

LB/087/03/08

Improving Agile Software Development: A Case Study on Adaptive System Development

Thesis Presented

By

G I U S Perera

Supervised by

Mr. M S D Fernando



University of Moratuwa, Sri Lanka.
LIBRARY
ELECTRONIC THESIS & DISSERTATIONS
UNIVERSITY OF MORATUWA, SRI LANKA
www.lib.mrt.ac.lk
MORATUWA

This thesis was submitted to the
Department of Computer Science and Engineering
of the
University of Moratuwa Sri Lanka
In partial fulfillment of the requirements for the
Degree of Master of Science

University of Moratuwa



90106

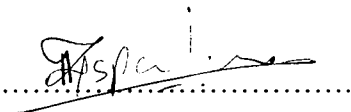
Department of Computer Science and Engineering
University of Moratuwa
Sri Lanka
January 2008

90106

90106

Declaration

“I certify that this dissertation does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any university; and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where due reference is made in the text”



G.I.U.S. Perera

UOM Verified Signature  University of Moratuwa, Sri Lanka.
& Electronic Theses & Dissertations
www.lib.mrt.ac.lk

Mr. M.S.D. Fernando

Supervisor,

Senior Lecturer,

Dept. of Computer Science and Engineering,

University of Moratuwa.

To



University *My Parents*, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk *and*

Teachers

Abstract

Software development is getting more and more complex with the improvement of technologies and tools for relative domains. Applying software solutions became an inevitable fact in the modern society. However, the domain conditions and the business environments change rapidly, by making the software norms obsolete. Furthermore, due to the intrinsic properties of software, situation becomes worse. Though the history confines to a few decades, software engineering owns dozens of software process paradigms to date. Introducing new process paradigms may resolve a selected problem, but not the others, and often introduces more issues. Moreover, it is evident that the improvements for the software processes only from the technical orientation do not solve all the issues in rapid changing environments.

Agile software process is a well known, lightweight, and flexible practice which was introduced as a remedy for the above crisis. In many aspects it serves its purpose. It introduced a paradigm shift to the software development for rapid delivery in uncertain environments. However, usage of well known Agile like software paradigms as it is in the practical conditions is arguable. The Agility concept endures certain flaws due to the native characteristics. Neither, it is the best software paradigm, nor the panacea for all software projects.

This research was derived with new perspectives from rational concepts in different domains, which were not yet been introduced to the Agile software process improvements. The research objectives were focused towards identifying existing bottlenecks in the Agile practices and potential improvements to those. Additionally, a case study was carried out in the mutual benefiting manner to the system's improvements and give flavor to the research with practical essence. The conducted study was successful in many aspects, and shows a significant impact to the Agile process. Furthermore, within the limited resource constraints the outcome of the study is really promising for the future study in the research area.

Acknowledgement

It is my great pleasure to acknowledge the contribution and assistance of a large number of individuals who made this research a complete and success. Without their help and guidance I won't be able to have a successful research experience. Though it is hard to appreciate everyone individually, there are some people who should be appreciated at least from a word.

First of all I would like to thank my supervisor Mr. Shantha Fernando, for accepting me as a research student and the guidance he provided. Despite from his busy schedule, he was always supportive and guiding me to the correct path. He was really helpful to me when I had doubts about the research activities. Specially, Mr. Fernando's valuable effort to drive me for an affluent research is highly appreciated.

My special thank is extended to Prof. Gihan Dias, who was my first M.Sc. research coordinator for the Dept. of CSE during the research, and Dr. Sanath Jayasena, the second M.Sc. research coordinator for my research. It was really a privilege to have such senior faculty members, which indeed helped me to improve my work.

I would also thank to Ms. Vishaka Nanayakkara, the Head of the Dept. of Computer Science and Engineering. She was always helped me with discussing LearnOrg related problems and potential enhancements. I specially thank her for allocating her busy time for discussing and helping me with my research.

My special thanks go to the progress committee members for this research, Mr. Raj Prasanna and Dr. Chandana Perera from the Dept. of Management of Technology. Their thoughts were really essential to add the management essence for this research.

I would like to thank all the other lecturers and staff members of the department of Computer Science and Engineering for the support they have given in numerous ways.

I should also thank my project members of the LearnOrg project and specially Mr. K. Sarveswaran, who was really helpful to me when I was doing my experiments. It was really a team experience when we work together.

Students who participated in the experiments are worth to mention for their honest and guileless data, which were really, helped me to make this research trustworthy.

I would like to thank my colleagues who studied with me for the M.Sc. at the Department. It was a wonderful experience to being with them.

Finally, I would like to be grateful to my parents and my brother, for encouraging me. They also helped me from all the way they could, even helping me to cover up my daily work when I was busy with this research.

In addition to the above, if anyone who is not mentioned there, I'm sorry for that. I appreciate everyone who helped me in my research.

Perera G.I.U.S.
Dept. of Computer Science and Engineering,
University of Moratuwa,
Sri Lanka.
10th January 2008

Table of Content

Declaration	ii
Abstract	iv
Acknowledgement	v
Table of Content	vii
List of Figures and Tables	xi
Symbols and Abbreviations	xii
Chapter 1 Introduction	1
1.1 Background	1
1.2 Problem Statement	2
1.2.1 Research questions	2
1.2.2 Research hypothesis – For experiments	4
1.3 Objectives of study	4
1.3.1 Justification	4
1.4 Significance of the study	5
1.5 Scope of the study	5
1.6 Structure of the thesis	6
Chapter 2 Literature review	8
2.1 Introduction	8
2.2 Software processes	8
2.2.1 Software process – What is it?	8
2.2.2 Why we need them?	9
2.3 Software process improvements	10
2.3.1 Problems with software processes	11
2.4 Software lifecycle engineering	12
2.4.1 Software lifecycle phases	12
2.5 Agile software development	13
2.5.1 Agile principle	14

2.5.2 Why Agile methods are better?	15
2.5.3. The areas to improve in Agile process	16
2.6 LEAN principle	17
2.6.1 LEAN principle – the framework	17
2.6.2 LEAN software development	19
2.6.3 Problems in LEAN production	21
2.7 Managerial decision making	21
2.7.1 Popular decision support methods	22
2.7.2 Motivation for development	23
2.8 Successful case studying	23
2.9 Concluding remarks	24
Chapter 3 The case study – System information	25
3.1 E-learning	25
3.2 Learning Management System – LMS	26
3.3 Content Management System – CMS	27
3.4 Generalized E-learning framework	28
3.4.1 The need of a generalized framework	29
3.4.2 The architecture	28
3.5 The LearnOrg system	31
3.5.1 The history	31
3.5.2 The current system	33
3.5.3 System problems	34
3.6 Concluding remarks	35
Chapter 4 Improvements	36
4.1 Problem background	36
4.1.1 Problems with Agile paradigm	36
4.2 LEAN – Agile hybrid model	37
4.2.1 Problems of LEAN software development	37
4.2.2 The model	38

4.2.3 The strengths of the model	39
4.3 R-C Matrix	40
4.3.1 The background	40
4.3.2 The model	42
4.3.3 Strengths of the model	44
4.4 Case suggestions	45
4.4.1 Solutions from other models	45
4.4.2 Suggested improvements – case specific	46
4.4.3 Rationalization of the improvements	46
4.5 Concluding remarks	47
Chapter 5 Experiment Methodology	48
5.1 Experiment background	48
5.2 Experiment 1: LEAN – Agile model	49
5.2.1 Experiment environment	49
5.2.2 Performance measures	50
5.2.3 Experiment rationalization	51
5.3 Experiment 2: R-C Matrix	52
5.3.1 Experiment environment	52
5.3.2 Performance measures	53
5.3.3 Experiment rationalization	53
5.4 Case study observations	54
5.4.1 Case study environment	54
5.4.2 Problem tracking	55
5.5 Concluding remarks	55
Chapter 6 Data and Analysis	56
6.1 Experiment 1 – LEAN-Agile hybrid process	56
6.1.1 Collected data	56
6.1.2 Analysis – Hypothesis testing 1	57
6.1.3 Analysis – Hypothesis testing 2	60

6.1.4 Analysis – Defects rate behavior	63
6.2 Experiment 2 – R-C Matrix evaluation	64
6.2.1 Collected data	64
6.2.2 Analysis – Fairness of the data	66
6.2.3 Analysis – Hypothesis testing	66
6.3 Case study information	68
6.3.1 Collected data	68
6.3.2 Analysis	69
6.4 Concluding remarks	70
Chapter 7 Discussion	71
7.1 Study outcomes	71
7.2 Experimental limitations	73
7.3 Summary	75
Conclusion	76
References	78
Annexes	87



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

List of Figures

Fig. 2.1 LEAN principles for success	18
Fig. 3.1 The architecture of the Generalized e-learning framework	30
Fig. 3.2 The University of Moratuwa e-learning system view	33
Fig. 6.1 MINITAB output - ANOVA results for hypothesis testing with LOC	59
Fig. 6.2 MINITAB output - ANOVA results for hypothesis testing with work levels	62
Fig. 6.3 Average defects rates in the examined time period for the two samples	63
Fig 6.4 MINITAB output - ANOVA results for the design phase times	66
Fig 6.5 MINITAB output - ANOVA results for hypothesis testing using T_0/T_2	67
Fig. 6.6 problem count for the LearnOrg during the study	70

List of Tables



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations

Table 2.1 The seven wastes of production	19
Table 2.2 Respective seven wastes in software development	20
Table 4.1 The R-C Matrix	43
Table 5.1: Project parameters and their results; collecting data	50
Table 6.1 a weekly data sample – normal agile practicing groups	56
Table 6.2 a weekly data sample – improved agile practicing groups	57
Table 6.3 sample A (normal agile) weighted sums of LOC	58
Table 6.4 sample B (hybrid agile) weighted sums of LOC	58
Table 6.5 sample A – deviation percentage from expected work amount	61
Table 6.6 sample B – deviation percentage from expected work amount	61
Table 6.7 Selected functional implementations without the matrix	65
Table 6.8 Selected functional implementations with the matrix	65

Abbreviations

AMI	Agility Measurement Index
ANOVA	ANalysis Of VAriance
BCG	Boston Consultation Group
CD - ROM	Compact Disk Read Only Memory
CMM	Capability Maturity Model
CMMI	Capability Maturity Model Integration
CMS	Content Management System
COCOMO2	COConstructive COst MOdel 2
CSE	Computer Science & Engineering
DEMP	Distance Education Modernization Project
DEPP	Distance Education Partnership Program
DSS	Decision Support Systems
IA-LMM	IT Aligned Learning Maturity Model
ICE	Information and Communication Engineering
ICT	Information and Communication Technology
ICTA	Information and Communication Technology Agency
IEEE	Institute of Electrical and Electronic Engineers
JIT	Just In Time
LAMP	Linux, Apache, MySQL, PHP
LM3	Learning Maturity Management Model
LMS	Learning Management System
LOC	Lines Of Codes
MOODLE	Modular Object Oriented Distance Learning Environment
NPV	Net Present Value
PEST	Political, Economical, Social, Technological
SCORM	Shareable Content Object Reference Model
SEI	Software Engineering Institute
SPICE	Software Process Improvement and Capability dEtermination
SPIN	Software Process Improvement Network
SWOT	Strengths, Weaknesses, Opportunities, Threats
VE	Value Engineering