Visualization of Interface Metaphor for Software:

An Engineering Approach

Dinesh S. Katre

DISSERTATION.COM

Boca Raton

Visualization of Interface Metaphor for Software:
An Engineering Approach

Dissertation.com
Boca Raton, Florida
USA • 2011

ISBN-10: 1-59942-377-4
ISBN-13: 978-1-59942-377-7

प्रज्ञा विवेकं लभते भिन्नैरागमदर्शनै :।
कियद्वा शक्यमुन्नेतुं स्वतर्कमनुधावता ।।
भर्तृहरि : वाक्यपदीय

By adopting and sharing multi-disciplinary approach,
knowledge is transformed into wisdom.
How much can one imagine by simply employing one's own logic?

Bhartruhari, Vākyapadiya, 5th Century AD

Acknowledgements

I am very grateful to my supervisor Dr. Mukul K. Sinha, Managing Director, Expert Software Consultant Ltd., New Delhi for mentoring me with compassion during the five years of doctoral research. He is the key motivator for me, who recognized my interest in research and boosted my spirit to work harder in that direction. I am thankful to him for his valuable guidance, counseling and help in difficult situations.

I am extremely thankful to C-DAC for its consistent support and encouragement to me during this research.

It is beyond my capacity to even express how much I owe to *Saint Dnyaneshwar (1275-1296),* whose metaphorical verses from *Dnyaneshwari* kept me enchanted for so many years and have been a great source of inspiration for me in this research on interface metaphors. I thank Sri Sakhare Maharaj for helping me in understanding the spiritual metaphors from *Dnyaneshwari*. I am indebted to Dr. Vijay Bhatkar, founder Executive Director of C-DAC, who entrusted the responsibility of multimedia rendering of *Dnyaneshwari* and introduced me to Shri Sakhare Maharaj.

I am grateful to Dr. Aaron Marcus, President, Aaron Marcus Associates, USA for appreciating my efforts and guiding me during his visit to India. I reproduce his comments on one of my articles [Katre, 2002] related to this dissertation as "I have for a long time thought that India and China might produce new applications that derived their metaphors from local culture/history/literature. Your article points the way. What seemed intriguing was to view a list of these metaphors and religious/historical references, and to compare them with some standard Western approaches. Your comparison at the end of categories or types of metaphors was useful." I am so thankful to him for his encouraging remarks and highlighting the uniqueness of this work.

I am thankful to Prof. V. N. Jha, Director, Center for Advanced Study in Sanskrit, University of Pune for sharing the knowledge of linguistic metaphors in *Vedic* literature. I also thank Dr. Mrs. Jha for providing appropriate reference from *Vedic* literature related to metaphors. I am extremely thankful to Prof. K. Srinivas Rao from IMSc, Chennai for his wisdom tips and guidance in improving my writing skills.

I am thankful to Film and Television Institute, Pune, SNDT University, Mumbai and Industrial Design Center, Indian Institute of Technology, Mumbai for organizing my lectures on the topic of research.

I express gratitude to Birla Institute of Technology and Science (BITS), Pilani for offering an opportunity to a professional like me to pursue research through its off-campus doctoral program. I also thank the staff of BITS, Pilani namely Dr. Ravi Prakash, Dean, Research and Consultancy division; Dr. Rahul Banerjee, Dr. Shan Balasubramanian and Dr. S. D. Pohekar for their guidance and support.

I am indebted to my colleague and friend Mr. Devashish Pandya for volunteering to be a sounding board for my ideas. He always brought new insights to me while brainstorming and debating with him. I also thank all the subjects who patiently participated in the tests that I performed and provided their valuable inputs.

Last but not the least, I gratefully acknowledge the trademarks belonging to their owners that are mentioned during our discussion and also the creators of the UI screenshots which are used as examples in appendices.

Dinesh S. Katre

Table of Contents

APPENDICES

List of Tables & Figures

Abbreviations

HCI	Human-Computer Interaction
UXD	User Experience Design
CIM	Candidate Interface Metaphor
CDM	Cross-Domain Mappings
UCs	Unmapped Concepts
URs	Unmapped Requirements
CTP	Commentary of Task Performance
IPS	Interface Play Script
DT	Dissection Termination
UDL	User Domain Lexicon
ADL	Application Domain Lexicon
UI	User Interface
IPS	Interface Play Script
SDLC	Software Development Life Cycle
UCD	User Centered Design
UEM	Usability Evaluation Method

Abstract

This dissertation presents a comprehensive process for ***visualization of interface metaphor*** for software, which is helpful in designing interactive user interfaces with magical super-affordances and definitive user experiences. The steps of this process are integrated with the ***Waterfall Model of Software Development Life Cycle (SDLC).*** It mainly focuses on ***pre-facto analysis, quantitative and qualitative evaluation*** of interface metaphor to be performed during ***Requirements Engineering (RE)*.** In this, the *candidate interface metaphors* are identified within the ***common knowledge dimensions*** of specified users. ***Commentary of Task Performance (CTP)*** and ***Interface Play Script (IPS)*** are written similar to a drama script; and then juxtaposed for capturing the ***mental model*** and ***sensory details*** of the tasks. The *candidate interface metaphors* are classified as ***coherent*** and ***diverse*** based on the interrelationships coded in the ***cognitive map***. The conceptual structures are further categorized at different levels as ***Domain, Conglomerate, Multitudinous, Singleton (Inanimate, Animate),*** and **Flat Concepts** to help in assessing their potential. Various aspects of the selected *candidate interface metaphors* are identified and tuned based on the determinants of software and user for crafting the desired user experiences. The ***analysis*** of interface metaphor is performed through its ***concept-by-concept dissection***. ***Cross-Domain Mappings (CDMs)*** are formed between the ***resonating*** metaphoric concepts and software requirements. The dissected conceptual structure is quantified to identify the ***Cross-Domain Mappings (CDMs), Unmapped Concepts (UCs), Unmapped Requirements (URs), Usable Conceptual Bandwidth and Coverage.*** After this ***User*** and ***Application Domain Lexicons*** are built to help in designing the ***linguistic metaphors*** to be used in the user interface. Interdisciplinary ***Usability Heuristics*** and ***Usability Indicators*** are identified for the ***qualitative evaluation*** of interface metaphor. ***Quantitative metrics*** is provided for quantifying the results of heuristic evaluation. Interface metaphor is given a ***tangible form*** through multimedia rendering. The documents and artifacts generated through the visualization process are linked with it for corroboration of results. Usability heuristics are applied and tested on the multimedia rendering. A ***Remote Usability Testing method*** is developed for evaluating the cross-cultural issues. The interface metaphor is then accepted for incorporation in the design of software after satisfactory clearance through usability evaluations and tests.

Chapter 1. Introduction

This dissertation presents a comprehensive process for ***visualization of interface metaphor*** for software applications. The steps of this process are integrated with the ***Waterfall Model of Software Development Life Cycle (SDLC)***. This process is designed to help the ***interface designers*** and **user experience designers** in performing ***pre-facto analysis*** and ***usability evaluation*** of ***Candidate Interface Metaphors (CIMs)*** *in the formative stages of software development*. It also helps the **software developers** in specifying the **software requirements** with full awareness of the proposed interface metaphor during the ***Requirements Engineering (RE)*** stage itself.

The ***user interface*** provides an effective ***communication medium*** between a human and a computer [Pressman, 2001]. User interfaces are designed with metaphors as they help you understand one thing in terms of the other [Lakoff et al., 1980]. A metaphor highlights similarities between the known and the unknown (user interface). In the context of this dissertation, the term ***visualization*** is to be interpreted as formation of mental visual images. This definition has acquired more meaning in terms of the recall and imagination of all ***sensory experiences*** [Owen, 1999].

Chapter 2 presents the arguments in different categories for defining the problem statement of this dissertation. ***The present understanding of the software community about application of interface metaphor is mostly based on post-facto analysis of successful software products.*** The researchers have stated that there isn't ***adequate guidance*** available for finding suitable interface metaphors [Vaantinen, 1994] [Smilowitz, 1996] and for carrying out its ***pre-facto analysis*** [Madsen, 1994]. There are several other related issues that remain to be addressed such as the ***theoretical basis for selection*** of interface metaphor [Madsen, 1994] [Smilowitz, 1996], the method for its ***optimization, evaluation*** and its ***qualitative study*** [Marcus, 1994, 1998] [Vaantinen, 1994]. Also, there is a

need to measure the ***applicability*** and ***potential*** of a metaphor [Palmquist, 1996] from the perspective of software design. The method for observing the ***trade-offs*** between the design of software and interface metaphor [Yousef, 2001] is needed so that the trade-offs could be regulated. Finally, the seamless ***fusion*** (the integration) of ***form*** (the interface metaphor) and ***function*** (the software) is extremely important [Gaver, 1995]. Poovaiah [1994] has highlighted the need of conceiving the design process of interface as a ***temporal process*** in terms of an interaction across time; as an organization of its various elements. All these issues together indicate that the process for ***visualization*** of interface metaphor needs to be defined. This chapter sets the objectives, defines the scope of work and the limitations of this research. **The dissertation presents a structured process for visualization of interface metaphor. It also shows how the steps of this process are integrated with Software Development Life Cycle (SDLC).**

The important steps of the proposed process are discussed from chapter 3 onwards. **Chapter 3 focuses on how to identify, classify, categorize, assess and then select the *candidate interface metaphors*.** The first section of this chapter brings out how the *candidate interface metaphors* can be identified based on the ***common knowledge dimensions*** of specified users. It explains the ***trilogy*** of knowledge dimensions between the user, the interface metaphor and the software. This section outlines the dimensions of knowledge in terms of ***professional, educational, day-to-day, cultural and miscellaneous***; which help in selecting the *candidate interface metaphors*. If the required knowledge dimension of a *candidate interface metaphor* is not matching with the knowledge available with users then it is rejected.

The second section of chapter 3 provides a technique for identification of *candidate interface metaphors*. It begins by explaining ***unmanifested*** and ***manifested*** states of interface metaphor. In this, ***Commentary of Task Performance (CTP)*** and ***Interface Play Script (IPS)*** are written similar to a drama script. These are juxtaposed and compared to reveal the *candidate interface metaphors*. This technique captures the ***mental model*** of users associated with the tasks taken up for computerization. It mainly captures the ***sensory details*** of the task, which are most essential while visualizing the interface metaphor. These

include visual, spatial, verbal, auditory and tactile details familiar to the user. Commentary of Task Performance (CTP) and Interface Play Script (IPS) also provide the ***justification*** for selecting a particular *candidate interface metaphor*. This technique is more useful for software projects that focus on computerization of existing processes. **It helps in seamless braiding of user's model, user requirements / user experience requirements and design model.**

The third section of chapter 3 shows how the *candidate interface metaphors* can be categorized in terms of ***animate*** and ***inanimate*** entities. The *candidate interface metaphors* are classified as ***coherent*** and ***diverse*** based on the interrelationships coded in their ***cognitive maps***.

The fourth section of chapter 3 shows how the ***potential*** of an interface metaphor can be assessed based on its conceptual structure. It helps the user interface designer in estimating its ***coverage*** as against the requirements of software. The *candidate interface metaphors* are further categorized at different ***levels*** in the ***conceptual structure***. These are termed as ***Domain, Conglomerate, Multitudinous, Singleton (Animate, Inanimate) conceptual structures*** and ***Flat concepts.***

Identification, classification, categorization and assessment constitute the sequential steps of the selection process of *candidate interface metaphors*. These are to be performed during the ***Requirements Elicitation*** stage of software.

Chapter 4 presents various qualitative aspects of interface metaphor and the determinants for tuning and crafting the desired user experiences. The ***aspects*** of interface metaphor such as ***perspective, focus, field of view, flavor, tone, affordances, fusion and multimedia representations*** can be tuned to synchronize with the respective ***determinants***. The tuning of interface metaphor helps in choosing the right approach for satisfying the ***usability objectives*** and ***user experience requirements*** of software. We have termed the ***sensation of similarities*** between the ***reference*** and **application domains** as ***conceptual resonance*.**

Chapter 5 presents a technique for ***analyzing*** the selected *candidate interface metaphors*. It provides detailed examples of ***dissecting*** the coherent, diverse and animate interface metaphors. It proves the possibility of dissecting the interface metaphor concept-by-concept; and forming the ***Cross-Domain Mappings (CDMs)*** between the reference and application domains. Dissection of coherent and diverse interface metaphors reveals their unique characteristics. In case of diverse interface metaphors - the ***conceptual proximity and alignment*** between ***core*** and ***supporting*** metaphors, ***cooperative*** and ***incoherent*** integration types are explained. Dissection of an ***animate metaphor*** shows how the ***behavior*** of software can be conceptualized on the basis of ***traits*** and ***actions***. This technique is extremely helpful in foreseeing the ***missed out*** and ***hidden software requirements.*** It also helps in identifying the skills and efforts required for the design and development of interface metaphor. This chapter introduces new terms like ***Unused Concepts (UCs)***, ***Unmapped Requirements (URs), Dissection Termination (DT) and Terminus Concept***, which are helpful in interpretation of results. **The dissection of interface metaphor provides objective basis for estimating its coverage and optimization.**

Chapter 6 shows how the ***coverage*** of *candidate interface metaphors* is ***quantified***. It also helps in ***measuring the extensibility*** of *candidate interface metaphors* for present and future versions of the software. The ***quantitative evaluation technique*** provides definite parameters for comparing the two or more *candidate interface metaphors*. It also identifies the ***weaker traits*** of the software from the perspective of animate metaphor for further improvement. The quantitative evaluation also enables the user interface designer in ***monitoring and regulating the trade-offs*** between the software and interface metaphor. These are mostly in terms of ***re-sequencing*** and ***re-structuring*** of software requirements as per the conceptual structure of the interface metaphor. The quantification technique provides several insights into overall conceptual structure of interface metaphor.

Chapter 7 presents a technique for building ***User and Application Domain Lexicons***. ***Fusion*** of ***User Domain Lexicon (UDL)*** and ***Application Domain Lexicon (ADL)*** helps in ***designing the linguistic metaphors*** for user interface.

Formats for building UDL and ADL are designed to integrate unique aspects of domain specific terms and phrases. Special attention is given to context of application and ***mental models*** associated with the user vocabulary, which helps in revealing the ***tacit knowledge*** of users. These lexicons are very helpful in not only ***reinforcing*** the main interface metaphor but also in ***capturing the vocabulary of users***. User Domain Lexicon (UDL) can be developed further beyond the scope of a software project.

Chapter 8 provides the ***Usability Heuristics of interface metaphor***. We have identified eight major heuristic criteria namely ***Familiarity, Representability, Similarity, Extensibility, Compatibility, Co-operability, Cognitive Ergonomics and Feasibility*** for ***qualitative evaluation*** of interface metaphors. Furthermore, there are ***23 sub-criteria*** and ***41 Usability Indicators*** identified for ensuring the usability of interface metaphor. The ***objective basis*** for evaluation is also indicated for each criterion. It is proposed that the ***pre-rendering evaluation*** of interface metaphor may be performed before taking it up for multimedia rendering. ***Quantitative metrics*** is also provided for quantifying the results of qualitative evaluation.

Chapter 9 shows how ***interactive multimedia*** can be used for ***design, rendering*** and ***usability testing*** of interface metaphor. ***The proposed application of multimedia integrates all the documents and artifacts generated through the visualization process.*** It allows the interface designer, software developers and users to ***refer, compare, crosscheck*** and ***corroborate*** the linked information. The multimedia rendering of interface metaphor reveals several ***hidden software requirements*** and ***usability problems*** related with it. It enables the software designer in ***foreseeing the implementation issues***. It can maintain the record of all ***evolutionary stages of visualization*** along with the ***reasons of modifications*****.** The user interface designer can test the interface metaphor over the specified users and fix the usability problems. Multimedia rendering of interface metaphor can produce several interface components in terms of graphics, layouts, animations, sounds, and the scheme of interaction design. These can be incorporated in the final software. A ***remote usability testing*** method is also developed for

testing the ***cross-cultural*** aspects pertaining to visual representations of interface metaphor. It involves the ***users from diverse geographic locations*** in the testing process.

Chapter 10 shows how the steps of visualization of interface metaphor can be ***integrated and synchronized with SDLC***. It synchronizes the steps of visualization with the steps of ***Requirements Engineering (RE)*** process; such as ***Requirements Elicitation, Analysis*** and ***Specification.*** The visualization process of interface metaphor is to be completed just before the Requirements Specification stage. It helps the software designer in designing the software with adequate understanding of the expected user interface or user experience.

After this the conclusions, the major contributions of this dissertation and future research extensions are documented.

Figure 1.1 on next page presents the structural diagram of the dissertation.

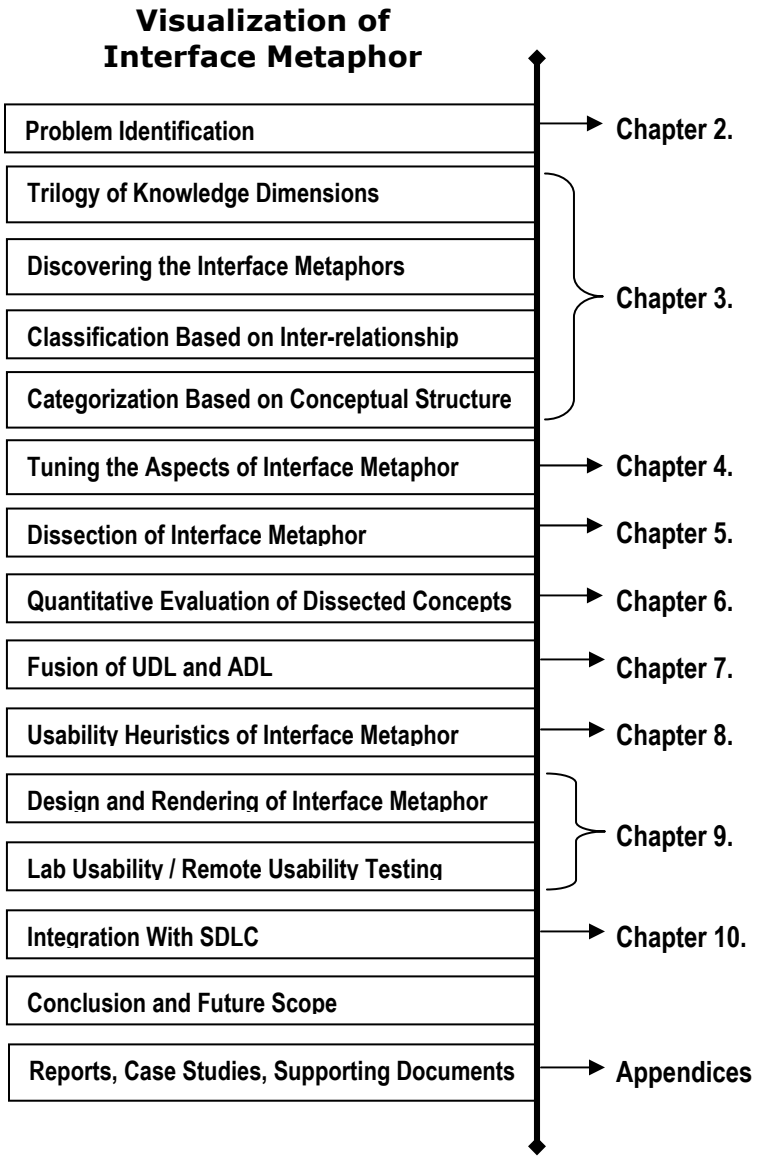

Figure 1.1 Structure of dissertation

Chapter 2. Problem Identification

2.1 Literature Survey

The topic of this dissertation is highly ***interdisciplinary*** and it touches upon diverse domains such as ***Software Engineering, Cognitive Science, Semiotics, Linguistics, Visual Communication and Multimedia Visualization.*** This topic of research belongs to the new emerging discipline called ***Human-Computer Interaction (HCI)***. It is based on the study of research publications mostly from diverse technical journals and magazines like ACM Interactions and Communications of the ACM, Human Factors in Computing, Man Machine Studies, IEEE Software and SIGGCHI Bulletin. These have regularly covered interdisciplinary articles and papers related to the topic of dissertation. Large number of research papers from the proceedings of international conferences such as Computer Human Interaction (CHI) and HCI International are studied. Most of the journals and conference proceedings were accessed from ACM and IEEE Digital Libraries. The study also includes research papers from a variety of other sources such as Journal of Educational Multimedia and Hypermedia, Journal of Computer Documentation, Journal of Human Computer Studies, etc. There are several other sources like ISO standard for usability and electronic proceedings of a variety of conferences. These are appropriately cited during the deliberations and enlisted in the list of references.

The research papers, articles and books authored / edited by the most notable researchers in the domain of HCI are studied. These include HCI experts like Aaron Marcus, Brenda Laurel, Jakob Nielsen, Thomas Erickson, John Carroll and Keran Holtzblatt. We have also studied the works of George Lakoff (Cognitive Linguist) and Donald Norman (Cognitive Psychologist and Usability Expert). These are appropriately cited while positioning the problem statement of the dissertation.

2.2 State-of-the-art

The research on interface metaphor during 1981–1990 broadly covered its emergence, benefits, understanding of how it works and its applications. Whereas, the later part of research during 1991–2005 raises the problems, issues and gaps in the process of development of interface metaphor. This is a very broad categorization based on prominent trends of the research publications on interface metaphor.

The first Graphical User Interface (GUI) that used menus, buttons, icons and mouse pointer emerged in 1981 from Xerox Palo Alto Research Center (PARC). It was used as part of the 8010 Star Information System designed for business professionals who handled information. Star's GUI represented the typical office environment in the form of Desktop metaphor. The success of Desktop metaphor captured the interest of user interface researchers [Johnson et al., 1989]. This indeed was the beginning of cautious use of metaphor as an integral part of user interface. It proved that interface metaphor could capture the mental model[1] of users [Erickson, 1990]. It inspired further research initiatives on interface metaphor in terms of using the cognitive psychology [Johnson et al., 2000] for improving the effectiveness of user interface.

But the Xerox PARC was not the first to use metaphors. Metaphors have been expressed through language since ages. Metaphors are pervasive in everyday life of humans. We commonly use them for understanding one thing in terms of the other [Lakoff et al., 1980]. If you look at the history, the writings in ancient religious scriptures and iconography are quite metaphorical in nature. *Nyāyasiddhānt Muktāvali* of *Vishwanāthanyāyapanchānan* [17th Century A.D.] provides the definition of metaphor in *Sanskrit* (an ancient Indian language) as below.

तद्भिन्नत्वे सति तद्गतभूयो धर्मवत्त्वम् सादृश्यम् ।

It can be translated as- being different from each other X and Y share many same properties. Humans have always used metaphors as the tool of innovation, creation

[1] The gut feeling based on experience used to understand the functionality or behavior of system (Dix et al., 2004).

and communication [Richard, 1941][Gasset, 1925] [Mountford, 2000]. However, the success of Desktop metaphor resulted in several articles and research publications that attempted to define the role, the cognitive model and advantages of using the interface metaphor.

The present understanding of interface metaphor broadly covers following points-

- **It serves as a 'cognitive mediator' [Laurel, 1993] or a 'cognitive ploy' [Palmquist, 1996] for ensuring effective user-system communication [Kass, 1988].**
- **It provides a conceptual model to users for predicting the functionality of a system [Norman, 1988].**
- **It can incorporate the features of real world into the computer application [Nielsen, 1994].**
- **It can help in determining the presentation and behavior of software [Brad, 1990].**
- **It also enables the users to start learning from what is already known and familiar to them [Erickson, 1990].**
- **It superimposes essential similarity (between the software and real world) through visuals (words and images) or through acoustic or tactile means. It also represents the mental models through navigation, tasks and roles [Marcus, 1998].**

With the faster insertion of information technology across the world, usability of user interface became the focal area of research. The trends of research in Human Computer Interaction (HCI) accelerated in the year 1990 [Carroll, 2002]. User interface became a basis for the product's usability and commercial success [Marcus, 1995]. Naturally, the use of interface metaphor is now considered most inevitable [Chen. 2002] aspect of user experience design. **In summary, the interface metaphor continues to be an important area of research in the 21st century [Hamilton, 2000][Marcus, 2002].**

The next section of this chapter identifies the requirements, gaps and problems pertaining to application of interface metaphor in software products.