AN AOP BASED APPROACH
TO REALIZE AN ESB PRODUCT LINE
Aspect Oriented Framework for ESB (A04ESB)

W.D.I.P. KUMARA

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

This Dissertation was submitted to the Department of Computer Science and Engineering of the University of Moratuwa in partial fulfillment of the requirements for the Degree of MSc in Computer Science specializing in Software Architecture

Department of Computer Science and Engineering
University of Moratuwa
Sri lanka
February 2010

96420

ABSTRACT

Nowadays, enterprises are utilizing Enterprise Application Integration (EAI) technologies to automate business processes and the Enterprise Service Bus (ESB) is the current state of art of the EAI technologies. ESB products are heterogeneous in terms of architectures, technologies, and features. Therefore, with an approach that can produce each variation of the heterogeneous ESBs, an ESB vendor can dominate the ESB market. The customer base of the ESB is also heterogeneous in terms of preference technologies, business domains, and application requirements. Each customer domain wants an ESB tailored to its specific attributes to solve their integration problems in a scalable and robust manner. Hence, with an approach that can produce individualized ESB products, an ESB vendor can dominate the ESB market.

Apparently, the required approach should possess one important property: *mass customization* - the ability to create many variations of ESB products. The strategic re-use of assets is the enabler of the mass customization. hence, the commonality and the variability of the ESB products should be realized as reusable software elements. A software development paradigm named Software Product Line (SPL) has emerged to support these requirements and a SPL of the ESB is a suitable approach to exploit the heterogeneity in the ESB product and customer base.

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

The objective of the research presented in this thesis N to propose an approach to realize an ESB product line. The commonalty of the ESB products is mainly ESB services such as routing, transform, security, and monitoring, whereas the variability includes architectures, technologies, and features. This research leverages the concepts of the Aspect Oriented Programming (AOP) to identify, separate, and modularize the ESB services that crosscut the heterogeneous ESBs and to exploit the variability suitably to produce each ESB variation.

This research presents A04ESB, an aspect-oriented framework that is developed to enable the realization of an ESB product line. The A04ESB consists of an aspect weaver, an aspect library, and an aspect definition language. The aspect weaver takes a newer approach that leverages the Pipe and Filters architectural style to make it suitable for an integration middleware. Furthermore, this research presents an ESB product line proposal and a case study that uses the A04ESB in a real world ESB.

DECLARATION

"The work included in this report was done by me, and only by me, and the work has not been submitted for any other academic qualification at any institution"

Indika

Name: W.D.I.P Kumara (088271N)
Date: 2010.02.10

"I certify that the declaration above by the candidate is true to the best of my knowledge and that this dissertation is acceptable for evaluation for the Degree of M.Sc in Computer Science specializing in Software Architecture"

UOM Verified Signature

Project Supervisor: Dr. Chandana Gamage
Date: 2010.02.10

ACKNOWLEDGMENTS

The dissertation was a long journey. Without assistance, support, and encouragement from others, I would not be able to end that journey.

First, I would like to thank the project supervisor Dr. Chandana Gamage and the project coordinator Dr. Sanath Jayasena. Dr Chandana Gamage as the project supervisor guided me through this journey. Without such guidance, definitely this dissertation would not be successful.

Second, I want to thank all people at my workplace, especially Dr. Sanjiva Weerawarana, Mr. Ruwan Linton, Mr. Asankha Perera, and the WSO2 ESB team for helping me through providing time enough to concentrate on the dissertation work.

I also want to thank all M.Sc 08 colleagues and all people who provided me helps during my time at the Department of Computer Science & Engineering, University of Moratuwa.

Finally yet importantly, I am extremely grateful for my family for love and encouragement throughout my graduate years.

TABLE OF CONTENTS

DECLARATION i

ABSTRACT ii

ACKNOWLEDGMENTS iii

TABLE OF CONTENTS iv

LIST OF FIGURES viii

LIST OF ABBREVIATIONS x

Chapter 1 Introduction 1

1.1 Heterogeneity in ESB Products and Customers 2

1.2 The Problem and Proposed Solution 3

1.3 The AO4ESB Approach 3

1.4 Contributions 6

1.5 Related Work 6

1.6 Organization of the Dissertation 7

Chapter 2 Literature Review 8

2.1 Related Technology 8

2.1.1 Enterprise Service Bus (ESB) 9

2.1.2 Aspect Oriented Programming (AOP) 12

2.1.3 Domain Specific Language (DSL) 18

2.1.4 Adaptive Middleware 19

2.1.5 Software Product Line (SPL) 20

2.2 Aspects 20

2.2.1 Logging Aspect 21

2.2.2 Security Aspect 23

2.2.3 Caching Aspect 24

2.2.4 Exception Handling Aspect 27

2.2.5 Validation Aspect 28

2.2.6 Persistence Aspect 29

2.2.7 Transaction Aspect 30

2.2.8 Fault Tolerance – Reliability Aspect 31

2.2.9 Monitoring Aspect 31

2.2.10 Load Balance Aspect 33

2.3 AOP System Architecture 35

2.3.1 Proxy Factory Based Architecture 35

2.3.2 Policy Injection Application Block Architecture 36

2.3.3 Composition Filters Model 37

2.3.4 AO Framework in OOP Environment 38

2.4 Aspect Mining.......38
2.4.1 Need for Aspect Mining.......38
2.4.2 Aspect Mining Techniques.......39
2.4.3 Feature Modeling.......40
2.4.4 Aspect Mining and Refactoring.......40
2.4.5 Approaches for Analyzing Legacy Systems.......41
2.5 AOP Based Software Product Line.......42
2.6 Related Work.......44
2.6.1 Research on AOP and ESB.......44
2.6.2 AOP Frameworks for Middleware.......44
2.6.3 Reusable Component Libraries.......45
2.7 Discussion.......45
Chapter 3 AO4ESB Design and Implementation.......47
3.1 Aspect Definition Language.......48
3.1.1 Design Goals.......48
3.1.2 Design Decisions.......48
3.1.3 BNF Grammar for Language.......49
3.1.4 ADL Examples.......50
3.1.5 Implementation.......53
3.1.6 Discussion.......54
3.2 Aspect Weaver.......54
3.2.1 Design Goals.......54
3.2.2 Design Decisions.......55
3.2.3 Implementation.......56
3.2.4 Discussion.......60
3.3 Aspect Library.......60
3.3.1 Design Goals.......60
3.3.2 Overview.......61
3.3.3 Log Aspect.......62
3.3.4 Exception Handling Aspect.......64
3.3.5 Persistence Aspect.......65
3.3.6 Cache Aspect.......67
3.3.7 Monitoring Aspect.......70
3.3.8 Load Balance Aspect.......71
3.3.9 Crypto Aspect.......74
3.3.10 Proxy Aspect.......75
3.3.11 Discussion.......75

3.4 Integration Layer 76
3.4.1 Design Goals 76
3.4.2 Implementation 76
3.4.3 Discussion 78
Chapter 4 A Case Study with Apache Synapse 79
4.1 Goals 79
4.2 Background 80
4.2.1 Apache Synapse 80
4.2.2 AO4ESB Integration 80
4.3 Scenarios 80
4.3.1 Log Aspect 81
4.3.2 Exception Handling Aspect 82
4.3.3 Persistence Aspect 82
4.3.4 Cache Aspect 83
4.3.5 Load Balance Aspect 83
4.4 Discussion 84
Chapter 5 ESB Product Line Proposal 86
5.1 Goals 87
5.2 Domain Engineering 87
5.2.1 Product Management 88
5.2.2 Domain Requirement Engineering 88
5.2.3 Domain Design and Realization 88
5.2.4 Domain Testing 89
5.3 High Level Procedure for the Domain Design and Realization 89
5.3.1 Goals 89
5.3.2 Methodology 89
5.4 Application Engineering 92
5.4.1 Application Requirement Engineering 92
5.4.2 Application Design and Realization 92
5.4.3 Application Testing 92
5.5 Discussion 93
Chapter 6 Analysis and Discussion of Results 94
6.1 Summary of Results 94
6.2 Analysis of Results 95
6.2.1 Goal Oriented Analysis 95
6.2.2 Quality Attribute Analysis 97
6.3 Discussion 104

UNIVERSITY OF MORATUWA
LIBRARY
SRI LANKA

Chapter 7 Conclusions and Future Work 105
7.1 Contributions 105
7.2 Limitations 108
7.3 Future Work 108
7.4 Concluding Remarks 108
REFERENCES 109
APPENDIX A – The Case Study Resources 113

LIST OF FIGURES

Figure 1.1. The Current State of Art of the ESB Products 4
Figure 1.2. The Effect of the ESB Product Line 4
Figure 1.3. The Key Idea of the AO4ESB Approach 5
Figure 2.1. A Generic ESB Architecture 10
Figure 2.2. The Crosscutting Phenomenon 13
Figure 2.3. Aspects, Advices, Point-cuts and Joint Points 15
Figure 2.4. The Static Weaving 17
Figure 2.5. The Dynamic Weaving 17
Figure 2.6. The Projection of DSL into UI 18
Figure 2.7. A Design of the Logging Application Block 22
Figure 2.8. A Design of the Security Application Block 23
Figure 2.9. A Design of the Cryptography Application Block 24
Figure 2.10. A Design of the Caching Application Block 26
Figure 2.11. A Design of the Exception Handling Application Block 28
Figure 2.12. A Design of the Validation Application Block 29
Figure 2.13. A Design of the Data Access Application Block 30
Figure 2.14. The QoS Observer Pattern 32
Figure 2.15. SLA Monitoring 32
Figure 2.16. Scaling by Distributing Load across Multiple Service Instances 35
Figure 2.17. The Proxy Factory Approach 36
Figure 2.18. Handler Pipelines of the Policy Injection Application Block 36
Figure 2.19. The Basic Structure of the Composition Filters Model 37
Figure 2.20. An Aspect Oriented Framework Architecture 38
Figure 2.21. An Approach for Analyzing Legacy Systems 41
Figure 2.22. An Aspect-Oriented Product Line Framework 43
Figure 3.1. The BNF Grammar of the ADL 49
Figure 3.2. An ADL Example Defining Aspects 50
Figure 3.3. An ADL Example Defining Aspect Composition 51
Figure 3.4. An Aspect Composition in Detail 52
Figure 3.5. The ADL Token Parsing 53
Figure 3.6. The ADL Configuration Building 53
Figure 3.7. The Channel Structure 56
Figure 3.8. A Pipeline 57
Figure 3.9. A Component Proxy 57
Figure 3.10. The Aspect Managing Process 58
Figure 3.11. The Aspect Chain of Aspect-to-Main Weaving 58

Figure 3.12. The Aspect Chain of Aspect-to-Aspect Weaving 59
Figure 3.13. A Resolved Aspect Chain 59
Figure 3.14. The Design of the Log Aspect 62
Figure 3.15. The Design of the Exception Handling Aspect 65
Figure 3.16. The design of the Persistence Aspect 66
Figure 3.17. The Cache, Cache Storage, and Listeners 68
Figure 3.18. The Design of the Cache Controlling 68
Figure 3.19. The Design of the Cache Loading 69
Figure 3.20. The Design of the Control Bus 70
Figure 3.21. The Design of the Monitoring Event Propagation 71
Figure 3.22. The Design of the Session-less Load Balancing 72
Figure 3.23. The Design of the Session-aware Load Balancing 73
Figure 3.24. The Design of the Crypto Aspect 75
Figure 3.25. The Design of the Interceptors 77
Figure 5.1. The Empty-Component Approach to Produce Components of Different ESBs 91
Figure A.1. The Synapse Configuration for Scenarios 1-12 113
Figure A.2. The ADL for Scenario 1 114
Figure A.3. Aspect Weaving for Scenario 1 114
Figure A.4. The ADL for Scenario 4 115
Figure A.5. Aspect Weaving for Scenario 4 116
Figure A.6. The ADL for Scenario 7 116
Figure A.7. The ADL for Scenario 9 116
Figure A.8. The ADL for Scenario 11 117
Figure A.9. Aspect Weaving For Scenario 11 118
Figure A.10. The Synapse Configuration for Scenarios 13 and 15 119
Figure A.11. The ADL for Scenario 13 120
Figure A.12. The Synapse Configuration for Scenarios 14 and 16 120
Figure A.13. The ADL for Scenario 14 121
Figure A.14. Aspect Weaving for Scenario 14 122
Figure A.15. The Synapse Configuration for Scenarios 17-22 123
Figure A.16. The ADL for Scenario 17 124
Figure A.17. Aspect Weaving for Scenario 17 124
Figure A.18. The ADL for Scenario 18 125
Figure A.19. The ADL for Scenario 20 126
Figure A.20. The ADL for Scenario 22 127

LIST OF ABBREVIATIONS

AOP	Aspect Oriented Programming
DSL	Domain Specific Language
OOP	Object Oriented Programming
AO	Aspect Orientation
ESB	Enterprise Service Bus
AOSD	Aspect-Oriented Software Development
SPL	Software Product Line
EAI	Enterprise Application Integration
JVM	Java Virtual Machine
POJO	Plain Old Java Object
JMS	Java Message Queue
API	Application Programming Interface
WS	Web Service
SPL	Software Product Line
PDG	Program Dependence Graph
MDSD	Model Driven Software Development
AOM	Aspect Oriented Modeling
ADL	Aspect Definition Language
JBI	Java Business Integration
SCA	Service Component Architecture
MQM	Message Queuing Middleware
SPEL	Software Product Line Engineering
SOA	Service Oriented Architecture
SCDL	Service Component Definition Language
MOM	Message Oriented Middleware
SEDA	Staged Event Driven Architecture
ATAM	Architecture Tradeoff Analysis Method