Investigating a Design Pattern to Support Personalized Human Computer Interaction

YI JI

University of Technology, Sydney

Submitted to the Faculty of Engineering and Information Technology in Fulfillment of the Requirements for the Degree of Doctor of Philosophy

2015

Certificate of Authorship / Originality

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree. I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledge- edged. In addition, I certify that all information sources and literature used are indicated in the thesis.

YI JI

January 2015

Contents

Acknowledgement

Foremost, I would like to express my sincere gratitude to my advisors Prof. Ernest Edmonds and Dr Andrew Johnston for the continuous support of my Ph.D study and research, for their patience, motivation, enthusiasm, and immense knowledge. Their guidance helped me in all the time of research and writing of this thesis. I could not achieve my goal without their supports; it is my pleasure to having them during my Ph.D study.

Besides my advisor, I would like to thank Rodney Berry for his encouragement, insightful comments, and hard questions.

My sincere thanks also goes to Prof Ying-Qing Xu for offering me the opportunities in their groups and leading me working on diverse exciting projects. In particular, I am grateful to Prof. Tom Hewett for enlightening me the first glance of research.

I thank my fellow researchers in the Creativity and Cognition Studios for the stimulating discussions, for collaboration in different projects, and for all the fun we have had in the last four years.

Many thanks also go to all the practitioners took place in the interviewing process for lending their invaluable insight to me and enriching my viewpoint on the thesis. Especially for the participants came from Tsinghua University, thank you for contributing helps and inspirations to my work.

Last but not the least, I would like to thank my family: my parents Weihong Ji and Guangtai Hou, for giving birth to me at the first place and supporting me spiritually throughout my life, and to my dear wife Yi Shi for giving me endless supports and love.

List of Figures

List of Tables

Abstract

For many interaction designers, human-computer interaction is a kind of human communication, in which the computer acts as an agent of the designer. When successful, users can communicate with the computer in an effective way. To this end, interaction design is really about how to support the way people communicate and interact in their everyday and working lives. A key challenge for human computer interaction design is to support a natural and intuitive interaction between the user and computer.

In this thesis, we propose an interaction design method: Interaction Language Design Pattern (ILDP) which focuses on allowing users to personalize their interaction. Different to existing interaction design methods, which aim to provide a designed interaction model, the ILDP centers on forming an effective way of customizing interactions - producing an interaction language to do so.

In order to create this interaction language to support effective interactions between human and computer, the key components and basic structure of language are discussed. The application of the interaction language is described by giving a comprehensive design guide. The quality of the human computer interactions is evaluated using two prototype studies. The first, paper prototype, study focuses on usability testing. The second, Hi-Fi prototype is created for a user experience study. According to the result of evaluation, personalizing interactions can help users improve their experiences. The findings show that this method is effective at supporting the personalization of interaction.

Chapter one: Introduction

1.1 Introduction to the Research

Human – centered interaction started as the study of an individual interacting with a computer (Petra Sundström , 2005). One main task of interaction design is to bring two worlds together, the world of technology and the world of people and human purpose (Kaptelinin and Bannon, 2012). To this end, Rogers et al point out that the focus of interaction design is on how to design interactive products that enhance and extend the way people communicate, interact and work (Rogers et al., 2011). Similarly, the Interaction Design Association gives detailed objectives for interaction design, claiming that the essential purpose of Interaction design is to help the user effectively communicate with a system. They argue that this is achieved by defining a system's responses to user interactions so as to create a meaningful dialog which is grounded in an understanding of real users (goals, tasks, experiences, needs, and expectation) and balances these needs with business goals and technological capabilities (IxDA, 2014).

However, the role of Human Computer Interaction (HCI) design as an activity within the design process can be problematic as the definition of interaction is different for the designer and the end user. A significant consequence is that the operation of human computer interface is becoming more and more complicated (Huang, 2009). Because that the interactions between the user and computer which are composed by specific interfaces and interaction forms are typically created by interaction designer rather than the real user. To produce a user-centered interaction the user should be allowed to fully control his/her interaction. In other words, to provide the best suitable interaction within given constraints, it requires to minimize the barrier between the human's cognitive model of what the users want to accomplish and the computer understanding of the user's task (Rogers et al., 2011) Otherwise, human computer interaction is considered as incomplete and unbalanced from users' perspective (Andreev, 2001).

We believe it is possible to improve the quality of interaction if the users are able to personalize the interaction, which is different from the existing interaction that is produced predominantly by interaction designers. Designers offer standard solutions to solve the users' individual problem. However, the users' interaction is more process-dependent than is commonly acknowledged: the way a problem is solved can fundamentally change the user's evaluation of the problem (Kohlhase, 2008).

Beaudouin-Lafon argues that users not only passively adapt to new technology but also adapt and appropriate it for their own needs (Beaudouin-Lafon, 2004). The challenge is not only to make information available to people at any time, at any place, and in any form, but more importantly to provide right thing at the right time in the right way (Fischer, 2001). A new design philosophy of human computer interaction design suggests a new perspective of designing HCI, that centred on the quality of interaction (Ryu and Monk, 2009). However, designing such interaction between human and computer faces a several challenges.

Firstly, interaction designers are increasingly finding themselves going beyond the design of usable interactive systems and are instead expected to tell a story of how to interact with the computer with great experiences (Lim et al., 2008). As McCarthy and Wright indicate, a user's experience is generated from the involved totality of people acting, sensing, thinking, feeling and meaning-making including their perception and sensation of the artifact in context (McCarthy and Wright, 2004). This means that user experience design needs to be concerned with all the qualities of the interactive experience that make it memorable, satisfying, enjoyable and rewarding for the user. This implies that it is very difficult to design user interaction experience without the user's active participation.

The second challenge is how to adopt a user's dynamic interaction experience to alter the interaction. McCarthy and Wright argue that the experience has to be understood as a whole and cannot be broken down into its constituent parts, because experience lies in the relations between the parts (Wright et al., 2008). In terms of constructing meaningful human computer interaction, interactivity, as we have seen, involves the combination of people, technologies, activities and the contexts in which the interaction happens. This context includes the wider social and cultural context as well as the immediate context of use.

The third challenge is how to create an interaction between a diverse range of users and the computer to match different purposes such as working, thinking, communicating, learning, critiquing, explaining, arguing, debating, observing, deciding, calculating, simulating, and designing. To establish such a comprehensive interaction between human and computer requires a powerful interaction tool, which does not constrain the users' interaction into a fixed interaction model.

Therefore, in order to overcome the above challenges we contend that a solution to establishing a mutual conversation between user and computer should be centered on the use of language in human computer communications. The fundamental idea is similar to some design approaches based on applying linguistic structure in designing interactive artefacts i.e. a linguistic design approach to interface design (Tidwell, 1999, Andreev, 2001) and human computer interaction design (Bueno and Barbosa, 2007, Erickson, 1998, Clay and Wilhelms, 1996, Branigan et al., 2010). Erikson suggests the notion of viewing Pattern Languages (PL) as a common language for HCI design claiming they are "*…a lingua fracas —a common language—for a design project. The idea is that a lingua franca is accessible to all stakeholders, particularly those who are traditionally marginalized in the design process: the user.*" (Erickson, 2000aP. 357) Additionally, the notion of creating a common language for human computer interaction has been repeated and advocated by many HCI researchers and practitioners (Borchers, 2000) (Denef and Keyson, 2012).

We argue that by developing a linguistic conversation design perspective we can enhance our coordination of attention as well as action by supporting a level of conversation that not only conveys the function and usability of a product, but also responds to the user's evolving reflection and experience. Following the above perspectives, in this thesis, we seek to establish a common language – interaction language, which supports effective human computer conversation.

As Alexander points out, language actually generates design (Alexander et al., 1977). In HCI design practice, many languages generate the design – human computer interaction. Some important languages have been used in interaction design for producing a variety of conversations between human and computer regarding to the users' individual perspectives. From the technical point of view, a variety of programming languages like C, C++, Java and pattern languages like interaction pattern language and User Modelling Language (UML) are used to produce multiple interactions. From a less technical perspective, natural language (e.g. English, French, etc.) and Interface Design Pattern Languages (IDPL) (Tidwell, 1999) have been used to support human computer interaction.

However, to date no common, widely applied interaction language has emerged in human and computer interaction (Pan and Stolterman, 2013). To this end, there is no user-oriented interaction language which helps users form a personalized, expressive interaction based on the user's needs and desires. One major challenge for building a common language is how to make the language of HCI able to support mutual communication between different users and the computer. As Erickson suggests, a common language is needed to bind pattern solutions together, to help the users evaluate the solution as a coherent whole (Erickson, 2000a). This language should represent interaction sequences from the users' point of view, a language that can be used to express a user's beliefs and expectations. In summary, the common language has to be recognized and utilized by both designers and the end users.

In this thesis, we will propose a technique for creating a human computer interaction language using a linguistic structure, to enable personalized interactions. The significant feature of the personalized interaction is represented in three aspects:

- The Interaction reflects and responds to the users' thinking, problem and experience dynamically.
- The interaction allows users to define a special meaning for the interaction by mapping, complementing and integrating existing elements of interaction..
- The interaction can improve users' interaction experience by customizing the usability and functionality of a computer.

1.2 Motivation for the Research

Harrison et al. identified three waves of paradigms within HCI, the first being "Human Factors/ Engineering", the second "Cognitive Revolution", and the third "Situated Perspectives". Change towards the third paradigm is evident, firstly, in the increased awareness of the dynamic character of user contexts; second, in the sociality and situation of interaction; thirdly, in issues related to learning environments; fourth, in non-task-oriented computing (such as ambient interface and experience-centered design); and fifth, in the roles of emotions in human–computer interaction (Lim et al., 2008). The fundamental goal for designing such inclusive interaction is to create interactions that elicit the desired perceptions of artifact qualities, affects and emotion in a natural way.

This has led me to explore a design method to easily and actively tailor a system to the user's personal needs. Without having comprehensive understanding of a particular interactive system, the user will find difficult to build up an effective communication with computer. The level of user interaction experience is largely determined by how the user and the technology cooperate within a particular context. Although a wide range of design methods like human-centered design, experience design, emotional design and aesthetic design that have been developed to increase designer's comprehension of user's capability and needs (Stanton et al., 2004), many of those methods have had limited approval in design practice and there is evidence of a poor fit between the structure of many methods and the ways in which users think and work (Waller et al., 2009). At a result, there are 'gaps' which emerge between the user and product: one is the 'execution gap; the other is the 'evaluation gap' (Norman, 2002, Norman, 1988). In short, problems of communication keep occurring during user and computer interaction and lead to miscommunication and undesirable user experiences.

1.3 Research aims

The research aims to investigate how to effectively personalize interaction and identify an interaction design pattern, which use an interaction language to establish balanced, bidirectional and natural communication between user and computer. This will provide a user with more a more flexible, appropriate experience when they interact with computers in a different contexts.

1.4 Objectives

The objects of this research are to:

- Identify strategies and produce a understandable design pattern centered on personalizing interaction from a user's individual standpoint; and
- Evaluate the application of the design pattern in user studies.

1.5 Research Questions

The research question is: how can we create personalized human computer interaction that fosters comprehensive communication between users and the computer. There are two sub-questions for this research:

- How should the users personalize their interaction?
- How does it support users to personalize their interactions?

1.6 Expected outcomes

In order to solve the problem and optimize the user's interaction with a particular system, we recognize that we need to establish user-oriented communication between the user and the computer. The underlying consideration for designing such "mutual perspective" interaction is to balance the conversation between the user and computer.

A new interaction design method: the Interaction Language Design Pattern (ILDP) is proposed. The Interaction Language Design Pattern (ILDP) aims to facilitate personalized human computer interaction by helping designers to construct a Domain Specific Interaction Language (DSIL) to create human computer interactions, which can adapt to specific user purposes. The ILDP is intended to help users to justify and organize their interaction using their language.

There are three research tasks for this research:

- Create a design pattern to support the generation of interactive artifact, which provides personalized interactions.
- Create an interactive product based on the design pattern in order to show how the interaction designer uses the Interaction Linguistic Design Patten (ILDP) to enable personalized interaction
- Evaluate product in order to identify whether the user experiences are positive and identify strengths and weaknesses of the ILDP.

1.7 Significance

With the increasing sophistication of interactive systems, more and more interaction design challenges and problems are being raised. The goal of research is to create new interactive frameworks that integrate human cognition and behavior into applications so they can be a pleasure to use. This research provides an additional perspective on interaction design from a different point of view, focusing on how the product can be created based on users' personal needs.

One particular purpose for this research is going to investigate how to integrate the user's personal experience and behaviour to lead interaction design rather than just assuming or modeling.

In this research, by looking beyond common notions of interactions (formalized action), we can increase the possibilities to open a personalized interaction design system based on a novel co-development, co-creation, and co-ownership interaction composition.

1.8 Contributions:

- Exploring an interaction language combined with other typical interactions to establish personalized interaction between the users and computer.
- Providing an Interaction Language Design Pattern (ILDP) to help designers to produce the Domain Specific Interaction Language (DSIL) and construct a text-based interaction that allows the personalization of human computer interaction.

1.9 Thesis structure

Chapter two: Literature Review

Chapter two reviews relevant research in the area of human computer interaction design. The first section aims to explore basic principles of human communication by investigating interpersonal interaction and two-way communication. In the second section, different interaction models of HCI as well as conversational interaction languages and resulting interaction experience are analyzed. The literature review highlights problematic areas of current interaction design methods, tools, languages and shows that there is little research on design methods for personalizing interaction.

Chapter three: Research Methodology

Chapter three reviews the methodological approaches which have been used to evaluate human computer interaction design works. The two aspects of usability testing, product usability testing and user experience studies are described. Usability testing focuses on evaluating the efficiency of human computer interaction carried out by a new interaction, which is the resulted of investigating a variety of theories, forms and design frameworks. User experience studies aim to gain new knowledge and deep understanding of the interaction between human and computer at different levels including visceral, behavior and affective.

Chapter Four: Interaction Language Design Pattern (ILDP)

Chapter four presents the Linguistic Interaction Design Pattern (ILDP). In the first section, following the principles of human communication we propose a theoretical framework for building an interaction language to establish effective communication between the user and computer. The rest of chapter focuses on two areas. First, the structure of interaction language is described. Second, the interaction language itself is developed in three steps: interaction language production, interaction language perception and interaction language pragmatic.

Chapter five: Usability Study

This chapter presents the findings from evaluations of prototype of a drawing systems built based on the Interaction Language Design Pattern (ILDP). In the first section, a paper prototype for a drawing system is created based on the theoretical framework of ILDP. In the second section, user feedback of using the paper prototype is presented and analyzed.

Chapter Six: Users experience study

Chapter six describes a user study which compares users' experiences using two different drawing systems: Photoshop CS and the Hi-Fi prototype based on the ILDP. This user study focuses on exploring different user interaction experiences. The results show that interaction language can assist the user to make an appropriate and effective interaction with the computer.

Chapter Seven: Conclusion

This chapter summarizes the results of the research and contribution of the research work. Following that, we present our future work in both short term and long term and discuss challenges for applying the Interaction Linguistic Design Pattern in different areas.

Chapter Two: Literature review

Through this literature review, we aim to answer the following questions:

- What is the main purpose of interaction design? What kind of role does the user play in human computer interaction?
- What are the problems for current human computer interaction and interaction design?
- What is an appropriate technique that is suitable for constructing effective interaction?
- What are the features of such an interaction?

In this chapter, we discuss the state of the art in human computer interaction. We explore ways to carry out personalised human computer interaction. We investigate and analyse existing human computer interaction models with regard to design methods, interaction languages and the resulting user experience.

The literature review is organized as follows. Initially, we investigate the definition of effective human interaction (2.1), exploring the basic principles and essential features of human natural communication (2.2) and the principles construct an mutual conversation (2.3). Second, we analyse a variety of current interaction design methods that produce human computer conversation from different aspects including: design methods (2.4); language of interaction; interaction model and resulting user's experience (2.5). Third, we investigate and elaborate the problems of human computer communication (2.6). A new design method to build mutual interaction is described in Chapter four and evaluation of the effective interactions is provided in Chapter five and six.

2.1 Effective human interaction

Interaction design is becoming more and more important. For years, many researchers have argued that the design of Human Computer Integration (HCI) is, at its heart, a communicative process. For example, usable and accessible products need to match a user's understanding of products and requirements (Langdon et al., 2012). A major view of human computer interaction, accepted by many designers and researchers in the HCI is that of designing interaction as a way of controlling and carrying out the conversation between human and computer (Dubberly et al., 2009). In this view, designers create different interactive artefacts by combining a variety of objects to convey different contents and meanings (Polovina and Pearson, 2006). In other words, the interactive artefact is a kind of designer-to-users message, representing the designer's solution to what he/she believes are the users' problems, needs and preferences. In this message, the designers tell the users directly or indirectly, what he/she had in mind when he/she conceived the application (Bueno and Barbosa, 2007).

Alternatively, adopting the user's perspectives and needs to carry out communication is critical in order to establish a mutual understanding and an effective conversation. For human natural face-to-face communication, conversation is considered as a collaborative activity that helps the conversation participants establish a mutual understanding and effective conversation (Tubbs, 2010). Some widely used design approaches focus on designing human computer interaction according to the user's perspectives and needs, including Computer-Supported Cooperative Work (CSCW), User-Centred Design (UCD) and User Experience Design (UED) etc. These approaches focus on helping the computer better support human work (Eseryel et al., 2002). And the others focuses on creating a desirable user experience, such as emotional experience, User Experience Design (UXD) based on the user's characteristics (Wright et al., 2008). However, a fact is that in established human computer interaction systems, users have less power compared to designers regarding changes to the program running on the computer. Moreover, "… currently designers do not use any tools that would support them in matching their intended design of products with the users' understanding of using the products" (Langdon et al., 2012). In other words, human computer interaction is unbalanced and incomplete (Andreev, 2001).

Therefore, we believe that effective human computer interaction should build on a user-centred communication that composed of presentation, discussion, disagreement, negotiation, and collaboration among a diverse array of communication participants. These participants include the designer, the user, the software engineer and other stakeholders of developing a particular interactive artefact. In order to have a deep understanding of human actions within natural communication, it is necessary to explore how human communication is defined and constructed.

2.2 Interpersonal communication and its basic principles of construction

Human communication has been broadly defined as “the sharing of experience” and interpersonal communication is the basic unit of human communication (Tubbs, 2010). In order to systematically understand how to establish effective interaction between diverse communicating participants including human and interactive artefacts, it is necessary to investigate the essential features of human interpersonal communication.

For this purpose, we use Activity-based Communication Analysis (ACA) to explore how to make effective interactions between humans and a variety of artefacts such as the computer. Activity-based Communication Analysis (ACA) is proposed and developed by Allwood (Allwood, 1976, Allwood et al., 1992, Allwood, 1977, Allwood, 2007). We chose this framework because ACA has been widely used to investigate the key features of establishing human communication (Jokinen, 2009).

From the point view of ACA, interpersonal communication

- … is a process of collaborative interaction between conversational participants. Conversational participants are engaged in an Ideal Cooperation (IC), if the following features are observed: joint purpose, cognitive consideration, ethical consideration and trust.

- … supports full-blown (complete) interaction, composed of contact, perception, understanding, and reaction.
- … happens at different levels: physical, biological, psychological and social organizational level.
- … is built on communicational participants' rational action, in which participants get rewarded for achievement when their goals are considered "rational" for their purpose and stored for future use. Rationality is not a property of a particular act, but related to the participant's goals and circumstances. It is encoded in the reasoning processes used to achieve the goal: it is a procedural concept that does not exist without the activity that drives interaction and communication in the first place. (Allwood, 1976).

In summary, interpersonal communication is based on the communicational participants' need to be understood, and carry out reasonable results that are developed by the communicating participants.

Alternatively, human communication is multimodal, which combines verbal and nonverbal interactions. Stivers and Sidnell indicated that face-to-face interaction is, by definition, multimodal interaction in which participants encounter a steady stream of meaningful facial expressions, gestures, body postures, head movements, words, grammatical constructions, and prosodic contours (Stivers and Sidnell, 2005). In addition, Kendon explains the multimodal human communication is established from the following process:

"*The transition into referential or language-like expressions involved hands and body, face and voice and mouth, all together, as an integrated ensemble. Modern humans, when they speak together in face-to-face situations… always mobilise face and hands and voice together in complex orchestration… Every single utterance using speech employs, in a completely integrated fashion, patterns of voicing and intonation. Pausing and rhythmicities, which are manifested not only audibly, but kinetically as well…* "(Kendon, 2009p. 363).

Therefore, based on the above perspective effective human interaction should be generated from a comprehensive communication that integrates verbal and nonverbal interactions. There are two essential factors: one is shared language that can support human to make effective communication; the other one is to build common ground on different levels to establish mutual communication.

In next section, firstly we investigate the basic functions of language regarding how the people communicate and determined their behaviour by using language (Winograd, 1986, Krippendorff, 2005). Secondly, we explore how the people establish variety of common grounds for mutual communication by integrating verbal and nonverbal interaction.

2.2.1 language in human conversation

Language is a primary means of communication because it organizes information to share with other people and directs their behaviour (Winograd, 1986). Allwood (1976) defines human linguistic communication as a rational and cooperative activity, because language can help the communicational participants generate an effective communication and cooperation among human beings. Participating in the communicative activity, participants have certain roles, which further determine their communicative activity. In other words, the conversational participant's behaviour is intentional and purposeful and intentionally controlled.

In this thesis, we are not discussing the whole history of language generation and usage. However, we will explore how language supports human communication by establishing a shared language. In the following section, we investigate the basic features and functions of natural language that are required in order to construct interpersonal collaboration.

Although people are aware of using a language, they rarely know much of what is going on while they are speaking, listening, or reading. According to Krippendorff, language usage can be conceptualized in four aspects (Krippendorff, 2005):

- a system of signs and symbols
- a medium of individual expression
- a medium of interpretation
- a process of coordinating the language user's perceptions and actions

We can see that in human natural communication, language plays diverse roles with various functionalities to support different communicative activities based on the conversational participant's needs.

First, language is a system of signs and symbols, and a medium of representing an object referred to in the physical world. Halliday (Halliday) claims that language is used as a tool for representing knowledge or for making meaning based on a Systemic-Functional Linguistics (SFL) system (Halliday, 2002). From this perspective, language is a medium mainly used to record particular information and to deliver intent and meaning between different peoples.

Second, language is a medium of individual expression. Language is used for the expression of content that combines people's individual experience of the outside world and the inner world of their own consciousness. Halliday points out that language can provide an "ideational function" that gives structure to experience and helps people to determine their way of looking at things (Halliday). As a result, people can obtain and utilize natural language to express their thinking and experience to communicate with other people. In natural interpersonal communication, people have ability to generate any meaning based on their intention by combining diverse basic components of language such as vocabulary and syntax.

Thirdly, language is a way for humans to coordinate their perceptions and actions in which the communicative participants live with each other: co-constructing their conceptions of each other and of each other's artefacts. Pappas points out that language *"is not a tool but a true medium, a mediator of* *understanding and the very basis of any common human understanding, from artistic communities to philosophical movements all the way to the strictest scientific and academic circles. Accepting this premise, it is the author's hope that fathoming the vital role language and linguistics will play in the forming of this new "language" will come naturally to the reader".* (Pappas, 2011p. 138).

Alternatively, Dearden indicates that *"from a pragmatic perspective, in particular notions of 'utterance' and 'speech genre', he developed an account of designing in relation to 'material utterance'. In addition, a key concept of genre is provoked that in considering how the experience and history of speaker and audience impact on the form of utterances."* (Dearden, 2006p.340)

In term of human computer interaction, language as a medium of interpretation concerns people's use of language to generate meanings through rearticulation of content and relies on a community to determine the legitimacy of the interpretation offered by communicational partners.

2.2.2 Common ground of interaction

More important is that language is used to construct a common ground for human communication and to generate new common ground for further conversation (Clark, 1996). Based on the study of natural communication, we realize that users' interaction experience and reaction will be continually changed during interaction. Furthermore, the users' experience and responses are led by the users' individual perspective and understanding of their interaction. This in turn is built upon the users' interactive semantic image (Zhuge, 2010).

Zhuge writes that a user's interactive semantics[1] are a semantic network, linking objects or points in a specific classification space shared between communication participants (see Figure 2. 1) (Zhuge, 2010). As a result, the user's interaction concepts change dynamically based on the user's dynamic interactive semantic images (Zhuge, 2010). He also describes the human interaction semantic image as a network of object or points in a

[1] Interactive semantic: the meaning a user discerns from the look, behaviour and

classification space shared between individuals. It forms and evolves through constant interaction. He further points out that the semantic image evolves and results in the formation of a variety of interaction semantics.

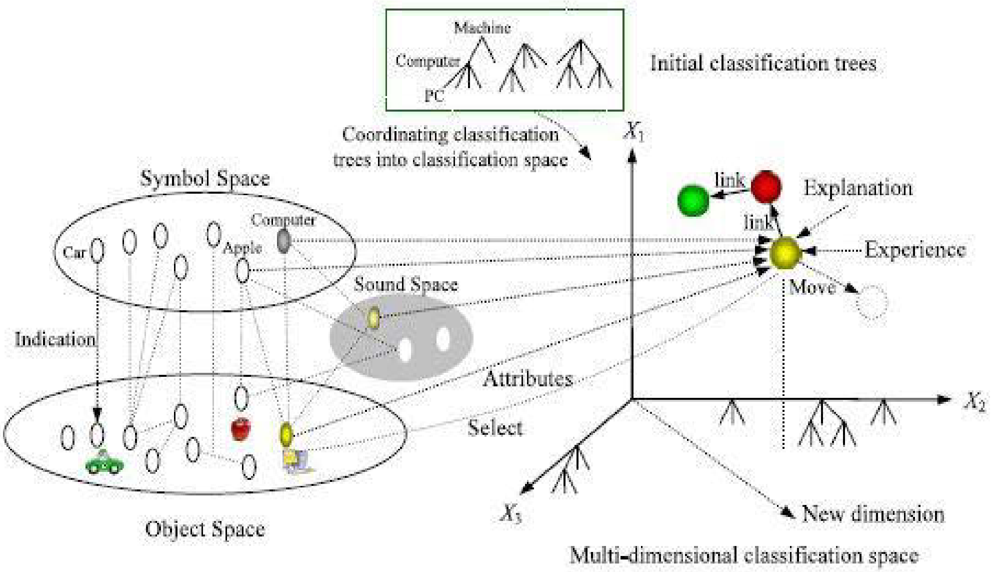


Figure 2. 1 Flowchart of interactive semantic images (Zhuge, 2010)

According to Zhuge's definition, objects and concepts are placed in evolving classification spaces that generate classes and subclasses. An object's semantic image reflects the attributes and class of the object from multiple facets like shape, colour, texture, position, and sound. An individual's semantic image is a network of objects or points in individual classification space. 'Rules' are the direct or indirect relations between two nodes (Zhuge, 2010).

In current human computer interaction design practice, the interaction concept is formed by integrating various interactive semantics and is delivered to the end users on two levels: the physical level which refers to the properties of the interactive artefact with regard to its usability; the cognitive level which centres on the user's experience of interacting with the product. Currently, in the interaction design community, different semantic syntaxes are employed to construct these interactive semantics and deliver them to the end user on different levels in order to accomplish the above interaction design concepts. Designers use programming language to realise object-focused interaction concepts regarding the user's physical interaction. At the same time, they employ pattern language to compose experience-focused interactive concepts based on the user's cognitive experiences. In other words, these interaction syntaxes formulate interactions between user and interactive artefact by defining how the interactive artefact and the user are operationalised, i.e., how they can be met and through what means. In detail, the designer is concerned with designing interaction, somewhat like sketching a story of an interaction, and describing the procedure of how the user is going to accomplish different tasks.

We claim that it is essential for the user to have similar power to have a communicational language that can help the user to specify interaction and to address his/her individual interactive semantic. Similarly, Forlizzi and Battarbee indicated that a meaningful interaction design should allow the user to express an individual concept and relationship to a product, or some aspect of it through an expressive interaction (Forlizzi and Battarbee, 2004). Thus, we aim to construct personalized human computer interaction by using a DSIL to allow for the user's evolving value judgments while interacting. That means that the user's interactive semantic models should co-exist, support each other, and play different roles in human computer interaction (Zhuge, 2010). From this perspective, the personalized interaction needs to implement the end user's concept and dynamic interactive semantic to carry out the interaction rather than to constrain the user through a fixed interaction model.

The next step is to study how the existing interaction design to adopt the user's interaction concepts and reflections to construct a real interactive artefact.

2.3 Human Computer interaction

Interaction design is all about *"how to optimize the user's interactions with a system, environment, or product, supporting and extending the user's activities in an effective, useful and usable way."* (Rogers et al., 2011p. 2) Currently, the focus of Human

Computer Interaction (HCI) has shifted from mainly being concerned with usability and functionality of product to address qualities of users' experiences (Giaccardi et al., 2013). Harrison et al. identified three waves of paradigms within HCI, the first being “Human Factors/ Engineering”, the second “Cognitive Revolution”, and the third “Situated Perspectives” (Sengers et al., 2009). Change towards the third paradigm is evident, firstly, in the increased awareness of the dynamic character of user contexts; second, in the sociality and situation of interaction; thirdly, in issues related to learning environments; fourth, in non-task-oriented computing (such as ambient interfaces and experience-centred design); and fifth, in the roles of emotions in human–computer interaction (Harrison et al., 2011). So, to create human computer interaction, we have to go beyond narrow usability issues and consider enhancing and augmenting people's work, communication and pleasure (Wright et al., 2008).

Preece & Rogers argue that interaction design contains several important steps. These are: taking the user from a goal and intention; formulating a task and sub tasks; carrying out these actions whilst receiving feedback on the physical level, and making an emotional reflection (Preece and Rogers, 2007). In the design process, designers have to recognize every factor that affects the user experience of interaction with the product and environment. General speaking, these factors happens at three different levels: visceral, behaviour and reflective (Norman, 1988). Additionally, he claims that it is possible to create an emotional product if the product can satisfying the user's needs on the above three levels (Norman and Draper, 1986, Norman, 1988).

To design effective human computer interaction, the main purpose is to build a unique human computer interaction by integrating various key factors including the users' characteristics, properties of hardware and the interaction context in an appropriate form to help users achieve their needs and goals. The key challenge is: how do we create the interactions that can elicit user's desired perceptions of artefact qualities, affects and emotion in a natural way?

In the following section, we investigate existing interaction design methods, interaction language and user experience to analysis the quality of human computer interaction and identify the problem of human computer interaction.

2.3.1 Design method of building human computer interaction

There are many different design methods that are used in interaction design. Essentially the approaches can be grouped into three categories: System-centred, Human-centred (Saffer, 2009) and interaction perspective model (Ryu and Monk, 2009). These design approaches focus on different aspects of human computer interaction and result in different models of interactions.

System-centred interaction design method

The first interaction design approach is system-centred, which is intended to capture the critical qualities of a system in terms of data abstractions or information modelling. It stresses usability of product, technical functionality and efficiency of interaction (Saffer, 2009). The represented interaction is instrumental interaction, which is constructed by using the system-centred design method leads to direct manipulation (Beaudouin-Lafon, 2004). The system-perspective system can provide an abstract representation to help the designer to code how the system reacts in particular situations, ensuring the appropriateness of system behaviour from moment to moment. The idea is to guarantee the logical adequacy of users' task sequences and to model those sequences in a meaningful way (Ryu and Monk, 2009).

Building a particular interactive system following the system-centred design method, designers rely on ergonomics relating to human anatomy, physiology and psychology. Ergonomics is one important component for the system-centred interaction model. As Jokinen indicates, the goal of ergonomics is to apply scientific information to the design of objects, systems and environments so that the user capabilities and limitations are taken into account (Jokinen, 2009). In many cases, ergonomics is applied to interaction and software design to help the end user easily understand

and manipulate the functionality of the system. Some important system-centred interaction design frameworks are described in Ryu and Monk (Ryu and Monk, 2009).

Although using ergonomics can help designers to understand and model users' task sequences in a logical way, while such models do not adapt to what the user will actually do as they use the system. In other words, the need to change the behaviour and form of a provided artefact is not usually considered. (Forlizzi and Battarbee, 2004). We can understand that instrumental interaction relies on the designer's observation and personal understanding of the user's activities and tasks to then design solutions to help the user accomplish the task by operating a particular tool, but only in circumstances that the designer can foresee.

Typically, the instrumental interaction model is represented as a type of direct manipulation. It introduces the notion of instruments as mediators between users and a target object. It is inspired by human daily experience of using tools, instruments and devices to operate on the physical world rather than using our body. Based on the principle of direct manipulation (Shneiderman, 1993), digital objects are embedded in a static interface so that they can be interacted with in ways that are analogous to how physical objects in the physical world are manipulated. In so doing, direct manipulation interfaces such as Windows Icon Mouse Pointer (WIMP) and Graphic User Interface (Voelter et al.) are assumed to enable users to feel that they are directly operating on the digital objects represented by the computer. The three core principles are (Rogers et al., 2011, Suchman, 1987):

- Continuous representation of the object and actions of interest
- Rapid reversible incremental actions with immediate feedback about the object of interest;
- Physical actions and button pressing instead of issuing commands with complex syntax.

According to these principles, an object on the screen remains visible while a user performs physical actions on it and any actions performed on it are immediately visible. For example, consider a simple navigation instrument: the scrollbar. This instrument is composed of the physical mouse and the on-screen virtual scrollbar. The user interacts with it thorough direct action, and receives direct feedback from the mouse and the virtual scrollbar on-screen. The scrollbar then translates a user's actions into scrolling commands. This interaction provides two responses: towards the instrument (updating the thumb of the scrollbar) and towards the user (moving the contents of the document).

The advantage of instrumental interaction is that the interaction is quick, efficient and intuitive. Examples include the repetitive actions of saving deleting and organizing files. As we mentioned before, the important feature of the instrumental interaction is that it relies on a system-centred design method focusing on a direct manipulation model. Many applications have been developed based on direct manipulation including word processors, virtual reality, video games, and image editing tools (Rogers et al., 2011). One successful example of the direct manipulation model is the desktop work environment. Apple was one of the first companies to explore an operating environment using direct manipulation interaction as its central mode of interaction. This type of interaction has been used for last 30 years and currently you can find it on Mac OSX and iOS.

The weakness of instrumental interaction is that it is unable to adapt to the user's dynamic responses. Because instrumental interaction carries out a series of actions: instructing and manipulating, each formed in a predefined format. Instructing describes how users carry out their tasks by following a system structure that guides the system's response to user commands. That means the user's commands are carried out in a closed sequence with the predefined system responding - instrumental interface. Manipulating focuses on operating the objects and capitalizes on user's knowledge of how they do so in the physical world. This can be done in a number of ways, including typing in commands, selecting options from menus in a WIMP interaction environment or on a multi-touchable screen, speaking out a command, gesturing, pressing a button or using a combination of function keys.

Human-centred interaction design method

The second design method is human-centred interaction design focusing on the human factors. Like some researchers indicate that the system should instead be regarded as a communicating agent that treats the computer as a partner (Beaudouin-Lafon, 2004). Human-centred design approaches emphasise the role of the user in the design cycle, and derives desirable experiences through the involvement of users in the design and testing of prototypes.

The essence of human-centred interaction design is to predict how the user would like to interact with the outside world (Langley et al., 2009). The users might be taken into account in different phases. For example, the users are considered early in the design cycle in order to provide information on the desired features and influence the system development the earlier stages. After that, the interactive design development then proceeds through iterative refinements whereby user feedback is used to modify and further elaborate the initial design concept (Rogers et al., 2011). Many research works employ the notion of human-centred design method, including computer supported cooperative work (CSCW) (Bull and Vatrapu, 2011), experience design (Wright et al., 2008), emotional design (Lim et al., 2008).

The focus of human-centred design is to generate diverse user models grounded in human cognitive psychology. These models are then used to produce human computer interaction. For instance, Norman developed a number of models of user interaction based on theories of cognitive processing, arising out of cognitive science. These were intended to explain the way users interacted with interactive artefacts. The model includes seven steps of actions that describe how users move from the conceptual idea to executing physical actions needed to achieve their goals (Norman, 1988).

From the above perspective, many research fields focus on exploring how to predict human interactive activity with diverse artefacts in different design contexts. Designers utilize different techniques to analyse the user's reflections and needs including Artificial Intelligence (AI), machine learning and ethnographic technique. A major purpose of AI and machine learning research is to embody anthropomorphic means of communication in the computer to cooperate with the users to complete a particular task. Some results of the research have been used in different areas such as emotional computing and Natural Interfaces (NI). On top of that, in terms of human computer interaction design, designers largely employ ethnographic research techniques such as observation, interviews and personae to understand and conceptualize a target user and their needs related to an interactive product (Cooper et al., 2012).

However, in many cases the problem is that user modelling is over inclusive, so that, by the time one has built a complete model of the user's assumptions and then tested it against a separate model of the system, it is difficult to match all the users' needs (Ryu and Monk, 2009). Because that the traditional ways in which people study human cognition and ways of human communication (such as document analysis and natural language processing approaches) try to obtain semantics from text and data that are limited in ability to contain human-level or society-level semantics (Zhuge, 2010). We will return to the problem of semantic communicating in chapter four.

In addition, as these models are mainly generated from ethnographic studies, they are often difficult to use by a designer in a real design practice. Ethnographic methods do not provide designers with a concrete design framework. More important issue is that the users' interaction cannot address all the interaction problems raised by individual users. In other words, embodied interaction, then, runs into trouble satisfying different requirements from diverse users as the interaction is designed to carry out human actions in a particular way rather than from the users themselves. Therefore, although embodied interaction is produced from comprehensible user modelling, it cannot fully enable the user to achieve the goal in a way that the user intends to use the computer during interaction with computer.

The results of research show even though many systems are advertised as being highly-customizable, very few support personalized human computer interaction. For example, some systems extend the idea of selection from different menu sets to allow users to change menu labels, key bindings, and menu composition and even to create macros, but the user is constrained to using the functions the system developer has provided, and unable to customise the actual behaviour of the system effectively.

Interaction perspective design method

The third interaction design method is the interaction perspective model. In contrast to user modelling, it describes interaction not only purely in terms of the knowledge of the user cognitive system, but also the quality of human computer interaction, explicitly considering the effects of user actions on the system and the resulting system feedback on the user (Ryu and Monk, 2009). The resulting interaction is expressive interaction that helps the user construct a relationship to the product (Forlizzi and Battarbee, 2004).

The Interaction perspective model proposes that a more useful level of abstraction can be achieved by including both the users' cognitive processes and the response of the system at the first level of description - the conceptual level. An early view of this model is captured in Monk and Dix's (1987) triangular model. The model shown in Figure 2. 2, it illustrates the possible expectations of design thinking about the process of human computer interaction (see Figure 2. 2). Monk and Dix provide an example of saving a document onto a floppy disk along with relevant state descriptions to exemplify this interaction model. To complete this task, the user needs to have the overall goal save the document onto a floppy disk. And then the user will seek to take relevant action to accomplish this goal. In this case, the user will most likely look through the menu to find tab “File”. The chosen action triggers system effects such as when a user aims to find File from a drop down menu, new system effects are presented, these changes are perceived, and new goals are generated or old goals eliminated (the effect to goal path). The new goal initiates another cycle until the original goal is accomplished (Monk and Dix, 1987) .

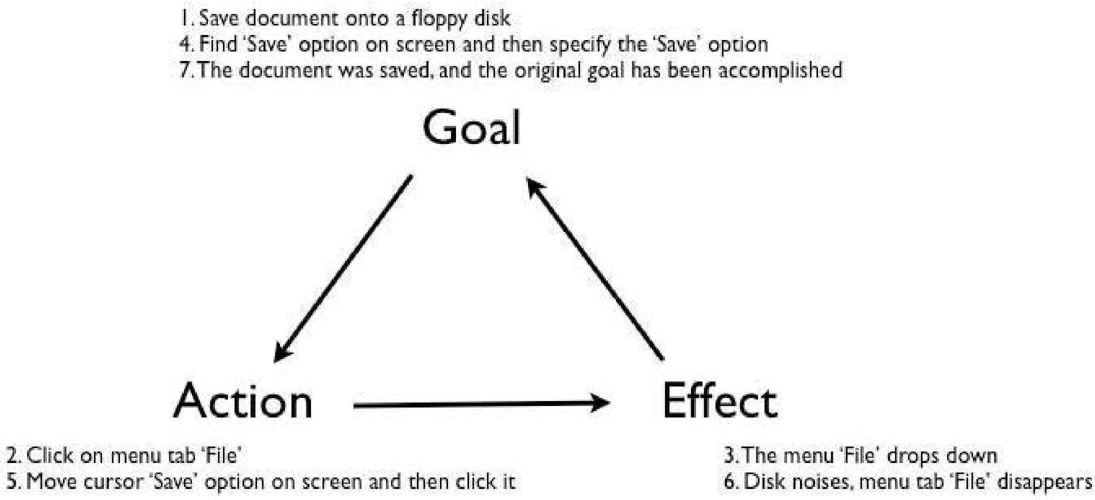


Figure 2. 2 An example of interaction perspective model (Monk and Dix)

From this point of view, conversation needs to enhance the richness of the interaction and allow more complex information to be conveyed than is possible in a single utterance such as a predefined interface. In this view, language understanding in interactive contexts becomes more complex. It is this third view of interaction that we choose to take here.

The way of building mutual interaction is to design and build human computer interaction in a personal way. Ryu and Monk argue that such an approach will lead to a much more effective collaboration between human and computer (Ryu and Monk, 2009). Forlizzi and Ford defined this type of interaction as expressive interaction (Forlizzi and Ford, 2000) It comes out with two significant features: first, the interaction provides the opportunity for the user to state what he/she wants to do in his/her own terms, just as he/she would to do with another person. The system takes care of the complexity. Second, the interaction enables the user to solve his/her problem in different ways by collaborating with the computer.

As a result, the expressive interaction not only carries out a series of commands, but involves the user's defining and exploring ways to perform the tasks, and collaborating to get them done. More importantly all human computer interactions are contextually interpreted with respect to the interactions performed up to this point, allowing the system to anticipate the user's needs and provide responses that best match the user's goals. Suchman observes that human computer interaction needs to "explicate the relationship between structures of action and the resources and constraints afforded by physical and social circumstances" (Suchman, 1987). By doing so, we could create a new paradigm for human computer interaction.

To achieve this, it is necessary to have a design approach for creating interactive products. As Monk states, the interaction perspective model provides a more useful level of conversation by integrating both users' characteristics and system capability. Unfortunately, although this interaction design model is of value helping the interaction designer to reason about what the user's cognitive process achieves, as well as what triggers that process, there is very little research on how to apply this interaction perspective model to build expressive human computer interaction (Ryu and Monk, 2009).

After investigating how designers create a variety of interactions by using different design methods, in the following section, we will discuss how users experience the above interactions.

2.3.2 Experience of variety Human Computer interactions

Interaction experience is concerned with user-product interaction in a meaningful context (McCarthy and Wright, 2004). More important for interaction design is the exploration of how to integrate the human cognitive system and the performance of a product into a meaningful interaction that directly affects users' experience of the interaction. That means an appropriate interaction must correspond with and adapt to the user's individual goal and need to complete a specific task naturally.

The research shows that building meaningful human computer interaction, that can yield desirable user experiences, requires that the interaction be as close as possible to natural communication. This communication should follow principles of interpersonal interaction in two key respects: supporting different levels of communication; and leading to the intended emotional experience on the part of the user.

In order to analyse different interactions and their relevant experiences we take advantage of an interaction experience evaluation framework created by Forlizzi and Ford (Forlizzi and Ford, 2000). This framework provides a comprehensive system to describe user product interactions, and their corresponding experiences. In the following sections, we will categorize current styles of interaction including instrumental interaction and embodied interaction, into the above framework. Through this, we investigate different users' experiences generated from the above human computer interactions. Additionally, we explore the interrelationship between users' interaction and their experience and discovering a way to help them to achieve their desirable experience from interaction.

Following the interaction experience-evaluating framework created by Forlizzi and Ford. Interactions and user experience can be divided into three categories: fluent interactions, cognitive interaction and expressive interaction; and the corresponding interaction experiences are: experience, an experience and co-experience (Forlizzi and Ford, 2000) (see Table 2.1).

Types of User-Product Interactions	Description	Example
Fluent	Automatic and skilled interactions with products	• riding a bicycle • making the morning coffee • checking the calendar by glancing at the PDA
Cognitive	Interactions that focus on the product at hand; result in knowledge or confusion and error	• trying to identify the flushing mechanism of a toilet in a foreign country • using online algebra tutor to solve a math problem
Expressive	Interactions that help the user form a relationship to the product	• restoring a chair and painting it a different color • setting background images for mobile phones • creating workarounds in complex software
Types of Experience	**Description**	**Example**
Experience	Constant stream of "self-talk" that happens when we interact with products	• walking in a park • doing light housekeeping • using instant messaging systems
An Experience	Can be articulated or named; has a beginning and end; inspires behavioral and emotional change	• going on a roller coaster ride • watching a movie • discovering an online community of interest
Co-Experience	Creating meaning and emotion together through product use	• interacting with others with a museum exhibit • commenting on a friend's remodeled kitchen • playing a mobile messaging game with friends

Table 2.1 Summary of a framework of user experience as it relates to the human computer interaction (Forlizzi and Ford, 2000)

2.3.2.1 Fluent interaction and experience

According to the definition given by Forlizzi and Ford that when interaction is "fluent", it does not compete for our attention, and people can focus on the consequences of their activities or other matters (Forlizzi and Ford, 2000).

Obviously, this type of interaction mainly focuses on specific human activities like direct manipulation such as instrumental interaction. A vital feature of this kind of interaction is that it is linear. Generally speaking linear interactions are broken into chunks called by the main loop independently from each other (Myers, 1991). Usman Haque argues that typical human computer interactions, such as typing, clicking and dragging, are not interaction. They are merely reaction, just as an automatic door opens as people pass in front of it (Haque, 2006).

The sequence of making this interaction has a logical flow that the users need to follow in order to complete a task effectively. For instance, using touch interface to control an objective. Usman Haque argues that for the existing interaction between human and computer, the transfer function composed of input and output is fixed, while in terms of interaction, the outcome of the interaction should be dynamic and reasonable (Haque, 2006). For example, in term of dynamic interaction, the precise way that input affects output can be changed by the end users from individual perspectives relate to a presentation (how the widget is displayed), behaviour (how it reacts to user input) and an application interface (how it signals its state changes and how its state can be changed by the application).

Therefore, instrumental interaction works well for direct interaction with an object, but works less well for complex tasks (Beaudouin-Lafon, 2004) and very poorly at translating user's abstract concepts into the computer (Oviatt, 2003).

The user's experience is part of the constant stream of "self-talk" that happens while we are conscious. It is formulated in relation to the useability of the product. The user's individual characteristics, which have a significant effect on the quality of user' experience of interaction, are not taken into account (Forlizzi and Battarbee, 2004) (See Table 2.2)

Model of user –product interaction	Key features of interaction model	User interaction experience	Examples
Fluent interactions	Fixed input and output	Experience (Usability)	Instrumental interaction- clicking button for next page

Table 2.2 Instrumental interactions, and users' corresponding experience

2.3.2.2 Cognitive interaction and experience

The second type of interaction is cognitive interaction. Cognitive interactions can result in knowledge, or confusion and error if a product does not match anything in our history of product use (Forlizzi and Ford, 2000). Embodied interaction belongs to this type of interaction. Normally, a designers' composed interaction provides the user with decision points allowing multiple branching in different contexts.

Conversational interaction is generated from Cognitive Interaction Modelling provides the user with a more useful level of abstraction that can help the user to understand their interaction with the computer. It achieves this by including both cognitive and system aspects at a meaningful level. In other words, the interaction designer starts to design an interaction from reasoning about what the cognitive process achieves as well as what triggers that process. At a result, designers can employ appropriate design alternatives and uncover relevant design issues using analytical methods to understand the real user and the context of interaction. At the same time, this interaction model is more comprehensive than the fluent interaction model as it addresses some well-known problems with user-computer interaction by attempting to reduce misfit and mismatching and therefore increasing user satisfaction (Guarino and Poli, 1995). Like we mentioned before, human centred interaction design method attempts to adjust their design decisions by adapting how the user is going to complete a particular task and what are the user's reflections (Ryu and Monk, 2009).

Nevertheless, this type of interaction model only covers user's needs for interaction at certain levels. On one hand, designers will be much better positioned to design systems that could match the way people behave and use technology if they begin by considering the actual details of a work practice. The benefits of doing so could then lead to the design of systems that are much more suited to the kinds of interpretative and problem-solving work that are central to human computer interaction (Suchman, 1987).

Alternatively, in many cases the cognitive interaction model is over inclusive, it takes too much time to build a complete model of the user's assumptions and then test it against a separate model of the system (Ryu and Monk, 2009). The cognitive interaction model tries to include as much information of a user's model as possible, based on a designer's analysis and understanding (Krippendorff, 2005). The result of the designer's understanding will determine how to sketch a variety of interfaces (GUI, TUI) and interactions (multimodal).

One example is Semantic User Interfaces (SUI), which contain pieces of contents with specific semantic meaning known by studying the user (Tilly and Porkoláb, 2010). This knowledge is used to produce a variety of models of users' mental states while using the applications. Semantic meaning is encoded by implementers into program components and data structures according to information gained about the application domain during system analysis and design. Interaction is thus seen as basically a mapping between the user's mental model and the representations of domain knowledge incorporated in the application. This mapping is nowadays implemented by hardcoded, application-specific associations between user interface and application level components using event handlers and relatively simple protocols. We call this feature strong semantic coupling between user interfaces and underlying application layers (Tilly and Porkoláb, 2010) (see Table 2.3).

Model of user interaction	Key feature of interaction model	User interaction experience	Examples
Cognitive interactions	Adaptable, interface, Conversational interaction	An experience (Engagement)	Completing a particular task by using Microsoft word or drawing system

Table 2.3 Cognitive interactions, and users' corresponding experience

The challenge to build a completed a suitable interactive artefact like adaptive interface forces designers to deal with a gulf of customization (Bentley and Dourish, 1995). The gulf of customization can be represented as the mismatch between interface and application functionality, in other words, the system does not reflect the customisation requirements of system's users (Bentley and Dourish, 1995). Bentley also indicates that, to create a collaborative system, we need "an approach that places the emphasis on customisation rather than rigid structures and policies governing system behaviour”. He emphasises developing systems, which users can adapt to meet their requirements, rather than systems that constrain interaction to some model of how they perform their work - manipulating representation instead of being manipulated by them. (Bentley and Dourish, 1995).

Alternatively, the other challenge for cognitive interaction is that current user modelling is built upon descriptive theory, and can be hard for designers to use in practice (Langdon et al., 2012). For example, ethnographic methods cannot provide designers with a comprehensive design framework, and specific users’ mental models are not stable and reliable enough for creating comprehensive interaction (Langdon et al., 2012).

As a result, the cognitive interaction model works very well for some of the users within certain interaction situations but possibly less well for other users as diverse users have different cognitive abilities and reflections regarding the same artefact. As we can see, the cognitive interaction model generally concentrates on designing an explicit interaction framework, such as creating interface and specifying a model of interaction, rather than providing a medium enabling users and computers to cooperate. In many cases, designers pay more attention on matching the interaction to the users’ current capabilities. Because that human’s individual development and reflection are not fully addressed. We contend that the best way to create effective interaction in terms of individual context is to allow the user to shape their interactions with the computer. Therefore, to be successful, interaction design must take users’ specific, personal characteristics into account.

2.3.2.3 Expressive interaction and experience

The third type of interaction is expressive interaction. Expressive interactions are interactions that help the user form a relationship to a product, or some aspect of it.

Generally speaking, the cognitive interaction model generally concentrates on designing an explicit interaction framework, such as creating interface and specifying a model of interaction, rather than providing a medium enabling users and computers to cooperate. In many cases, designers pay more attention on matching the interaction to the users' current capabilities. This concept has been emphasis in interaction perspective design method that we have mentioned before. Although human's individual development and reflection are not fully addressed by this design method but we contend that the best way to create effective interaction in terms of individual context is to allow the user to shape their interactions with the computer. Therefore, we agree that to be successful, interaction design must take users' personal specific, personal characteristics into account.

Eventually, through the expressive interaction, the users may consistently express their intention and emotion to a system that they may or may not feel at that point in time. A personalized interaction is derived by an expressive interaction and semantic user interface (see Table 2.4).

Model of user interaction	Key feature of interaction model	User interaction experience	Examples
Expressive interactions	Personalized interaction Semantic User Interface	Co-experience Emotion experience	Operating a system in a personal way

Table 2.4 Expressive interactions, and user's corresponding experience

In the following section, we will discuss how we use different languages produces the above interactions. On top of that, we will explore a way that how to create mutual interaction by integrating language and human computer interaction.

2.4 Human Computer Interaction and Language application

Designing and evaluating human computer interaction requires a comprehensive understanding of the potential end users and of human communication skills (Card et al., 1986). These skills have developed over our abilities to perceive the world through our multiple senses to act upon objects by physical action and communicate with other humans using language (Baudel and Beaudouin-Lafon, 1993). From the above perspective, language is used to support all or part of human interaction. There is a context of language users, physical objects, and practices that give meaning to the communication between the users and computer (Berlo, 1960).

Many researchers in the interaction design community consider language as the primary dimension of human cooperative activity (Winograd, 1986).From a language/action perspective, we can say that people act through language. The fundamental purpose of bringing a linguistic perspective to interaction design is to emphasise an issue of collaboration between human and computer (Winograd, 1986). Andreev indicates that the linguistic perspective derives some important features of human communication. For example, some researchers' have developed pattern languages to build concrete interfaces and interaction models (Van Welie and Van der Veer, 2003). Pattern Languages of interaction design focuses on solutions that improve usability. While, the need for building conversational common ground, without which a pair of participants (like the user and the computer) would not be able to exchange information. As we discussed before, human communication is:

- Contextual, that is, generated by a specialized and shared language;
- Bidirectional and a balance of information exchanging process.

Thus, the solution to the problem of constructing mutual communication between different participants is centred on language (Andreev, 2001). Erickson, for example, has argued that it is essential to create a common language in designing HCI. In addition, he suggests that pattern language is a potential common language (Erickson, 2000a, Erickson, 2000b).

In term of designing human computer interaction, designers mainly use two different languages to convey their design concept through different approaches (Djajadiningrat et al., 2002): natural language focuses on semantic approach and using programming language and interaction design pattern language

- For semantic approach, the basic idea is that using the knowledge and experience of the user the product can communicate meaning using symbols and signs (Krippendorff, 2005). For example, the appearance of product and its controls become signs, communicating their meaning through reference.

- The basic concept of direct approach is that meaning is created in the interaction. That means the meaning generated from humans' perceptions and what they do with their bodies directly depends on individual perceptual and bodily skills.

Next, we will discuss how the above languages that are employed to organize interaction and convey the intended meanings between human and computers. From the linguistic pragmatic point view, languages have two key factors, which are required to carry out effective communication: utterance of language and genre of language (Dearden, 2006). He further points out that the form and meaning of the product is shaped by the designer, user, history and genre, but also by the materials available in making the utterance (Dearden, 2006). For the human computer interaction, language utterance is about the content of communication that is represented through specific interface of an artefact. Meanwhile, an interactive artefact provides the users with a way to affect the interactive product through particular modes of interaction that relate to the 'genre' of the language. At a result, the artefact makes sense to the user with a certain meaning (usability, functionality and desired interaction experience) in a specific context.

Our study concentrates on investigating the above key aspects of language usage within human computer interaction in order to clarify the relationship between language and interaction and identify what kind of language is needed to establish effective human computer interaction. In details, there are two factors are studied in the following: *utterance* of language[2] focuses on how to deliver content between human and computer by using different languages, and *genre* of language[3] concentrates on analysing how the languages support the users to archive their goal based on their perspectives and intention.

2.4.1 Programming language

Computer programming languages has be used to create human computer interaction through direct manipulation (Beaudouin-Lafon, 2004). The design process starts from implementation considerations (e.g., available hardware functions or object of interaction) and ends up with concrete standard user interfaces such as Graphic User Interface (Voelter et al., 2013). Additionally, as Dearden points out, programming languages can be considered as an extreme example of notational systems. By using a programming language the designer is able to manipulate the interpretation of interaction that is conveyed between users and computer in a specified formula according to the technology they use or product type they want to build (Dearden, 2006).

Typically, the architecture of programming language depends on a linear conversation formula (Winskel, 1993) that defines a semantic framework that, in turn, maps out the interaction between user and interface through a pre-defined interaction model provided by interaction designers. For instance, when people use software to complete a task the user can directly

---

[2] Utterance: output from a computer intended to convey some meaning to the user through a specific interface.

[3] Genre: the mode of interaction available to the user to interact with the computer e.g. instrument interaction, embodied interaction and linguistic interaction etc.

interact with the computer through a well-organized navigation system and recognizes its corresponding interface, which is formalized and constructed using computer programming language. A basic architecture of computer programming language is shown in Figure 2. 3:

```java
// class declaration
public class ProgrammingExample {

    // method declaration
    public void sayHello() {

        // method output
        System.out.println("Hello World!");
    }
}
```

Figure 2. 3 Screenshot of programming language usage-Java

To better understand programming language usage in a conversational context of human computer interaction, we analysis how the programming language is used to construct human computer interaction from two aspects: utterance and genre corresponds to the user interface and user's interaction model to evaluate the quality of human computer interaction.

2.4.1.1 Utterance of using programming language

Graphic User Interface represents one type of utterance of computer programming language. The most famous type of GUI is the Window Icon Menu Pointer (WIMP) (Beaudouin-Lafon, 2004) (see Figure 2. 4) interface, which is represented as an instrumental interface. The architecture of programming language depends on a linear conversation formula that defines a semantic framework that, in turn, maps out the interaction between user and interface through a specific interaction model. This semantic framework based on What You See Is What You Get (WYSIWYG) (Beaudouin-Lafon, 2004) and realized by a presentation-navigation-selection-action cycle. According to these semantics of interaction, a user interface presents a set of content objects for the user, who assembles requests by navigating desired objects, selecting them, assigning a semantic role to each of them, and performing a terminating action to initiate request executing. Eventually, the system executes the request and returns a result with actually displayed pieces of contents.

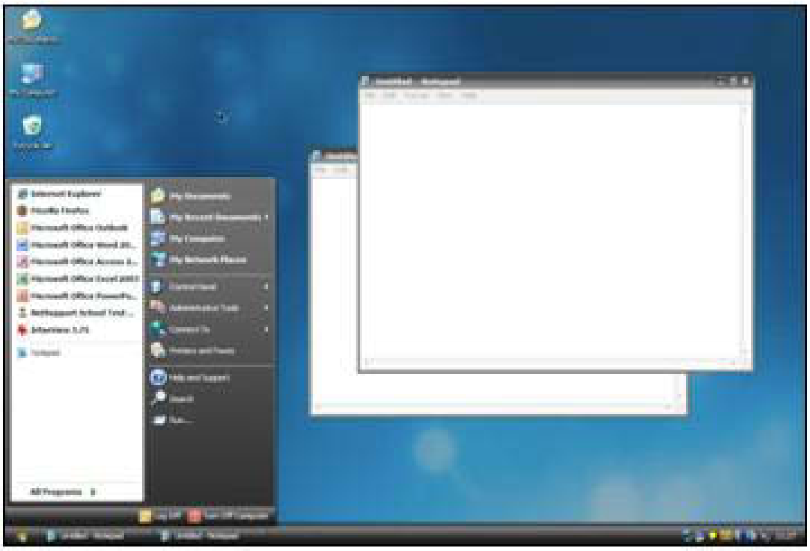


Figure 2. 4 An example of Graphic User Interface (GUI)

The key features of a WIMP interface include an iconic finder, applications with a menu bar and overlapping windows and a limited set of interaction techniques. These include drag and drop and copy-paste to transfer data within and across applications. In many cases, WIMP interface is very suitable for human computer interaction in variety of contexts and has been used for many years (Beaudouin-Lafon, 2004)

However, we realized that WIMP is not well suited to some applications, and lack of technical support increases difficulty for the users to develop their interaction.

For the first aspects, Michel Beaudoiun-Lafon argues that, although WIMP interface has reached their limits because of three major challenges: the exponential growth in the amount of information each individual user deals with; the distribution of this information over multiple computers and devices, including mainframes, desktop, computer, laptops, PDAs and cell phones;

and the growing range of computer users, with their wide variety of skills, needs and expectations (Beaudouin-Lafon, 2004).

The WIMP-based user interface provides permanent interactional architecture to the user by providing windows with menus such as opening certain files and activating certain functions (see Figure 2. 5). To complete such tasks, the GUI presents the user with a single logic set (predefined architecture function layer), implemented within the limited computer screen.

For the second aspect, the main challenge for ordinary users is to modify the interaction by using programming language. For example, there are many tasks that a user must repeat same action many times when using a GUI it does not reflect the end user's purpose, intention and experience. In next section, we will discuss the reason why the users feel difficult to change the interaction based on their perspectives and understanding.

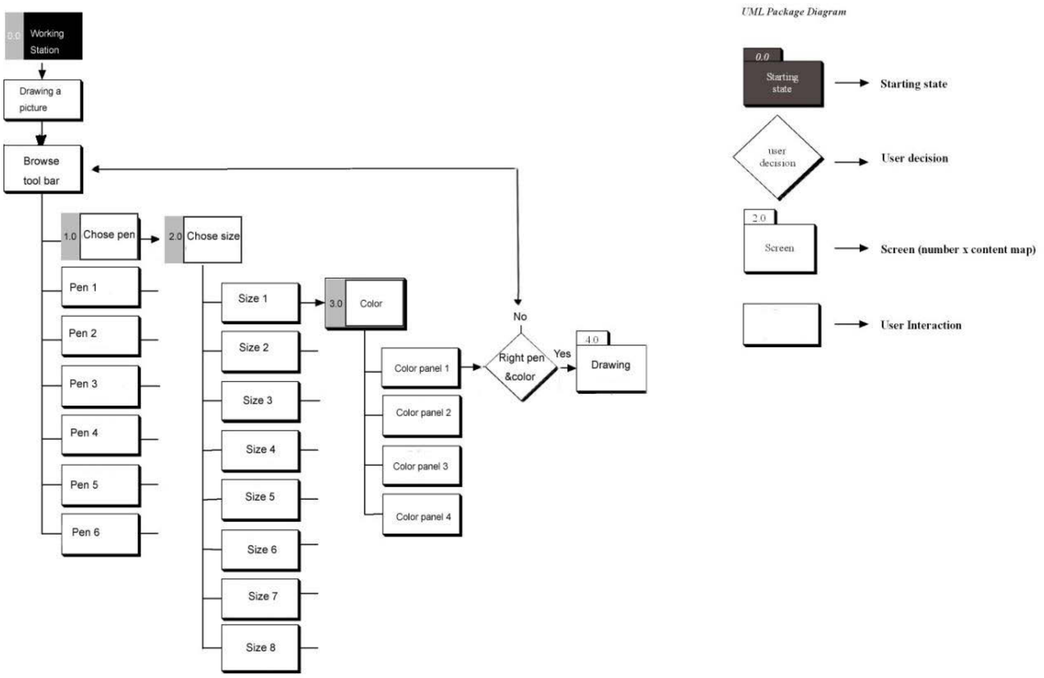


Figure 2. 5 An interaction structure of a user operate a system through GUI

2.4.1.2 Genre of programming language

One important reason causes the above problem is that there is no easy way for the users to modify the interface and interaction model that has been pre-defined by the designers by using programming language. As we know, by using programming language, designers are able to create a type of interaction that lets users communicate with a computer or system directly such as instrumental interaction. The advantage and disadvantage of this type of interaction has been discussed in section 2.3.

Accordingly, one side, programming languages have been used to compose the structure of instrumental interaction, determined by affordances of the tool (Beaudouin-Lafon, 2004) and the middleware that is used to create the interactive system (Hartson, 2003).

On the other side, the fact is that programming languages is programmer-oriented language rather user-oriented. Because programming language does not readily support users' reflection on their actions during the interaction (Pane and Myers, 2006). A research for building a natural programming language aims to make the programming language become easier for normal people to learn and use (Pane and Myers, 2006). Consequently, if the users want to change the results of interaction they need to know how to use the programming language. Following this perspectives, programming languages are seen as a common language between programmer and computer as the programmer familiar with the vocabulary and grammar of computer language. It is critical for programmer to translate his/her idea or command to change the interface or get a rational reaction from computer easily. Accordingly, If a user is able to use the computer programming language then the user becomes a co-designer for his/her interaction (Myers et al., 2004).

Above all, programming language can support people who know how to use the programming language to establish a comprehensive conversation between human and computer. Because it focuses on instructing interactions to obey users' orders rather than supporting users to reach their conversational models.by adapting users' dynamic responses to the interaction. We concern that programming language limits the users' conversation with a computer.

2.4.2 HCI Pattern language

The original concept of Pattern Language was first introduced by Christopher Alexander. He and his colleagues use Pattern Language in architecture and urban design. The pattern language is proposed as one step toward building a more usable design system. (Alexander et al., 1977). Later, the language concept was adapted into interaction design. Early work on patterns in software engineering included solutions for user-interface software design. Since then, the use of patterns in interaction design or human computer interaction design and related fields is gaining increased attention. Meanwhile, many pattern libraries have been published (Tidwell, 1997, Van Duyne et al., 2007) and more are appearing each year. In addition, a number of interaction design pattern languages have been published in book form including Borcher's triple languages for the development of interactive exhibits (Borchers, 2000), as well as many more web-based collections(e.g. Tidwell's design pattern language library (Tidwell, 2010).

To understand the basic definition of pattern and pattern language for human computer interaction, we introduce some well-known characterizations by different HCI researchers.

HCI patterns contain some common attributes that have been inherited from architecture and also have some specific features. The Pattern of HCI has been defined as follows: (Dearden and Finlay, 2006)

- Pattern is a form, template, or model or more abstractly, a set of rules, which can be used to make or to generate things or parts of a thing.

- Pattern is an invariant solution to solve a recurrent design problem within a specific context (Monk and Dix, 1987). For example, some patterns in HCI deal with high-level issues such as business process or task structure (Petrelli et al.), while others address low-level details of GUI construction such as the layout of tables (Tidwell).

- Pattern is part of a language: Patterns are related to other patterns and work together to resolve the complexity of system problems.
- Pattern reflects design values. In other words, patterns are not neutral but explicitly reflect design values. The selection of pattern and the recording of patterns are value-constructing activities, reflecting the priorities and motivations of the designers (users).
- A pattern is grounded in a domain: patterns relate to specific domains and have no meaning outside those domains.
- Patterns capture design practices. From this perspective, patterns are derived from actual practice, not theoretical or conceptual proposals. Patterns can only be completed in the process of actual use by users, so that emphasis is on the processes of identifying and developing patterns.

Some additional definition items come from other researchers (Bayle et al., 1998):

- Patterns provide a "lingua franca", that means patterns support people who are not specialists in the domain. Additionally, patterns in HCI should be accessible and understandable by the end users.
- Different patterns deal with problems at different scales: some patterns in HCI deal with high-level issues such as business process or task structure, while others address low level details of GUI construction such as the layout of tables.
- Patterns capture design practice: patterns are derived from actual practices not theoretical or conceptual proposals.

As in typical design patterns, van Welie's description of each pattern follows the following structure(Van Welie and Van der Veer, 2003) and more patterns can be found in his online interaction pattern language library:

- Name: brief description used to refer to the pattern.
- Problem: a description of the problem to be solved, from the user's point-of-view. This problem should be related to user's goals and tasks.
- When to use: the characteristic of the context of the user that determined when the pattern can be applied.
- Solution: a concrete and illustrated description of the core solution to the problem,
- Why: the underlying rational, describing the value of the pattern and why it works.
- Examples: instances of the pattern in use, i.e., where the pattern has be successfully applied. It typically includes a screenshot, and may be annotated.

Based on the above perspectives, HCI pattern is problem-oriented, yet not toolkit-specific. The patterns provide an opportunity to bring a User Interface (UI) design solution and a software implementation solution in the same place together. Deardan and Finlay suggest that patterns can be used to contribute to interaction design from several aspects:

- to create participation between the user and the computer;
- as technical lexicon; pattern as organizational memory;
- to construct a Lingua Franca and as design rationale (Dearden and Finlay, 2006)

Although a full range of perspectives can be considered, the quality of interaction relates to the users' individual purpose of interaction, which depends on the user's understanding and reaction.

From Alexander's point of view, an individual pattern may already be very valuable for designers, but when patterns are related to each other they can potentially become far more valuable (Alexander et al., 1977) such a set of connected patterns is called a pattern language. Originally, the purposes of Alexander's pattern language were to support non-architects participation in the design of their own environment. Alexander's pattern language addresses this goal by providing its users with a common language that enables them to reflect on their experience and on the relationship between their experiences and their environment according to the Pattern Language of Architecture (Alexander et al., 1977).

Thus, Alexander's pattern language is a kind of meta-language (Erickson, 1998). This means that both the language and individual patterns are changeable and enable the participants to generate common languages for a particular architecture design (Alexander et al., 1977). Following this perspective, Erikson point out that users modify existing patterns and create new patterns (Erickson, 2000a). In contrast to Alexander's pattern language, existing pattern languages in HCI belong to an object-oriented pattern language or problem-centred pattern language (Dearden and Finlay, 2006). For instance, Borchers organizes pattern language into high-level and low-level patterns, in which high-level patterns address large-scale design issues and lower-level patterns describe the solution (Borchers, 2000). Therefore, a Pattern Language (PL) is a group of patterns that attempt to give a solution to a recurring problem (Dearden and Finlay, 2006).

The fundamental concept of a pattern language is that patterns are related to each other, forming a network of connected patterns that support communication between designers responsible for the definition of the overall architecture of a system, and designers responsible for applications software. Norman observed that using pattern language has the potential to organize and affect people's recognition and behaviour (Norman, 1988).

Relationships are at the heart of pattern language because they create actual additional value, such as a design concept over single patterns (Erickson, 2000b). In the interaction design domain, several alternative organizations have been proposed. Fischer investigated some of possibilities for structuring pattern language (Fischer, 2001). Mahemoff classified interaction pattern for tasks, users, user-interface elements and entire systems (Mahemoff and Johnston, 1998). There is general agreement that patterns provide some rationale for particular design decisions, and the structure of such rationales, so different authors producing different interaction pattern languages reflect their different understandings on their design rationale. This highlights a fact that patterns have diverse meanings in different contexts: On the one hand, there is a meaning defined by the position of the pattern in the language that has been composed by a designer, i.e. a pattern language for designing a particular interactive product. On the other hand, in order to make the pattern useful, the pattern needs to refer to the nature of the interaction environment in which the pattern is to be applied by a user. The user's individual situation is the pre-condition (Dearden and Finlay, 2006).

The interaction design itself can be considered as forming an object-specific pattern language, which is used to guide reflection and dialog among the interaction, including the proposed specific interface design and interaction model that carries out the activities of the inhabitants.

2.4.2.1 Utterance of pattern language

In interaction design practice, designers typically create interaction using a hierarchical structure where they start with gaining an understanding of the users and their tasks, the clients' purposes, technical environment, business context etc.

The pattern language is mainly used to produce a user interface with a hierarchy (top-down) design process. On the top level, the designers mainly work out how to create a product to match the target users' needs or specific task. On the bottom level the designers aim to build concrete interface and interaction model for the product. Here, we will give an example presented by Van Welie et al (Van Welie and Van der Veer, 2003). The objective of this example interaction design is to create an interface for an online shopping website. To achieve this, designers lay out the foundations of the application and items of interface in terms of the site concept, information architecture, and basic functionality by using pattern language formula. A concrete interface is complete when interaction patterns are generated at all levels. Meanwhile, Van Welie et al have provided several patterns at different levels in a top-down order to design human computer interaction, namely: posture, experience, task and action:

- Posture pattern, is about the purpose or reason for existence of commercial sites, which usually includes business goals to be achieved. These stated business goals feed into the choice of a 'Kind of site' that is adequate and effective. It describes the essence of the site: what kind of site structure will be used, which elements make up the homepage and also the main experience that such a site is supposed to offer.

- The experience pattern is about the main user goals and tasks that need to be supported and to what extent. The experience is not just about tasks and goals but also about how the users reach their goals using an overall site concept, how they perceive the site and whether it gives them the appropriate satisfaction. Experiences need to be understood as the broader goal for which we are designing. We should note that experience-level patterns describe a common experience and which lower level pattern can be used to create that experience. In this case, typical experiences are activities such as shopping, playing browsing, information gathering, problem solving or sharing thoughts. Here, we give an example of a shopping experience pattern.

- Task patterns are about describing a solution to small user problems that are part of a higher-level experience. Typically a task pattern describes a series of interactions on one or more objects for solving a problem. Such a series corresponds to a task sequence needed to achieve a task goal.

- Action patterns are about real user actions during the interaction with a website, such as having push button or a clear exit. These are actions that are only meaningful in real tasks such as moving to the next step etc. They are often widgets and are the lowest level of building blocks.

The pattern language is proposed to design meaningful interaction by helping the user to recognize the content and use it in an effective way. For example, many of pattern languages have been created to help the designers creating adaptive or adaptable interface and conversational interaction like Tidwell pattern language and online pattern library. The purpose of doing this is to make the interaction, however, the involvement of the user is limited to defining how the web pages are organized and how navigation and interaction should take place.

Therefore, it is different to human natural interaction, According to Bueno and Barbosa there is no de facto standard language for the user to represent their reflections. Therefore, we argue that designer-oriented languages like pattern languages and programming languages may not be enough to address both the designer's and the user's distinct issues at the interface level (Bueno and Barbosa, 2007). There are at least two different levels of representation that must be embodied for a user-oriented interface design: the first level is about the initial abstract or concept of building a user interface delivering designer's message, where most HCI design patterns are defined. The second level, the interface, has to express the user's activities i.e., the definition of the user-system conversation and the processes that occur during interaction, which should be revealed by elements emerged at the user interface (Bueno and Barbosa, 2007).

For instance, the purpose of creating an adaptive interface is to match the functionality of the computer and user's variety of needs and requirements. On top of that, many researchers and practitioners believe that if they can analyse the example users' whole cognitive system and build a comprehensive user model, then they can create a fully adaptive interface. This would fundamentally improve the users' experience of interaction. In fact, it is a big challenge for designers to construct a user-oriented interaction model and develop an inclusive interface to represent the diversity of potential interaction specifics at one time (Langley and Hirsh, 1999).

2.4.2.2 Genre of pattern language

By using pattern language the designer is able to adopt the users based on acknowledge of entire physical and social situation of the users. By doing so, the designers "explicate the relationship between structures of action and the resources and constraints afforded by physical and social circumstances" (Suchman, 1987). To achieve this goal, a range of techniques are used to build diverse interactive product that better adapted (and adaptable) to different context of use. One widely used method is ethnographic (i.e. carrying out extensive observations, interviews and note-taking of a particular setting) in order to understand a context of use. Typically, the findings are contrasted with the prescribed way of doing things, i.e. how people ought to be using technology given the way it has been designed. As a result, different interaction models represent different levels of building interaction: instrumental interaction regarding user's ability on the physical level; and conversational interaction concerning human reactions on the cognitive level. An essential contribution has been on a *cognitive* level, providing accounts of working practices that have provided a background from which to talk about high-level concepts like natural interaction. In details, interaction designers employ multiple pattern languages to create an interaction following different design concepts and purposes. Typically, it start with gaining understanding of the users and their tasks, the clients' purposes, technical environment, business context etc. (Tidwell, 1997)

However, pattern language is mainly used by designers rather than users. The corresponding interaction of using pattern language is conversational interaction. Accordingly, conversational interaction can only work very well in certain domains if the interaction pattern language can be fully understood by the users. It results in a situation where the interaction is restricted by the designer and is relatively inflexible in responding to user's individual and specific needs.

Therefore, we argue that the user should be a co-designer of the interaction in order to produce an inclusive and effective interaction. Different users work differently and, even the same user will apply different interaction patterns according to the interaction context. Clearly, people do not act or interact with an interactive product (system) in exactly the same way as described by these kinds of situated manipulation models (Mackay).

2.5 Problem of HCI design

In current interaction design practices, power has its roots primarily with the designer not with the user. For example, design researchers have explored a variety of interaction forms, new design methods and frameworks, all focusing on how to build a digital product that can satisfy a user's different needs (Ryu and Monk, 2009). In other words, existing interaction design is built upon a one-stop interaction model rather than adopting to real users' reflections in the real world. Consequently, the existing interactions cause many interaction problems between users and an interactive products. Norman summarised it in terms of execution gap and evaluation gap (Norman, 1988, Norman, 2007) (see Figure 2. 6). According to the definition, the gulf of evaluation is about degree to which the system/artefact provide representations that can be directly perceived and interpreted in terms of the expectations and intentions of the user according to the user's personal background knowledge and characteristics. Gulf of execution is the difference between the intentions of the users and what the system allows them to do or how well the system supports those actions.

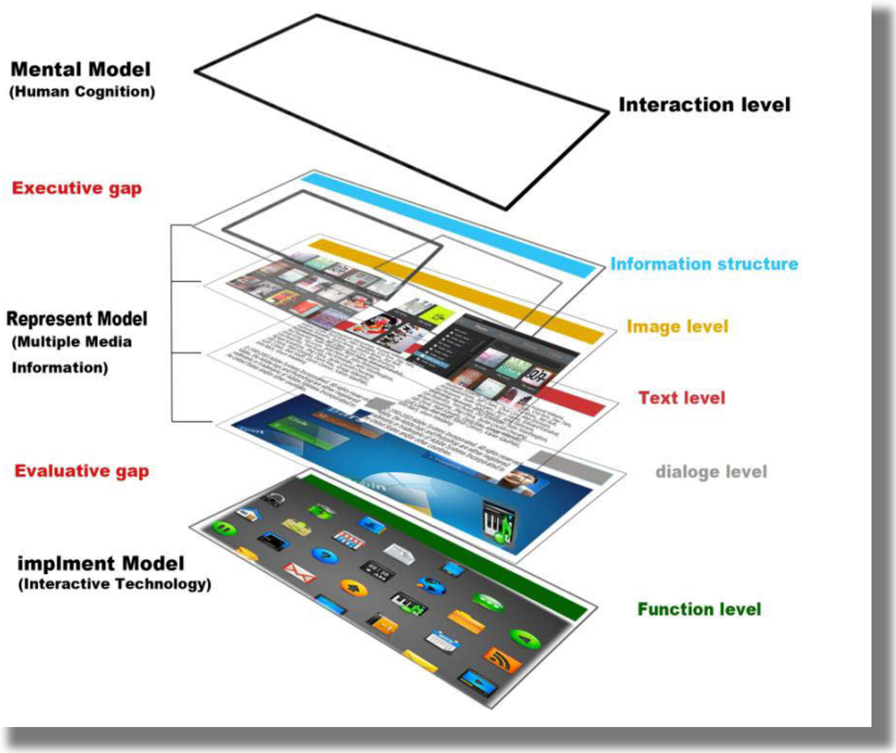


Figure 2. 6 Gaps of Human Computer Interaction

Current human computer interaction is not designed for generating personalized interaction that an individual user's characteristics can be adopted to construct human–centred interaction. This type of interaction will bring a user's with individual emotional experience (enjoyment, pleasure, creative) and result in a personal relationship to an interactive artefact. Thuring & Mahlke provide a diagram to explain how the user's experience is established (Thuring and Mahlke, 2007) (see Figure 2. 7). We claim that existing interaction models limit the users' ability to tailor their behaviour and experiences. For example, a user's sense of making of interaction is not dependent on his/her predisposition (moods, goals, preferences, previous experience) as well as dynamic interaction behaviour. Indeed, the interaction is a consequence of interpreting a predetermined interactive ontology created by designers or producers in a closed design framework.

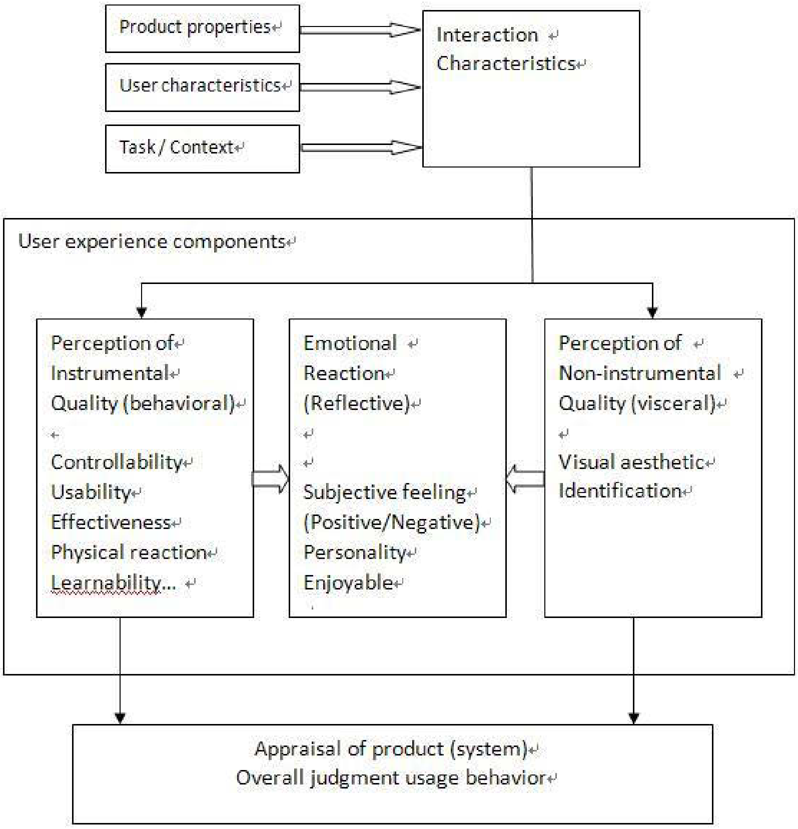


Figure 2. 7 CUM model (Thuring and Mahlke, 2007 p. 262)

General speaking, users are not allowed to make changes to the interaction content and structure until they can use the same languages that are used to create the interaction and interface. As a result, this leads the users into a situation where it is difficult for them to recognize the meaning of the interaction depending on the output of the interactive product. Additionally, the form of interaction of existing interaction is more like a mechanical interaction than human natural interaction. We believe that existing interaction styles such as instrumental interaction and conversational interaction are limited and can be improved by building an expressive interaction based on user's activities.

2.5.1 Interaction design problem

We argue that human computer communication is about transmitting a sense, a meaning based on a computer-mediated communication through physical interaction: interface, interaction model. To build effective interaction, like natural communication, it must take into account the setting up a common ground between human and computer.

To understand the problems, we have to distinguish the differences between human natural communication and human computer interaction. Based on the research of human computer interaction down by Sager that generally human computer interaction belongs to a Scientific and Technical Communication (STC) (Sager, 1994). The human computer interaction is created focusing on forming a process of gathering, organizing, presenting and refining information. The fundamental essence of STC is to correctly convey information to a target audience with unified forms of communication. The primary object of STC is to construct an effective communication by following accuracy, clarity and thoroughness communicative principles (Sager, 1994). This type communication results in a communication that is called *computational communication* or *programming conversation because that* to achieve a goal or tasks, a user is required to complete a variety of activities indirectly and repetitively (Sager, 1994). Indeed, rationality is not a property of a particular act, but related to the participant's goals and circumstances.

Nowadays, designers are facing many challenges in creating interaction that match the users' needs in their daily life, including work, think, communicate, learn, critique, and design. In particular, the challenge is how to create interfaces for millions of users while making it work as if it were designed for each individual user. So the problem of human computer interaction is normally assumed and predefined by the designer based on a second order understanding (Krippendorff, 2005).

Comparing human natural communication with human computer interaction, the existing interaction design systems face three key challenges to create effective interaction to solve the above problems regarding to the principles of human natural communication.

- Engaging users to archive desired interaction experience through the interaction

- Exchanging comprehensive meaning between diversity of interaction participators
- Building mutual human computer interaction through personalizing interaction based on the user individual perspectives and intention.

In the following, we investigate the above key factors of human communication and explore the problems of human computer interaction using current interaction design methods. This can help us to clarify how to create reasonable and meaningful human computer interaction, aligned with principles of communication.

2.5.1.1 Type of interaction

At the beginning of this section, we discussed the level of human communication. It is a key consideration in making meaningful interaction for humans. Allwood identifies three types of communication: physical, psychological and social (Allwood, 1976). As we know, human natural communication covers these three types of communication but the core of human communication takes place mainly on the psychological and social levels (Allwood, 2007).

In contrast, existing human computer interaction mainly works on the physical and psychological levels (2.3). Physical communication is constructed as an object-oriented interaction. In Psychological communication, a users' internal cognitive development is not transferred to their behaviour because the user's activity is constrained by the available interaction with the computer. Unfortunately, in current interaction design it is a challenge to enable the user to generate an expressive interaction (see table 5).

Currently, social or personal level communication between human and computer is not really addressed. Some research areas such as Artificial Intelligence (AI), Social Computing (SC) and Computer-Supported Cooperative Work (CSCW) are moving toward this from different perspectives. In the interaction design community, many interaction researchers and practitioners are working on creating reasonable interaction like emotional interaction (Lim et al., 2008) and User Experience (UX) design (Wright et al., 2008). The above approaches typically concentrate on perspectives from one of two participants: computer or human.

In our research, we consider the best way to solve the problem is to let the user choose the level on which they want to interact with a computer (see Table 2.5). What we are interested in is how to design an interaction that can engage a user to make interaction personalized according to their various culture, language, knowledge and purposes.

Level of Human-product interaction	Human computer interaction	Examples
Physical level	Instrumental interaction	Clicking button in order to move to next page
Psychological level	Conversational interaction,	Completing a task: Typing the words; drawing
Affective level	User personalized Interaction	Creative interaction (Personalizing system)

Table 2.5 Levels of HCI

2.5.1.2 Experience of human computer interaction

Conventional human computer interaction is a process of persuasion, which often demands objectivity to convince an audience (Sager, 1994). Fischer proposed different views of communication influencing designers of human computer interaction (Fischer, 2012). He mentioned two structures of interactive communication: alignment communication and collaborative communion. For the first structure, communication happens, partly based on automatic responses to the communication partner's behaviours. That means people are only contributing to the interaction process when they complete a prefixed communication for a particular task or goal. Alignment, in this view, assumes that the user's interaction is produced based on automatic and subconscious response to the computer's utterance. Therefore, in many cases of HCI, the interaction is more likely to align a user with the computer, or align the computer to a typical group of user, than to make a real collaboration between two of them. In human computer interaction, dialogue technology and industrial applications support fairly well structured, goal-directed dialogues with well-defined user inputs and system responses (Jokinen, 2009). A drawback of this type of interaction is that this type of interaction lowers the level of collaboration because the users who express their needs freely may not get appropriate responses. This will lead to misrecognitions, misunderstandings, and other problem in the interaction (Bogdan et al., 2008).

Compared to human-to-human communication, human computer interactions have so far been very limited, both in subject domain and in communicative task. In other words, there has been very limited cooperation during the interaction. In particular, the problem with dialogue between human and computer is that the response generation is usually seen as a one-shot process whereby the system is supposed to give the best and most appropriate response at once (Jokinen, 2009). It is difficult for the designer to give a right answer without knowing the user's expertise, etc. In contrast, in natural communicative dialog, people usually ask follow-up questions, elaborations and re-explanation, and monitor the decisions and responses made by the partners.

Recently, more and more attention has shifted to constructing collaborative communication based on human natural communication. They are interested in modelling interaction itself by introducing an interaction unit design framework to help the interaction designer to reason about what the cognitive process achieves, as well as what triggers that process (Ryu and Monk, 2009). These models explicitly consider the effects of user actions on the system and the resulting system feedback to the user. The type of model is similar to an interaction-centred model, assumed to provide a more useful level of abstraction and be achieved by including both cognitive and system aspects at the earlier stage of design. Nevertheless, they did not give a concrete framework to illustrate how to use it in a real design practice and, to date, it is just a theoretical framework. In the context of a collaborative situation where different tasks are performed automatically with the help of sensing and interactive technology, the system tends to be regarded as a communicating agent. By doing so, an individual user's characteristics can be taken into the consideration when designing human–centred interaction.

In summary, alignment structure of interaction is still playing a critical role in human computer interaction. Consequently, within computational communication the users are not allowed to make expressive interaction or cooperate with the computer in a balanced way. This is caused by a problem of HCI: the language of HCI. On one side, there has been over emphasis on treating computer as a partner; while on the other side, the metaphor of the computer as a tool does not lend itself to designing such full-blown communication (Beaudouin-Lafon, 2004). A balanced situation might be to consider communication in the context of a smart environment where different tasks are performed automatically with the help of sensing and interactive technology, the system tends to be regarded as a communicating agent.

2.5.2 Language problem

From a communicative perspective, the major problem of human computer interaction is a language pragmatic issue (Branigan et al., 2010). The problem of interaction is that, compared with human-human communication, computer-human interaction lacks a common language that helps the user construct a personalised environment to support a more effective, 'collaborative' interaction. While, the existing languages used in human computer interaction are hard for the users to construct a common ground as most of them are mainly designed and used by the designer or software developer rather than the user. On one hand, programming languages and pattern languages are not suitable for normal users because the users are not trained as professional software developer or programmer (Pane and Myers, 2006). On the other hand, pattern languages of interaction design focuses on solutions that improve usability. Although, interaction design patterns can benefit human computer interaction in certain levels, they lack the appropriate focus and foundation for positional users. For example, the problem statement in Tidwell's patterns language is typically of the form "How can the artefact best show..." which does not explicitly relate to a usage problem of users (Pan and Stolterman, 2013).

Alternatively, much research and practice has explored the use of natural languages to control computers, including Natural Language Interface (NLI) and Natural Language Processing (NLP) (Valencia-García and García-Sánchez, 2013). As we mentioned at the beginning of this chapter, natural language has the ability to carry out a generic interaction between human and computer such as in a Natural language Interface (NLI). However, it is difficult to build user-specified collaborative interaction. Because natural language is too complex to be understand by computer, it results in many ambiguities and mistakes during the interaction (Clay and Wilhelms, 1996). Therefore, this kind of human computer interaction does not allow the user to have an emotional conversation with the product. For instance, many people have used iPhone Siri, although Siri can understand some simple conversations and give a reasonable response in many cases, users are not happy with the conversation.

Clark points out that language should be used to construct a common ground based on sources of conversation and ability to coordinate. In addition, common ground is a set of knowledge, beliefs and suppositions that the conversant' believe they share (Monk, 2009). Therefore, a common language, which can be used to establish common ground between and interactive artefact is needed. Consequently, interaction makes it possible for a speaker and a hearer to coordinate on what the speaker means and what the hearer understands the speaker to mean (Clark and Marshall, 2002). For instance, in human computer interaction, the users monitor the conversation for evidence that the other partner, the computer, is or is not communicating sufficiently well for their purposes. Thus, each of participants should be able to take appropriate actions to make mutual interaction.

2.6 Findings from literature review

2.6.1 Personalization human computer interaction

Pappas indicates that an overarching goal of interaction design is to develop products that elicit positive responses from the users such as feeling at ease, being comfortable, and enjoying the experience of using the provided interactive product or technology. For the designers, they need to be concerned with how to create interactive products that elicit specific kinds of emotional responses in users, such as motivating them to learn, play, or creative or social (Pappas, 2011).

Therefore, a very important task for designers is to create desirable experiences for users when they use a provided interactive product. Interaction design needs to focus on constructing the ways people interact with objects and systems - the product of interaction is almost entirely the quality of the user's experience. Moreover, Coiera suggests that rather than focusing solely on characteristics of individual technologies or psychological and social issues, these can be combined to explain the overall decisions that individuals make when using technologies (Coiera, 2003). The purpose of personalizing human computer interaction is to produce pleasurable and effective human computer interactions. In interaction design, personalized interaction generates an individual interaction experience that is conceptualized in terms of emotion, pleasure, and user experience: Norman's (2007) emotional design model (Norman, 2007); Jordan's pleasure model for product design (Jordan, 2002), and McCarthy and Wright's technology as experience framework (McCarthy and Wright, 2004) all approach this in different ways.

Several technologies have been used to build personalization into human computer interaction. These include using Artificial Intelligence (AI), user modelling and emotional design or experience design. In this thesis, we focus on investigating the design methods that are used to build a variety of human computer interactions.

2.6.2 User-oriented interaction language

To build a common ground between human and computer requires a common language to let participants understand each other. In terms of HCI, from a linguistic pragmatic perspective, language is used to construct common ground by changing the behaviour of the computer, including interface and interaction model (Clark, 1996). From this point of view, human computer interaction is accessible and changeable to the end user and it is the basis of developing an emotional interaction.

Language is used in a controlled way in HCI, which implies conciseness, explicitness, and accuracy. In other words, pattern language and programming language form an essential barrier to establishing user-oriented interaction. Thus, the language used in HCI is the key to exploring a user-oriented interaction language which can used in human computer conversation as in human natural communication.

2.7 Chapter summary

In this chapter, we investigated a variety of languages used in HCI with particular emphasis on users and the problems they may encounter. On a practical level, we focus on actual use, co-ordination and interpretation of language in practical interaction between human and computer. The results suggest that interaction design should provide a communication medium to support diverse types of conversations to improve the quality of human computer interaction. To make the interaction between human and computer become more efficient and meaningful we propose a communicative tool that takes into account the ideas discussed so far.

In this literature review, we have studied principles of human-to-human communication and have identified some important characteristics of natural interaction that can be a guideline for designing human computer interaction (2.3.1). Several important features of human interaction are missed in current interaction design methods and it leads to undesirable experiences of interaction (2.3.2). We claim that the main problem is that we do not have a common language in human computer interaction (2.5.2).

Human-computer interaction can be seen as a communication situation where the participants have various kinds of intentions and expectations concerning the course and flow of the communication. Therefore, the existing human computer interaction designs have to find a suitable way that are able to fix all of the problems that arise when communicating with the computer (2.6).

The key object of our research is to find a way to generate a meaningful and effective interaction based on a user-oriented interaction between human and computer. It is time to establish a user-centred systematic design pattern to organize context-sensitive interaction content and user's interactive behaviour to create a meaningful interaction.

Chapter three: Methodology

Chapter two explored the principles of human communication and functionality of language in producing natural communication. This chapter will define the research approach and method used to investigate an interaction design pattern to support the customization of user interaction using domain specific interaction language. First, a methodological review is carried out (Section3.1). Second, a detailed study method is selected and its criteria for this research context are introduced (Sections 3.2 and 3.3). In particular, the methods for data gathering and data analysis are described (Section 3.4 and 3.5). Third, the research usability testing conducted is described (Section 3.6).

3.1 Methodological approach

In the last section, we initially analysed the functionality of language for human communication and then we claimed that it is possible to create effective interaction between users and an interactive system[4] by building a DSIL which allows for personalized interaction. The personalized interaction is composed of semantic interaction (Kohlhase, 2008) which can establish reasonable interaction based on the users' interactive meaning to derive their desired interactive experience. To achieve the goal, in this thesis we propose an Interaction Language Design Pattern (ILDP) to produce personalized human computer interaction. The design pattern specifies a Domain Specific Interaction Language[5] carried out by text-based interaction.

In this approach, interaction designers specify visuals, sounds, gestural inputs, etc. which are used to create an interactive system for particular end-users such as illustrators, novelists, etc. through a well-defined domain

---

[4] Interactive system: Some kind of designed artefact which users interact with. e.g. drawing system-Photoshop.

[5] Domain specific interaction language: a specific visuals, sounds, which are employed to convey the interactive meaning of an interactive system to particular end-users.

specific interaction language. Users are then able to personalize the interactive system by customising the domain specific interaction language.

In order to evaluate how well or badly the proposed design pattern supports designers in the creation of systems which support personalized human computer interaction we chose a well-known evaluation approach: the D E C I D E framework to collect and analysis the data (Rogers et al., 2011 p348). The D E C I D E provides a template to help us design our evaluation method. The steps in the D E C I D E framework are:

- Determine the overall goals that the evaluation addresses.
- Explore the specific questions to be answered
- Choose the evaluation paradigm and techniques to answer the questions
- Identify the practical issues that must be addressed, such as selecting participants
- Decide how to deal with the ethical issues
- Evaluate, interpret and present the data

Our main purpose here is to evaluate a design method for building systems, which support personalized interaction between the end user and computer. As we discussed in chapter 1, the research question is: how can we create personalized human computer interaction that fosters comprehensive communication between users and the computer? In order to assess whether the ILDP is useful for designers to create effective interaction between diverse users and interactive systems and how the users experience the given interaction, we need some form of usability testing.

There are many different evaluation methods, some of which involve the user directly like usability testing(Nielsen, 1994). Others such as predictive analytical evaluation do not involve users directly (Nielsen et al., 2002). Depending on circumstances, the evaluation can be conducted in various situations including in a laboratory, natural work environment or the home for example.

Generally speaking, there are three major evaluation approaches: field studies, analysis evaluation and usability testing (Preece and Rogers, 2007). Each of the method is based on a distinct set of values and assumptions. In addition, different evaluation methods employ different technologies associated with data gathering and analysis.

The first evaluation method is a field study. The feature of field studies is that they are made in a natural setting with the aim of understanding what people do naturally and how products mediate their activities. The field study is more suitable to following purposes:

- To help identify opportunities for new technologies or establish the requirements for design.
- To facilitate the introduction of technology.
- To deploy and/or evaluate existing technology in a new context.
- The second evaluation method is predictive analytical evaluation. This method is composed of two categories: inspections, which include heuristic evaluation and walkthroughs, and theoretically based models, which are used to predict user performance (Rogers et al., 2011). A key feature of these methods is that users need not be involved. The idea is that knowledge of regular users forms the basis of established design guidelines which are used to identify usability problems.

The third method is usability testing, which is primarily aimed at evaluating how a user responds to the product (Rubin and Chisnell, 2008). Usability testing was the dominant approach in the 1980s (Rubin and Chisnell, 2008). Until now, usability testing is still a central aspect of interaction design. In particular, the user testing is the most suitable for testing preliminary prototype (Rogers et al., 2011 p. 430).

Generally speaking, usability testing involves users to examine typical users' performance and experience on typical tasks (Preece and Rogers, 2007). For example, the evaluation is usually focused on noting the number and kinds of errors that the users make and recording the time that it takes them to complete specific tasks (Bevan). In addition, the defining characteristic of usability testing is that the test environment and the format of the testing is controlled by the evaluator (Bailey et al. 2009). This kind of usability testing usually takes place in a laboratory or in a semi-laboratory situation where the user is not disturbed by other people or the surrounding environment such as talking to colleagues, checking email, or doing any of the others things that are (arguably) not related to the task. For this form of usability testing it is particularly important to develop a product prototype that can be efficiently used by diverse users. According to the ISO 9241 Usability Standard there are three base criteria for usability tests: completion effectiveness, efficiency and user satisfaction (Bevan, 1999).

After considering the above different evaluation methods we argue that usability testing is the most suitable method for this research for the following reasons:

- Usability testing focuses on testing a concrete interactive artefact to explore any problems users' have in using the artefact.
- Usability testing is a major evaluation standard in human computer interaction design.
- Usability testing provides comprehensive data about users' interaction and experiences using a particular interactive artefact.

The purpose of this usability testing is to test whether the new interaction design pattern can improve the quality of interaction between a user and interactive artefact. It is essential to have a prototype, which reflects how an interactive product built using this pattern performs and responds to the user's activity. We argue that a user's personal experience of interacting with a new product is best observed through evaluating a prototype.

A second benefit of using usability testing is that it is a standard method, which is widely used in industry areas.

Third, usability testing can provide an effective way to gather comprehensive data on a user's interaction and interaction experience. In this research, we create multiple prototypes of systems which implement the ILDP in order to answer two key questions:

- Is the DSIL useful for the user to operate the interactive artefact from the user's perspective?
- Does the user have enjoyable interaction experiences when interacting with the interactive artefact?

In the next section, we will illustrate how we evaluate a particular interactive system built using the new design pattern by using the usability testing method.

3.2 Research method

3.2.1 Usability testing

In this research, usability testing is employed to evaluate whether the interactive product we develop is usable and fitting for the intended users. The usability testing is conducted in two steps:

1. Identify tasks.
2. Build and evaluate prototypes

Initially, we decided to build a drawing system to evaluate our proposed design pattern. We then chose three simple drawing tasks to examine. The tasks are:

- Draw a simple shape by using different drawing tools (pencil, pen, oil pen and crayon).
- Draw multiple lines with different size.

- Draw different shapes (circle, rectangle and square) with different color.

The second step of the usability testing is to create prototype of the drawing system. We first created a paper prototype in order to assess the broad applicability of the design pattern, gather user feedback and to establish whether we were on the right track. Based on the evaluation of the paper prototype we then developed a hi-fi prototype – an actual working drawing system which implemented the proposed design pattern.

The paper prototype is a low-fidelity prototype. The reason we employ paper prototype is that it is very simple, cheap and quick to produce. That means that we can easily and quickly to realize a design concept and idea. On top of that, we can explore alternative designs and ideas by modifying the paper prototype (Rogers et al., 2011). And the results of testing paper prototype can be used to build a Hi-Fi prototype.

The aim of Hi-Fi prototype study is to investigate users' experiences using an actual drawing system. Hi-fidelity prototyping uses materials that expect to be in the final product and produces a prototype that looks much more like the final thing. In this case, the Hi-Fi prototype of the drawing system is built by using a variety of languages, including programming languages and interface design pattern language. For more details see Chapter 6.

For the user experience study we asked participants to complete the predefined drawing tasks using two different types of drawing systems: Photoshop CS and the Hi-Fi prototype. The main aims for the users experience study were to:

1. Compare users' interaction experience with the two drawing systems.
2. Investigate the participants' interaction experiences by asking a user to describe his/her experience of using the two drawing systems.

3. Analyze the user's experiences of the interaction at different levels: physical level, cognitive level and reflective level.

3.3 Criteria of usability testing

Harker defines usability as the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use (Harker, 1995). In industry area, a well-known deification of the usability is ISO 9241 which is composed of three parts: effectiveness, efficiency and satisfaction (Abran et al., 2003).

According to Nilsson's definition of usability, usability is composed of five quality components (Nielsen, 1994):

1. Learnability: How easy is it for users to accomplish basic tasks the first time they encounter the design?
2. Efficiency: Once users have learned the design, how quickly can they perform tasks?
3. Memorability: When users return to the design after a period of not using it, how easily can they re-establish proficiency?
4. Errors: How many errors do users make, how severe are these errors, and how easily can they recover from the errors?
5. Satisfaction: How pleasant is it to use the design?

From the above definitions we can see that there are several criteria for usability, but that they mainly relate to two key aspects:

- Usability of product including efficiency, learnability effectiveness; and
- User experience of using the product in a particular situation.

The results of user studies should provide measures of comprehensive base criteria of useability including completion effectiveness, efficiency related to the artefact usability and user experiences. Usability testing is typically carried out in laboratory settings. We designed multiple questions built upon the criteria of usability to evaluate the usability of IDPL in paper prototype study. The questions can be found in section 3.4.2 and appendix B.

For user experience study, the participants are required to complete a set of drawing tasks by using two digital drawing systems (Photoshop CS2 and a prototype that is built upon ILDP). User satisfaction is assessed based on the users' feedback on their experience with the systems such as satisfaction, fun and/or frustration by using a well-designed questionnaire. The questionnaire that is used in user experience study is listed in section 3.4.3 and appendix C.

3.4 Data gathering methods

In order to get comprehensive feedback from users on the prototypes, a range of data gathering methods were used. These are outlined here.

3.4.1 Data gathering: Observation

Observation helps the researcher learn what is taken for granted in a situation and to discover what is going on by watching and listening (Richards, 2009). As Rogers et al point out, observation is a useful data gathering technique at any stage during product development (Rogers et al., 2011). For instance, in the early stage of designing interactive products, observation techniques can help the designer to identify needs leading to building a new type of prototype, and at the second stage, it helps the designer to evaluate prototypes by investigating how well the prototype supports the task and goals.

In this research, a significant challenge for the evaluator was how to observe without disturbing the people being observed. In this research, in order to minimize the disturbing with the user, we set up an experiment environment for our usability testing in order to prevent participants being disturbed by other people or other things. That means during the testing we did not talk to the observers and they are required to concentrate on completing the tasks unless they have questions about the tasks.

3.4.1.1 User testing performance

Measuring user performance to compare two or more designs is a major object of the user testing. Normally, the user testing is conducted in a controlled setting and involves typical users performing typical well-defined tasks.

For this study, we produced a set of basic drawing tasks to test user's performance. Both the paper prototype and the hi-fi prototype were tested. In user experience study, the user is required to complete these tasks using two different drawing systems: Photoshop CS2 and the prototype based on the Interaction Language Design Pattern.

Criteria for successfully completing the predefined tasks are listed for evaluating the user performance. The three drawing tasks are described in section 3.2.1. After collecting data, the participants' performance can be analysed such as the time taken to complete a task, the number of errors made, and the navigation path through the product are recorded. Additionally the time taken to complete the tasks was automatically recorded and calculated from the video and interaction log data. Combined with data from interviews and questionnaires, it can help us to understand the participant's performance comprehensively.

3.4.2 Data gathering: Interviews

Interviews can be thought of as conversations with a purpose (Kahn and Cannell, 1957). How like an ordinary conversation the interview is depends on the questions that need to be answered and the interview method that has been used. There are four main types of interview: open-ended or unstructured, structured, semi-structured, and group interview (Fontana and Frey, 1994). These types of interviews relate to the level of control by the interviewer to represent predefined questions.

The most appropriate approach for interviewing depends on the evaluation goals, the questions aim to address, and the paradigm adopted. For example, if the goal is to gain first impressions about how users react to a new design idea, such as an interface, then an informal, open-ended interview is often the best approach. But if the goal is to get feedback about a particular design feature, such as the layout of new interaction model, then a structured interview or questionnaire is the often better. The reason is because the goals and questions are more specific in the latter case.

In paper prototype study, we used a semi-structured interview to collect data. Semi-structured interviews combine features of structured and open-end interview and use both closed and open questions (Rogers et al., 2011). In other words, through semi-structured interviews the interviewer can start with pre-planned questions and then probe the interviewee to give ideas for some open questions. The key feature of semi-structured interviews is that there is no standard answer or predefined responses. Interviewees are given more time and chance to say what they want to indicate. The features of semi-structured interview allows us to ask the users some questions about their responses to using a paper prototype of the DSIL and their personal impressions of the new interaction design – drawing system to have a deeper understanding of the usability of the domain specific interaction language.

To do this, the interview questions are composed of a mix of closed and open questions.

The closed questions focus on getting feedback about particular design goals or features. In this study the questions were:

- Do you feel more creative while you were drawing?
- Do you feel free or easy to achieve what you want to do with this drawing system? In other words, does understand what you want it to do.
- Do you like this way of drawing?

The open questions relate to exploring the users' individual experiences how well the product supports them to complete a particular task and what the other supports are needed. The questions are listed as following:

- What do you think about the drawing system?
- How did you use the drawing system to complete the tasks?
- Do you find any problem or difficulty in using the drawing system? What were they?
- What were the main things you liked about this way of drawing?
- What were the main things you disliked about this way of drawing?
- Overall did you enjoy the experience of drawing with this system?
- Do you have any other comments?

3.4.3 Questionnaires

Questionnaires are a well-established technique for collecting demographic data and user opinions. Sharp et.al mention that clearly worded questions are particularly important to ensure that the researchers can collect the data from the participants effectively. So a well -designed questionnaire is good at getting answers to specific questions from a group of people (Preece and Rogers, 2007). Comparing the two designs is a major object of doing the questionnaire.

The questionnaires focus on comparing the participants' interaction experiences using the drawing systems, and are designed based on the tasks that we created. In order to ensure the questions are clearly worded and appropriate, all the questions are built upon a well – known questionnaire format: the Questionnaire for User Interaction Satisfaction (QUIS). The Questionnaire for User Interaction Satisfaction (QUIS) it was developed by the University of Maryland Human-Computer Interaction Laboratory and is one the most widely used questionnaires for evaluating interfaces (Chin et al., 1988, Shneiderman et al., 2013). The advantage of this questionnaire is that it has gone through many cycles of refinement and has been used for many evaluation studies. The questionnaire consists of 12 parts that is displayed as following:

- System experience (i.e. time spent on this system)
- Past experience (i.e. experience with other system)
- Overall user reactions
- Interface design
- Terminology and system information
- Learning (i.e. to operate the system)
- System capabilities (i.e. the time it takes to perform operations)
- Technical manuals and online help
- Online tutorials
- Multimedia
- Teleconferencing
- Software installation

Based on the basic format of the user interaction satisfaction questionnaire, we produced a specific user satisfaction questionnaire. In order to explore the participant's interaction experience, three different groups of questions are created based on the level of the participant's interaction, including visceral level, behavior level and affective level (Norman and Draper, 1986).

The questionnaire is shown below, and also in appendix B.

Resulting questions
1.1 I can effectively make drawing
1.2 It was easy to learn how to make a drawing

1.3 I found it difficult to find the tools I needed
1.4 The interface of this system is pleasant
1.5 The system has all the functions and capabilities I expect it to have
2.1 It was uncomfortable making a drawing
2.2 I felt more creative while drawing
2.3 It was easy to complete the tasks
2.4 It took too long to complete the task
2.5 I found the interaction attractive
2.6 I like this way of drawing
3.1 Using the program was enjoyable.
3.2 It was difficult to express myself using this program
3.3 Drawing with the program felt 'natural'
3.4 What is your overall impression of the value of this program?
3.5 Is This program is novel

Table 3. 1 User satisfaction questionnaire for human computer interaction

Different types of questions require different types of responses. Two types of rating scales are employed in the questionnaire: Likert scale and semantic differential scale. Likert scales are used for measuring opinions, attitudes, and beliefs, and consequently they are widely used for evaluating user satisfaction with productions (Allen and Seaman, 2007). For example, gathering users' opinions about their experiences with interactive language interaction could be evaluated with a Likert scale using a range of numbers (1 to 5) or with words (strongly disagree, agree, neutral, agree, and strongly agree).

Semantic differential scales are used to explore a range of bipolar attitudes about a particular artefact. Each pair of attitudes is represented as a pair of adjectives (Bradley and Lang, 1994). In our cases, the participants are asked to choose a number of positions between two extremes to indicate agreement with the poles. For example, to evaluate an interaction of operating a drawing system to draw a picture the participants are asked to choose a number (1 to 5) between artificial and natural.

The purpose of using these two rating scale is to elicit a range of responses to a question that can be compared across respondents. They are good for getting people to make judgment about a particular interactive artifact, e.g. how easy, how effective, how enjoyable etc., and therefore are important for this user experience study (Pappas, 2011).

3.5 Data analysis methods

The purpose of data analysis is to develop an understanding or interpretation that answers the basic question being asked. B. Kaplan and J.A. Maxwell give five purposes for using qualitative methods in evaluating computer information systems (Kaplan and Maxwell, 2005):

1. Understanding how a system's users perceive and evaluate that system and what meanings the system has for them.
2. Understanding the influence of social and organizational context on systems use.
3. Investigating causal processes.
4. Providing formative evaluation that is aimed at improving a program under development, rather than assessing an existing one.
5. Increasing the utilization of evaluation results.

As we have described, the data gathering method employed in this research is to set up appropriate user tests to assess whether the product we develop is usable and fitting for the intended user population to achieve their objectives. We set up two different prototype studies of a drawing system to clarify our design method. One is paper prototype study and the other one is Hi-Fi prototype study.

Accordingly, we employ different data analysis techniques to evaluate the data from each of the two prototype studies. Normally, there are four basic techniques of qualitative data analysis: coding, analytical memos, and contextual and narrative analysis (Kaplan and Maxwell, 2005). These methods help us to identify themes, develop categories and explore similarities and differences in the data, and relationships among them.

3.5.1 Coding

To analyses the data that is gathered primarily from the first study through observing, interviewing and asking participants questions. The produces qualitative data demonstrates whether the DSIL can support the user to achieve his/her goal effectively. The analysis method we use is coding and analytical memos. We use video to capture everything that the users did during the usability testing of using paper prototype. The coding is created by the users when they doing the paper prototype study focuses on evaluating usability of the domain specific interaction language, which has been developed from the ISO (ISO 9241 Usability Standard) base criteria of task effectiveness and efficiency. For example, we code the participants' performance when they were required to complete few specific tasks by using the DSIL refer to its usability, effectiveness and efficiency in a specified context of use.

3.5.2 Analytical memos

The other analysis method is analytical memos. We asked the participants different questions about their responses of using the paper prototype. Through these questions we are able to have a deeper understanding of what the participants think about of using the interactive artefact come through the domain specific interaction language. As we mentioned in section 3.4.2, there are two types of questions are employed in the interview: close question and open question. The close questions focus on getting feedback about a particular design feature, such as the way of interaction that the interactive artefact provided. The open question related to exploring the users' individual experiences how well a product supports them to complete a particular task and what other supports are needed. At a result, the general questions (open) and specific questions (close) that contribute to bring comprehensive feedback for the evaluation goal are asked.

3.5.3 Contextual and narrative analysis

For the Hi-Fi prototype study, we use contextual and narrative analysis method to analysis the data that is generated from the observation and user satisfaction questionnaire. Through the Hi-Fi prototype study, we category and analysis the results of user testing to evaluate whether the participants enjoyed using the DSIL to improve their interaction experience by constructing personalized interaction.

In the Hi-Fi prototype study, we observe user using the Hi-Fi prototype to exam whether the developing product meets users' needs. We use video to capture everything that the users did during the usability testing including keystrokes, mouse clicks, and other interactions. Through the observation data, we can clarify that and analyse what users do and how long they spend on completing different tasks. It also provides insights into users' affective reactions that related to the users' experiences as satisfaction and frustration. Moreover, the user satisfaction questionnaire is used to clarify and deepen understanding of the users' experiences.

We designed a specific questionnaire to evaluate users' satisfaction with some specific features of the Hi-Fi prototype. As we mentioned before, the questionnaire that we are created in different levels: visceral, behaviour and affective (Norman, 2002). And then, we compare outcomes of users testing performance on manipulating two digital drawing systems and asking the users about their opinions based on their interaction experience through a user satisfaction questionnaire. The first drawing system is a well-known drawing system-Photoshop CS2; the second one is the Hi-Fi prototype of drawing system. The produces qualitative data demonstrates how the different participants make sense of the interactive artifact.

3.6 Research context and environment

To construct a suitable environment for usability testing we need to solve practical issues including designing typical tasks, selecting target users, prepare the testing conditions, setting up a variety of tests and dealing with the ethical issues.

The tasks for the usability testing are designed to compare users' experiences creating a drawing using the two prototypes: the paper prototype and the Hi-Fi prototype. There are three drawing tasks:

- Draw a simple shape by using different drawing tools (pencil, pen, oil pen and crayon).
- Draw multiple lines with different size.
- Draw different shapes (circle, rectangle and square) with different color.

Next, we need to select suitable users to evaluate the systems - people that somehow represent those that the product is designed for. For example, some products are targeted at specific types of users like seniors, children, novices, or experienced people. In our case, the product is a digital drawing system, so the specific user audiences are computer users who use drawing systems.

In addition, users' prior experience with a particular classes of product are different so selecting a range of users with different backgrounds is important. For example, a group of people who are using the web for the first time are likely to express different opinions to another group with five years of web experience.

To gather a range of views from different perspectives, we chose two different groups of people: novice users and experienced users with different backgrounds. These two groups of users participated in both prototype studies.

To achieve a gender balance, in the paper prototype study we employed an equal number of males and females aged is between 21 and 38, including three males and three females from different disciplines and living areas. In the Hi-Fi prototype study, we recruited 30 representative users from, including 16 males and 14 females. All of them have different study backgrounds and nationalities.

Finally, it is necessary to prepare the test conditions and set up user tests for paper prototype testing and Hi-Fi prototype testing in a lab situation. For both of the prototype studies, it requires the testing environment to be controlled to prevent unwanted disturbances and noise that may distort the result. As the product we are going to test will mainly be used in an office situation and environment we establish a simulated working environment by setting up a usability testing laboratory to carry out the above user tests. The facilities that have been used in laboratory are different for each of the two user tests. For the first test, after the user experimented with the paper prototype, we asked them questions to capture their opinions through a semi-structured interview. For the hi-fi prototype evaluation, we set up a video camera in the laboratory to record all the data that the user generated during operating the provided Hi-Fi prototype of the interaction language and the Photoshop CS2. After the user testing, the users were asked to complete the questionnaire.

The other important issue is research ethics. When we collect data during an evaluation it is necessary to consider ethical issues. We followed the ethics procedures in place at the University of Technology, Sydney. The concert form is included in appendix A.

3.7 Chapter summary

In this chapter we presented the research methodology that has been used to evaluate the interaction language design method.

Firstly, a methodological review is carried out following an evaluation framework - DECIDE it contains following several factors: Determine the goals of the prototype studies; Explore the research questions; Choose the appropriate evaluation paradigm and techniques; Identify the practical issues; Decide how to deal with the ethical issues; Evaluate, interpret and present the data. We demonstrate different key evaluation paradigms, including usability testing, analysis evolution and field studies (Section3.1). We briefly describe each of them and identify similarities and differences. The usability testing method selected for this research context is introduced (section 3.2). In particular, the methods and techniques for data gathering (3.4) and data analysis are described (3.5).

Thirdly, we described how we designed two different prototype studies of ILDP to carry out the usability testing to evaluate ILDP for two different purposes: usability study and user experience study (3.5).

Finally the practical context of the usability tests was described (3.6). The outcomes of the two prototype studies are provided in chapter 5 and 6. The materials that have been used in the usability and user experience studies are in appendix B and C.

Chapter Four: Interaction Language Design Pattern (ILDP)

Chapter four presents the Interaction Language Design Pattern (ILDP)[6]. In the first section, following principles of human communication and the fundamental role of language, we propose a design framework for building an interaction language to enable personalized human computer interaction. There are two characteristics of using this interaction language: First, the interaction is built upon a convergence of both non-linguistic and linguistic behaviour. Second, the interaction language provides the user with a suitable way to customize their interaction with the computer.

The rest of chapter contains two sections: The first describes the definition and structure of the interaction language. The second develops a user-oriented interaction language in three steps: interaction language production; interaction language perception and interaction language application.

4.1 Introduction

This chapter presents the Interaction Language Design Pattern (ILDP), proposed to address a number of identified problems in designing personalized interaction. In order to establish personalized interaction between a variety of users and the computer, we identify the following key features of human communication and the role of interaction language:

- Interaction is a cooperative process based on a participant's rational behaviors through his/her contact, perception, understanding, and attitudinal reactions (Allwood, 2007).

[6] The interaction language design pattern is a design pattern for producing personalized domain specific interactive systems. It specifies how to design a domain specific interaction language and linguistic interaction customization form.

- Human communication is a complicated semantic system, built upon a clear combination of both linguistic and non-linguistic behavior (Press, 1996).

- Language plays diverse roles with various functionalities to support different communicative activities based on the conversational participant's needs.

From the above perspectives, the main tasks for the interaction designer should not only be concentrated on setting measurable usability specifications and evaluating interaction designs with a variety of users, but, more importantly, to secure a rich experience for the end user (Wright et al., 2008). Human computer interaction development does not end with its first release, with its original setting interface and interaction model, but is changed by the end user and further evolved on the basis of the user's feedback (Dearden, 2006).

We claim that constructing personalized interaction experiences requires a personalized form of interaction. To achieve this, the user needs not only a useful interactive artefact, but also to be able to customize the interactive product based on his/her perspectives. We will therefore explore effective language-like interaction for human computer interaction in order to establish a common ground between users and the computer (Monk, 2009).

According to the perspective of language/action theory:

> *"Human beings are fundamentally linguistic beings: action happens in language in a world constituted through language."* (Flores et al., 1988 p. 156).

Some researchers go so far as to suggest that it is necessary for interaction designers to become linguists of behavior, storytellers, and poets who are striving to create conversations that fit within the context in which that conversation unfolds (Winograd, 1997, McCarthy and Wright, 2004). The research of interactional linguistic is to explore how languages are shaped by interaction and how interactional practices are molded through specific languages (Couper-Kuhlen and Selting, 2001). From this perspective, it treats speech as an ongoing or emergent product in a social semiotic event and language as providing one set of resources for the accomplishment of goals or tasks within this event. The resources which language provides are used methodically, it is assumed; with them speakers engage in practices, routine and recognizable ways of carrying our sequentially situated actions and activities (Schegloff, 1997). “Interaction itself is expected to be shaped by language and, on a universal level, to embody subtly different interactional practices depending on language type” (Couper-Kuhlen and Selting, 2001). Moreover,

“Linguistic productions – since they take shape in interaction- can no longer be conceptualized as the product of a single speaker… Thus, linguistic structure are both emergent in interaction and heavily context- sensitive, in that their use reelects- and may even contribute to creating- conversational structure.” (Couper-Kuhlen and Selting, 2001)

We believe that a user-oriented interaction language can construct common ground between a variety of users and the computer. Such a user-oriented Interaction language would support interaction between human and computer that allows the participants to easily perform their activity. The user-oriented interaction language is created to organize concrete interaction artefacts through the interface and provide interaction models to convey the user’s intended meaning.

In this chapter, we propose an Interaction Language Design Pattern (ILDP) to build a user-oriented interaction language that leads to personalized interaction. It is intended to optimize the user’s interactions with the computer by supporting and extending the user’s activities in a useful, effective and reasoned way. The Interaction Language Design Pattern (ILDP) aims to build the interaction language in two purposes:

First, the ILDP provides interaction designers with a linguistic design system to support designers in the creation of an interactive artefact that is meaningful to the target user. The linguistic design system helps designers to transfer their design concepts into a different form that results in the building of a particular interactive artefact, composed of diverse interfaces and reasoned interactions.

Second, the ILDP provides a way for the user to adjust the interaction based on his/her concepts using natural language. In other words, the interaction is personalized to express the user's 'interactive semantics' [7]through linguistic interaction. Eventually, the user's interactive semantics (mental model) are used to build a particular interactive artefact with personalized interfaces and interactions.

As a result, a user-oriented interaction language for a particular human computer interaction in a specified domain will be constructed. Generally speaking, the user-oriented interaction language is defined by its vocabulary (4.4.1), syntax (4.4.2), and resulting interactive semantics (4.4.3) in a specific domain. The interaction vocabulary is a basic component of interaction that is generated from domain knowledge. The interaction syntax is a structure that designers or users use to construct an interactive artefact to arrange interaction and model of interaction. Typically, designers define a special interaction structure to realize a specific interaction concept by building an interactive artefact. The designer's interaction design concept is about how to build an interactive system according to their understanding of the potential user's needs. That is, the designer's understanding is 'second-order' (Krippendorff, 2005).

In this chapter, we aim to provide a theoretical framework of ILDP to create personalized human computer interaction. The ILDP is proposed to improve the quality of human computer interaction by reforming the interaction design based on the end-user's individual context. Thus, all participants of

[7] Interactive semantic: the meaning a user discerns from the look, behaviour and affordances of the interactive system. The user's interactive semantic is different to the designer's interactive semantic, the designer's interactive semantic for building a particular interactive product based on a second order understanding (Krippendorff, 2005)

an interaction, especially the end user, are able to contribute in constructing the interaction from different perspectives. This interaction design method emphases the fact that the end-user cannot be designed and defined by the designers (Cho and Yoon, 2013). We believe that the user is the only one who can optimize his/her interaction. So, the ILDP provides users with a DSIL to support personalized interaction with an interactive artefact. The personalized human computer interaction is able to integrate the user's individual characters and the performance of the interactive artefact.

4.2 Background and motivation

In the chapter two, we noted that effective human communication is built upon participants' individual perspective and activity. In other words, human communication is a personalized activity. In addition, we argued that language plays a critical role in constructing shared understanding. We recognize that humans use language to coordinate their perceptions and actions in the context of living with one another as communicative participants (Clark, 1996). By doing so, we co-construct our conceptions of each other and of each other's artefacts (Eggins, 2004). Therefore, language is the way in which, as humans, we experience what we call reality.

Human computer interaction is a computer-mediated communication, belonging to social communication (Zhuge, 2010). It differs from traditional human communication using sounds and letters because digital media does not have physical form and it can be easily changed to any number of different representing formats (Martinec and Van Leeuwen, 2009). The multiplicity of digital media forms a major resource for human computer communication. Martinec and Van Leeuwen assert that, essentially, multiple digital media are, or should be, like language, and are resources for communication. Through them, the user can, or should be able to, associate with meanings (Martinec and Van Leeuwen, 2009). From this perspective, we argue that all components of human computer interaction, such as text, image, sound and time, space, as well as human activity, can be organized under a conversational system according to a linguistic structure.

As a result, it is possible to compose a more suitable interaction to suit a variety of users. In other words, a well-defined linguistic structure can help designers semantically map out human computer interaction based on the designers' interaction design concepts. The design process can be understood as a process of conveying pre-defined messages between a particular interactive artefact and its users. According to human communication models, the transformation process comprises two phases: in the first phase, the original message-design concepts are created by the interaction designers when they produce an actual interactive artefact with specific interface layouts and interaction patterns. In the second phase, the users are supposed to receive the meanings through the interface and the navigation system. The quality of interaction depends on the user's perspectives and the performance of the interactive artefact. Cooper (Cooper et al., 2012) points out that the closer the match between the designer's conceptual model and the user's mental model, the easier it is for the user to manipulate the product and understand the meaning of the interaction in order to effectively collaborate with the computer.

At the same time, we realize that a user-specific human computer interaction model that completely matches the user's mental model and the behavior of the computer does not exist. The main reason is that we lack an interaction design system that truly supports user specific interaction. Langdon et al write that "there has been no effective way to build human product interaction that can adapt to end users' individual mental models" (Langdon et al., 2012). In other words, the meanings of interactions are different to different end users.

Many researchers and practitioners agree that in current human computer interaction, designers or producers of interaction have more power than the end users to control the interactive artefact (Cho and Yoon, 2013). Because that the form and meaning of interaction is shaped by the interaction designer, using different design approaches such as system-centered design method and human-centered design method (2.3). Additionally, we identified the problems of using existing design approaches to construct human computer interaction in (2.6.2). To this end, we claim that, to solve existing problems, the users need to be offered a powerful communicational tool. They need a language through which to communicate with the interactive artefact just like programming languages for the programmer and interface design pattern languages for the interface designers. Existing interaction languages used in human computer interaction, including a variety of programming languages, interaction design pattern languages and interface design pattern languages, are not user-oriented. We also know that the other important human computer interaction language, natural language, is hard to be understood by computer appropriately. A good example is using voice to control a system in the iPhone.

Based on the above perspectives, we believe that it is necessary to construct a user-oriented interaction language on top of creating a concrete interactive artefact. According to research on human natural communication, humans rarely use fixed associations between content or meaning and its physical representation. Instead, people use language to encode content or meaning into a form appropriate for the situation and purpose of their communication (Polovina and Pearson, 2006). Thus, the interactions between human and computer also need to be carried out with different representing forms, all of which attempt to ensure that the content or meaning is communicated between the participants in the most accurate and efficient manner.

Language is a crucial method with which to realize effective human and computer interaction. This is because it allows the user to form a personalized common ground of interaction with the computer through personalized interaction. By using a domain specific interaction language, users can generate customized interactions and are able to form a mutual interaction with the computer. As a result, human computer interactions not only support end users in completing tasks in a pre-determined way, but also let users tailor the interaction to better adapt to their needs. Therefore, the DSIL can improve the quality of human computer interaction in three ways:

- Letting a user become to an active participant in building the interactive system
- Establishing a common ground between the user and the computer

Allowing the user to solve communicational problems by personalizing the interaction

First, the DSIL supports the user's active participation in building an interactive artefact. That means the language supports the human computer interaction to be a cooperative process. For that reason, the interaction language focuses on dealing with two communicational factors that have been ignored in existing design methods: the users' individual interaction responsiveness and the role of the users. Users' individual experiences plays an important role that affects the quality of interaction through their interactive activity (Allwood, 1976). The other factor is the role of the users that emphases a fact that users need to become the owners of their interaction in order to fully control the interaction (Cho and Yoon, 2013). This point of view has been adopted and applied in many different design areas such as participatory design (Schuler and Namioka, 1993, Bødker et al., 2000), experience design (McCarthy and Wright, 2004, Wright et al., 2008), End User Development (Fischer and Giaccardi, 2006), User Centered Design (Norman and Draper, 1986) and inclusive design (Langdon et al., 2012), but none of them provide a comprehensive design system to produce personalized interaction focusing on the user's individual perspectives and experiences.

The second factor is that the DSIL is proposed to establish common ground between interaction participants. Clark points out that language is used to generate common ground by supporting its users to express their thinking and experience through their interactions (Clark, 1996). Similarly, Flores and Winograd claim that “human beings are fundamentally linguistic beings: action happens in language in a world constituted through language.”(Flores et al., 1988, Winograd, 1986). In details, interaction language can be used to build common ground using two approaches: semantic approach and direct approach. Semantic approach relies on linguistic behavior and direct approach is carried out by a particular interactive artefact. Correspondingly, a common ground is built upon both linguistic and non-linguistic behavior.

The third aspect is that the DSIL allows the user to personalize their interaction. An important benefit is that it helps the end user to solve diverse problems of computer-mediated communication in an individual context. The DSIL provides the end user with an interaction language with which to customize interaction. In this research, we create a text-based linguistic interaction model to carry out interaction language. Through this text-based interaction, the user can organize his/her activity based on their personal and social values. The users’ interaction is realized through the computer to build the resulting interfaces and interaction models.

Related research shows that participants are able to understand each other when they align their model of the situation in a conversation (Branigan et al., 2010). The textual interaction customization language keeps the users and the computer aligned at different levels: visceral, behavior and affective (Norman, 1988). To support personalized interaction, the interaction language needs to work on different levels, including the representation level (interface), behavior level (computer system) and communicational level (interaction). Consequently, a user-oriented interaction language can be formed by integrating diverse languages: computer programming languages, design pattern languages and human natural languages.

In summary, the DSIL provides a method for participants (human and computer) to have a mutual understanding through interaction, just as might happen between two people., The user can organize his/her interaction based on his/her own ability to use language. Therefore, the users will remain involved in developing the interaction according to their needs. As a result, it is possible to build a suitable tool for a specific user as they can now design and develop their own tools (Pane and Myers, 2006).

In the following section, we will illustrate the definition and structure of the DSIL and explain how to create a DSIL focusing on producing personalized human computer interaction.

4.3 Interaction language design pattern

In this writing, we concern ourselves with the meaning of interaction as determined by the participants' perception and altered by the participants' interaction. In other words, when the participants of interaction change, the network of object or points in the classification space will change, along with the meaning of the interaction. For example, in HCI, designers employ programming languages and pattern languages to compose different the interaction concepts related to usability goals and experience goals.

We propose the interaction language design pattern to create a personalized interaction between human and computer (Figure 4. 1).

The Interaction Language Design Pattern contains two sub design tasks:

- Creating an interaction design concept to organize diverse interactive semantics in a specific domain
- Establishing a DSIL to carry out the interaction concepts

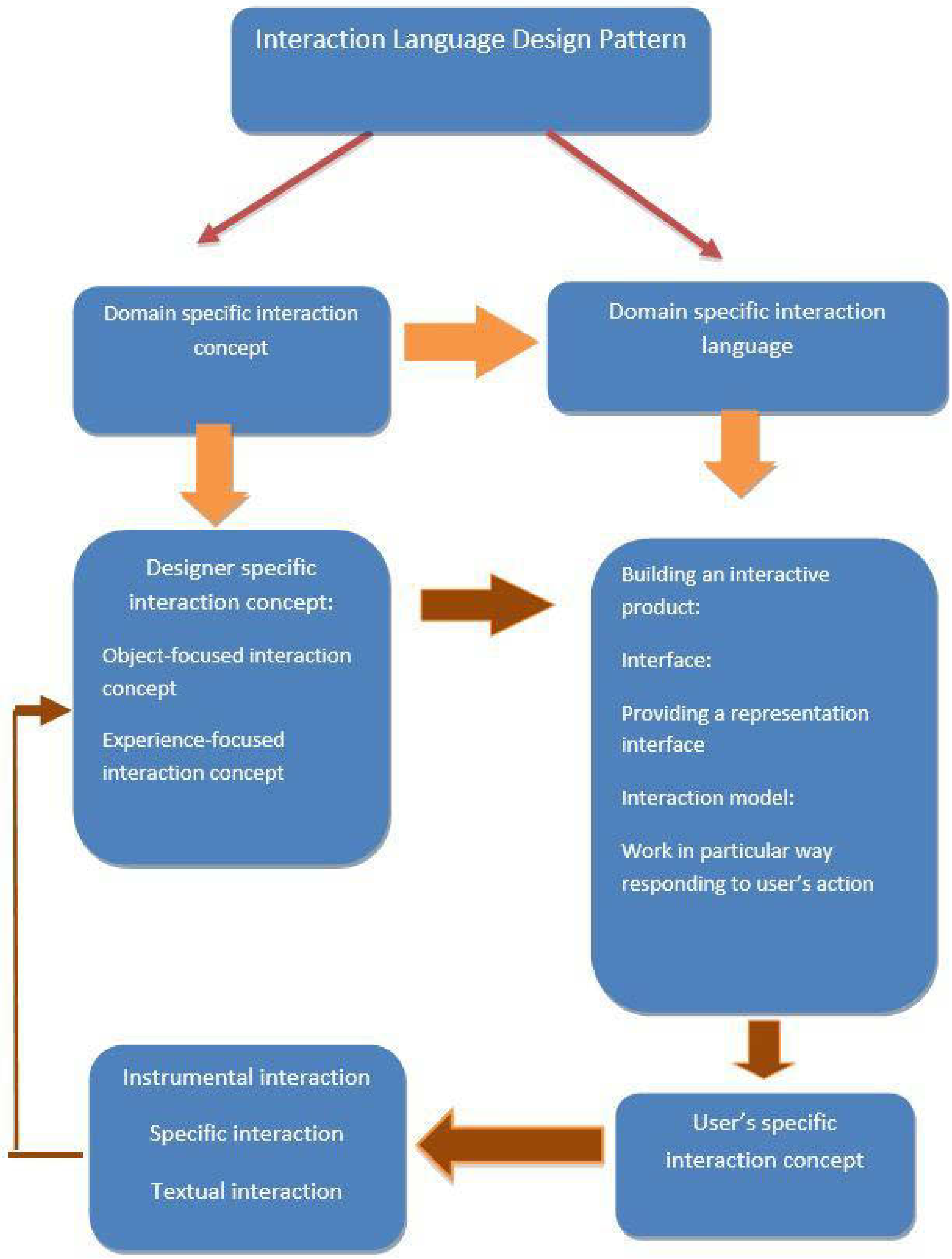


Figure 4. 1 Design architecture for building personalized interaction based on Interaction Language Design Pattern

4.3.1 Defining domain specific interaction concept

Before creating a concrete interactive artefact, we usually have an original definition statement - a concise, concrete declaration of the main interactive artefact, interaction context and intended user experience. It is the first step in order to build an understandable interaction between human and computer. On top of that, the purpose of the domain specific interaction design concept is to construct meanings of human computer interaction. As we mentioned before, the original meaning of interaction is built upon the designer's interaction concepts and depends on the designer's interactive semantics, combining diverse components of interaction. These include properties of the product, interaction context and user's characteristics.

The first task is to set up interaction design concepts by building an interactive artefact with a particular interface and interaction model. There are two relevant interaction concepts here. One is the designers' interaction concept, and the other is the user's interaction concept. For the designer's specific interaction design concept, we investigate two types of major interactive semantics. Table 4. 1 shows the above two forms of interaction concepts and corresponding examples of the interaction concepts.

Two ways to specify Interaction concept	
Interaction concept	Example of interaction concept
Designer specific interaction concepts:	• Object-focused interactive semantic A concept model of functionality and usability of a product. • Experience-focused interactive semantic An experience model of using a specific interactive artefact.
User specific interaction concept:	• A user specific perspectives and understanding of the interaction provided by a particular interactive

	artefact. This depends on his/her personal knowledge and reaction to a particular interactive product.

Table 4. 1 Tow ways to specific interaction concept

The designer's interaction design concepts are mainly composed of two types of interactive semantics: object-focused interactive semantics and experience-focused interactive semantics. Generally speaking, object-focused interactive semantics focus on a usability goal, while experience-focused interactive semantics aim to accomplish a desired experience.

For the user's interaction concept, that obviously depends on the participant's individual interaction experiences. The user's interaction concept is affected by the above two interactive semantics provided by the designer. The user's object-focused interactive semantics are shaped by how the user perceives a particular interactive product through using it. The user's experience-focused interactive semantics are built in respect of the user's individual recognition and emotional reflection. This is discussed in more detail in 4.3.2.3.

As a result, both the user's object-focused and experience-focused interactive semantics are determined by the user's different interaction semantic images. Important semantic images include: classification semantic image – reflecting the evolving classification spaces of a particular artefact; and object semantic image - reflecting the attributes and class of the object from multiple facets such as function, features and other relative aspects of the artefact. As a result, in many cases, the object-oriented interaction concept of usability of interactive artefact can be realized on a physical level carried out by instrumental interaction.

Alternatively, according to Systemic Functional Linguistics (SFL), communication using language is always patterned to simultaneously communicate three broad types of meaning: ideational; textual and interpersonal (Eggins, 2004). Based on the Systemic Functional Linguistic (SFL), we maintain that the interaction language should be able to generate an interaction concept ideationally, textually and interpersonally, all at the same. In other words, interactive semantics is divided into three components: ideational semantics, textual semantics and interpersonal semantics. Table 2 displays three types of interaction concept and meaning provided by using the interaction language (see Table 4. 2).

The way of expressing an interaction concept		
Ideational semantic	Textual semantic	Interpersonal semantic
Designers' and end users' abstract concepts. For example, an abstract concept about a website, or interactive system.	Describing an interactive product using natural language. For example, describing how an object is going to be used by a user in a particular interaction context using text.	A concrete interactive product, which is built upon the interaction concept.

Table 4. 2 Forms of expressing interaction language concept

Therefore, in terms of designing human computer interaction, a concrete interactive product, created by designers, presents a particular interaction concept by mapping out a variety of interaction vocabularies. The meaning of interaction is generated through the user forming a network of objects or points in the classification space.. This meaning will evolve through the user's ongoing interaction.

Different specific interaction semantics reflect user specific perspectives and needs of interaction on many levels, including physical and cognitive. The interactive semantics, such as a specific terminology a user might choose to define an interaction, depends on his/her personal knowledge and reaction to a particular interactive product. Therefore, the user's interaction concept addresses different interactive semantics and maps out how the user is going to use the interactive artefact. A user's specific interaction syntax realises specifies how that user can define a particular interaction context and tool properties.

4.3.2 Building DSIL

Typically, interaction concepts are derived from two main factors: one is the designer's intentions and domain knowledge used to construct the interaction that is mainly completed in the stage of designing an interaction. The other one is based on the end user's individual domain knowledge and requirements during their interactions with the computer.

On one hand, the DSIL needs to produce concrete interactive artefacts to convey the designer's interaction concepts. On the other hand, the user should be able to customize the interaction to solve a communicational problem by using the DSIL during the interaction.

Building a DSIL enables every participant to make inferences and predictions, understand and explain phenomena of an interaction, and decide what actions to perform and control their performance (Johnson-Laird, 1983). So our DSIL is proposed to solve communication problems in a particular interaction domain. Additionally, the interaction language will continually transfer the user's abstract interaction concepts to a concrete interactive artefact. This requires different types of interaction syntaxes to complete different interaction concepts (object-oriented interaction concept, experience-oriented interaction concept and user specified interaction concept).

The major components of interaction domain language include: interaction vocabulary, interaction syntax and interactive semantics. In details, the interaction vocabulary comprises the fundamental elements of composing human computer interaction, including text, picture, sound, movie, animation, human interaction behaviour, and so on. The interaction syntax is a form that designer uses to combine the variety of fundamental components of human computer interaction to convey a specific interaction concepts that comes from the designer's understanding and intention.

A particular DSIL is created in three steps:

- Identify vocabulary

- Create interactive artefact
- Realize user's interactive meaning

From the preceding discussion, we believe that a user-customizable DSIL can help to establish personalized human computer interaction. The purpose of creating a DSIL is to allow the participants of interaction to remain actively involved in the continuous development of their interaction on the representational level, functional level and emotional level. By doing so, the end users are able to tailor the interaction according to their perspective and experience in order to better adapt to their needs. In other words, the users are able to perform personalized interactions when they are allowed to develop their individual domain specific interaction language.

To achieve the goal, the DSIL needs satisfy the following requirements:

- Establishing a common ground between the user and the computer (system)
- Supporting the user to solve communicational problems by personalizing the interaction

For the first requirement, a common language between the users and the computer is needed. Like natural languages, such as English or Chinese, the DSIL has to provide a common ground for human computer interaction. That means, by using the language, the users and the computer can exchange specific concepts and meanings appropriately through the interaction. The DSIL generates unique meaning in a specific domain. Domain specific languages can be found in a variety of domains such as visual art, music and painting. Through using the languages of these domains practitioners can make sense of, communicate and understand domain-specific concepts or meaning. In the same way, designers using a domain specific language can create an interactive artefact that will help the end user to appropriately understand the intended meaning.

Alternatively, to support the user in solving interaction problems, the language has to support the user of the language to build an understandable, rational and meaningful interaction. Clark claims that language should be used by the user to establish a shared understanding and common ground (Clark, 1996). From a linguistic pragmatic perspective, a key concept in language pragmatics is the utterance (Austin, 1975, Fish, 1980). Dearden argues that

> *"To contrast with a sentence, an utterance is a specific instance of a specific speaker addressing a specific audience (that may be immediately present or otherwise). An utterance is always situated in a particular context. An utterance always includes some expectation of the response from the audience. Such discussions draw attention to the way in which the form and meaning of our utterances depend upon the context in which the utterance is made."* (Dearden, 2006p. 340)

The other key notion is speech *genre,* evoked by Russian literary theorist Mikhail Bakhtin (and colleagues). Genre relates to how the experience and history of speaker and audience impact on the form of utterances (Morris, 1994). Morson and Emerson point out that different speech genre lend themselves to expression of different facets of our experience; they are associated with particular ways of seeing the world, highlighting certain aspects whilst ignoring others (Morson and Emerson, 1990).

We believe that, in terms of human computer interaction, a DSIL can be employed to conduct effective communication between the user and an interactive artefact, centred on the two essential factors: customizable interactive artefact performance (utterance) and user-oriented interaction model (genre).

In terms of human computer interaction, utterance[8] refers to performance of the interactive artefact that is built upon the linguistic structure. Genre[4]

[3] Utterance: performance of a computer which is intended to convey some meaning to the user.

refers to the manner of the user's utilizing the interactive artefact that is, in turn, built upon the interaction language. Generally speaking, the utterance of the interaction language is the diverse range of interfaces, and the genre of the interaction language refers to different interaction models.

To summarise:

- the utterance of interactive product's interface builds upon the interaction language;
- Each element of the interface between the user and the computer can be understood and modified by the end user based on individual context and background.

This model of interaction enables users to customize the way they perform the necessary activities to achieve their tasks and application purposes. Supporting the user to develop such interaction language at different stages of the conversation, will enable successful communication (Branigan et al., 2010).

The essential idea to be explored here is that, as a part of interaction design, an interaction language is needed for building personalized interaction. In detail, the DSIL can be used to build personalized interaction in two stages:

In the first stage, the DSIL helps the designer to produce understandable utterances of interactive artefacts - for example, creating an interface by adapting to the user's individual ability and knowledge in a specific application domain based on linguistic interaction. In other words, each of the interface elements is labelled by an appropriate text and specific multimedia represents to convey the specific meaning to the end users.

---

[4] Genre: the mode of interaction used in human computer interaction e.g. un-linguistic interaction model (direct manipulation, gesture, touch) and linguistic interaction model (textual, voice), etc.

According to systemic linguistics (Halliday, 1994), language is patterned to simultaneously communicate three broad types of meaning: ideational, interpersonal and textual. Similarly, the DSIL represents a particular concept of human computer interaction in three forms:

- Ideational meaning – a mental model of interaction that depends on participants' interaction semantics
- Textual meaning - the content and meaning as described using languages such as English or Chinese. The natural language is a basic resource to construct an interaction domain specific language.
- Interpersonal meaning- a specific meaning that is conveyed between different peoples and it is represented through different media such as interactive artefact

Typically, to create human computer interaction the designers need have an original concept or idea to organize a variety of components of interaction and to build a concept model (Rogers et al., 2011). The main task in this stage is to define the major interaction concepts using a domain specific interaction language, based on specific interaction design objectives. The meaning of interaction is mainly generated from a combination of three key aspects: user (character of user), product (property of product) and interaction context. And then the concept will lead to the creation of a concrete interactive artefact, built by integrating various elements of interaction (e.g. text, image, sound, animation and interactivity) using a specific interaction design tool such as programming language and pattern language. As a result, there are different interfaces and specific interaction models that will be constructed. Figure 4. 2 illustrates how the designer's concepts have been transformed to build a particular interactive artefact through a linguistic translation process.

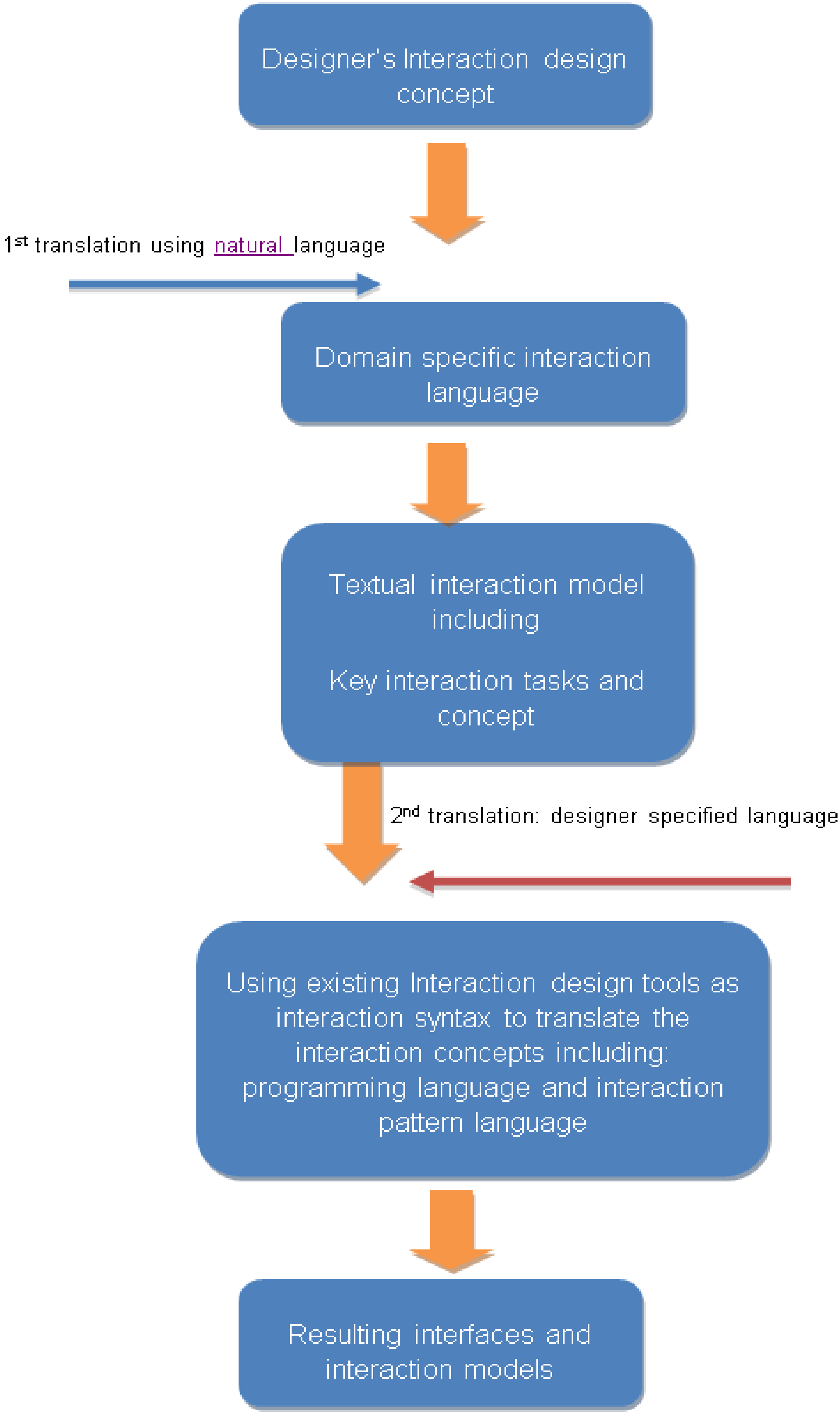


Figure 4. 2 The structure of linguistic translation from the designer's concept to concrete interactive artefact

In the second stage, the interactive artefact and its embodied meaning can be understood by the end users in two ways: on one side, the user is able to recognize the content and meaning of the interactive artefact in a semantic way such as reading the underlined text and word, ‘sign’. On the other hand, the user will make sense of the interactive artefact through multiple interfaces and interaction models. Consequently, the user will generate his/her understanding from experiencing the designed interfaces and interaction models according to his/her domain knowledge and individual perspective.

The DSIL enriches human computer communication by providing a powerful communicational tool between the users and the interactive artefact. For example, the user is able to co-develop the interface and interaction model by giving an appropriate word to represent the user’s thoughts and reactions. By taking advantage of language to make a linguistic interaction, we argue that the user will become to the owner of the interaction by being able to personalize the interaction based on his/her knowledge and characteristics. The end users’ linguistic interaction process is illustrated in Figure 4. 3.

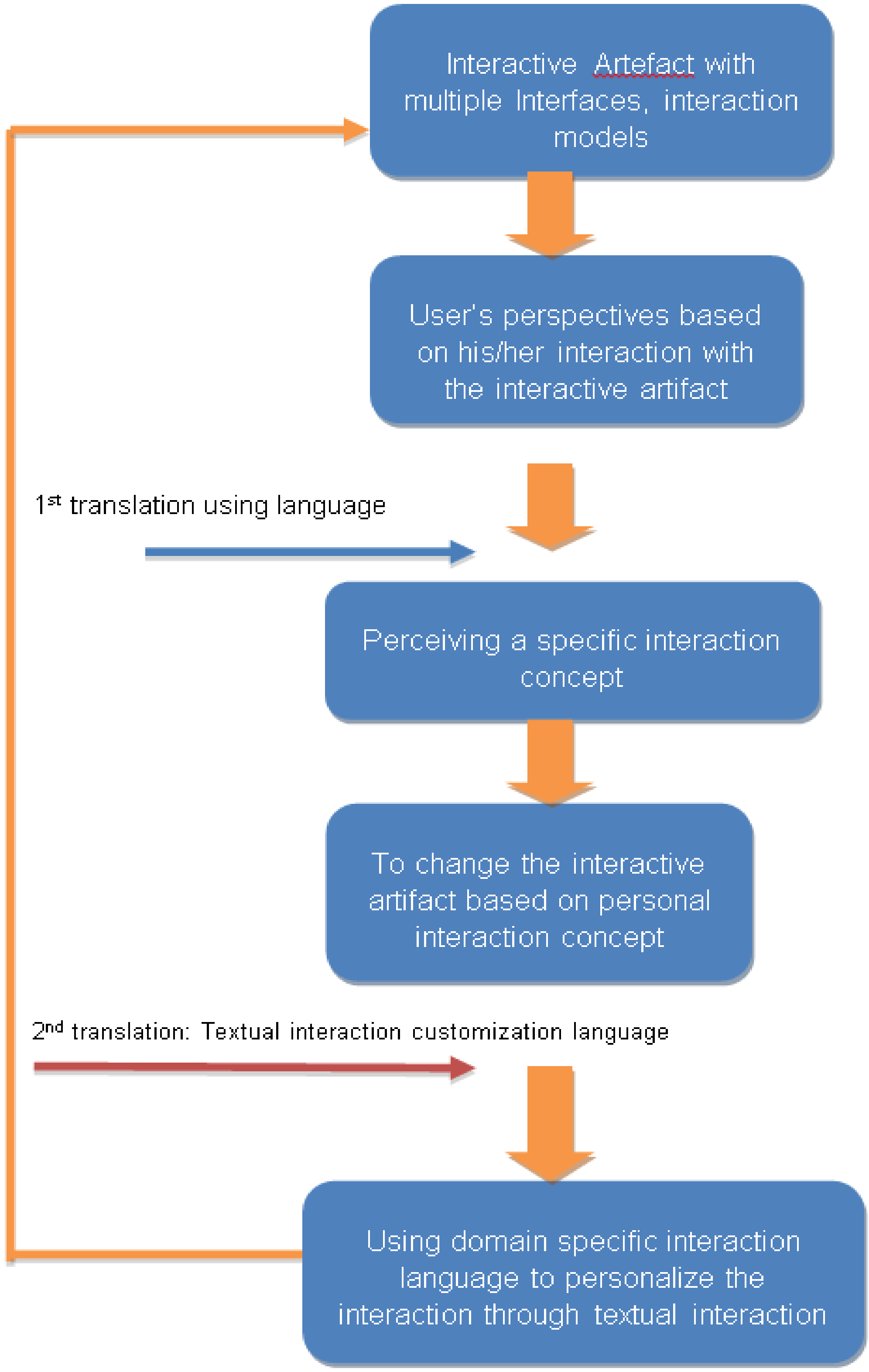


Figure 4. 3 The process of linguistic translation from the user's concept to the interactive artefact

Based on the above perspectives, we believe that building a user-oriented DSIL could become a central part of human computer interaction design work. The outcome of interaction design is to construct a common language of human computer interaction that allows end users to become co-designers of their interaction (Erickson, 2000b). That means a user should be able to generate personalized interaction to construct a common ground by integrating linguistic interaction (textual interaction) and non-linguistic interaction (direct interaction).

In summary, we are concerned with creating human computer interaction, including interactive artefacts and the resulting meaning of the interaction. This is produced by the interaction designer in the first place but developed by the users based on how they understand the interaction and the intended meaning of the designed interaction. Accordingly, the process of developing interaction has two–phases. In the first phase, the designers form a meta-interaction entity representing multiple original interaction concepts. The main task for this stage is to construct meaningful utterance [9] and genre [10] using the Domain Specific Interaction Language. This is used to engage the user to recognize the proposed concept and meaning, based on the designers' interaction design concept and purpose. In the second phase, the design concept is used to produce an interactive artefact, and will be represented by multiple interfaces and interaction models. To the end, the provided design concept will gradually realized by the end user. Meanwhile, the end user can accept or reconstruct the meaning of the interactive product based on his/her understanding and reflection.

Using linguistic structure to build human computer interaction allows users to capture the meaning of an interactive artefact and reorganize their

---

[9] Utterance: output from a computer which is intended to convey some meaning to the user.

[10] Genre: the mode of interaction used in human computer interaction e.g. un-linguistic interaction model (direct manipulation, gesture, touch) and linguistic interaction model (textual, voice), etc.

interaction. This is done by using a DSIL while operating a provided interactive artefact. In addition, the DSIL creates a common ground between the users and a particular interactive artefact in a specific domain. As a result, a personalized interaction is produced that involves both participants as a consequence of the progressive insights gained by the different stakeholders into the design process. It is also a consequence of the feedback provided by end users working with the system in a particular domain.

Accordingly, interaction could be altered by a variety of users in different ways coordinating with their domain knowledge and interaction needs through the interactive language interaction. That means a user should be able to produce personalized interaction by using a DSIL to define the utterance of interactive artefact and the genre of interaction. As a result, end users are involved in the creation of the interactive artefact from top to bottom. The resulting interaction is generated from the collaboration between the designers and the end users. This can improve the user's experience of interaction.

Like any other stakeholder, the end user, as co-designer, is allowed to comprehensively understand the content and meaning of interaction in a personal way. That means the user interacts with the interactive artefact to achieve his/her goals either through a predefined interface and interaction model or through the user's own personalizing of the interaction. The users are free to modify the interactive artefact through textual interaction customization language. As a result, the user gradually adapts his/her mental model and concept in building the imperative artefact. The interactive product will become more logical and customized to the end user.

Following, we will illustrate how to create a DSIL and describe its key components according to linguistic structure. We will exemplify how the DSIL optimizes human computer interaction by reflecting a user's individual interaction concepts.

4.3.2.1 Domain specific interaction vocabulary

On the top level, three essential interaction vocabularies make up every regular human computer interaction: user, computer (system), and interaction context. The above interaction vocabularies are shaped by a variety of sub components. The sub-components include words, picture, sound, movie, animation and so on. These are the interaction vocabulary of human computer interaction. Moggridge and Smith group elements of interaction vocabulary into different dimensions: 1D: words, 2D: visual representations, 3D: physical objects or space, 4D: time, 5D: behaviour (Moggridge and Smith, 2007). Figure 4. 4 shows five domains of interaction vocabulary of interaction language and illustrates how the interaction vocabulary is used to form human computer interaction.

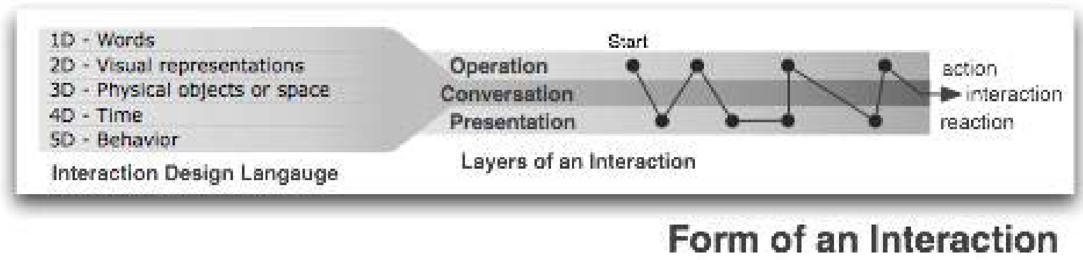


Figure 4. 4 Vocabulary of interaction

Following Moggridge and Smith's definition, the first dimension of interaction vocabulary is word or text. Each word or text corresponds to a different object, refers to different actions and represents a different meaning in a specific domain (Moggridge and Smith, 2007). Different working domains use different domain languages, so there will be a distinct painting language, music language etc. Text is normally used to describe key concepts and to construct domain specific knowledge. In interaction design, text can let the user quickly understand the function or meaning of an interaction.

The vocabulary of the second dimension includes painting, typography, diagrams, and icons. When we look at a painting, even if it is not representational, people can still understand it on some level. This understanding draws on the viewer's background experience and knowledge.

The vocabulary of the third dimension includes the physical properties of a product and how they convey some sense of the kind of activities one might perform with the product. It relates to how people make sense of the product. For example, if a product has a handle, we know are meant to grab it. Designers create different affordances in products by integrating the diverse physical elements of the product.

The vocabulary of the fourth dimension is related to time. Sound, film and animation belong to this type of vocabulary.

The last dimension of vocabulary is human behaviour. It relates to the human's action in interacting with a product. The interaction is based on the human's previous knowledge and experience and must be seen in the wider context of human social interaction (Zhuge, 2010).

Identify vocabulary

The first key task to build a DSIL is to identify the vocabulary of the interaction language in a specific domain. The designer must utilize multiple interaction vocabularies and choose appropriate interaction syntaxes to organize the interaction vocabularies to, in turn realize an intended interactive semantic. In other words, it is a process of transferring the designer's abstract interaction concepts to the users through different forms, such as building an interactive artefact and employing text to convey the intended meaning.

In terms of designing human computer interaction, designers need to explore intended interaction design concepts regarding the performance of an interactive artefact (object-focused interactive semantics), as well as the user's interaction experience (experience-focused interactive semantic), within a particular context. Table 3 presents key components of building DSIL for a typical interactive artefact (see Table 4. 3).

Key components of domain specific interaction language		
Interaction vocabulary	Interaction syntax	Interactive Meaning
• Text • Visual presentation • Physical object/space • Time • Human behaviour	Object focused interaction syntax-programming language	Based on designer's interaction concept: • Object-focused interaction semantic: A predefined functionality and usability for a product. • Experience-oriented interaction concept: A user experience model of using a specific interactive artefact.
	Experience focused interaction syntax- pattern language	
	User specific interaction syntax- user oriented domain specific interaction language	Based on user's interaction concept

Table 4. 3 Structure of the interaction domain specific language

Next, different types of interaction syntaxes are used to realize different interaction concepts (object-oriented interaction concept, experience-oriented interaction concept and user specified interaction concept). To do this, programming language and pattern language will be used. For example, designing a particular human computer interaction is to carry out a specific interaction concept to the end users when they using an interactive product.

Therefore, the interactive product comes from mapping out diverse interaction vocabulary in a network of objects or points in the classification space (see Figure 4. 5). For example, the object-focused interaction concept aims to produce a useful product under a usability goal. While the experience-focused interaction concept focuses on helping the users to have a desired experience. Eventually, the users are assumed to be able to explore the intended meaning and concept of interaction, provided by the designers through multiple user interfaces and special interaction models during the interaction.

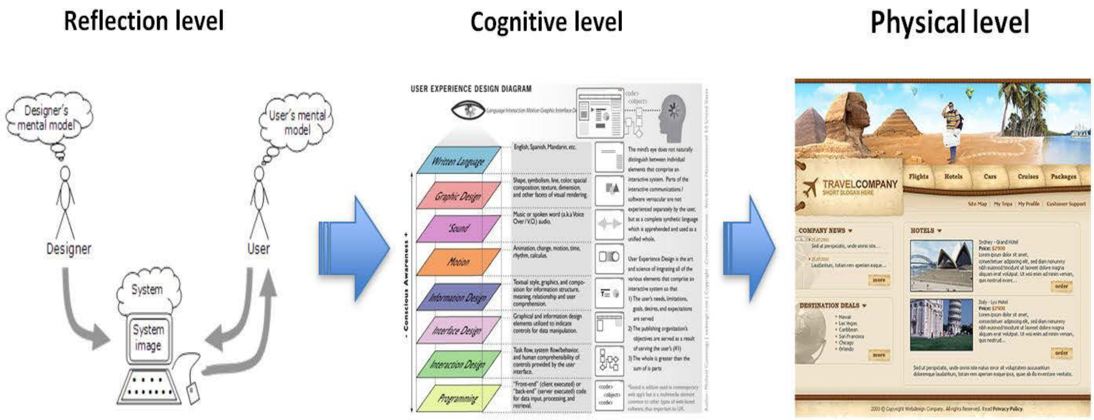


Figure 4. 5 Translating interaction concept to interactive artefact in different levels

The other significant outcome of using a ILDP is to work out the interaction concepts in a textual way. It is important to clarify the interaction concepts between different users and computers when we build human computer interaction. To do that, the textual interaction concept provides a way for the user to realize the meaning of interactions through text and to convey specified interaction concepts appropriately.

On top of that, a user-oriented interaction system has been established in the first place. Thus it changes the situation that the interactive semantics are mostly based on the designer's concepts rather than the end user. As we know, the problem of building human computer interaction based on the designer's understanding. As Krippendorff points out, the designer's understanding is a second-order understanding, which is different to the user's own understanding (Krippendorff, 2005). Here we aim to construct textual interaction concepts to appropriately transfer the designer's interaction concept to the end user. Moreover, the textual interaction concept can effectively affect the user's interactive semantic images and allow the user to make a direct expression using a specific interaction domain terminology through a textual interaction model. The user is not only allowed the predefined interaction concepts, referring to the meanings through the textual interaction concept. The user is also allowed to express his/her own interaction concept by using text-based interaction to point the text according to his/her recognition and experience of a provided interactive artefact and interaction context.

Accordingly, the computer system is set up under a communication architecture that can provide reasonable and appropriate responses to the user in a different ways, including the textual interaction mode. More important is that the interactive product is employed to realize the user's interaction concept. In particular, the participants' (users and designers) can perform their interaction concepts by choosing appropriate interface and models of interaction. That enables the user to clarify and modify the interaction semantics of interactive artefact and context easily, the interaction concepts are organised by well-defined domain specific terminologies based on domain knowledge.

The interaction designer combines the above interaction vocabularies to generate a variety of specific meanings through text and a designed artefact. In the following section, we will explore the structure of combining multiple interaction vocabularies. These vocabularies create meaning through the interaction syntax of the interaction language. Then, we will discuss how to create special meanings for human computer interaction through different interaction syntaxes.

4.3.2.2 Domain specific interaction syntax

The main goal for the designer is to build a particular interactive artefact to carry out variety of intended design purposes. Generally speaking, there are two essential interaction concepts: object-centred interaction concept and experience-oriented interaction concept. Both are used to produce concrete interactive artefacts to convey those interaction concepts. Primarily, two types of interaction syntaxes are employed to transform the above two interaction concepts: programming language and interaction pattern language.

Accordingly, object-focused interaction concept focuses on a usability goal, while the experience-oriented interaction design concept aims to accomplish a desired experience. Different interaction concepts are built upon the participant's different interaction semantic. In detail, interaction syntax used to transport the object-oriented interaction concept is programming language. Programming language has been used to construct object-focused interaction concepts by creating a concrete interactive artefact. By doing so, the interactive artefact can be knowledge in a way that the designer intends to let the user operate it. For instance, using this interaction syntax we can construct manipulation interface and instrumental interaction. In section 4.3.2.1 we have described these types of interface and interaction.

Additionally, design pattern language has been used to realize the experience-focused interaction concept. The main purpose of pattern language is to set up an interaction situation that derives from how a user interacts with the computer. Typically, the interaction between human and computer are carried out with a particular interface and interaction model and proceed to transfer different kinds of interaction experience, including physical, cognitive and reflective. In detail, the designer uses pattern language as experience-focused interaction syntax to make out intended experience-focused interaction semantics by combining variety components of interaction. The resulting interface and interaction model are: situated interface and specific interaction.

The syntax of the DSIL describes the essential structures that are employed to construct an interaction and the resulting meaning. The domain comprises features of interaction and the relations between various elements of interaction. From the perspective of language theory, syntax is a basic component of language and the bottom-level architecture of language (Allwood, 1976). The primary function of syntax is to produce a meaningful interactive product by semantically mapping out a variety of the interaction vocabularies already mentioned.

By investigating the interaction syntax, we aim to identify some basic constructs for building effective interaction and intended meaning between diverse users and the computer.

In the following, we will demonstrate how the interaction syntax can be used to structure human computer interaction and intended meaning by integrating the basic elements of interaction: character of user, attribute of artefact, and interaction context built upon on linguistic theory.

This means that diverse participators, including designers and users, can make a variety of meanings for a particular interaction through different interaction syntax architectures. From a communicating point of view, we believe that, when designers and users aim to construct an interaction, they are not just building a concrete interacting entity. They are, more importantly, conveying an intended meaning and desired experiences.

In this section, we explore three types of interaction syntaxes that are employed to generate diverse meanings and interacting entities between human and computer. These are: object-centred interaction syntax; experience-centred interaction syntax and user-oriented interaction syntax. Three different languages are employed to carry out the above three interaction syntaxes: programming language, design pattern language and domain specific interaction language. For instance, different interfaces and interaction models demonstrate the resulting meaning of each type of syntax.

4.3.2.2.1 Object-centred interaction syntax

The first syntax of the DSIL is object-centred interaction syntax. The object-centred interaction syntax aims to build a useful interactive product. This type of interaction syntax centres on an object-centred design concept such as system interaction design method (Saffer, 2007). Object-centred interaction syntax is mainly used to control the physical performance of an interactive artefact. It primarily pertains to usability and functionality of the product. In many cases, the designers construct human computer interaction regarding specific functions, features and properties of the interactive artefact. On a practical level, the users are allowed access to the interactive artefact's function, features and property. For example, a user utilizes a drawing system to make painting.

Programming languages are employed to carry out object-centred interaction syntax. By using programming languages, designers can comprehensively map out how the interactive product behaves. Programming languages are often object-oriented languages (Beaudouin-Lafon, 2004). In order to let a user control an interactive artefact or use a function similarly to its use in the natural world, programming languages are used to enable designers to enable human computer interaction. Essentially, programming languages try to transform the real world model into computer code using some basic semantic abstractions such as class, object, instance, inheritance, method, message, encapsulation and polymorphism (Rumbaugh et al., 1999). The design starts from the available hardware or task and ends up producing an interactive system that allows a user to manipulate the compute-based objects directly.

As a result, object-centred interaction syntax manifest in a programming language leads to an interaction that can support the user's direct manipulation of the interactive product.

From the above we can see that object-centered interaction syntax defines how a tool is going to be used by a user in a particular interaction context based on the designers' design concept and purposes. The designer aims to engage the user to operate the interactive artefact in a predefined way and context. Next, I will give an example to explain how designers construct an interactive product using object-centered interaction syntax - programming language.

Programming language

Objective-C is a general-purpose, object-oriented programming language[11]. The programming language Objective-C was originally developed in the

[11] http://en.wikipedia.org/wiki/C_%28programming_language%29

early 1980s. It is the main programming language used by Apple for the OS X and iOS operating systems, and their respective application programming interfaces (APIs), Cocoa and Cocoa Touch[12].

Manipulation interface

As we know that interface and interaction of iPhone can be created by using a programming language - Xcode. Xcode is an integrated development environment (IDE) containing a suite of software development tools developed by Apple for developing software for OS X and iOS.13. Figure 4. 6 shows the basic functionality of the system constructs outline of the iPhone interface and its basic component elements, including a limited set of components of interaction, such as “Widgets” (application window, buttons, dialog boxes, scrollbars), drag-drop and copy-paste to transfer data within and across applications. (See Figure 4. 6). The object of designers/developers is to try to convey their interaction design concept to match the user’s needs and requirements in certain application contexts.

[12] http://en.wikipedia.org/wiki/Objective-C

[13] "Xcode on the Mac App Store". Apple Inc. Retrieved October 3, 2012.

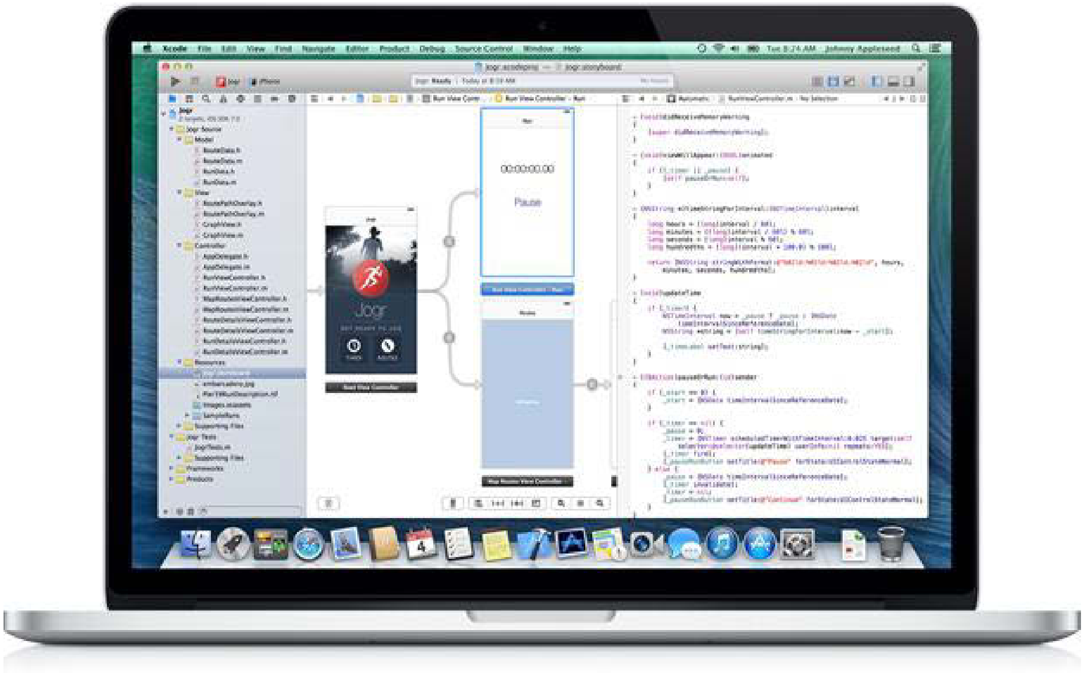


Figure 4. 6 Layers of Instrumental interface of iPhone

Instrumental interaction

Programming languages can be used to produce a form of interaction– instrumental interaction – shaped by the affordances of the tool. Instrumental interaction centers on direct manipulation that supports tools as mediators between users and a target object. In so doing, instrumental interaction enables users to directly operate on diverse digital objects presented by iPhone through its operating interface.

In actual interaction design, much research and practice concentrates on task analysis and usability testing to clarify a suitable structure for representing the interactive product. Normally, the design work starts by considering the user's goal and intentions, formulating primary task and sub tasks (e.g. find an appropriate tool, change the property of a tool). Figure 6 illustrates a set of instrumental interactions to control iPhone. These actions are carried out when the user wants to achieve a goal such as opening a file or getting an appropriate tool (see Figure 4. 7).

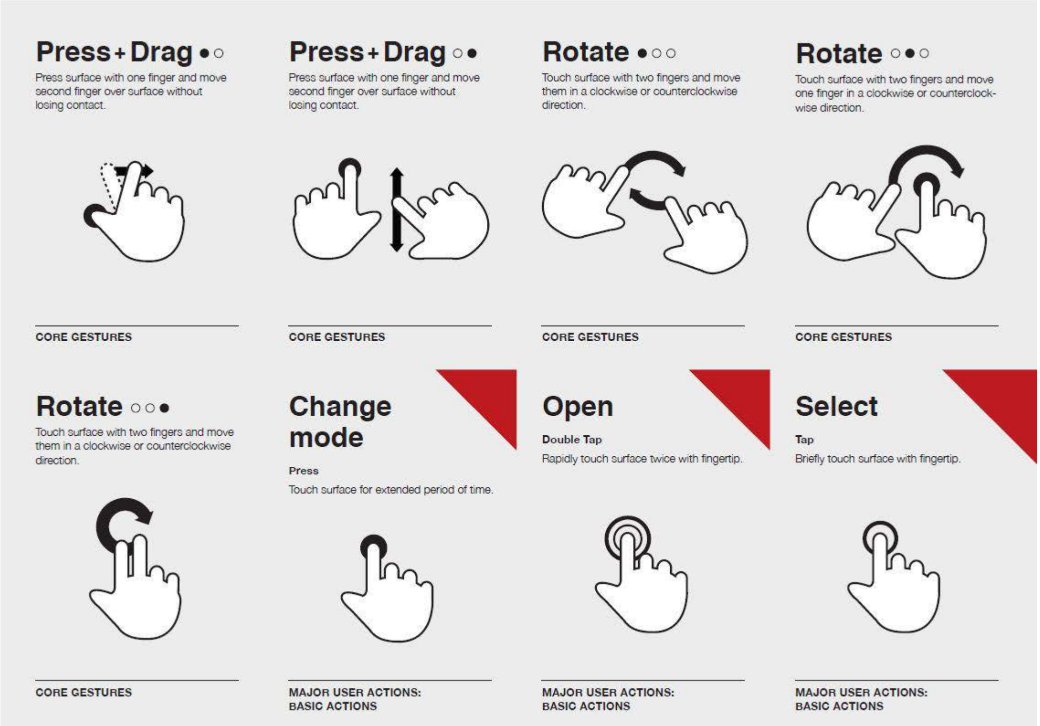


Figure 4. 7 A set of instrumental interactions designed for operating iPhone
http://www.lukew.com/ff/entry.asp?1071

This type of interaction can support most typical straightforward human machine interactions. However, from a cognitive psychology point of view, in order to make the user effectively operate a product, users need to address different issues of manipulating the interactive product at different levels: visceral, behavioral and reflective, rather than simply forcing the users to use it. The subsequent problems have been discussed in Chapter 2.

4.3.2.2.2 Experience-oriented interaction syntax

The second syntax is experience-oriented interaction syntax. Experience-oriented interaction syntax is used to generate situated interaction by constructing a specific interaction context regarding the user on the cognitive level. It formalizes a user's cognitive behaviour and performance of an interactive artefact (system) to derive diverse intended interaction experiences. Initially, these interaction experiences are defined by the designer based on the designer's perspectives to a specific task according to different classes such as useful, effective, enjoyable, pleasurable and aesthetically pleasing. Generally speaking, experience-oriented interaction syntax aims to characterize a particular interaction to bring a user a desired experience while using an object. In other words, a specific interaction is created according to the desired experience to organize a user's interacting cognition and activity.

Experience-oriented interaction syntax takes advantage of human-centred design methods focusing on the user's interaction experience (Wright et al., 2008). From a human-centred design perspective, designers are more likely to deal with how to provide the user with a desirable experience from interaction (Rogers et al., 2011). This approach mainly depends on the designers' understanding, which is based on analysis of the user's standard cognitive workflow, understandings, knowledge and abilities within a particular interaction environment. Much work has been done in this area and many technologies are provided for designing interaction, such as storyboards, task workflows, scenarios, personas and so on. Figure 4. 8 displays a design framework for creating user interface of website focusing on user experience (see Figure 4. 8).

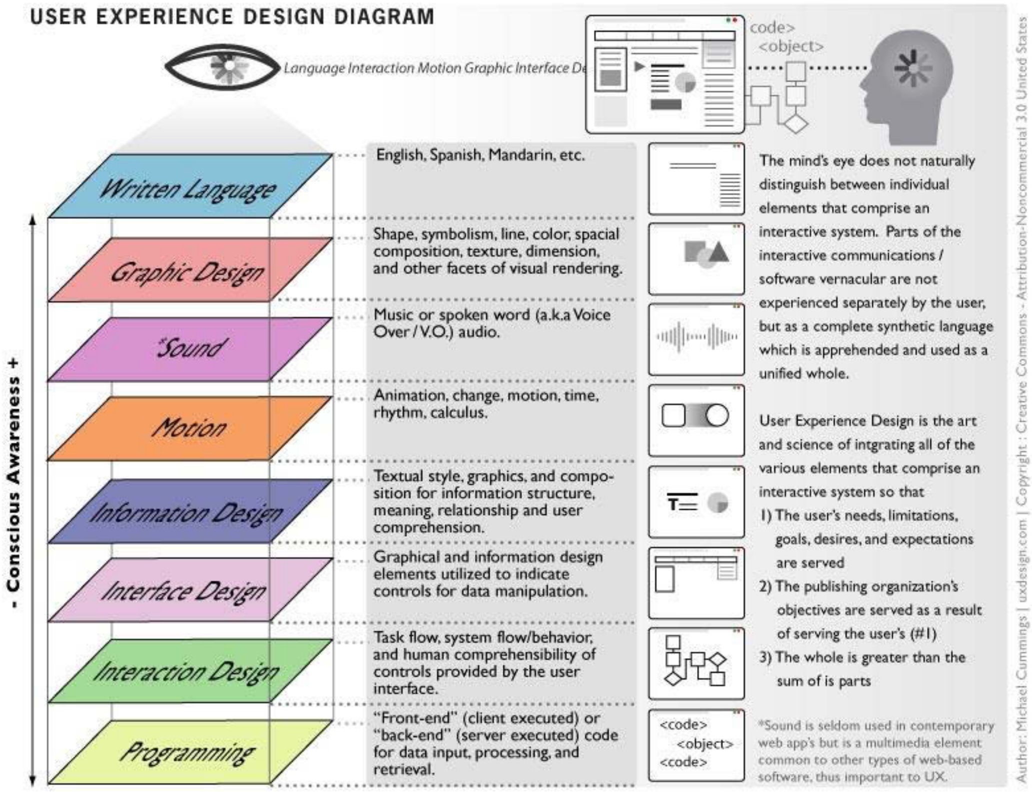


Figure 4. 8 Architecture of building user interface focusing on user experience (http://uxdesign.com/ux-defined)

Pattern language

Experience-oriented interaction syntax is generally realized using pattern languages. Pattern languages propose particular interaction designs based on the ideas of how users interact with the computer. A basic assumption for creating an interaction design pattern language is that the pattern provides a way to store the basic design knowledge of human computer interaction (Dearden and Finlay, 2006). There is general agreement that patterns provide some rationale for particular design decisions. Norman describes pattern languages as having the potential to organize and affect people's recognition and behavior (Norman, 1988).

In addition, patterns are related to each other. That means the patterns can be formed into a network of connected patterns to support the designers responsible for the definition of the system's overall architecture. The relationships are at the heart of the pattern language because they create actual additional value over that of a single pattern.

Simply speaking, pattern languages help designers to design different interaction experiences including physical, cognitive and reflective by forming diverse specific interactions and particular interfaces in the first place. Thus, we claim that the designer uses pattern languages as experience-oriented interaction syntax to create intended experience-oriented interaction semantics by combining various components of interaction.

In the interaction design domain, several alternative organizations of pattern have been proposed. Fischer investigated some of possibilities for structuring pattern languages (Fischer, 2001). Mahemoff and Johnston classified interaction patterns for tasks, users, user-interface elements and entire systems (Mahemoff and Johnston, 1998).

Interaction design pattern languages are hierarchical. They can help the designers to create a comprehensive interaction when the patterns are available at all levels of the user's interaction system. Table 4. 4 provides an example of an interface design pattern language made by Tidwell (see Table 4. 4). Interface design pattern languages have been used for designing different user interface for many years and have proven to be a very useful design method regarding to user experience.

Pattern name	Fat Menus
What	Display a long list of navigation options in drop-down or fly-out menus. Use these to show all the subpages in site sections. Organise them with care, using well-chosen categories or a natural sorting order, and spread them out horizontally.
Use when	The site or app has many pages in many categories, possibly in a hierarchy with three or more levels. You want to expose most of these pages to people casually exploring the site, so they can see what's available. Your users are comfortable with drop-down menus (click to see them) or fly-outs (roll over them with the pointer).
Why	Fat Menus make a complex site more discoverable. They expose many more navigation options to visitors than they might otherwise

<table>
<tr><td></td><td>find.

By showing so many links on every page, you make it possible for a user to jump directly from any subpage to any other subpage (for most subpages, anyhow). You thus turn a multi-level site—where subpages aren’t linked to the subpages in other site sections—into a fully connected site.</td></tr>
<tr><td>How</td><td>On each menu, present a well-organised list of links. Arrange them into Titled Sections if they fit into subcategories; if not, use a sorting order that suits the nature of the content, such as an alphabetical or time-based list.

Use headers, dividers, generous whitespace, modest graphic elements, and whatever else you need to visually organise those links. And take advantage of horizontal space—you can spread the menu across the entire page if you wish. Many sites make excellent use of multiple columns to present categories. If you make the menu too tall, it might go right off the end of the browser page. (The user controls how tall the browser is; guess conservatively)

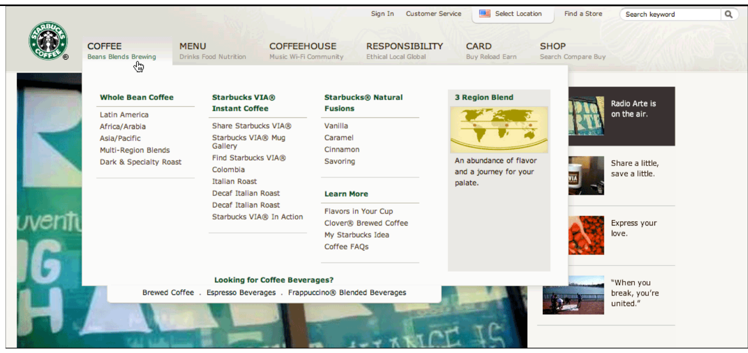
The best sites have Fat Menus that work stylistically with the rest of the site. Design them to fit well into the colour scheme, grid, and so on of the page.</td></tr>
<tr><td>Examples</td><td>The Fat Menus on the Starbucks website are very well designed. Each menu is a different height but the same width, and follows a strict common page grid (they’re all laid out the same way). The style blends in with the site, and the generous whitespace makes it easy to read. Ads are worked into the design, but not obnoxiously. The non-rectangular shape adds a polished look.</td></tr>
<tr><td></td><td>
</td></tr>
</table>

Table 4. 4 Interface design pattern language of fat menu created by Tidwell, Jenifer (http://designinginterfaces.com/patterns/fat-menus/)

Situated interface

Typically, in interaction design practice, interaction designers create an interface by following a hierarchy. They start by gaining an understanding of the users and their tasks, the users' purposes, technical environment, business context etc. (Rogers et al., 2011). Erickson sees the use of patterns as "lingua franca" to support and enhance communication about design. In particular, he advocates the use of patterns to help users engage with the design processes (Erickson, 2000a)

Here, the interface is considered a specific situated space in which to construct a communicative space according to a particular interaction situation. Interaction designers construct an actual interactive system: Photoshop, based on a design concept of providing the users with a useful painting system. Initially, the designers aim to produce a painting context by integrating various painting tools and palettes to make drawing become more effective. Accordingly, the situated interface guides the user's understanding of the interaction context. You can see a situated interface is built for a drawing system, Photoshop. It is designed like a drawing board, comprising a tool bar, side, menu bar drawing window, and top menu bar. The purpose of designing such an interface is to make the user feel familiar and comfortable to use the drawing system. This is based on the analysis of how people make drawings in the actual world, for example, using a colour board to make colour. Therefore, when the user takes use of the provided interface, they are assumed to recognize it easily and use it effectively reflecting on experience they have had in the real world (see Figure 4. 9).

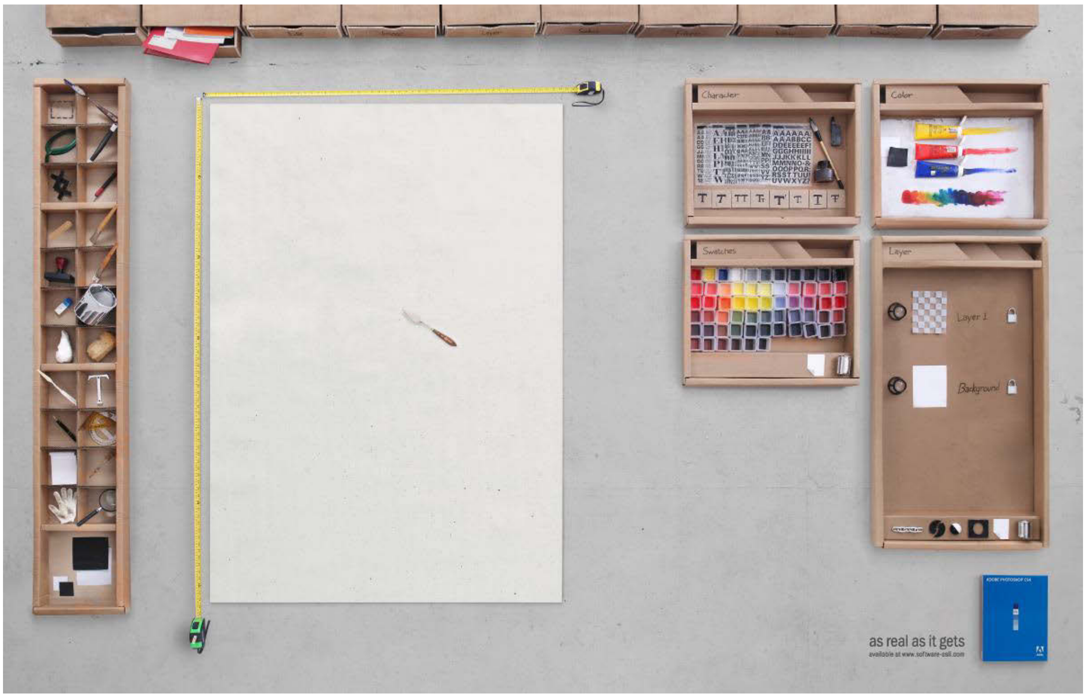


Figure 4. 9 Interface of Photoshop (http://tanhands.blogspot.com.au/2010/02/real-life-photoshop.html)

Specific interaction

Eventually, designers will generate a suitable interaction model based on their user research results. This is a human-centered design model, focused on human cognitive psychology and interaction behavior. Figure 4. 10 provides a simple structure of composing a specific interaction. By following the structure, the users can operate a particular product, in this case a drawing system, effectively and successfully. In general, the structure of this specific interaction and function that cannot be changed by the end user. It requires the user to follow the order of interaction and find the location of the button in order to find the appropriate function and to complete the special task successfully and effectively at any time.

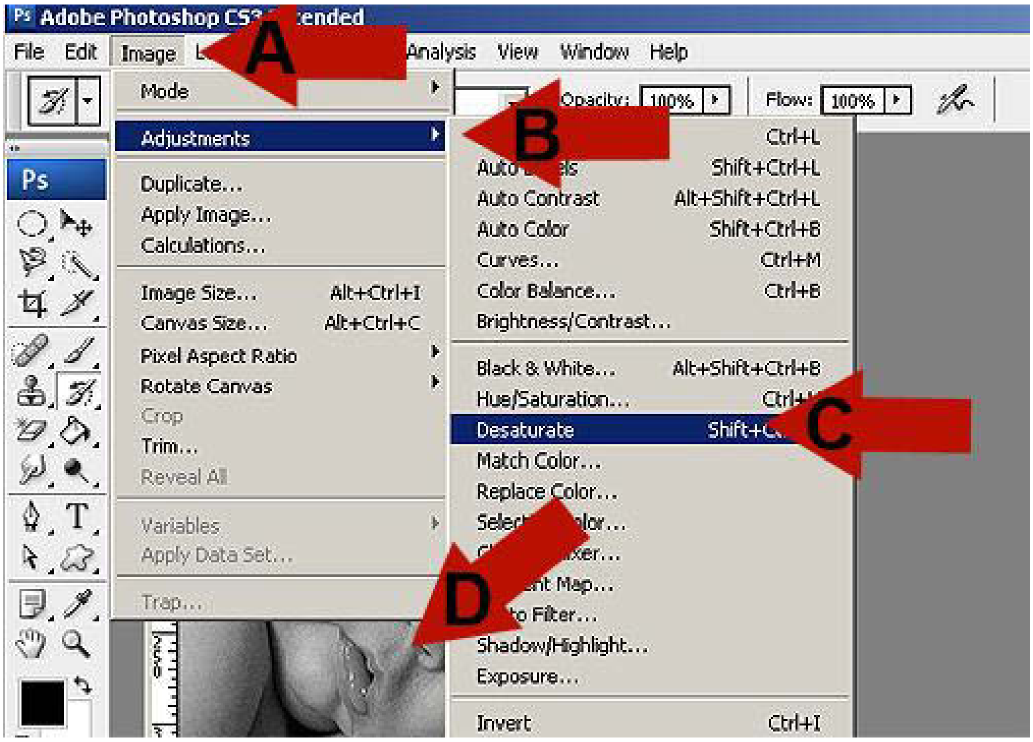


Figure 4. 10 Structure of specific interaction for operating Photoshop (http://www.lonerobot.com/images/Photoshop/psclasses.jpg)

The specific interaction leads the users to realize its underlining interaction structure in operating the system and allows the users to use the system in certain ways.

However, the situated interface and specific interactions work very well for some of the users in a particular situation but work less well for other users in different situations. People simply do not act or interact with an interactive product (system) in the same way prescribed by these kinds of situated manipulation models (Rumpe., 2004). Although user modelling is a way to generate meaningful interaction in a certain application but it is hard to create one model to suit all applications (Zhuge, 2010).

Therefore, it is necessary to provide a user-oriented language to balance the communication between the user and the computer. Research shows that different users work in very different ways (Fischer, 2001). This is true, even for the same user who will apply different interaction patterns according to different interaction contexts (Mackay, 2002).

In the next section, we propose a user-oriented DSIL with which to carry out user specific interaction syntax to establish a mutual conversation with a computer.

In the next section, we will illustrate that how the domain specific language facilitates the end user's expression of his/her interaction semantic to establish a mutual conversation with a computer.

4.3.2.3 Realize user's interactive meaning

The DSIL pragmatic is the user defined interaction syntax. For the DSIL to transform the user's abstract interaction concepts into a concrete interactive artefact, it requires different types of interaction syntaxes to complete different interaction concepts (object-oriented interaction concept, experience-oriented interaction concept and user specified interaction concept). As a result, the DSIL allows each participant of human computer interaction to communicate with each other on a physical, behavioural and emotional level. This can achieved because the domain specific interaction design concept can be fully understood by the end user through interacting with concepts via text.

More importantly, after experiencing the interactive object through the provided interface and interaction models, the user will perform his/her own interactive semantic for the interaction. This type of interaction can be accomplished through textual interaction.

As we mentioned, DSIL is a user-oriented interaction language. The DSIL is proposed as the user's interaction syntax, used to alter the outcome of interacting with computer. Consequently, the DSIL would help the end user to solve the problem of interaction at different stages and levels during the interaction. By using a domain specific interaction language, a user can create a specific meaning for the interaction with which to express their interactive concepts. The result will be performed by computer in the form of a changing interface and interaction model.

From a cognitive psychology point of view, different interfaces and types of interaction will evoke the user's experience of interaction at different levels such as visceral, behavioural and reflective (Norman, 1988). For instance, on the physical level, the user will empathise with the object-oriented interaction concepts through instrumental interactions. Similarly, the user is supposed to capture the experience-oriented interaction concept by employing specific interaction models. Figure 4. 11 shows a procedure of how an interactive product is changed by the user, based on his/her perspective and purpose through using language.

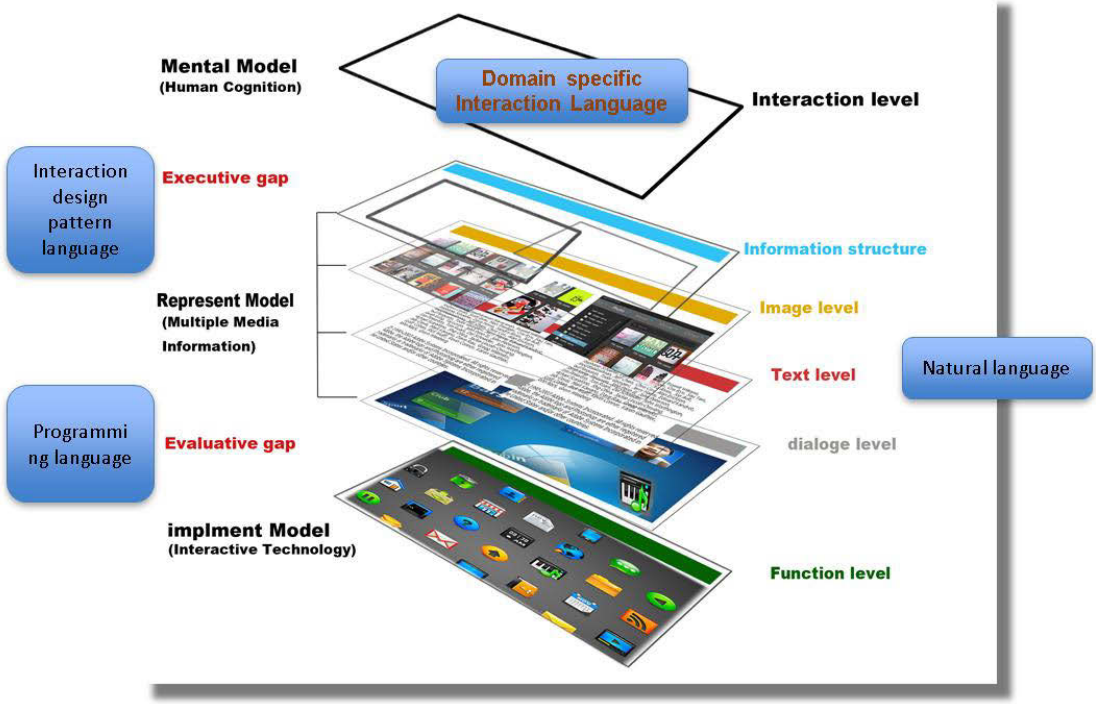


Figure 4. 11 Procedure of building personalized human computer interaction

DSIL provides a critical expressive interaction method that permits the user to personalise the interaction according to his/her personal interaction concepts including object-oriented, experience-oriented interaction concepts. For example, to let the user express his/her interactive semantics, it is best to let the user to perform an appropriate interaction directly. An example would be to use text to demonstrate the concept that he/she wants to achieve. Human natural communication is a good example.

Figure 4. 12 shows an example of how the mutual interaction workflow happens between human and computer.

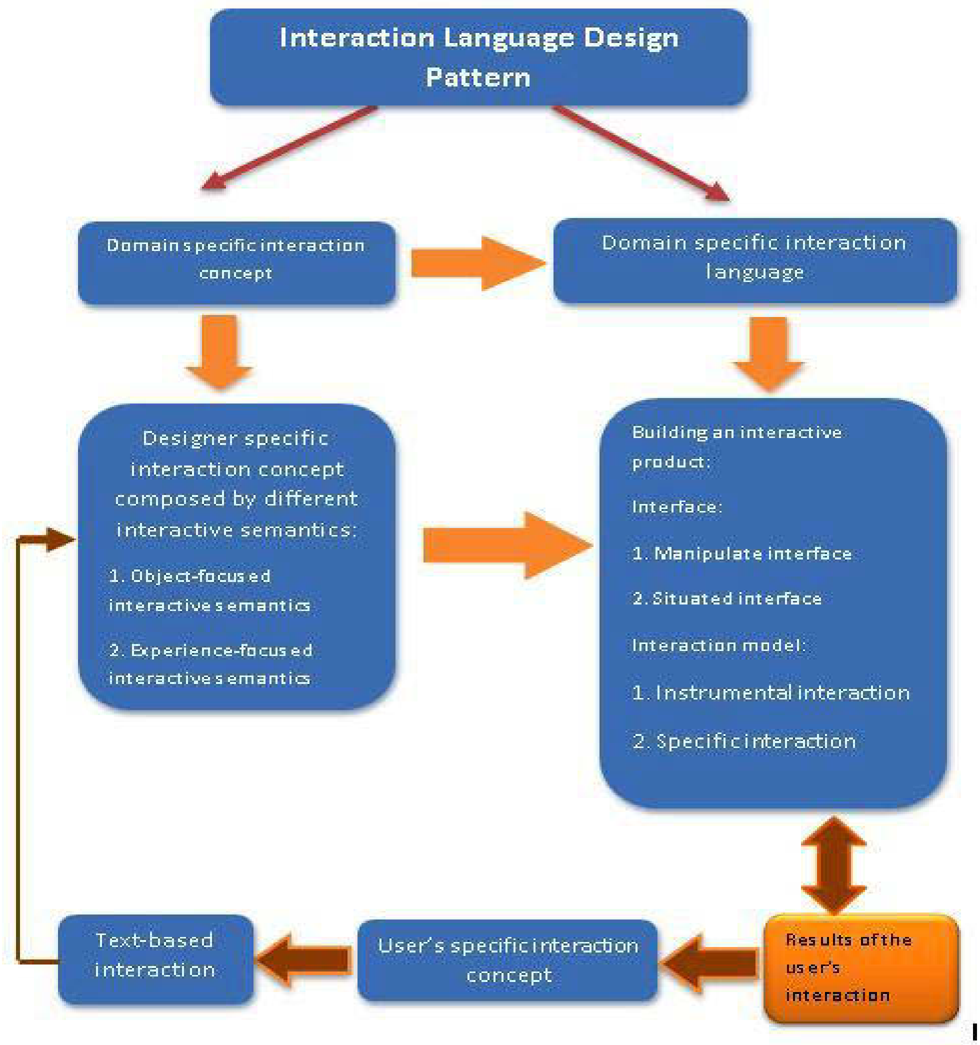


Figure 4. 12 Workflow of mutual interaction between human and computer

There are two aspects to applying the DSIL to the process of realizing a user's interaction concepts. The first is object-oriented interaction semantics and the second is experience-oriented interaction concept.

Table 4. 5 demonstrates how the user utilizes interactive language interaction to construct an appropriate common ground at different levels.

Ultimately, the interaction language pragmatic pattern is going to be accomplished by the user when the DSIL is used to optimize the system created by the designer. As a result, through the interactive language interaction and built upon the interaction domain specific language, both user and computer (designer) are able to understand and actively cooperate with each other.

DSIL pragmatic		
The user's interactive semantics	Interaction models	Level of interaction
Object-focused interaction semantic	Manipulation interface and Instrumental interaction	Physical level
Experience-oriented interaction concept	Situated interface and Specific interaction	Cognitive level
User specific interaction semantic	Text based interaction model - Using the DSIL to make interaction	Emotional level

Table 4. 5 DSIL pragmatics

In the following section, we will elaborate on how textual interaction supports the user's collaboration with the computer according to the user's expectations. In particular, we will give a detailed description of how the interactive language interaction is used by the user to interact with the computer.

User specific interaction syntax

The third interaction syntax is user specific interaction syntax. The user specific interaction syntax reflects a user's specific perspectives and needs of interaction. The way a user defines an interaction depends on his/her personal knowledge and reaction to a particular interactive product. In more detail, the user's individual interactive semantic images drive how the user recognizes the interactive artefact (Zhuge, 2010).

User specific interaction syntax determines how a user will define a particular interaction context and property of a tool under his/her concept or mental model (Norman and Draper, 1986). This model is generated from the user's interactive semantics. We will illustrate how the user's interactive semantic image can be generated by a user using a drawing system in paper prototype study and Hi-Fi prototype study.

We argue that the user specific interaction syntax is a useful way for the user to form an individual interaction model with which to transfer a specific interactive concept. The personalized interaction model and concept play a key role in constructing mutual communication between the user and the artefact.

David Liddle points out that the most important thing for interaction design is to capture the user's conceptual model or mental model. Everything else should be subordinated to making that model clear, obvious, and substantial (Liddle, 1996). Accordingly, through using the user specific interaction syntax, the user can express his/her interaction concept or mental model by personalizing the interaction. In other words, if we can provide the user with a way to define his/her interaction, it is possible to build a personalized interactive product that the user already has in his/her head.

Unfortunately, user specified interaction syntax, focusing on building user-oriented interaction, is not yet fully investigated in current HCI. One significant problem is that the user's interaction concepts and interaction syntaxes are intangible and are stored in user's mind. These are difficult to catch and it is even more difficult to create an interactive artefact, which can adapt to a diversity of users' concepts and requirements.

Although many works have been developed to help people become the owner of their interaction, they have had only limited success. For example, highly customized interfaces are created based on the above perspectives. These include adaptable, adaptive interface and natural interface designs (Dumas et al., 2009)

Another solution from software engineering is to make a computer language easy to learn by normal users. These are referred to as natural programming languages (Pane and Myers, 2006). Natural programming languages can improve the user's ability to use computer language at a certain level, but, so far, regular users still find it difficult to learn computer language in a short time (Pane and Myers, 2006).

The other important research area focuses on this set of problems is Artificial Intelligence (AI). For instance, an important research goal of AI is to make a computer or machine able to understand the human's thinking through different interaction methods, including gesture, facial expression, language etc.

In view of this, we claim that a user specific interaction can best be formed by the user him/herself rather than other people such as the designer. What is needed is a type of interaction that allows users to express themselves naturally. Forlizzi and Battarbee call this type of interaction *expressive interaction* (Forlizzi and Battarbee, 2004).

To create expressive interaction, we have to explore a user's actual use, co-ordination and interpretation of interaction and its meaning. Research shows that human beings can conceptualize outside world to build a semantic worldview and are able to exchange their meaning with other individuals through language. On top of that, human interaction and society are inseparable, and human interaction is the most basic social behaviour (Zhuge, 2010).

These characteristics result in different users' experiences depending on the levels of communication, including visceral, behavioural and affective (Norman, 2007). In other words, the quality of the user's interaction determines whether the user can reach various types of intended purposes, derived from interaction such as usability, effectiveness and emotion.

Many researchers agree that it is necessary to have a common language for human computer interaction (Erickson, 1998, Erickson, 2000a, Winograd, 1986). Through the common language, users are able to personalise the interaction by expressing the various things they need from the interaction.

User defined object-oriented interaction concept

Interaction designers utilize programming languages and pattern languages to build an initial interaction space. An object-oriented interaction provides the necessary functionality and usability. At the same time, the attributes of the interactive artefact can be manipulated using the text-based interaction domain specific language. Through this, interaction can be developed in a personalised manner. The user of the interactive product can use the DSIL to achieve an objective that is appropriate for the user.

The textual interaction concept can fix some potential problems that the user may encounter while operating the physical product. For example, using traditional approaches, a user may have difficulty finding an appropriate function when he/she is not familiar with the attributes and structure of the interactive product. The interactions do not allow the user to work with the computer in a personalised way. This problem happens more often when a user interacts with a complex system but lacks specific interaction domain knowledge to operate the system. Different users will have different domain knowledge, even within a particular domain. For example, two different ‘expert’ Photoshop users might each use a distinct subset of the available features of the program. Although each knows what they want and what it is called, they will have to wade through dozens of superfluous menu items for functions they will never use in order to find the one thing they will use.

Initially, on the physical level, through a text-based interaction the users are able to use interaction language to clarify and modify the object-oriented interaction concept. For example, this might involve immediately getting an appropriate tool and redefining it with a new concept.

User defined experience-oriented interaction concept

On the cognitive level, even for the same interactive artefact, different users will have different understandings. The user's understanding or experience will lead them to make rational reactions. As we mentioned before, current interactive artefacts can be difficult to personalise. This is because the form of the communication usually comes from the designer, not the user him/herself. This causes some recognition gaps such as executive gaps and evaluative gaps (Norman, 1988). HCI research shows us that, in many cases, the semantics of interaction created by the designer become problematic after it has been transferred to users through a particular interactive artefact (de Souza et al., 2001). Normally, only part of the interactive semantic, including a concept of product and experience of interaction, can be conveyed to the end user, as each user has very different understandings and abilities. The conversational problem still remains in current human computer interaction (de Souza et al., 2001, Ryu and Monk, 2009).

4.4 Outcome of using DSIL

4.4.1 Semantic interface

Here, we will show how a user converts a standard user interface model to a user defined semantic interface. Then we will describe the actual user interface rendering system and how it is facilitated by interactive language interaction. Meanwhile, we will illustrate how a concrete user interface can be automatically generated to express a user specific interface, based on a user's interactive language interaction.

Initially, the interaction designer or producer provides an original interface for a particular project. An interface can be recognized on two levels: physical level and cognitive level. On the physical level, the interface presents a notational system that typically appears as an instruction interface. The user can manipulate the interactive artefact and change the interface. Figure 4. 13 gives an example to show this procedure.

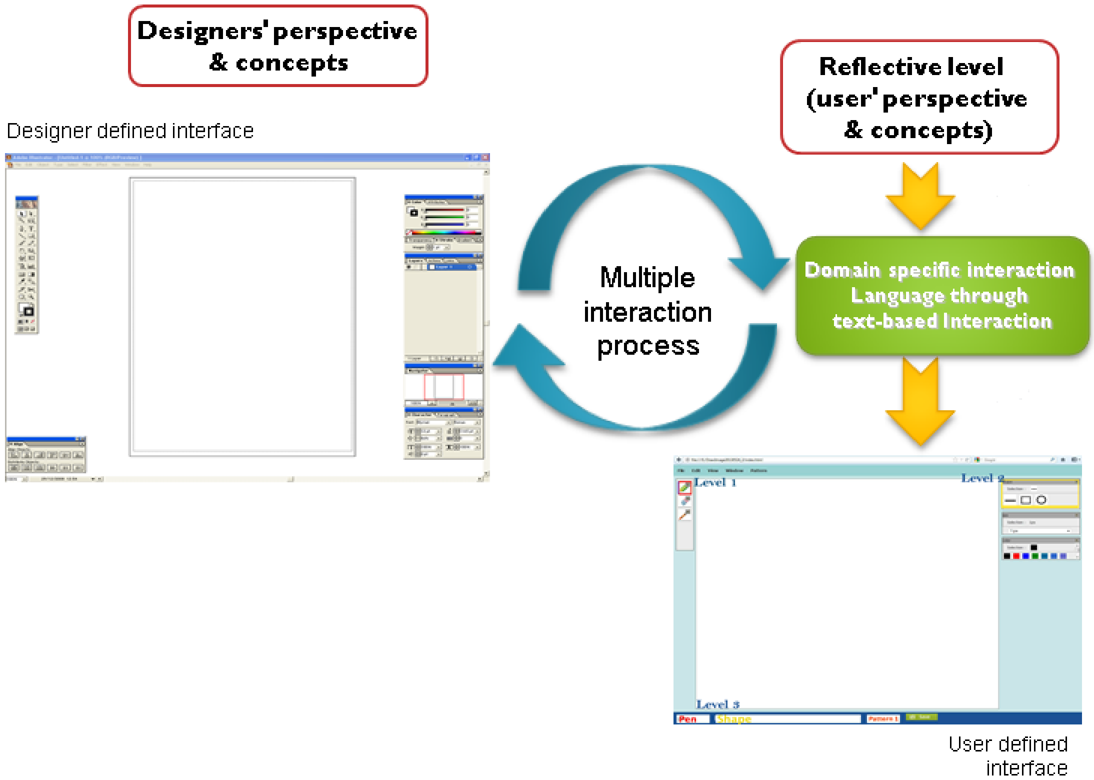


Figure 4. 13 Procedure of constructing semantic interface

Users are able to make changes to their interaction in order to make it better fit their needs. During the process of interactive language interaction, the rendering (i.e. the appearance and layout of the application) is changed based on device profiles, application-specific style guides and the particular situation. The device profiles contain device specific constraints regarding the device's capabilities. The user preferences and application-specific style guides influence the layout, selection and rendering of widgets.

4.4.2 Personalised interaction

The main purpose of interaction language is to facilitate personalised interaction. As we have discussed, information or content is mainly exchanged between human and computer through two approaches: linguistic interaction (text) and direct approaches like observing, listening and touching (Djajadiningrat et al., 2002). For example, information can be delivered to the user by reading signs, pictures or browsing a drop down menu to find an item. This may become difficult when the user is required to repeat a meaningless action - including clicking buttons and looking through function menus to complete tasks – many times.

When the user faces a problem in manipulating the system, he/she can alter the object (to trigger a new function), by typing a domain specific word, to get to the appropriate function. In addition, the user only needs to type the first letter to have semantic support from a well-defined menu to find an appropriate option. Once the user chooses a word, it triggers the corresponding function and its icon appears on the screen. Based on the same reasoning, the user can modify the interface to carry out his/her work more easily. Consequently, a personalised interaction is co-shaped by a user's typing a word and the computer's responses.

Additionally, the quality of interaction can be improved by providing a modified semantic selection list based on the user's preference. For example, each time the user types in a particular word, the system recognises the word and offers up a list of options in a particular order. Additionally, the user is able to define the functions and patterns by naming them. By doing this, human computer interaction becomes a more cooperative activity and supports diverse contexts and levels of communication. In order to produce a personalised activity, the user will consider how to make the interaction more effective, suitable and sustainable. Consequently, the meaning and form of interaction is transformed such that the user is able to code the interaction with special object-oriented and experience-oriented interaction concepts. In return, the system will become more productive and effective.

On the affective level, the user can systemize his/her interaction to complete a particular task or a goal, based on the user's personal intention or mental model. That means that a user-oriented interaction will be rendered based on a user's affective participation through his/her interaction. In other words, an individual common ground (physical, cognition and affective) is established - a personalising interaction pattern. The pattern indicates how a user builds a particular interface and interaction model based on individual knowledge.

The DSIL aids the process of realizing a user's interactive semantic as language in use. Effective personalised interaction can only be produced from a well-defined and rich semantic representation and a user-derived interaction. The level of interaction determines the quality of conversation and the relationship that is generated through interaction. The users utilize a user-oriented language to find an appropriate function or to arrange a property of a tool that is the same as when a designer uses computer language to control the computer.

The key notion of interactive language interaction is that it involves the end user's interactive semantics, through an interactive language interaction, to establish and change relations between human and system. Interaction also influences the motion and evolution of the object and evolves the participants' interactive semantic images. As Zhuge points out, the semantic image is a network of objects or points in the classification space shared between individuals. It forms and evolves through constant interaction. Human interaction semantic image is dynamically reflected by different images such as classification, object, individual, relation and rule (Zhuge, 2010). Accordingly, interactive language interaction is built based on a user's dynamic semantic images and the evolution of the human computer interaction over time. As a result, effective personalized interaction will emerge when the interactive artefact and the situation of interaction are both altered and customized based on the user's concept and his/her interactive semantic. In next two chapters we will create a DSIL for a drawing system to illustrate the producing process of building a personalized interaction using the interaction language design pattern.

4.5 Chapter summary

This chapter described how to produce a personalised interaction using ILDP. The outcomes of ILDP give weight to our assumption that end users should be able to define and alter their interaction according to their perspective and needs. The new interaction model allows the user to personalise their interactions with the computer.

A user specified interaction language has been constructed to enable personalised interaction. The user specified interaction language is the result of the user's evaluation of the interactive system on physical, cognitive and affective levels. Consequently, the ILDP constructs a balanced conversation between human and computer by incorporating the principles of human communication discussed at the beginning of this chapter.

In section 4.2 we explained how the users express their experience through constructing a domain specific language. In addition, during the interaction, the users play an active role in defining their interaction. ILDP constructs a personalised interaction by encouraging the users to make decisions according to their knowledge, perception, understanding and attitudinal reaction.

In 4.3.2, DSIL interaction is introduced as an important outcome of ILDP. This interaction language interaction supports a balanced conversation by enacting a semantic communication between a user and computer. For instance, interactive language interaction allows the user to input a special word or sentence to enrich the user's experience of interaction based on the user's individual intention of interaction and their domain knowledge. Through the interactive language interaction, the users sustain a conversational interaction with the computer.

In section 4.3.2.1and 4.3.2.2 we introduced key components to develop DSIL for generating personalized interaction. This interaction language enables the user to collaborate with computer on a semantic level. In section 4.3.2.3, we illustrated how the users express their interactive semantics using the domain specific domain language that provides an important way to support interaction, which is closer to natural human communication. All the words have unique meanings to different users in a different circumstances or stages of the application. The domain specific words help users to operate a system in a more appropriate and personal way as the interactive language interaction utilizes specific domain vocabulary, directly linked to the target of interaction through property of object and features of context.

Finally, in section 4.4 the outcomes of using the domain specific inaction language interaction are described, including a semantic interface and a personalised interaction model. That means a common language for human computer interaction will emerge when the users are able to fully participate in the process of designing interaction and are engaged in a co-experience developed cooperatively with the computer.

In the next chapter, we will give a demonstration to illustrate how a user specified interaction pattern language generates a semantic interface and personalised interaction and hence, communicate effectively.

Chapter five: Usability Study

This chapter presents the case study of using Interaction Linguistic Design Pattern (ILDP) to design human computer interaction. In the first section, a paper prototype for a drawing system is created based on the theoretical framework of ILDP. In the second section, usability testing is conducted to complete a set of drawing tasks. The production process is described, and comments from usability testing are analyzed.

5.1 Rationale

In this chapter, we are going to give an example to illustrate how we can use the Interaction Linguistic Design Pattern (ILDP) to create a user specific interaction language, which allows the user to operate a computer in a more personal way. Interaction language increases productivity by enabling the user to be a co-designer of designing the interaction. Here we create an interaction language for a drawing system. The intention is that a user specified Interaction language produces a semantic interface and personal interaction that makes the conversation becomes more effective between the user and computer.

By providing a user-specified language, it enables a more reasonable interaction. Specifically, we provide a linguistic input bar which allow a user to input any word or sentence related to the interaction (5.3). That means a user is allowed to input a word to operate the system based on his/her conceptual idea, which is difficult to achieve by using traditional interaction. Furthermore, the words are user-oriented domain specific linguistic vocabulary (5.3.1.1). Additionally, all words are specific to users who use a particular system with different application purposes, and support the users to carry out semantic interactions. Through the semantic interaction, the intention is that human computer interaction becomes more appropriate and productive compared to traditional interaction such as instrumental interaction and specific interaction (5.3.1.2).

An important outcome of using ILDP is that a user semantically models the interface and interaction as a result of conversation with computer (5.3.1.3). Eventually, the interface will be automatically modified according to the users' interactive semantics (Zhuge, 2010). Accordingly, the system performs rational responses in a collaborative way by providing the user with a meaningful interface (5.3.2.6).

A paper prototype has been built for a paper prototype study (5.3.2.6). The results of the study have been used to build a Hi-Fi prototype, and it will be used for Hi-Fi prototype study and drawing system user testing (5.4).

5.2 Research aims and questions

In this chapter, a main objective for this case study is to explore how to create a user personalized interaction by using ILDP by creating a drawing system. The intention is that through the personalized interaction, a user is able to construct common ground during the process of interacting with the system. Textual interaction is used to personalize the interface.

The specific research questions for this case study are:

- How does the interaction designer use the Interaction Language Design Patten (ILDP) to create a DSIL (DSIL) to personalize interaction?
- Is DSIL successful from the user's perspective?

5.3 Method

Initially, by producing a paper prototype we aim to answer the first part of the research question. To evaluate whether the paper prototype matches our design purpose we created a questionnaire to ask the user's opinion of operating the drawing system. The feedback and comments from user interviews are analyzed and used to develop a Hi-Fi prototype. The final Hi-Fi prototype is used for testing user' experiences regarding to the second research question. The investigation of the user experience of interaction is discussed in chapter six.

5.3.1 Building paper prototype of domain specific interaction language

In chapter four, we have discussed how to use ILDP to create an interaction language and realize personalized human computer interaction. In this chapter, we produce a paper prototype of an interaction language for a digital drawing system by using the ILDP.

5.3.1.1 Establishing interaction concept

Before creating a concrete interactive artefact, we usually have a definition statement - a concise, concrete declaration of purpose of the main interactive artefact, the interaction context and intended user's experience. It is the first step towards building a common ground for creating understandable interaction between human and computer. The domain specific interaction concept is created by the designers based on analysis and researching in a specific domain.

In this case, our interaction concept is to help the user easily and naturally to create drawings using a specific drawing system. The primary task is to set up interaction design concepts to build a particular interactive system for particular purposes. At this point, the focus is on creating a domain specific language for painting to let the user produce their intended interaction concept and establish common ground for mutual interaction. In the following section we will demonstrate how to build such specific interaction language and how the users are going to use it to personalize their interaction.

5.3.1.2 Building domain specific interaction language

In chapter four, we have indicated that the DSIL is designed to realize participants' interaction concepts. Accordingly, we have proposed a linguistic design framework to build a user-oriented DSIL through three steps: language production, interaction language perception and language pragmatics. In the following, we are going to construct a paper prototype to illustrate the development process by constructing a DSIL for the drawing system.

5.3.1.2.1 DSIL production

Initially, a designer needs to give an original design idea for a particular interactive product to set up basic interaction between a user and computer. The designers’ concept makes use of various interactive techniques. The DSIL is used by users to convey their intended meanings to the computer.

The overall goal of the interaction concept is to create an adaptive drawing system based on ILDP for diverse users with different background knowledge and skills in order to make drawing easy and effective. The interaction design process starts from:

- Clarifying the concept of creating a suitable drawing tool for diverse users.
- Designing to support primary tasks and sub tasks to let the users carry out actions whilst receiving meaningful feedback.
- Supporting users so they build up an individual relationship with the tool.

The vocabulary of the painting language has five dimensions:

- 1D vocabulary: text.
- 2D vocabulary: visual representation (picture, figures).
- 3D vocabulary: physical object/ space.
- 4D vocabulary: sound: music, animation.
- 5D vocabulary: human interaction activity.

In the following we will demonstrate how to build the drawing system by integrating the above dimensions. To test the basic concepts of our drawing system we use a ‘low-fi’ paper prototype.

To build the paper prototype of the drawing system, pattern languages are an important form of storing the basic design knowledge of human computer interaction (Dearden and Finlay, 2006) which help us construct reasonable interaction.

In this case, we utilize interaction design patterns created by Jenifer Tidwell (Tidwell, 2010). The other important design resource is Adobe Photoshop guideline https://www.ischool.utexas.edu/technology/tutorials/graphics/photoshop7/section4.html.

By using pattern language and design guideline, we created a paper prototype to explore how the drawing system would be used by the end users. To achieve this we draw on diverse types of interaction vocabulary, including visual representation (icon, button), information architecture (palette), structure human interaction activity (see Figure 5. 1 and Figure 5. 2).

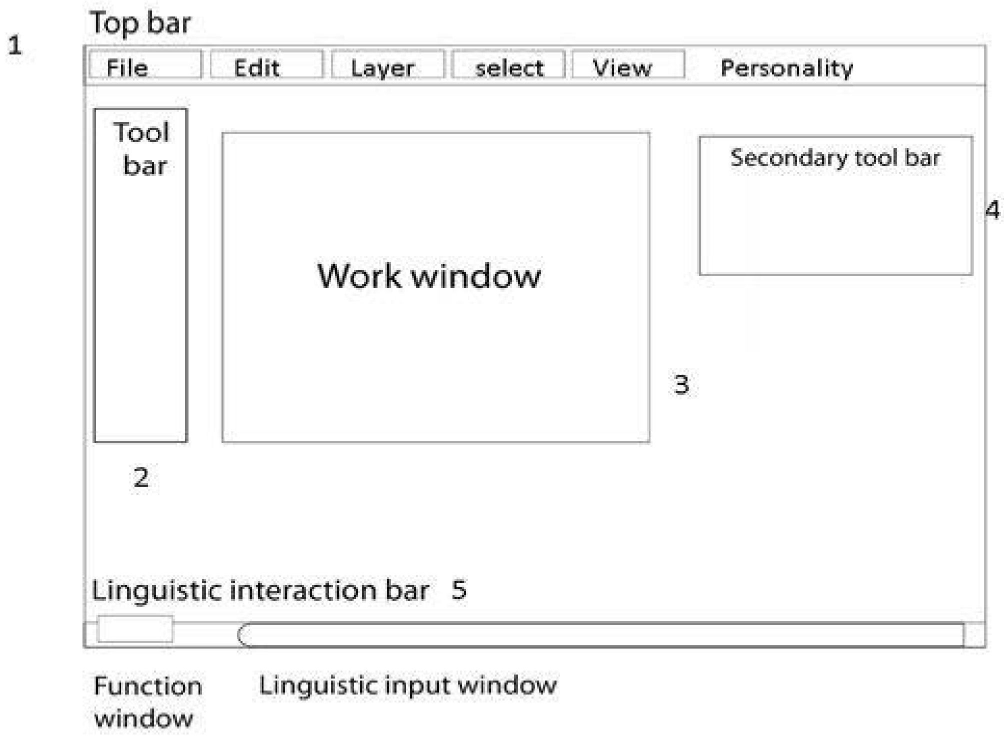


Figure 5. 1 Interface layout of the paper prototype of drawing system

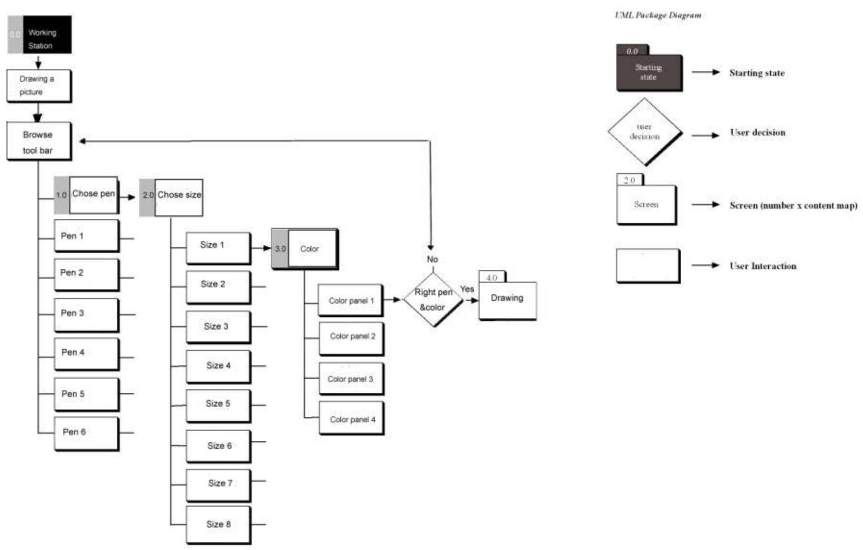


Figure 5. 2 Interaction architecture of completing a drawing task

On the reflective level, we use a DSIL to allow the user to link the text and the other types of interaction vocabulary together. In other words, the domain specific language uses painting terminology that has been used by the users. The painting language contains a domain dictionary, terms, concepts, their relations and properties. It gives us a basic scope of knowledge of painting and helps us to develop a drawing system in the painting domain. For example, the painting language contains the basic concepts of painting including point, line surface, color, space and so on. More importantly it built on a fundamental knowledge of how people to make paint in the real world and their relative drawing circumstances. Figure 5. 3shows some key words and icon used in our drawing domain referring to Adobe Photoshop guideline.

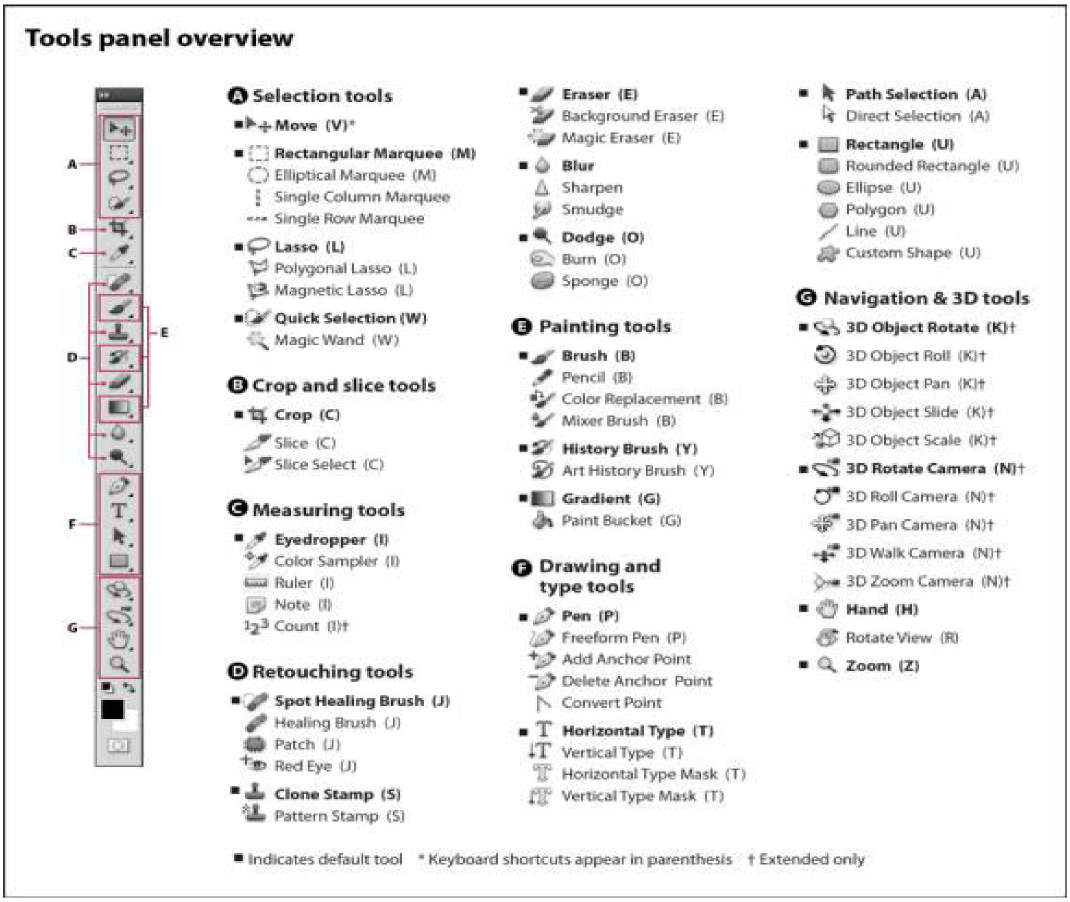


Figure 5. 3 Terminology of components for tool panel of Photoshop

(http://helpx.adobe.com/photoshop/using/tools.html)

The DSIL is a user-oriented language that allows users to construct their basic interactive semantics related to the painting tool using textual interaction. The text the users enter into the drawing system will affect the corresponding interface and interaction model. In doing so, the DSIL allows users to establish a mutual interaction with the drawing system based on their domain knowledge, interaction concepts, needs and purposes. Users utilize the DSIL to address different problem of using the system to painting and lead diverse users to develop the drawing system particular in a personal way.

5.3.1.2.2 DSIL perception

Once the user familiar with how to operate drawing system through a text-based interaction, it starts to move into the second stage that is a dialog phrase. In this stage, the outcome of the interaction reflects how the users understand the provided interactive artifact- the drawing system and its underlying interactive semantics. This process of perception takes place in three layers: physical, cognitive and reflective. The results of the language perception are based on the users' individual background knowledge and skills that shape how the user interacts with the drawing system. Additionally, we anticipate that the result of the interaction language perception will lead the users to react by semantically mapping out their interaction by using the domain specific interaction language.

5.3.1.2.3 DSIL pragmatics

The point of using DSIL is to allow users to define their interaction in an appropriate and natural way through a textual interaction method. To this end, we construct a linguistic input bar to allow a user to input a word or sentence.

On a practical level, a user develops his/her interactive semantics to cooperate with the computer in three stages: physical, cognitive and reflective. This process is called 'language pragmatics' and relates to how a particular user uses his/her domain knowledge to personalize the interaction. The user' interactions cover diverse aspects and levels of participants' communication with the computer.

In a situation where the users cannot find an appropriate function, or are struggling with operating the drawing tool, users are able to use text-based interactions to customize the application in order to achieve their purposes. The textual interaction allows users to personalize the interaction according to his/her personal experience. As a result, the interaction becomes more meaningful and effective.

We defined two types of interaction terminology to facilitate personalized interaction: ‘word for object-entered’ interaction concept and ‘word for experience-focused’ interactive semantics.

Script vocabulary links a variety of drawing tools and patterns of functionality. Action vocabulary relates to the attributes of a tool. For example, to draw a line, the user can type the word “pen” to get pen pallet (or chose a pen from a tool bar if the pen is currently shown on the interface). Additionally, the user can input the word “larger” to make a pallet for changing the pen size appear on the workplace screen. Thus, through text-based interaction the user chooses an appropriate tool with the right size to complete a drawing task (see Figure 5. 1Figure 5. 4).

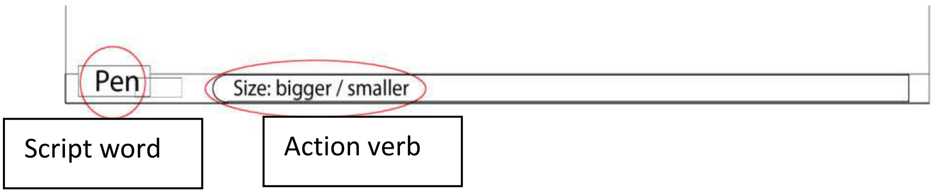


Figure 5. 4 Using interaction vocabulary to express interactive meaning

In the design phase, the interactive semantics are developed and produced using a linguistic architecture. After the product is delivered to the end users, it will start a second design lifecycle, as the use the text-based language to personalize the interaction.

In the next section, we will elaborate on how the DSIL supports users to make a drawing.

5.3.2 Adapting the drawing system to match a user’s intention

Originally, how a user work with the drawing system based on his/her knowledge of other drawing systems. As we know, users have different levels of domain knowledge for a particular domain. Sometimes users can easily figure out how to operate the interface and manipulate the drawing system easily. At other times however, the user may look through the interface but be unable to find the appropriate tool to complete his/her tasks. With a traditional graphical user interface the users have to go through the menu in order to find the right tool. In contrast, the prototype ILDP application enables users to locate the tool and make it work immediately by inputting a relevant word to the functional area (see Figure 5. 5). Using text-based interaction, the users can customize every functional element of the working area of interface by adding or removing elements.

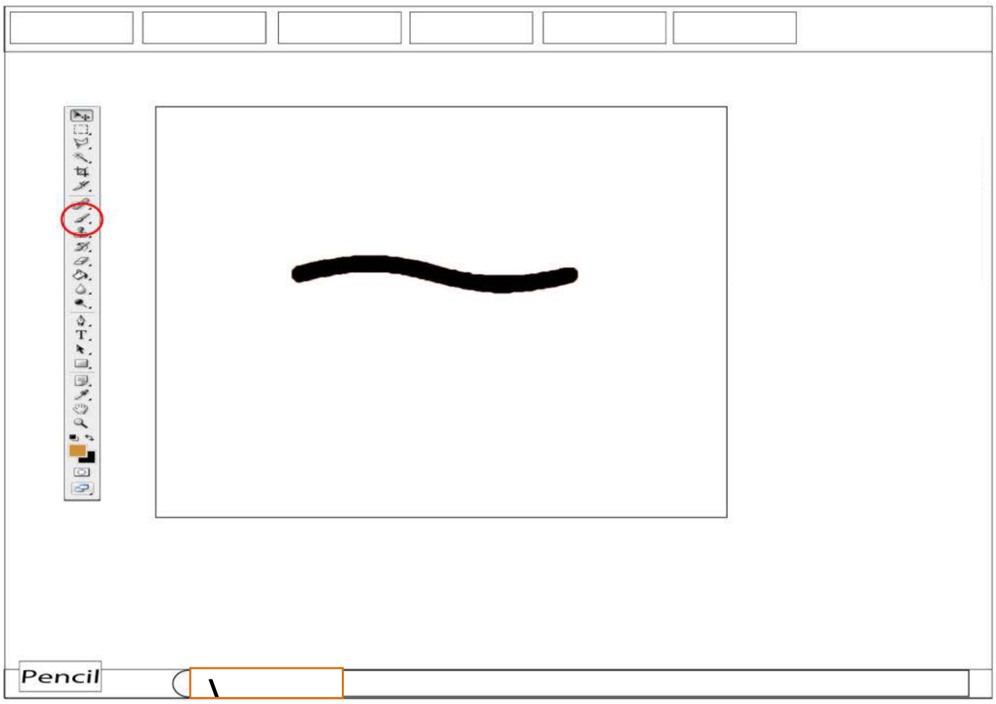


Figure 5. 5 Text-based interaction

As we mentioned before, this type of interaction based on an object-centered syntax of interaction. Learning how to use the vocabulary is the fist step towards generating user specific interaction semantics.

5.3.2.1 Semantic interface

A predefined semantic interface is the representation and structure of the visible (or audible) forms of a designer's syntax of interaction. The syntax rules or grammar of a language determines the basic tasks and the ways in which they have been organized by designers.

On the physical level, users mostly work on each separate meta-components and understanding of different elements related to the users' tasks or purposes (see fFigure 5. 6).

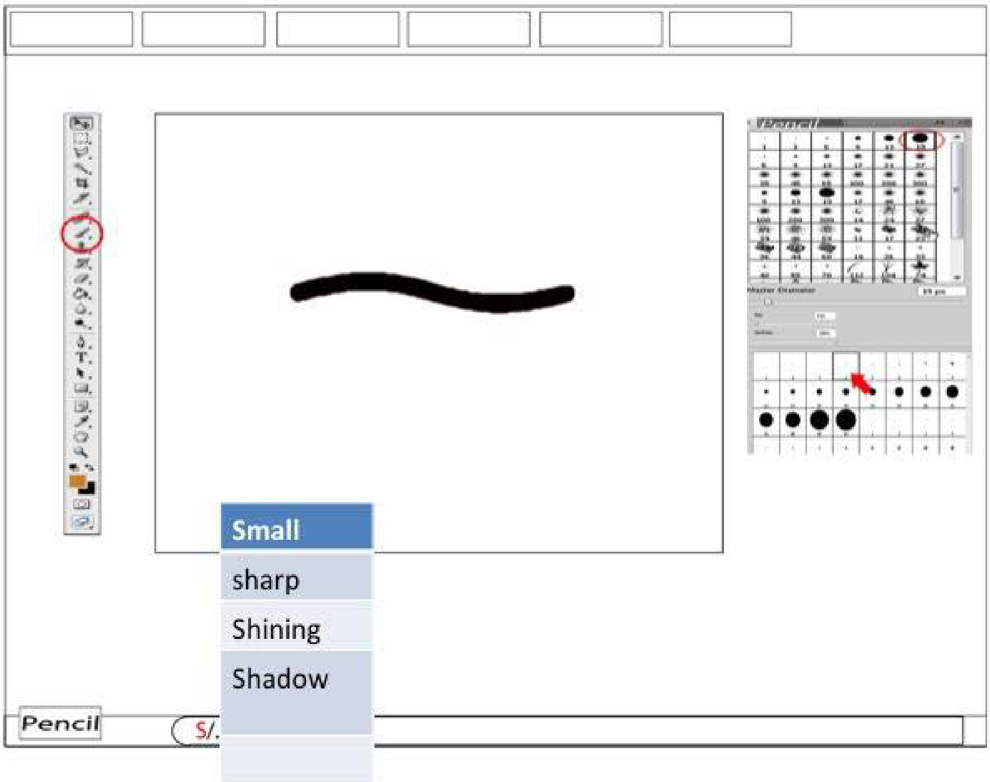


Figure 5. 6 Semantic interface generated from the text-based interaction

While using a product or system, the users may like to customize the system by making their interaction more appropriate in a personal way. Here, the users' knowledge and semantic interaction model enables the users to input specific words to redefine the interface. These words will trigger particular functions and display related information on the interface to help the user complete their task at hand.

On the cognitive level, the intention is that users establish a common ground between the interface for drawing composed by the designer and a space of potential meanings of their interaction.

For the end-users, the level of recognition of designer's interactive semantics depends on how well the users' understand and use the drawing system including classification, object, individual, relation, rule, and interaction). In addition, different users will have different reactions according to their classification spaces, perceptive ability and purpose of interaction.

The new interface is created through a constant conversation with computer through a meaningful interaction which depends on their interaction meaning.

5.3.2.2 Personalized interaction

On the affective level, a user-specified interaction language is produced through constant interaction with the drawing system through customizing the system. By using the semantic interaction, a user sketches his/her own interaction pattern language for specific tasks or purposes in a specific context related to his/her mental model (Norman, 1988). The intention is that a common ground interface for specific users' is gradually generated from a user's participation in the personalization process.

Users are able to systemize their interaction to complete their tasks or goals based on personal intention. For instance, the more confident users feel in modifying the interface, the easier it is for users to archive the goals they have.

As a result of personalizing the interface using the interaction language, the user's interaction pattern reflects and responds to the users' thinking, problems and experience in a dynamic process of interaction. In addition, the interaction allows users to map, complement and integrate existing elements of interaction including all available vocabulary like pictures, sounds, animations and interaction design patterns that are created by software engineers and interaction designers. As a result, the intention is that the interaction helps the user recognize the drawing system and explore their own relationship with the system.

5.4 Paper prototype study design

We create a questionnaire to carry out the paper prototype study in order to get feedback from users (Rogers et al., 2011). The questionnaire gathers data relating to the user's opinion of the paper prototype and evaluates the usability of the domain specific interaction language. This questionnaire has several purposes including asking users what they think of the prototype of ILDP; whether it does what they want; whether they like it; whether they had problems using the prototype; whether they want to use it again and what kind of things they want to change for the further development of the Hi-Fi prototype. The question is included in Appendix B.

5.4.1 Participants

Participants in the paper prototype study include three females and three males recruited from the university campus. The average age is 28 years old, ranging from 21 to 38 years of age. Two participants come from the design area they are familiar with drawing software. The others come from different areas with different backgrounds such as computer science, education and they do not have much experience of using digital drawing software.

5.4.2 Materials

5.4.2.1 Questionnaire

Each participant is asked nine questions. The questions range from a specific to general and it include many open questions in order to gather as many opinions as possible from the participants. The questions are shown in Table 5. 1.

1	What do you think about the drawing system when you want to make a drawing?
2	How did you use the drawing system to complete the tasks?
3	Do you find any problem or difficulty by using the drawing system? What were they?
4	What were the main things you like about the way of drawing (interaction)?

5	What were the main things you dislike about the way of drawing (interaction)?
6	Do you feel more creative while you drawing? For example it can derive your creative response?
7	Do you feel free or easy to achieve what you want to do with this drawing system? In other words, the system can understand what you want it to do.
8	Do you like the way of drawing?
9	Overall did you enjoy the experience of drawing?
10	Do you have any other comments?

Table 5. 1 Questionnaire used in the paper prototype study

5.4.2.2 Paper prototype

The paper prototypes, which have been shown in this chapter will be used to help the participants complete the questionnaire. The paper prototypes we have used for the case study are shown as following (see Figure 5. 7 to Figure 5. 12):

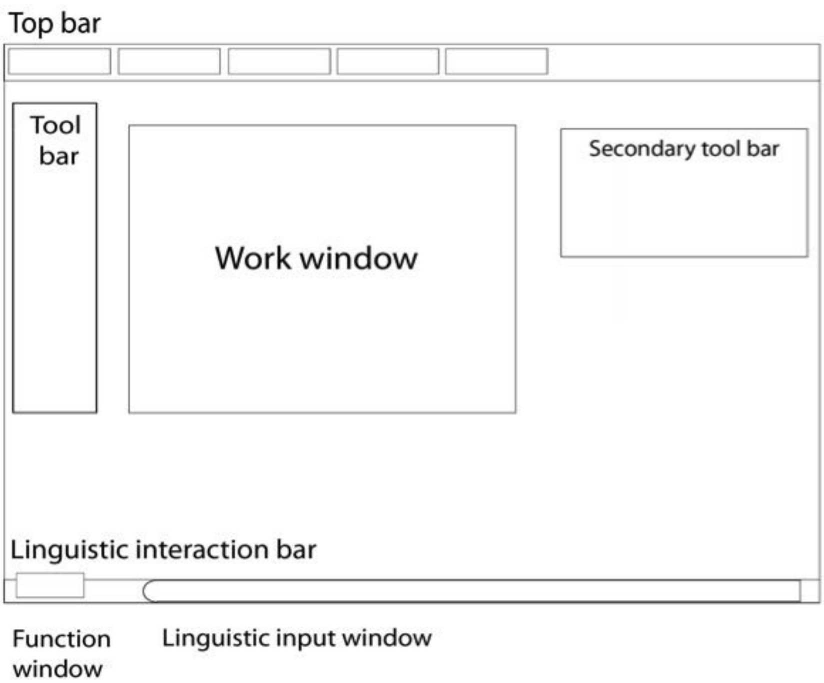


Figure 5. 7 Paper prototype 1

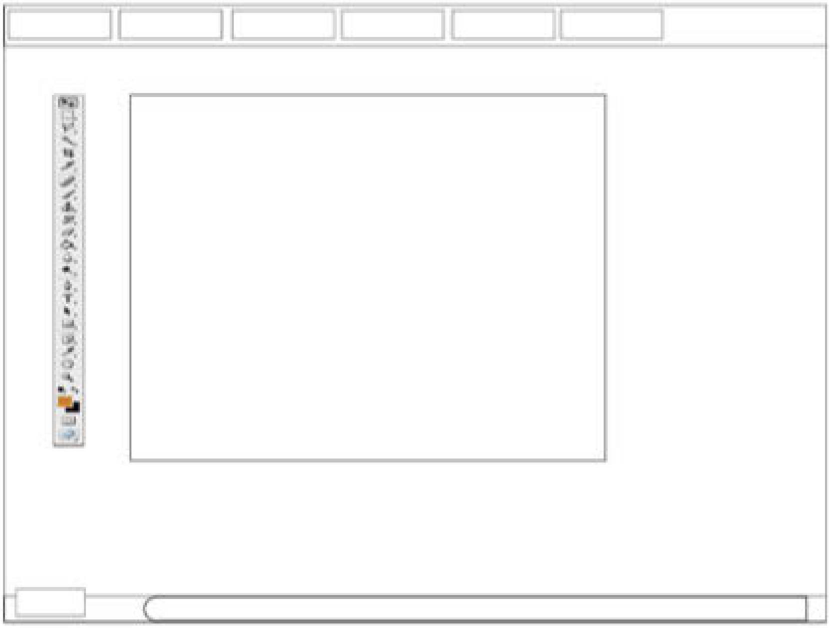


Figure 5. 8 Paper prototype 2

Figure 5. 9 Paper prototype 3

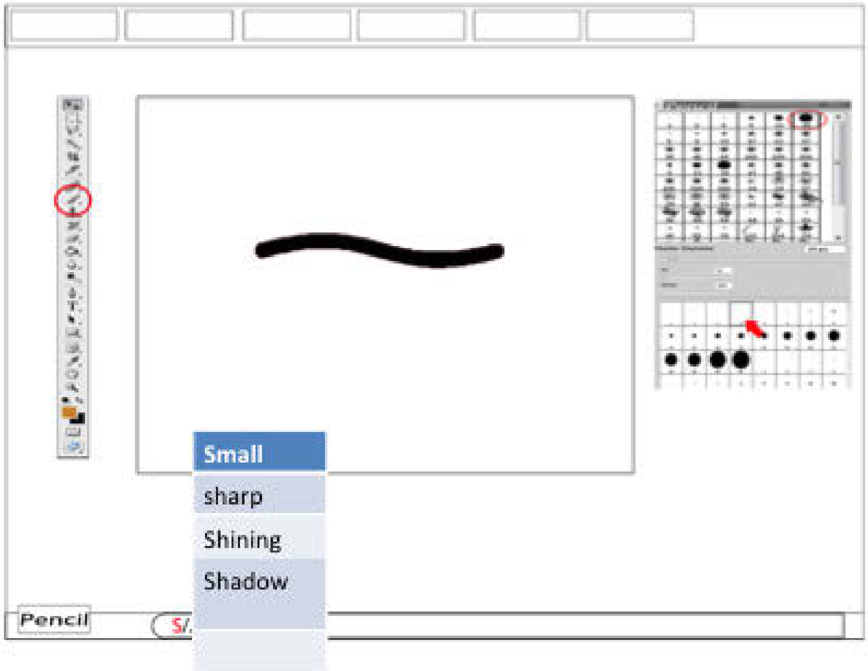


Figure 5. 10 Paper prototype 4

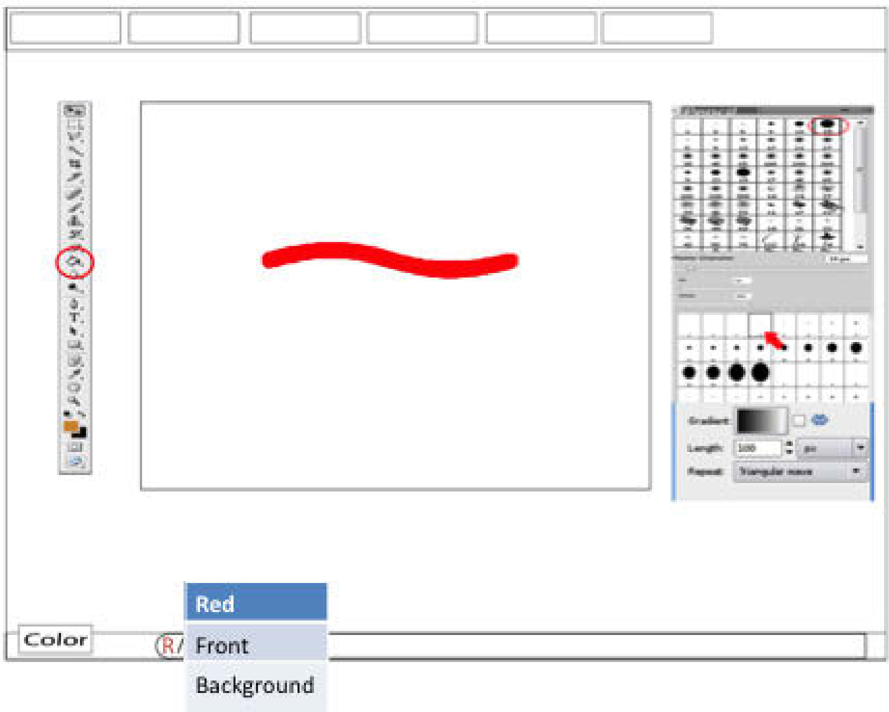


Figure 5. 11 Paper prototype 5

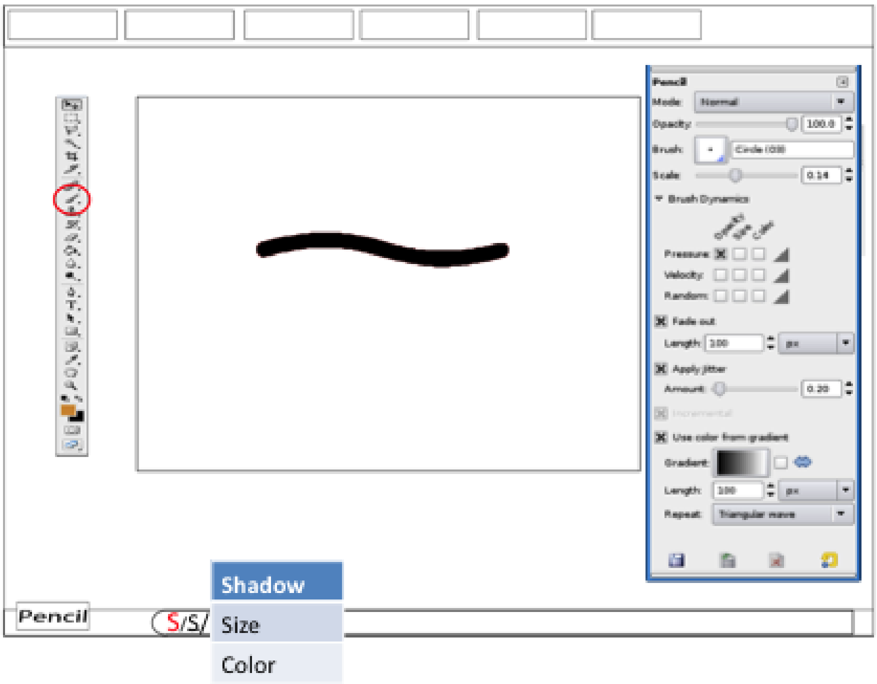


Figure 5. 12 Paper prototype 6

5.4.2.3 Tasks

We have established three simple tasks for the paper prototype study to explore the interactions between the participants and the drawing system through domain specific interaction language:

- Task one requires the user to draw a simple shape by using different drawing tools (pencil, pen, oil pen and crayon).
- Task two requires the user to draw multiple lines with different size.
- Task three is to draw different shapes (circle, rectangle and squire) with different color.

5.5 Usability study

5.5.1 Procedure

The study contains two parts the first part involved showing the paper prototype to the participant with a brief introduction. After that, we asked the participant to answer some questions based on his/her experiences. Each case took approximately twenty minutes in total: ten minutes for drawing tasks and ten minutes for completing the questionnaire. Before we start to do the survey, we gave participants a very brief introduction about the case study. In some cases, we extended the task time as some participants wanted to ask more questions of the prototype and needed more time to complete it.

5.5.2 Data preparation and coding

After completing all six participants' survey, we collected the data and analyzed the feedback and comments from every participant for each question. The details have been demonstrated in following tables.

Question one	What do you think about the drawing system when you want to make a drawing?					
Participants	Participant 1 (female, expert)	Participant 2 (female, novice)	Participant 3 (female, novice)	Participant 4 (male, novice)	Participant 5 (male, novice)	Participant 6 (male, expert)
Selected Original comments from the participants	"Interface is simple to use and I do need to spend long time to go through many items. So I concern that this drawing system is for me to	"It looks like a drawing board, which is easy for me start drawing directly."	"The drawing system is quite special because I did not find lots of button and menu on the interface."	"I am not sure how this drawing system is going to work and I would like to try it."	"I feel free to make drawing and I think it could let me drawing a much better picture that using other	"It doesn't have lots of function I can use so I cannot tell how I can make a better drawing. Not really like the way of

	draw something."				drawing system like Photoshop."	making drawing."

Table 5. 2 Comments of question one for paper prototype study

Question two	How did you use the drawing system to complete the tasks?					
Participants	Participant 1 (female, expert)	Participant 2 (female, novice)	Participant 3 (female, novice)	Participant 4 (male, novice)	Participant 5 (male, novice)	Participant 6 (male, expert)
Selected Original comments from the participants	"Simply find the appropriate tool by input the word to complete the task."	"Directly type the name of tool."	"Directly type the name to get tool and complete the task."	"Input the name of tool."	"Learning the function of linguistic input and figure out how to make drawing."	"Tying the word to get tool is cool, but he prefers using shortcut to get the tool."

Table 5. 3 Comments of question two for paper prototype study

Question three	Do you find any problem or difficulty by using the drawing system? What were they?

Participants	Participant 1 (female, expert)	Participant 2 (female, novice)	Participant 3 (female, novice)	Participant 4 (male, novice)	Participant 5 (male, novice)	Participant 6 (male, expert)
Selected Original comments from the participants	"No. I think this drawing system is good for the novices rather experts because the experts are more likely keep using the same system for a long period. "	"Interesting to use it, but the interface needs to be improved. "	"It is not difficult to control and learn. It can satisfy my requirements."	"There is no significant problem for using this drawing system."	"I am not sure what is difference for using different drawing system such as Photoshop, Illustrator, Freehand and so on."	"Interesting to see more functions for this drawing system."

Table 5. 4 Comments of question three for paper prototype study

Question four	What were the main things you like about the way of drawing (interaction)?

Participants	Participant 1 (female, expert)	Participant 2 (female, novice)	Participant 3 (female, novice)	Participant 4 (male, novice)	Participant 5 (male, novice)	Participant 6 (male, expert)
Selected Original comments from the participants	"It is easy to find a tool that I want to use."	"I can drawing as what I want to draw"	"Focusing on the drawing rather than learning how to use the system."	"I felt more creatively to draw."	"Focusing on the drawing rather than learning how to use the system."	"It is good for novice to use the system."

Table 5. 5 Comments of question four for paper prototype study

Question five	What were the main thing you dislike about the way of drawing (interaction)?					
Participants	Participant 1 (female, expert)	Participant 2 (female, novice)	Participant 3 (female, novice)	Participant 4 (male, novice)	Participant 5 (male, novice)	Participant 6 (male, expert)
Selected Original comments from the	"…Less function and not suitable for the	"Interface design need to be improved	"Functionalities are poor. It is little hard to	"Don't like using mouse to draw.	"I would like to have a personalized	"… a little bit confuse about the linguistic

the participants	professional painter or designer. I mean I like to make drawing using drawing board or sketch board. "	and I need more support and recommendation to find the word. "	understand the concept of the drawing system at the beginning."	Maybe it is better to create a drawing system cooperate with using drawing board. "	interface and functionality."	input bar and how it is going to work and support my design works…"

Table 5. 6 Comments of question five for paper prototype study

Question six	Do you feel more creative while you drawing? For example it can derive your creative responds?					
Participants	Participant 1 (female, expert)	Participant 2 (female, novice)	Participant 3 (female, novice)	Participant 4 (male, novice)	Participant 5 (male, novice)	Participant 6 (male, expert)
Selected Original comments from the participants	"Yes it is. I attempt to find a design tool that can inspire me when I need to	"Sure. It is very special and I can keep using it to draw a picture without	"I think I do not need to remember the entire menu and the location for different tool that is a big	"It gives some helps to make me feel more creative. "	"… sometimes it can be inspiring. "	"Not really. I prefer using Photoshop as I am very familiar."

	create something."	any trouble."	problem when I used other drawing system."			with it.

Table 5. 7 Comments of question six for paper prototype study

Question seven	Do you feel free or easy to achieve what you want to do with this drawing system? In other words, the system can understand what you want it to do.					
Participants	Participant 1 (female, expert)	Participant 2 (female, novice)	Participant 3 (female, novice)	Participant 4 (male, novice)	Participant 5 (male, novice)	Participant 6 (male, expert)
Selected Original comments from the participants	"Quite good. For me it is very easy to understand how to find the tool and make drawing."	"… It is very helpful and effective. It can save me lots of time to learn how to operate it."	"I like the way to interact with the system as it can give me what I want in a short time."	"Yes, I think so. It would be great if the system can be personalized."	"It is. However I do think the system understands what I want but it is very helpful to let drawing."	"I am not sure whether I can fully complicate the drawing based on what I think."

Table 5. 8 Comments of question seven for paper prototype study

Question eight	Do you like the way of drawing?					
Participants	Participant 1 (female, expert)	Participant 2 (female, novice)	Participant 3 (female, novice)	Participant 4 (male, novice)	Participant 5 (male, novice)	Participant 6 (male, expert)
Selected Original comments from the participants	"Yes, it is an easy way for me to drawing something."	"It quite impressive and I'd like to use it in the future. "	"It is a good way to make drawing, but I prefer using pen or pencil to paint."	"Don't like use mouse to draw…"	"… The convenience of controlling and expressing is main thing I like it."	"Not really, I like using electric drawing board to paint a picture."

Table 5. 9 Comments of question eight for paper prototype study

Question nine	Overall did you enjoy the experience of drawing?					
Participants	Participant 1 (female, expert)	Participant 2 (female, novice)	Participant 3 (female, novice)	Participant 4 (male, novice)	Participant 5 (male, novice)	Participant 6 (male, expert)
Selected Original comments	"I think so if the function	"Sure, I like drawing	"… It easy to learn and	"It can be more comfort	"… Would like to try	"I t may confuse sometime

from the participant s	are enough for me to drawing different types of painting. "	and the system can me feel more enjoyable to make drawing."	effective to inspire creativity."	able and efficient if I can use the real product. "	next version."	and it depends on what kind of work I am working on."

Table 5. 10 Comments of question nine for paper prototype study

Question ten	Do you have any other comments?					
Participant s	Particip ant 1 (female, expert)	Participa nt 2 (female, novice)	Participant 3 (female, novice)	Particip ant 4 (male, novice)	Participa nt 5 (male, novice)	Participa nt 6 (male, expert)
Selected Original comments from the participant s	"It would be better to persona lize the interfac e."	"I think this system need add more functions and make the operation becomes more freedom."	"I will use the system if its function can match all of my need to complete a complex drawing task. I like to customize	"… It is good to create more useful drawing patterns and record what I have done that I	"Good interactio n design but need works on developin g the system"	"… It is a good tool for designer to sketch an idea."

			the interface and function panel based on my needs.”	can easily to go back to make change. ”		

Table 5. 11 Comments of question ten for paper prototype study

5.5.3 Responses

The response of the using paper prototype based on the questionnaire shows the following aspects from user’s points:

- According to the result from the questions 4, 6, 7 and 9, we found that five of the six participants agreed that it is very useful of having a DSIL to accomplish a task, as it established a common ground of the interaction. Consequently, the interaction becomes more effective and natural to control the interactive artifact.

- Five of the six users believed that it is important to express their individual values and judgments during interacting with the system based on the participants’ comments from questions 3 and 5. In question 1, four of six participants agreed that the interaction matched their perspectives and background knowledge. Four of the six users said that they felt it is easy to manipulate the system through their interaction referring to the question 2 and 8.

- The results from question 10 also shown that five of the six users are willing to continue to use the drawing system to make a drawing when we asked the participants do they have any comments about using our prototype of drawing system.

5.5.4 User problem

There have some problems that have been found form this survey for the paper prototype.

Firstly, the paper prototype was not good enough for users to have a comprehensive understanding of how the interaction language might assist them. Most users indicated they were interested in trying the real function works a working system and hope to explore more possibilities to construct their own interaction language to make a drawing.

Secondly, some experts prefer to use traditional interaction methods, especially using shortcuts to control the system. We found that expert users of a particular system are accustomed to having it behave in a particular way, so it is difficult for them to change interaction model.

Thirdly, the task we had the participants complete was very simple and may not complex enough to fully explore the differences between the semantic interaction and traditional interaction model. So for the next stage, more complicated tasks are needed for further user testing. Furthermore, it would be useful to compare the prototype with other existing products.

5.6 Results

The paper prototype study for user specified interaction language resulted in useful feedback from potential users. On top of that, the results of the study show that the user's responses have approved our assumption that most of the participants are happy to have a tool help them to personalize interaction. In addition, the results of the survey help us to make a decision about how to develop a Hi-Fi prototype of the user specified interaction language and how to address some problems. Through the prototype evaluation, we have a deeper understanding of the user's requirements and confusions that become the starting point for developing a Hi-Fi prototype. In following section, we will utilize what we found from the survey to build our HI-Fi prototype and aim to fix the problems which imaged in the survey.

5.7 Chapter summary

In this chapter, we describe the use of the ILDP to create a paper and Hi-Fi prototype of user specific interaction language between human and computer. The collaborative interaction presents a comprehensible message to the user, as well as understanding people’s use of language. In using this approach the linguistic choices of designer will be driven by their communicative purposes and by their target users. This means ILDP helps designers to create content of interaction under a readable architecture in order to make it fully understandable by end user (5.3).

In addition we have described a user study which examined users’ experiences with the paper prototype.

Chapter Six: Users Experiences Study

Chapter six describes a user study examining users' experiences using different drawing systems. Participants in the study were asked to complete a set of drawing tasks using two different types of drawing systems: Photoshop CS5 and a Hi-Fi prototype based on the Interaction Language Design Pattern (ILDP). This user study focuses on exploring different user interaction experiences. The results show that DSIL can assist the user to make an appropriate and effective interaction with the computer.

6.1Rationale

In last chapter we have illustrated how to use ILDP to create a user specified interaction language helping users to manipulate interaction. In this chapter we aim to analyses how users experience different interactions by using two different drawing systems: Photoshop CS5 and the Hi-Fi prototype of ILDP.

To help understand how the users interact with the computer, we compare traditional interaction and our proposed interaction model- personalized interaction to study users' experience of interaction. The strategy for the user study is to do user testing on a same task by using different drawing system. This case study explores the user's experience of interaction with a product on three levels: physical, cognitive and reflective. It measures different aspects of interaction including ease of use, and effectiveness, efficiency, and satisfaction from user's perspective.

6.2 Research aims and questions

6.2.1Research aims

The specific research aims for the user study are as follows:

1. To compare existing human computer interaction approaches to the approaches described in the Interaction Linguistic Design Pattern. Asking a user to describe his/her experience of using two types of interaction including conventional interaction and personalized interaction that is generated from the user's individual perspective and aim.

2. To analyze the quality of human computer interaction at different levels: physical level, cognitive level and reflective level.

6.2.2 Research Questions

The specific research questions for this users study are:

- How does the user interact with two different drawing systems – one using a traditional design method and the other based on the ILDP?
- Does the DSIL support effective human-computer interaction?
- Does the user enjoy the resulting interaction by using ILDP? Was the use of the ILDP successful from this point of view?

6.3 Method

6.3.1 Hi-Fi prototype study design

To address the above aims we have designed a set of represented drawing tasks, which have been used in examining our paper prototype, which participants complete using two drawing tools: Photoshop CS5 and our Hi-Fi prototype of drawing tool in a certain time period.

We select Photoshop CS5 as a comparable example because of it is a powerful and widely used drawing tool which many people are familiar with. The other reason is PhotoshopCS5 is a good example which was designed using traditional interaction design framework based on a method of User-centered Design (UCD) (Bearden, 2011). The features of Photoshop CS5 includes a user friendly interface and user-centered situated interaction pattern (Evening, 2013).

For the evaluation, we measure target users' performance on our predefined tasks. In addition, we observe users and ask for their opinions as discussed in chapter 3.

The user tests are video (and audio) recorded. This allows the details of the users' behaviour and comments to be carefully examined. After recording the user testing, we have classified and analysed the data in order to clarify the usability and efficiency of the two systems.

We also use a questionnaire to elicit participants' opinion on their interaction - whether they like or dislike it. To evaluate users' emotional experience of interaction, we designed a user satisfaction questionnaire to address the user's experience of interaction at different levels.

6.3.2 Participant

We have recruited 30 participants from different fields (academy and industry) from different countries (Australia and China). The participants contain 16 females and 14 males. The average age of the participants is 25 and ranges from 21 to 38. The participants are divided into two groups, each group has 15 participants. The first group of participants has significant experience with digital drawing system and most of them have drawing knowledge background. In addition, 10 of participants are working or have been working in the IT industry or HCI area with average more than 1 year working experiences. They use drawing tools like Photoshop very often more 10 hours per week and they have lots of experience with the drawing software. The second group of participants has little experience with digital drawing systems and little knowledge about how to draw. Most of them have heard about Photoshop but never use it, or use Photoshop CS5 less than 5 times a year.

6.3.3 Material

6.3.3.1 Photoshop CS5

We use the Photoshop CS5 in our users study as this is a very successful digital drawing system and many of participants have heard and/or used it previously. As we know, Photoshop CS5 is a very powerful drawing tool， which satisfy most drawing requirements. Meanwhile, this software is widely used in design industry. The interface of Photoshop CS5 is shown in Figure 6. 1.

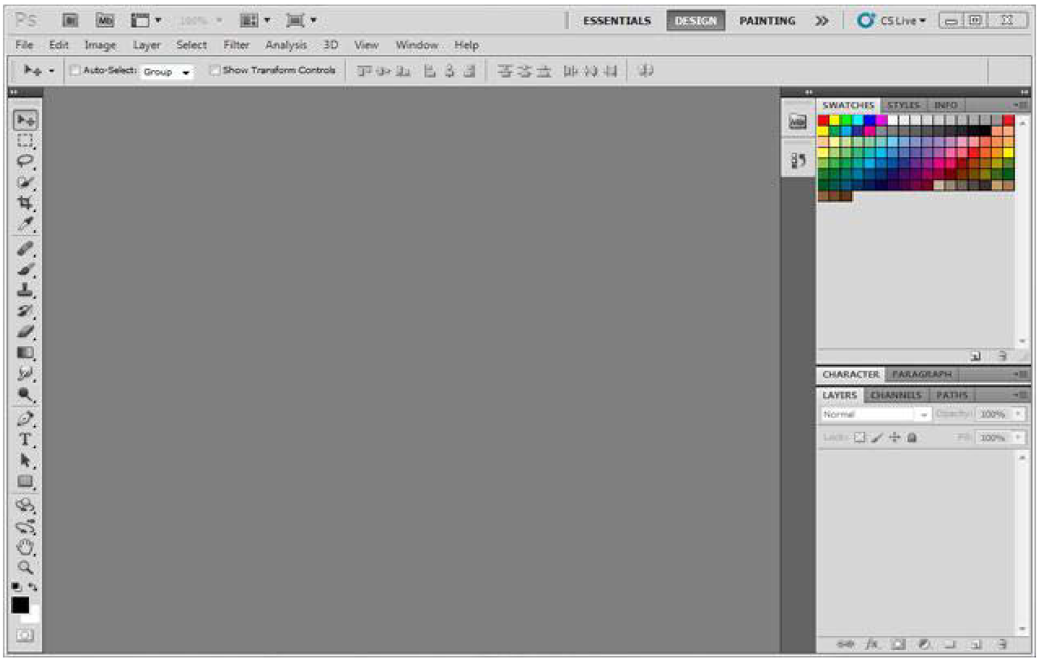


Figure 6. 1 Interface of Photoshop CS5

6.3.3.2 Hi-Fi prototype of DSIL for drawing system

In this section we will describe the Hi-Fi prototype.

6.3.3.2.1 DSIL production

In chapter four, we have mentioned that we are going to use three types of concrete syntaxes to build an interaction language based on conveying interaction semantics. This is achieved by combining three specific languages: programming language, pattern language and user-oriented domain specific language. In practice, different stakeholders work on building a particular concrete interaction entity composed of interface and interaction model based on their understanding of the basic interactive semantics. For instance, the drawing system has two essential interactive goals from the designers' perspective that are: usability goal of drawing and experience goal of drawing.

At the functional level, programming languages are used to build fundamental functions of the drawing system. We create some basic functions to let the user make drawings using different painting tools such as pen, pencil, brush, rubber, ruler, drawing board and so on. We create the functionality of the drawing system like pen, pencil, erasure, ruler using

JavaScript. In the same way, we have built different types of functionalities. These functions allow a user to effectively create drawing.

We use JavaScript and HTML5 to realize all the functions representing the various drawing-related tools such as pens, pencils, oil pens, brushes and other relevant objects in a web-based application. Figure 6. 2 shows a screenshot of the Hi-Fi prototype of the new drawing system.

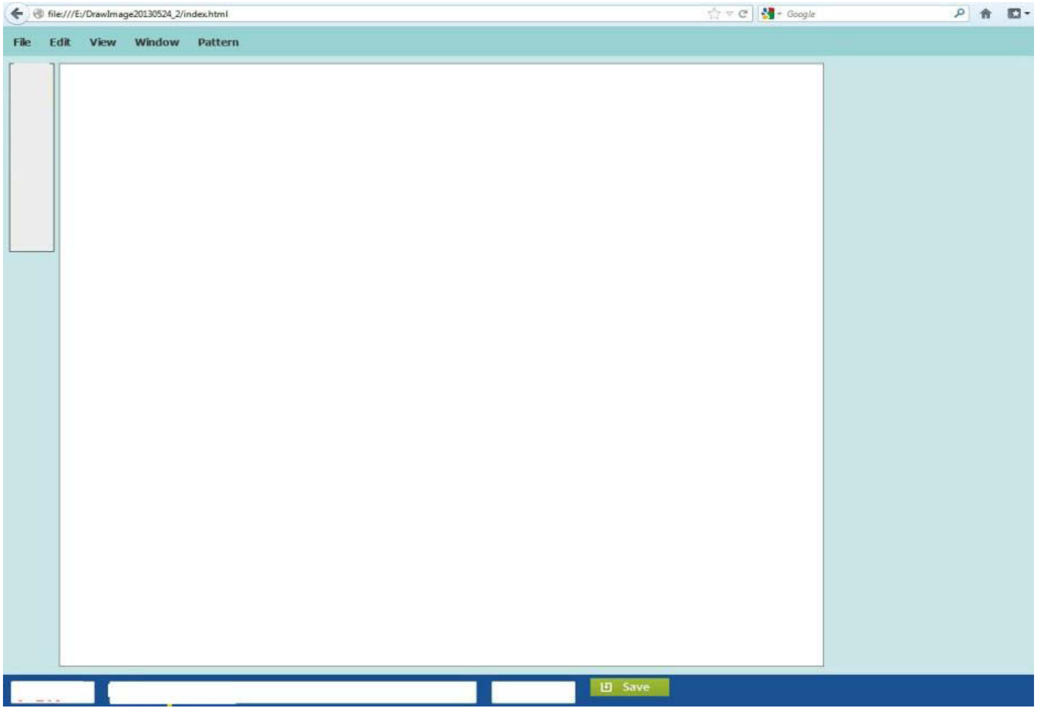


Figure 6. 2 Screenshot of the new drawing system Hi-Fi prototype

In addition, interaction designer maps out different drawing contexts with certain concrete syntaxes, here we use Photoshop CS5 guideline as pattern language to organize the various elements of interface to make the user's work effectively.

6.3.3.2.2 Interaction language pragmatic pattern

The domain specific language links the user's semantic values to the performance of the system. Key terms can be entered in a text box at the bottom of the screen and each of a word can trigger a unique function of the system and a particular interaction pattern (see Figure 6. 3). Moreover, when the user types a word to indicate his/her intended action, the system will accordingly provide a well-defined shortlist of options according to the structure of the domain language: vocabulary, syntax, semantic (see Figure 6. 4).

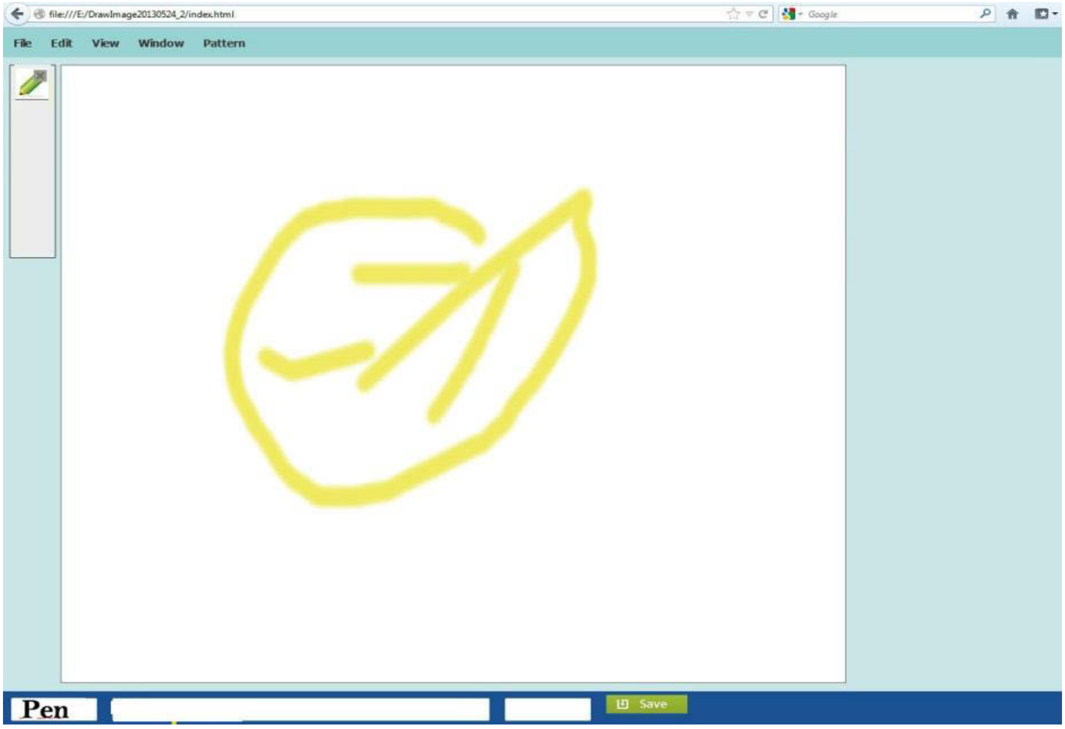


Figure 6. 3 User specified drawing interface through text-based interaction

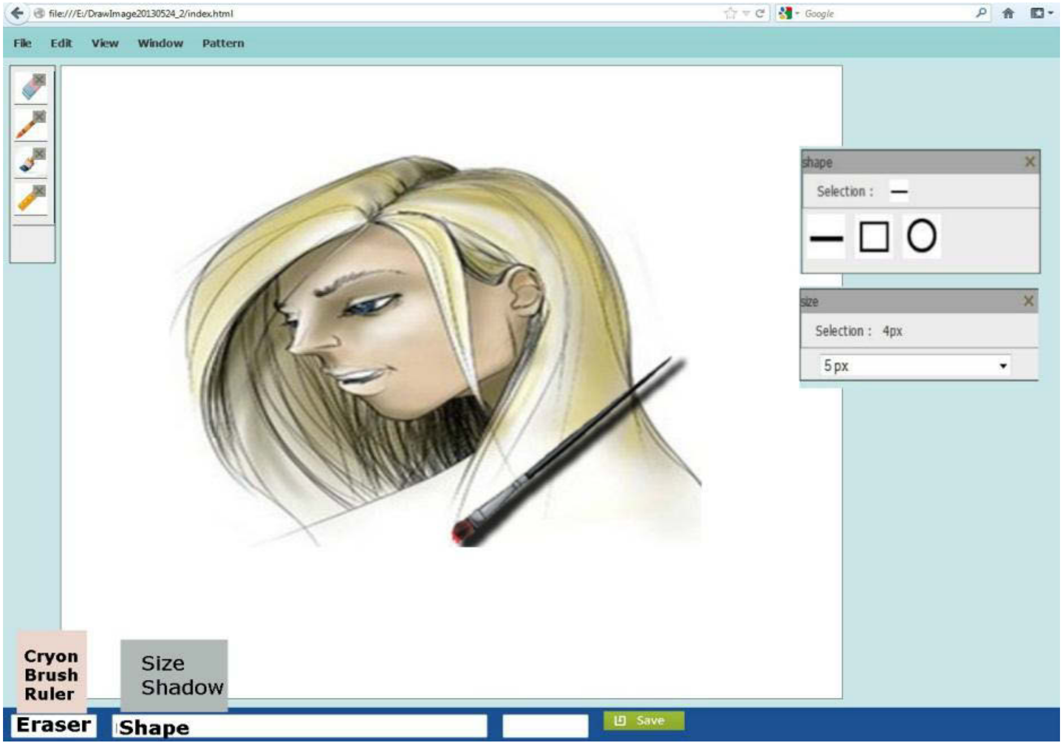


Figure 6. 4 Semantic interface generated based on user's need

Figure 5 Hi-Fi prototype of interaction language pragmatic pattern- user specified drawing interface through text-based interaction (screenshot)

So, when the user aims to personalize interaction, the text-based interaction allows the users to enter a word in order to control the operation of the system. All words are meaningful to the use and user can reorganize them based on his/her purposes, in this case related to the drawing domain. Initially, the words help users to operate the system using interaction language rather than looking through function menus and clicking buttons from a pointing device such as mouse.

When the user points any object such as a tool (represent as icon) or palette (title of object) to complete a specific task and then the user is able to make a personal interaction pattern that contains all the tools and palette have been used. Eventually, the interactive system keeps a complete archived record of all the words the user provides as input to generate a new interface dynamically (see Figure 6. 5). Moreover, the users can form a user's personal interaction language by defining and saving a particular set of tools and settings they have used in their personal profile (see Figure 6. 6).

This means that the users are able to change any object's properties to optimize the tools for their drawing. Through this personalized interaction pattern the users can increase productivity of the interaction and satisfaction through simple, language-based manipulation.

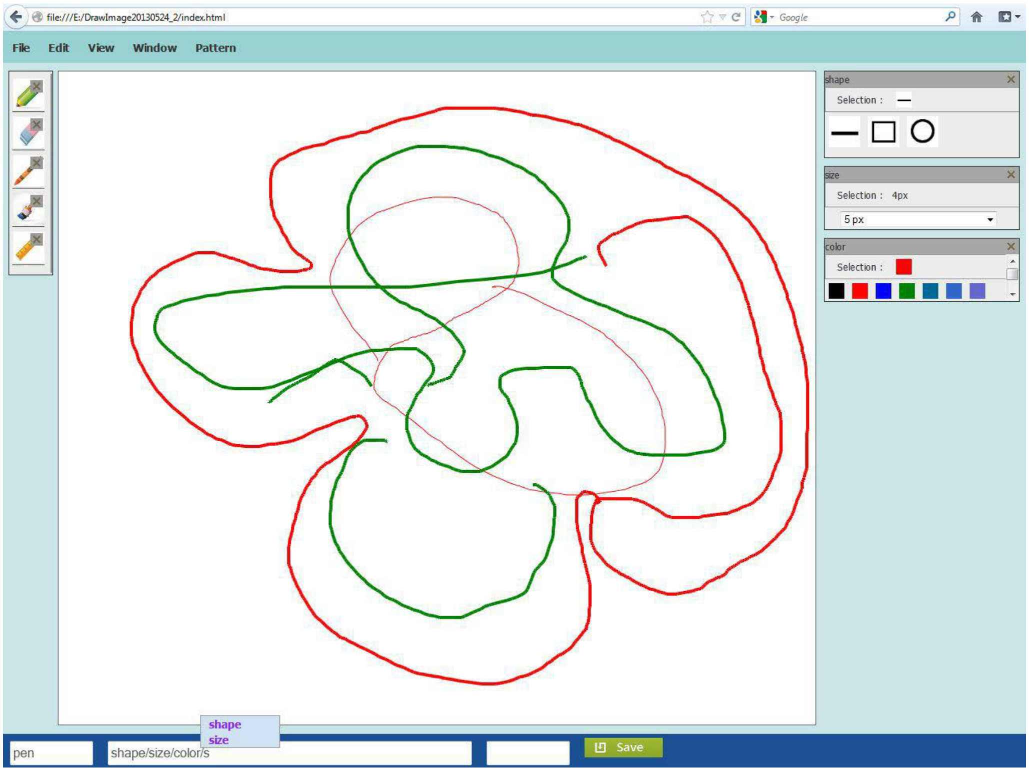


Figure 6. 5 A user personal interaction pattern

Figure 6 semantic interface generated based on the user's texted-based interaction

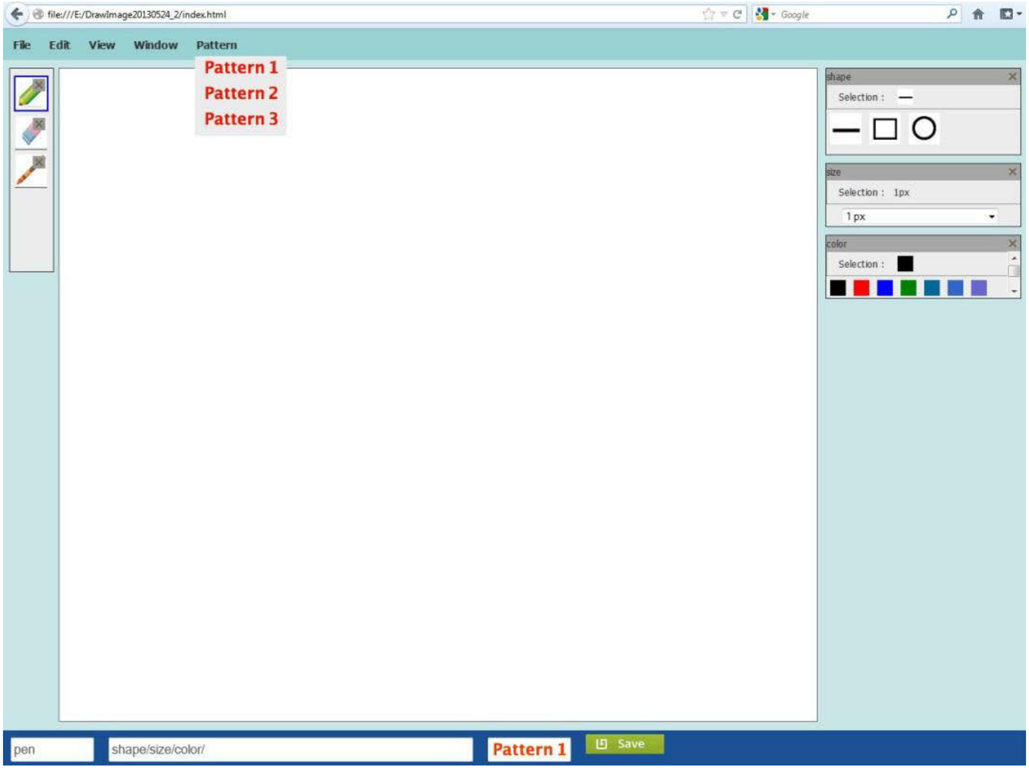


Figure 6. 6 Generating user interaction pattern language through language pragmatic

6.3.3.3 User satisfaction questionnaire

A user satisfaction questionnaire was created to analyses the users' experiences of interaction. The questionnaire is composed of three different parts corresponding to three aspects of the user interaction: physical, cognitive and affective. After completing the user testing we ask participants to complete the questionnaire. The full questionnaire can be found in appendix C.

6.3.4Tasks

First, the participants were required to use two digital drawing systems: Photoshop CS5 and then our Hi-Fi prototype to complete a set of drawing tasks. In order to make the result become more obvious we set up a time to complete every task. In using our HI-Fi prototype the participants can operate the system either through menu or typing the words to achieve their goals. The task contains three parts:

Task 1: Drawing a simple line with different painting tools including pen pencil and canyon

Description:

The first task for the participants is to draw a simple line by using different tools like pen, pencil and crayon. This task is required to be finished within 5 minutes.

Task 2: Drawing multiple lines in different sizes by using different painting tools

Description:

The participants need to use above tools to draw multiple lines in different sizes in the work window. This task is required to be finished within 5 minutes.

Task 3: Drawing different standard shapes include circle, rectangle and triangle with different colour (red, blue and yellow).

Description:

In the last task, the participants need to draw multiple shapes with different colours such as red, blue and yellow. The participants are free to choose the colours for different shapes. This task is required to be finished within 5 minutes.

6.3.5 User experience study

6.3.5.1 Procedure

The whole user study composes of four parts: introduction, user testing, answering questionnaire and interview. The user study will take about 1 hour to complete depending on the participants pre-existing skill levels.

For the questionnaire study, participants were given a brief introduction about the purpose of the research and ourselves. After this, the participants are required to accomplish the above three tasks by using Photoshop CS5 (15 minutes) and Hi-Fi prototype (15 minutes) within 30 minutes. After each user testing a questionnaire will be given the participants to complete.

5.3.5.2 Data analysis

In this users study case, we collected and analysed the data of the participants after they completed the tasks. We used three types of evaluation techniques to carry out our study that are observing users, user testing, and user satisfaction questionnaire.

6.3.5.2.1 Observing user

We observe and video record the process when the participants complete each task. The purpose of observing users is to examine the users' behavioural tendencies, the performance of the drawing system and the effectiveness of the system usage regarding to the time required for input operations.

To avoid disturbing the participants we did not talk to participant while they were completing the tasks unless the participants have problem with the tasks. However, did not answer questions relating to how to use the system such as telling the participants where the pen is located and how to make special effects of a tool. To collect and analyse large quantities of data from the users we classify the data into four key factors: task, main action, response to users' actions and rates of successful task completion within required time period.

6.3.5.2.2 User testing

In the user testing, we set up a three independent tests for the drawing tasks within fifteen minutes (five minutes each). The user testing is focused on usability and efficiency. Every step of user performances of two drawing systems was recorded by video. The results from each user test are given as following.

A. Results of Photoshop CS5 user testing

In Photoshop CS5, drawing objects are created by selecting the appropriate tool from the palette or tool bar and inserting the object into the drawing space; once created an object can be changed by manipulating its attributes such as size, shape, fill colour, etc. Some of these changes can be made directly such as drawing a line by using pen tool, some indirectly such as changing size of object, which requires the user to find the setting via menu options.

The result of user testing using Photoshop CS5 is summarised in the following tables.

Task	Scenario	Main action	Observations of users' actions	Successful completion of the task within the required period (5 minutes)
Drawing a simple line by using different tools like pen, pencil and crayon	Pen is shown on the tool bar, the others are not in the tool bar	Select tool by looking for the tool from interface and browsing menu	The other tools do not appear on the tool bar, the participants have to look for the tool from pallet or menu to get appropriate tools. So the user cannot directly make drawing. Ten participants get confused.	Expert group (13 / 15) Novice group (5/15)
Drawing multiple lines in different	The tool for changing the size does not	Looking through the interface to find the tool	The other tools do not appear on the tool bar, the participants	Expert group (12 / 15) Novice group

sizes by using different painting tools	appear on the interface		have to look for the tool from pallet or menu to get appropriate tools. So users cannot directly make drawing. Seven participants get confused.	(8/15)
Drawing different standard shapes including circle, rectangle and triangle with different colours (red, blue and yellow).	Standard shape tool is shown on the tool bar. The other shapes are not in the tool bar. The colour pallet appears on the interface.	Looking for the tool from tool bar and menu. Select colour from colour palette	The tool for different shapes does not appear on the toolbar and pallet, so user cannot directly make the change. Nine participants get confused.	Expert group (12 / 15) Novice group (6/15)

Table 6. 1 Results of user test using Photoshop CS5

B. Results of Hi-Fi prototype user testing

The results of analysing of the participants using Hi-Fi prototype is listed in the Table 6. 2.

Task	Scenario	Main action	Observations of users' actions	Successful completion of task within 5 minutes
Drawing a simple line by using different tools like pen, pencil and crayon	Pen is shown on the tool bar, the others are not in the tool bar	Select a linguistic input tool. Input the word of the tool The tool is appear on the palate, so user can directly make change and select tool from interface	Using the linguistic input bar to get the appropriate tool. The participants can directly make drawing	Expert group (15 / 15) Novice group (15/15)
Drawing multiple lines in different sizes by using different painting tools.	The tool to change the size does not appear on the interface	Input the name of the tool and select tool from interface.	Using the linguistic input bar to get the appropriate tool. The participants can directly make drawing. Only one participant find out the tool what he want using dropdown	Expert group (15 / 15) Novice group (14/15)

			menu.	
Drawing different standard shapes with different colour (red, blue and yellow).	Standard shape tool is shown on the tool bar. The colour pallet appears on the interface.	Inputting the word of the tool and select appropriate tool from interface.	Using the linguistic input bar to get the appropriate tool. The participants can directly make change.	Expert group (15 / 15) Novice group (14/15)

Table 6. 2 Results of user testing using Hi-Fi prototype

6.3.5.2.3 User satisfaction questionnaire study

Analysis user's experience

Following the usability testing, we employed a user satisfaction questionnaire to analysee the participants' experiences of the interaction. This questionnaire, based on a user satisfaction questionnaire (Harper and Norman, 1993), is composed of three sections which focus on participants' experiences at different levels: usability, efficiency and emotional. To measure subjective feelings, the participants complete a self-assessment questionnaire consisting of a 5-point choice for each question, which is based on the dimensions of valence and arousal (Russell et al., 1989).

Usability experience of drawing system

Questions	Photoshop CS5		Hi-Fi prototype	
	Experts	Novice	Experts	Novice

1.1 I can effectively make drawing Strongly disagree = 1 Disagree = 2 Neutral = 3 Agree = 4 Strongly agree = 5	Average response to Q1 was 3.66. Overall, 11 of 15 agree or strongly agree.	Average response to Q1 was 2. Overall, 6 of 15 agree or strongly agree.	Average response to Q1 was 4.4. Overall, 13 of 15 agree or strongly agree.	Average response to Q1 was 4.7. Overall, 14 of 15 agree or strongly agree.
1.2 It was easy to learn how to make a drawing Strongly disagree = 1 Disagree = 2 Neutral = 3 Agree = 4 Strongly agree = 5	Average response to Q2 was 3. Overall, 9 of 15 agree.	Average response to Q2 was 2. Overall, 3 of 15 agree or strongly agree.	Average response to Q2 was 4. Overall, 12 of 15 agree or strongly agree.	Average response to Q2 was 4.7. Overall, 14 of 15 agree or strongly agree.
1.3 I found it difficult to find the tools I needed Strongly disagree = 1 Disagree = 2 Neutral = 3 Agree = 4 Strongly agree = 5	Average response to Q3 was 2. Overall, 6 of 15 agree or strongly agree.	Average response to Q3 was 4.7. Overall, 14 of 15 agree or strongly agree. (14 of 15 novices agree)	Average response to Q3 was 0.7. Overall, 2 of 15 agree or strongly agree. (2 of 15 experts agree)	Average response to Q3 was 0. Overall, 0 of 15 agree or strongly agree.

1.4 The interface of this system is pleasant Strongly disagree = 1 Disagree = 2 Neutral = 3 Agree = 4 Strongly agree = 5	Average response to Q4 was 3.7. Overall, 11 of 15 agree or strongly agree.	Average response to Q4 was 2.7. Overall, 8 of 15 agree or strongly agree.	Average response to Q4 was 3. Overall, 9 of 15 agree or strongly agree.	Average response to Q4 was 4. Overall, 12 of 15 agree or strongly agree.
1.5 The system has all the functions capabilities I expect it to have Strongly disagree = 1 Disagree = 2 Neutral = 3 Agree = 4 Strongly agree = 5	Average response to Q5 was 4.3. Overall, 13 of 15 agree or strongly agree.	Average response to Q5 was 2.66. Overall, 8 of 15 agree or strongly agree.	Average response to Q5 was 3.3. Overall, 10 of 15 agree or strongly agree.	Average response to Q5 was 4. Overall, 12 of 15 agree or strongly agree.

Table 6. 3 The result of usability experience of using Photoshop CS5 and Hi-Fi prototype

Cognitive experience of drawing system

Questions	Photoshop CS5		Hi-Fi prototype	
	Experts	Novice	Experts	Novice
2.1 It was uncomfortable making a drawing Strongly disagree = 1	Average response to Q1 was 2. Overall, 6 of 15 agree or	Average response to Q1 was 4. Overall, 13 of 15 agree	Average response to Q1 was 3. Overall, 9 of 15 agree or	Average response to Q1 was 0. Overall, 15 of 15

Disagree = 2 Neutral = 3 Agree = 4 Strongly agree = 5	strongly agree.	or strongly agree.	strongly agree.	disagree or strongly disagree.
2.2 I felt more creative while drawing Strongly disagree = 1 Disagree = 2 Neutral = 3 Agree = 4 Strongly agree = 5	Average response to Q2 was 3.7. Overall, 11 of 15 agree or strongly agree.	Average response to Q2 was 1.3. Overall, 4 of 15 agree or strongly agree.	Average response to Q2 was 3. Overall, 9 of 15 agree or strongly agree.	Average response to Q2 was 4.3. Overall, 13 of 15 agree or strongly agree.
2.3 It was easy to complete the tasks Strongly disagree = 1 Disagree = 2 Neutral = 3 Agree = 4 Strongly agree = 5	Average response to Q3 was 3.7. Overall, 11 of 15 agree or strongly agree.	Average response to Q3 was 1.3. Overall, 4 of 15 agree or strongly agree.	Average response to Q3 was 4. Overall, 12 of 15 agree or strongly agree.	Average response to Q3 was 4.7. Overall, 14 of 15 agree or strongly agree.
2.4 It took too long to complete the task Strongly disagree = 1 Disagree = 2 Neutral = 3	Average response to Q4 was 1. Overall, 3 of 15 agree or strongly	Average response to Q4 was 4.3. Overall, 13 of 15 agree or strongly	Average response to Q4 was 0. Overall, 0 of 15 agree or strongly	Average response to Q4 was 0. Overall, 15 of 15 agree or strongly

Agree = 4 Strongly agree = 5	agree.	agree.	agree. 0%	agree.
2.5 I found the interaction attractiv Strongly disagree = 1 Disagree = 2 Neutral = 3 Agree = 4 Strongly agree = 5	Average response to Q5 was 3.3. Overall, 10 of 15 agree or strongly agree.	Average response to Q5 was 1.3. Overall, 4 of 15 agree or strongly agree.	Average response to Q5 was 3.3. Overall, 10 of 15 agree or strongly agree.	Average response to Q5 was 4.3. Overall, 13 of 15 agree or strongly agree.
2.6 I like this way of drawing Strongly disagree = 1 Disagree = 2 Neutral = 3 Agree = 4 Strongly agree = 5	Average response to Q6 was 3. Overall, 9 of 15 agree or strongly agree.	Average response to Q6 was 1.3. Overall, 4 of 15 agree or strongly agree.	Average response to Q6 was 3.6. Overall, 11 of 15 agree or strongly agree.	Average response to Q6 was 4.7. Overall, 14 of 15 agree or strongly agree.

Table 6. 4 Results of cognitive experience of using Photoshop CS5 and Hi-Fi prototype

Emotional experience of drawing system

Questions	Photoshop CS5		Hi-Fi prototype	
	Experts	Novice	Experts	Novice
3.1 Using the program was enjoyable. Strongly disagree = 1 Disagree = 2	Average response to Q1 was 3.3. Overall, 10 of 15 agree or	Average response to Q1 was 1. Overall, 3 of 15 agree	Average response to Q1 was 3.3. Overall, 10 of 15 agree or	Average response to Q1 was 4. Overall, 12 of 15 agree

Neutral = 3 Agree = 4 Strongly agree = 5	strongly agree.	or strongly agree.	strongly agree.	or strongly agree.
3.2 It was difficult to express myself using this program Strongly disagree = 1 Disagree = 2 Neutral = 3 Agree = 4 Strongly agree = 5	Average response to Q2 was 2.3. Overall, 7 of 15 agree or strongly agree.	Average response to Q2 was 4.3. Overall, 13 of 15 agree or strongly agree.	Average response to Q2 was 1.6. Overall, 5 of 15 agree or strongly agree.	Average response to Q2 was 0. Overall, 15 of 15 agree or strongly agree.0% (0 of 15 novices agree)
3.3 Drawing with the program felt 'natural' Strongly disagree = 1 Disagree = 2 Neutral = 3 Agree = 4 Strongly agree = 5	Average response to Q3 was 2. Overall, 6 of 15 agree or strongly agree.	Average response to Q3 was 1.2. Overall, 2 of 15 agree or strongly agree.	Average response to Q3 was 3.7. Overall, 11 of 15 agree or strongly agree.	Average response to Q3 was 4.3. Overall, 13 of 15 agree or strongly agree.
3.4 What is your overall impression of the value of this program? Very Worthless = 1	Average response to Q4 was 2.3. Overall, 7 of 15 feel	Average response to Q4 was 0.7. Overall, 2 of 15 feel	Average response to Q4 was 1. Overall, 7 of 15 feel	Average response to Q4 was 5. Overall, 15 of 15 feel

Worthless = 2 Neutral = 3 Precious = 4 Very Precious = 5	precious or very precious.	precious or very precious.	precious or very precious. (3 of 15 experts feel worthless)	precious or very precious.)
3.5 This program is novel Strongly disagree = 1 Disagree = 2 Neutral = 3 Agree = 4 Strongly agree = 5	Average response to Q5 was 3.3. Overall, 10 of 15 agree or strongly agree.	Average response to Q5 was 3. Overall, 9 of 15 agree or strongly agree.	Average response to Q5 was 3.7. Overall, 11 of 15 agree or strongly agree.	Average response to Q5 was 4.7. Overall, 14 of 15 agree or strongly agree.

Table 6. 5 Results of emotional experience of using Photoshop CS5 and Hi-Fi prototype

6.3.6 Discussion

From observing the participants using two drawing systems to accomplish the drawing tasks different interaction patterns were found for the Photoshop CS5 and the Hi-Fi prototype. The Hi-Fi prototype received more positive responses and participants using it completed the tasks with a higher success rate. In contrast, Photoshop CS5 received more negative responses and fewer users successfully completed the tasks.

Comparing the two systems with different interaction patterns, users found that inputting word to express their intention is effective and easy to use.

It is worth noting that Photoshop CS5 provides a powerful function bar to help the user finding a right tool, which is similar to our linguistic input bar. However, the fundamental difference is that help function of Photoshop CS5 just offering a path to show where the user can find the tool. In contrast, the linguistic interaction make the interaction become more natural and meaningful to the user. More importantly, the user can continue changing the attributes of the tool using the linguistic input bar.

Based on our observations, we found that the user specified interaction language though a semantic interaction could improve the collaboration between the user and computer. This consequence is also reflected in the result of user experience of interaction.

In the user experience study, the HI-Fi prototype was experienced as more positive on all appraisal dimensions than the Photoshop CS5. For the usability experience of using the drawing system, the Hi-Fi prototype is intuitive for user to manipulate and learn. These results apply to both expert (80%) and novice (90%). This means that the Hi-Fi prototype is more effective at supporting the kinds of tasks in the usability test. This is an essential step to drive the user to construct a positive cognitive experience and emotional experience as the user's interactions derive appropriate functionality and meaningful interface. For the user's cognitive experience, the result of questionnaire study shows that the more a user is able to explore a system, the more they can complete a task creatively and effectively. That means the interaction becomes to a balanced conversation and collaborative interaction is emerged. We can see more than 80 percent participants have shown this intention such as the participants like the way of interaction and enjoy completing their tasks. From this perspective, the key point for designing an interaction is to support the user to produce personalized interaction to make communicate with artefact.

6.4 Result

The results of the user study can be grouped into two factors. The first factor is to clarify the problems of the existing human computer interaction that are generated by using current design methods. The second aspect focuses on analyzing the quality of the interaction that the Hi-Fi prototype of ILDP provided on different levels, including physical, cognitive and reflective.

For the first factor, we claim that current interaction design approaches such as human-cantered and system-centred are not always effective at supporting the user to solve their interaction problems. Based on the results of this study, we have shown that there is a need to consider alternative ways of designing for human computer interaction.

The second result from the user experience study is to show that the proposed design method- ILDP can provide a meaningful and effective interaction between users and a particular interactive product by allowing the user to personalize the interaction. The observations of user testing showed how the application of a DSIL can help address some problems in interaction design.

Alternatively, some of points have been highlighted:

1. The data collected from user experience analysis shows that the interaction provided by our Hi-Fi prototype is more suitable for the user to realize his/her values and work more effectively, at least for some tasks.
2. As we saw in the study, the use of ILDP provides a technique for developing a personalize interaction model in human computer interaction as the user's value judgments are fulfilled by modifying the interaction with the computer in a personal way.
3. Through the interaction language, user-oriented interaction can be constructed based on the user's personal interaction experiences.

6.5 Chapter summary

In this chapter we presented the user experience study, which is composed of Hi-Fi prototype testing and user satisfaction questionnaire study. Through the user experience study we identified some problems of human computer interaction, and the possibilities of using the ILDP to personalize interaction.

To build such common language, we argued that the interaction between the users and computers needs to be built on a deeper model that can adapts each user's intentions and history in order to better serve the user's needs and to eliminate unnecessary repetitive activities in an unobtrusive way. In this case, with the textual based interaction, the user can manage knowledge and understanding of what the computer can do for them. Meanwhile, the technology covering the forms of group abstraction can be a foundation for creating an interaction language, which can be used for the realization of system communications.

The results of the user experience study show that on the physical level, the users gain a satisfying usability experience when using text-based interaction. On the cognitive level, the result shows that personalised interaction can be a more balanced and 'natural' way to carry out human-cantered interaction. For the emotional experience, an enjoyable user experience can be achieved by personalizing the interaction using the domain specific interaction language.

Chapter 7: Conclusion

This chapter summarises the contribution of the research work and presents some future challenges for designing human computer interaction. Initially, we argue the Interaction Language Design Pattern provides a way to establish effective human-computer interaction where the computer responds to the user's personal requirements in their particular situation. The intention is that the interaction reflects the end users' perspective and relies on their background knowledge and ability rather than just being based on pre-designed interaction technology and forms. As we have shown, there is potential for this type of interaction to provide more effective interaction which users perceive as 'natural' than other types of interaction. The main challenge for our future work is to build a more powerful DSIL to optimize the interaction between user and computer.

7.1 Overview

Based on the literature review, we indicated that there is potential to users' coordination of attention as well as action by supporting multiple levels of conversation. That means the interface between human and computer not only conveys the functionality of the product, but also responds to the user's developing personal reflection and experience. In this case, HCI design is considered as an inherently communicative practice (Andreev, 2001) where design activities enact the creation of interaction that supports effective human computer conversation.

To achieve this objective, it is necessary for users to no longer passively adapt to new technology but actively re-shape and utilize the technology to fulfil their own needs (Beaudouin-Lafon, 2004). More and more researchers and practitioners agreed that HCI should build the interaction focusing on building a user-oriented interaction.

The challenge in creating such interactions is not only making features available to people, but more importantly to provide, the right thing at the right time in the right way (Fischer, 2001). As a result, interaction designers are increasingly finding themselves going beyond the creation of usable interactive systems to provide the user of the computer with great experiences (Lim et al., 2008). McCarthy and Wright argue that user experience is generated from the involved totality of people acting, sensing, thinking, feeling and meaning-making including their perception and sensation of the artifact in specific contexts. In addition, they point out that designing for user experience needs to take a holistic approach. They further argue that the user's experience has to be understood as a whole and cannot be broken down into their constituent parts because experience lies in the relations between the parts (McCarthy and Wright, 2004).

From the above perspectives, the creation of desirable user experiences requires careful consideration of the entirety of user needs during the interaction to make the interaction memorable, satisfying, enjoyable and rewarding.

To support this, in this thesis we have proposed a design pattern for creating interactive artefacts, which provide desirable user experience on three levels: physical, cognitive and emotional. In other words, the design method helps designers create products, which match diverse user purposes (such as working, thinking, communicating, learning, critiquing, explaining, arguing, debating, observing, deciding, calculating, simulating, and designing). The LIDP that is built based on the idea of concerning human computer interaction as producing common ground between diverse participators (computer and users) and covers different aspects and levels of communication, built on various types of communicative activities. To produce such interaction designer need to provide a DSIL for the user as a communication tool to make the interaction become more effective and sustainable. In other word, the DSIL is not only driving the designers to create a useful product but more important to support the end user to make the interaction becomes more effective and optimize.

The design method- the Interaction Language Design Pattern (ILDP) - is built upon a language/action perspective (Winograd). Many designers and practitioners use similar concept in different ways to create human computer interaction. E.g. a linguistic design approach to interface design (Andreev, 2001, Tidwell, 1999, Tidwell, 2010) and human computer interaction design (Bueno and Barbosa, 2007, Erickson, 2000a, Branigan et al., 2010).

A key outcome of using the Interaction Language Design Pattern is the creation of a common language: the DSIL - between the user and the computer. The intention is that the DSIL is *“a lingua franca for a design project which is accessible to all stakeholders, particular those who are traditionally marginalized in the design process: the user”* (Erickson, 2000a p357).

The DSIL allows the user to manipulate a particular interactive artefact on a semantic level in a personal context. Essentially, through the DSIL the users can extensively customize their interactions with the computer.

The results of our prototype study show that the DSIL helps users establish a variety of conversations on different levels: physical, cognitive and emotional.

7.2 Contributions

There are two major contributions in this research. The first contribution, which is the primary contribution, is that we exploring an interaction language that can be combined with typical interactions to establish personalized interaction between the users and the computer. The second contribution is the creation of the Interaction Language Design Pattern (ILDP) to help the designers to generate the DSIL and construct a text-based interaction that allows the users to personalize their interaction.

The contributions help to answer the research question that we posted at beginning of the thesis: How can we create personalized human computer interaction that fosters comprehensive communication between users and the computer? We have proposed DSIL to establish a comprehensive communication between the users and computer. The results show that it is valuable form the participants’ perspective. We also provided ILDP to support the designers create DSIL and construct a text-based interaction to allow the users to personalize their interaction. The goal of constructing personalized interaction between the user and computer has been achieved and demonstrated by the studies.

For the first major contribution, the DSIL is proposed to bind with other type of interactions to establish effective human computer interaction by allowing the end users to customise their interactions. Just as "pattern languages are used by interaction designers to bind pattern solutions together, and help the designers evaluate the solution as a coherent whole". (Erickson, 2000a) The purpose of producing the DSIL is to assist the end users to realize personalized interactions. The intention is that the DSIL is understood and utilized by both designers and end users.

Based on the findings of our user studies we argue that the use of interaction language supports the following outcomes:

- Improved customization. Systems are able to reflect and respond to the users' actions, specific problems and experiences. Enriched the user's interaction experience and construct personal relationship with an interactive product.
- Greater ease of use, particularly for novice users.
- Improved ability for users to find desired functionality especially for the novice users.

For the second contribution, we use ILDP to generate a concrete DSIL and establish common ground for effective interaction between the users and a particular system through the use of a domain specific interaction language. Through this the user is able to personalize their interaction based on individual dynamic interaction experience.

7.3 Application

In this thesis, we have applied the ILDP to create a DSIL for drawing to demonstrate and evaluate our proposals. By using the interaction language of painting and drawing, the users are able to customise and control the drawing system through textual interaction. As a result, the users enable to customize the interactive artefact according to the users' specific tasks and purposes.

We conducted a user experience study to examine whether software created using the ILDP helps users express particular concepts and reflections based on their individual experience during the interaction with the computer (see the results of question from 3.1 to 3.5). The study shows that 90% of the participants are able to effectively operate a system using the text-based interaction (see the results of question from 2.1 to 2.6). Most of the participants, including experts and novices can complete predefined tasks effectively and successfully (see the results of question from 1.1 to 1.5).

According to a user experience study, the textual interaction can improve the users' interaction experiences from two aspects. First, it helps the users to construct a semantic interface-semantic interface through text-based interaction. Users are able to reorganize the interface based on their task and purpose. Second, the personalized interaction helps users improve their interaction experience by completing a particular task or work in a personal way.

7.4 Future work

The future work for this research is to complete the DSIL for the drawing system. The short-term goal focuses on building more inclusive DSIL for a specific application. The intention is to refine the interaction language and to improve the quality of human and computer interaction. In details, once the user can express his/her intentions and cooperate with the system it will generate a user-oriented emotional experience with the system based on the user's personality and affectivity. Essentially experience is a totality, engaging self in relationship with object in a situation. In turn, the user will keep continuing using the system and personalize the system. The concrete outcome is semantic interface and collaborative interaction.

In the longer term we would like to apply the ILDP to other domains. By using the Interaction Language Design Pattern, we can create a variety of domain specific interaction languages for different interactive artefacts. The Interaction Language Design Pattern can be used to create human computer interactions in different domains such as office work, music composition, learning systems and games. For instance, we could create a DSIL for music composition or audio editing software, etc.

7.5 Summary

In this chapter, we summarised the works what I have done and the contribution of this thesis and presented some future directions.

Reference

ABRAN, A., KHELIFI, A., SURYN, W. & SEFFAH, A. 2003. Usability meanings and interpretations in ISO standards. *Software Quality Journal,* 11**,** 325-338.

ALEXANDER, C., ISHIKAWA, S. & SILVERSTEIN, M. 1977. Pattern languages. *Center for Environmental Structure,* 2.

ALLEN, I. E. & SEAMAN, C. A. 2007. Likert scales and data analyses. *Quality Progress,* 40**,** 64-65.

ALLWOOD, J. 1976. Linguistic communication as action and cooperation. *Gothenburg monographs in linguistics,* 2**,** 637-663.

ALLWOOD, J. 1977. A critical look at speech act theory. *Logic, pragmatics and grammar. Lund: Studentlitteratur***,** 53-69.

ALLWOOD, J. 2007. Activity based studies of linguistic interaction.

ALLWOOD, J., NIVRE, J. & AHLSÉN, E. 1992. On the semantics and pragmatics of linguistic feedback. *Journal of semantics,* 9**,** 1-26.

ANDREEV, R. D. 2001. A linguistic approach to user interface design. *Interacting with Computers,* 13**,** 581-599.

AUSTIN, J. L. 1975. How to do things with words. 1962. *Cambridge, MA: Harvard UP*.

BAUDEL, T. & BEAUDOUIN-LAFON, M. 1993. Charade: remote control of objects using free-hand gestures. *Communications of the ACM,* 36**,** 28-35.

BAYLE, E., BELLAMY, R., CASADAY, G., ERICKSON, T., FINCHER, S., GRINTER, B., GROSS, B., LEHDER, D., MARMOLIN, H. & MOORE, B. 1998. Putting it all together: towards a pattern language for interaction design: A CHI 97 workshop. *ACM SIGCHI Bulletin,* 30**,** 17-23.

BEARDEN, M. O. R. 2011. *Eliminate the Photoshop CS5 Learning Curve with Plug-ins and Actions: Adjusting your photos the quick and easy way.... Going from bland to WOW in one or two steps*, CreateSpace.

BEAUDOUIN-LAFON, M. Designing interaction, not interfaces. Proceedings of the working conference on Advanced visual interfaces, 2004. ACM, 15-22.

BENTLEY, R. & DOURISH, P. Medium versus mechanism: Supporting collaboration through customisation. Proceedings of the Fourth European Conference on Computer-Supported Cooperative Work ECSCW'95, 1995. Springer, 133-148.

BERLO, D. K. 1960. *The process of communication: An introduction to theory and practice*, Holt, Rinehart and Winston New York.

BEVAN, N. 1999. Quality in use: Meeting user needs for quality. *Journal of Systems and Software,* 49**,** 89-96.

BØDKER, S., EHN, P., SJÖGREN, D. & SUNDBLAD, Y. Co-operative Design—perspectives on 20 years with 'the Scandinavian IT Design Model'. Proceedings of NordiCHI, 2000. University of Notre Dame, 22-24.

BOGDAN, C., KAINDL, H., FALB, J. & POPP, R. Modeling of interaction design by end users through discourse modeling. Proceedings of the 13th international conference on Intelligent user interfaces, 2008. ACM, 305-308.

BORCHERS, J. O. A pattern approach to interaction design. Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques, 2000. ACM, 369-378.

BRADLEY, M. M. & LANG, P. J. 1994. Measuring emotion: the self-assessment manikin and the semantic differential. *Journal of behavior therapy and experimental psychiatry,* 25**,** 49-59.

BRANIGAN, H. P., PICKERING, M. J., PEARSON, J. & MCLEAN, J. F. 2010. Linguistic alignment between people and computers. *Journal of Pragmatics,* 42**,** 2355-2368.

BUENO, A. M. & BARBOSA, S. D. J. 2007. Using an interaction-as-conversation diagram as a glue language for HCI design patterns on the web. *Task Models and Diagrams for Users Interface Design.* Springer.

BULL, S. & VATRAPU, R. 2011. Supporting Collaborative Interaction with Open Learner Models: Existing Approaches and Open Questions. *Proceedings of the Computer Supported Collaborative Learning (CSCL) 2011*.

CARD, S. K., MORAN, T. P. & NEWELL, A. 1986. *The psychology of human-computer interaction*, CRC Press.

CHIN, J. P., DIEHL, V. A. & NORMAN, K. L. Development of an instrument measuring user satisfaction of the human-computer interface. Proceedings of the SIGCHI conference on Human factors in computing systems, 1988. ACM, 213-218.

CHO, H. & YOON, J. 2013. Toward a New Design Philosophy of HCI: Knowledge of Collaborative Action of "We" Human-and-Technology. *Human-Computer Interaction. Human-Centred Design Approaches, Methods, Tools, and Environments.* Springer.

CLARK, H. H. 1996. *Using language*, Cambridge University Press Cambridge.

CLARK, H. H. & MARSHALL, C. R. 2002. Definite reference and mutual knowledge. *Psycholinguistics: critical concepts in psychology***,** 414.

CLAY, S. R. & WILHELMS, J. 1996. Put: Language-based interactive manipulation of objects. *Computer Graphics and Applications, IEEE,* 16**,** 31-39.

COIERA, E. 2003. Interaction design theory. *International journal of medical informatics,* 69**,** 205-222.

COOPER, A., REIMANN, R. & CRONIN, D. 2012. *About face 3: the essentials of interaction design*, John Wiley & Sons.

COUPER-KUHLEN, E. & SELTING, M. 2001. Introducing interactional linguistics. *Studies in interactional linguistics,* 122.

DE SOUZA, C. S., BARBOSA, S. D. J. & PRATES, R. O. 2001. A semiotic engineering approach to user interface design. *Knowledge-Based Systems,* 14**,** 461-465.

DEARDEN, A. 2006. Designing as a conversation with digital materials. *Design studies,* 27**,** 399-421.

DEARDEN, A. & FINLAY, J. 2006. Pattern languages in HCI: A critical review. *Human–computer interaction,* 21**,** 49-102.

DENEF, S. & KEYSON, D. Talking about implications for design in pattern language. Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems, 2012. ACM, 2509-2518.

DJAJADININGRAT, T., OVERBEEKE, K. & WENSVEEN, S. But how, Donald, tell us how?: on the creation of meaning in interaction design through feedforward and inherent feedback. Proceedings of the 4th conference on Designing interactive systems: processes, practices, methods, and techniques, 2002. ACM, 285-291.

DUBBERLY, H., PANGARO, P. & HAQUE, U. 2009. ON MODELING What is interaction?: are there different types? *interactions,* 16**,** 69-75.

DUMAS, B., LALANNE, D. & OVIATT, S. 2009. Multimodal interfaces: A survey of principles, models and frameworks. *Human Machine Interaction.* Springer.

EGGINS, S. 2004. *Introduction to systemic functional linguistics*, Continuum.

ERICKSON, T. Interaction pattern languages: A lingua franca for interaction design. UPA 98 Conference, 1998.

ERICKSON, T. Lingua Francas for design: sacred places and pattern languages. Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques, 2000a. ACM, 357-368.

ERICKSON, T. Pattern Languages as Languages. CHI'2000 Workshop: Pattern Languages for Interaction Design, 2000b. Citeseer.

ESERYEL, D., GANESAN, R. & EDMONDS, G. S. 2002. Review of computer-supported collaborative work systems.

EVENING, M. 2013. *Adobe Photoshop CS5 for Photographers: A Professional Image Editor's Guide to the Creative Use of Photoshop for the Macintosh and PC*, Focal Press.

FISCHER, G. 2001. User modeling in human–computer interaction. *User modeling and user-adapted interaction,* 11**,** 65-86.

FISCHER, G. & GIACCARDI, E. 2006. Meta-design: A framework for the future of end-user development. *End user development.* Springer.

FISCHER, K. 2012. How Different Views of Communication Influence HCI Design: The Example of Shaping.

FISH, S. 1980. *Is There a Text in this Class?: The Authority of Interpretive Communitites*, Harvard University Press.

FLORES, F., GRAVES, M., HARTFIELD, B. & WINOGRAD, T. 1988. Computer systems and the ontology of organization interaction. *ACM TOIS,* 6.

FONTANA, A. & FREY, J. 1994. The art of science. *The handbook of qualitative research***,** 361-376.

FORLIZZI, J. & BATTARBEE, K. Understanding experience in interactive systems. Proceedings of the 5th conference on Designing interactive systems: processes, practices, methods, and techniques, 2004. ACM, 261-268.

FORLIZZI, J. & FORD, S. The building blocks of experience: an early framework for interaction designers. Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques, 2000. ACM, 419-423.

GIACCARDI, E., CIOLFI, L., HORNECKER, E., SPEED, C., BARDZELL, S., STAPPERS, P. J., HEKKERT, P. & ROZENDAAL, M. Explorations in social interaction design. CHI'13 Extended Abstracts on Human Factors in Computing Systems, 2013. ACM, 3259-3262.

GUARINO, N. & POLI, R. 1995. Formal ontology, conceptual analysis and knowledge representation. *International Journal of Human Computer Studies,* 43**,** 625-640.

HALLIDAY, M. A. 1994. Functional grammar. *London: Edward Arnold*.

HALLIDAY, M. A. K. 2002. *On grammar*, Continuum.

HAQUE, U. 2006. Architecture, interaction, systems. *AU: Arquitetura & Urbanismo,* 147**,** 68-71.

HARKER, S. 1995. The development of ergonomics standards for software. *Applied ergonomics,* 26**,** 275-279.

HARPER, B. D. & NORMAN, K. L. Improving user satisfaction: The questionnaire for user interaction satisfaction version 5.5. Proceedings of the 1st Annual Mid-Atlantic Human Factors Conference, 1993. 224-228.

HARRISON, S., SENGERS, P. & TATAR, D. 2011. Making epistemological trouble: Third-paradigm HCI as successor science. *Interacting with Computers,* 23**,** 385-392.

HARTSON, R. 2003. Cognitive, physical, sensory, and functional affordances in interaction design. *Behaviour & Information Technology,* 22**,** 315-338.

HUANG, K.-Y. Challenges in human-computer interaction design for mobile devices. Proceedings of the World Congress on Engineering and Computer Science, 2009. 236-241.

IXDA. 2014. Available: http://www.ixda.org/about/ixda-mission.

JOHNSON-LAIRD, P. N. 1983. *Mental models: Towards a cognitive science of language, inference, and consciousness*, Harvard University Press.

JOKINEN, K. 2009. Constructive Dialogue Model (CDM). *Constructive Dialogue Modelling: Speech Interaction and Rational Agents***,** 53-72.

JORDAN, P. W. 2002. *Designing pleasurable products: An introduction to the new human factors*, CRC Press.

KAHN, R. L. & CANNELL, C. F. 1957. The dynamics of interviewing; theory, technique, and cases.

KAPLAN, B. & MAXWELL, J. A. 2005. Qualitative research methods for evaluating computer information systems. *Evaluating the Organizational Impact of Healthcare Information Systems.* Springer.

KAPTELININ, V. & BANNON, L. J. 2012. Interaction design beyond the product: Creating technology-enhanced activity spaces. *Human,ÄìComputer Interaction,* 27**,** 277-309.

KENDON, A. 2009. Language's matrix. *Gesture,* 9**,** 355.

KOHLHASE, A. 2008. *Semantic interaction design: composing knowledge with CPoint.* University of Bremen.

KRIPPENDORFF, K. 2005. *The semantic turn: A new foundation for design*, crc Press.

LANGDON, P., CLARKSON, J., ROBINSON, P., LAZAR, J. & HEYLIGHEN, A. 2012. *Designing Inclusive Systems*, Springer.

LANGLEY, P. & HIRSH, H. 1999. User modeling in adaptive interfaces. *Courses and lectures-international centre for mechanical sciences*, 357-370.

LANGLEY, P., LAIRD, J. E. & ROGERS, S. 2009. Cognitive architectures: Research issues and challenges. *Cognitive Systems Research,* 10, 141-160.

LIDDLE, D. Design of the conceptual model. Bringing design to software, 1996. ACM, 17-36.

LIM, Y.-K., DONALDSON, J., JUNG, H., KUNZ, B., ROYER, D., RAMALINGAM, S., THIRUMARAN, S. & STOLTERMAN, E. 2008. Emotional experience and interaction design. *Affect and Emotion in Human-Computer Interaction.* Springer.

MACKAY, W. E. Which interaction technique works when?: floating palettes, marking menus and toolglasses support different task strategies. Proceedings of the Working Conference on Advanced Visual Interfaces, 2002. ACM, 203-208.

MAHEMOFF, M. J. & JOHNSTON, L. J. Pattern languages for usability: An investigation of alternative approaches. Computer Human Interaction, 1998. Proceedings. 3rd Asia Pacific, 1998. IEEE, 25-30.

MARTINEC, R. & VAN LEEUWEN, T. 2009. *The language of new media design: Theory and practice*, Routledge.

MCCARTHY, J. & WRIGHT, P. 2004. Technology as experience. *interactions,* 11, 42-43.

MOGGRIDGE, B. & SMITH, G. C. 2007. *Designing interactions*, MIT press Cambridge.

MONK, A. 2009. Common ground in electronically mediated communication: Clark's theory of language use. *HCI models, theories, and frameworks: Toward a multidisciplinary science*, 265-289.

MONK, A. F. & DIX, A. J. Refining Early Design Decision with a Black-Box Model. BCS HCI, 1987. 147-158.

MORRIS, P. 1994. The Bakhtin Reader Edward Arorld, London

MORSON, G. S. & EMERSON, C. 1990. *Mikhail Bakhtin: Creation of a prosaics*, Stanford University Press.

MYERS, B. A. Separating application code from toolkits: eliminating the spaghetti of call-backs. Proceedings of the 4th annual ACM symposium on User interface software and technology, 1991. ACM, 211-220.

MYERS, B. A., PANE, J. F. & KO, A. 2004. Natural programming languages and environments. *Communications of the ACM,* 47, 47-52.

NIELSEN, J. Usability inspection methods. Conference companion on Human factors in computing systems, 1994. ACM, 413-414.

NIELSEN, J., TAHIR, M. & TAHIR, M. 2002. *Homepage usability: 50 websites deconstructed*, New Riders Indianapolis, IN.

NORMAN, D. A. 1988. *The psychology of everyday things*, Basic Books (AZ).

NORMAN, D. A. 2002. *The design of everyday things*, Basic Books (AZ).

NORMAN, D. A. 2007. *Emotional design: Why we love (or hate) everyday things*, Basic books.

NORMAN, D. A. & DRAPER, S. W. 1986. *User centered system design; new perspectives on human-computer interaction*, L. Erlbaum Associates Inc.

OVIATT, S. 2003. User-centered modeling and evaluation of multimodal interfaces. *Proceedings of the IEEE,* 91**,** 1457-1468.

PAN, Y. & STOLTERMAN, E. Pattern language and HCI: expectations and experiences. CHI'13 Extended Abstracts on Human Factors in Computing Systems, 2013. ACM, 1989-1998.

PANE, J. F. & MYERS, B. A. 2006. More natural programming languages and environments. *End User Development.* Springer.

PAPPAS, H. 2011. *HCI and Interaction Design: In search of a unfied language.* KTH.

PETRA SUNDSTRÖM1, A. S. H., KRISTINA HÖÖK 2005. A user-centered approach to affective interaction. *Affective Computing and Intelligent Interaction.* Springer.

PETRELLI, D., DE ANGELI, A. & CONVERTINO, G. 1999. A User-Centered Approach to User Modelling. *COURSES AND LECTURES-INTERNATIONAL CENTRE FOR MECHANICAL SCIENCES***,** 255-264.

POLOVINA, S. & PEARSON, W. 2006. Communication+ Dynamic Interface= Better User Experience. *Encyclopaedia of Human Computer Interaction***,** 85.

PREECE, J. R. & ROGERS, Y. 2007. SHARP (2002): Interaction Design: Beyond Human-Computer Interaction. *Crawfordsville: John Wiley and Sons, Inc. Answers. com Technology*.

PRESS, N. C. D. 1996. Organizational communication. *An Integrated Approach to Communication Theory and Research***,** 383.

RICHARDS, L. 2009. *Handling qualitative data: A practical guide*, Sage Publications.

ROGERS, Y., SHARP, H. & PREECE, J. 2011. *Interaction design: beyond human-computer interaction*, Wiley.

RUBIN, J. & CHISNELL, D. 2008. *Handbook of usability testing: howto plan, design, and conduct effective tests*, John Wiley & Sons.

RUMBAUGH, J., JACOBSON, I. & BOOCH, G. 1999. The unified modeling language reference manual.

RUMPE., D. H. A. B. 2004. Meaningful Modeling: What's the Semantics of "Semantics"? *Computer,* 37**,** 64–72.

RUSSELL, J. A., LEWICKA, M. & NIIT, T. 1989. A cross-cultural study of a circumplex model of affect. *Journal of personality and social psychology,* 57**,** 848.

RYU, H. & MONK, A. 2009. Interaction unit analysis: A new interaction design framework. *Human,ÄìComputer Interaction,* 24**,** 367-407.

SAFFER, D. 2007. Designing for interaktion. Berkeley Calif: New Rider.

SAFFER, D. 2009. *Designing for interaction: creating innovative applications and devices*, New Riders.

SAGER, J. C. 1994. *Language engineering and translation: consequences of automation*, John Benjamins Publishing.

SCHEGLOFF, E. A. 1997. Whose text? Whose context? *Discourse & Society,* 8**,** 165-187.

SCHULER, D. & NAMIOKA, A. 1993. *Participatory design: Principles and practices*, Routledge.

SENGERS, P., BOEHNER, K. & KNOUF, N. Sustainable HCI meets third wave HCI: 4 themes. CHI 2009 workshop, 2009.

SHNEIDERMAN, B. 1993. 1.1 direct manipulation: a step beyond programming languages. *Sparks of Innovation in Human-Computer Interaction***,** 17.

SHNEIDERMAN, B., NORMAN, K. L., PLAISANT, C., BEDERSON, B. B., DRUIN, A. & GOLBECK, J. 2013. 30 years at the University of Maryland's human-computer interaction lab (HCIL). *interactions,* 20**,** 50-57.

STANTON, N. A., HEDGE, A., BROOKHUIS, K., SALAS, E. & HENDRICK, H. W. 2004. *Handbook of human factors and ergonomics methods*, CRC Press.

STIVERS, T. & SIDNELL, J. 2005. Introduction: multimodal interaction. *Semiotica,* 2005**,** 1-20.

SUCHMAN, L. A. 1987. *Plans and situated actions: the problem of human-machine communication*, Cambridge university press.

THURING, M. & MAHLKE, S. 2007. Usability, aesthetics and emotions in human,Äìtechnology interaction. *International Journal of Psychology,* 42**,** 253-264.

TIDWELL, J. 1997. A pattern language for human-computer interface design. *Available via DIALOG*.

TIDWELL, J. 1999. Common ground: A pattern language for human-computer interface design.

TIDWELL, J. 2010. *Designing interfaces*, O'Reilly.

TILLY, K. & PORKOLÁB, Z. 2010. Semantic user interfaces. *International Journal of Enterprise Information Systems (IJEIS),* 6**,** 29-43.

TUBBS, S. 2010. human communication principle and context.

VALENCIA-GARCÍA, R. & GARCÍA-SÁNCHEZ, F. 2013. Natural Language Processing and Human-Computer Interaction. *Computer Standards & Interfaces*.

VAN DUYNE, D. K., LANDAY, J. A. & HONG, J. I. 2007. *The design of sites: Patterns for creating winning web sites*, Prentice Hall Professional.

VAN WELIE, M. & VAN DER VEER, G. C. Pattern languages in interaction design: Structure and organization. Proceedings of interact, 2003. 1-5.

VOELTER, M., BENZ, S., DIETRICH, C., ENGELMANN, B., HELANDER, M., KATS, L. C., VISSER, E. & WACHSMUTH, G. 2013. *DSL Engineering: Designing, Implementing and Using Domain-specific Languages*, CreateSpace Independent Publishing Platform.

WALLER, S., GOODMAN-DEAN, J., LANGDON, P., JOHNSON, D. M. & CLARKSON, P. J. Developing a method for assessing product inclusivity. Proceedings of ICED 09, the 17th International Conference on Engineering Design, Volume 5: Design Methods and Tools, Part 1, 2009. Design Society, 335-346.

WINOGRAD, T. A language/action perspective on the design of cooperative work. Proceedings of the 1986 ACM conference on Computer-supported cooperative work, 1986. ACM, 203-220.

WINOGRAD, T. 1997. The design of interaction. *Beyond calculation.* Springer.

WINSKEL, G. 1993. *The formal semantics of programming languages: an introduction*, The MIT Press.

WRIGHT, P., WALLACE, J. & MCCARTHY, J. 2008. Aesthetics and experience-centered design. *ACM Transactions on Computer-Human Interaction (TOCHI),* 15**,** 18.

ZHUGE, H. 2010. Interactive semantics. *Artificial Intelligence,* 174**,** 190-204.

Glossary:

1. Interactive system: Some kind of designed artefact which users interact with. e.g. Computer GUI system such as drawing system, word processor, web browser, etc.
2. Interaction language: Visuals, text, sound, gestural input, etc which designers use to create an interactive experience for users.
3. Domain Specific Interaction Language (DSIL): Specific visuals, sounds, gestural inputs, etc. which are used to create an interactive system for particular end-users such as illustrators, novelists, etc.
4. Interaction Language Design Pattern (ILDP): A design pattern for producing effective interactive systems. It specifies how to design domain specific interaction languages and personalized interaction.
5. Interactive semantics: The meaning users discern from the look, behaviour and affordances of the interactive system.
6. Utterance: output from a computer intended to convey some meaning to the user through a specific interface.
7. Genre: the mode of interaction available to the user to interact with the computer e.g. instrument interaction, embodied interaction and linguistic interaction etc.

Appendix A

UNIVERSITY OF
TECHNOLOGY SYDNEY

UTS: IT: CREATIVITY& COGNITION STUDIOS

" ETHNOGRAPHIC RESEARCH OF INTERACTION DESIGNER FOCUSING ON AN INTERACTION-ORIENTED DESIGN METHOD" UTS HREC REF NO 2010-072P

CONSENT FORM

I ______________________________ *(participant's name)* agree to participate in the research project "Ethnographic research of interaction designer focusing on an interaction-oriented design method" being conducted by YI JI of the Creativity and Cognition Studios at the University of Technology, Sydney.

I understand that the purpose of this study is to provide an understanding of the designers' perception of interaction, and how they utilize design method to create an interactive artefact. This is part of a program of postgraduate research leading toward a PhD of interaction design. Data gathered in these studies will contribute to ongoing research on a new design method for interaction design.

I understand I will be interviewed about my work for approximately 45 minutes.

I understand the researcher will observe me as I go about my work for up to 12 hours over 10 days.

I understand the interview will be audio or video recorded.

I agree that the research data gathered from this study may be published in a form that does not identify me in any way.

I am aware that I can contact Prof Ernest Edmonds (02 9514 4640) or Dr Andrew Johnston at UTS (02 9514 4497) or the University of Technology, Sydney Human Research Ethics Committee (see note below) if I have any concerns about the research. I also understand that I am free to withdraw my participation from this research project at any time I wish and without giving any reasons.

I agree that YI JI has answered all my questions fully and clearly.

____ ____/____/____

Signed by

____ ____/____/____

Witnessed by

NOTE:

This study has been approved by the University of Technology, Sydney, Human Research Ethics Committee.

If you have any complaints or reservations about any aspect of your participation in this research that you cannot resolve with the researcher, you may contact the UTS Ethics Committee through the Research Ethics Officer at UTS Broadway, Building 1, Level 14; or 9514 9772; or Research.Ethics@uts.edu.au. Please quote the UTS HREC reference number.

Any complaint you make will be treated in confidence and investigated fully and you will be informed of the outcome.

Appendix B

Questions for user interview

1. What do you think about the drawing system when you want to make a drawing?
2. How did you use the drawing system to complete the tasks?
3. Do you find any problem or difficulty by using the drawing system? What were they?
4. What were the main things you like about the way of drawing (interaction)?
5. What were the main things you dislike about the way of drawing (interaction)?
6. Do you feel more creative while you drawing? For example it can derive your creative responds?
7. Do you feel free or easy to achieve what you want to do with this drawing system? In other words, the system can understand what you want it to do.
8. Do you like the way of drawing?
9. Overall did you enjoy the experience of drawing?
10. Do you have any other comments?

Appendix C

Questionnaire for user experience study

Basic Information

Name:

Age: A. 18-27 B. 28-37 C. 38-47 D. 48-57

Job:

Level of using drawing system: A. Beginner B. Meddle C. Expert

On average, how often do you use a drawing system?

- Every day
- Once per week
- Once per month
- Once every 6 months
- Once a year
- Less than once per year

Task:

We are going to provide you a drawing tool for a simple sketching practice. Once you have completed the task, kindly describe your feelings about the drawing practice and the overall experiences as an answer to the following questions;

PART 1: How do you evaluate the drawing system

Please circle the numbers that most appropriately reflect your expectation of interaction (Drawing)

.

1.1 I can effectively make a drawing

1	2	3	4	5
Strong disagree	Disagree	Neutral	Agree	Strong Agree

1.2 It was easy to learn how to make a drawing

1	2	3	4	5
Strong disagree	Disagree	Neutral	Agree	Strong Agree

1.3 I found it difficult to find the tools I needed

1 2 3 4 5

Strong disagree	Disagree	Neutral	Agree	Strong Agree

1.4 The interface of this system is pleasant

1	2	3	4	5
Strong disagree	Disagree	Neutral	Agree	Strong Agree

1.5 The system has all the functions and capabilities

I expect it to have

1	2	3	4	5
Strong disagree	Disagree	Neutral	Agree	Strong Agree

PART 2: How do you defined your drawing behaviour

2.1 It was uncomfortable making a drawing

1	2	3	4	5
Strong disagree	Disagree	Neutral	Agree	Strong Agree

2.2 I felt more creative while drawing

1	2	3	4	5
Strong disagree	Disagree	Neutral	Agree	Strong Agree

2.3 It was easy to complete the task

1	2	3	4	5

Strong disagree	Disagree	Neutral	Agree	Strong Agree

2.4 It took too long to complete the task

1	2	3	4	5
Strong disagree	Disagree	Neutral	Agree	Strong Agree

2.5 I found the interaction attractive

1	2	3	4	5
Strong disagree	Disagree	Neutral	Agree	Strong Agree

2.6 I like this way of drawing

1	2	3	4	5

Strong disagree	Disagree	Neutral	Agree	Strong Agree

PART 3: How do you evaluate your drawing experience

3.1 Using the program was enjoyable.

1	2	3	4	5
Strong disagree	Disagree	Neutral	Agree	Strong Agree

3.2 It was difficult to express myself using this program

1	2	3	4	5
Strong disagree	Disagree	Neutral	Agree	Strong Agree

3.3 Drawing with the program felt 'natural'.

1	2	3	4	5
Strong disagree	Disagree	Neutral	Agree	Strong Agree

3.4 What is your overall impression of the value of this program?

1	2	3	4	5
Very worthless	Worthless	Neutral	Precious	Very Precious

3.5 This program is novel

1	2	3	4	5
Strong disagree	Disagree	Neutral	Agree	Strong Agree

Appendix D

Interaction Language Design Pattern application guideline

Introduction

The purpose of this design guide is to illustrate how to use our design pattern to create an interaction language that supports human computer collaborative interaction. In this case, the collaborative interaction means that the computer will support the users to complete their work in a personal way. This design guide introduces a linguistic design pattern to organize different components of interaction and convey variety of concepts between a user and a real interactive product. By following the linguistic design pattern the designers are able to build up an interaction language for the users to interact with the computer. This guide could be used in different design contexts by different designers include software engineers, interaction designers, multimedia designer to accomplish a particular interaction design. To successfully use this design guide it is necessary to take specific needs and constraints of participants and contexts into consideration.

Purpose

When designing meaningful interaction between human and computer, we need to figure out how the interaction fulfils the user's individual purpose. This design pattern helps designers to have a comprehensive understanding of designing human computer interaction. It assists the designers to create personalized interactions that allow the user to solve his/her individual problem of interaction with the computer. In particular, the problems related to: How does an interactive system meet the user's need? What are the tasks the user hopes to accomplish? How do these tasks fit into the user's individual interaction context?

Interaction language

To build personalized interaction, we consider it is necessary to produce a user-oriented interaction domain specific language. The language should allow the user direct involvement in the continuous development, use and evolution of interaction with the resultant system.

According to the linguistic theory, a user specific interaction language can lead to personalized human computer interaction through a linguistic structure:

- Defining diverse of interaction vocabulary to represent an interactive artefact.
- Organizing the interactive artefact in a particular way following different specific interaction syntaxes.
- Realizing the meaningful responses to user's actions and command conveying a predefined interaction semantic

Interaction Language Design Pattern

To design personalized interaction between human and computer, we must take into account a number of key factors such as user characteristics, properties of the interactive artefact and the context of a particular interaction. The Interaction Language Design Pattern (ILDP) is composed of two sub design tasks. Each task has specific attributes and purposes:

Interaction language concept: generate the design concept for an interaction design task using a linguistic model

Domain specific interaction language: realize a specific interactive artefact based on the design concepts

This guide explains how to apply the design patterns in actual design practice by giving diverse examples. The structure of the interaction language design pattern is shown as follow.

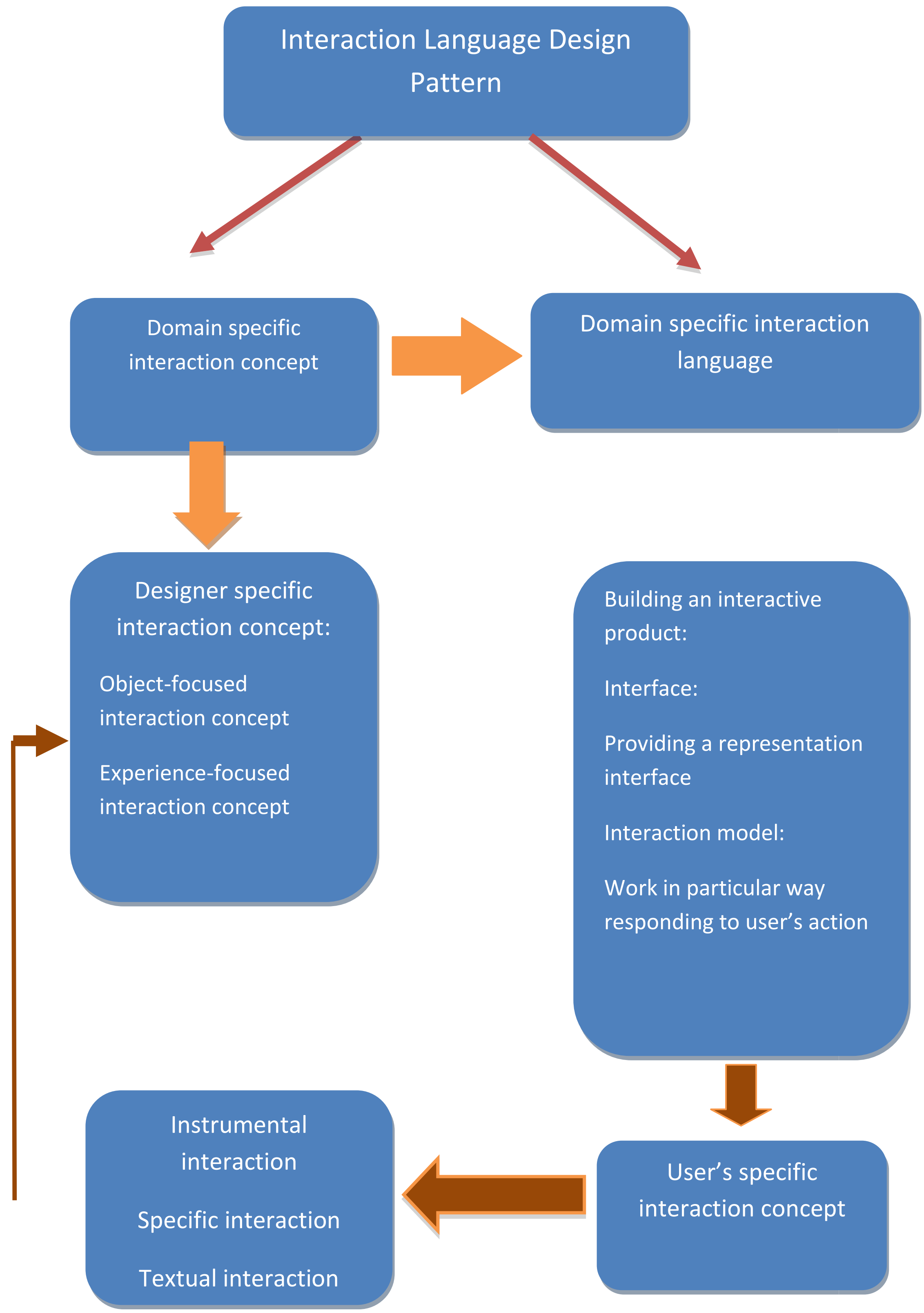


Table 1 Components of Interaction Language Design Pattern

In this guide we will not describe the design pattern in full detail. A complete description of the ILDP can be found in Chapter 4. Instead, we provide a practical guide to illustrate how the design pattern is used and help the designers to produce a collaborative interaction between the human and computer.

1. Interaction language concept

Before creating a concrete interactive artefact, we usually have a definition statement – a concise, concrete declaration of the main interactive artefact, interaction context and intended user's experience. This is the first step of designing collaborative interaction between the users and the interactive artefact. The purpose of constructing Interaction language concept is to create theoretical prototypes of interactive artefacts based on both object-oriented, experience-oriented interaction concepts. As a result, the interactive artefact conveys two types of interaction concepts:

- Object-oriented interaction concept
- Experience-oriented interaction concept

Object-oriented interaction concept

This interaction concept describes interaction of an interactive artefact to explain how it is going to be used by a user.

Example of object-oriented interaction concept

We have built three object-oriented interaction concepts for a drawing system including pen, crayon and eraser. These objects can be represented by different interaction vocabulary like text and graphic icons (see figure 1).

Figure 1 Graphic icon of interaction objects

- Each object has different functionality and attributes.
- Pen: a user can draw a painting by using pen. When the user presses the icon the predefined function will be unfolded.
- Crayon: a user can draw a crayon picture by using the crayon tool.
- Eraser: a user can clean the content what he/she has done.

Experience-oriented interaction concept

The experience–oriented interaction concept describes a desired user experience through the interaction. For example, designers can present an object and organize a variety of objects in different ways such as its attributes, visual representation, physical behaviour, sound effect and so on.

Example of experience-oriented interaction concept

The pen object in our system needs a visual representation. In this step, we design the look of the pen as it will appear on screen (figure 1). We also specify the behaviour of the pen object. In our example system, when the pen object is clicked, a range of specific pen types (e.g. pencil, pen, paint brush, etc) appear next to the main pen icon (see figure 2). Figure 2 displays the designers' experience-oriented interaction concept of the pen object.

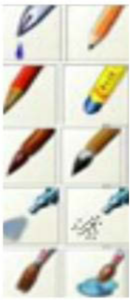


Figure 2 A pattern of showing pen objects

Outcome of interaction language concept

The Interaction language concept semantically maps out the interaction between users and interactive artefact under a linguistic structure: vocabulary, syntax and semantic. As we mentioned before, the semantic of human computer interaction mainly focuses on two aspects: object-oriented semantics and experience-oriented semantics. Following the interaction language concept we can establish a meaningful and personalized interaction by building an interactive artefact. That means the designers are able to drive how a user's interaction experience through an interactive artefact. Meanwhile, interaction designers provide the key words represent the concept of objects and patterns, which is essential for the user to recognize a particular product to complete a particular task. It should capture all the tasks related to the interaction.

2. **Domain specific interaction language**

At this stage, designers will use whatever tools they need in order to create a final interactive product. To do this, programming language and interaction pattern language will be used. Figure 3 is an example of using JavaScript to create a drawing pen object (see figure 3). The end result will be an interactive product, which has two main elements. The first is a graphical user interface, which is mostly controlled using a mouse. The second will be a textual input feature, which allows the user to type in words to control the tools of the system.

```
<script type="text/JavaScript">

    //Create jsGraphics object

    var gr = new
jsGraphics(document.getElementById("canvas"));

    //Create jsColor object

    var col = new jsColor("red");


    //Create jsPen object

    var pen = new jsPen(col,1);

    //Draw a Line between 2 points

    var pt1 = new jsPoint(20,30);

    var pt2 = new jsPoint(120,230);

    gr.drawLine(pen,pt1,pt2);

    //Draw filled circle with pt1 as center point and
radius 30.

    gr.fillCircle(col,pt1,30);

    //You can also code with inline object
instantiation like below

    gr.drawLine(pen,new jsPoint(40,10),new
jsPoint(70,150));


</script>
```

Figure 3 functional performance of drawing pen created by using JavaScript

Example of DSIL

For example, the text boxes labelled 'Level 3' in figure 3 let the user type in object names (e.g. 'pen', 'eraser', etc.). When they type in the object name, the system will automatically bring up the object with associated properties (see the red colour circle in figure 3). The visual elements are linked to the textual input system. This means that when the user types in 'pen' and then enters 'colour' the pen selection icons and the colour tool will appear on screen (see the yellow circle in figure 3).

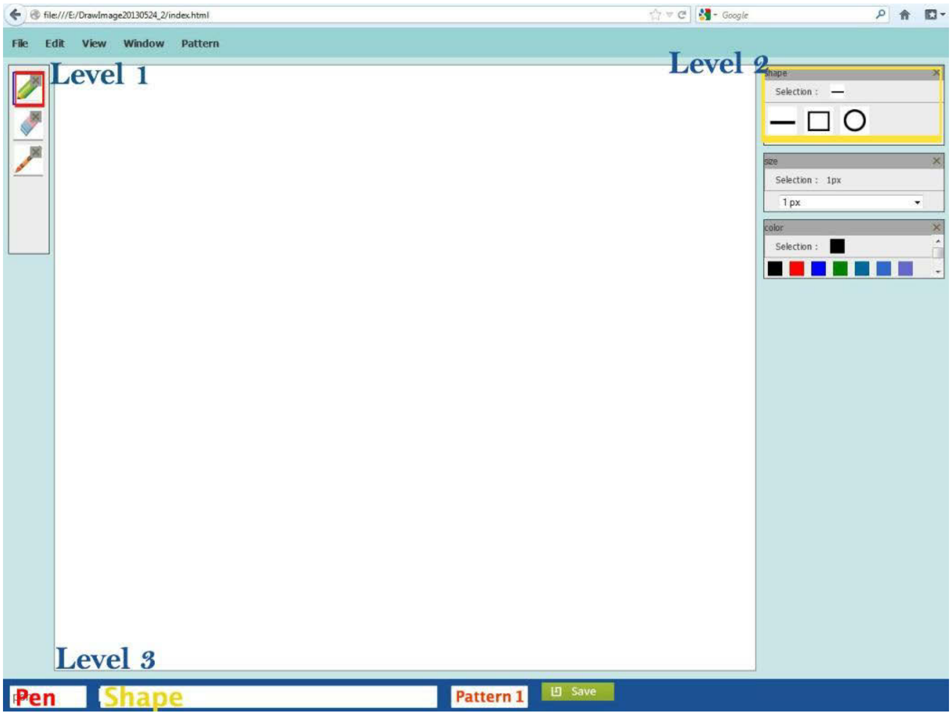


Figure 3 Composing an interface using an interaction domain specific language

Following the linguistic structure, the system provides two different inputting functionalities: the first one is designed for object; the second one is designed for property of object (see figure 4). As a result, when the user needs a tool s/he has to types the name of object into the object-oriented input box. Similarly, if the user wants to change attributes of an object s/he needs to type the word into the experience-oriented input boxes. For example, if the user enters “pen” in the object input box the pen selection icons will appear on the screen. While, if the user enters “colour” in the property box the system will bring the user with a particular tool.

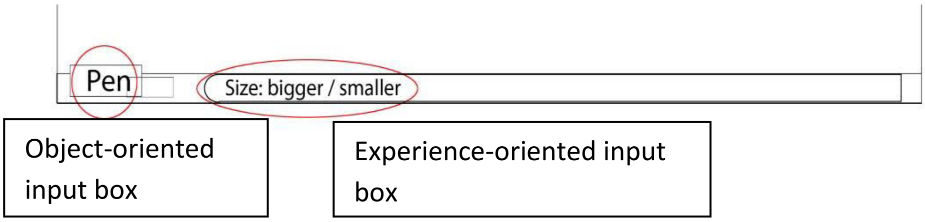


Figure 4 Different input functionalities for different purposes.

In order to optimize the user’s participation regarding the users’ background knowledge and experience during the interaction. Therefore, a well-designed semantic selection system will bring the relevant pre-selected function list to the users when the users use the semantic selection system to complete a specific task. The first list focuses on the object-oriented interaction concept; the second list relates to the experience-oriented interaction concept (see figure 5). This assist system is built on the specific interaction domain language system that is a base from which a designer can create the function and interaction context of the drawing system.

Figure 5 An example of semantic selection system

Of course, the objects and their on-screen appearance will be different depending on the context for which the application is developed. For example, the order that users type in object names determines the on-screen organization of their icons. For example, if the user types in 'pen' followed by 'eraser' followed by 'crayon', then the icons will be arranged on screen in that order (pen icon next to eraser icon next to crayon).

For the end user, once s/he realizes the principle of interaction the user is ready to build up his/her individual interaction. Because of the system not only allows the user to construct a meaningful interface and special interaction model but also can save it. That means the user is able to use this particular interface and model of interaction in the future.

At this stage, designers need to consider the desired users' experience as identified in the previous step ('Interaction Language Concept') and design the system to make these concepts a reality. Existing design patterns are likely to be helpful here. Eventually, the interaction domain specific language is created based on the interaction between the user and the system. In other words, the concept of designing interaction can be transferred to generate a concrete interactive artefact. In turn, the interaction reflects the user's individual interaction concepts, which is generated from a continuing interaction with interactive artefact by redefining the meaning and structure of the interaction.

Outcome of interaction domain specific language

In this stage, the designers will build up a concrete communication between human and interactive artefact. The interaction is a communication development process composed of two phases: the first phrase is about how the designers embody their interaction concepts in a concrete interactive artefact. In the following example, the designers will create a concrete drawing system using two interaction syntaxes: programming language and interaction design pattern language. The second phase is an internal cognitive process that reflects how the users understand the interactive artefact and its underlining interaction semantics.

Example: how the user uses the drawing system that applies this design pattern

The interaction language supports the user to take a reasonable action. The drawing system is generated from how the user defines his/her personalizing interaction to better cooperate with the computer. To achieve this goal, it requires a specific interaction domain language for a particular human computer interaction. It is the key element that we can establish a common ground between the users and the designer who created interactive artefact. For example, to create a painting system we need to establish a painting language. By doing so, all participants of the interaction will understand each other by using the domain language.

Next, we will illustrate how the user uses the drawing system to draw a simple picture step by step:

Following is the original interface when the user starts to use it (see figure 6)

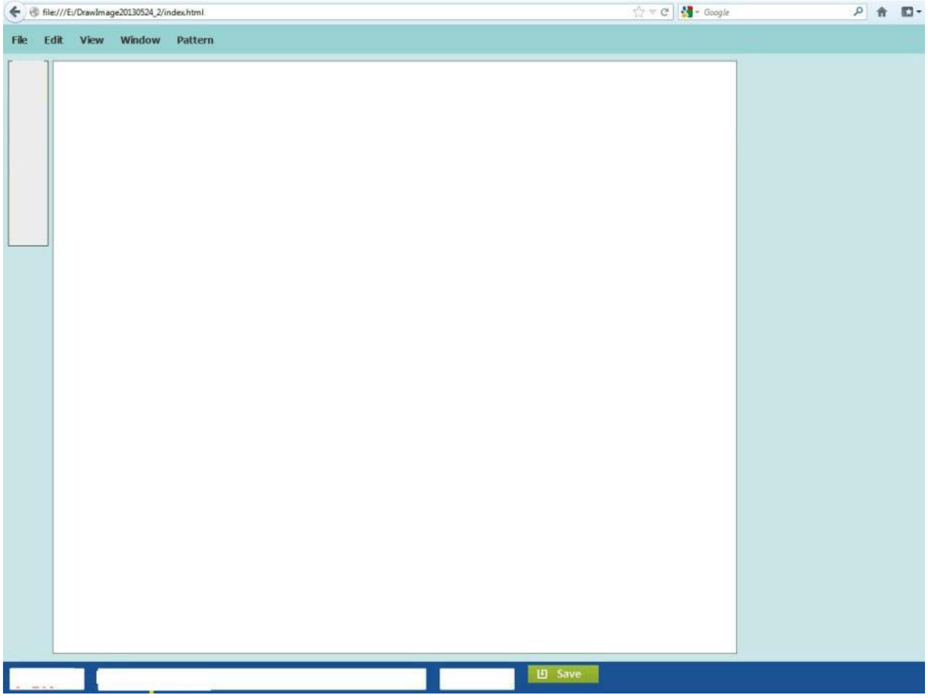


Figure 6. The original interface of drawing system screenshot

The first step for the user to make drawing is to find a pen. So the user is able to have the pen by typing in the object name “pen” in the object box and then the object pen will appear on the screen with a selection icon (see figure 7).

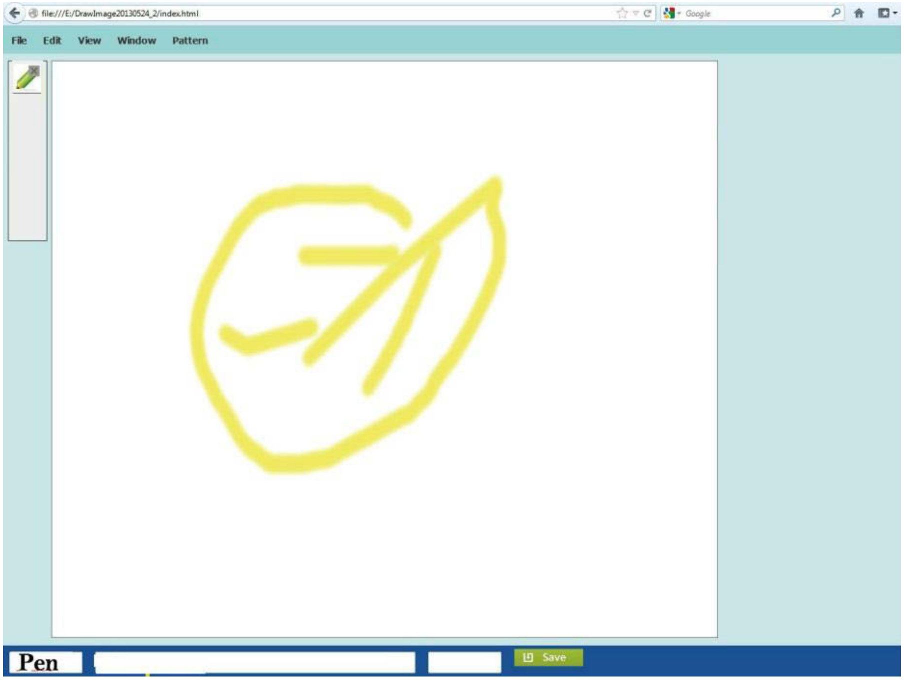


Figure 7 working interface screenshot 1

Meanwhile, the user is allowed to have any tool by typing in the object name, which s/he wants to use (see figure 8).

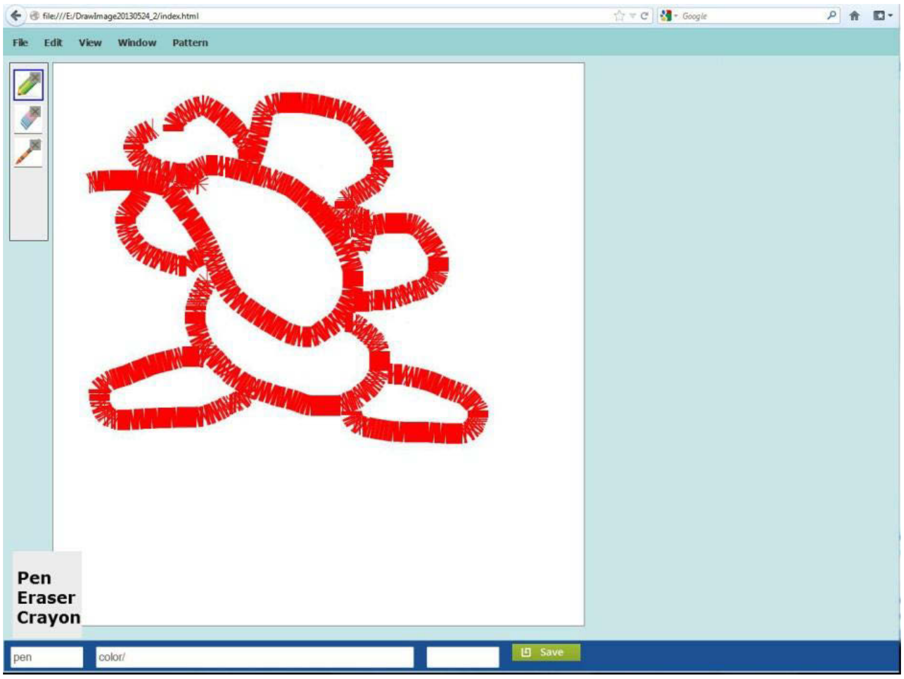


Figure 8 working interface screenshot 2

The second step is to change the property of the object. For example the user can change the size, shape and colour of the pen by typing the name of the property (see the figure 9).

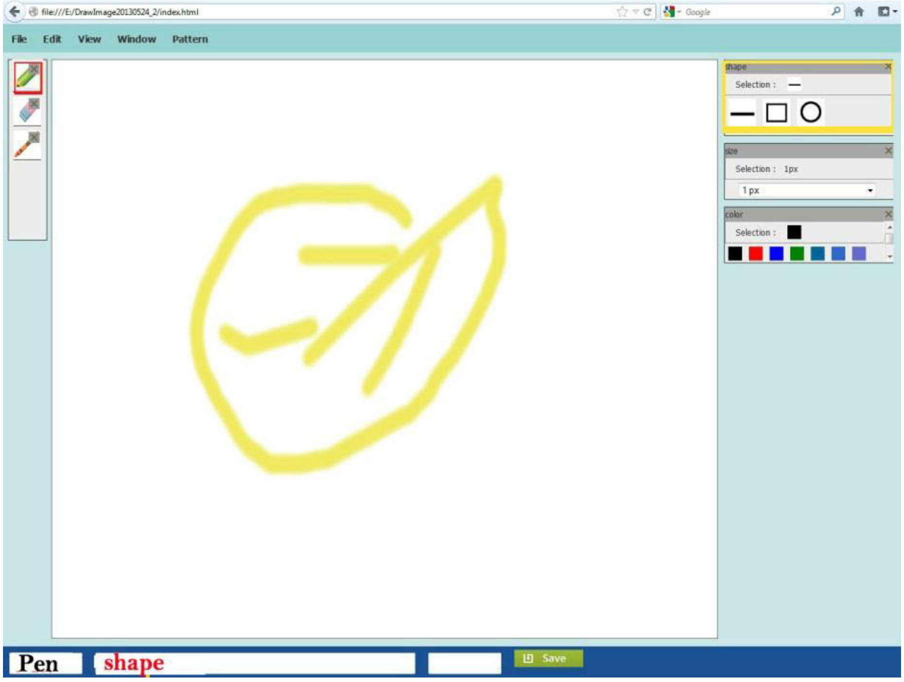


Figure 9 working interface screenshot 3

In step 3 the user is going to construct a particular interaction context and property of a tool utilizing the provided interaction domain specific language. For example, the Users aim to have an appropriate tool to complete a task- drawing a simple line. In order to find the tool the users either to look through a drop down menu or directly express his/her concept by typing in an object name. Through the interaction, the user eventually constructs personalizing interaction according to the user's individual intention (see figure 10).

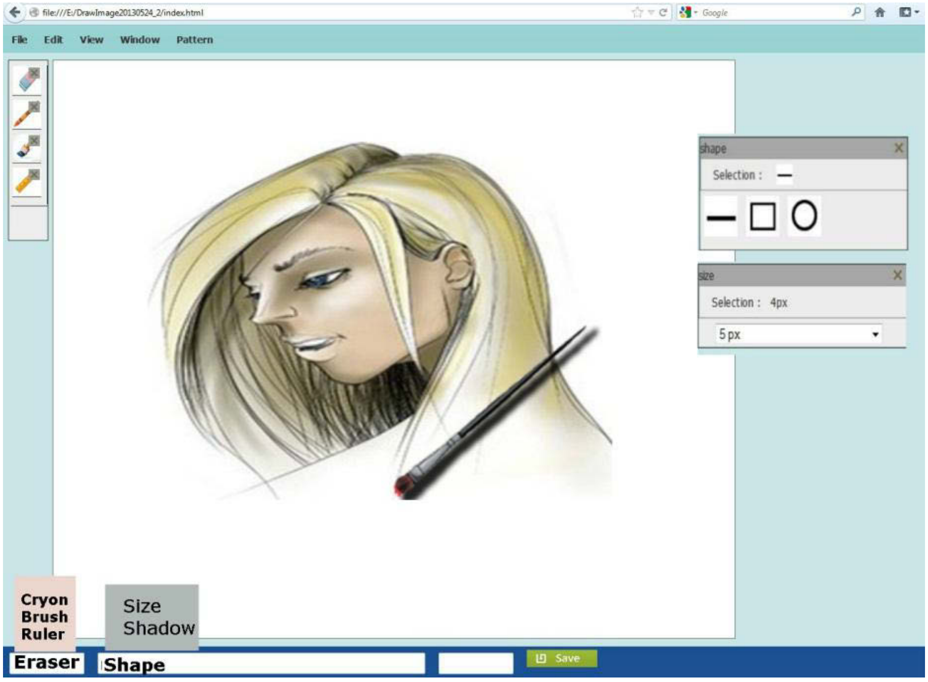


Figure 10 working interface screenshot 4

In the fourth step, once the user completes the construction of an individual interaction by organizing a variety of interaction elements, they can save the interaction and use it in any time. Because of the system recodes the user activity during the interaction when the user changed the order and structure of the interface using interaction domain specific language. Finally, the user needs to give a name to define the customized interface and interaction to represent his/her interaction concept. For example, the user enters a name in our example system which is tagged as ‘pattern 1’ and press the green button. And then later when the user opens the system s/he will find the “pattern 1” in drop down menu of pattern. The “pattern 1” composed of the user customized interface and interaction, which is constructed by a vanity of interaction elements in a specific structure (see figure 11)

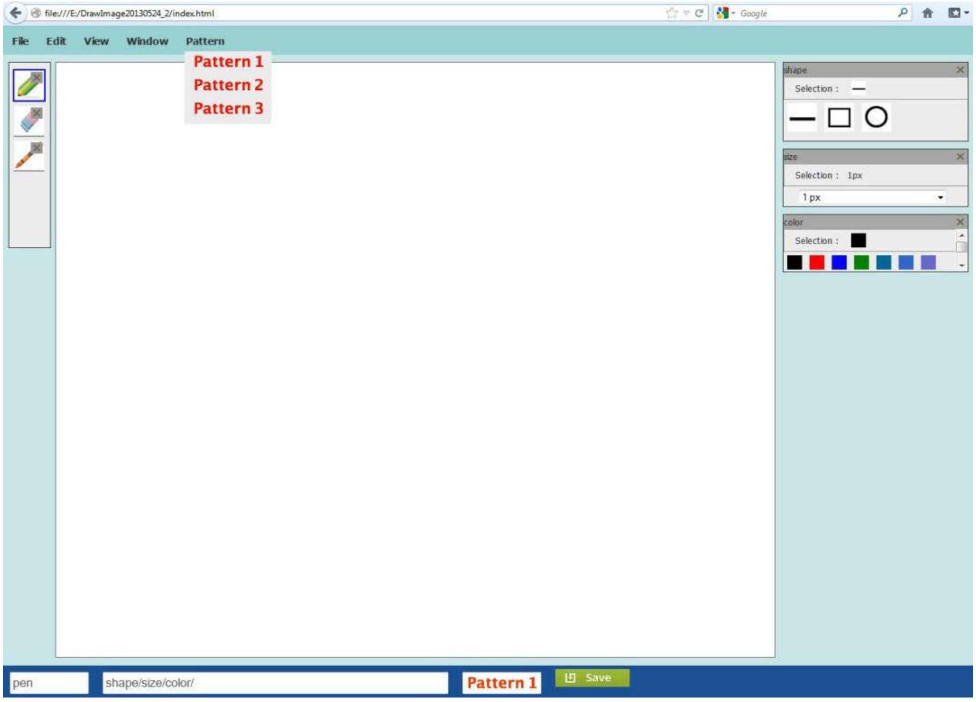


Figure 11 personalized interaction pattern

In other words, the end users have ability to operate the system same as other stakeholders like software designers and interface designer. Consequently, the users are able to collaborate with the computer at different levels including physical, cognitive and emotional.

Outcome of Interaction language pragmatic pattern

In this case, human computer interaction as operating the drawing system that is a kind of collaborative action, as the user's interaction based on the user's individual reflection. That means when the users have problem to understand the meaning of interaction, they are enable to solve the problem by co-operating with the computer by defining an interaction language to understand each other. Therefore, the interaction language is an essential tool to make mutual communication between the users and computer.

Appendix E

Selected user study

Questionnaire for user experience study

Basic Information

Name: 王海

Age: A. 18-27 B. 28-37 C. 38-47 D. 48-57

Job:

Level of using drawing system: A. Beginner B. Meddle C. Expert

On average, how often do you use a drawing system?

- Every day
- Once per week
- Once per month
- Once every 6 months
- Once a year
- Less than once per year

Task:
We are going to provide you a drawing tool for a simple sketching practice. Once you have completed the task, kindly describe your feelings about the drawing practice and the overall experiences as an answer to the following questions;

Second part of survey

We are going to provide you a drawing tool for a simple sketching practice. Once you have completed the task, kindly describe your feelings about the drawing practice and the overall experiences as an answer to the following questions;

PART 1: Definition of interaction (Drawing)

Please circle the numbers which most appropriately reflect your definition of interaction (Drawing)

1.1	I can effectively make drawing	Strongly disagree Strongly agree 1 2 3 4 5 6
1.2	It was easy to learn how to make drawing	Strongly disagree Strongly agree 1 2 3 4 5 6
1.3	It gives error messages that clearly tell me how to fix problems	Strongly disagree Strongly agree 1 2 3 4 5 6
1.4	It is easy to find the information I needed	Strongly disagree Strongly agree 1 2 3 4 5 6
1.5	The interface of this system is pleasant	Strongly disagree Strongly agree 1 2 3 4 5 6
1.6	It has all the functions and capabilities I expect it to have	Strongly disagree Strongly agree 1 2 3 4 5 6

PART 2: Definition of user experience

2.1	I felt comfortable making drawing	Strongly disagree — Strongly agree	1 2 **3** 4 5 6
2.2	I felt more creative while drawing	Strongly disagree — Strongly agree	1 2 **3** 4 5 6
2.3	I was satisfied with how easy it was to Complete the task	Strongly disagree — Strongly agree	1 2 **3** 4 5 6
2.4	I was satisfied with the amount of time to Complete the task	Strongly disagree — Strongly agree	1 2 3 **4** 5 6
2.5	I found the interaction attractive	Strongly disagree — Strongly agree	1 2 3 **4** 5 6
2.6	I like this way of drawing	Strongly disagree — Strongly agree	1 2 **3** 4 5 6

2.7 What were the main things you like about the system

绘图的形式丰富多样

2.8 What were the main things you disliked About the system

~~田~~ 工具的提示~~田~~ 工具不丰富.

2.9 Do you have any other comments ?

工具还需丰富多样.

PART 3: Evaluation of user emotional experience

3.1	When you started working with the program, did you want to continue working with it?	Strongly disagree — Strongly agree 1 2 3 4 5 6
3.2	I enjoyed myself when I was working with the program?	Strongly disagree — Strongly agree 1 2 3 4 5 6
3.3	I feel free to express what I want to do with program	Strongly disagree — Strongly agree 1 2 3 4 5 6
3.4	This program felt nature what I was drawing	Artificial — Natural 1 2 3 4 5 6
3.5	What do you think about this system	Worthless — Precious- 1 2 3 4 5 6
3.6	How do you evaluate the interaction experience	Retro — Novel 1 2 3 4 5 6
3.7	What kinds of tools do you like to make Drawing?	

[illegible]

Questionnaire for user experience study

Basic Information

Name: Jadica.

Age: A. 18-27 B. 28-37 C. 38-47 D. 48-57

Job:

Level of using drawing system: A. Beginner B. Meddle C. Expert

On average, how often do you use a drawing system?

- Every day
- Once per week
- Once per month
- Once every 6 months
- Once a year
- Less than once per year

Task:
We are going to provide you a drawing tool for a simple sketching practice. Once you have completed the task, kindly describe your feelings about the drawing practice and the overall experiences as an answer to the following questions;

Second part of survey

We are going to provide you a drawing tool for a simple sketching practice. Once you have completed the task, kindly describe your feelings about the drawing practice and the overall experiences as an answer to the following questions;

PART 1: Definition of interaction (Drawing)

Please circle the numbers which most appropriately reflect your definition of interaction (Drawing)

		Strongly disagree					Strongly agree
1.1	I can effectively make drawing	1	2	3	4	5	6
1.2	It was easy to learn how to make drawing	1	2	3	4	5	6
1.3	It gives error messages that clearly tell me how to fix problems	1	2	3	4	5	6
1.4	It is easy to find the information I needed	1	2	3	4	5	6
1.5	The interface of this system is pleasant	1	2	3	4	5	6
1.6	It has all the functions and capabilities I expect it to have	1	2	3	4	5	6

PART 2: Definition of user experience

2.1	I felt comfortable making drawing	Strongly disagree Strongly agree 1 2 3 4 5 6
2.2	I felt more creative while drawing	Strongly disagree Strongly agree 1 2 3 4 5 6
2.3	I was satisfied with how easy it was to Complete the task	Strongly disagree Strongly agree 1 2 3 4 5 6
2.4	I was satisfied with the amount of time to Complete the task	Strongly disagree Strongly agree 1 2 3 4 5 6
2.5	I found the interaction attractive	Strongly disagree Strongly agree 1 2 3 4 5 6
2.6	I like this way of drawing	Strongly disagree Strongly agree 1 2 3 4 5 6

2.7 What were the main things you like about the system

个性化

2.8 What were the main things you disliked About the system 没什么特别不喜欢的，除了蜡笔、橡皮工具

2.9 Do you have any other comments ?

No

PART 3: Evaluation of user emotional experience

3.1	When you started working with the program, did you want to continue working with it?	Strongly disagree — Strongly agree 1 2 3 4 5 6
3.2	I enjoyed myself when I was working with the program?	Strongly disagree — Strongly agree 1 2 3 4 5 6
3.3	I feel free to express what I want to do with program	Strongly disagree — Strongly agree 1 2 3 4 5 6
3.4	This program felt nature what I was drawing	Artificial — Natural 1 2 3 4 5 6
3.5	What do you think about this system	Worthless — Precious- 1 2 3 4 5 6
3.6	How do you evaluate the interaction experience	Retro — Novel 1 2 3 4 5 6
3.7	What kinds of tools do you like to make Drawing?	

尼龙笔，颜料，

绘图板.

Questionnaire for user experience study

Basic Information

Name: 李洁

Age: A. 18-27 B. 28-37 C. 38-47 D. 48-57

Job:

Level of using drawing system: A. Beginner B. Meddle C. Expert

On average, how often do you use a drawing system?

- Every day
- Once per week
- Once per month
- Once every 6 months
- Once a year
- Less than once per year

Task:
We are going to provide you a drawing tool for a simple sketching practice. Once you have completed the task, kindly describe your feelings about the drawing practice and the overall experiences as an answer to the following questions;

Second part of survey

We are going to provide you a drawing tool for a simple sketching practice. Once you have completed the task, kindly describe your feelings about the drawing practice and the overall experiences as an answer to the following questions;

PART 1: Definition of interaction (Drawing)

Please circle the numbers which most appropriately reflect your definition of interaction (Drawing)

		Strongly disagree — Strongly agree
1.1	I can effectively make drawing	1 2 3 4 5 6
1.2	It was easy to learn how to make drawing	1 2 3 4 5 6
1.3	It gives error messages that clearly tell me how to fix problems	1 2 3 4 5 6
1.4	It is easy to find the information I needed	1 2 3 4 5 6
1.5	The interface of this system is pleasant	1 2 3 4 5 6
1.6	It has all the functions and capabilities I expect it to have	1 2 3 4 5 6

PART 2: Definition of user experience

2.1	I felt comfortable making drawing	Strongly disagree 1 2 3 4 5 6 Strongly agree	2
2.2	I felt more creative while drawing	Strongly disagree 1 2 3 4 5 6 Strongly agree	2
2.3	I was satisfied with how easy it was to Complete the task	Strongly disagree 1 2 3 4 5 6 Strongly agree	2
2.4	I was satisfied with the amount of time to Complete the task	Strongly disagree 1 2 3 4 5 6 Strongly agree	3
2.5	I found the interaction attractive	Strongly disagree 1 2 3 4 5 6 Strongly agree	2
2.6	I like this way of drawing	Strongly disagree 1 2 3 4 5 6 Strongly agree	2

2.7 What were the main things you like about the system

蜡笔.

2.8 What were the main things you disliked About the system

橡皮

2.9 Do you have any other comments ?

工具还可以增加，如填充．规则图形的绘制．颜色的选择等…

PART 3: Evaluation of user emotional experience

3.1	When you started working with the program, did you want to continue working with it?	Strongly disagree / Strongly agree	1 2 3 4 5 6
3.2	I enjoyed myself when I was working with the program?	Strongly disagree / Strongly agree	1 2 3 4 5 6
3.3	I feel free to express what I want to do with program	Strongly disagree / Strongly agree	1 2 3 4 5 6
3.4	This program felt nature what I was drawing	Artificial / Natural	1 2 3 4 5 6
3.5	What do you think about this system	Worthless / Precious-	1 2 3 4 5 6
3.6	How do you evaluate the interaction experience	Retro / Novel	1 2 3 4 5 6
3.7	What kinds of tools do you like to make Drawing?		

~~毛~~

针管笔.

Questionnaire for user experience study

Basic Information

Name: WEI WANG

Age: A. 18-27 B. 28-37 C. 38-47 D. 48-57

Job:

Level of using drawing system: A. Beginner B. Meddle C. Expert

On average, how often do you use a drawing system?

- Every day
- Once per week
- Once per month
- Once every 6 months
- Once a year
- Less than once per year

Task:
We are going to provide you a drawing tool for a simple sketching practice. Once you have completed the task, kindly describe your feelings about the drawing practice and the overall experiences as an answer to the following questions;

Second part of survey

We are going to provide you a drawing tool for a simple sketching practice. Once you have completed the task, kindly describe your feelings about the drawing practice and the overall experiences as an answer to the following questions;

PART 1: Definition of interaction (Drawing)

Please circle the numbers which most appropriately reflect your definition of interaction (Drawing)

1.1	I can effectively make drawing	Strongly disagree 1 2 3 4 5 6 Strongly agree	
1.2	It was easy to learn how to make drawing	Strongly disagree 1 2 3 4 5 6 Strongly agree	
1.3	It gives error messages that clearly tell me how to fix problems	Strongly disagree 1 2 3 4 5 6 Strongly agree	
1.4	It is easy to find the information I needed	Strongly disagree 1 2 3 4 5 6 Strongly agree	
1.5	The interface of this system is pleasant	Strongly disagree 1 2 3 4 5 6 Strongly agree	
1.6	It has all the functions and capabilities I expect it to have	Strongly disagree 1 2 3 4 5 6 Strongly agree	

PART 2: Definition of user experience

2.1	I felt comfortable making drawing	Strongly disagree 1 2 3 4 5 6 Strongly agree	
2.2	I felt more creative while drawing	Strongly disagree 1 2 3 4 5 6 Strongly agree	
2.3	I was satisfied with how easy it was to Complete the task	Strongly disagree 1 2 3 4 5 6 Strongly agree	
2.4	I was satisfied with the amount of time to Complete the task	Strongly disagree 1 2 3 4 5 6 Strongly agree	
2.5	I found the interaction attractive	Strongly disagree 1 2 3 4 5 6 Strongly agree	
2.6	I like this way of drawing	Strongly disagree 1 2 3 4 5 6 Strongly agree	

2.7 What were the main things you like about the system

~~convienate~~ convenient.

2.8 What were the main things you disliked About the system I have to learn how to use the system firstly.

2.9 Do you have any other comments ?

No.

PART 3: Evaluation of user emotional experience

3.1	When you started working with the program, did you want to continue working with it?	Strongly disagree — Strongly agree 1 2 3 4 5 6
3.2	I enjoyed myself when I was working with the program?	Strongly disagree — Strongly agree 1 2 3 4 5 6
3.3	I feel free to express what I want to do with program	Strongly disagree — Strongly agree 1 2 3 4 5 6
3.4	This program felt nature what I was drawing	Artificial — Natural 1 2 3 4 5 6
3.5	What do you think about this system	Worthless — Precious- 1 2 3 4 5 6
3.6	How do you evaluate the interaction experience	Retro — Novel 1 2 3 4 5 6
3.7	What kinds of tools do you like to make Drawing?	

have lots of fectents. easy to learn how to use and convienent.

Questionnaire for user experience study

Basic Information

Name: Divid Jing

Age: A. 18-27 B. 28-37 C. 38-47 D. 48-57

Job:

Level of using drawing system: A. Beginner B. Meddle C. Expert

On average, how often do you use a drawing system?

- Every day
- Once per week
- Once per month
- Once every 6 months
- Once a year
- Less than once per year

Task:
We are going to provide you a drawing tool for a simple sketching practice. Once you have completed the task, kindly describe your feelings about the drawing practice and the overall experiences as an answer to the following questions;

Second part of survey

We are going to provide you a drawing tool for a simple sketching practice. Once you have completed the task, kindly describe your feelings about the drawing practice and the overall experiences as an answer to the following questions;

PART 1: Definition of interaction (Drawing)

Please circle the numbers which most appropriately reflect your definition of interaction (Drawing)

		Strongly disagree					Strongly agree
1.1	I can effectively make drawing	1	2	3	4	5	6
1.2	It was easy to learn how to make drawing	1	2	3	4	5	6
1.3	It gives error messages that clearly tell me how to fix problems	1	2	3	4	5	6
1.4	It is easy to find the information I needed	1	2	3	4	5	6
1.5	The interface of this system is pleasant	1	2	3	4	5	6
1.6	It has all the functions and capabilities I expect it to have	1	2	3	4	5	6

PART 2: Definition of user experience

2.1	I felt comfortable making drawing	Strongly disagree 1 2 3 4	Strongly agree 5 6
2.2	I felt more creative while drawing	Strongly disagree 1 2 3 4	Strongly agree 5 6
2.3	I was satisfied with how easy it was to Complete the task	Strongly disagree 1 2 3 4	Strongly agree 5 6
2.4	I was satisfied with the amount of time to Complete the task	Strongly disagree 1 2 3 4	Strongly agree 5 6
2.5	I found the interaction attractive	Strongly disagree 1 2 3 4	Strongly agree 5 6
2.6	I like this way of drawing	Strongly disagree 1 2 3 4	Strongly agree 5 6

2.7 What were the main things you like about the system

colour

2.8 What were the main things you disliked About the system 上端窗口略不显眼

2.9 Do you have any other comments ?

PART 3: Evaluation of user emotional experience

3.1	When you started working with the program, did you want to continue working with it?	Strongly disagree — Strongly agree 1 2 3 4 5 6
3.2	I enjoyed myself when I was working with the program?	Strongly disagree — Strongly agree 1 2 3 4 5 6
3.3	I feel free to express what I want to do with program	Strongly disagree — Strongly agree 1 2 3 4 5 6
3.4	This program felt nature what I was drawing	Artificial — Natural 1 2 3 4 5 6
3.5	What do you think about this system	Worthless — Precious- 1 2 3 4 5 6
3.6	How do you evaluate the interaction experience	Retro — Novel 1 2 3 4 5 6
3.7	What kinds of tools do you like to make Drawing?	

Appendix F

Publishing

YI JI, Chek Tien Tan, Ernest Edmonds (2015) Towards Personalized Interfaces for Mobile Applications Using a Natural Text-based Interaction in ‘Proceedings of 13th International Conference on Human-Computer Interaction’ (accepted).

YI JI, Ernest Edmonds, Andrew Johnston, (2014) Establishing Semantic Human Computer interaction using Specific Interaction Language in ‘Proceedings of International Symposium on Interaction design and Human Factors (IDHF)’ (in press).

Damian, Hills, YI JI, Ernest Edmonds, (2013) An Enactive Multimodal Interface for Narrative Construction in ‘Proceedings of 3rd Art and Science Interactional Symposium’, Chinese Building Industry Publishing House, pp 260-266.

YI JI, (2011) Design Language for Creating Meaningful Interaction Design in ‘Proceedings of Second Conference on Creativity and Innovation in Design (DESIRE’11)’, ACM New York, NY, USA pp. 429-430.

YI JI, (2011) Research on Sustainable Design Framework for Human-Centred Interaction Design. ‘Proceedings of an International Conference on Sustainable Design Strategies in a Globalization Context’, Tsinghua University Publishing House pp. 220-225.

YI JI, (2011) Aesthetics in Interactive Interface Design”. In ‘Proceedings of International Conference on Machine Learning and Computing’, IEEE Press pp.523-525.

Exhibition

Sydney Design Festival 2014

Sydney Design festival 2014 presents ‘Design Futures’ that is hosted by Powerhouse museum, Sydney August 16 to August 24 2014.

The 1st Shanghai Design Exhibition

Organized by Shanghai Federation of Literary and Art Circles, Shanghai Municipal. Administration of Culture, Radio, Film & TV and hosted by Shanghai Designers Association and Power Station of Art (Shanghai Contemporary Art Museum) from November 30, 2013 to March 30, 2014.

The 3rd Art and Science International Exhibition
This exhibition was organised by Tsinghua University and held at China National Science and Technology Museum from Nov 1 to Nov 30, 2012

Design pattern

Design patented for “Semantic Interaction Design System” (patent reg. AU2013901013) under examination.