ZOOMING OUT FROM THE DESKTOP

The use of metaphors in Human-Computer Interface design

Giuseppe Costanza
MA Communication Design,
Digital Media 2008

Zooming out from the desktop

The use of metaphors in Human–Computer Interface design

Giuseppe Costanza
MA Communication Design, Digital Media
2008

Abstract

The desktop/office metaphor is at the base in the interface of the majority of computers currently in use. The desktop metaphor was introduced in late 1970s to make computers friendlier to office workers. Today this type of interfaces and metaphors are not adequate with computer users needs. This dissertation explains why this obsolete concept is still in use. Then some alternative, emerging interfaces are presented. The last chapter then describes the One Laptop Per Child project as an example of how interface design can successfully take different routes from what is considered the industry standard.

Table of contents

Preface

This dissertation deals with screen based media and was written and designed mainly on a screen. Reading it as a printed book could appear in contrast with the content and the design process but I strongly believe that paper and books, as a medium, have many qualities that any screen based device cannot equal. Books are tangible, easy to read and carry, they allow to easily take notes and their interface is universally known. At the same time this dissertation contains references to multimedia content that can hardly be rendered in printed form. Instead of discarding the book I decided to provide a small screen based device to support the printed medium without affecting its mentioned qualities. The mobile phone that comes with this dissertation gives access to multimedia documents that complement the discussed topics. Every time on a page appears a rounded black symbol similar to the one at the end of this paragraph, just point the mobile phone camera on it and the screen will display the related content.

D-touch, the technology used to link content on the mobile phone with the book was developed and provided by Enrico Costanza, as part of his research on visual markers. More information can be found on http://www.d-touch.org/

I thanks Enrico, Rathna Ramanathan and Axel Vogelsang for their support in developing this dissertation and all the other people that helped me.

Introduction

Humans and computers use very different languages for communicating. Human communication is rich, people use voice, writing, gestures, facial expressions. Computers work in a very different way, they are based on a simple binary code. When people and computers have to communicate with each other they need to find a point of contact, a common language that can be interpreted by both. The point where humans and computers meet is named Human-Computer Interface. An interface has to allow users to instruct computers and receive the produced results. Human-Computer Interfaces can have many different forms, today the most common interfaces are based on mouse and keyboard as input devices and screens as output devices. Human-Computer Interfaces should allow people to easily control computers expressing their will in a natural way, preventing errors and misunderstandings. The main issues concerning human-computer interface design are discussed in this dissertation. Common interfaces will be presented, as well as, possible alternatives to them.

The first chapter briefly exposes the history of human computer interfaces. In computer history processing speed has always increased, while prices have fallen down. This tendency allowed to employ an increasing amount of computing power in interfaces, making computer's language easier for humans and shifting the communicational effort from people to computers. (Dourish 2001) The most important moment in human computer interface history is the introduction of graphical interfaces and the adoption of the desktop metaphor. The desktop metaphor was presented to the general public at the end of 1970s; (Moggridge 2007) it allowed students, office workers and businessmen to approach computers for the first time. The desktop metaphor perfectly suited users needs of that time.

Now, 30 years after the desktop metaphor introduction, the way people use computers is very different. Computers are used to play videogames, listen to music, watch movies, browse the web, communicate with other people; they are used at home on a table, on the sofa, on the bed, in cars. But our indispensable assistants are still communicating with us using the same desktop metaphor. In the first chapter it is pointed out that not much has happened in human computer interface history, since the introduction of the desktop metaphor. Most computer users experienced folders and file cabinets first as computer icons and later as real world tools to organize documents. This dissertation argues that the way computers organize information is constrained to concepts that are not up-to-date.

In the second chapter it is explained why the desktop metaphor was chosen. Some considerations, supported by authoritative points of view, about the use of metaphors in interface design are presented. Metaphors are powerful linguistic instruments used to compare something unknown to something known, but if used in a wrong way they can be misleading. At the end of the chapter it is illustrated the reasons why computer industry is anchored to the use of out-of-date concepts and metaphors.

The third chapter introduces some commercial and experimental interfaces aimed at improving or replacing the desktop metaphor. If the desktop metaphor is still in use it means that any alternative is good enough to replace the current paradigm. Then zooming interface paradigm is presented as an emerging alternative to the desktop metaphor. Zooming interfaces give users the possibility to easily control the level of displayed information, zooming on digital items as if they laid on an infinite plane. The zooming paradigm could mark a new big switch in human computer interface history. Also this chapter explains how zooming interfaces can better exploit the power of digital tools since they are less anchored to real world metaphors. Some zooming interface concepts already appears in current operating systems, but their full potentials have not been explored for the general use, yet.

In the last chapter, the One Laptop Per Child (OLPC) project is presented as a positive example of how interface design can take different routes. The OLPC project realized a laptop addressed at developing countries children. OLPC designers decided to realize an interface well suited for intended users, discarding the desktop metaphor and obsolete interface design conventions. OLPC interface is based on a zoom metaphor and it widely relies on spatialization as a universal concept understandable by different cultures.

1 History of computer interfaces

This chapter briefly illustrates the evolution of computer interfaces, from punched cards to the most recent graphical interfaces. The introduction of Graphical User Interfaces (GUI) is a fundamental event in the history of computing. The reasons that brought to the conventions in use nowadays are explored more in depth.

1.1 The 3 eras of computer interfaces

Batch computing

Looking at human-computer interface history it is possible to notice a trend in making computer interfaces more abstract and natural for people. When computers first appeared, in 1940s, they were very expensive and not very powerful; their calculation power was considered precious. Early computers were mainly used for military or commercial purposes. From an economic point of view, computers working time was more expensive than human workers time, so interfaces were designed to avoid computational efforts in interpreting and compiling the code, leaving all the effort to human operators. (Dourish 2001, pp.1-2) Software and dataset were inputted through piles of punched cards that

Fig.1 Punched Cards and IBM *29 card punch* punching machine.

had to be prepared on special external machines. (Fig.1) Due to the complex syntax and the low usability level of punching machines, there was a high error rate; completing a single job took hours or entire days. Following Eric S. Raymond's classification,[1] this first era of computing is defined *batch computing*. Batch computers were used approximately from 1945 to 1968. The following eras are *command-line* and *graphical*. (Raymond 2004)

Command-line

Around 1969 the early Command-line interfaces became available. Their interaction model was based on requests expressed as textual commands in a specialized vocabulary. That type of interfaces allowed a near real-time feedback making it possible for users to interact with computers in a much more powerful way. It is important to notice that earliest command-line systems used teletypes as input devices. Teletypes were a mature technology in use since the beginning of 20th century and played an important role in making computers familiar to many users. While early command line systems used printers to output responses, in the mid-1970s video display started becoming widespread, allowing a faster interaction and a sensible cut of costs for ink and paper. Command-line interfaces were much easier to use than batch computers but complex commands syntax had to be learnt by users. (Raymond 2004)

1 Eric S. Raymond is an Internet developer and writer, is one of the main representative of the open source movement.

Video 1 Engelbart presentation of direct manipulation on a graphical interface.
Point the mobile phone on the symbol below to watch the video.

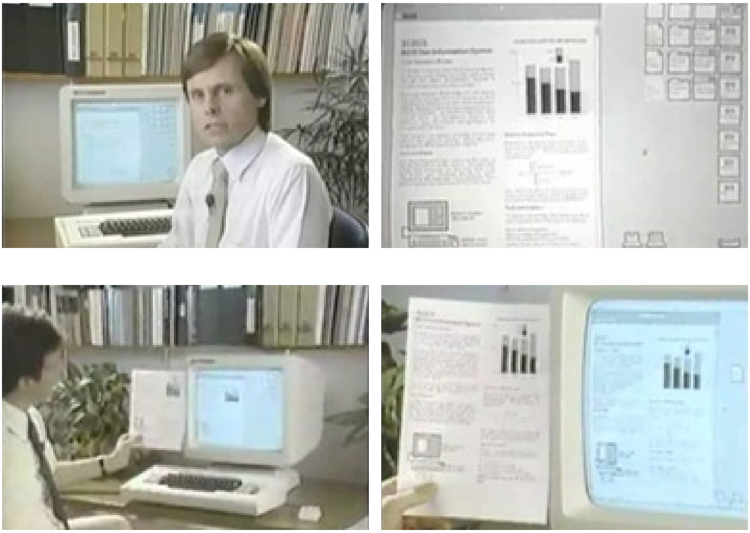
Graphical interfaces

The third and last era of computer interfaces is the Graphical Interfaces era. Graphical interfaces were formed by the type of interfaces commonly used nowadays. Graphical User Interfaces (GUI) were first introduced by Douglas Engelbart.[1] In 1968, when Engelbart was a researcher at Stanford Research Institute he presented for the first time direct manipulation on a graphical interface using a computer with a screen, a standard typewriter keyboard and a small rolling box called *mouse*. He demonstrated how easy it was to edit text documents thanks to that interface. In Engelbart's system it was possible to correct, format and print a text in a way never imagined before by most of the people. (Video 1) (Moggridge 2007)

In 1970s some of the most influential researchers moved from Stanford to a brand new research centre realized in Palo Alto by the paper-copier company Xerox. At Palo Alto Research Center (PARC) were introduced some of the most remarkable computer innovations: laser printers, Smalltalk,[2] WIMP interfaces[3] and the Desktop metaphor. (*Palo Alto Research Center* 2008, *Encyclopædia Britannica,* viewed 8 October 2008, <http://www.britannica.com/EBchecked/topic/440290/Palo-Alto-Research-Center>)

The idea of using an office/desktop metaphor in computer interfaces was introduced by Tim Mott and Larry Tesler to help graphic designers in understanding how to behave with electronic data.

1 Engelbart is also considered the inventor of the mouse.
2 Smalltalk is one of the first object oriented programming language.
3 Interfaces based on the use of Windows, icons, menus and point device.



Video 2 Xerox Star, the first commercial computer with a graphical interface.
Point the mobile phone on the symbol below to watch the video.

As Tim Mott tells in an interview: (Moggridge 2007, p.53)

> I was thinking about what happens in an office. Someone's got a document and they want to file it, so they walk over the file cabinet and put it in the file cabinet; or if they want to make a copy of it, they walk over to the copier and they make a copy of it; or they want to throw it away, so they reach under their desk and throw it in the trash can.
> […] What ended up on the bar napkin was what Larry and I called "Office Schematic." It was a set of icons for a file cabinet, and a copier, or a printer in this case, and a trash can. The metaphor was that entire documents could be grabbed by the mouse and moved around on the screen.

Mott and Tesler's "Office Schematic" principle is exactly what is in use in today computers.

1.2 Xerox Star

Few years later, in 1981, the desktop metaphor reached the market for the first time. The first commercial product using that type of interface was the Xerox Star. Star computer was addressed to businessmen and organizations. The system had to be used to produce, organize and distribute different types of business documents such as presentations, memos and reports. Target users were people without a background in computer science. The system had to set them free from technical issues. All the applications needed by target users were included in the system and automatically associated with corresponding documents. Thus users did not have to worry about starting the right application for a given task or file. (Video 2) (Johnson et al. 1999, *David Curbow*, viewed

Fig.2 Page 6 of the original brochure released in 1983 to launch Apple Macintosh. The pictures show the different users Macintosh was addressed to.

29 April 2008, <http://www.digibarn.com/friends/curbow/star/retrospect>)

Despite its innovative concept, Star was a commercial failure, mainly because of the scarce power, causing slow speed, a high price and inept marketing strategies. (Raymond 2004)

Star flop was one of the main reasons of brain drain from PARC. Around 1980, the most creative minds at PARC were anxious of seeing their ideas moving from the lab to the market; Xerox instead was more interested in maintaining dominance in the copier market rather than producing computers. (Moggridge 2007, p.71)

1.3 Apple Macintosh

The history of the GUI at this point moves to the world famous computers company, founded by Steve Wozniak and Steve Jobs: Apple Computers. (Mary Bellis 2008, *The New York Times Company*, viewed 29 April 2008, <http://inventors.about.com/od/ cstartinventions/a/Apple_Computers.htm>)

Larry Tesler was one of the researchers that moved from PARC to Apple. At Apple Tesler started working with Bill Atkinson, their creative partnership produced some of the most significant innovations still in use in GUIs, such as pull-down menus and dialog boxes. (Moggridge 2007)

After a first commercial failure with the Lisa, in 1984 Apple released the Macintosh. The Macintosh was the first commercially successful computer using a *point and click* graphical interface controlled through a mouse. The success of Apple Macintosh was due to the whole design concept: the price was affordable ($2,495), the case was compact and light (8Kg, less than many portables of the time), the interface was simple and user friendly. (Fig.2)

Fig.3 Nextstep screenshot. Nextstep introuduced the sharp three-dimensional look and the x symbol to close windows that became a standard in following operating systems.

(Marcin Wichary 2004, *Marcin Wichary*, viewed 29 April 2008, <http://www.aresluna.org/attached/computerhistory/articles/macintosh20yearslater>)

As Eric S. Raymond (2004) points out, Apple Macintosh proved that a well designed GUI could make a difference in the mass market. While the Alto interface was a 'mere laboratory toy' (Raymond 2004), Apple transformed it in a usable interface thanks to five years spent on interface psychology and design investigation.[1] It was the first time that so much effort was put in computer interface design. In the early 1980s, other competitors introduced GUIs in the market such as Amiga Kickstart or Microsoft Windows, but they were only rough clones of the PARC GUI with all of its limitations. The Apple Macintosh soon became the reference for all the future GUIs.

1.4 Microsoft Window's rise

In the second half of the 1980s minor innovations were introduced by other companies; the *Dock* appeared for the first time in 1987 in Acorn Computers operating system. The Dock is a shelf placed at the bottom of the screen, used to put shortcuts to most used programs and files. (Jeremy Reimer 2005, *Ars Technica*, viewed 4 August 2008 <http://arstechnica.com/articles/paedia/gui.ars/1>)

1988 Nextstep operating system introduced a sharp three-dimensional look to all the interface components and the X symbol to close windows. (Fig.3) (Jeremy Reimer 2005, *Ars Technica*, viewed 4 August 2008 <http://arstechnica.com/articles/paedia/

1 For example The Xerox Star mouse had three buttons and a very complex interaction scheme, for Apple's mouse Jef Raskin designed a much simpler one-button mouse. (Raskin 2000, pp. 207-209).

Fig.4 Mac OS X screenshot. Windows have a colorful, jelly look, the bottom of the screen presents the Dock.

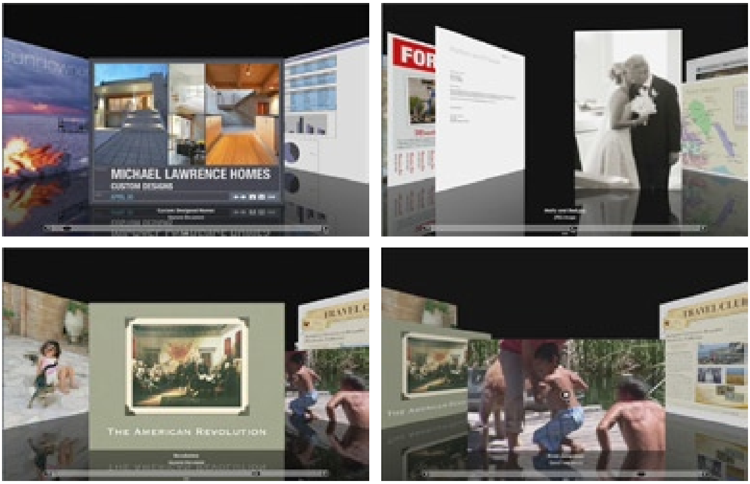
gui.ars/1>) Both features became a standard in following OS.

In 1990s, most of the companies left the market. Microsoft became the market leader, thanks to Windows 3 released in 1990 and Windows 3.1 released in 1992, while Apple lost its innovative design approach and was relegated to a niche. Raymond (2004) claims that Microsoft 1990s' success is due to the care they took in adding in its OS features the customers require. Another reason of Windows 3 success is the widespread third-party support that meant a huge quantity of available software. (Mary Bellis n.d., *The New York Times Company*, viewed 4 August 2008, <http://inventors.about.com/od/mstartinventions/a/Windows.htm>)

In 1995, Windows 95 introduced two new features the *Start* menu to access all the installed programs and the *Taskbar* to switch between open windows. Microsoft supremacy was mainly based on trade agreements. The easy-of-use principle was a key one in Windows Operating systems, however in 1990s there were not many innovations in interface design.

1.5 Mac OS X

In 2001 Apple released the 10th version of its operating system. Mac OS X is the first Mac OS version that introduced a quantity of new features after a period poor in innovations. The most evident breakthrough introduced by OS X was its visual appearance clearly in contrast with the industry standards. The classical grey and sharp interface was replaced by a shiny, colourful, jelly look. More than just a visual make-up the new interface presented some functional innovations such as transparency and shadow effects to highlight windows stack order. Another crucial element of the new interface was the *Dock*, a multipurpose bar that provides easy access to some applications and folders, displays



Video 3 Cover Flow allows to visually browse through huge quantity of folders and documents in a fast way. Point the mobile phone on the symbol below to watch the video.

information about running applications and holds windows in their minimized state. (Fig.4) Besides its strong visual appearance, the Dock was already present in other operating systems.[1] In future releases of os x, other interesting features were added, such as the powerful textual search tool named *Spotlight* and the *Dashboard*. The Dashboard is a system application that gives easy access to some useful tools such as a clock, a calendar, a calculator and it allows a better integration of the Internet inside the operating system, thanks to some tools able to retrieve information from the Net (weather forecasts, stock market). (John Siracusa 2005, Ars Technica, viewed 5 August 2008 <http://arstechnica.com/reviews/os/macosx-10-4.ars/1>)

The last version of os x named Leopard, released in 2007, introduced two innovative features. The first one is *Cover Flow*, a three dimensional interface that allows to visually browse trough huge quantity of folders and documents in a fast way. (Video 3) Cover Flow is in opposition with the actual trend of using textual searches to easily retrieve information locally and on-line. The other innovation introduced in Leopard is the *Time Machine*, an easy to use backup and recovery system. The backup function is completely automated, the only choice is the hard drive where to save backups. Time Machine introduces the possibility to browse through all the past versions of files contained in a folder. The interface is stylish and effective, it allows to scroll in a 3D view the content of the selected folder and to retrieve it. (John Siracusa 2007, *Ars Technica LLC*, viewed 5 August 2008 <http://arstechnica.com/reviews/os/mac-os-x-10-5.ars/14>)
Some of the new features introduced in Mac os x show a tendency to replace the desktop metaphor, in favour of abstract tools that

1 In Acorn Computers os, cf. paragraph 1.4

allow users to better organize and retrieve digital information. However, the main elements of the interface and the interaction model are not so different from early GUIs designed 30 years ago. Looking at the short history of graphical interfaces, it is easy to notice how an early creative period was followed by one without significant innovations. The result is a lack of proper tools to use the always increasing powerful computers.

The next chapter explains more in depth why the Desktop metaphor was so successful, since its introduction in the market, why it is still in use nowadays, as well as, what its limits are.

2 The Desktop Metaphor

Today the most widespread operating systems are based on a desktop metaphor. The office-desktop visual metaphor is so common for most of the computer users that it appears to be the obvious way of interacting with them. This chapter illustrates the reasons why Xerox PARC researchers decided to introduce this metaphor, as long as, its strengths and weaknesses.

2.1 Why a desktop metaphor

As outlined in the previous chapter, the desktop metaphor was introduced for the first time in Xerox Star interface.[1] The Star was the first computer meant for an audience of office workers. At that time computers were mainly used for research, a lot of technical knowledge was required to interact with them. In an office environment, computers were used for productivity tasks such as business reports and presentations. Users were businessmen and secretaries without any skill in computer science but interested in using the powerful digital tools to accomplish their tasks faster and better. (Johnson et al. 1999, *David Curbow*, viewed 29 April 2008, <http://www.digibarn.com/friends/curbow/star/retrospect/>)

1 Cf. paragraph 1.2

Tim Mott came out with the idea of using metaphorical icons to compare computer's objects with office's elements. (Moggridge 2007) He represented computer files and directories with graphical icons of folders and paper documents; the possibility of deleting files and retrieving them was represented as a trash can, computer windows were stackable as sheets of paper and they could be unrolled as scrolls of papyrus. The computer desktop, as a real desktop, was the main working space where to arrange working documents.

The idea was successful. The metaphor used was perfectly suitable for people approaching computers for the first time in an office environment. The graphical interface enabled users to perform common tasks just pointing and clicking on visual icons. This was a great improvement compared to typing textual commands. Moreover, the desktop metaphor focus on documents editing. Other systems, even if using GUIs, were based on a tool metaphor.[1] The user had to invoke an application, the tool, and then specify which file to edit with it. (Johnson et al. 1999, *David Curbow*, viewed 29 April 2008, <http://www.digibarn.com/friends/curbow/star/retrospect/>)

2.2 Criticisms to the desktop metaphor

Today computers can store thousands of multimedia files and access an infinite amount of information on the Internet. The desktop has become just a tiny visual space where few files can be stored, all the other documents have to be recalled remembering the textual categories of a vast folders/sub-folders structure

1 Systems such as Salltalk-80, Cedar, and various Lisp environments. (Johnson et al. 1999, *David Curbow*, viewed 29 April 2008, <http://www.digibarn.com/friends/curbow/star/retrospect/>)

or typing the exact file name (Johnson 1997, pp.78-79).[1] Most of the benefits of using a graphical interface have disappeared and there is a tendency to go back to textual interfaces.[2]

Another problem is that today computers are used by different categories of people, not only by office workers. For example, Cordell Ratzlaff [3] (cited in Moggridge 2007, p.149) points out that today children begin using computers at a very early age. Metaphors such as folders or file cabinets make no sense for them, since they first experience those concepts on the screen and only in a second time in the real world.

Many HCI (Human-Computer Interaction) researchers argue that the desktop metaphor is not a good model for interface design. Most of them believe that any type of metaphor should be avoided.

Ted Nelson[4] (1990) states that the desktop metaphor is totally unsuccessful. He argues that 'The alternative to metaphorics is the construction of well-thought-out unifying ideas, embodied in richer graphic expressions that are not chained to silly comparisons.' (Nelson 1990, p.237)

Brenda Laurel (1993), instead, points out that metaphors are useful when there is a corresponding behaviour between the two poles, but they become problematic when the interface does not

---

1 Some software allow full-text search on documents but most of the files have as the only reference the file name.

2 Textual search engines, such as Google Search or Apple Spotlight, are widely used to retrieve information both on line and on local hard disks.

3 Cordell Ratzlaff is the manager of Human Interface Group at Apple.

4 Ted Nelson is an American sociologist, philosopher, and pioneer of information technology.

show what users usually find in the referring domain. In a similar way problems arise when something in the interface works in an unexpected way according to the metaphor. The use of the trash can to ejects disks can be an example.[1]

Jenny Preece[2] (Preece et al. 1994) makes a distinction between verbal metaphors and interface metaphors. While verbal metaphors invite people to compare similarities and differences between the system and the familiar domain, interface metaphors confuse the two parts and the metaphor becomes the interface. Verbal metaphors can be a starting point to learn a new system, developing a good mental model of it. Instead, interface metaphors become the learned model and the structural aspects of the system remain unaware.

According to Norman (1998) metaphors are always wrong. The only exception to this is when the properties of the metaphor source domain and the unfamiliar target are closely related. In a similar way to Preece, Norman argues that metaphors can be useful in the early stages of learning, but later on prevents users from building a correct mental model of the system. The solution, according to Norman (1998), is using a clear conceptual model. A conceptual model is defined as 'a story that puts together the behaviour and appearance of a device in a sensible, comprehensible pattern' (Norman 1998, p.177). A good conceptual model allows people to understand the different controls and alternative actions taken without knowing how the system really works.

Cooper (2003) suggests to replace the use of metaphors with 'idiomatic interfaces', as he calls them. Idiomatic user interfaces are based on the use of non-metaphorical visual and behavioural

---

1 Cf. paragraph 2.4

2 Jenny Preece is an online community researcher and Professor and Dean at the College of Information Studies, U. of Maryland.

idioms to accomplish goals and tasks. While metaphors are based on user's intuition, idiomatic interfaces have to be learnt. Cooper's thought (2003, p.250) is based on the evidence that idiomatic expressions such as 'beat around the bush' or 'cool' make sense only if people have been taught them. It seems that human mind has an attitude in understanding and remembering a huge quantity of idioms. According to Cooper (2003) idioms are already used in today interfaces, actually most of the intuitive elements in GUIs are visual idioms. The way windows, drop-down menus and mice work is not based on real-world metaphors but it is learnt idiomatically.

Cooper (2003) argues that the success of the Macintosh interface does not derive from the use of the desktop metaphor but from several other reasons as the easy to use mouse driven interface, the possibility of visually manipulate objects in a WYSIWYG (What You See Is What You Get) way or the high resolution display that allowed reliable previews of printed output.

Using Cooper's idiomatic interface concept it is possible to explain why people who are not familiar with real-office tools, as children, are able to easily use a desktop interface. They learn the way tools work in an idiomatic way, without intuiting the connection with the physical world.

Gelernter (2000) suggests to abandon current operating systems based on folders, named files and textual search engines.

> Modern computing is based on an analogy between computers and file cabinets that is fundamentally wrong and affects nearly every move we make. (We store "files" on disks, write "records," organize files into "folders" — file-cabinet language.) Computers are fundamentally unlike file cabinets because they can take action.

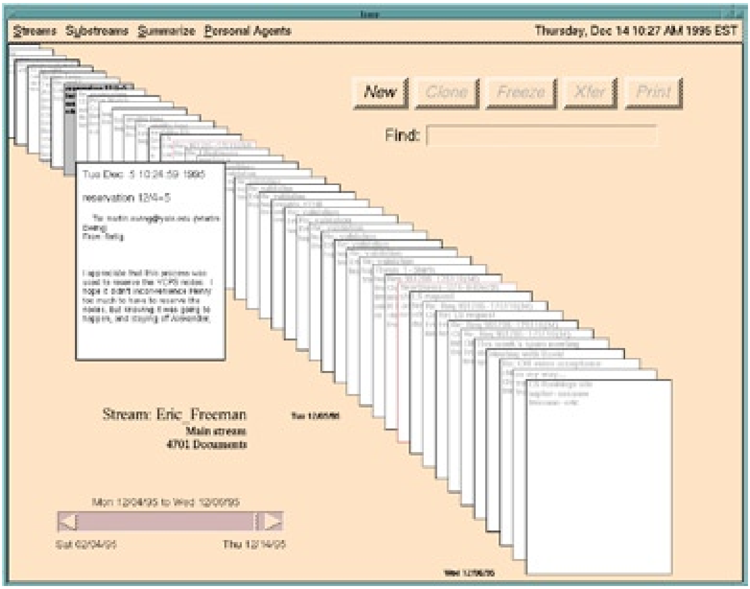


Fig.5 Lifestreams screenshot. Experimental interface based on a stream of documents organized in chronological order.

> Metaphors have a profound effect on computing: the file-cabinet metaphor traps us in a "passive" instead of "active" view of information management that is fundamentally wrong for computers.

Following Nelson's thought, Freeman and Gelernter (2007, p.23) state that, instead of using metaphors, interfaces have to be built around simple unifying ideas, leading to new organizing strategies. In 1994, Freeman and Gelernter designed *Lifestreams,* an experimental interface based on a stream of documents organized in chronological order with easy to use tools to search and organize information. (Fig.5) (Freeman & Gelernter 2007, p.20) As the two authors point out, explaining the data structure of their interface, 'a lifestream is a time-ordered stream of documents that functions as a diary of your electronic life'. (Ibid., p.26) Freeman and Gelernter refuse the use of metaphors in interface design, but they use a diary metaphor to explain how their system works. The point is that Lifestreams is based on very general metaphors that do not trap the design in obsolete concepts.

Drawing a conclusion, many HCI researchers warn about the risks of using metaphors in interface design, but, as Lakoff and Johnson (1980) point out, metaphors are fundamental tools in human mental processes. Actually most of human conceptual systems are based on metaphors. Lakoff and Johnson (1980), for instance, use the example of the conceptual metaphor 'argument is war'. Expressions such as 'win an argument' or 'attack weak points' are not only linguistic idioms but are evidences that people think about arguments as wars from a conceptual point of view. If other metaphoric expressions were used when referring to arguments, such as 'argument is a dance', the mental model of arguments would be differently shaped by the metaphor. Lakoff and Johnson (1980) argue that most of the mental concepts are understood in a metaphorical way; the only concepts that are understood without the mediation of metaphors are the ones based on the spatial experience related to the human body. Most of the other concepts derive from that physicality. For example, social status is defined in terms of high status (up) and low status (down); happiness, health and well-being are defined as high while the contrary is low ('high quality work', 'things are at an all time low'). (Lakoff & Johnson, p.16)

In interface design, metaphors can be successfully used to develop a vocabulary to speak about abstract concepts in a comfortable way but, as Cooper (2003) suggests, it is important to avoid bending the interface to fit a metaphor, or using Johnson's words 'replace the good faith of user friendly metaphors' with the 'hysteria of total simulation'. (Johnson 1997, p.60)

For example, in on-line shopping context, the shopping cart metaphor is well suited to refer to the virtual place where it is possible to review the list of items a user is interested in buying. The on-line shopping cart, as the real trolley, is the place where users put their goods in before paying them. It is easy for people with previous experience of shopping malls, to understand what properties the virtual cart has in common with real world carts. Usually, in e-commerce web sites there is no visual reference to shopping malls or real trolleys. Using the word 'shopping cart' is enough to get the idea: users will just see a page with the list of all the selected items where it is possible to check the due amount of money, taking something off and checking out. In a similar way, it is easy for users to understand that clicking on the button 'add to cart' in a product page will add the product to the list of chosen products.

2.3 Magic Cup and Microsoft Bob

To support the theories illustrated above, two examples of how metaphors can be used in an extremely negative way are presented. Magic Cup and Microsoft Bob were two commercial interfaces designed in mid 1990s, with the aim to expand and enrich the desktop metaphor. Magic Cup was an interface for handheld digital assistant introduced in 1994. Magic Cup was based on an extended office metaphor, every aspect of the interface was represented by its physical counterpart creating a simulation rather than a useful interface. (Fig.6) That use of a pervasive metaphor prevented users from benefiting the power of computers versatility, remaining attached to physical world constraints. Speaking about Magic Cup, Cooper (2003, p.253) explains:

> It may seem clever to represent your dial-up service with a picture of a telephone sitting on a desk, but it actually imprisons you in a limited design. The original maker of the telephone would have been ecstatic if they could have created a phone that let you call your friends just by pointing to pictures of them.

In 1995 Microsoft introduced Bob, an interface based on a living room metaphor drawn with a cartoon style. (Fig.7) Bob was meant for neophytes but, as Johnson (1999) points out, Bob was much like a simulation, rather than a productive interface: instead of highlighting the potentials and power of computers it was hooked to physical limits. Computer novices, more than others, should be guided in understanding the differences of computer tools from their physical counterparts. (Johnson 1999, p.61)

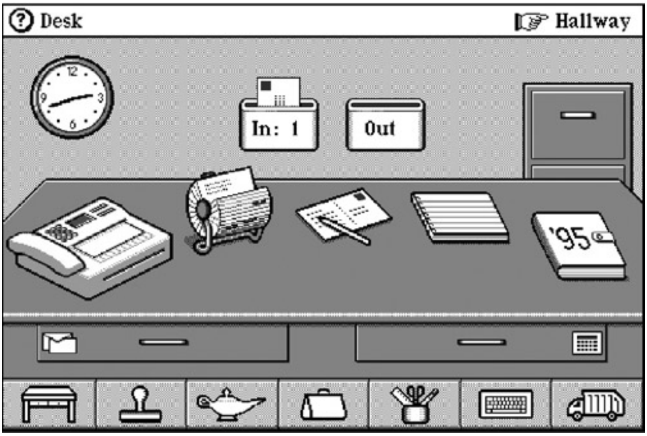


Fig.6 Magic Cup screenshot. Magic Cup was based on an extended office metaphor, every aspect of the interface was represented by its physical counterpart.

Fig.7 Microsoft Bob screenshot. Interface based on a living room metaphor rendered with a cartoon style.

Magic Cup and Microsoft Bob resulted totally unsuccessful. When using metaphors it is important to maintain a good balance between real world resemblance and abstraction; in this respect mainstream desktop interfaces can be considered successful.

2.4 Why is the desktop metaphor still in use?

As was presented in the previous paragraphs many alternatives to the desktop metaphor were theorized, in the next chapter will be illustrated some existing alternatives to the classical desktop interface. Probably, none of them can be seen as a definitive solution to user interface design but they try to solve some of the conceptual problems and practical lacks that current desktop interfaces present.

If the desktop metaphor and WIMP interface are not good for users' need and some alternatives to them already exist, why then are they still in use?

Changing something that works is always problematic even in a lively field as technology. The social scientist Herbert Simon (cited in Tidwell 2005, p.12) defined the behaviour of people in social and economic situations as *satisficing*. The word comes from the union of *satisfying* and *sufficing*. It explains how people prefer to maintain what they already have, if they consider it to be good enough for their needs and their alternatives have a high cost in terms of time and effort. This behaviour is valid also when referring to interfaces. One of the first rules in interface design is maintaining consistency with conventions and standards. (Tidwell 2005) A good example of this trend is the trash icon to eject disks on Mac OS. On Macintosh operating system to eject an external disk users have to drag its icon on the trash can icon, the same icon used to delete files.[1] That counter-intuitive behaviour has a historical reason: the very first Macintosh had no hard disk and only a floppy disk drive. At that time, the trash can was used to delete the icon of a previously inserted diskette from the desktop. Programmers thought that ejecting the diskette was a desired task for users who deleted the disk icon from the desktop. That behaviour aimed at making things easier for the average users, following Apple philosophy. The convention of using the trash can icon to eject disks has been left unchanged in the following releases of Mac OS and it is still in use nowadays. (Erickson, 'Creativity and Design', in Laurel 1990, p.2)

Differently, Raskin ascribes interfaces inertia more to industries rather than to users behaviour: (Jef Raskin 2005, *Jef Raskin web site*, viewed 9 October 2008, <http://jef.raskincenter.org/humane_interface/summary_of_thi.html>)

> Normal human inertia makes it difficult to effect sweeping changes, even when the need for them is clear. It is widely recognized by users and commentators that present-day interfaces and their supporting software systems are not satisfactory, yet to many people inside the industry the need to revise our present interface protocols does not seem pressing.

Designing a brand new interface requires a huge effort in terms of programming and user testing. Furthermore, new software and computer systems are meant for people with a previous experience with computers. This audience is thought to prefer already known interfaces.

---

1 In Mac OS X when a disc is dragged the trash icon becomes an arrow to avoid the ambiguity between deleting and ejecting, but this behaviour still has no consistency.

In the last years, hints of a switch in the paradigm can be seen in interface design. The increasing availability of ubiquitous digital devices and the growing importance of networking and social computing, are determining some important changes in the way computers are intended by users and more versatile interfaces are required. In the next chapter some possible alternatives to the desktop metaphor are presented, most of them result not good enough to exceed the *satisficing* attitude but others are emerging as valid alternatives to the desktop interface.

3 Alternatives to the desktop metaphor

In the previous chapter the limits of desktop interface were pointed out, now some alternative interfaces are presented. First are illustrated some examples of interfaces that are not superior enough to completely replace the actual desktop paradigm. Then Zooming Interface Paradigm (ZIP) is explored as an alternative that is slowly gaining ground in user interfaces.

3.1 Project X - HotSauce

During the last decade, Apple and Microsoft started looking at innovative ways of navigating information stored in computers to replace the classical desktop-folders interface.

Project X, also known as HotSauce, is a three dimensional file management interface with a spaceship-videogame style realized at Apple in the mid-1990s. (Fig.8) Folders and files were presented as planes floating in a three-dimensional virtual space. Clicking on a folder, instead of opening a window with the corresponding content, caused the whole screen to zoom-in, revealing files as a group of satellites surrounding their planet. (Johnson 1997, p.80) In HotSauce it was also possible to navigate the file system moving freely around the three-dimensional space. As Johnson (1997) points out, HotSauce gave the idea of really navigating computer

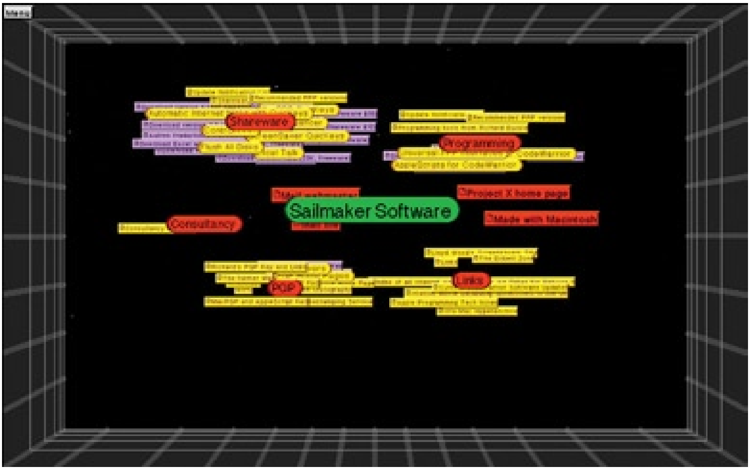


Fig.8 HotSauce screenshot. In HotSauce it was also possible to navigate the file system moving freely around the three-dimensional space.

data as a virtual space, users could remember files structures, thanks to their spatial position instead of remembering folders names or keywords as people use to do in common GUIs. The main problem in HotSauce was the difficulty in moving around in a three-dimensional space. 'I ended up thinking more about how to steer the device than about the data I was looking for'. (Johnson 1997, p.80) Project HotSauce development stopped in 1997.

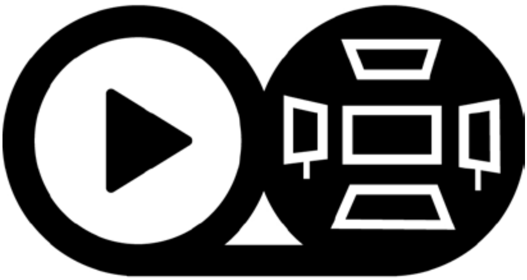
3.2 Task Gallery

In 1999 Microsoft started working on a three-dimensional interface prototype named Task Gallery. Task Gallery is based on an art gallery metaphor. The user can move in a linear three-dimensional hallway. Tasks[1] are arranged on the walls of the gallery. The current task is displayed on a stage in the front wall, while frequently used documents can be recalled from a Start Palette where documents are represented by their actual content, instead of icons. (Robertson et al. 2001) Similarly to HotSauce, Task Gallery is based on the use of spatial memory to easily retrieve documents. Differently from Apple's prototype, the TaskGallery uses a linear hallway to avoid users getting lost in the three-dimensional space. (Video 4)

According to Robertson (Robertson et al. 2001) user studies demonstrated that people found Task Gallery easy to use in navigating the space, finding tasks and switching between them. Even if using a metaphorical visual language, the Task Gallery allows a fast and productive interaction without suffering the problems

1 A task is a group of program windows used in a single job.

Video 4 Task Gallery is an interface based on an art gallery metaphor. Point the mobile phone on the symbol below to watch the video.



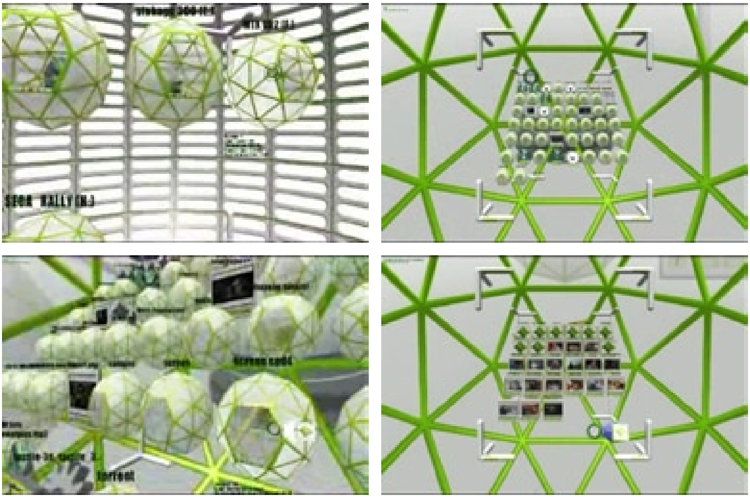
of Magic Cup and Microsoft Bob.[1] However, up until today, Task Gallery has never been implemented in any Microsoft operating system or released as a commercial product. (Microsoft Research 2005, *Microsoft*, viewed 9 October 2008, <http://research.microsoft.com/ui/TaskGallery/>)

3.3 Other interfaces based on three-dimensional graphic

Both Apple HotSauce and Microsoft Task Gallery are three dimensional interfaces. Many of the interfaces that try to replace or improve the desktop metaphor are based on a heavy use of 3D graphic. For example, Tactile 3D[2] is a files navigation system that displays folders and files in a videogame style three-dimensional virtual space. Folders are represented as spheres, to access files users have to get inside the corresponding sphere. Inside they will find classical icons arranged in 3D. Tactile 3D presents the same usability problems of Apple HotSauce. It can be seen more as a funny way to explore the file system, rather than a real productive alternative. (Video 5)

In a similar way, 3DNA Desktop[3] is an application that features customizable three-dimensional environments to replace the desktop. The user navigates in imaginary videogames-like places with a first-person point of view (Doom style) where it is possible to insert shortcuts to applications and documents. 3DNA does not allow to visually navigate the file system and its folders structure, 3DNA turns to be more a game, rather than an application.

1 Cf. paragraph 2.3
2 Tactile 3D is a commercial software released by Upper Bounds Interactive in 2005 http://www.tactile3d.com
3 3DNA Desktop is a commercial software released by 3DNA corp. in 2003 http://www.3dna.net



Video 5 Tactile 3D is a files navigation system that displays folders and files in a 3D virtual space. Point the mobile phone on the symbol below to watch the video.

(Graham N. 2004, Ziff Davis Publishing Holdings, viewed 29 April 2008, <http://www.pcmag.com/article2/0,1759,1593539,00.asp>)

In Project Looking Glass the appearance of windows and desktop environment remains unaltered however, windows can be rotated in all directions, allowing thus to use the back of each window to take notes and to use their sides to read its title when the window is rotated.[1] (Fig.9-10) (Project Looking Glass 2005, *Sun Microsystems, Inc.*, viewed 8 August 2008, <http://www.sun.com/software/looking_glass/index.xml>)

BumpTop is a three-dimensional desktop interface that uses physical simulation. Folders and filling cabinets metaphor are replaced by piling. Icons and windows in BumpTop are three dimensional objects that can be piled to allow a greater number of visible objects on the desktop. (Video 6) Virtual piles should allow to easily remember where an object is, as it happens with real world piles. (Agarawala A. & Balakrishnan R., 2006, ' Keepin' it Real: Pushing the Desktop Metaphor with Physics, Piles and the Pen', *Proceedings of CHI 2006 - the ACM Conference on Human Factors in Computing Systems,* viewed 8 August 2008, <http://bumptop.com/BumpTop.pdf>)

All of the above mentioned examples are based on a heavy use of three dimensional graphic. They start from the assumption that a three-dimensional interface is more powerful and more natural than a two-dimensional one, because it better resembles the real world. Regarding the use of three-dimensional interfaces, Jacob Nielsen (1999) argues that despite general thought, navigating in a three-dimensional space is unnatural

---

1 Project Looking Glass is a three dimensional desktop that can be used on Linux and Windows. The project started in 2004 and is sponsored by Sun Microsystems.

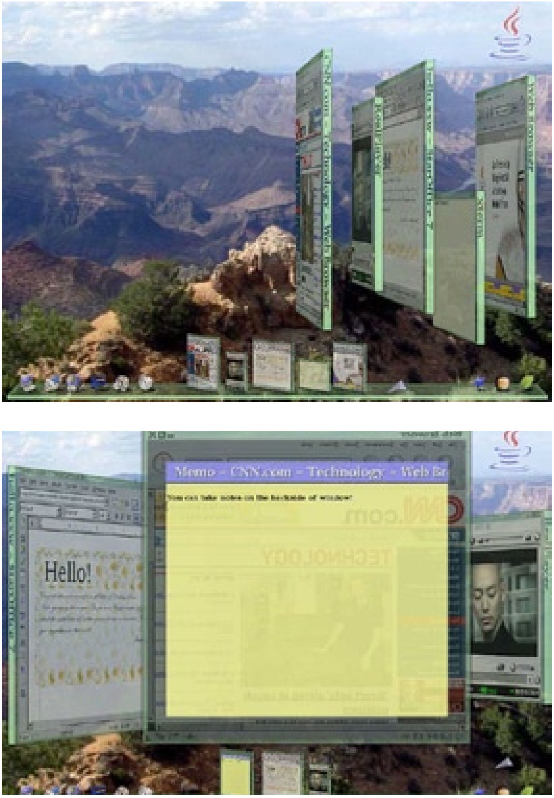


Fig 9-10 Project Looking Glass screenshots. In Project Looking Glass Windows can be rotated in 3D. Windows' sides allow to read title while the back of each window can be used to take notes.

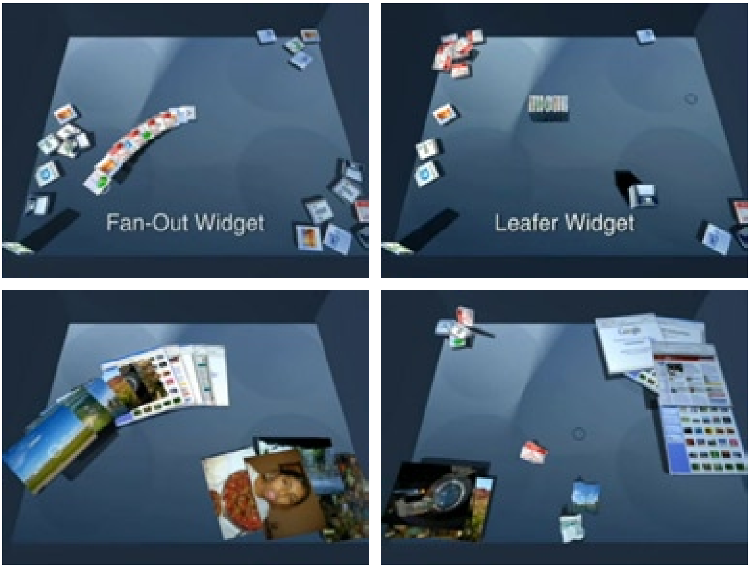
for humans, especially if they float in a free space. Moving on a surface is much easier than in a volume.[1] (Nielsen 1999, p.222) To get oriented in real-world, people always use bidimensional aids; for example, maps and signs are used when going around in a unknown city or museum.

Also Norman (1998) is sceptical about the use of three-dimensional virtual spaces as a solution to interface problems. He points out that usually three-dimensional interfaces supporters confuse vision with space and they overestimate people's spatial abilities: (Norman 1998, p.101)

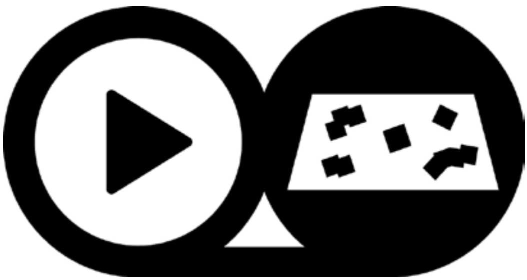
> We are pretty good at remembering physical locations. But good is not perfect. And there are huge individual differences; what is easy for some is difficult for others.
> The real problem, however, is that what many technologists seem to propose as the solution is not true spatial representation. What most technologists seem to want to do is present a picture of a three-dimensional world on the screen, letting us move the picture around so that the scenes that are visible are the ones we would see if we were moving inside the space. But there is no movement; the visual world moves, but we ourselves stand still. That is not at all the same as the real situation in which the world stays still and it is we who move. This is confusing visual images with spatiality.

Some of the main impediments when navigating in three dimensions are the occlusions of information, due to overlapping and

1 The author refers to web interfaces, but his arguments can be applied to computer interfaces in general.



Video 6 Bumptop is a three-dimensional desktop interface based on piling and physical simulation. Point the mobile phone on the symbol below to watch the video.



the difficulties in recognizing distant objects. The third dimension can be valuable only if used in a reasonable way.

3.4 Zooming interfaces

A zooming interface is a graphical user interface where all the information is laid out on a single giant plane and it can be accessed zooming and panning on the plane surface. In a zooming interface there are no windows, icons or folders. All the documents and their content can be accessed zooming. Zooming interfaces relay on human spatial abilities, people can easily remember objects position thanks to spatial references. Spatialization is one of the few universal concepts. (Lakoff & Johnson 1980) It comes from the spatial experience of living in the physical world common to all humans. Most of the other concepts are understood through spatial metaphors. (Lakoff & Johnson 1980, p.56) In current interfaces, the desktop can be used to arrange documents in a visual space, as well, but, due to its small dimensions, most of the documents have to be recalled from menus or hierarchical folder structures.[1] In zooming interfaces an infinite plane allow to arrange all of the documents in a visual way while the zooming tools allows to easily pass from a very wide overview to detailed information.

Pad++

The first implementation of zooming interface is Pad++,[2] an interface framework that provides tools for navigating and editing data

1 Cf. paragraph 2.2

2 Pad++ is a project developed at New York University and University of New Mexico in 1990s.

in a zoomable environment. (Bederson B. et al. 1994, 'Pad++: A Zoomable Graphical Sketchpad For Exploring Alternate Interface Physics', *Symposium on User Interface Software and Technology*, <http://portal.acm.org/citation.cfm?doid=192426.192435>)

The navigation interface allows smooth zooming and panning on the surface. To compare physically distant data in the same screen, Pad++ introduces the concept of portal. A portal is an item that provides a view of an area or a surface different from the current one. Portals also allow to change the way objects are presented and to filter data. Semantic zooming changes the way objects appear, depending on the zoom level: instead of showing a scaled version of the object, a different representation is displayed. For example, a digital clock can show hours and minutes at its normal size; zooming-in, instead of displaying a bigger text size, can add seconds and date while, when zooming-out, it can show only hours at a readable size. (Ibid.) Textual searches are linked to the visual interface through animations showing the path from the actual view to the place where the searched element is. Pad++ authors point out that their aim is to replace old interface concepts, based on mimicking reality, with an interface based on simulation of physics behaviours but free from metaphors. This way of thinking led the authors to use a real world inspired zoom mechanism using behaviors, such as Semantic zooming, that move away from physical resemblance. Even if Pad++ uses metaphors to explain its interface elements, the metaphors used are detached from a physical reality, allowing an higher level of abstraction and better exploiting the potentials of digital tools.

ZoomWorld

Jef Raskin[1] contributed a lot to zooming interfaces. The author (Raskin, 2000) describes actual computer interfaces with windows, menus, icons as mazes made of little rooms with doors that do not give clear cues about where they lead. Raskin (2000) identifies the solution in an interface where it is possible to see all the data and the paths leading from a place to another. The author (Ibid.) describes the Zooming Interface Paradigm (ZIP) as the possibility to fly over the maze allowing a clear overview. Raskin's interface is named ZoomWorld and it is based on a infinite plane of information where everything needed is displayed, whether it is on the local disk or on a network. Zooming is like flying high or diving down to see details. In ZoomWorld there is no need for icons, documents are already open on the infinite plane. To access their content, users only need to zoom in. In ZoomWorld there are no imposed structures for organizing files. Raskin (2000, p.154) suggests using size differences to create hierarchies; important information can be bigger in size and less important information smaller. Also, links can be repleaced by zooming; for example, a footnote, can display the entire referenced work at a very small size, instead of reporting a link to the source. To compare distant documents, ZoomWorld features splitscreens with independent control in a similar way to Pad++ Portals. (Video 7)

Raskin realized a ZoomWorld application to computerize a big sized medical chart used in hospital's intensive care units (ICU) (Raskin 2000, p.157). Hospitals are particularly good testbeds for information technology systems; due to the critical

1 Jef Raskin is one of the designers who started the Apple Macintosh project.

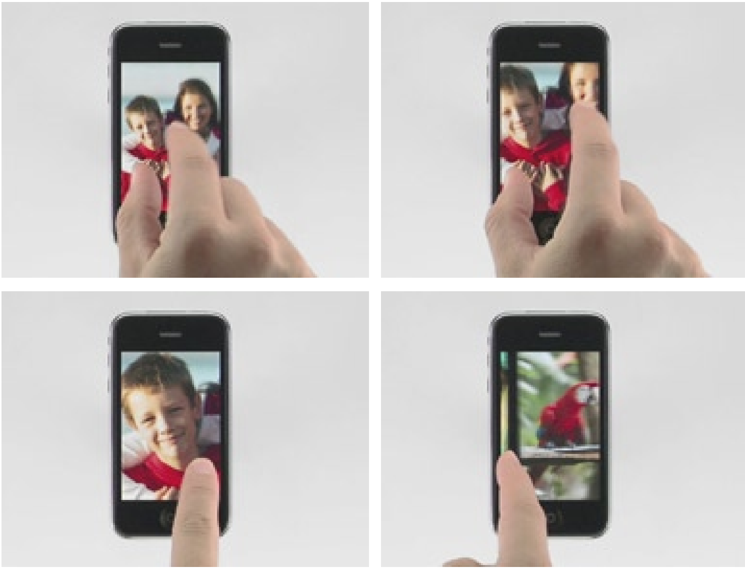
Video 7 Raskin's Zooming User Interface demo. Point the mobile phone on the symbol below to watch the video.

situation, information access has to be reliable, fast and versatile.[1] ZoomWorld is used to provide all the information of the ICU chart and also to give access to different types of data on the enterprise database. The zooming interface allows to easily pass from a very detailed level of information, details about a single patient, to the floor level, where rooms information is layed resembling the actual floor map. Zooming out further gives access to the map of all the hospitals network, zooming on another building allows to access information of other hospitals. Within the same interface, zooming out further gives access to the World Wide Web. During user studies, all the other digital interfaces tested proved to have a slower interaction than paper and needed excessive training, while ZoomWorld proved that nurses where able to use the system after just one minute of training, taking advantage of all the benefits of the digital interface. (Raskin 2000)

Zoomable maps interfaces

In the example above a zooming interface is applied to a big screen context, but zooming interfaces paradigm is versatile enough to be successfully adopted in different sizes devices. Some examples show how zooming interfaces work fine in standard environments or with small screen devices. For example zooming interface has become the standard for navigating digital maps. Software such as Google Maps demonstrate how high resolution satellite images can be explored in an easy and fast way, thanks to the zoom and pan interface. Such interface makes millions of streets and landmarks accessible in few seconds. It is

1 Cf. Bardam J. 2007 *From Desktop Management to Ubiquitous Activity-Based Computing.*



Video 8 Thanks to its multi-touch screen technology, Iphone makes zooming and panning operations very easy. Point the mobile phone on the symbol below to watch the video.

easy to imagine how difficult to explore maps using a standard windows and folders interface would be: maps would be hidden in a hierarchical structure made of folders and subfolders representing continents, countries, cities and streets.

Iphone

Apple Iphone, even if it does not use the zooming interface paradigm, can give an idea of how zooming interfaces could be effectively implemented on mobile devices. Thanks to its multitouch screen technology, Iphone makes zooming and panning operations very easy. Iphone gives the possibility to explore geographical maps, picture galleries, web pages in a very easy way, thanks to its zooming functions. To zoom a map or a picture users just press the screen with two fingers and spread them farther apart or closer together. To pan users touch the screen and drag one finger on the surface: the displayed content will move accordingly. Apple Iphone operating system does not follow the principles of zooming interfaces: screens slides in and out as they were on an infinite plane, but there is no consistency in the absolute position of screens and it is not possible to zoom out to have an overview of the entire system. (Video 8)

Jeff Han's Multi-Touch Interaction Research

An example of how zooming interfaces can be used in big screen devices comes from Jeff Han's *Multi-Touch Interaction Research.* (Han J 2006, *Multi-Touch Interaction Research*, viewed 11 august 2008, <http://www.cs.nyu.edu/~jhan/ftirtouch/>) Han demonstrated, on a 1 meter large display, how simple and natural navigating a map or browsing pictures is with a zooming interface

Video 9 Jeff Han’s Interface is an example of zooming interface on a big screen. Point the mobile phone on the symbol below to watch the video.

on a multi-touch screen, using hand gestures to pan and zoom. Han interaction scheme is very similar to the Iphone's, that demonstrate how zooming interfaces can be successfully applied in tools of different sizes maintaining consistency.[1] (Video 9)

Conclusions

The few examples presented demonstrate how zooming interfaces can be adopted in devices of different sizes, allowing greater consistency between tools that are becoming more and more interconnected in everyday life. Zooming interfaces can be the answer to needs that desktop interface cannot satisfy, such as easily storing and retrieving thousands of documents using more widespread metaphors.

Current operating systems present some features that can be seen as little steps towards zooming interface direction. Some systems, for instance, give the possibility to switch between multiple desktops,[2] where to store more documents in a visual way, while *Exposè* arranges all the open windows in a single view, scaling them down creating a zoom-out effect.[3]

From a hardware point of view, standard mice with a scroll-wheel are good devices to interact with zooming interfaces, but new input devices are emerging. Some laptop computers implement multi-touch capabilities on the touchpad,[4] allowing

---

1 Jeff Han's *Multi-Touch Interaction Research* was presented one year before Apple Iphone.

2 The possibility to have multiple desktops is available on Unix and Linux since 1980s, it was implemented as a standard feature of Mac OS X in 2007.

3 Exposè is a feature of Mac OS X.

4 Today only MacBook laptops and Asus eee PCs

interaction capabilities similar to Iphone. This trend could bring to an increasing use of zooming interfaces.

OLPC case study is the subject of the next chapter. OLPC is a project presenting a brand new interface for a laptop, giving up the desktop metaphor. Due to the unusual context,[1] more widespread spatial metaphors are used, including a zooming metaphor rendered in a completely different way from what seen in this chapter.

1 OLPC project is addressed at children without any knowledge of computing.

4 One Laptop per Child case study

4.1 The One Laptop Per Child project

The One Laptop Per Child (OLPC) is a non-profit organization founded by MIT professor Nicholas Negroponte in 2005. (*One Laptop Per Child Vision* n.d., *One Laptop Per Child*, Cambridge, viewed 8 October 2008, <http://laptop.org/vision/index.shtml>) The aim of the OLPC project is to fill the socio-economic gap of developing countries thanks to the help of information and communication technologies. OLPC short term objective is to provide one personal laptop to any child as a learning resource.

OLPC laptop user interface was designed from scratch by the project team.[1] Developers took the bold decision to avoid the use of the desktop metaphor. As it was pointed out,[2] the desktop metaphor is not suitable for people not working in a office.

1 In this dissertation it is taken into account the original laptop interface (named *Sugar*) designed by the OLPC team and based on Linux, however OLPC laptops can run Windows XP as well. (Barak S. and Cassia F. 2008, *Peru lures children into the world of Windows*, *The Enquirer London*, viewed 8 October 2008, <http://www.theinquirer.net/gb/inquirer/news/2008/09/17/peru-first-corrupt-children>).

2 Cf. paragraph 2.2

Fig.11 Children in Nepal using XO laptops.

Moreover, OLPC laptop users are children without any knowledge of computing, they cannot take advantage from the metaphor nor from previous experience with computers. These reasons make the OLPC project an ideal situation where experimenting different interface solutions without taking into account people's satisficing behaviour.[1] (Fig.11)

One Laptop Per Child concept is based on the theories of constructionism elaborated in the 1960s by Seymour Papert. Constructionist learning is based on the principle that learning is an active process. Learners build mental models and theories of the world around them. Papert thinks that computers are particularly suitable for learning math and logic when using programming environments devoted to education, such as LOGO. (Harel & Papert 1991)

In Negroponte's vision, a laptop computer, especially designed for children, gives them the opportunity to access learning resources and information on the Internet, to develop their own creativity, problem solving potentials and to become part of a worldwide community of peers. (*One Laptop Per Child Mission* n.d., *One Laptop Per Child*, Cambridge, viewed 8 October 2008, <http://laptop.org/vision/mission/>)

The choice of laptop computers, instead of school based computers, comes from the idea that the computer will be a replacement for books nowadays too expensive for developing countries' financial possibilities. A laptop can be brought home, thus becoming a resource for families. Moreover it can become the only educational opportunity for children that have any possibility to attend school. (Video 10)

1 Cf. paragraph 2.4

Video 10 Nicholas Negroponte speaks about One Laptop per Child on CBS program *60 minutes*. Point the mobile phone on the symbol below to watch the video.

4.2 OLPC laptop design

Developing the OLPC laptop was a huge design challenge. The laptop needs to be strong and safe enough to be suitable for the target users (6 to 16 years old children); it has to be water proof and dust resistant to be used in extreme conditions; it also has to be easy to use, to require low maintenance and to work in places where electric energy is not a common facility. A lot of attention was also put in communication capabilities, the possibility of connecting to the Internet, or to a local area network where no Internet connection is available. Thus was one of the priorities. Another fundamental issue was the price. The OLPC computer has to be cheap enough to allow developing countries governments to afford buying a computer for any child. The project is also known as the "$100 laptop".[1]

Due to its peculiarity, the overall design of the OLPC laptop will be now illustrated, while the next paragraph will deal with innovations in user interface design.

The look of the case is very playful. The laptop is white with a green frame, on the top cover there is a coloured logo representing a joyful child stylized as a big "X" representing a child body with open arms and a small "o" as head. From that logo derives the name of the machine, called the XO laptop.

XO has a big handle integrated in its mould to be carried easily without any bag. Another very remarkable part of the case is represented by the two WI-FI antennas that come out as ears.

1 The current price of the laptop is higher than $100 but, according to Negroponte, the price should lower to $100. (*One Laptop Per Child* n.d., One Laptop Per Child, Cambridge, viewed 8 October 2008, <http://laptop.org/en/laptop/hardware/features.shtml/>)

Video 11 David Pogue reviews XO laptop. Point the mobile phone on the symbol below to watch the video.

The choice of a very recognizable and playful appearance has two goals: to appeal to children that have to experience the laptop as an engaging learning tool and to discourage improper usage as reselling them through the grey market. (*One Laptop Per Child Features* n.d., *One Laptop Per Child*, Cambridge, viewed 8 October 2008, <http://laptop.org/en/laptop/hardware/features.shtml/>)

The screen is a dual-mode 7.5 inches LCD. It can be used in standard colour mode or in high resolution (200 dpi) monochrome mode. The second modality features high contrast to allow readability under sunlight and low power consumption. It also preserves actual content when the laptop is turned off. The pointing device is a wide touchpad that can be used with fingers or with a pen to draw. The XO features multimedia capabilities, thanks to stereo speakers, an integrated microphone and a webcam. (Video 11)

4.3 XO user interface

Much attention was also put in user interface design. Starting from the assumption that target users are children from developing countries, it seemed of no benefit for developers to use metaphors such as office and desktop and any reference to existing operating system seemed of any help. The OLPC software design team took the hard decision to design a brand new user interface; some of the established computer concepts were changed in order to be more understandable for the target audience. The concept of *software application* was replaced by *activity* to focus on childrens' expressions and collaboration; every activity takes advantage of the peer-to-peer mesh network that connects XOs to each other. The exchange of ideas between peers is seen as the best way to obtain engagement and to stimulate critical thinking.

According to constructionist learning theories, every child has to be considered as a learner and a teacher at the same time. (*One Laptop Per Child Mission* n.d., *One Laptop Per Child*, Cambridge, viewed 8 October 2008, <http://laptop.org/vision/mission/>)

The concept of *file* is replaced by the one of *object*: every *activity* done by the user produces *objects*. In object creation real-world metaphors are used; a *text file* becomes a *story*, a *graphic file* a *drawing*, an *audio file* an actual *sound*. The objects in file-system are organized using a journal metaphor. The concept of journal as a written record of daily activities seems to be generally understood by different cultures. (*OLPC Human Interface Guidelines* 2008, *One Laptop Per Child*, viewed 8 October 2008, <Cambridgehttp://wiki.laptop.org/go/OLPC_Human_Interface_Guidelines>) The filesystem records all the activities the child has done, as well as, the ones he took part in with other children. The objects are organized in a chronological way. Objects can also be arranged and searched using keywords, but OLPC designers argue that chronological order is the most natural way to search through objects. The filesystem structure has to be seen as a portfolio or the scrapbook history of child interaction with laptop and with friends. (*OLPC Human Interface Guidelines* 2008, *One Laptop Per Child*, viewed 8 October 2008, <Cambridge http://wiki.laptop.org/go/OLPC_Human_Interface_Guidelines>)

Since the laptop is intended for young users with no previous experience with computers and Information Technologies, developers had more freedom in exploring new concepts and metaphors. The drawback is the lack of a common cultural background usually adopted in interface design.

The XO has to transcend a broad range of cultures and languages. XO developers were conscious that most of the concepts and metaphors used in interfaces are familiar only in western countries. For example, from the western culture point of view, using a photo camera icon could seem the best way to label the camera function of the laptop. However considering that most of the children that use the XO laptop have never used or seen a photo camera, that label will be meaningless to them. Metaphors referring to human body are universal (Lakoff & Johnson 1980) so, they better accomplish their role in OLPC context. The webcam can be described as the laptop eye and an eye icon can be used to label it. (*OLPC Human Interface Guidelines* 2008, *One Laptop Per Child*, viewed 8 October 2008, <http://wiki.laptop.org/go/OLPC_Human_Interface_Guidelines>

The XO low price goal determined lower computing speed compared to standard laptops. Only a limited number of activities can run at the same time, this detail is exposed to users in a direct way. Running activities are represented by their icons inside a ring, the size of the ring segment each activity occupies represents memory usage. This allows children to build a correct mental model of how the machine works, what activities can be opened and how to act when there is no more space to open new activities.

Another decision taken to prevent speed problems is the removal of multi-tasking and stackable windows. Activities take all screen space. It is not possible to display more than one activity at the same time. Other then a technical issue this limitation comes from the decision to focus user's effort on one task only to avoid distraction in children. (*OLPC Human Interface Guidelines* 2008, *One Laptop Per Child*, viewed 8 October 2008, Cambridge <http://wiki.laptop.org/go/OLPC_Human_Interface_Guidelines>)

As previously said, the desktop metaphor was totally discarded by XO interface designers. XO main user interface is made of four views, bound together by a zoom metaphor. The four

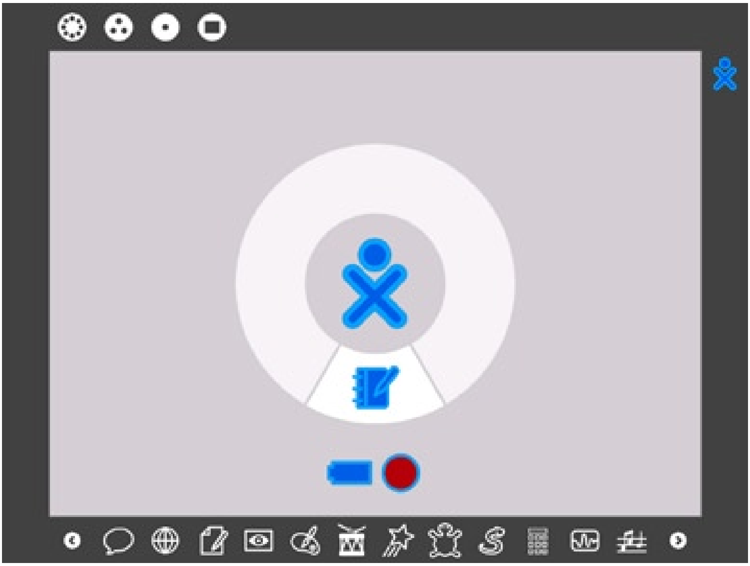


Fig.12 XO GUI home view screenshot with the activity ring.

views represent children's environment from the current activity running on the laptop to the whole network. The 4 levels are named *Home*, *Groups*, *Neighbourhood*, and *Activity*.

The first level is the Home screen. The Home screen features an XO icon representing the child, whom the laptop belongs to, at the centre of the screen. The colours of the icon are chosen by the laptop owner and are the same colours used to represent the laptop on the network. The character is surrounded by the activity ring. The home view is used to start, end or switch between activities. (Fig. 12)

The Groups view is at an outer level of zoom from the Home view. It is used to see groups of friends, classmates or other groups the user belongs to. The network is represented as the group of children connected to it. Children are represented by a coloured XO icon, different colours are used to identify different children. Children's icons can be spatially rearranged to create logical clusters. The class group is automatically created, based on the children attending the same class. Children have the possibility to freely create other groups. People in a group share a bulletin board, a chat and group activities. In Groups view it is also possible to see what activity other children are currently engaged in.

The Neighbourhood is the most external zoom level, it represents all the computers in the same mesh, (usually a school or part of it). The Neighbourhood is the place where to meet new friends. In the Neighbourhood view it is possible to see what activity other students are taking part in and join them. Children in the Neighbourhood view are visually arranged around their activity to give an overview of activities popularity. (Fig.13)

The inner zoom level is the Activity view. The Activity view is the equivalent of the software window in common operating

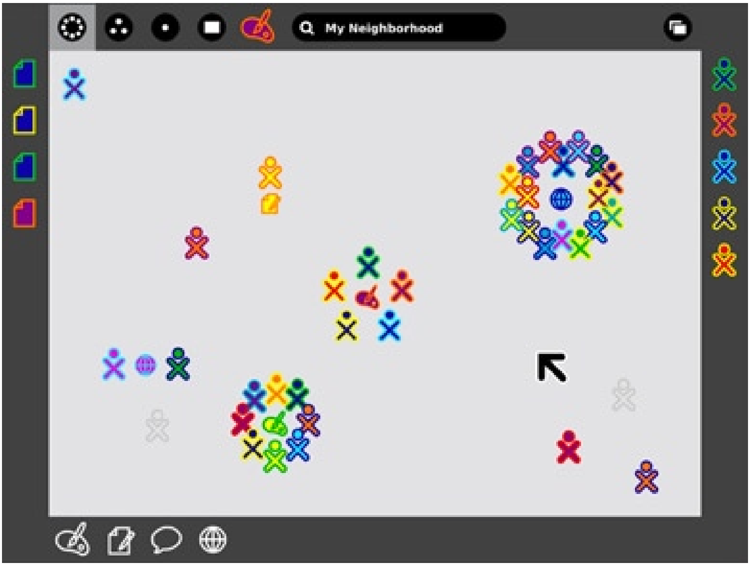


Fig.13 XO Neighbourhood view screenshot with active frame.

systems. It is the place where actual creation, exploration, and collaboration take place. Its appearance changes depending on the current running activity.

Other main elements of the XO interface are the *Frame*, the *Bulletin Board* and the *Journal*. The Frame is an area that pops-up when activated by moving the mouse cursor near screen corners. The Frame presents the same information at all zoom levels and it is used to display places, people, objects and actions. It also features a search bar. Detailed information about users, whom the child is sharing an activity with, are dislpayed in the Frame. It, also, gives notifications and shows incoming invitations. The bottom part is an application launcher with an iconic list of all the activities. The left side is an extended clipboard where it is possible to drag and drop objects that the child wants to pass from an activity to another or that he wants to share with other children.

The Bulletin Board is the place where to post objects and ideas. It is a contextual space and it changes based on what zoom level the child actually is in. The Bulletin Board gives different opportunities of interaction among users. In the Home view, it becomes a way of keeping notes for oneself.

Inside the Bulletin Board there is a spatial chatting interface and a file transfer facility. Chat windows can be spatially positioned by the children, giving a higher level of communication among users. Chat windows can also be used to point at some details in the interface and can explain how a specific function works. It can, also, be used by a teacher in proofreading to highlight errors and corrections in real time. (*OLPC Human Interface Guidelines* 2008, One Laptop Per Child, viewed 8 October 2008, <http://wiki.laptop.org/go/OLPC_Human_Interface_Guidelines>)

The Journal is the filesystem explorer. Concepts of filesystem, folders, save and open are all discarded in XO laptop. Designers decided to use time as the main way of retrieving activities. Children do not need to worry about saving files and choosing hierarchical paths for storing the results of their activities. The system automatically takes track of different states of an activity, enabling users to easily find them using time as a reference. Referring to natural capacity of human memory, the system maintains different states of recent activities, while an "intelligent algorithm" suggests what activity can be cancelled to keep the journal tidy and preventing the system to run out of memory. (*OLPC Human Interface Guidelines* 2008, *One Laptop Per Child*, viewed 8 October 2008, <http://wiki.laptop.org/go/OLPC_Human_Interface_Guidelines>)

4.4 Criticisms

As some reviewers point out,[1] the XO interface is discouraging for people used to standard operating systems. There are no roll-down menus, no windows, no Start button and no desktop. The icons in some cases are very different from current operating systems conventions.

Some reviewers[2] think that the radical choice made by OLPC designer is unacceptable. They argue that using a totally different paradigm is an error, as it means to ignore what has been developed and improved in the last 30 years. The simplicity at the base of XO interface, the lack of menus and sub-menus or folders

1 Cf. *One Laptop Per Child, il tour guidato* 2007, *De Andreis Editore*, viewed 8 October 2008, <http://punto-informatico.it/p.aspx?i=2011294>

2 Cf. Thom Holwerda 2006, *OSNews LLC*, viewed 8 October 2008, <http://www.osnews.com/story/16582/The-OLPC-Sugar-Interface-Dont-Do-it>

and sub-folders, is seen as an underestimation of developing countries children abilities (Thom Holwerda 2006, *OSNEWS LLC*, viewed 8 October 2008, <http://www.osnews.com/story/16582/The-OLPC-Sugar-Interface-Dont-Do-it>). Actually, there are evidences that children in poor and rural regions of the world are able to use computers as easily as all other children.[1]

This conservative assumptions are based on the already mentioned inertia that makes people prefer to relate with what they already know and consider familiar. The fact that learning a system based on desktop metaphor is not difficult even for children who have no previous experience with real world counterpart, does not prevent from searching for better conceptual models. As a matter of facts, early testing with XO demonstrated that children take just few minutes to learn how to use it. (One Laptop Per Child 2007, video recording, *CBS NEWS*, <http://www.cbsnews.com/video/watch/?id=2830221n>)

In a scenario where interfaces remained unaltered for almost 30 years, it can be difficult to imagine alternative solutions to desktop metaphor, but OLPC laptop is a good example of how changes are possible.

1 That emerged from a pilot project conducted by Negroponte in 1982 using Apple II computers. (*One Laptop Per Child Progress* n.d., *One Laptop Per Child*, Cambridge, viewed 8 October 2008, < http://laptop.org/en/vision/progress/index.shtml/>)

Conclusions

The previous chapters illustrated the problems arising when people have to communicate with computers. Communication with computers is a mediation between two entities that use a completely different language. The place where these two entities meet is the interface. Human-computer Interface designers have to find good ways to make communication successful and smooth.

At the very beginning of computing, in the 1950s, machines were really difficult to use. A lot of effort and knowledge was needed. Computers were mainly used in research laboratories and universities. Through the years computers became more and more powerful and versatile. More people started benefiting from computers aid, interfaces became more user friendly and allowed to accomplish tasks never thought possible before. At the end of the 1970s, computers became cheap enough to be affordable for consumers, but, due to their technical issues, they were used only by experts and enthusiasts that had the will to learn difficult commands and sequences to obtain only few benefits. The real turning point arrived in the mid-1980s, when the advent of the GUI (Graphical User Interface) allowed a wider audience to easily obtain interesting results with much less effort and at a decreasing cost. Since then, the IT market has always been very lively. Innovations and new technologies are introduced at a fast pace: components prices go down, computer efficiency increases, new products are presented every day and new possible uses arise. Nowadays, computers are used, besides for productivity, also to communicate through the Internet, to store pictures, watch movies, listen to music, play videogames. Although many changes are occurring, human computer interfaces have not changed much since the introduction of the GUI and the desktop metaphor.

Is the 30-year-old desktop metaphor interface so suitable that no other innovation is needed? There are multiple evidences that the desktop metaphor is not up-to-date with actual users' needs. The reasons for this gap were found in software market and people inertia and in the apparent lack of any alternative, good enough to go over the *satisficing* attitude.[1]

Some alternatives to the desktop interfaces were illustrated, the zooming interface paradigm was presented as a possible valid alternative to current interfaces. Since it uses universal spatial metaphors, it is suitable for a wider range of users. Moreover, it was explained how it can be successfully applied to a different scale: from big screens to small devices.
The last chapter presented the One Laptop Per Child project. The aim of this ambitious project is to massively introduce Information Technologies in developing countries education system. A special laptop is to be produced for the OLPC project scope. OLPC interface designers decided to abandon most of the

1 Satisficing means "satisfying" and "sufficing" Cf. paragraph 2.4

current interface conventions and tried to introduce metaphors and conceptual models more adequate for that brand new target. The decision of changing actual interface standards was criticized by some commentators; at present, there are not enough evidences to judge the success of this operation, but it is an important decision that can lay the basis for a switch in computer interfaces and in the way people think computers in general.

If looking at the past can help in understanding the present, then it is possible to look at early reviews of graphic user interfaces, when they started to reach the market in early 1980s. Reviewers gave a very cold welcome to those new devices. Some reviews discarded the visual language as childish (Johnson 1997, p.54). It seemed a crazy idea that such colourful, drawn interfaces could be used in a working environment. Others thought that the GUI was useful only for artists and designers. The most open-minded were able to understand the power of using the desktop metaphor but were sceptical about the need to tie it with icons and a pointing device. (Ibid. p.55)

Computer users of the early 1980s were used to control their machines with command-line interfaces and they were able to perform a lot of powerful tasks in an easy and fast way, compared to previous technologies. These days nearly all the categories of computer users who have experienced pre-desktop interfaces remember those days as difficult old times. Maybe it will be the same in a few years, when thinking back to the hard times of desktop interfaces.

Bibliography

Books

Barfield, L 1993, *The user interface concept and design,* Addison-Wesley, New York.

Cooper, A & Reimann, R 2003, *About face 2.0: the essentials of interaction design,* Wiley, Indianapolis.

Dertouzos, M 2001, *The unfinished revolution: human-centered computers and what they can do for us*, Harper Collins Publishers, New York.

Dourish, P 2001, *Where the action is: the foundations of embodied interaction*, The MIT Press, Cambridge.

Harel, I & Papert, S 1991, *Constructionism*, Ablex Publishing, New York.

Johnson, S 1999, *Interface culture*, Basic Books, New York.

Kaptelinin, V & Czerwinski, M (eds.) 2007, *Beyond the desktop metaphor*, The MIT Press, Cambridge.

Lakoff, G & Johnson, M 1980, *Metaphors we live by*, University Of Chicago Press, Chicago.

Laurel, B (ed.) 1990, *The art of human-computer interface design*, Addison-Wesley Professional, Boston.

Laurel, B 1993, *Computers as theatre*, Addison-Wesley Professional, Boston.

Macintosh human interface guidelines 1993, Addison-Wesley Professional, New York.

Moggridge, B 2007, *Designing interactions,* The MIT Press, Cambridge.

Nielsen, J 1999, *Design web usability: the practice of simplicity*, New Riders Publishing, Indianapolis.

Norman, D 1998, *The invisible computer: why good products can fail, the personal computer is so complex, and information appliances are the solution,* The MIT Press, Cambridge.

Norman, D 2002, *The design of everyday things*, Basic Books, New York.

Preece, J Rogers, Y Sharp, H Benyon, D Holland, S & Carey, T 1994, *Human computer interaction*, Addison Wesley Publishing Company, New York.

Raskin, J 2000, *The humane interface: new directions for designing interactive systems*, Addison-Wesley Professional, New York.

Sellen, AJ & Harper, R 2001, *The myth of the paperless office,* The MIT Press, Cambridge.

Sharp, H Rogers, Y & Preece, J 2007, *Interaction design: beyond human-computer interaction,* Wiley, New York.

Tidwell, J 2005, *Designing interfaces: patterns for effective interaction design,* O'Reilly Media, Sebastopol.

Electronic sources

Web pages

3dna 2004, *3dna corp.*, viewed 8 august 2008, <http://www.3dna.net>.

Bellis, M 2008, *The New York Times Company*, viewed 29 april 2008, <http://inventors.about.com/od/cstartinventions/a/Apple_Computers.htm>.

Bellis, M 2008, *The New York Times Company*, viewed 4 august 2008, <http://inventors.about.com/od/ mstartinventions/a/Windows.htm>.

Davidson, Dove & Weltz 1999, *Laura Dove*, viewed 29 april 2008, <http://www.lauradove.info/reports/mental%20models.htm>.

Gelernter D, n.d., *Edge*, viewed 29 april 2008, <http://www.edge.org/3rd_culture/gelernter/gelernter_p1.html>.

Graham, N 2004, *Ziff Davis Publishing Holdings*, viewed 29 april 2008, <http://www.pcmag.com/article2/0,1759,1593539,00.asp>.

Han, J 2006, *Jeff Han*, viewed 11 august 2008, <http://www.cs.nyu.edu/~jhan/ftirtouch/>.

Holwerda, T 2006, *OSNews LLC*, viewed 29 april 2008, <http://www.osnews.com/story/16582/The-OLPC-Sugar-Interface-Dont-Do-it/>.

Johnson, J Roberts, TL Verplank, W Smith, D Irby, C Beard, M & Mackey, K 1999, *David Curbow*, viewed 29 april 2008, <http://www.digibarn.com/friends/ curbow/star/retrospect/>.

Johnson, S 2002, *Washington Post. Newsweek Interactive*, viewed 29 april 2008, <http://www.slate.com/id/2075219/>.

Lund, A & Waterworth, J n.d., *Umeå university*, viewed 29 april 2008, <http://www.informatik.umu.se/~jwworth/MetaDesign.html>.

Norman, D 2004, *Don Norman*, viewed 29 april 2008, <http://www.jnd.org/dn.mss/affordance_conv.html>.

OLPC Human Interface Guidelines 2008, *One Laptop per Child association*, 29 april 2008, <http://wiki.laptop.org/go/OLPC_Human_Interface_Guidelines>.

One Laptop Per Child, il tour guidato 2007, *De Andreis Editore*, viewed 29 april 2008, < http://punto-informatico.it/p.aspx?i=2011294>.

One Laptop per Child (OLPC), Laptop: a learning tool created expressly for the children in developing nations 2007, viewed 29 april 2008, <http://laptop.org/en/laptop/hardware/features.shtml>.

One Laptop per Child (OLPC), Progress: Discover the origins of OLPC 2007, viewed 29 april 2008, <http://laptop.org/en/vision/progress/index.shtml>.

Palo Alto Research Center 2008, *Encyclopædia Britannica*, viewed 8 October 2008, <http://www.britannica.com/EBchecked/topic/440290/Palo-Alto-Research-Center>.

PARC innovation milestones 2007, *PARC*, viewed 29 april 2008, <http://www.parc.com/ about/ history>.

Project Looking Glass 2005, Sun Microsystems, Inc., viewed 8 august 2008, <http://www.sun.com/software/looking_glass/index.xml>.

Raymond, ES & Landley, RW 2004, *The art of unix usability*, viewed 29 april 2008, <http://www.catb.org/~esr/writings/taouu/html/index.html>.

Raskin, J 2005, *Jef Raskin*, viewed 29 april 2008, <http://jef.raskincenter.org/humane_interface/summary_of_thi.html>.

Reimer, J 2005, *Ars Technica*, viewed 4 august 2008, <http://arstechnica.com/articles/paedia/gui.ars/1 >.

Siracusa, J 2005, *Ars Technica*, viewed 5 august 2008 <http://arstechnica.com/reviews/os/macosx-10-4.ars/1>.

Steinberg, S 2004, *Wired Digital*, viewed 29 april 2008, <http://www.wired.com/wired/archive/5.02/fflifestreams_pr.html>.

Tactile 3d interface 2007, *Upper Bounds Interactive Inc.*, viewed 8 august 2008, <http://www.tactile3d.com>.

The TaskGallery 2005, *Microsoft Corporation*, viewed 8 august 2008, <http://research.microsoft.com/adapt/TaskGallery/index.htm>.

Tristram, C 2001, *Massachusetts Institute of Technology*, viewed 29 april 2008, <http://www.technologyreview.com/Infotech/12703/>.

Tufte, E 2008, *Edward R. Tufte*, viewed 14 october 2008, <http://www.edwardtufte.com/bboard/q-and-a-fetch-msg?msg_id=00036T>.

Wichary, M 2004, *Marcin Wichary*, viewed 29 april 2008, <http://www.aresluna.org/attached/computerhistory/articles/macintosh20yearslater>.

Conference papers

Agarawala, A & Balakrishnan, R, 2006, 'Keepin' it real: pushing the desktop metaphor with physics, piles and the pen', *Proceedings of CHI 2006 - the ACM Conference on Human Factors in Computing Systems,* viewed 8 august 2008, <http://bumptop.com/BumpTop.pdf>.

Bederson B., Hollan J., Perlin K., Meyer J., Bacon D. & Furnas G. 1994, 'Pad++: a zoomable graphical sketchpad for exploring alternate interface physics', *Symposium on User Interface Software and Technology Proceedings of the 7th annual ACM symposium on User interface software and technology,* viewed 8 August 2008, <http://portal.acm.org/citation.cfm?doid=192426.192435>.

Videos

Anand Agarawala: BumpTop desktop is a beautiful mess 2007, video recording, TED conferences, Long Beach. Viewed 8 August 2008, <http://www.ted.com/index.php/talks/anand_agarawala_demos_his_bumptop_desktop.html>.

Douglas Engelbart: The Mother of All Demos 1968, video recording. Viewed 8 August 2008, <http://uk.youtube.com/watch?v=JfIgzSoTMOs>.

Iphone 3G. A guided tour 2008, video recording, Apple Inc., Cupertino. Viewed 8 August 2008, <http://www.apple.com/iphone/guidedtour/>.

Jeff Han: Unveiling the genius of multi-touch interface design 2006, video recording, TED conferences, Long Beach. Viewed 8 August 2008, <http://www.ted.com/index.php/talks/jeff_han_demos_his_breakthrough_touchscreen.html>.

Mac OS X Leopard. A guided tour 2008, video recording, Apple Inc., Cupertino. Viewed 8 August 2008, <http://www.apple.com/macosx/guidedtour/>.

One Laptop Per Child 2007, television program, *60 minutes*, CBS NEWS, New York, 26 August. Producer Catherine Olian. Viewed 8 August 2008, <http://www.cbsnews.com/video/watch/?id=2830221n>.

One Laptop Per Child 2007, video recording, The New York Times Company, New York, 4 October. Producer David Pogue. Viewed 8 August 2008, <http://www.nytimes.com/2007/10/04/technology/circuits/04pogue.html?partner=permalink&exprod=permalink>.

The Knowledge Navigator 1987, 35mm film, Jane Hernandez, San Francisco, Apple Computer, viewed 8 August 2008, <http://www.billzarchy.com/clips/clips_apple_nav_navigator.htm>.

The Star user interface: an overview 1984, video recording, Xerox Corporation. Viewed 8 August 2008, <http://uk.youtube.com/watch?v=XQ6ng1_TMN4>.

The Task Gallery 2005, video recording, Microsoft Corporation. Viewed 8 August 2008, <http://research.microsoft.com/ui/TaskGallery/pages/video.htm>.