

Introduction

1.1 Introduction

In this chapter it will be discussed about the web-service communication, its advantages and how important to optimize it. Also it has taken the human communication as an example for context based communication. Then it has mentioned the aim and objectives of this project and finally it has summarized the proposed solution.

1.2 Web-service communication

Inter-process communication is a vital factor in the distributed world of software applications. Distributed software applications have become more and more popular because of its advantages including increased productivity, cost efficiency, improved business decisions and flexibility and scalability [5]. With more added advantages of improved interoperability, open standards and easy to work on internet, Web Services has become the most widely used inter-application communication method nowadays.

SOAP – Simple Object Access Protocol was originally developed by Microsoft, IBM, DevelopMentor and Userland Software and was then submitted to the Internet Engineering Task Force (IETF). IETF eventually made it an official recommendation. SOAP is a simple solution for interaction of different applications built in different languages and running on different platforms as it uses HTTP as its transport and XML as its payload for sending and receiving messages. It is a lightweight and a loosely coupled protocol for exchange of information in a decentralized and a distributed environment.

Although the simplicity of Web services is an advantage in some respects, it can also be a hindrance. Web services use plain text protocols that use a fairly verbose method to identify data. This means that Web service requests are larger than requests encoded with a binary protocol. The extra size is really only an issue of low-speed connections, or over extremely busy connections [1]. Most of the commercial applications do pass the same message over and over with having only part of the

message changed. Most of the time this changing part is depends on the context of the message that is being passed. The static content of the message is being passed unnecessarily because machines cannot identify and filter them to improve the communication.

1.3 Human communication as an example

In relation to human communication when two people are having a discussion, at the start of the discussion the messages are more descriptive about the subject that is being discussed. As discussion progress the messages become shorter and shorter and only the context relevant to the new details being transferred with some reference to the previous knowledge. The complete message is being rebuilt in the receiver with his intelligence and the knowledge being gathered throughout the conversation. Same is applicable when communicating, two organizations communicate for cooperation. The message is being transferred is only a fraction of the required knowledge [8]. G. Andrighetto and P. Assogna address the problem of the mutual understanding from an anthropomorphic view point on their paper [8]. In this paper they discuss about the communication which occurs between people as well as organizations in relation with the knowledge and the context they are talking about. They have used thoughts of Ernst von Glasersfeld [7], Gian Battista Vico [10] J. B. Plotkin [22] and many other researches for this discussion. Furthermore it also states that “when two organizations communicate for cooperation, a message usually transports only a fraction of the required knowledge”

But when it comes to SOAP this is not the case. The complete knowledge about the message is being transferred on the message in order for the receiver to be able to understand it.

1.4 Aim

To build a multi agent system to identify the redundant parts as the static content of the SOAP message and then optimize the SOAP message by replacing the pre-identified static part by a key and then transfer the other content.

1.5 Objectives

- To study about context based communication
- To study about multi agent models for data clustering
- To design and develop a prototype context based message passing system
- To evaluate the prototype using web services
- To document the final thesis

1.6 Proposed solution

The SOAP optimizer tool will be used by web service developers as a plugin to their web-service applications. But during the operation of the system there are no human interaction involved with the system and therefore the users of the system will be the web-service applications to which the system is plugged into.

There will be two parts of the system. One is running at the web-service client side and other one is running at the application server. When a message is sent from either the client or the service the input to the system at the sender side will be the SOAP message. At the message receiver side the input will be the optimized message that was transferred through the network.

Output of the system at the message sender side will be the optimized message generated based on the previously identified static content patterns (templates). The output of the system at the message receiver side will be the original SOAP message from which the optimized message is generated.

When a SOAP message is due to transfer through the network first the system will search for previously identified static content templates. If it is possible to identify such a template then system will use that template and do the message optimization. Otherwise system will submit the current message as a candidate to find static content template. System will automatically generate the static content templates by processing the SOAP message. Then it will distribute the template to all the nodes on the application environment (all the application servers and clients running at the moment). At the receiver end, if the message is an optimized message then it will

regenerate the original SOAP message using the same set of templates that was used to optimize the message.

1.7 Resource Requirements

JAVA is used as the programming language to build the prototype system. Eclipse IDE is used as the development environment and MADKit and xmlbeans are used as third party libraries. A standard computer has been used to develop the application. Then using another computer as the host to run the application server with the web-service the prototype has been tested.

1.8 Structure of the thesis

The chapter 2 of this thesis contains the current approaches taken to optimize the web-service communication. In chapter 3 it has been discussed the two main technologies used as the solution. That is genetic algorithms and multi agent systems. The proposed approach to optimize the web services is described in chapter 4. Chapter 5 of this thesis contains the details about the analysis and design phase of the project. The implementation is described on chapter 6. The system is distributed as a JAVA library. How to use this library and how to plug it into another web-service is described on practical usages on chapter 7. Chapter 8 contains the evaluation procedure and the evaluation results. The conclusion is included in the chapter 9.

1.9 Summary

In this chapter it has given an introduction about the web-services. Then it has discussed about the communication in relation with human communication and how it is being representing the knowledge. Then it has presented the aim and objectives of this project. After that it has discussed the proposed solution and the resource requirements. Finally this chapter has presented the structure of the rest of this thesis. In next chapter it will be discussed about the current approaches to optimize the web-service communication.