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.

METAPHOR-ENABLED INTERFACE ARCHITECTURES

by

Haseeb Quazi

A thesis submitted in partial fulfilment of
the requirements for the degree of

Masters of Information Sciences

Massey University

2005

Approved by

Alexei Tretiakov
Supervisor

_____

_____

_____

Date

Massey University

Abstract

METAPHOR-ENABLED INTERFACE ARCHITECTURES

by Haseeb Quazi

Supervisor: Alexei Tretiakov, Department of Information Systems

TABLE OF CONTENTS

Chapter 1 1
1. Introduction 1
1.1 Interfaces 2
1.2 Metaphors 3
1.3 User Types 4
1.4 Architectures 6
1.5 Technologies 6
1.6 The objective of this Research 7
Chapter 2 9
2. Literature Review 9
2.1 Metaphor Theory 9
2.2 Incorporating Metaphors in Systems design 10
2.3 Interface Metaphor Usability 13
2.4 Personalisation of Metaphors in websites 13
2.5 Domain-Oriented Interface Metaphors 15
2.6 Interface Architectures 18
2.6.1 Seeheim Model of UI Architecture 18
2.6.2 Model View Controller (MVC) architecture 20
2.6.3 JSP Model 1 and Model 2 Architectures 21
2.7 Clickstream Analysis 23
2.8 Service Agents 24
2.9 Intercepting filter 25
Chapter 3 26
3. Analysis and design of the protoype 26
3.1 Research Area 26
3.2 Project content 26
3.3 Project Scope and Feasibility 27
3.4 Prototype Analysis 27
3.4.1 Conceptual model for a system managing metaphorical structures in WIS 28
3.4.2 Design Analysis 29
3.4.3 Internal Behaviour 32
3.4.4 Configuration Requirements 32
3.5 Database Design 33
3.6 HERM 34
3.6.1 Database Schema: 36
3.6.1.1 Level 0: (Entities) 36
3.6.1.2 Level 1: (Relationships) 36

3.6.1.3 Relations: 37
3.6.2 List of Entity Types 38
3.6.2.1 Metaphor 38
3.6.2.2 Customer 39
3.6.2.3 Customer_Type: 39
3.6.2.4 Business Functions: 40
3.6.2.5 Metaphor Explanation 41
3.6.3 Relationship Types 41
3.6.3.1 Metaphor Allocation 41
Chapter 4 48
4. Implementing the prototype 48
4.1 Proposed System 49
4.2 Database and Java connection 50
Chapter 5 54
5. Evaluating the prototype 54
5.1 Introduction 54
5.1.1 Hypothesis 54
5.1.2 Crossover Trial Design 55
5.1.2.1 Identification of benefits 56
5.1.2.2 Evaluation Population 56
5.1.2.3 Sampling Procedure 57
5.1.2.4 Evaluation Method 57
5.1.2.5 Questionnaire Design 59
5.1.2.6 Evaluation-Questionnaire Implementation 60
5.2 Expected Results from the questionnaire. 61
Chapter 6 63
6. Conclusions 63
Bibliography 72
Appendix 78

LIST OF FIGURES

Number *Page*

Figure 1: Model 1 architecture (Cavaness, 2004) 22
Figure 2: Model 2 architecture (Cavaness, 2004) 23
Figure 3. Initial concept. 28
Figure 4. Use-case diagram showing a Metaphor-enabled system with respect to a banking domain. 30
Figure 5. Example of documented textual use cases 31
Figure 6: Conceptual Model of the Metaphor Database. 35
Figure 7: Transfer Funds example 40
Figure 8: Oracle iSQL*Plus interface 43
Figure 9: Querying the Metaphor Database through the Oracle iSQL*Plus Interface 44
Figure 10: Proposed Metaphor-enabled Interface Architecture 48
Figure 11: Table of data from the evaluation exercise. 63
Figure 12: Graph showing the individual and average time spent doing the tasks. 64
Figure 13: Graph showing the time taken to do the tasks for the different User types in the New Zealand lot 65
Figure 14: Graph showing the time taken to do the tasks for the different User types in the Pakistan lot. 66
Figure 15: Graph showing the time taken for user who preferred both visual and textual websites. 67
Figure 16: Graph showing the time taken for user who preferred graphical style websites 68
Figure 17: Graph showing the difficulty levels for the New Zealand lot with Metaphors enabled and disabled 69
Figure 18: Graph showing the difficulty levels for the Pakistan lot with Metaphors enabled and disabled 70

ACKNOWLEGMENTS

Praise be to Almighty God, Lord of the Universe.

First and foremost I would like to thank God for the motivation and guidance that He has given me to make this report possible.

My deepest gratitude and thanks is made to the following people in one way or another who have helped in the completion of this report:

To my supervisor Alexei Tretiakov, for his inspiration, guidance and expert advice, whom without none of this would have been possible.

To Dr Roland Kaschek for his valuable assistance and information with the literature of my study.

Finally, I am deeply indebted to my family, who have been the motivational force in my life. I thank them for their prayers, love and support.

Chapter 1

1. INTRODUCTION

In today's contemporary society, the most common method in searching for information is through Web Information Systems (WIS). The Internet is open to people regardless of the different backgrounds one may come from e.g. ethnicities, gender, age, culture etc.

The Internet is also widely used because of its simple layout and to an extent the language used is easy to understand. Although this may not be true in all cases. Some of the key websites such as e.g. Google, Yahoo and Wikipedia are designed to have easy and quick access to information.

Most languages incorporate metaphors and to an extent, so does the Web. Users interact with metaphors and respond to them in different ways and also they can search for information on the Web and use different web-based applications in their daily life such as Internet banking, online flight booking, online libraries etc. The information presented to the users is mainly controlled by the website and it might not be presented in a favourable manner that may suit the user.

Metaphors can be introduced into websites to enhance the presentation of the information and the way users interact with websites. The aim of this thesis is to identify an approach where metaphors can be employed as an extension of websites and adapt to the different user types in order to make their interaction with the website more efficient and effective.

Metaphors have been used in the computing domain since the early days (Jacko & Stephanidis, 2003). One of the common metaphors used in computing is that of the desktop we see when to turn on our computer.

Metaphors can relate concepts from one domain to another. For example a novice user of a particular domain may not be well acquainted with its concepts but using metaphors such as relating the concept at hand to one already known to the user. Metaphors are based on universal or local knowledge of the users and relating to already established concepts could help the users in understanding and grasping new concepts.

Deploying metaphors onto websites is a concept that is still in its early stages, hence metaphors should be carefully chosen. Although we assume that well-chosen metaphors will make the user interaction with a website or any domain much easier and clear. On the contrary, badly chosen metaphors can lead to confusion and misconceptions.

This thesis will give us insight into the assumptions made here.

1.1 Interfaces

The noun *interface* refers to be a discrete and tangible feature that we can map, draw, design, implement and attach to an existing bundle of functionality. We naturally visualise an interface as the place where contact between two entities occurs. An interface is a contact surface. It reflects the physical properties of the interacting entity, the functions to be performed and the balance of power and control.

When designing Interfaces the first and most important question to ask is, *what does the user want to do?* There is still much debate that agrees and disagrees on the

designing of Interfaces being a separate thing from applications engineering (Jacko & Stephanidis, 2003).

In the context of this thesis, several Interface Architectures will be investigated which will lead to the proposal of a metaphor enabled interface architecture. Metaphors can be developed for particular domains and user types. In addition, Metaphors designed for particular domains and user type will allow other systems of the same domain to use them. For example, different banks can use metaphors designed for the banking domain. Therefore, the concept is somewhat of a plug and play technology where a metaphor interface application can be plugged into the existing system and subsequently metaphors will be invoked on the existing systems front end.

However in reality, software and hardware specifications have to be met by the domains system in order to access such applications.

In order to make a solid start to the project "Metaphor-Enabled Interface Architectures", selection of a domain for research had to take place. As the domain of electronic/Internet banking gains popularity, some initial investigation was undertaken regarding the way the users interact with such websites. Many banking websites were visited in order to gain an understanding of how the different applications worked. This led to the question of whether using metaphors in website interfaces will make user interaction more effective and efficient.

The next section is the history of the induction of metaphors into user interfaces.

1.2 Metaphors

Metaphors are frequently used in our everyday language. The characteristics of the metaphors in our language are the bases of how metaphors work in an

interface. Metaphors occur throughout interfaces that we come across in our daily computing. Taking the example of the email is one of the metaphors, which has been well established in the Internet domain. Emails can be compared to the real life scenario of receiving mail in a mailbox. A user who has a mailbox on their computer can open this to receive their mail. The technology that sends the message to and fro can be taken as the virtual postman.

Not all interface metáphors are easy to use; some can be misleading. For example, a metaphor debated till now is the Trashcan on the Macintosh desktop. In order to eject a disc inserted into a disk drive one has to drag and drop the disks image on the desktop onto the Trashcan. This cannot be compared to someone throwing a disc into a Trashcan in real life. Hence it is vital that Interface metaphors are well designed in order to make the user interaction with system easier.

It should be taken into consideration that there are thousands of domains that use their products over the web. This thesis will concentrate on targeting a particular domain in order to get the best possible proposal for a Metaphor-enabled Interface architecture.

One of the most important aspects in designing such architectures is to look into the way users interact with the system. The next section discusses the importance and use of User types in the development of such architectures.

1.3 User Types

Over the years, studies on the usability of hundreds of product and web site designs have been carried out. There have been designs that were incredibly effective for users and designs that fell tremendously short. One emerging pattern in our ongoing research is that design teams that know a lot about their users are more likely to produce user experiences where the systems are usable,

effective, and pleasing, as users can think in rather familiar ways as compared to a new domain (Kobsa, Koenemann & Pohl, 2001).

Using interface metaphors make the users more comfortable to understand a system. There are diverse users using the Internet to obtain the information they need. These users do not think in similar fashion and they are most likely to carry out tasks in different ways. Therefore, to cater for the different user types, systems could make use of metaphors and mapping them to the compatible users.

Metaphors have been used in teaching novice users as they are based on common knowledge. Different domains have different users and capturing the best way to present the information or functionality to them may be effectively done through the use of metaphors. It must also be made clear that all functionalities of systems may not be modelled with a metaphor.

Capturing information about the users of a particular system can make customisation of Interfaces more suitable and effective. User Types for different domains can be categorized by different information about the user. The term user data introduced by (Kobsa, Koenemann & Pohl, 2001) is to denote information about personal characteristics of the user, while the term usage data is related to user's behaviour.

The user data is directly obtained from the user while the usage data is taken from observing the users movements/behaviours. There have been systems developed using such techniques to obtain valuable information about users (Kobsa & Wahlster, 1989). In this thesis an architecture will be proposed, which will cater for adaptively viewing metaphors according to different customer types.

1.4 Architectures

"The architecture of a software system defines that system, in terms of computational components and interactions among those components" (Shaw & Garlan, 1996). In information sciences, architecture is a term applied to both the process and the outcome of thinking out and specifying the overall structure, logical components and the logical interrelationships of a computer its operating system, a network, or other conception.

Architecture is a model guiding implementation. Usually architecture is dealt with after a conceptual model has been obtained. Once the conceptual model is obtained, the architecture is then validated against it. Architectures consist of Syntax (structure of the architecture), Semantics (meaning of components and interactions) and Pragmatics (reasons behind structure & meaning).

In this thesis our aim is to propose an architecture for the Metaphor-Enabled Interfaces by formulising a conceptual model and validating it against an implementation of the Architecture.

As a result various architectures will be investigated in this thesis in order to develop the best possible architecture for the Metaphor-Enabled Interfaces.

1.5 Technologies

There are a wide range of technologies, which can be used in the development of such systems. The main focus of this thesis is to work towards the proposal of a metaphor-enabled web-based interface architecture. In this case technologies such as XML and XSL currently are in great demand.

Trausan, Novischi, Cerri & Maraschi, have developed a system for processing personalized Metaphors on the Web for learning a Foreign Language. XML was used for the annotation of Metaphors and XSL for visualization (Trausan,

Novischi, Cerri & Maraschi, 2000). Other server side technologies have to be taken into consideration so that the dynamic functionality of the system could be handled. In this case, technologies such as Java Server Pages, Active Server Pages, Servlets, PHP, Java Applets and Java Beans will be investigated and subsequently one will be chosen to develop a working prototype of a Metaphor-Enabled Web-based Interface Architecture. In further chapters various technologies will be looked into in detail in order to compare and evaluate in the areas of functionality compatible to the chosen domain.

1.6 The objective of this Research

The aim of this research is to firstly propose a conceptual model that will map User Types to metaphors. The Higher-order Entity - Relationship Modelling (HERM) language was used to formalise the model. The conceptual model also helps in the creation of metaphors because other data regarding metaphors is required in order for it to fit into the model. The model also helps with User Type creation in a similar fashion.

Secondly it was required to design and develop a Metaphor-Enabled Web-based Interface Architecture. Metaphors have been a part of the interfaces with respect to computers from the early days. Now businesses are coming towards the Internet and the confidence is building as more and more commercial sites are setup everyday (Coffman & Odlyzko, 2002) (Odlyzko, 2003).

This means the user base will also expand, bringing diverse users to the already diverse Internet. In order to cater for such a diverse audience, an architecture can be designed where common knowledge is displayed as Metaphors. This will make the interaction easier and customisable, which will be a great step towards making websites friendlier. This can eventually bring in more customers and more returns to businesses. Therefore with the proposed conceptual model a

Metaphor-Enabled Web-based Interface Architecture was proposed that enables metaphors on a website to be invoked according to the User types.

Analysis on the feasibility of the architecture and the conceptual model are given throughout this thesis and a formal user evaluation exercise took place in order to validate the concept.