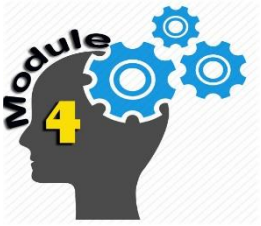

PARADIGMS

MODULE INTRODUCTION:

With the rapid change of modern computing technologies, it is essential to study the history of interactive system design that promotes paradigms for maximum usability. This module introduces the effective strategies for building interactive systems which provide a good perspective for innovative design that the students may apply. Thus, the principles and theoretical models for interaction will be explored by the students for them to come up with a successful design.

MODULE OUTCOMES:

At the end of this module, the students must have:

1. explained various historic human-computer interaction (HCI) paradigms; and
2. identified different paradigms involving HCI.

Lesson 1: Paradigms for Interaction

Objective:

At the end of the lesson, the student must be able to discuss the different paradigms that promote the usability of interactive systems.

Lesson Content:

As explained in Human-Computer Interaction by Dix, A., et al (2004), the primary objective of an interactive system is to allow the user to achieve particular goals in some application domain.[1] With this, studying paradigms are essential to further address the concerns in designing the interactive system:

- how can an interactive system be developed to ensure its usability?
- how can the usability of an interactive system be demonstrated or measured?

The term 'paradigm' coined from Thomas Kuhn's theory of the structure of scientific revolutions as a way to describe waves of research in a field derives. Kuhn describes not an accretive model of scientific knowledge, but one of the successive and overlapping waves in which ideas are fundamentally re-framed. Additionally, in science and philosophy as introduced in Wikipedia, a paradigm is a "distinct set of concepts or thought patterns, including theories, research methods, postulates, and standards for what constitutes legitimate contributions to a field." [2]

Carroll, John M. (1989) cited three distinct paradigms or orientations, to HCI research and application such as evaluation, description, and invention. The author explained that the human factors evaluation and cognitive description paradigms share basic assumptions about the position of psychological analysis in HCI. In which, psychology operates outside the development process even the research prototyping process. HCI is about designing new software tools and user interfaces (Advances in Computers, volume 29, pp. 47-77). [3]

In this module, the paradigms for usability and its principles will thoroughly be discussed based on the study of Dix, A., et al (2004).

Time-sharing (Interactive computing)

In the 1940s and 1950s, the significant advances in computing consisted of new hardware technologies where there was explosive technological growth. By the 1960s, it needs to channel power. J.C.R. Licklider became one of the leading advocates of research into the human-centered application of computing technology at the Advanced Research Projects Agency (ARPA).

The concept of time-sharing was considered as one of the major contributions to come out of the new emphasis in research in which a single computer supporting multiple users instead of batch processing.

Video Display Units

It provides a more suitable medium than paper for presenting computer outputs. In 1962, Ivan Sutherland a young graduate student at the Massachusetts Institute of Technology (MIT), astounded the established computer science community with his Sketchpad program. Sutherland's Sketchpad has realized the capabilities of visual images. The computers could be used for more than data processing, visualizing, and manipulating.

Programming toolkits

Since the early 1950s, Douglas Engelbart desired to use computing technology to complement man's problem-solving activity. As a graduate student at the University of California at Berkeley, Engelbart's idea is to use the computer to teach humans. He introduced the idea of programming toolkits in 1963. Smaller systems created larger systems and components. According to him, providing the right toolkit is the secret to producing computing equipment that aided human problem-solving ability. The right programming toolkit provides building blocks to producing complex interactive systems like carpenters produce beautiful woodwork with the right tools.

Personal computing

In the 1970s, the emergence of computing power called LOGO aimed at the masses invented by Seymour Papert. It is one of the first demonstrations that the powerful tools of the hacker made accessible to the computer novice. It is a graphics programming language for simple children. By typing in English phrases like go, turn left, child programmer could teach the turtle to draw circles, squares, etc. By adapting the graphical programming language to a model that children could understand and use it. Papert demonstrated a valuable maxim for interactive system development where no matter how powerful a system may be, it will always be more powerful if it is easier to use.

Alan Kay and his founding team of researchers at the Xerox Palo Alto Research Center (PARC) developed *Smalltalk* for the personal computing hardware that was just becoming feasible in the 1970s. As technology progresses, Kay's vision of the ultimate personal computer is called Dynabook.

Window systems and the WIMP interface

Humans can pursue more than one task at a time, and accomplish some piece of work or other related pieces of work. The term 'WIMP' stands for Windows, Icons, Menus, and Pointers (or maybe Windows, Icons, Mouse, Pull-down menus). It is a style of the graphical user interface that uses widgets. Also, sometimes called a paradigm for human-computer interaction but not interaction style.

Based on the Interaction Design Foundation, WIMP was invented at Xerox PARC, and this was popularized by the Apple Macintosh and recently available in other varieties such

as the Microsoft Windows operating system, the X Window System, OSF/Motif, NeWS, and RISC OS. [4] Windows used for dialogue partitioning, to “change the topic”.

The interaction based on the WIMP interface is now commonplace. In April 1981, the interaction devices first appeared in the commercial marketplace, when Xerox Corporation introduced the 8010 Star Information System. However, many of the interaction techniques underlying a windowing system were used in Engelbart’s group in NLS and at Xerox PARC in the experimental precursor to Star, the Alto.

Metaphor

In developing the LOGO language to teach children by Papert which was previously discussed in the personal computing, he used the metaphor of a turtle dragging its tail in the dirt. Relating computing to other real-world activity is an effective teaching technique like in:

- file management on an office desktop
- word processing as typing
- financial analysis on spreadsheets
- virtual reality (VR) systems - user portrayed inside the metaphor

Some tasks do not fit into a given metaphor e.g. typewriter keyboard has no space bar which computer keyboard has. Another problem with a metaphor is the cultural bias that it portrays, e.g. meaning of a tick and an x. Xerox star and alto were the first systems to use the concept of metaphors, which led to the spontaneity of the interface.

Direct manipulation

Known as for rapid feedback which is considered as one feature of the interaction technique. This was introduced by Ben Shneiderman in 1982. Shneiderman coined the phrase to the appeal of graphical-based interaction systems such as Sketchpad and the Xerox Alto and Star. He also highlighted the following features:

- visibility of objects
- incremental action at the interface with rapid feedback on all actions
- reversibility encourages exploration without fearing severe penalties
- syntactic correctness of all actions so that every action is a legal operation
- replacement of complex command languages with actions to manipulate directly the visible objects, hence direct manipulation.

In 1984, the Apple Mac PC was first to use in which reduced the chances for syntactic errors. The role of the interface in the model-world metaphor is not so much one of mediating between the user and the underlying system. From the user’s perspective, the interface is the system. Additionally, the WYSIWYG paradigm is somewhat related to the visualization provided by direct manipulation which stands for ‘what you see is what you get’.

Language versus Action

Direct manipulation interfaces indeed make some tasks easier to perform correctly, it is equally true that some tasks are more difficult, if not impossible. Contrary to popular wisdom, it is not generally true that actions do not always speak louder than words in some cases e.g. information retrieval tasks. As such, the image of the interface was interpreted with the following:

- the interface of direct manipulation replaces the underlying system.
- language paradigm
- interface as mediator
- interface acts as an intelligent agent
- programming by example is both action and language
- User issues instructions in natural language
- The interface presents instructions for processing and returns results e.g. Querying the British highway code database about speed limits on various roads.

Hypertext

In 1945, Vannevar Bush introduced the Hypertext. He then described the memex as innovative and futuristic information storage and retrieval apparatus. Whereas, the memex was a desk with the ability to produce and store a massive quantity of photographic copies of documented information. Also, it could keep track of links between parts of different documents.

In the mid-1960s, Nelson describes the phrase hypertext as a non-linear (interlinked nodes) browsing structure. Wherein, the term hypermedia (or multimedia) is used for non-linear storage of all forms of electronic media.

Multimodality

As defined in Wikipedia, multimodal interaction provides the user with multiple modes of interacting with a system in which the interface has several distinct tools for input and output of data. [5] A mode is a human communication channel. A multi-modal interactive system is a system that relies on the use of multiple human communication channels. Each different channel for the user is referred to as a modality of interaction. However, genuine multi-modal systems rely on the simultaneous use of multiple channels for input and output e.g. both audio (beeps), touch (keyboard, mouse..), and visual (screen).

Computer Supported Cooperative Work (CSCW)

In the 1960s, the CSCW is another development in computing that was the establishment of the first computer networks which allowed communication between separate machines. The main difference between CSCW systems and interactive systems designed for a single user is that designers can no longer neglect the social aspects. CSCW systems are built to allow interaction between humans via the computer and so the needs of the many users must be represented in the one product. An example of this is electronic mail, it is the most prominent success. Others are Google docs and Dropbox (groupware).

The World Wide Web

Dix, A., et al (2004) described the worldwide web as a global hypermedia system. Its applications include online help, education, and e-commerce. The design for the world wide web illustrates general hypermedia design but also has its special problems. Dynamic web content can be used for a simple online demonstration or complete web-based business applications. All the computers with internet connection communicate using common data transmission protocols (TCP/IP) and addressing systems (IP addresses and domain names). Just simple, the universal protocols (HTTP) and the mark-up languages (HTML) made publishing and accessing easy.

The worldwide web project was conceived by Tim Berners-Lee in 1989 as a means to enable the widespread distribution of scientific data generated at CERN and to share information between physicists worldwide.

Agent-based Interfaces

In reality, software agents act on behalf of people in the electronic world. Agents can perform repetitive tasks, e.g. web crawlers and e-mail filters. They can learn from the user's actions. It is indeed difficult to find a language suitable for the user and agent, in the sense that the user may not receive immediate feedback.

Some of the agents use artificial intelligence techniques to learn based on the user's actions. For example, Eager (the cat icon) HyperCard. It is the earliest example of AI Agent learning based on user actions. Eager had a clear 'embodiment', the cat on-screen User free to ignore the agent. In contrast, MS Excel, the intelligence in this is not embodied but diffused wherein 'the system'.

Agent-based systems include aspects of both language and action paradigms. Old command-based systems acted as intermediaries. Agents act on the user's behalf. Unlike with direct manipulation emphasizes the user's action-based or commands performed on "world" representation. The agent is usually acting in an environment the user can also act upon.

Ubiquitous Computing

Rab Nawaz Bashir, et al (2014) defined ubiquitous computing as a "new computing paradigm with seamless integration of hundreds and thousands of self-communicating small-scale computers and intelligent devices into the user environment and daily life activities." [6]

Ubiquitous computing is also known as pervasive computing the same as sensor-based/context-aware computing. Computing is made to appear anytime and everywhere. Thus, it is the most currently active research area in HCI.

Sensor-based and Context-aware Interaction

This interaction is usually used in everyday activities as a source of context information. Humans are good at recognizing the “context” of a situation and reacting appropriately. Whereas, interaction in context-aware computing is more implicit. By using heuristics and other semi-intelligent in the computer or more accurately the sensor-enhanced environment helps to predict what would be useful for the user. Automatically sensing physical phenomena (e.g., light, temp, location, identity) becoming easier.

Teaching Delivery (TLAs)

- Lecture and Class Discussion (Synchronous and Asynchronous Learning)

Assessment:


The students will be given a summative test using zip grade.

Reference:

1. Dix, A., Finlay, J., Abowd, G.D., & Beale, R. (2004). Human-Computer Interaction, 3rd Edition. Hillsdale, NJ: Prentice-Hall
2. Paradigm. Retrieved from https://en.wikipedia.org/wiki/Paradigm.
3. Carroll, John M. (1989). Evaluation, Description and Invention: Paradigms for Human-Computer Interaction. Advances in Computers (volume 29, pp. 47-77). Academic Press, Inc. Published by Elsevier Ltd. Retrieved from https://www.sciencedirect.com/science/article/pii/S006524580860532X
4. WIMP. Retrieved from https://www.interaction-design.org/literature/book/the-glossary-of-human-computer-interaction/wimp
5. Multimodal Interaction. Retrieved from https://en.wikipedia.org/wiki/Multimodal_interaction
6. Rab Nawaz Bashir, Salman Qadri, Rana Muhammad Saleem, Muhammad Naeem, and Yasir Ghafoor. (2014). Human-Computer Interaction (HCI) in Ubiquitous Computing. International Journal of Innovation and Applied Studies (vol. 9, no. 2, pp. 534–540)