

A framework to assist in the assessment and tailoring of
agile software development methods

Asif Qumer

A dissertation submitted in fulfillment of the requirements for the degree of

Doctor of Philosophy in Computing Sciences

School of Software

Faculty of Engineering and Information Technology

University of Technology, Sydney

July 2010

CERTIFICATE OF AUTHORSHIP

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree except as fully acknowledged within the text. I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

Production Note:
Signature removed prior to publication.



Acknowledgements

I have been privileged to have Prof. Brian Henderson-Sellers and Prof. John Debenham as my supervisors. I would like to express my sincere gratitude to my supervisor Prof. Brian Henderson-Sellers for providing me an opportunity to work with him on this research project. His highly valuable support, coaching, encouragement, quick feedback and guidance helped me to finish this research project.

This study, off course, would not have been possible without the support of my family. I am very grateful to my parents Mr. Riaz Ahmad Gill and Asghari Baigum (late) for their precious love, blessings, support, encouragement and care since childhood. I am thankful to my lovely wife Saja and her family, my dearest sisters Fakhra, Fozia and Zobia, and my brother Kashif.

I would like to express my sincere thanks to my all colleagues and friends for their valuable feedback and support, especially Prof. Hung Nguyen, Dr. Tom McBride and my friends Ommar Sabbagh and Adam Alami.

At last, but most dearest, I am really grateful to my late mother for her support and encouragement to do the PhD. May God bless her soul and a place in heaven. I want to name this thesis to my late mother who would not be able to see me graduating.

I also wish to thank the Australian Research Council for the financial support. I am thankful to the people from both the research community and the software industry experts who helped me with their valuable feedback and experience throughout this research project.

I am thankful to all the reviewers for the valuable feedback and comments.

Thank you all.

Research Contribution and Publications

During this PhD. research project, I collaborated with my supervisor and other colleagues, and published the components of this research work (ASSF) in a number of rigorously reviewed international conference papers and scientific journals. The writing of papers, for publication, was an opportunity to present my work for review before including in my thesis. Please find the list of the publications, which have been included in this thesis, in Appendix F.

Table of Contents

1. INTRODUCTION.....	1
1.1. BACKGROUND AND CONTEXT	1
1.2. AGILE SOFTWARE DEVELOPMENT	2
1.3. THE RESEARCH PROBLEM	4
1.4. THE RESEARCH AIM.....	6
1.5. SCOPE	7
1.6. CONTRIBUTIONS.....	8
1.7. APPLICATION AND USERS	10
1.7.1. <i>Assessment: Adoption and Improvement</i>	10
1.7.2. <i>Tailoring: Adoption and Improvement</i>	12
1.8. RESEARCH STRATEGY	13
1.9. THESIS ORGANIZATION	14
2. AGILE SOFTWARE DEVELOPMENT REVIEW	15
2.1. AGILE SOFTWARE DEVELOPMENT	15
2.1.1. <i>Evolution of Software Development Methods</i>	16
2.1.2. <i>Code-and-Fix</i>	16
2.1.3. <i>Plan-Driven</i>	17
2.1.4. <i>Agile</i>	18
2.2. AN OVERVIEW OF AGILE METHODS.....	22
2.2.1. <i>Extreme Programming - XP</i>	22
2.2.2. <i>Scrum</i>	24
2.2.3. <i>Feature Driven Development – FDD</i>	26
2.2.4. <i>Adaptive Software Development – ASD</i>	27
2.2.5. <i>Dynamic Software Development Method – DSDM</i>	28
2.2.6. <i>Crystal Methodologies</i>	30
2.3. PLAN-DRIVEN METHODS	32
2.3.1. <i>Waterfall and Spiral Software Development Processes</i>	32
2.4. REVIEW AND COMPARISON OF AGILE AND NON-AGILE METHODS	34
2.4.1. <i>The 4-Dimensional Analytical Tool</i>	34
2.4.2. <i>The Review and Analysis</i>	36
2.5. AGILE PROCESS LIFECYCLE MANAGEMENT: ASSESSMENT AND TAILORING.....	46
2.6. ANALYSIS RESULTS AND INTERPRETATION.....	50
2.6.1. <i>Software Development Methodology Characteristics</i>	50
2.6.2. <i>Lifecycle Management</i>	53
2.7. METHOD ENGINEERING.....	54
2.7.1. <i>Defining Method Engineering</i>	55
2.7.2. <i>Why is Method Engineering Important?</i>	55
2.7.3. <i>Method Engineering and Method Construction</i>	56
2.7.4. <i>Metamodels</i>	58
2.8. SUMMARY AND CONCLUSION	60
3. RESEARCH QUESTION	61
3.1. SITUATIONAL METHOD ENGINEERING PERSPECTIVES.....	61
3.2. METHOD CHARACTERISTICS	63
3.3. LIFECYCLE MANAGEMENT.....	63
3.4. THE RESEARCH QUESTION	64
3.5. SUMMARY AND CONCLUSION	65
4. RESEARCH DESIGN	66
4.1. BACKGROUND AND CONTEXT	66
4.2. REVIEW OF RESEARCH METHODOLOGIES	66
4.2.1. <i>Action Research</i>	67
4.2.2. <i>Grounded Theory</i>	68
4.2.3. <i>Case Study</i>	69
4.3. RESEARCH METHODOLOGY	71
4.4. RESEARCH INSTRUMENTS	76

4.4.1.	<i>Extant Processes and Experience</i>	77
4.4.2.	<i>Software Engineering - Metamodels for Development Methodologies</i>	77
4.4.3.	<i>Feedback from Industry and Researchers</i>	78
4.5.	RESEARCH ETHICS	79
4.6.	VALIDITY	79
4.7.	SUMMARY AND CONCLUSION	80
5.	FRAMEWORK CHARACTERISTICS	81
5.1.	BACKGROUND	81
5.2.	THE AGILE SOFTWARE SOLUTION FRAMEWORK (ASSF)	82
5.3.	FRAMEWORK CHARACTERISTICS	83
5.3.1.	<i>People</i>	84
5.3.2.	<i>Process</i>	85
5.3.3.	<i>Product</i>	86
5.3.4.	<i>Tools</i>	87
5.3.5.	<i>Agility</i>	88
5.3.6.	<i>Abstraction</i>	91
5.3.7.	<i>Business Value</i>	94
5.3.8.	<i>Policy</i>	94
5.3.9.	<i>Rules</i>	95
5.3.10.	<i>Legal</i>	96
5.4.	THE FRAMEWORK CHARACTERISTICS SET – ARCHITECTURE	96
5.5.	SUMMARY AND CONCLUSION	100
6.	AGILE PROCESS LIFECYCLE MANAGEMENT	101
6.1.	THE AGILITY ADOPTION AND IMPROVEMENT LIFECYCLE (AAIL)	103
6.1.1.	<i>Initiation</i>	104
6.1.2.	<i>Development</i>	104
6.1.3.	<i>Deployment</i>	106
6.1.4.	<i>Administration</i>	107
6.1.5.	<i>Management</i>	107
6.1.6.	<i>Governance</i>	108
6.2.	THE AGILITY ADOPTION AND IMPROVEMENT PROCESS (AAIP)	111
6.2.1.	<i>Initiation</i>	112
6.2.2.	<i>Development</i>	114
6.2.3.	<i>Deployment</i>	116
6.2.4.	<i>Administration</i>	118
6.2.5.	<i>Management</i>	119
6.2.6.	<i>Governance</i>	120
6.3.	THE CONTEXTUAL ANALYSIS MODEL (CAM)	122
6.3.1.	<i>Prospect: “Who”</i>	125
6.3.2.	<i>Business Context: “What”</i>	126
6.3.3.	<i>Readiness: “Why”</i>	127
6.3.4.	<i>Timeframe: “When”</i>	132
6.3.5.	<i>Area: “Where”</i>	132
6.3.6.	<i>Approach: “How”</i>	132
6.3.7.	<i>Summary</i>	133
6.4.	THE KEY AGILITY INDICATORS INDEX (KAII)	133
6.4.1.	<i>Key Agility Indicators: Degree of Agility</i>	136
6.4.2.	<i>Key Agility Indicators: Agile Level</i>	142
6.4.3.	<i>Application of Degree of Agility and Agile Levels</i>	144
6.5.	THE AGILITY ADOPTION AND IMPROVEMENT MODEL (AAIM)	145
6.5.1.	<i>Agile Levels</i>	146
6.5.2.	<i>Agile Level Assessment Questionnaire</i>	151
6.6.	THE AGILITY ADOPTION AND IMPROVEMENT SCORECARD	153
6.7.	THE AGILE TOOLKIT	154
6.7.1.	<i>Agile Knowledge-base</i>	155
6.7.2.	<i>Composer</i>	157
6.7.3.	<i>Agility Calculator</i>	157
6.7.4.	<i>Publisher</i>	157

6.7.5.	Registry.....	158
6.7.6.	Knowledge Transformer.....	161
6.7.7.	Visualiser.....	161
6.8.	SUMMARY AND CONCLUSION.....	162
7.	TESTING AND EMPIRICAL ANALYSIS - I.....	163
7.1.	TESTING.....	163
7.1.1.	Case Study-01: Agile Process Tailoring and Adoption.....	164
7.1.2.	Agile Product Enhancement Process Model.....	166
7.1.3.	Case Study-02: Agile Process Tailoring and Adoption.....	168
7.1.4.	Agile Service Oriented Process Model.....	170
7.1.5.	Test Results Interpretation.....	172
7.1.6.	Reflection and Opportunities for Improvement.....	173
7.1.7.	Validity and Limitation.....	175
7.2.	EMPIRICAL ANALYSIS - I.....	175
7.2.1.	Software Development Methodology Characteristics.....	177
7.2.2.	Agility.....	179
7.2.3.	People, Process and Product.....	180
7.2.4.	Tools.....	182
7.2.5.	Knowledge Management.....	183
7.2.6.	IT Governance.....	185
7.2.7.	Validity and Limitations.....	186
7.3.	SUMMARY AND CONCLUSION.....	187
8.	EMPIRICAL ANALYSIS - II.....	189
8.1.	ASSESSMENT INSTRUMENT.....	189
8.2.	CATEGORICAL DATA ANALYSIS.....	191
8.2.1.	Purpose.....	191
8.2.2.	Comprehensive.....	192
8.2.3.	Relevance.....	192
8.2.4.	Importance.....	193
8.2.5.	Understandable.....	193
8.2.6.	Practical.....	194
8.2.7.	Overall Analysis Results.....	194
8.3.	SUBJECTIVE EVALUATION.....	195
8.3.1.	Experts Comments.....	195
8.3.2.	Suggested Changes and Response.....	196
8.4.	VALIDITY AND LIMITATIONS.....	200
8.5.	SUMMARY AND CONCLUSION.....	202
9.	CONCLUSION AND FUTURE RESEARCH.....	203
9.1.	RESEARCH OUTCOMES.....	203
9.1.1.	Framework Characteristics (FCs).....	203
9.1.2.	Agile Process Lifecycle Management (APLM).....	204
9.2.	RESEARCH STRATEGY AND EMPIRICAL ANALYSIS.....	206
9.2.1.	Testing.....	206
9.2.2.	Empirical Study 1.....	206
9.2.3.	Empirical Study 2.....	207
9.3.	KEY RESEARCH PUBLICATIONS AND CONTRIBUTIONS.....	207
9.3.1.	Evaluation of Degree of Agility and Adoptability of Agile Methods.....	208
9.3.2.	Agile Agent-Oriented Software Development Framework.....	208
9.3.3.	Agile Software Solution Framework.....	209
9.4.	FUTURE RESEARCH.....	209
9.5.	CONCLUSION.....	210
	BIBLIOGRAPHY.....	212
	APPENDIX A – ETHICS APPROVAL.....	233
	APPENDIX B – STRUCTURED QUESTIONNAIRE.....	234
	APPENDIX C – CONSENT FORM.....	238

APPENDIX D – APLM EMPIRICAL ASSESSMENT INSTRUMENT.....	239
APPENDIX E – DEGREE OF AGILITY ANALYSIS (QUANTITATIVE)	241
APPENDIX F – RESEARCH PAPERS	245
APPENDIX G – EMPIRICAL STUDY DATA	248

List of Figures

FIGURE 1-1: THE RESEARCH QUESTION DIMENSIONS	6
FIGURE 1-2: AGILE SOFTWARE SOLUTION FRAMEWORK (AN OVERVIEW DIAGRAM)	8
FIGURE 2-1: SOFTWARE DEVELOPMENT EVOLUTION	16
FIGURE 2-2: EMERGENCE OF HYBRID APPROACH (MIXINGS AGILE AND NON-AGILE PRACTICES)	48
FIGURE 2-3: OMG'S MULTILAYER HIERARCHY – ADOPTED 3 METAMODEL LAYERS (PERSONAL COMMUNICATION, PROF. BRIAN HENDERSON-SELLERS 2008).....	58
FIGURE 3-1: RESEARCH TOPIC DIMENSIONS	65
FIGURE 4-1: THE RESEARCH PROCESS AND ASSF DEVELOPMENT	74
FIGURE 5-1: THE AGILE SOFTWARE SOLUTION FRAMEWORK – HIGH LEVEL ARCHITECTURE	82
FIGURE 5-2: FRAMEWORK CHARACTERISTICS.....	84
FIGURE 5-3: PEOPLE CHARACTERISTIC	85
FIGURE 5-4: PROCESS CHARACTERISTIC	86
FIGURE 5-5: PRODUCT CHARACTERISTIC	87
FIGURE 5-6: TOOLS CHARACTERISTIC.....	88
FIGURE 5-7: AGILITY CHARACTERISTIC	88
FIGURE 5-8: ABSTRACTION CHARACTERISTIC.....	91
FIGURE 5-9: POLICY CHARACTERISTIC.....	95
FIGURE 5-10: FRAMEWORK CHARACTERISTICS SET – ARCHITECTURE	98
FIGURE 6-1: AGILE PROCESS LIFECYCLE MANAGEMENT COMPONENTS (DETAILED ARCHITECTURE – AFTER QUMER AND HENDERSON-SELLERS 2010).....	102
FIGURE 6-2: AGILITY ADOPTION AND IMPROVEMENT LIFECYCLE (AAIL).....	103
FIGURE 6-3: CONTEXTUAL ANALYSIS MODEL (CAM).....	124
FIGURE 6-4: AGILITY ANALYSIS	130
FIGURE 6-5: KEY AGILITY INDICATORS INDEX WHEEL.....	134
FIGURE 6-6: CLASS MODEL DEPICTING KAI I	135
FIGURE 6-7: KEY AGILITY INDICATORS (LABELED KAIS).....	137
FIGURE 6-8: AGILE PRACTICE SELECTION PROCESS.....	145
FIGURE 6-9: AGILITY ADOPTION AND IMPROVEMENT MODEL (AAIM V2.0) – (AN IMPROVED VERSION OF THAT IN QUMER <i>ET AL.</i> , 2007).....	147
FIGURE 6-10: AGILE TOOLKIT ARCHITECTURE	155
FIGURE 6-11: AGILE KNOWLEDGE-BASE METADATA EXPRESSED AS A CLASS MODEL.....	156
FIGURE 6-12: REGISTRY CLASSIFICATION STRUCTURE AND FILES.....	158
FIGURE 7-1: AGILE PRODUCT-ENHANCEMENT PROCESS MODEL	167
FIGURE 7-2: AGILE SERVICE ORIENTED PROCESS MODEL	171
FIGURE 9-1: STAGES AND PRACTICES.....	204

List of Tables

TABLE 2-1: THE XP PROCESS	23
TABLE 2-2: THE XP PRIMARY PRACTICES	24
TABLE 2-3: THE SCRUM PROCESS.....	25
TABLE 2-4: THE SCRUM PRACTICES	25
TABLE 2-5: THE FDD PROCESS	26
TABLE 2-6: THE FDD PRACTICES	27
TABLE 2-7: THE ASD PROCESS	27
TABLE 2-8: THE ASD PRACTICES	28
TABLE 2-9: THE DSDM PROCESS.....	29
TABLE 2-10: THE DSDM PRACTICES	30
TABLE 2-11: THE CRYSTAL PROCESS	31
TABLE 2-12: THE CRYSTAL PRACTICES.....	31
TABLE 2-13: 4-DAT’S FOUR DIMENSIONS (UPDATED AFTER QUMER AND HENDERSON-SELLERS 2006A; 2008A).....	35
TABLE 2-14: REVIEW OF AGILE METHODS	39
TABLE 2-15: REVIEW OF PLAN-DRIVEN METHODS	40
TABLE 2-16: DEGREE OF AGILITY FOR SIX AGILE METHODS (UPDATED AFTER QUMER AND HENDERSON-SELLERS 2008A)	42
TABLE 2-17: DEGREE OF AGILITY WATERFALL AND SPIRAL APPROACHES (AFTER QUMER AND HENDERSON-SELLERS 2008A)	43
TABLE 2-18: SOFTWARE PROCESSES IN SIX AGILE METHODS (UPDATED AFTER QUMER AND HENDERSON-SELLERS 2008A)	45
TABLE 2-19: SOFTWARE PROCESS IN THE TWO PLAN-DRIVEN APPROACHES (AFTER QUMER AND HENDERSON-SELLERS 2008A)	46
4-1: RESEARCH DATA RESOURCES AND CLASSIFICATION.....	76
TABLE 5-1: PERSONAL AND TEAM ATTRIBUTES CHECKLIST.....	84
TABLE 5-2: PROCESS, SUB-PROCESS AND PRACTICE OR PROCESS FRAGMENT.....	85
TABLE 5-3: PRODUCT.....	86
TABLE 5-4: TOOLS CHARACTERISTIC	87
TABLE 5-5: AGILITY ATTRIBUTES (AFTER QUMER AND HENDERSON-SELLERS 2006C)	90
TABLE 6-1: PRACTICE SPECIFICATION STRUCTURE OR TEMPLATE SAMPLE (BASED ON 4-DAT (QUMER AND HENDERSON-SELLERS 2006A)	105
TABLE 6-2: CATEGORISATION OF IT GOVERNANCE KEY CONCEPTS (AFTER QUMER 2007).....	109
TABLE 6-3: SOFTWARE METHODOLOGY CONSTRUCTION AND SELECTION FACTORS AND THEIR PERCEIVED IMPORTANCE (AFTER QUMER AND HENDERSON-SELLERS 2009A).....	126
TABLE 6-4: AGILITY BUSINESS GOALS INDEX	128
TABLE 6-5: AGILITY SUCCESS INDICATORS INDEX	129
TABLE 6-6: AGILITY FINANCIAL ANALYSIS MATRIX	132
TABLE 6-7: KAIS MATRIX	138
TABLE 6-8: AGILITY ATTRIBUTES KAI MATRIX.....	139
TABLE 6-9: AGILE VALUES KAI MATRIX	140
TABLE 6-10: AGILE VALUES AND AGILITY ATTRIBUTES	141
TABLE 6-11: AGILE PRINCIPLES KAI MATRIX	141
TABLE 6-12: AGILE PRINCIPLES AND AGILE VALUES.....	142
TABLE 6-13: AGILE LEVELS AND AGILE PRINCIPLES	143
TABLE 6-14: AGILE LEVEL ASSESSMENT QUESTIONNAIRE	152
TABLE 6-15: AGILITY ADOPTION AND IMPROVEMENT SCORECARD (AAIS) TEMPLATE	153
TABLE 6-16: AGILE TOOLKIT STORYBOARD	159
TABLE 7-1: CASE-01	164
TABLE 7-2: AGILE ADOPTION (AGILE PRACTICES FROM AGILE KNOWLEDGE BASE).....	165
TABLE 7-3: AGILE PRODUCT ENHANCEMENT PROCESS (APEP)	166
TABLE 7-4: CASE-02	168
TABLE 7-5: AGILE ADOPTION (AGILE PRACTICES FROM AGILE KNOWLEDGE BASE).....	170
TABLE 7-6: AGILE SERVICE ORIENTED PROCESS (ASOP).....	171
TABLE 7-7: PROFILE OF THE PARTICIPANTS (AFTER QUMER AND HENDERSON-SELLERS 2009A).....	177
TABLE 7-8: METHODOLOGY CHARACTERISTICS (AFTER QUMER AND HENDERSON-SELLERS 2009A).....	178
TABLE 7-9: CORE METHOD SELECTION OR CONSTRUCTION FACTORS (AFTER QUMER AND HENDERSON-SELLERS 2009A)	178

TABLE 7-10: GOVERNANCE, KE&M AND BUSINESS (AFTER QUMER AND HENDERSON-SELLERS 2009A)	179
TABLE 7-11: AGILITY ATTRIBUTES	179
TABLE 7-12: DEGREE OF AGILITY	180
TABLE 7-13: AGILITY ADOPTION AND IMPROVEMENT MODEL (AAIM)	180
TABLE 7-14: PEOPLE CHARACTERISTIC.....	180
TABLE 7-15: PROCESS CHARACTERISTICS.....	181
TABLE 7-16: PROCESS FRAGMENT CHARACTERISTICS	181
TABLE 7-17: PRODUCT FRAGMENT ATTRIBUTES	182
TABLE 7-18: TOOLS AND WORKSPACE ATTRIBUTES.....	182
TABLE 7-19: AGILE COMMUNICATION TOOL	183
TABLE 7-20: TOOLS AND WORKSPACE TYPES.....	183
TABLE 7-21: SOFTWARE DEVELOPMENT KNOWLEDGE MANAGEMENT.....	184
TABLE 7-22: SOFTWARE DEVELOPMENT KNOWLEDGE TYPES	184
TABLE 7-23: SOFTWARE DEVELOPMENT KNOWLEDGE MANAGEMENT APPROACH	184
TABLE 7-24: GOVERNANCE ASPECT.....	185
TABLE 7-25: GOVERNANCE TYPES.....	185
TABLE 7-26: GOVERNANCE ELEMENTS	186
TABLE 8-1: ASSESSMENT SCALE	190
TABLE 8-2: APLM PURPOSE.....	191
TABLE 8-3: APLM - COMPREHENSIVE	192
TABLE 8-4: APLM - RELEVANCE.....	193
TABLE 8-5: APLM - IMPORTANCE	193
TABLE 8-6: APLM - EMPIRICAL ANALYSIS RESULTS	195
TABLE 8-7: CHANGES AND RESPONSE.....	197

Abbreviations, Acronyms

AAIL	Agility Adoption and Improvement Lifecycle
AAIP	Agility Adoption and Improvement Process
AAIM	Agility Adoption and Improvement Model
AAIS	Agility Adoption and Improvement Scorecard
ASD	Adaptive Software Development
APEP	Agile Product-Enhancement Process
ASOP	Agile Service Oriented Process
ASSF	Agile Software Solution Framework
APLM	Agile Process Lifecycle Management
ASL	Application Service Library
BDI	Belief, Desire and Intention
CAM	Contextual Analysis Model
CAME	Computer Assisted Method Engineering
CMMI	Capability Maturity Model Integration
DSDM	Dynamic Software Development Method
FCs	Framework Characteristics
FaSS	Framework as Software Service
FDD	Feature Driven Development
FOI	Freedom of Information
FY	Flexibility
IEC	International Electrotechnical Commission
ISD	Information Systems Development
ISO	International Organization for Standardization
ITIL	Information Technology Infrastructure Library
KAI	Key Agility Indicators Index
LG	Learning
LS	Leanness
ME	Method Engineering
OPF	OPEN Process Framework
OMG	Object Management Group
OOSPICE	Object-Oriented Software Process Improvement and Capability

	dEtermination
PRINCE	Projects IN Control Environment
PMO	Project Management Office
RS	Responsiveness
SD	Speed
SDLC	Software Development Lifecycle
SDM	Software Development Methodology
SME	Situational Method Engineering
SE	Software Engineering
SEMMDM	Software Engineering Metamodel for Development Methodologies
SOA	Service Oriented Architecture
SPEM	Software Process Engineering Metamodel
SPI	Software Process Improvement
XP	eXtreme Programming
4-DAT	4-Dimensional Analytical Tool

Glossary

Agility	Agility refers to agility attributes such as speed, flexibility, learning, leanness and responsiveness as well as agile values and principles.
Agility Adoption	Agile process adoption in a non-agile environment.
Agility Improvement	Agile process improvement in an existing agile environment.
Abstraction	It refers to a logical view of an entity such as object, service, agent or component.
Agile Knowledge	It refers to the knowledge that a person or group knows about agile software development.
Framework	It refers to a set of software development element and components that can be combined to produce a tailored software process/method.
Multi-Abstraction	It refers to a situation/project which involves more than one abstraction such as object and agent or agent and service.
Method Engineering/ Tailoring	The method engineering and method tailoring have been used interchangeably in this thesis. [Although, it is acknowledged that in other areas of software engineering these terms are used differently – these two terms have been used interchangeably in order to avoid any arguments.]
Process/Method	The process and method have been used interchangeably in this thesis. [Although, it is acknowledged that in other areas of software engineering these terms are used differently. Each forum seems to have its own individual usage – these two terms have been used interchangeably in order to avoid these arguments.]
Process/Method Fragment	A smallest unit of software process or method.
Traditional/ Plan- Driven Methods	The traditional and plan-driven methods have been used interchangeably in this thesis. [Although, it is acknowledged that in other areas of software engineering these terms are used differently – these two terms have been used interchangeably in order to avoid any arguments.]
Practice	It refers to an activity or fragment in a software process/method.

Abstract

The innovative well-known agile methods offer many powerful agile software development practices and have received considerable attention from both practitioners as well as the research community. While many organizations are interested in adopting agile methods suitable to their local circumstances, there is little guidance available on how to do so. Organizations, especially on the large-scale, currently lack systematic support for adopting agile methods in their complex software development settings. To address this important issue, this research proposes an agile software solution framework (ASSF) to both assistance in the assessment of the capability of the organization or team and tailoring of agile method in order to support the systematic adoption and improvement of agility in both agile and, incidentally, non-agile software development environments - especially formal and large environments. The ASSF has been incrementally developed by the iterative application of build, review and adjust research activities, which is called here a “qualitative empirical” research method. The ASSF is intended for use by agile coaches and consultants as a comprehensive information guide. The ASSF has two main components: framework characteristics and lifecycle management. The framework characteristics component incorporates 10 main elements or attributes to describe the agile-hybrid software development methodologies: (1) people (2) process, (3) product, (4) tools, (5) agility, (6) abstraction, (7) business value, (8) policy (9) rules and (10) legal. The framework lifecycle management component specifies the stages, practices and resources in order to support the systematic adoption and improvement of agility. The framework stages refer to an agility adoption and improvement lifecycle, its practices refer to an agility adoption and improvement process, and its resources refer to models, templates and toolkit that can be used during the agility adoption and improvement process such as the contextual analysis model, a key agility indicators index, an agility adoption and improvement model, an agility adoption and improvement scorecard, and an agile toolkit. The components of this framework have been empirically analysed and reviewed by experts from industry as well as the research community, and updated based on the feedback received. The results of this research indicated that the proposed ASSF framework may be considered reasonable for a gradual successful transition or adoption of agile practices in formal and large software development environments.