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.

Complexity and Maintenance: A Comparative Study of Object-Oriented and Structured Methodologies

A thesis presented in partial fulfilment of the requirements

for the degree of

Master of Business Studies

in Information Systems at

Massey University

Michael Andrew Bailey

1996

ORIGINAL

ABSTRACT

Maintenance has been found to be one of the most expensive phases in the life of an information system. It has been suggested that the use of object-oriented methods instead of traditional structured methods may be one way of reducing the cost of maintenance required for an information system.

This thesis is an attempt to determine whether the object-oriented approach does in fact undergo a relatively smaller increase in complexity when subjected to a change in specifications than a similar system that is developed using a "structured methodology", and is therefore easier to maintain.

The methodologies used in this study were Yourdon's (1989) Modern Structured Methodology and Booch's (1994) Object-Oriented methodology. The analysis phase of both methodologies were applied to the same case study twice in order to evaluate the effects of a change in the system's specifications.

Once the two models for each methodology were complete, various metrics were applied to the structured system and a separate set of metrics were applied to the object-oriented system. The results of the models and the metrics were then analysed and validated in order to determine which system suffered a smaller proportional increase in complexity as a result of the changes to the system.

It was found that overall, the object-oriented system proved to undergo a smaller increase in complexity, and it was therefore easier to maintain as a result of the changes than the structured system.

ACKNOWLEDGEMENTS

Firstly, I would like to thank my Thesis Supervisor, Dr. Richard (Dick) Whiddett for providing invaluable assistance and support throughout the entire preparation of this Thesis. I am very grateful for his input into this thesis.

I would like to thank Mr. Barry Jackson for letting me borrow a guide on how to use Oracle's CASE tool. Without this I would not have been able to complete the structured models. I am also grateful for being able to use the Trusty Furniture Company case study.

I would also like to thank the students who provided the two structured systems used in the validation of this study.

A special thanks to my family for putting up with me for yet another year.

Finally, I would like to thank everyone else who has helped me during the preparation of this thesis who I haven't specifically mentioned (you know who you are).

Rational Rose and Rational Rose/C++ are trademarks of Rational Corporation

Oracle CASE is a trademark of Oracle Corporation

TABLE OF CONTENTS

	ABSTRACT		ii
	ACKNOWLEDGEMENTS		iii
	TABLE OF CONTENTS		iv
	LIST OF FIGURES		xiii
	LIST OF TABLES		xiv
1.	CHAPTER 1: INTRODUCTION		1
	1.1	Background	1
	1.2	Thesis	1
	1.3	Methodology	2
	1.4	Structure of the Thesis	4
2.	CHAPTER 2: OVERVIEW OF MAINTENANCE AND MEASUREMENT		6
	2.1	Introduction	6
	2.2	Definitions of Maintenance	6
	2.3	Definitions of Maintainability	8
	2.4	Overview of Maintenance	10
		2.4.1 Cost of maintenance	10
		2.4.2 Concepts of maintenance	11
		2.4.3 Obsolescence and maintenance	12

2.4.4 Strategies to remedy obsolescence 13
2.4.4.1 Restoration 14
2.4.4.2 Expansion 14
2.4.4.3 Elimination 14
2.4.4.4 Replacement 15
2.4.5 Non-monetary costs of maintenance 15
2.4.6 Types of maintenance 16
2.4.6.1 Corrective maintenance 17
2.4.6.2 Adaptive maintenance 17
2.4.6.3 Perfective maintenance 18
2.4.6.4 Preventive maintenance 18
2.4.7 An alternative categorisation of maintenance 19
2.4.7.1 Corrective maintenance 19
2.4.7.2 Deferred maintenance 20
2.4.7.3 Preventive maintenance 20
2.4.7.4 Emergency maintenance 21
2.4.7.5 File maintenance 21
2.4.7.6 Program maintenance 22
2.4.7.7 Scheduled maintenance 22
2.5 Systems Maintenance Lifecycle 23
2.5.1 Record and assign 25
2.5.2 Analyse and approve 26
2.5.3 Design and code 26
2.5.4 Test and train 27
2.5.5 Implement and monitor 28
2.6 Measuring Maintenance 29
2.6.1 Methodology independent metrics 29
2.6.1.1 Number of changes 29
2.6.1.2 Other methods 30

2.7 Measuring Object-Oriented Systems 31

2.7.1 MOOSE 32

2.7.1.1 Weighted methods per class 32

2.7.1.2 Depth of inheritance tree 34

2.7.1.3 Number of children 36

2.7.1.4 Coupling between objects 37

2.7.1.5 Response for a class 38

2.7.1.6 Lack of cohesion in methods 39

2.8 Measuring Structured Systems 41

2.8.1 McCabe's cyclomatic complexity 41

2.9 Comparisons Between the Metrics 42

2.10 How this Overview Relates to the Thesis 45

2.11 Summary 45

3. CHAPTER 3. THE CASE STUDY 47

3.1 Introduction 47

3.2 The Trusty Furniture Company Case Study 47

3.3 Description of the Changes Made to the Case Study 50

4. CHAPTER 4. STRUCTURED ANALYSIS DEFINITIONS AND TECHNIQUES 52

4.1 Introduction 52

4.2 Advantages of Structured Methods 53

4.3 Disadvantages of Structured Methods 54

4.4 Structured Analysis Definitions 56

4.4.1 Processes 56

4.4.2 Dataflows 57

4.4.3 Datastores 58

4.4.4 Terminators 58

4.4.5 Entities 59

4.4.6 Relationships 60

4.5 Overview of Yourdon's (1989) Modern Structured Methodology 61
4.5.1 The dataflow diagram 62
4.5.1.1 Processes 63
4.5.1.2 Terminators 64
4.5.1.3 Datastores 65
4.5.1.4 Dataflows 65
4.5.2 The entity relationship diagram 67
4.5.2.1 Entities 67
4.5.2.2 Relationships 69
4.6 Summary 69

5. CHAPTER 5. THE CREATION OF A STRUCTURED MODEL 70
5.1 Introduction 70
5.2 Statement of Purpose 70
5.3 Event List 71
5.4 Development of an Entity Relationship Diagram 71
5.4.1 Identifying the entities 72
5.4.2 Identifying the relationships 72
5.4.3 Identifying the attributes 73
5.5 Development of a Context Diagram 73
5.5.1 Identifying the external entities 73
5.5.2 Identifying the dataflows 74
5.6 Development of the Event Response Diagrams 74
5.6.1 Identifying datastores 75
5.6.2 Creation of an unlevelled dataflow diagram 75
5.7 Levelling the Dataflow Diagram 76
5.7.1 Level 0 76
5.7.2 Checking the dataflow diagrams 77

5.8 Implementing Changes to the System 78
5.8.1 Creating a new event 78
5.8.2 Modifying the entity relationship diagram 78
5.8.3 Modifying the dataflow diagram 79
5.8.4 Completing the changes 79
5.9 Application of Metrics to the System 80
5.9.1 Applying McCabe's cyclomatic complexity metric 80
5.9.2 Applying the coupling between objects metric 81
5.10 Summary 82

6. CHAPTER 6. OBJECT-ORIENTED DEFINITIONS AND TECHNIQUES 84
6.1 Introduction 84
6.2 What Does Object-Oriented Mean? 84
6.3 Advantages of Object-Oriented Methods 86
6.4 Disadvantages of Object-Oriented Methods 88
6.5 Basic Object-Oriented Concepts 89
6.5.1 Objects 90
6.5.2 Classes 92
6.5.3 Types 94
6.5.4 Abstraction 95
6.5.5 Encapsulation 96
6.5.6 Inheritance 98
6.5.7 Polymorphism 99
6.6 Overview of Booch's Methodology 100
6.6.1 Dimensions of analysis and design 101
6.6.2 Class diagrams 105
6.6.3 Object diagrams 107
6.6.4 Interaction diagrams 109
6.7 Booch: The Rational Approach 111
6.7.1 Requirements analysis 111
6.7.2 Domain analysis 112

6.8 Summary 113

7. CHAPTER 7. THE PROCESS OF CREATING AN OBJECT-ORIENTED MODEL 115

7.1 Introduction 115

7.2 Development of the Initial Class Diagram 115

7.2.1 Identifying the key classes 115

7.2.2 Identifying the relationships between classes 116

7.2.2.1 Identifying inheritance relationships 116

7.2.2.2 Identifying uses relationships 117

7.2.2.3 Identifying has relationships 118

7.2.2.4 Identifying association relationships 119

7.2.3 Defining classes and relationships 120

7.3 Development of the Interaction Diagrams 120

7.3.1 Identifying classes and objects 120

7.3.2 Identifying methods 121

7.3.3 Adding scripts to the diagram 121

7.4 Development of the Object Diagrams 122

7.5 Why the Module and Process Diagrams Were Not Done 122

7.6 Checking the Model for Consistency 123

7.7 Making Changes to the Object-Oriented System 123

7.7.1 Modifying the system function statement 123

7.7.2 Identifying the new classes and relationships 124

7.7.3 Creating a new interaction diagram 125

7.7.4 Generating a new object diagram 125

7.8 Applying the MOOSE Metric Set 126

7.8.1 Depth of inheritance tree 126

7.8.2 Number of children 127

7.8.3 Weighted methods per class 128

7.8.4 Coupling between objects 129

7.8.5 Response for a class 130

7.8.6 Lack of cohesion in methods 131

7.9 Applying MOOSE to the Modified System 133
7.10 Summary 133

8. CHAPTER 8. RESULTS 135
8.1 Introduction 135
8.2 How Comparisons Were Made 135
8.3 Structured Results 136
8.3.1 Complexity of the dataflow diagram 136
8.3.1.1 Analysis of the dataflow diagram results 137
8.3.2 Complexity of the entity relationship diagram 138
8.3.2.1 Analysis of the entity relationship diagram results 139
8.4 Object-Oriented Results 141
8.4.1 Analysis of the weighted methods per class results 142
8.4.2 Analysis of the depth of inheritance tree results 144
8.4.3 Analysis of the number of children results 146
8.4.4 Analysis of the coupling between objects results 147
8.4.5 Analysis of the response for a class results 149
8.4.6 Analysis of the lack of cohesion in methods results 151
8.5 Comparisons Between the Structured and Object-Oriented Models 154
8.5.1 Comparisons between the coupling between objects results 154
8.5.2 Comparing coupling between objects on an individual basis 155
8.6 Validation of the Metrics Used in this Study 156
8.6.1 Structured metrics validation 157
8.6.1.1 Dataflow diagrams 157
8.6.1.2 Entity relationship diagrams 158
8.6.2 Object-oriented metrics validation 161
8.7 Model Validation 162
8.7.1 Structured model validation 163
8.7.1.1 Dataflow diagram validation 163
8.7.1.2 Entity relationship diagram validation 170
8.7.2 Object-oriented model validation 172
8.8 Summary 176

9. CHAPTER 9. DISCUSSION 178

9.1 Introduction 178

9.2 Depth of Model Development 178

9.3 Weighting Many-to-Many Relationships 180

9.4 GQM 181

9.4.1 Goal 182

9.4.2 Question 182

9.4.3 Metric 182

9.5 Alternative Metrics 184

9.5.1 Token counts 184

9.5.2 Information flow complexity 186

9.6 Problems With the Study 187

9.7 Discussion of Results 188

9.8 Summary 191

10. CHAPTER 10. CONCLUSIONS AND FUTURE WORK 193

10.1 Conclusions 193

10.2 Future Work 195

11. APPENDICES

APPENDIX A: Outputs from Structured Model Before Changes 197

APPENDIX B: Outputs from Structured Model After Changes 213

APPENDIX C: Structured Model Metric Results 230

APPENDIX D: Structured Model Validation Results 233

APPENDIX E: Outputs from Object-Oriented Model Before Changes 236

APPENDIX F: Outputs from Object-Oriented Model After Changes 262

APPENDIX G: Object-Oriented Model Metric Results 292

APPENDIX H: Object-Oriented Model Validation Results 295

APPENDIX I: Comparison of CBO Results on an Individual Basis 298

12. BIBLIOGRAPHY 300

LIST OF FIGURES

Figure 1:	The Systems Maintenance Life Cycle	23
Figure 2:	Example of a Process	63
Figure 3:	Example of an External Entity	64
Figure 4:	Example of a Datastore	65
Figure 5:	Example of a Dataflow	66
Figure 6:	Example of a Composite Dataflow	66
Figure 7:	Example of an Entity	68
Figure 8:	Example of the Notations Used to Represent a Relationship in this Study	69
Figure 9:	The Models of Object-Oriented Development	101
Figure 10:	Class Icon	106
Figure 11:	Example of an Interaction Diagram	110
Figure 12:	The Inheritance Relationship	117
Figure 13:	The Uses Relationship	118
Figure 14:	The Has Relationship	119
Figure 15:	The Association Relationship	120
Figure 16:	Depth of Inheritance Tree for the class Order Document Set	127
Figure 17:	Number Of Children for the class Order	128
Figure 18:	Coupling Between Objects for the class Customer	130

LIST OF TABLES

Table 1: Results of McCabe's Complexity Metric Before and After Changes Were Made to the Structured Model 136

Table 2: Results of the Coupling Between Objects Metric Applied to the Entity Relationship Diagram Before and After Changes Were Made to the Structured System 138

Table 3: Summary of the Results From the MOOSE Metric Set Before and After Changes Were Made to the Object-Oriented Model 142

Table 4: Results of McCabe's Complexity Metric After Being Applied to the Student Models in Comparison to the Trusty Furniture Company's Order Processing System 158

Table 5: Results of the Coupling Between Objects Metric After Being Applied to the Student Models in Comparison to the Trusty Furniture Company's Order Processing System 160

Table 6: Summary of the Results From the Application of the MOOSE Metrics to Booch's (1994) Ordering System in Comparison to the Trusty Furniture Company's Order Processing System 161

Table 7: Summary of the Terminators Used on the Context Diagram 164

Table 8: Summary of the Processes Used on the Lowest Level of the Dataflow Diagrams 166

Table 9: Summary of the Entities Used in the Entity Relationship Diagrams 170

Table 10: Summary of the Equivalent Classes Used in the Class Diagrams 173