven software development approach: An unended quest

by

Nehemiah Mavetera

Submitted in fulfilment of the requirements for the degree

Doctor of Philosophy in Information Technology

in the

Faculty of Engineering, Built Environment and Information Technology

University of Pretoria

Pretoria, South Africa 10 November 2011
Candidate: Nehemiah Mavetera

Promoter: Prof. J.H. Kroeze

Department: School of Information Technology

Degree: Doctor of Philosophy in Information Technology
Declaration

I, Nehemiah Mavetera, declare that:

Towards an ontology-driven software development approach: An unended quest which I hereby submit for the degree of Doctor of Philosophy in Information Technology at the University of Pretoria, is my own work and has not previously been submitted by me for a degree at this or any other tertiary institution.

____________________________________
Signature of candidate

This 10th day of November, 2011.
Preface

“There is no royal road, but there is a road”

Fredrick, P. Brooks, 1987

It is common knowledge that software development projects more often than not, have missed their schedules, are out of budget or result in flawed products. A plethora of software development methods have been proposed and used. These can be classified abstractly as structured methodologies, object-oriented methodologies and, lastly, as a more recent invention, agile methodologies.

To support these methodologies, several methods, techniques and tools have been developed and used. All these were roads that were developed and used in an attempt to get to the end, but they are not royal roads. As is common in a methodological scientific way of solving a problem, the first step leads to the second, and the future step should be preceded by the current step. All these methodologies have brought the software development process to its current state and it is at this stage that we need to evaluate the successes achieved so far and, in certain cases, the failures too. In the end, we need to find a software development approach and methodology that suits the requirements of the future.

‘Towards an ontology-driven software development approach: An unended quest’ is a research study motivated by the lack of return on investment in current information systems. The root cause of this lack of return on investment is attributed to the way in which software products are developed. Software practitioners have adopted and used industrial engineering principles to fashion software products. Unlike the process of fashioning machines, the fashioning of software products requires an appreciation of the human element that is always present and resident in organizational information systems.

Failure to incorporate the humanistic element in the development process will lead to the resultant products being mechanistic, rule-based and no better than machines. A new road that fosters a transition from the development of mechanistic software products to the development of romantic software products is proposed in this thesis. Briefly, romantic systems are implemented using romantic software products. These are systems that imitate the way human beings behave in organizational information systems. They understand the organizational culture, context, capture tacit and intuitive knowledge among other things. Starting from the classical definition of ontology as a study of existence, the growth and incorporation of such ontology to new uses that focus on “What is there that we can
individuate and characterize?”, a new software development approach is proposed. This research study presents a new framework that can be used to develop software products that exhibit human behavioural characteristics and that are both adaptable and evolvable.
Abstract

Towards an ontology-driven software development approach: An unended quest

Nehemiah Mavetera

Over the years the field of software development has undergone a series of mutations, particularly in the types of approaches and methodologies that are applied during the development process. One thing that has not been fully achieved by software development method engineers is to move the development process completely from the mechanistic functionalist paradigm to a neo-humanist romantic paradigm. Although many researchers claim to have introduced new development paradigms in software development, these are merely new methodologies that are grounded in old paradigms. There are three fundamental development approaches that lie in the hard systems approach: the traditional structured, object-oriented and the recently invented agile approaches that have been widely adopted by software practitioners. Few of these practitioners have embraced the soft systems approach and their development methods have not migrated from the syntactic processes of the hard-systems approach.

Another problem that software developers continue to face is a lack of a method or tool that can augment current syntactic programming language technologies and software development methods by the addition of semantic-based tools to facilitate the construction of romantic, adaptive and evolvable software products. In fact, most of the problems encountered in software development can be attributed to deficiencies in the methodologies, that is: the approaches, methods, techniques and tools used during the development of the software product. This research study introduces the concept of ontologies in software development and motivates for an ontology-driven approach to software development that reduces the mechanistic nature of software products but increases their adaptability and usability.

Although current industrial and academic research has focused at the semantic properties of ontologies in software development, researchers have not considered how the methodological process can be designed and used to develop romantic software products. This research study used one variant of GTM and followed an interpretive approach in the investigation of several issues that are known and documented but not addressed by the current software development approaches. The field of software development has been investigated and a framework of requirements that enables the development of romantic
systems is presented. The ontology discipline, focusing on the semantic, pragmatic and contextual characteristics of ontologies, was also consulted. Starting from a set of differentiated ontological frameworks and from the syntactic, semantic and pragmatic nature of ontologies, the research then presents a framework of ontologies that can be co-opted into a software development approach to address the deficiencies in current software development approaches highlighted in the framework of software development requirements.

As part of the research findings, a new definition of ontology, as well as a framework of components that make up the ontology and a theoretical translation model that is needed to develop romantic software products, are presented. The theoretical translation model comprises three parts: an ontology-driven software development framework, an ontological approach to software development and an ontology-based methodology for software development. Of note is the use of domain, method, process, intentional, and status ontologies at different stages of software development to cater for the semantic, pragmatic and contextual gaps that are not currently addressed by existing development approaches. However, in this study, a balance is reached between addressing the needs of current and future developers of software products, that is, one that reacts to an urgent market need, as well as addressing a software development approach need that is heavily grounded in the softer, neo-humanist paradigm.

**KEYWORDS**

Software development, Framework, Approach, Methodology, Romanticism, Ontology.
To

My Late Parents, Nehemiah (Snr.), Gamuchirai and Wife, Tsungai Mavetera

and

My Lovely Wife, Chipo Gertrude and my children

Samantha, Patience, Kudzai Rejoice, Tinotenda and Kudzanai Nehemiah (Jnr.)
ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my wife, Chipo Gertrude, for her support during this trying time. I certainly could not have got this far without her unwavering support, both at home and work. I will not forget my children, Samantha, Patience, Kudzai, Tinotenda and Kudzanai. I will never forget the number of times you asked about the chapter I was working on.

I would also like to thank my promoter, Professor Jan Kroeze, for his insightful reflections, mentoring and encouragement. It was a great privilege to work with you. I learnt of many professional issues that had been always been elusive from my lower and junior studies. I cannot forget my Head of Department: Professor Carina De Villiers, I say thank you very much.

I cannot forget my workmates. Thank you for allowing me to go on study leave and covering for my absence. It is time for you to get the benefits of my short absence from work. My thanks are also extended to the North West University and to the National Research Foundation (NRF) of South Africa; both these institutions co-funded this research study. Without their financial support, I certainly would not have been able to undertake this study.

To all my interviewees, my sincere thanks. Without your cooperation, I certainly would not have discovered anything. I wish there were more people like you who were willing to provide information for the sake of academic development and research.

And, lastly,

Thank You, Almighty God, for With You Nothing is Impossible.

Matthew 19, verse 26
This thesis, *Towards an ontology-driven software development approach: An unended quest*, by Mr. Nehemiah Mavetera, was edited for language, grammar and style by Ciaran Michael Mac Carron MA, Hons. BSc, who has been a member of the South African Translators’ Institute for over 25 years and has had over 25 years’ experience in the editing and translation of theses, manuals etc.

Signed

C M Mac Carron

20 August 2010
# TABLE OF CONTENTS

CHAPTER ONE ......................................................................................................................... 1

INTRODUCTION AND RESEARCH MOTIVATION ................................................................. 1

1.0 Introduction ...................................................................................................................... 1

1.1 Research Motivation ...................................................................................................... 2
   1.1.1 Communication Requirements in Software Development ........................................ 5

1.2 The Research Interest .................................................................................................... 6
   1.2.1 Gap in the Field of Study ....................................................................................... 7
   1.2.1.1 The Software Development Process .................................................................. 9
   1.2.1.2 Software Development and Ontologies .......................................................... 10

1.3 Research Methodology ................................................................................................. 11
   1.3.1 GTM .................................................................................................................... 11

1.4 Generic Research Propositions ..................................................................................... 12

1.5 Preliminary Research Questions ................................................................................... 13
   1.5.1 Main Research Question ...................................................................................... 14
   1.5.2 Secondary Research Questions .......................................................................... 15
   1.5.2.1 The ‘What is?’ type of questions. .................................................................... 16
   1.5.2.2 Why ontology? ............................................................................................... 19
   1.5.2.3 The “How should?” And “How does?” types of questions .............................. 21

1.6 Aims and Objectives of the Study ............................................................................... 21
   1.6.1 Aims of the Study .............................................................................................. 21
   1.6.2 Objectives of the Study ....................................................................................... 22

1.7 Research Scope and Delimitation ............................................................................... 23
   1.7.1 What this Research Study is Not .......................................................................... 24

1.8 Contribution of Research to the Body of Knowledge ............................................... 25

1.9 Structure of the Thesis ................................................................................................. 26
CHAPTER TWO .................................................................................................31

UNDERSTANDINGS IN INFORMATION SYSTEMS RESEARCH ......................31

2.0 Introduction ............................................................................................31

2.1 What is theory? .......................................................................................32

2.1.1 General Types of Theory ..................................................................33

2.2 Strategies for Conducting Research .......................................................35

2.3 Types of Research in Information Systems ...........................................35

2.3.1 Theoretical Research .................................................................36

2.3.2 Empirical Research ........................................................................36

2.4 Research Tasks in Information Systems .................................................36

2.4.1 Constructive Research Task .......................................................37

2.4.2 Nomothetic Research Task ..........................................................37

2.4.2.1 Deductive Theory ..................................................................37

2.4.3 Ideographic Research Task ............................................................38

2.5 Sociological Paradigms Applied to Information Systems .................38

2.5.1 The Functionalist Paradigm ..........................................................40

2.5.2 The Interpretive Paradigm .............................................................41

2.5.3 The Radical Structuralist Paradigm ..............................................41

2.5.4 The Neo-Humanist Paradigm .......................................................42

2.6 Philosophical Groundings in Information Systems Research ...........42

2.6.1 Ontological Grounding to Information Systems Research ............42

2.6.1.1 Objectivism ...........................................................................43

2.6.1.2 Constructionism (Constructivism) .........................................44

2.6.2 Epistemological Grounding to Information Systems Research ..........44

2.6.2.1 Positivism versus Anti-Positivism ..........................................45
3.3.3.1 Qualitative Data Acquisition 65
3.3.3.2 Data Acquisition Methods for this Study 66
3.3.3.3 Primary Data Acquisition and Sampling 67
3.3.3.4 Secondary Data Acquisition Methods 68
3.3.4 The Research Goals 69
3.3.5 Data Analysis in GTM 71
3.3.5.1 GTM Coding 71

3.4 The Double Mapping Principle .................................................................75

3.5 From Substantive to Formal Theory..........................................................77

3.6 The Reflexive Grounded Theory Process..................................................77

3.7 Judging the Fidelity of Generated Grounded Theory.................................79
3.7.1 Confirmability 80
3.7.2 Dependability/Auditability 81
3.7.3 Internal Consistency 81
3.7.4 Transferability 81

3.8 Evaluation of Qualitative Interpretive Research Study ..............................82
3.8.1 The Fundamental Principle of the Hermeneutic Cycle 82
3.8.2 The Principle of Contextualization 83
3.8.3 The Principle of Interaction between the Researcher and the Subjects 83
3.8.4 The Principle of Abstraction and Generalisation 84
3.8.5 The Principle of Dialogical Reasoning 84
3.8.6 The Principle of Multiple Interpretations 85
3.8.7 The Principle of Suspicion 85

3.9 Limitations of the Research Methodology ...............................................85

3.10 Evaluation Criteria for the Generated Theory .........................................86

3.11 Summary ....................................................................................................88
CHAPTER FOUR

SOFTWARE DEVELOPMENT PRACTICES, INFORMATION SYSTEMS AND ORGANIZATIONS

4.0 Introduction

4.1 The Nature of Organizational Systems

4.1.1 The Complex Nature of Organizational Systems

4.1.2 Culture in Organizations

4.1.3 The Role of Concepts in Information Systems Development (ISD)

4.1.4 Context in Information Systems

4.2 The Practice of Software Engineering

4.2.1 The Software Development Problem

4.3 Software Product Development Practices

4.3.1 Communication in Software Development

4.3.2 Software Engineering Reference Model

4.4 The Need for a Conceptual Schema

4.5 The Mechanistic Nature of Information Systems

4.5.1 Mechanistic Systems, Systematicity and System-Formation

4.5.1.1 Romanticism

4.5.2 Transition to Romantic Systems

4.6 Activity Theory, Actor-Network Theory and Theory of Organized Action

4.6.1 Activity Theory

4.6.2 The Social Network Aspect of Activity Systems

4.6.3 Human Activity Systems (HAS)

4.6.4 Theory of Organized Activity (TOA)

4.7 The Approach versus Methodology Debate

4.8 Software Development Approaches

4.8.1 Structured Development Approach
6.5.1.1 The Software Development World View 180
6.5.1.2 The Software Development Paradigm 180
6.5.1.3 The Software Development Approach 182
6.5.1.4 The Software Development Method 185
6.5.1.5 Adaptive and Evolving Software Development 187
6.5.1.6 The Software Development Environment 188
6.5.2 Refined Research Questions 190

6.6 GTM Theoretical Sampling and Saturation ......................................................192
6.6.1 Propositions PA and PB Story Lines 192
6.6.2 Proposition PB Story Line 193
6.6.3 Proposition PC Story Line 193
6.6.4 Proposition PD Story Line 195
6.6.5 Proposition PE Story Line 195

6.7 The Double-Mapping Principle Revisited ......................................................197

6.8 A Requirements Framework for a Software Development Approach-The What Part ........................................................................................................198

6.9 The Ontology-Driven Software Development Framework–The How Part ......202
6.10 An Ontology-Driven Software Development Approach-The How Should Part. ........................................................................................................207

6.11 Discussions ....................................................................................................211
6.12 Summary ........................................................................................................212

CHAPTER SEVEN ...............................................................................................................213

RESEARCH EVALUATION, CONCLUDING STATEMENTS AND FUTURE WORK. 213

7.0 Introduction .......................................................................................................213
7.1 Research Quality Considerations ........................................................................213
7.2 Contribution to the Field of Software Development .......................................215
7.3 Evaluation of the Study’s Contribution to the Body of Knowledge.................218
7.3.1 Evaluation of the Generated Theory 219
7.3.2 Evaluation 226
7.4 Recommendations, Future Work and Limitations...........................................227
7.5 Conclusion ........................................................................................................229

BIBLIOGRAPHY ........................................................................................................230

APPENDICES ...........................................................................................................253
Appendix A: Data Gathering Interview Questions 253
Appendix B: Publications Generated from the Research Process 261
Appendix C: Incidents for Proposition PA 264
Appendix D: Incidents for Proposition PB 265
Appendix E: Incidents for Proposition PC 267
Appendix F: Incidents for Proposition PD 270
Appendix G: Incidents for Proposition PE 273
Appendix H: Full list of interview audio recordings (on CD) 275
Appendix I: Full list of interview transcripts (on CD) 275
Appendix J: Full list of incidents and codes for all the interviews (on CD) 275
Appendix K: Network diagram (on CD) 275
Appendix L: Full List of codes and code families for all the interviews (on CD) 275
Appendix M: Full list of incidents and memos for all the interviews (on CD) 275
Appendix N: Definition of Concepts Used in the Thesis 275

TABLE OF FIGURES
Figure 1.1: Preliminary Research Questions (Adapted from Roode, 1993)..................15
Figure 1.2: Structure of Thesis..................................................................................27
Figure 2.1: Sociological Paradigms (Adapted from Burrell and Morgan, 1979).........40
Figure 3.1: Three-Tier Incident and Concept Mappings ............................................76
Figure 3.2: A Reflective Grounded Theory Process (Adapted from Gasson, 2003)......78
Figure 4.1: Communication Gap in Software Development ......................................105
Figure 4.2: Software Reference Model (Adapted from Gunter et al., 2000)...............108
Figure 4.3: Components of an Integrated Information System (Adapted from Sowa, 2000) 109
Figure 4.4: The Concept of an Activity System (Adapted from Roque et al., 2003) .......... 118
Figure 5.1: Different Realms of the World (Adapted from Goldkuhl, 2002) ......................... 138
Figure 5.2: Structural Representation of Ontology ................................................................. 141
Figure 5.3: OntoSoft Case Tool Architecture (Adapted from Mavetera, 2007) ....................... 153
Figure 5.4: A Linguistic Model (Adapted from Molenaar, 1998) ........................................... 160
Figure 6.1: Aspects of Consideration in Software Development ......................................... 177
Figure 6.2: Expanded Double Mapping Principle .................................................................... 198

LIST OF TABLES
Table 3.1: Research Goals and Data Gathering Methods ......................................................... 70
Table 3.2: Criteria for Judging Qualitative-Interpretive Research ........................................... 80
Table 4.1: Characterizing Approach, Methodology and Method ........................................... 125
Table 4.2: Philosophical Groundings of Software Development Approaches (Adapted from
Brown et al, 2004:4139) ........................................................................................................... 128
Table 6.1: Interview Themes ...................................................................................................... 167
Table 6.2A: Categories from Open Coding Process ................................................................. 172
Table 6.2B: Thematic Areas Identified from Open Coding Process ........................................ 173
Table 6.3: Software Development Relationships from Practitioners’ Perspective .................. 176
Table 6.4: Requirements Framework for a Software Development Approach ...................... 199
Table 6.5: Ontology-Driven Software Development Framework ............................................ 205
Table 6.6: An Ontology-Driven Approach to Software Development ................................... 208
Table 6.7: An Ontology-Driven Software Development Methodology ................................... 209