

Context based approach to intelligent message passing



University of Moratuwa, Sri Lanka.
S.S.M. Silva
Electronic Theses & Dissertations
www.lib.mrt.08/10013

Dissertation submitted to the Faculty of Information Technology, University of Moratuwa, Sri Lanka for the partial fulfillment of the requirements of the Degree of MSc in Artificial Intelligence

September 2010

Declaration

I declare that this dissertation does not incorporate, without acknowledgment, any material previously submitted for a Degree or a 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 or myself except where due reference is made in the text. I also hereby give consent for my dissertation, if accepted, to be made available for photocopying and for interlibrary loans, and for the title and summary to be made available to outside organization.

S. S. M. Silva

Name of Student

Signature of Student

Date



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

Supervised by

Prof. Asoka S. Karunananda

Name of Supervisor(s)

Signature of Supervisor(s)

Date

Dedication



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

Dedicated to my beloved wife.

Acknowledgements

First and foremost I would like to thank my project supervisor Prof. Asoka S. Karunananda, Dean, Faculty of Information Technology, University of Moratuwa for all the encouragement, invaluable suggestions and most of all the constructive criticism. I would also like to extend my gratitude to my family members and friends for all their support and understanding. Last but not least I would like to thank all the researchers who have done their research work related to Web Service optimization and text classification based on AI areas.



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

Abstract

Being a widely used method of communication for distributed applications SOAP has a major drawback with its large message size. This is mainly because of the excessive number of tags being used on SOAP for data definition. On a frequently used web-service, some data of the message can be repeating from one message to other while only few fields of the message are changing. These characteristics of the message are totally depends on the nature of the application and the data associated with it. One cannot really predict on it at the time of the development of the web-service.

The hypothesis of this thesis is that messaging in web-services can be optimized by the concept of Negotiation in Multi Agent Technology. First we identify the XML tags as the static content of the message and automatically store as a message template to use in future messages. Then we process the XML node values to further identify the static content. We define two major Agents as Cluster Agent and Text Agent. Cluster Agent and the Text Agent negotiate with each other to perform a good cluster. Then we use Genetic Algorithms to improve the cluster. The common text pattern of the cluster is identified as the static content. After we identified static content pattern of the message we keep that pattern as a pre-knowledge of the message. Then on message transferring we delete the pre-identified static content from the SOAP message being transferred. At the receiver end we again apply the same pattern to build the original SOAP message.

We have tested the prototype using a sample web-service which gives a list of airports of a country when specified the country code. The test results have shown that it has been reduced 75% of the message on average by this method. Test results also show that the total time also reduced by our method and this reduction is proportional to the length of the message.

Contents

	Page
Chapter 1 - Introduction	1
1.1 Introduction	1
1.2 Web-service communication	1
1.3 Human communication as an example	2
1.4 Aim	2
1.5 Objectives	3
1.6 Proposed solution	3
1.7 Resource Requirements	4
1.8 Structure of the thesis	4
1.9 Summary	4
Chapter 2 - Current approaches to SOAP optimization	5
2.1 Introduction	5
2.2 Static context based approach	5
2.3 Differential parsing based approach	6
2.4 Comparison of encoders	6
2.5 Network adaptable middleware	7
2.6 Problem in brief	7
2.7 Summary	7
Chapter 3 - AI to SOAP optimization	9
3.1 Introduction	9
3.2 Text Classification using GA	9
3.3 Text classification using Swarm-Intelligence	11
3.4 Summary	12

	Page
Chapter 4 - SOAP Optimization using Multi Agents	13
4.1 Introduction	13
4.2 Users	13
4.3 Input to the system	13
4.4 Output of the system	14
4.5 Process	14
4.5.1 SOAP message optimizing process	14
4.5.2 Clustering process	15
4.6 Features	17
4.7 Summary	17
Chapter 5 - Analysis and Design	18
5.1 Introduction	18
5.2 Functional requirements	18
5.3 Analysis	19
5.3.1 Message Organizer	19
5.3.2 Pattern Recognizer	20
5.3.3 Pattern Distributor	21
5.3.4 Message Re-builder	22
5.4 Design	22
5.5 Ontology	24
5.6 Summary	24
Chapter 6 - Implementation	25
6.1 Introduction	25
6.2 Major technologies used.	25
6.3 XML Processing	26
6.4 Agents Implementation	27

	Page	
6.4.1	TemplateRequest	28
6.4.2	TemplateResponse	28
6.4.3	OptimizeRequest	28
6.4.4	OptimizeResponse	28
6.4.5	NewTemplateMessage	29
6.4.6	AllTemplateRequest	29
6.4.7	TemplateOptimizeRequest	29
6.4.8	MembershipRequest	29
6.4.9	MembershipResponse	30
6.4.10	MembershipConfirmRequest	30
6.4.11	MembershipConfirmResponse	30
6.4.12	PerformNewClusterRequest	30
6.4.13	PerformNewClusterResponse	30
6.4.14	ClusteringEndMessage	31
6.4.15	NewCluaterPatternMessage	31
6.4.16	AbstractSOAPAgent	31
6.4.17	SOAPMessageAgent	31
6.4.18	TemplateAgent	32
6.4.19	TemplateManagerAgent	34
6.4.20	TemplateBuiledrAgent	35
6.4.21	TemplateOptimizerAgent	35
6.4.22	ClusterAgent	35
6.4.23	TextAgent	37
6.5	Genetic Algorithms	40
6.6	MADkit kernel	41
6.7	Ontology	42
6.8	Clustering Viewer	42
6.9	Summary	42

	Page
Chapter 7 - Practical Usage	43
7.1 Introduction	43
7.2 How to write the plug-in	43
7.2.1 Server side plug-in	43
7.2.2 Client side plug-in	44
7.3 Summary	44
Chapter 8 - Evaluation	45
8.1 Introduction	45
8.2 Experiment setup	45
8.3 Test results	46
8.4 Summary	48
Chapter 9 - Conclusion and Further Work	49
9.1 Introduction	49
9.2 Conclusion	49
9.3 Problems encountered	50
9.4 Further improvements	50
9.5 Summary	51
References	52
Appendix A	55
Appendix B	58
Appendix C	62
Appendix D	64

List of Figures

	Page
Figure 5.1 - Message flow diagram of core modules	20
Figure 5.2 - Data flow diagram of message sender	21
Figure 5.3 - Data flow diagram of message receiver	21
Figure 5.4 - Architecture diagram – Message optimization process	22
Figure 5.5 - Architecture diagram – data clustering	23
Figure 6.1 - Agent collaboration diagram when new type of message arrives	38
Figure 6.2 - Agent collaboration diagram during data clustering	39
Figure 6.3 - Agent collaboration diagram when pre defined template exists	39
Figure 8.1 - Data reduction ratio comparison graph	47
Figure 8.2 - Messaging time comparison graph	47
Figure B.1 - Data clustering – Text agents seek for cluster	58
Figure B.2 - Data clustering – A cluster agent response for requests	59
Figure B.3 - Data clustering – One cluster has been performed	60
Figure B.4 - Data clustering – Cluster agents response for new requests	60
Figure B.5 - Data clustering – Clustering has finished with two clusters.	61
Figure C.1 - Sample SOAP Message	62
Figure C.2 - Sample Optimized Message	63
Figure D.1 - Sample Template File	64

List of Tables

	Page
Table 8.1 - Summary of evaluation data	46
Table A.1 - Data collected during evaluation	56
Table A.2 - Evaluation data comparison	57



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